Silesian
University
of Technology

DOCTORAL THESIS

Development of Algorithms for Automatic Detection of Eye Movement Signal Events

Birtukan Adamu Birawo
Student identification number: 4952

Discipline: Information and Communication Technology
Specialization: Informatics

SUPERVISOR
Pawel Kasprowski, PhD, DSc
Department of Applied Informatics
Faculty of Automatic Control, Electronics and Computer Science

Gliwice 2026

Declaration

I, Birtukan Adamu Birawo, declare that this thesis titled, "Development of Algorithms for Automatic Detection of Eye Movement Signal Events", and the work presented in it are my own. I confirm that this work was conducted entirely or primarily during my candidature for a research degree at Silesian University of Technology. Any part of this thesis previously submitted for a degree or other qualification at this or any other institution has been clearly identified. All references to the published work of others are properly attributed and clearly cited. All quotations from the works of others are clearly identified, and the sources are appropriately credited. Except for these quotations, this thesis is entirely my original work. I have acknowledged all major sources of assistance and support. For work conducted jointly with others, I have explicitly stated what contributions were made by others and what I contributed myself.

Author's Signature: _____

Date: _____

Dissertation Copyright Authorization

I hereby grant authorization for this thesis to Silesian University of Technology. The University and Library are permitted to incorporate, duplicate, and use it in various formats (paper, CD-ROM, other digital media) without restrictions on location, duration, or frequency. In accordance with the Copyright Law, users are permitted to search online, read, download, or print the thesis.

Acknowledgement

Completing this PhD journey has been one of the most challenging and rewarding experiences of my life. It would not have been possible without the steadfast support, guidance, and encouragement of many remarkable individuals, to all of whom I express my sincere gratitude.

First and foremost, I would like to express my deepest appreciation to my supervisor, **Dr. hab. inż. Paweł Kasprowski, PhD, DSc,** for his invaluable guidance, patience, and expert advice throughout this journey. His dedication and thoughtful mentorship helped me navigate even the most challenging phases of this research. I am especially grateful for his consistent attention to my progress and for his insightful questions, which strengthened this thesis and taught me to approach problems with clarity, rigor, and confidence. Beyond academic guidance, his understanding and support during moments of personal difficulty reminded me that outstanding mentorship extends beyond research alone. Much of what I have achieved is due to his guidance, for which I am profoundly thankful.

I am deeply grateful to my family for being my constant source of strength and stability. Your sacrifices, both large and small, and your unwavering belief in me made it possible for me to pursue this path. Mom, your prayers sustained me during moments of exhaustion. This thesis is dedicated to you, Dad, whose early encouragement instilled in me the value of education and whose belief in me continues to guide me, even in your absence. I am also deeply thankful to my siblings for believing in me and for looking up to me; knowing that I could be a source of inspiration for them motivated me to push forward, even when the journey felt overwhelming.

I would also like to thank my friends for their encouragement, shared perseverance, and meaningful discussions. This journey would have been far lonelier without your support.

I am thankful to the Silesian University of Technology for providing the resources, infrastructure, and intellectual environment that enabled this research. Special thanks go to my department for its financial support, which allowed me to focus fully on my doctoral work. I am also sincerely grateful to the participants in my study for trusting me with their experiences and insights; your contributions form the foundation of this thesis.

Finally, I extend my gratitude to all those who supported me indirectly, teachers who sparked my interest in this field, mentors who shaped my thinking, and individuals

whose kindness offered encouragement during challenging moments. This achievement is not mine alone; it reflects the collective support, wisdom, and generosity of all who accompanied me along this journey.

Abstract

Eye movement analysis plays a critical role in understanding human visual attention, perception, and cognition. The accurate detection of fundamental eye movement events: fixations, saccades, smooth pursuits (SPs), and post saccadic oscillations (PSOs) is essential for reliable interpretation of eye-tracking data across fields such as psychology, neuroscience, and human computer interaction. Event detection, however, remains challenging due to the variability of gaze patterns across tasks, the inherent difficulty of distinguishing short duration events such as PSOs, and the lack of standardized evaluation protocols. This thesis addresses these challenges by conducting a comprehensive investigation of event detection methods, spanning threshold-based, machine learning, and deep learning models, with a particular focus on generalizability across datasets, feature selection, and the methodological importance of PSO detection.

The work begins by systematically evaluating a wide range of event detection algorithms using high-resolution eye tracking data. Traditional threshold-based methods such as IVT and IDT were benchmarked against machine learning approaches (Random Forest) and deep learning models (CNN and LSTM). Unlike prior comparative studies that often used different datasets or evaluation criteria, all algorithms here were evaluated under identical conditions using manually annotated ground truth labels and consistent performance metrics. Results confirmed that threshold-based methods are efficient for binary classification of fixations and saccades but are highly sensitive to parameter tuning and unable to generalize well. In contrast, machine learning and deep learning methods achieved superior robustness and higher agreement with human coders, with CNNs showing the best overall accuracy. These findings establish a clear methodological baseline for the advantages of data-driven models while also demonstrating the critical role of evaluation methodology particularly cross-validation strategies—in producing reliable results.

Building on this foundation, the thesis proposes a hybrid 2D-CNN-LSTM network for simultaneous classification of fixations, saccades, PSOs, and SPs. The CNN layers act as spatial feature extractors, while the LSTM layers capture temporal dependencies, making the architecture well-suited for sequential gaze data. To assess the impact of input representation, four kinematic features: velocity, acceleration, jerk, and direction were systematically combined into feature sets (VD, VAD, VJD, VAJD). The analysis revealed that classification performance is strongly feature-dependent: combinations including velo-

city and direction with either acceleration or jerk produced the most robust results, while the full four-feature set did not consistently improve accuracy. Importantly, this feature analysis shed light on the persistent challenge of PSO detection. Although PSOs were often confused with short saccades or fixations, feature sets incorporating jerk improved detection rates, reaching 67% accuracy compared to substantially higher performance for fixations, saccades, and SPs. These findings underscore the difficulty of PSO classification but also demonstrate the promise of hybrid models in approaching this challenge.

The thesis next examines the cross-task generalization of eye movement event detection. Eye movement data were analyzed across diverse visual tasks, including reading, static image viewing, video watching, and moving-dot tracking. Statistical analyses confirmed that oculomotor behavior is task-dependent: reading is characterized by rapid fixations and saccades, dynamic stimuli elicit smooth pursuits, and post-saccadic oscillations (PSOs) show task-specific variability. A CNN trained and tested across tasks demonstrated high within-task accuracy but suffered significant degradation in cross-task transfer, especially for PSOs. These results highlight the limitations of current models when applied beyond their training domain and point to the necessity of developing domain-generalized event detection approaches capable of handling heterogeneous visual and cognitive contexts.

Finally, the thesis turns to the applied significance of PSO detection, focusing on reading research. Using CNN-based classification, results were compared against commercial software that ignores PSOs and labels only fixations and saccades. The findings showed that excluding PSOs inflates fixation durations and alters widely used reading metrics such as average fixation length, which are critical for evaluating text complexity and cognitive processing. By incorporating PSOs into detection, the analysis produced more accurate and nuanced measures of reading behavior, confirming that PSOs are not merely artifacts but meaningful components of gaze dynamics. This provides an important methodological insight: ignoring PSOs risks systematic misinterpretation of eye-tracking studies in reading and related domains.

In summary, this thesis contributes: (i) a rigorous benchmarking of classical, machine learning, and deep learning algorithms under standardized conditions; (ii) the introduction of a hybrid CNN-LSTM framework with systematic feature analysis; (iii) empirical evidence for the critical importance of PSO detection in applied reading studies; and (iv) one of the first comprehensive evaluations of cross-task generalization in event detection. Collectively, the findings advance the methodological foundations of eye movement research and have broad implications for building adaptive, accurate, and context-aware eye movement analysis systems for diverse scientific and applied domains.

Abstract-Polish

Analiza ruchów gałek ocznych odgrywa kluczową rolę w badaniach nad ludzką uwagą wzrokową, percepcją oraz procesami poznawczymi. Dokładna detekcja podstawowych zdarzeń okulomotorycznych fiksacji, sakkad, płynnych ruchów podążających (smooth pursuits, SP) oraz oscylacji postsakkadowych (post-saccadic oscillations, PSO) jest niezbędna do rzetelnej interpretacji danych okulograficznych w takich dziedzinach jak psychologia, neuronauka czy interakcja człowiek–komputer. Detekcja zdarzeń pozostaje jednak zadaniem trudnym ze względu na dużą zmienność wzorców spojrzeń pomiędzy zadaniami, trudności w rozróżnianiu krótkotrwałych zdarzeń, takich jak PSO, oraz brak ustandaryzowanych protokołów ewaluacyjnych. Niniejsza rozprawa doktorska podejmuje te wyzwania poprzez kompleksową analizę metod detekcji zdarzeń okulomotorycznych, - od metod progowych, przez algorytmy uczenia maszynowego, po modele głębokiego uczenia, - ze szczególnym uwzględnieniem ich uogólnialności, doboru cech oraz metodologicznego znaczenia detekcji PSO.

Praca rozpoczyna się od systematycznej ewaluacji szerokiego spektrum algorytmów detekcji zdarzeń z wykorzystaniem danych z wysokorozdzielczych systemów śledzenia wzroku. Tradycyjne metody progowe, takie jak IVT i IDT, zostały porównane z podejściami opartymi na uczeniu maszynowym (Random Forest) oraz modelami głębokiego uczenia (CNN i LSTM). W przeciwieństwie do wcześniejszych badań porównawczych, które często opierały się na różnych zbiorach danych lub odmiennych kryteriach oceny, wszystkie algorytmy w niniejszej pracy były testowane w identycznych warunkach, z wykorzystaniem ręcznie anotowanych danych referencyjnych oraz spójnych miar jakości. Wyniki potwierdziły, że metody progowe są wydajne w binarnej klasyfikacji fiksacji i sakkad, jednak cechują się dużą wrażliwością na dobór parametrów i ograniczoną zdolnością do uogólniania. Z kolei metody uczenia maszynowego i głębokiego osiągnęły wyższą odporność oraz lepszą zgodność z anotacjami eksperckimi, przy czym sieci konwolucyjne uzyskały najwyższą dokładność ogólną. Wyniki te ustanawiają solidną podstawę metodologiczną dla przewagi podejść opartych na danych oraz podkreślają kluczowe znaczenie procedur ewaluacyjnych w szczególności strategii walidacji krzyżowej dla uzyskania wiarygodnych rezultatów.

Na tej podstawie zaproponowano hybrydową architekturę 2D-CNN-LSTM do jednoczesnej klasyfikacji fiksacji, sakkad, PSO oraz SP. Warstwy konwolucyjne pełnią rolę

automatycznych ekstraktorów cech, natomiast warstwy LSTM modelują zależności czasowe, co czyni architekturę szczególnie odpowiednią do analizy sekwencyjnych danych okulograficznych. W celu oceny wpływu reprezentacji wejściowej, systematycznie analizowano cztery cechy kinematyczne: prędkość, przyspieszenie, jerk oraz kierunek ruchu, łącząc je w różne zestawy cech. Analiza wykazała, że skuteczność klasyfikacji jest silnie zależna od doboru cech. Kombinacje obejmujące prędkość i kierunek wraz z przyspieszeniem lub jerkiem dawały najbardziej stabilne wyniki, natomiast pełny zestaw czterech cech nie prowadził do jednoznacznej poprawy dokładności. Szczególną uwagę poświęcono detekcji PSO, która pozostaje istotnym wyzwaniem. Chociaż PSO były często mylone z krótkimi sakkadami lub fiksacjami, uwzględnienie cechy jerk pozwoliło zwiększyć dokładność ich detekcji do poziomu 67%, przy znacznie wyższej skuteczności dla pozostałych klas zdarzeń. Wyniki te podkreślają trudność klasyfikacji PSO, ale jednocześnie wskazują na potencjał modeli hybrydowych w radzeniu sobie z tym problemem.

Kolejna część pracy koncentruje się na praktycznym znaczeniu detekcji PSO w badaniach nad czytaniem. Wyniki klasyfikacji opartej na sieciach CNN zostały porównane z rezultatami uzyskanymi przy użyciu komercyjnego oprogramowania, które nie uwzględnia PSO i klasyfikuje wyłącznie fiksacje oraz sakkady. Analiza wykazała, że pomijanie PSO prowadzi do zawyżenia czasów trwania fiksacji oraz modyfikuje powszechnie stosowane miary czytania, takie jak średni czas fiksacji, kluczowe dla oceny złożoności tekstu i obciążenia poznawczego. Uwzględnienie PSO pozwoliło uzyskać bardziej precyzyjne i zniuansowane miary zachowania wzrokowego, potwierdzając, że PSO nie są jedynie artefaktami, lecz istotnymi elementami dynamiki spojrzeń. Stanowi to ważny wkład metodologiczny, wskazujący, że ignorowanie PSO może prowadzić do systematycznych błędów interpretacyjnych w badaniach okulograficznych nad czytaniem i pokrewnymi dziedzinami.

W końcowej części rozprawy rozszerzono analizę na zagadnienie uogólniania modeli detekcji zdarzeń pomiędzy różnymi zadaniami wizualnymi. Dane okulograficzne analizowano w kontekstach takich jak czytanie, oglądanie statycznych obrazów, obserwacja materiałów wideo oraz śledzenie poruszającego się punktu. Analizy statystyczne potwierdziły, że zachowanie okulomotoryczne jest silnie zależne od rodzaju zadania: czytanie charakteryzuje się szybkimi fiksacjami i sakkadami, bodźce dynamiczne wywołują płynne ruchy podążające, a PSO wykazują zmienność zależną od kontekstu zadaniowego. Model CNN trenowany i testowany w obrębie jednego zadania osiągał wysoką dokładność, jednak jego skuteczność znacząco spadała w przypadku transferu między zadaniami, zwłaszcza w odniesieniu do PSO. Wyniki te uwidaczniają ograniczenia obecnych modeli poza domeną treningową oraz wskazują na potrzebę opracowania metod detekcji zdarzeń o większej zdolności uogólniania, dostosowanych do zróżnicowanych kontekstów wizualnych i poznawczych.

Podsumowując, niniejsza rozprawa wnosi następujący wkład: (i) rygorystyczną ewalu-

ację klasycznych, opartych na uczeniu maszynowym i głębokim algorytmów detekcji zdarzeń w ustandaryzowanych warunkach; (ii) wprowadzenie hybrydowego modelu CNN–LSTM wraz z systematyczną analizą doboru cech; (iii) empiryczne potwierdzenie kluczowego znaczenia detekcji PSO w badaniach nad czytaniem; oraz (iv) jedną z pierwszych kompleksowych analiz uogólniania modeli detekcji zdarzeń pomiędzy zadaniami. Łącznie uzyskane wyniki wzmacniają metodologiczne podstawy badań nad ruchami gałek ocznych i mają istotne implikacje dla projektowania adaptacyjnych, precyzyjnych i kontekstowo świadomych systemów analizy okulograficznej w szerokim spektrum zastosowań naukowych i praktycznych.

Contents

Chapter 1

Introduction

This doctoral thesis deals with the development of methods for authomatic event detection in eye-tracking signals, and strategies for evaluation of such methods. Eye-tracking technology has emerged as a powerful tool for analyzing human visual attention, cognitive processes, and decision-making behaviors. By recording eye movements and estimating gaze direction, eye-tracking systems enable researchers and practitioners across multiple disciplines including cognitive science, psychology, neurology, engineering, medicine, and marketing to gain deeper insights into human perception and interaction [83] [12].

Eye-tracking technology has been widely adopted across diverse domains, including human–computer interaction, psychology, neuroscience, education, marketing, automotive research, and assistive technologies. In human–computer interaction, gaze has been used as an explicit input modality for hands-free interaction and accessibility applications [47, 69, 53]. In particular, gaze-based interaction empowers individuals with physical disabilities to control digital interfaces using their eyes [76]. In psychology and cognitive science, eye movements provide a non-invasive window into attention, perception, and cognitive processing [72, 21]. Beyond laboratory settings, eye tracking has also been applied in real-world contexts such as marketing research [6, 103], education [18], and assistive communication systems [68, 11]. In automotive research, eye tracking has been employed to monitor driver attention and situational awareness, contributing to accident prevention and driver safety systems [75]. Together, these broad applications underscore the importance of reliable and generalizable eye movement analysis methods.

Central to eye-tracking research is the identification and classification of fundamental eye movement events, which include fixations, saccades, smooth pursuits (SPs), and post-saccadic oscillations (PSOs). Fixations occur when the eye remains relatively stable,focusing on an object to stabilize its image on the fovea for clear vision [90] [61]. Saccades are rapid, ballistic movements that reposition the gaze from one location to another, playing a crucial role in visual exploration [41]. Smooth pursuits occur when the eyes track a continuously moving object [7, 106], whereas PSOs are small oscillatory movements that may occur immediately after a saccade before the gaze stabilizes on a new target. They

can be described as instabilities that appear at the end of a saccade and are characterized by a slight wobbling movement that leads to fixation after a saccade [106, 79]. The accurate detection of these events is essential for understanding visual processing in diverse scenarios, from reading behavior to interactive system design.

Despite significant advancements in eye-tracking technology, the automatic detection of these events remains a challenging problem. The presence of noise in raw gaze signals, the difficulty of distinguishing overlapping event types (e.g., SPs and fixations), and variations in recording conditions introduce considerable complexity. Even manual classification by experienced coders is subject to inconsistencies, highlighting the necessity for robust and automated event detection methods [43, 2].

Nowadays, event detection is almost exclusively done by applying a detection algorithm to the raw recorded eye-tracking data. Historically, eye movement event detection has relied on threshold-based algorithms, which classify movements based on predefined spatial or velocity criteria. For a long time, two broad classes of threshold-based algorithms were used for eye movement event detection. The first class is the I-DT (Identification by Dispersion Threshold) algorithms that detect fixations by grouping gaze points that remain within a spatially constrained area for a minimum duration, with all other movements classified as saccades [109]. The most well-known dispersion-based algorithm is the I-DT algorithm by Salvucci and Goldberg [91]. These algorithms detect the event by defining a spatial box that the raw recorded data must fit for a specified minimum time. The second class is the velocity-based algorithms that detect saccades and assume the rest to be fixations. Conversely, velocity-based algorithms, such as the I-VT (Identification by Velocity Threshold) algorithm, classify saccades by identifying points that exceed a predefined velocity threshold, with all remaining points assumed to be fixations [100, 33]. While these methods have been widely used, they suffer from limitations, including the need for manual parameter tuning and their inability to reliably classify more complex movement patterns, such as smooth pursuits and PSOs.

To address these shortcomings, researchers have explored alternative approaches, including multi-stage classification models and hybrid detection frameworks [56, 62]. The approach performed the classifications through iterative adjustments in three stages: a preliminary segmentation evaluating the characteristics of each such segment and reorganizing the preliminary segments into fixations and smooth pursuit events. This approach was proposed to address the misclassification between fixations and smooth pursuits. However, just like all other threshold-based approaches, this approach also needs the end users to set threshold parameters manually, and the three-stage identification process makes the approach complex and time-consuming.

Recent advances in machine learning and deep learning have introduced promising solutions for automated event detection. Machine learning models, such as Random Forest classifiers, have been successfully applied to classify fixations, saccades, and PSOs, while

deep learning architectures, including Convolutional Neural Networks (CNNs), have demonstrated effectiveness in detecting smooth pursuits by analyzing signal frequencies rather than raw amplitudes [111, 44].

Temporal convolutional networks [110] and hybrid models [98] have further improved classification accuracy, yet most existing methods still struggle to simultaneously detect all four key eye movement events.

The primary aim of this doctoral research is to advance the development of automated methods for detecting and classifying eye movement events from raw eye-tracking data. By addressing the limitations of existing approaches, this work seeks to improve the accuracy, robustness, and efficiency of event detection in diverse eye-tracking applications. The research explores various event types, including fixations, saccades, smooth pursuits (SPs), and post-saccadic oscillations (PSOs), and develops novel methodologies for their identification and classification.

To achieve this aim, the thesis conducts a comprehensive analysis and benchmarking of existing event detection methods, including threshold-based and machine-learning approaches, to identify their strengths and weaknesses. It investigates the most informative eye-tracking signal features to optimize classification accuracy across different event types. This thesis conducts a comparative evaluation of traditional threshold-based models against machine learning and deep learning approaches for eye movement event classification. In addition, it focuses on the design and implementation of a multi-class classification model capable of simultaneously detecting fixations, saccades, smooth pursuits (SPs), and post-saccadic oscillations (PSOs).

This thesis proposes a hybrid deep learning architecture that combines two-dimensional Convolutional Neural Networks (2D-CNNs) with Long Short-Term Memory (LSTM) networks for the simultaneous detection of four core eye movement event types: fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs). Relevant features are extracted and engineered from raw gaze data, including velocity, acceleration, jerk, and direction, and used as inputs to train the proposed deep learning model. The impact of these feature sets on classification performance was extensively analyzed. By evaluating different feature configurations, the thesis offers insights into the role of input features in event detection accuracy.

In addition to developing and evaluating the detection model, the thesis explores how the structure of classification tasks effects model performance. Specifically, the evaluation is extended by training and testing the model under two alternative three-class configurations: one comprising fixations, saccades, and PSOs; and the other comprising fixations, saccades, and SPs. This comparison aims to understand how the composition of event classes influences detection accuracy and inter-class confusion. This analysis is crucial, as PSOs and SPs often exhibit signal characteristics that overlap with those of fixations and saccades, leading to misclassifications in automated systems. SPs are typically pro-

longed movements occurring during smooth target tracking, while PSOs are brief, oscillatory events that follow saccades. These subtle temporal and spatial distinctions present unique challenges for machine learning models. The extended analysis conducted in this thesis not only identifies event types that are inherently more difficult to detect but also demonstrates how the inclusion or exclusion of specific classes affects the model's ability to differentiate between remaining events. This contributes to a deeper understanding of the complexities involved in deep learning-based eye movement event detection.

Additionally, we conducted a comprehensive analysis of eye movement data collected across different visual tasks: reading, image viewing, video watching, and moving dot tracking. Specifically, the velocity characteristics, frequency, and duration of eye movement events (fixations, saccades, post-saccadic oscillations, and smooth pursuits) across these tasks were investigated. In addition, a convolutional neural network (CNN)–based event detection model was developed and evaluated across three distinct visual tasks—static image viewing, video watching, and moving dot tracking. The model was designed to classify three event types: fixations, saccades, and post-saccadic oscillations (PSOs). To assess cross-task generalization, it was trained on one task and tested on the others, and vice versa.

Furthermore, a novel approach is proposed and evaluated for the detection of PSOs within reading tasks, contributing to more precise gaze analysis in cognitive and linguistic research.

This research provides a comprehensive framework for event detection in eye-tracking studies, bridging the gap between classical methods and emerging deep-learning-based solutions. The findings contribute to advancements in human-computer interaction, cognitive science, and medical diagnostics by enhancing the precision and automation of gaze-based behavioral analysis.

The remainder of this thesis is structured as follows: Chapter 2 outlines the research objectives and scope; Chapter 3 provides a detailed review of relevant literature on eye anatomy, movement types, and detection algorithms; subsequent chapters delve into the proposed methodology, data analysis, evaluation, and results, culminating in discussions on the findings and their implications for future research.

1.1 Scope of Thesis

The scope of this thesis is to advance the methodological foundations of automatic eye movement event detection by integrating both classical and modern approaches, with a strong emphasis on deep learning. The scope covers the entire pipeline of event detection, starting from the analysis of oculomotor behavior across diverse visual tasks to the design, implementation, and evaluation of advanced detection models.

The research encompasses several key areas. First, classical threshold-based algorithms,

machine learning models, and deep learning approaches are systematically evaluated and compared under consistent conditions, using both manual annotations and quantitative performance metrics. Second, a hybrid CNN–LSTM framework is introduced with detailed investigation into the role of motion-derived features (velocity, acceleration, jerk, and direction) in distinguishing overlapping event types. Third, the thesis addresses the importance of post-saccadic oscillations (PSOs), demonstrating how their detection influences reading research outcomes. Finally, the analysis of oculomotor behavior across diverse visual tasks examines the extent to which models trained on one visual task generalize to others, thereby providing new insights into cross-task event detection.

While broad in methodological coverage, the scope is bounded by several practical considerations. The datasets used stem from controlled laboratory conditions involving reading, image viewing, video watching, and moving-dot tracking, without extending to mobile or real-world eye tracking scenarios. Feature extraction is restricted to kinematic properties derived from gaze coordinates, and multimodal signals (e.g., pupil size, EEG) remain outside the present scope. Likewise, the thesis focuses on classification accuracy and inter-rater agreement rather than computational efficiency or real-time deployment. Furthermore, comparisons with previous event detection methods are limited, since different researchers often rely on different visual tasks, event types, annotation schemes, and devices, making direct benchmarking difficult or unfair [99]. To address this, the thesis emphasizes controlled comparisons—either by testing the same model across feature combinations and event classes, or by comparing multiple algorithms under identical conditions.

1.2 Hypotheses

Hypothesis 1 (H1): *Threshold-based algorithms are insufficient in many applications involving eye movement events detection, and it is advisable to replace them with machine learning algorithms.*

Threshold-based algorithms remain widely used for eye movement event detection, but they are constrained by several drawbacks: they typically support only binary classification at a time, require multi-stage processing to detect more than two events, and rely on manually defined threshold values. The threshold values vary across tasks, datasets, and developers, leading to inconsistent results and poor generalizability. In contrast, machine learning and deep learning approaches are hypothesized to provide more efficient and robust alternatives. By learning discriminative patterns directly from data, they eliminate the need for hand-crafted parameters, enable end-to-end multi-class detection in a single step, and scale effectively with large datasets. As such, ML/DL models are expected to achieve higher accuracy, robustness, and consistency across diverse eye movement event types compared to traditional threshold-based methods.

Hypothesis 2 (H2): *The CNN-LSTM hybrid model, when provided with appropriate kinematic feature combinations (velocity, acceleration, jerk, and direction), can robustly and simultaneously classify the four primary eye movement events—fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs).*

The hypothesis suggests that the performance of the model is influenced by the selection and combination of input features. Specifically, it posits that different event types benefit from distinct sets of features such as velocity, acceleration, jerk, and direction and that optimizing these combinations enhances classification accuracy. For example, fixations and saccades may be reliably detected using velocity and acceleration, while the detection of SPs and PSOs requires additional information such as jerk and movement direction. Systematic evaluation of feature combinations is expected to identify the most informative sets for each event class, thereby improving the overall performance and generalizability of the model.

It is further hypothesized that event configurations (e.g., fixation-saccade-PSO [FSPso] vs. fixation-saccade-SP [FSSP] configurations) influence classification accuracy and inter-class confusion, reflecting the inherent complexity of simultaneously detecting diverse oculomotor behaviors. The choice of which events to include in a classification task is not merely a question of model architecture, but also of inter-class relationships and visual behavior overlap.

Hypothesis 3 (H3): *Machine learning-based eye movement detection models demonstrate high performance when applied to the same visual task on which they were trained, but show reduced accuracy when applied to other tasks, reflecting limited cross-task generalization.*

This limitation arises because eye movement events differ systematically in their frequency, duration, and velocity across visual tasks (e.g., reading, image viewing, video watching, and moving-dot tracking). These task-dependent variations in oculomotor behavior influence the statistical properties of events and consequently affect detection performance when transferring models between tasks.

Hypothesis 4 (H4): *Post-saccadic oscillations (PSOs) are often overlooked in eye movement analysis, particularly in reading data, yet their presence directly affects fixation and saccade classification.*

It is hypothesized that explicitly detecting PSOs during event classification, especially in reading tasks, significantly alters the boundaries between fixations and saccades. Misclassifying PSOs as either fixations or saccades distorts key statistical measures such as event duration, velocity, and frequency. Incorporating explicit PSO detection is therefore expected to improve overall classification accuracy and enhance the reliability of eye-tracking analyses in reading research.

Chapter 2

Literature Review

2.1 The Anatomy of a Human Eye

The human eye is a sophisticated organ that plays a critical role in the cognitive system, allowing us to perceive, process, and interpret visual information. It functions as a complex optical system, capturing light from the external environment and converting it into neural signals that are processed by the brain. The structure of the human eye is intricately designed to perform this function, consisting of several key components.

The sensory systems in the human body consist of sensory receptor cells that are stimulated from internal or external sources in the body, neural pathways that transfer the sensory information to the brain, and parts of the brain where the sensory information is processed [105]. In the human body, there are several sensory systems, for example, the auditory for hearing, the vestibular for balance, and the visual system for vision. The visual system is the sensory system that makes it possible for us to process visual information that we capture through our eyes [105]. It comprises the eyeball, the muscles surrounding it, and the neural pathway transmitting the signals to the brain. The function of the eyes in the visual system is to focus light from objects around us to the rear part of the eyeball and convert the light to electrical signals that are transmitted to the brain for further processing [65]. The eye is a liquid filled ball that is enclosed by a white surface called the sclera. An illustration of the human eye is shown in Fig 2.1. From the outside, parts of the sclera are seen together with the colored iris and the black pupil. The sclera surrounds the eyeball except for its most anterior surface which is the thin transparent and protective layer called the cornea. The cornea is the first medium of the eye to reflect and refract incoming light, before it passes through the pupil and further to the lens where the light refracts once more [22]. The size of the pupil changes with the ambient light conditions and controls the amount of light entering the eye and the lens. When the light refracts in the lens, fine adjustments are performed before the light continues through the liquid filled globe to the rear parts of the eyeball, the retina. The retina is a thin layer of tissue that

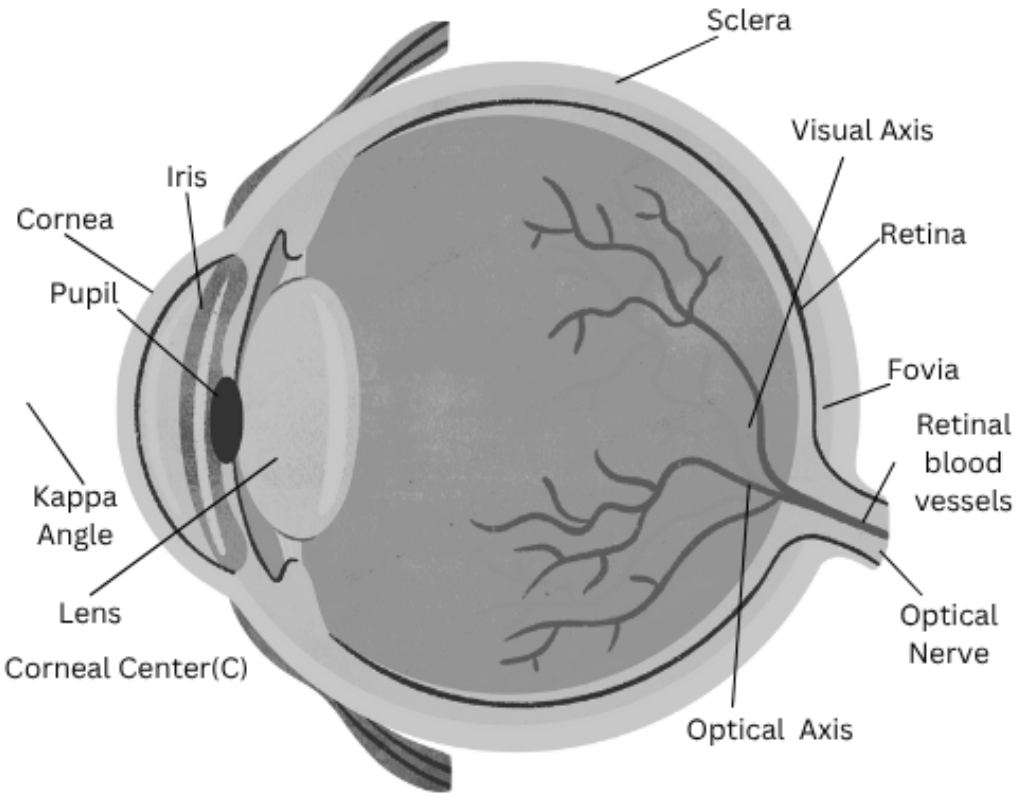


Figure 2.1: The human eye anatomy

covers most of the inner walls of the eyeball. It is sensitive to light and consists mainly of photoreceptor cells, nerve cells, and glial cells [14]. There are two types of photoreceptive cells: rods and cones. These two types of photoreceptive cells have different functions; cones enable color vision and high visual acuity, while the rods are important for night vision and for detection of motion.

2.2 Eye Movements

Eye movements are essential for visual perception, allowing humans to explore and interact with their environment. They enable the eyes to focus on objects of interest, maintain stability during motion, and track dynamic stimuli. Eye movements are controlled by a complex coordination of the oculomotor system, extraocular muscles, and neural pathways. These movements are categorized into distinct types based on their function, speed, and purpose [88].

Accurate detection of eye movement events such as fixations, saccades, and smooth pursuits is a foundational requirement for meaningful interpretation of gaze data. Event segmentation directly determines the validity of downstream measures including fixation duration, saccade amplitude, and scanpath structure [20]. Errors at this stage propagate into higher-level cognitive and behavioral inferences, affecting conclusions across application domains [55]. As a result, event detection has remained an active research area, evolving from early heuristic approaches to more recent machine learning and deep learning-based methods [30, 29].

2.2.1 Fixations

Fixations occur when the eyes remain relatively still, focusing on an object to stabilize its image on the fovea for clear vision. This is the most frequent eye movement type during visual exploration and is critical for acquiring detailed information [61]. Fixations typically last between 200 and 600 milliseconds, depending on the task.

During fixations, three types of micro-movements occur to prevent visual fading: tremor, slow drift, and microsaccades [61]. Tremor is a small wave-like motion of the eye, with an amplitude around 0.01 ∘ and a frequency below 150 Hz [61]. The function of tremor is still largely unknown. Drift is a slow motion of the eye, which occurs simultaneously with tremor [70]. It was for a long time believed that drift was a random motion of the eye due to instability in the oculomotor system. Later, it was found that drift has a compensatory role to maintain visual acuity during fixations, when there are not sufficiently many microsaccades. A microsaccade is the fastest of the fixational eye movements and has a duration of about 25 ms [70]. Microsaccades occur around 1-2 times per second, depending on the task. Even though the amplitude of a microsaccade is lower than for a normal saccade, they share many properties. Recently, it was found that microsaccades may be voluntary movements when performed during natural tasks [89].

2.2.2 Saccades

Saccades are rapid, ballistic movements of the eyes that reposition the gaze from one point of interest to another. They are the fastest type of eye movement, with velocities reaching up to 500°/second and durations ranging from 30 to 80 milliseconds [41]. A relationship exists between the duration, amplitude, and velocity of a saccade, which suggests that larger saccades have larger velocities, and last longer [4]. The latency in the saccadic system is around 200 ms, and corresponds to the time from the onset of the stimulus to the initiation of the eye movement. This includes the time it takes for the central nervous system to determine whether a saccade should be initiated or not, and, if this is the case, calculate the distance that the eye should move, and transmit the neural pulses to the muscles that move the eyes. A common assumption is that a saccade is

a straight line between point A and point B. However, in reality, a saccade is seldom a straight line, instead it most often has a slightly curved trajectory [67]. Saccades are critical for visual exploration, enabling the eyes to scan scenes and acquire new information.

2.2.3 Smooth Pursuits

A smooth pursuit is performed when the eyes track a moving object, e.g., follow a bird that flies across the sky. A smooth pursuit movement can only be performed when there is a moving object to follow [58]. The latency of the smooth pursuitsystem is about 100 ms, which is slightly shorter than for saccades [41]. The latency of the smooth pursuit system corresponds to the time it takes for the eye to start moving from the onset of the target motion. A smooth pursuit movement can broadly be divided into two stages: open-loop and closed-loop [106]. The open loop stage is the pre-programmed initiation stage of the smooth pursuit where the eye accelerates in order to catch up with the moving target. The closed-loop stage starts when the eye has caught up with the target and follows it with a velocity similar to that of the target. In order to be able to follow the moving target in the closed-loop stage, the velocity of the moving target is estimated and compared to the velocity of the eye. If the velocity of the two are different, e.g., the eye lags behind the moving target, a movement known as a catch-up saccade is performed in order to catch up with the target again. The human eye can follow a target at velocities up to 100°/s [73]. The higher the velocity of the moving target, the more catch-up saccades are needed in order for the eye to be able to follow the target. However, most often smooth pursuit movements have velocities below 30°/s. If the stimulus only consists of one moving target that moves in a predictable way, the eye will be able to follow it more accurately, with fewer catch-up saccades [64].

2.2.4 Post Saccadic Oscilations

Rapid oscillatory movements that may occur immediately after the saccade, are referred to as post saccadic oscillations (PSO). They can be described as instabilities or oscillatory movements that may occur at the end of a saccade [4]. The amplitudes of PSO range from 0.25° up to over 1° and there are large individual differences in both the amplitude and occurrence of PSO [79, 42].

The characteristics of the eye-tracking data originating from different recording systems may vary depending on the measurement principle, sampling frequency, and internal filters of the system. The appearances of most types of eye movements are the same, but the appearance of PSO is different for different recording systems. This fact has made researchers to start questioning whether PSO represent eye movements or are artifacts from the eye-tracker. PSO have been reported from search coil systems [51], DPI eye trackers [15] and VOG-systems [79]. In [15], simultaneous recordings with search coils and a Dual

Purkinje Image eye-tracker showed PSO in data recorded when using the DPI, but not in the data from the search coils. Terefore, Deubel and Bridgeman [15] concluded that PSO originate from lens wobbling and not from rotations of the eye. Recent research suggests that it is the pupil that is moving inside the iris, causing the PSO in data from video based eye-trackers [79]. Regardless of the origin of PSO, and their consequences for perception, the question of whether PSO should be classified as belonging to saccades, as belonging to fixations, or simply be removed from the recorded data remains unsolved.

2.2.5 Vergence eye movements

When the two eyes point at the same object, the eyes need to be directed in slightly different directions. This is due to the fact that the two eyes are separated by a few centimeters. The movements that the eyes perform when they either move towards each other, convergence, or away from each other, divergence, are often referred to as vergence eye movements [64]. Each eye needs to be controlled separately, in order to keep the same object on the fovea of both eyes. This is especially important for objects that are at a close distance. In order for the brain to be able to combine the images of the object from the two eyes into one, the object must lie on the corresponding spot on the retina of each eye. The maximum visual angle that the object can be apart on the retina is called Panum's area [64]. This means that if the object seen from the two eyes is within this area, the images are combined into one, and if not, the object will be interpreted as two and double vision will occur [64].

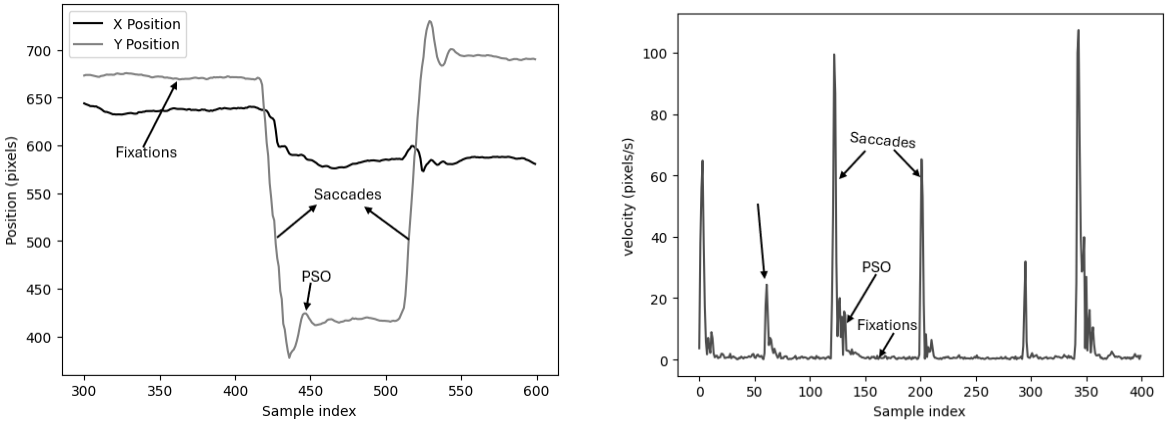
2.2.6 Vestibular eye movements

The function of vestibular eye movements is to stabilize the image on the fovea in order to sustain clear vision during head rotations [64]. Since the vestibular eye movements respond to signals from the vestibular system, the latency for these eye movements is shorter than for eye movements initiated by the visual system. The latency of the system can be as short as 7-15ms, compared to about 200ms for the saccadic system. The eye movement vestibular ocular reflex, VOR, responds to both translational and rotational head movements, which both are natural movements in everyday life. The translational head movements are performed when the head moves from left to right, up and down, or forward and backward, with the nose pointing in the same direction. For rotational head movements, the head can rotate around three axes: horizontally, vertically, and torsionally. Horizontal rotation corresponds to shaking the head, vertical rotation corresponds to nodding, and torsional rotation corresponds to lying the head against one of the shoulders. The size of the compensatory eye movements that are needed to keep the image on the fovea during these head movements is larger for closer objects than for distant objects.

2.2.7 Optokinetic and Nystagmus quick phase

Optokinetic eye movements are similar to vestibular eye movements in the sense that they are initiated in order to keep the image on the fovea and compensate for head movements. Optokinetic eye movements respond to sustained head rotations, e.g., when sitting in a spinning chair. In order for the eye not to get stuck in the outer part of the eye socket in the opposite direction of the rotation and not be able to make any movements during sustained head rotations, the eye needs to quickly move in the same direction as the rotation, referred to as the quick phase of nystagmus [64].

Out of the functional classes of eye movements presented in this section, fixations, saccades, PSO, and smooth pursuit movements are the ones that most often are considered by event detection algorithms 2.2.



Position over time (a) Velocity over time (b)

Figure 2.2: Example of saccades, fixations, and PSOs. (a) Position over sample index, (b) velocity over sample index.

2.3 The Eye Trackers

Originally, an eye-tracker referred to an equipment that was used to measure the orientation of the eye, while a gaze-tracker was used to estimate where a person was looking. Over time, these two terms have been used interchangeably, and in the following the term eye-tracker is used to refer to the equipment that tracks the movements of the eyes and that estimates the direction and position of gaze. Over the last 100 years, several different types of measurement techniques have evolved.

This section provides an overview of eye-tracking technology, its principles, types, and how it applies to the analysis of eye movement during various visual tasks. Eye trackers are essential tools for capturing and analyzing eye movements, enabling the investigation of cognitive processes like attention, memory, and perception by recording where and how long a person looks at specific visual stimuli.

Eye-tracking technology typically uses infrared light to illuminate the eye and capture reflections from the cornea and pupil. These reflections are used to calculate the gaze point on a screen or in the environment. Eye trackers vary in design but generally fall into two categories: remote and wearable. The choice of eye tracker depends on the research objectives, the need for mobility, and the type of task studied.

Modern eye trackers are capable of capturing high-speed, high-precision data that can be used to measure various eye movements, such as fixations, saccades, post saccadic oscillations and smooth pursuits. The data generated can be used to understand a wide range of cognitive and behavioral processes, from reading comprehension to decision making in visual tasks.

2.3.1 Video-oculography (VOG)

The most widely used type of the eye tracker today, VOG, uses infrared cameras to track the pupil and corneal reflections (Purkinje images). Systems range from tower-mounted (high precision, low mobility) to remote (moderate precision, high comfort) to mobile (high ecological validity). The VOG setup consists of infrared light sources and high-speed cameras, typically sampling at 60–1200 Hz, depending on the research demands [20] [41].

The eye detection step involves isolating the pupil and corneal reflection (CR), followed by gaze estimation through geometrical modeling. Factors such as lighting conditions, head movement, and calibration drift affect data accuracy.

Eye detection

With very few exceptions, VOG systems consist of one or several cameras that record the eye and one or several infrared light sources directed towards the eye. Since the light is infrared, it is not visible to the human eye and will therefore not distract the user [30]. The infrared light sources give rise to reflections in the eye, referred to as Purkinje images. The first Purkinje image, is the reflection in the cornea, and is therefore called the corneal reflection (CR). In the eye, there are four changes in medium that may reflect the incoming light and give rise to Purkinje images. An illustration of these four reflections is shown in Fig. 2.3. To detect and track the eye in the image captured by the camera is a challenging task, e.g., due to occlusion of the eye, the degree of openness of the eye, variation in size, reflections, viewing angle, head pose, eye color, light conditions, and variation in eye shape [30]. Several methods have been proposed in order to overcome these challenges. The methods are divided into three main categories: shape-based, appearance-based, and hybrid methods. The shape-based methods use either models that rely on local features or contours of the eye. A wide range of models have been used, from simple elliptic models to more complex models that take both the shape of the eye and the structure that surrounds

it into account. The simpler methods are not robust to changes in light and focus of the camera, and to occlusion, i.e., periods when the user closes the eyelid. On the other hand, the more complex models suffer from being computationally demanding, in need of high resolution images with high contrast, sensitive to changes in pose, and also to occlusion of the eye [30]. The appearance-based methods are based on templates that detect and track the eye based on the distribution of color or responses from a filter bank that enhance desired features in the image. The weaknesses of appearance-based methods are that they are not invariant to scale and rotation of the eye, and since a template is used, it is difficult to capture all variations of human eyes. The hybrid models combine shape-based methods with the appearance-based methods in order to overcome the limitations of each method. One such method uses part-based modeling, which attempts to build a general model out of smaller parts of the image [30]. One limitation with this type of method is that a specific model needs to be built for each person [30].

Gaze Estimation

The goal of the gaze estimation part of the eye-tracker is to convert the information extracted from the image of the eye into a gaze direction or the position of gaze [30]. Most gaze estimation methods are based on features, which means that they extract features such as the contours of the eye and the pupil, and different reflections in the surfaces of the eye and based on these features calculates the direction of gaze [30] Feature-based methods can be divided into two main categories: interpolation-based methods and model-based methods. Characteristic for the interpolation-based methods is that the extracted features from the image are mapped to gaze coordinates by a mapping function that most often is a parametric function, e.g., a polynomial function. Other non parametric functions may also be used, e.g., a neural network [48]. In the interpolation-based methods, the gaze position is explicitly calculated without previous calculation of the direction of the gaze. The model-based methods are based on a geometric model of the eye and the objects that are being viewed, and the gaze direction is estimated based on the features extracted from the image of the eye. The position of gaze, referred to the point-of-regard (POR), is estimated as the intersection between the gaze direction and the nearest viewed object, e.g., the monitor.

The simplest VOG eye tracker is based on one camera and one light source. The idea behind this type of setup is that when the eye moves the pupil moves with it. Instead of measuring the movement of the eye directly, it is indirectly measured by the motion of the pupil in the recorded image. This setup assumes that the CR does not move much when the eye moves, and because of that, the CR can be used as a reference position in the recorded image. Thus, when the user looks in different directions, the relationship between the CR and the pupil changes. By asking the user to look at a number of predefined positions on a monitor, referred to as calibration points, a relationship between the relative positions

of the CR and the pupil, and the positions on the monitor can be established. This setup works well when the head is fixated, e.g., for the tower mounted setup. In order to be able to perform gaze estimation in front of a computer screen when the head is allowed to move, e.g., when using a remote system setup, the number of light sources and/or the number of cameras needs to be increased. By using a setup with one camera and multiple light sources the setup is made invariant to head pose, e.g., by placing four IR-light sources on the corners of the monitor that the test person is facing, and by calculating the projection of the light sources on the surface of the cornea, the gaze can be estimated [108]. The method is head pose invariant, but sensitive to changes in depth, i.e., if the distance between the user and the monitor changes.

In the setup with one camera and multiple light sources, there is often a trade off between a wide angle camera that allows for large head movements and an image of the eye with high enough resolution and contrast in order to be able to detect and track the eye in the image. In order to solve this problem multiple cameras and multiple light sources can be used. In a multiple camera setup, one wide angle camera and one narrow angle camera that is directed towards the eye may be used. When using multiple cameras, the cameras need to be calibrated in order to avoid problems when matching images from the two cameras to each other, and in addition, it is more data to process. For a complete review of eye detection and gaze estimation methods, see [30].

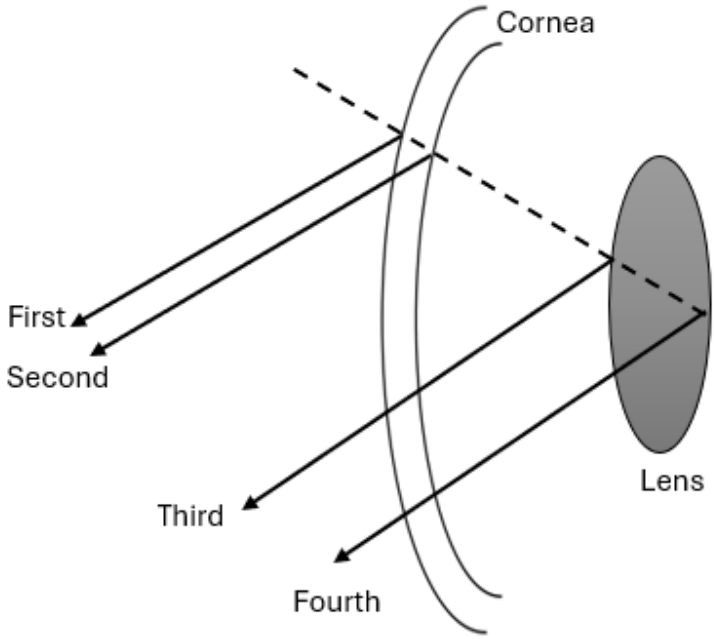


Figure 2.3: An illustration of the four Purkinje images.

Types of VOG systems and their data

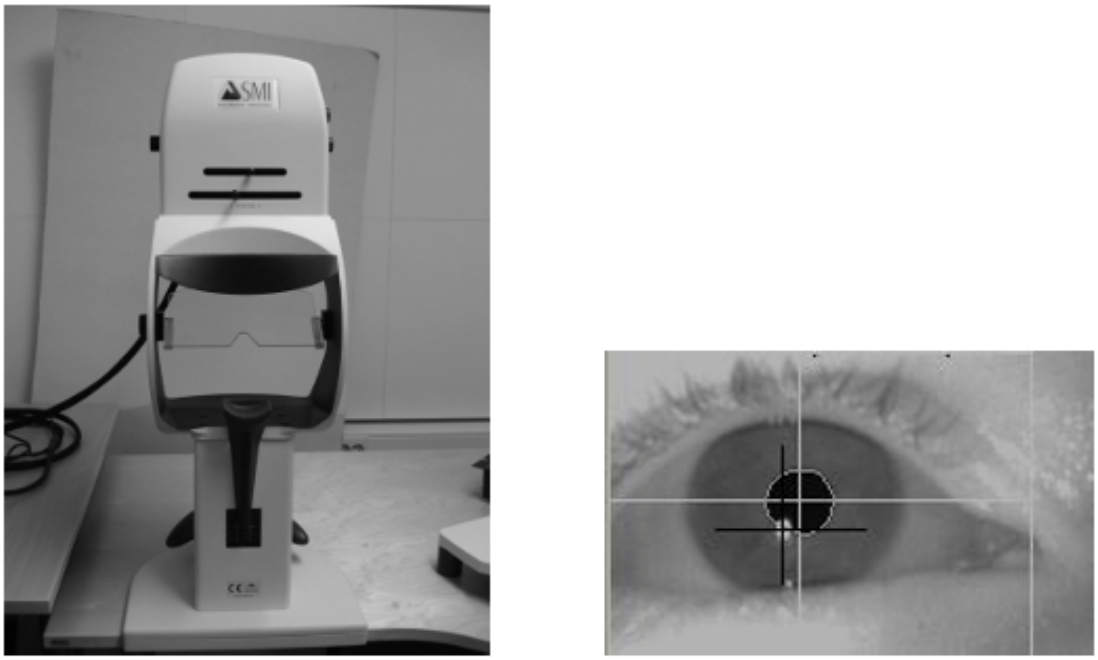
Tower Mounted Eye Trackers

A tower mounted eye tracker, see Fig. 2.4 (a), consists of a pillar where the user places the head. At the top of the pillar a camera and an infrared light source pointing downwards are attached. The camera films the eye through a mirror, which is placed in front of the user's eyes. The camera is typically a high speed camera with a frame rate of frames/s. The infrared light source which is directed via the mirror towards the eye, gives rise to the CR. In the image of the eye, captured by the camera, the CR and the pupil are detected, see Fig. 2.4 (b). The tower mounted eye-tracker requires that the head of the user is fixated in order to record data with high quality.

Because of its size and that the setup requires the user to have the head fixated, a tower mounted eye-tracker is typically used for laboratory experiments. A typical setup is that the tower mounted eye-tracker is placed in front of a computer screen where the stimuli are presented to the user. Eye-tracking data recorded from a tower mounted eye-tracker are shown in Fig. 2.4, where the user is reading a text.

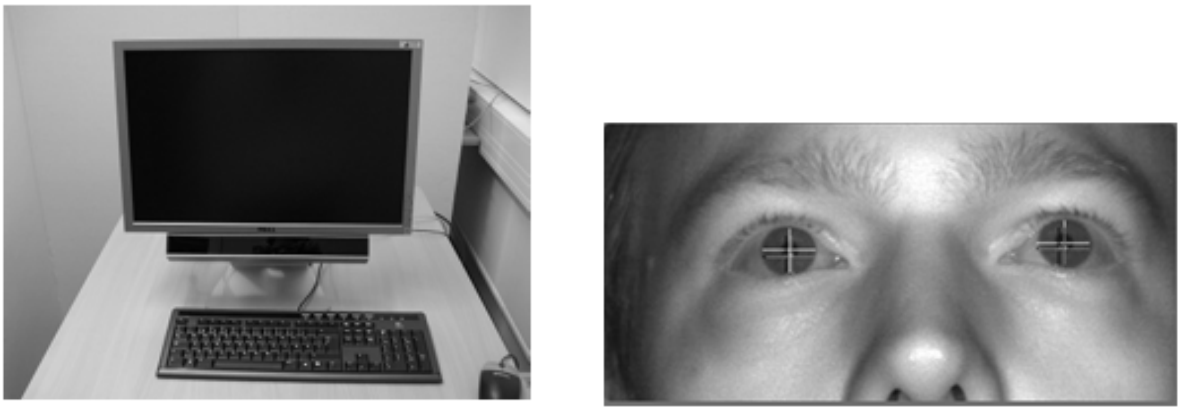
Remote Eye Trackers

In a remote eye-tracking system, illustrated in Fig. 2.5a, the camera or cameras are attached below a computer screen. In contrast to the tower mounted eye-tracking system where the image from the camera covers only the eye, the image from a re mote camera covers larger parts of the face, see Fig. 2.5b. Since the user has the freedom to move, although within a limited range, the eye detection methods need to be more robust to larger movements of the eyes between frames, than the methods in the tower mounted eye-tracking system. Thee cameras in a remote system can either be integrated in the computer monitor or be a separate device that can be mounted on any monitor or laptop. Remote eye-trackers are available in many forms, from a low cost web-camera solution that samples at 25 Hz, to fully integrated systems with sampling frequencies up to 1000 Hz. Since the user is able to move during the recordings, remote eye-trackers are very popular. But the freedom for the user to move during the recording comes at the cost of less precise and less accurate data compared to data recorded with the tower mounted eye-tracker. Examples of research where a remote eye-tracker has been used are infant studies [38] and recordings of school children [40], where a tower mounted eye-tracker often is not applicable



(a) (b)

Figure 2.4: (a) An example of a tower mounted VOG-system. (b) An image of an eye captured with the camera of the system in (a), where the detected pupil and CRare marked with a white and a black cross



(a) (b)

Figure 2.5: (a) An example of a remote VOG-system. (b)An image of the eyes captured with a remote VOG-system, where the detected pupils and CRs are marked with crosses

Mobile eye-trackers

Both tower mounted systems and remote systems are used in laboratory experiments where a participant is seated in front of a computer screen. The third type of system is a mobile system, where the eye-tracking equipment is attached to a helmet, a cap, or a pair of glasses. The mobile eye-tracking system typically consists of a camera that records the eye, some versions have one camera for each eye, and one camera, referred to as the scene camera, that captures the scene that the user is exploring. The camera(s) that films the eye can be placed either on the head of the user filming the eye through a mirror, or be integrated on the inside of the frame of a pair of glasses filming the eye directly. Since the camera is placed on the user, the eye movements are recorded in relation to the movements of the head, referred to as eye-in-head motion. In order to record eye movements in relation to a world coordinate system, referred to as eye-in-space motion, the position of the head and the body needs to be measured with either external equipment or through the scene camera. In many mobile eye-tracking systems, the recorded eye-in-head signal is given in the coordinate system of the scene camera, and the gaze is marked with a cross or a dot in the scene camera video. The sampling frequency of the recorded eye-tracking signal presently ranges from 25 Hz up to 100 Hz.

The flexibility of mobile eye-trackers opens up for experiments outside the lab oratory when studying, e.g., decision making in the supermarket [28] or eye movements during sports activities [60].

2.4 Event Detection Algorithms

Event detection can be done manually by human coders or by algorithms. In manual event detection, one or more human experts classify raw eye tracking data into different

event types based on subjective threshold values. Manual classification is still a common method for evaluating event detection algorithms and is treated as a "golden standard" [43]. However, manual event classification is not an effective way to classify events. Firstly, it is time-consuming and secondly, different coders may use different subjective selection rules that give different results [25].

Threshold-based methods are historically the first automated eye movement event classification algorithms and are still frequently used nowadays [91]. For instance, Salvucci and Goldberg evaluated five different threshold-based event detection algorithms introduced by different authors. The evaluation was based on spatial and temporal criteria. The algorithms are namely, (I-VT and I-DT [91]), I-HMM ([92]), I-AOI and I-MST([91]). The I-DT is the most straightforward and obvious eye movement event detection algorithm that classifies fixation and saccade points based on the dispersion of subsequent sample coordinates. The algorithm identifies gaze samples as belonging to fixation when the samples are located within a spatially limited area (for example 0.5°) for minimum allowed fixation duration [97]. It follows the assumption that fixation points generally occur close to each other. Saccades are then detected implicitly as everything else [91]. However, the dispersion threshold methods exhibit poor performance in detecting fixations and saccades when the signal is noisy [31]. Therefore, choosing the optimum threshold values is the most challenging step in the I-DT because the impact of varying threshold values on the classification performance leads to biased results and misclassifications.

The I-VT event detection algorithm is another threshold-based algorithm and the foundation for automated objective standard event detection. Many studies adopted this approach [93, 24]. It utilizes the fact that saccadic eye movements are characterized by high velocity and fixational movements are characterized by low velocity. The I-VT method identifies events by calculating point-to-point velocity and then classifies the event as fixation or saccade based on the relation to the predefined velocity threshold [91]. However, just like I-DT, the classic I-VT method is designed to classify eye-tracking input data into fixations and saccades only. The other event types, like smooth pursuits, post-saccadic oscillations, and noises, are not considered. Finding the optimum threshold value is also challenging.

One of the main problems with threshold-based event detection methods discussed in this section is that they can only identify fixation and saccade movements. In order to address this problem, Komogortsev and Karpov [56] proposed the first ternary automated event detection methods. The methods classify the raw eye-tracking data into fixations, saccades, and smooth pursuits. The methods are I-VVT, I-VDT, and I-VMP. The I-VVT identifies fixations, saccades, and smooth pursuits (SP) by applying first the I-VT to classify fixations and saccades using the existing threshold and then identifies SPs from fixations by adding one more velocity threshold. The I-VMP first classifies fixations and saccades by applying the I-VT algorithm and then distinguishes smooth pursuits

from fixations using the movement pattern. As discussed in the I-VT-based classification method, the measured velocity can be used to classify gaze samples as fixations or saccades. However, as smooth pursuit movements can have similar velocities to fixations, the simple velocity method can not differentiate smooth pursuits from fixations. The I-VDT algorithm integrates both I-VT and I-DT to classify fixations, saccades, and smooth pursuits. As in I-VVT, I-VDT first applies the velocity threshold to classify saccades and fixations, and then the dispersion threshold is applied to distinguish fixations and smooth pursuits. The drawback of these methods is that they are still threshold-based, so the user must manually set the optimum threshold values. Moreover, the identification process is done in two steps and requires two optimum threshold values.

Finding the optimum threshold value in threshold-based event detection methods is challenging as the threshold values vary with the type of task, data quality, and the user [83]. In order to address these issues, authors in [78] proposed an automated velocity threshold data-driven event classification method . The algorithm is able to find the threshold adaptively and avoid the influence of noise. Additionally, it identifies the glissades as separate event types. It is designed to overcome the noise sensitivity that occurs in previous algorithms by designing adaptive VT values considering different levels of noise occurrence. However, the glissades are detected based on duration only, which may lead to the situation that short saccades are classified as glissades or long glissades are classified as saccades. The algorithm can identify fixations, saccades and glissades only. It does not consider other event types, such as PSO and smooth pursuits.

The threshold-based event detection methods require a number of parameters that have to be adjusted based on eye movement data quality and finding the optimum threshold values is challenging. Machine learning and deep learning approaches can address these problems by applying end-to-end fully automated event detection without human intervention or without setting parameters like thresholds manually. These approaches are becoming popular in eye movement event detection. For instance, authors in [112] proposed a fully automated eye movement events classification using a Random Forest classifier. The method classifies input data into fixations, saccades, and PSOs. According to the authors, the machine learning approach outperforms other event detection methods. However, this method does not consider the frequent event type, which is smooth pursuit. The other fully automated event detection method was proposed in [44]. It is based on the deep Convolutional Neural Network that predicts probabilities for each sample to belong to a fixation, saccade, or smooth pursuit based on a sequence of gaze samples. The method tries to address the drawback of previous methods, which use signal shape and amplitude to classify the eye movement events, which may be problematic, for instance, for smooth pursuits. The proposed method uses the signal's frequency to classify the data into event types. The method classifies fixation, saccades, and smooth pursuits only. The comparison is done with velocity and dispersion threshold algorithms only.

Authors in [98] showed another example of applications of deep learning approaches in event detection. The proposed network is a joint of the 1D-Convolutional network and the BLSTM layer. It classifies the raw eye movement data into fixations, saccades, and smooth pursuits. Individual feature sets for the model are raw xy coordinates, speed, direction, and acceleration. However, the method exhibits poor performance when it takes a combination of parameters. According to the authors, the combination of direction and speed showed a noticeable improvement over using them separately. Acceleration as an additional feature did not improve average detection performance, probably due to its inability to distinguish smooth pursuits from fixations. The algorithm identifies fixations, saccades, and smooth pursuit. It does not consider PSO event types. Another deep learning approach in [23] is proposed for online event event classification for fixations, saccades and smooth pursuits.

To summarize, there are many attempts to utilize machine learning and deep learning for eye movement event detection. However, to the authors' knowledge, none of them tried to combine four eye movement features used in this work, and none of them tried to build a model that simultaneously classifies eye movement data into four event types: fixations, saccades, smooth pursuits, and post-saccadic oscillations.

Chapter 3

Comparative Analysis of Eye-Tracking Data Across Different Visual Tasks

3.1 Introduction

Understanding eye movement behavior across diverse visual tasks provides crucial insight into both cognitive processing and oculomotor control. Different tasks—such as reading, static image viewing, dynamic video watching, and moving-dot tracking—elicit distinct patterns of fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs). These task-dependent variations arise from differences in perceptual demands, attentional allocation, stimulus dynamics, and motor strategies, making cross-task analysis particularly informative.

Previous research has demonstrated that eye movement behavior is strongly modulated by task demands and cognitive goals. Reading, scene viewing, dynamic video observation, and target tracking each produce characteristic oculomotor signatures in terms of fixation durations, saccade amplitudes, and velocity profiles [85, 103, 21]. Importantly, these differences are systematic rather than incidental, reflecting underlying perceptual and cognitive constraints that shape gaze control [49, 45]. Accordingly, a detailed statistical characterization of eye movement events across tasks provides essential insight into how gaze behavior adapts to different visual and cognitive contexts. Such analysis establishes a necessary empirical foundation for the development, evaluation, and interpretation of task-general computational models of eye movement event detection.

This chapter presents a comprehensive analysis of eye movement data collected during four distinct visual tasks: reading, image viewing, video watching, and moving-dot tracking. The datasets were drawn from publicly available sources: the CopCo dataset for reading, the Lund2013 dataset for image viewing, the Video Watching dataset for

dynamic tasks, and the Smooth Pursuit Eye Movement dataset for moving-dot tracking. Each dataset was recorded using different eye-tracking devices and annotated either manually or with software, with variation in annotation granularity. Specifically, some datasets contained only fixations and saccades, while others included additional labels for PSOs and smooth pursuits. These differences in recording protocols and annotation detail introduce challenges for direct comparisons; however, the focus of this analysis is on identifying generalizable behavioral patterns across tasks.

The analysis focuses on characterizing eye movement patterns across tasks, including scanpaths and event amplitudes, as well as event-level metrics such as the duration, frequency, and velocity of fixations, saccades, PSOs, and smooth pursuits. By examining both the overall patterns of gaze behavior and the quantitative properties of individual events, we highlight how oculomotor dynamics are shaped by the demands of different visual tasks.

The results presented here not only characterize task-specific eye movement behavior but also establish an empirical foundation for evaluating the performance of event detection methods. In particular, the observed overlap between event types (e.g., fixations and smooth pursuits, or saccades and PSOs) demonstrates why accurate detection is challenging and why feature selection and model design are critical in addressing these challenges.

3.2 Methodology

3.2.1 Datasets

This study draws on two complementary eye-tracking datasets: the Lund2013 dataset [2], hereafter referred to as MN-RA, and the CopCo eye-tracking corpus [39]. Together, these datasets cover four visual tasks: Reading, *Image Viewing*, *Video Watching*, and *Moving Dot Tracking*. They were used for two complementary purposes: (i) statistical characterization of eye movement behavior (velocity profiles, event statistics, scanpaths), and (ii) training and evaluation of machine-learning models for event classification.

Lund2013 Dataset (MN-RA)

The first dataset, Lund2013, is a publicly available eye-tracking dataset recorded using a Hi-Speed 1250 eye tracker from SensoMotoric Instruments (SMI, Teltow, Germany) at a sampling rate of 500 Hz [2]. Participants were presented with four types of stimuli: static images, text passages, short video clips, and simple moving dot stimuli.

All data were manually annotated by two expert raters, Marcus Nyström (MN) and Richard Andersson (RA), into the following categories: fixations, saccades, post-saccadic

oscillations (PSOs), smooth pursuits (SPs), blinks, and undefined movements. For the purposes of this study, we focused on the subsets corresponding to:

- **Image Viewing**: Labeled with fixations, saccades, and PSOs. This subset captures exploratory scanpaths guided by visual saliency and scene semantics. Seven image files were used for model training.

- **Video Watching**: Labeled with fixations, saccades, PSOs, and SPs. This condition introduces dynamic stimuli and provides naturalistic combinations of event types.

- **Moving Dot Tracking**: Labeled with fixations, saccades, PSOs, and SPs. This condition provides controlled smooth pursuit segments and corrective saccades under predictable motion trajectories.

The Lund2013 dataset was central to the machine learning experiments, as it provides rich manual labels across multiple event categories. For model training and evaluation, we used the harmonized subset of three classes (**fixation, saccade, PSO**), excluding SPs to maintain compatibility across all conditions. The dataset is openly available at: https://github.com/richardandersson/EyeMovementDetectorEvaluation.

CopCo Corpus

The second dataset is the CopCo eye-tracking corpus, designed for psycholinguistics and natural language processing research [39]. The data were collected with an EyeLink 1000 system (SR Research) at a sampling rate of 1000 Hz. The corpus includes reading behavior in Danish texts across different participant groups: (i) readers without dyslexia, (ii) readers with dyslexia, and (iii) non-native speakers. In total, the dataset comprises 58 participants (17 men and 41 women).

For this study, we used data from 13 healthy participants without dyslexia. Eye movements were segmented into fixations and saccades using SR Research software. To ensure comparability with the MN–RA dataset. This dataset provides high-quality reading data with structured scanpaths but lacks PSO and SP annotations, and was therefore excluded from the machine learning experiments while being fully integrated into the statistical analyses.

The CopCo corpus is openly available at: https://osf.io/ud8s5/.

All four visual tasks (reading, image viewing, video watching, moving dot tracking) were included in the statistical analyses (Sections 3.4.1 and 3.4.2) to characterize cross-task oculomotor behavior. For machine learning experiments, only the MN–RA subsets (*image viewing*, *video watching*, *moving dot tracking*) were used, because they provide the necessary event categories for multi-class classification.

To harmonize across tasks, the label set was restricted to Fixation, Saccade, and PSO, with SPs excluded (absent in the image-viewing data), ensuring a consistent three-class taxonomy across all modeling conditions.

3.3 Task-Specific Eye Movement Behavior

In this section, we describe the raw scanpath patterns and oculomotor behaviors observed across the four experimental paradigms: reading, image viewing, video watching, and moving dot tracking. Each task imposes distinct perceptual and cognitive demands, which are reflected in characteristic distributions of fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs). By examining these task-dependent differences, we establish the behavioral foundation for the subsequent quantitative velocity and event analyses.

3.3.1 Reading Task

The reading task, derived from the CopCo eye-tracking dataset, is characterized by a sequential processing of textual information. Eye movements during reading primarily consist of fixations and saccades, which are indicative of the cognitive effort involved in text comprehension.

Reading is a structured, cognitively demanding activity involving the decoding and integration of textual information. It is characterized predominantly by sequences of fixations and saccades along horizontal lines. Fixations typically last between 200–250 ms, though this duration can vary based on word frequency, syntactic complexity, and reader skill [85, 86]. Saccades in reading are brief, and regressions (backward saccades) occur frequently when comprehension breaks down or for reanalysis.

Reading tasks emphasize Predictable scanpaths: Most movements are horizontal and linear, reflecting line-by-line progression. Low occurrence of smooth pursuit or PSOs: The stimuli (text) are static, and eye motion is driven purely by cognitive processing, not moving targets. High fixation count: Indicates close visual scrutiny and cognitive load during lexical access and integration [87, 50].

In the reading task, scan paths in Figure 3.2 (a) display a structured linear progression across lines of text.

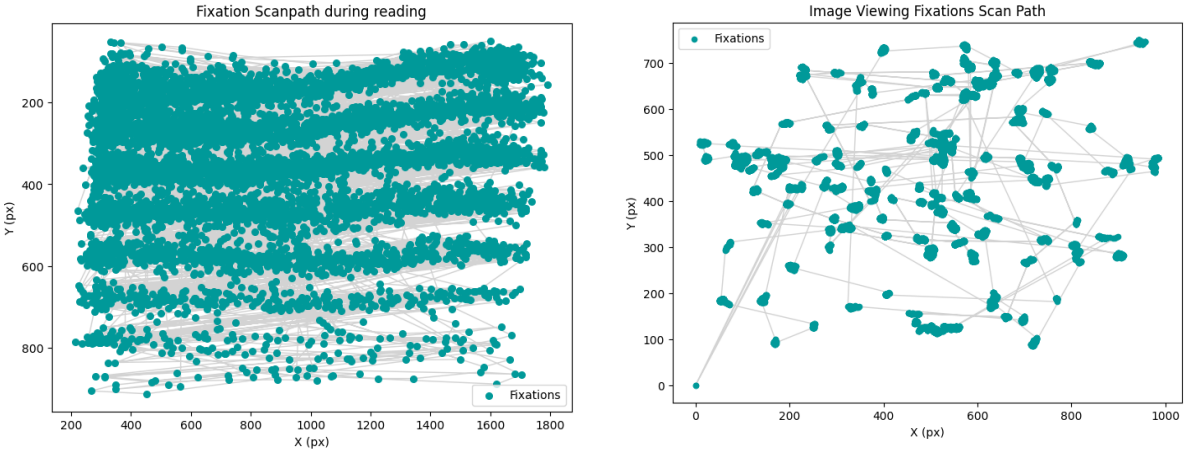
3.3.2 Image Viewing Task

Image viewing, in contrast, is an exploratory task where the observer's attention is guided by visual saliency, scene semantics, and individual interest. Eye movement patterns are less structured and more variable, both across individuals and within a session. Saccades tend to be larger, fixations shorter (100–150 ms), and viewing is distributed across

a wider spatial area [101, 46, 36]. Compared to reading, image viewing elicits larger and more variable saccades, shorter fixations, and less predictable scanpaths, reflecting the absence of a prescribed viewing order [37, 101].

For this analysis we used the MN-RA data set recorded during image viewing and manually labeled into fixations, saccades and PSO by two experts MN and RA. It provides insight into the exploratory nature of image viewing, where attention is directed toward visually salient features of the scene. This task typically involves rapid shifts of gaze, reflecting the need to gather spatial information from different regions of the image.

In Figure 3.1 (b) the scanpaths are markedly less constrained and exhibit a wide spatial dispersion. The key findings from this data set are as follows: Fixations are shorter and saccades are longer and more exploratory, indicating a visual scanning strategy aimed at gathering information from various focal points. The variability in gaze patterns reflects both the content of the image and individual differences in attention.



(a) (b)

Figure 3.1: Example of eye movement patterns for (a) Reading movement pattern. (b) Image viewing pattern

Key characteristics of image viewing task

- **Scene-dependent gaze behavior:** Observers focus on regions of high saliency.
- **Increased variability in fixation duration and saccade length.**
- **Potential for post-saccadic oscillations (PSOs),** especially if saccades land on ambiguous or complex areas.

3.3.3 Video Watching Task

Video watching introduces temporal dynamics and moving stimuli, engaging both reflexive and volitional components of gaze control. The eye movement profile includes

fixations, saccades, smooth pursuits (SPs), and PSOs. Viewers tend to fixate on moving objects or socially relevant regions and follow their motion across frames. Previous studies of naturalistic video viewing have shown that eye movements are strongly shaped by motion cues, scene transitions, and socially relevant content, resulting in synchronized viewing patterns across observers [19, 74]. The coexistence of fixations, saccades, smooth pursuits, and PSOs reflects the need to balance attentional stability with continuous tracking of dynamic visual information [96].

Key characteristics of video watching task

- **Smooth pursuit becomes prominent:** Eyes match object motion to maintain foveation.
- **Temporal alignment with scene changes:** Saccadic eye movements frequently align with cinematic transitions.
- **Increased occurrence of PSOs:** Rapid scene changes or motion discontinuities can trigger PSOs.

The video watching task in this section, derived from a dynamic video dataset, captures eye movements as participants track moving objects within the video. This task combines fixations, saccades, smooth pursuits, and PSOs to adapt to changing stimuli. In the video watching task, the scanpath visualization (Figure 3.1 a) illustrates the dynamic and adaptive nature of gaze behavior during exposure to moving visual content. The fixation points are clustered around salient, moving elements within the scene, reflecting the viewer's attempts to anchor attention to objects of interest. The connecting saccades, shown as directional lines, highlight rapid shifts in gaze as the viewer reorients to new stimuli, often triggered by motion or scene changes. Notably, the presence of smooth, curved trajectories between some fixations suggests periods of smooth pursuit.

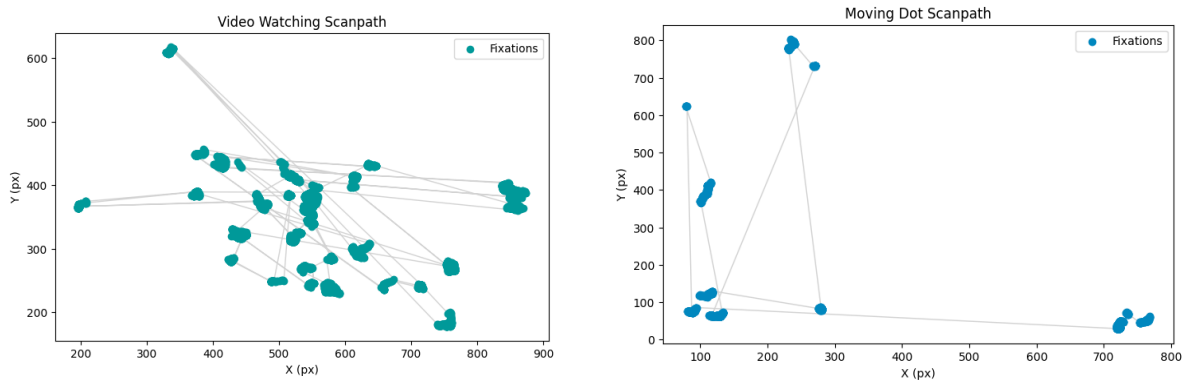
3.3.4 Moving Dot Tracking Task

In the moving dot tracking task, the scanpath visualization (Figure 3.2 b) shows a smooth and continuous gaze trajectory that closely mirrors the motion path of the dot stimulus. Unlike the more fragmented scanpaths observed in other tasks, this pattern is characterized by elongated, curved segments indicative of sustained smooth pursuit movements. Fixation points are sparse and typically appear during brief pauses in dot movement or transitional moments requiring gaze recalibration. The occasional presence of short saccades, visible as sharp directional jumps in the scanpath, reflects corrective actions when smooth pursuit is momentarily disrupted. Controlled moving-dot paradigms have long served as benchmarks for studying smooth pursuit eye movements, eliciting

sustained low-velocity tracking with occasional corrective saccades when pursuit gain decreases or target motion changes abruptly [84, 66].

Distinctive properties of moving dot tracking task

- **Smooth pursuits dominate:** Continuous, low-velocity eye movements are the primary behavior.
- **Minimal fixations:** Fixations are rare and typically occur only during trajectory pauses or abrupt changes.
- **Lower inter-subject variability:** The uniformity of the stimulus and the explicit nature of the tracking task result in more consistent gaze patterns.



(a) (b)

Figure 3.2: Example of eye movement patterns for (a) Video watching scan path pattern. (b) Moving dot tracking scan path pattern

Analyzing these four tasks reveals fundamental differences in eye movement strategies. Eye movement patterns are linear during reading, exploratory during image viewing, and dominated by sustained smooth pursuit with occasional corrective saccades during moving dot tracking. Video watching involves a mixture of fixations, saccades, smooth pursuits, and PSOs reflecting the dynamic nature of the task.

3.4 Statistical Analysis

This section presents a quantitative characterization of eye movement behavior across the four visual tasks. We examined key kinematic and temporal features of fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs) using descriptive statistics. The analyses focus on velocity distributions, event duration, and event point counts to reveal task-dependent differences in ocular motor dynamics.

3.4.1 Velocity Profile Analysis

Eye movement velocity is one of the most informative kinematic markers of oculomotor behavior and is routinely used to distinguish between different eye movement event types and has been extensively studied across different movement types. Saccades are characterized by rapid, ballistic velocity profiles, whereas fixations and smooth pursuits exhibit substantially lower velocities [64]. Importantly, saccadic velocity is not fixed but varies systematically with both movement amplitude and task demands [4, 5].

In this section, we demonstrated clear task-dependent differences in oculomotor dynamics by analyzing the velocity profiles of fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs). Previous research on motor planning models have shown that eye movement velocity is adjusted in response to task-dependent uncertainty and noise, allowing flexible control of movement speed across different contexts [32]. More recent evidence further indicates that increased cognitive workload can reduce saccadic peak velocity, highlighting the sensitivity of oculomotor dynamics to task demands beyond purely spatial factors [16].

To characterize the kinematic properties of eye movements across the four visual tasks, we analyzed the velocity profiles of fixations, saccades, PSOs, and SPs. Because velocity distributions are often skewed and contain extreme values particularly for saccades and PSOs we report the mean velocity, standard deviation and key distribution percentiles (P5, P25, P50, P75, P95). These metrics provide a detailed description of both the central tendency and spread of the velocity distribution, enabling robust comparisons across tasks. Velocity (V) was calculated as the Euclidean distance between successive gaze coordinates (x_i, y_i) and (x_{i+1}, y_{i+1}), di vided by the fixed sampling interval (Δt) in milliseconds:

$$V_i = \frac{\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}}{\Delta t}, \qquad \Delta t = \frac{1000}{f_s}, \tag{3.1}$$

where f_s is the sampling rate in Hz. Equation (3.1) ensures velocity is expressed in pixels per millisecond (px/ms) and directly reflects spatial displacement over time. Sequential duplicate samples were removed prior to computing V_i to minimize zero-velocity artifacts and tracker-induced repetition.

Table 3.1 presents the descriptive statistics of velocity distributions, including the mean, standard deviation, and selected percentiles (P5, P25, P50, P75, and P95) for each event type and task.

Fixations: Fixation velocities were consistently the lowest across all tasks, reflecting the relative stability of gaze during visual information extraction. The reading task exhibited the highest average fixation velocity (mean = 0.93 px/ms), whereas image viewing showed the lowest (mean = 0.39 px/ms), with video watching and moving-dot tracking displaying intermediate values. Percentile-based distributions (Table 3.1) demonstrate that

Table 3.1: Velocity statistics (px/ms) for each eye movement type across tasks. Values represent mean, standard deviation (Std), and percentiles of the velocity distributions.

Task	**Event**	**Mean**	**Std**	**P5**	**P25**	**P50**	**P75**	**P95**
Reading	Fixation	0.93	3.14	0.20	0.45	0.78	1.22	2.11
	Saccade	**9.08**	**15.25**	**0.71**	**2.75**	**6.53**	**12.73**	**23.95**
Image Viewing	Fixation	0.39	3.62	0.07	0.17	0.28	0.42	0.72
	Saccade	6.60	6.50	0.61	1.99	4.58	9.33	17.28
	PSO	1.65	1.75	0.23	0.65	1.14	2.00	4.85
Video Watching	Fixation	0.57	1.12	0.09	0.23	0.36	0.56	1.28
	Saccade	6.30	5.60	0.60	2.02	4.56	9.09	17.26
	PSO	2.16	2.78	0.30	0.70	1.28	2.41	8.79
	SP	0.52	0.70	0.09	0.22	0.37	0.57	1.25
Moving Dot	Fixation	0.61	10.31	0.08	0.16	0.26	0.39	0.65
	Saccade	3.27	3.65	0.26	0.83	1.73	4.23	11.83
	PSO	1.96	1.78	0.28	0.51	1.48	2.57	5.30
	SP	1.00	16.10	0.08	0.19	0.31	0.47	0.89

fixation velocities were tightly clustered in the lower range for all tasks, with comparatively narrow interquartile spans and limited upper tails, particularly for static viewing conditions. In contrast, reading showed a broader distribution and a heavier upper tail, indicating more frequent micro-corrective movements during sequential lexical processing.

Saccades: Saccades showed the highest velocities and the highest variability across tasks. Reading produced the fastest saccades (mean = 9.08 px/ms; P95 = 23.95 px/ms), followed by image viewing (mean = 6.60 px/ms) and video watching (mean = 6.30 px/ms). The moving dot task showed substantially slower saccades (mean = 3.27 px/ms), suggesting that this task is dominated by corrective low-amplitude saccades rather than long ballistic gaze shifts. The percentile spreads demonstrate large right-skewed distributions with high-velocity tails that reflect rapid, goal-directed ballistic movements. While saccades retain a qualitatively high-velocity ballistic profile across tasks, their quantitative velocity distributions vary systematically with task structure, stimulus dynamics, and cognitive demands. This is consistent with prior work showing that saccadic kinematics are not fixed, but adapt to behavioral goals and viewing conditions [5].

Post-saccadic oscillations (PSOs): PSO velocities were consistently lower than saccades but higher than fixations, reflecting their role as fine corrective movements following saccadic landing. Image viewing showed the lowest PSO velocities (mean = 1.65 px/ms), while video and moving-dot tasks showed slightly higher values (means = 2.16 and 1.96 px/ms, respectively). The Percentile ranges indicate that most PSO velocities fall between 0.2–4.8 px/ms, which overlaps with both the fixation and the low-velocity pursuit ranges.

Smooth pursuits (SPs): Smooth pursuits were present only in dynamic tasks. Video

watching produced slow and stable SPs (mean = 0.52 px/ms), whereas moving-dot tracking showed faster pursuits (mean = 1.00 px/ms) and a wider distribution, reflecting rapid target movement and corrective adjustments. Percentile values confirm that most SP velocities remain below 1.25 px/ms.

Overall, the velocity distributions demonstrate clear task-dependent modulation of oculomotor kinematics. Reading was characterized by relatively fast saccades and moderately elevated fixation velocities, reflecting the rapid spatial reorientations and frequent micro-adjustments required for sequential text processing. Image viewing showed slower and more stable fixations with intermediate saccadic velocities, consistent with sustained inspection of static visual content. Video watching exhibited a combination of slow smooth pursuits and medium-velocity saccades, reflecting the balance between continuous tracking and discrete corrective movements in dynamic scenes. Moving-dot tracking was dominated by faster smooth pursuits accompanied by slower, corrective saccades.

The moving-dot task showed disproportionately large standard deviations relative to its mean and median velocities, particularly for fixations and smooth pursuits. This apparent discrepancy reflects the uneven distribution of velocities in dynamic tracking tasks. Percentile analyses revealed that the vast majority of velocity samples in the moving-dot condition remained confined to a narrow low-velocity range (P95 $<$ 1 px/ms for fixations and pursuits), while a small number of transient high-velocity events produced an extended right tail. These rare but extreme velocities correspond to rapid corrective saccades that intermittently interrupt smooth pursuit when the eye lags behind the moving target. Consequently, the increased standard deviation in the moving-dot task reflects alternations between slow continuous tracking and brief corrective saccades, providing an informative indicator of task-specific gaze dynamics.

Eye Movement Velocity Patterns Across Visual Tasks

To further characterize visual behavior, velocity signals were visualized across tasks to examine the temporal structure of ocular motor activity under different perceptual and cognitive demands. The velocity profiles reveal clear task-dependent patterns, consistent with the statistical distributions reported in Table 3.1.

For clarity of visualization, velocity profiles for static tasks (reading and image viewing) are shown in Figure 3.3, while velocity profiles for dynamic tasks (video watching and moving-dot tracking) are shown in Figure 3.4. Plotting the velocity over extended sample indices allows the temporal structure and recurrence of velocity patterns to be observed more clearly, highlighting task-specific characteristics of eye movements.

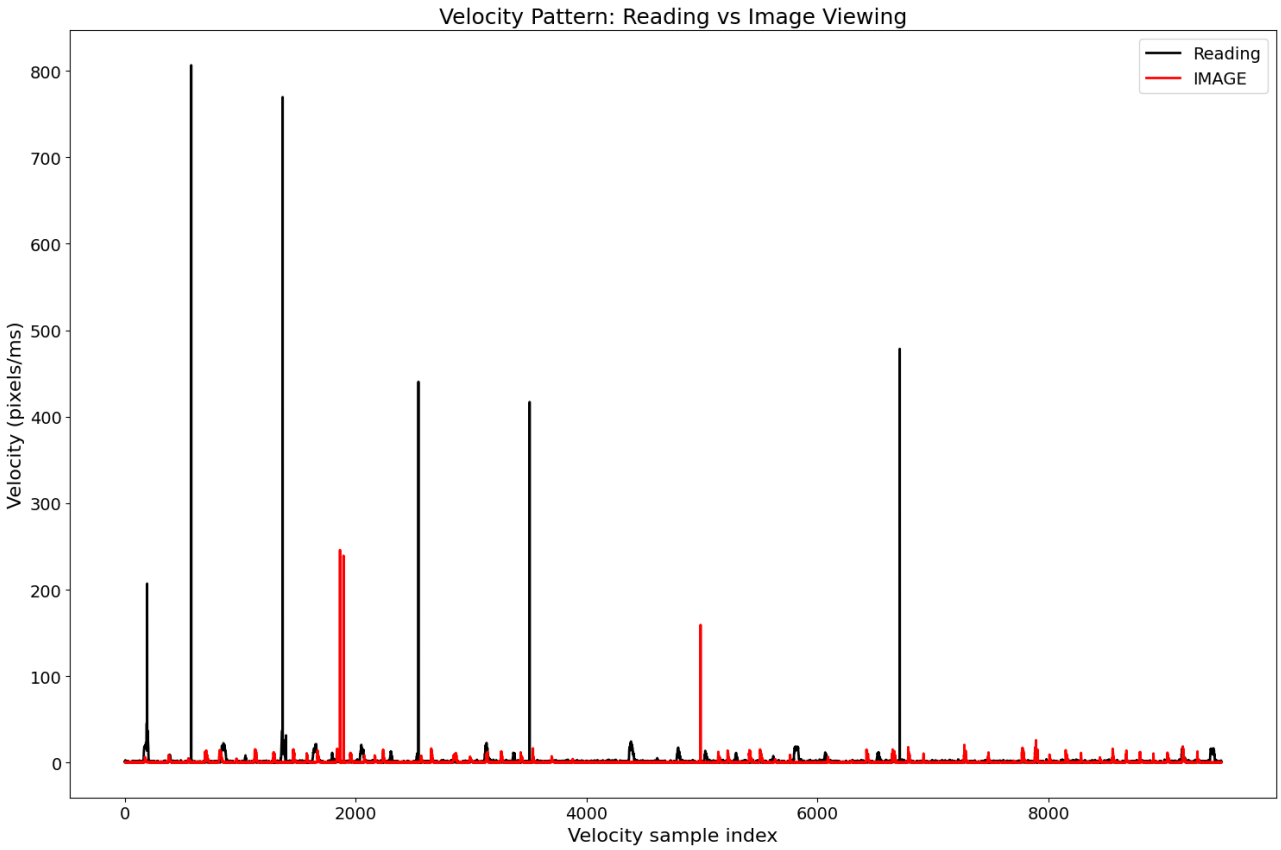


Figure 3.3: Velocity patterns for reading and image viewing tasks.

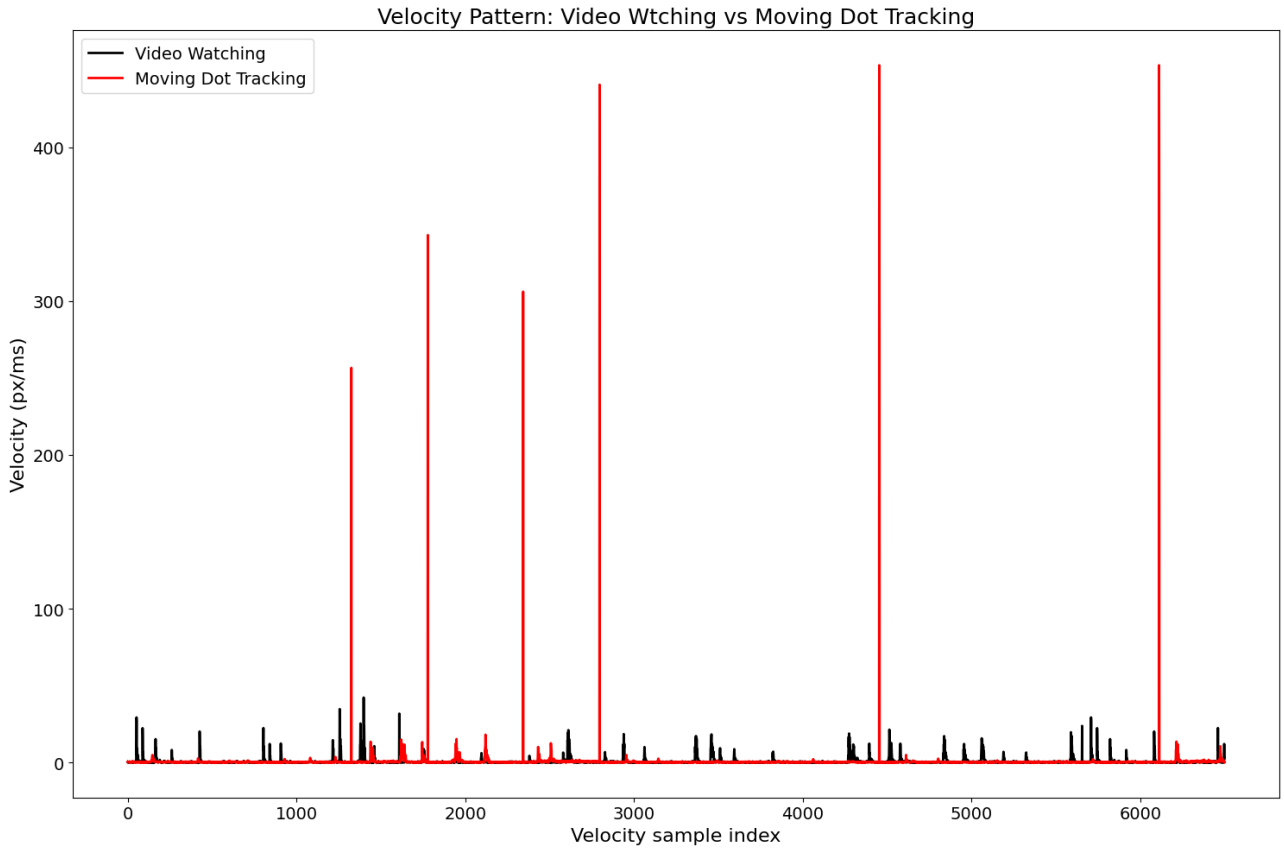


Figure 3.4: Velocity patterns for video watching and moving-dot tracking tasks.

As shown in the Figure 3.3, In the Image Viewing task, the velocity amplitudes show moderate fluctuations, with occasional high-velocity peaks. These peaks are indicative of saccadic movements or potential post-saccadic oscillations, occurring as the eye shifts between points of interest within the image reflecting the exploratory nature of image viewing and the variability in fixation duration driven by visual content.

In contrast, the Reading task demonstrates a more structured and rhythmic velocity pattern. Frequent, regularly spaced velocity peaks are observed, corresponding to saccades between words and characters, while larger velocity spikes appear intermittently and are associated with return-sweep saccades when transitioning between lines of text.The consistent spacing and repetition of these velocity patterns indicate structured eye movements and longer, more stable fixation periods inherent to reading.This pattern reflects the structured and repetitive nature of reading.

As shown in Figure 3.4, the Video Watching task exhibits a relatively uniform velocity profile over time, characterized by smooth transitions and fewer abrupt velocity changes. This pattern reflects continuous visual engagement with predictable motion, where gaze shifts are largely guided by the content of the video.

In contrast, the Moving Dot Tracking task displays frequent and irregular velocity fluctuations, manifested as repeated velocity spikes across the temporal sequence. These fluctuations are indicative of continuous corrective movements, including catch-up sac-

cades, as the visual system attempts to maintain alignment with the moving target. The lack of regular spacing between spikes highlights the reactive and adjustment-driven nature of this task.

3.4.2 Event Statistics

To better understand the distribution and temporal characteristics of different eye movement events across visual scenes, we analyzed the number event points and duration of events for each task. Table 3.2 summarizes the event counts and duration statistics.

Table 3.2: Event Point Counts and Duration Statistics (Mean and (SD))

Task	**Event Type**	**Event Point Count**	**Mean Duration (ms)**	**SD (ms)**
Reading	Fixation	1,413,801	198.67	88.05
	Saccade	280,929	36.47	18.97
	PSO	–	–	–
	SP	–	–	–
Image Viewing	Fixation	24,400	240.39	205.75
	Saccade	2,803	27.63	14.90
	PSO	1,730	19.01	8.50
	SP	–	–	–
Video Watching	Fixation	6,177	224.65	213.93
	Saccade	1,156	24.86	15.77
	PSO	529	16.03	7.11
	SP	9,089	443.37	268.22
Moving Dot	Fixation	1,104	184.00	103.88
	Saccade	332	25.54	12.64
	PSO	97	14.92	9.58
	SP	5,033	419.50	327.36

Event Duration Analysis Across Tasks

Event durations were analyzed for different tasks accross participants. For Reading, data from four participants were used, while Image Viewing, Video Watching, and Moving Dot Tracking included seven, six, and seven participants, respectively. For each task, datasets from all participants were treated as a single dataset, and the variability is reported as the standard deviation (SD) across all events within that task.

Reading fixations had a mean duration of 198.67 $\pm$ 88.05 ms, showing sustained attention for sequential text processing. Saccades were also relatively long in Reading (36.47 $\pm$ 18.97 ms). PSOs and smooth pursuits were absent in Reading.

Image Viewing fixations were longer on average (240.39 $\pm$ 205.75 ms) than Reading, but with much higher variability, indicating that some fixations were very short while

others were much longer depending on the visual content. Saccades in Image Viewing were shorter (27.63 ± 14.90 ms), and PSOs were present with mean duration 19.01 ± 8.50 ms. Smooth pursuits were not observed.

Video Watching fixations averaged 224.65 ± 213.93 ms, with high within-task variability. Saccades averaged 24.86 ± 15.77 ms, PSOs 16.03 ± 7.11 ms, and smooth pursuits were observed with mean duration 443.37 ± 268.22 ms, reflecting sustained tracking of dynamic targets.

Moving Dot Tracking fixations were shorter (184.00 ± 103.88 ms), saccades averaged 25.54 ± 12.64 ms, PSOs 14.92 ± 9.58 ms, and smooth pursuits 419.50 ± 327.36 ms, again showing high variability depending on participant behavior and target motion.

Cross-Task Duration Comparison: Across tasks, mean fixation durations were longest in Image Viewing, followed by Video Watching, Reading, and Moving Dot Tracking. Reading fixations, however, were more consistent (lower SD) than those in other tasks, indicating a more stable allocation of visual attention. Saccades were consistently longest in Reading and shortest in Moving Dot Tracking. PSOs were shorter than fixations and saccades and appeared primarily in non-reading tasks. Smooth pursuits occurred only in dynamic tasks and exhibited high variability, reflecting differences in continuous-tracking demands across trials.

Overall, event durations varied substantially across tasks. Both the mean duration and within-task variability of eye movement events were clearly task-dependent, highlighting the influence of cognitive and perceptual demands associated with different visual tasks.

Count Analysis

Figure 3.5 illustrates the proportion of each eye movement event points type across tasks, highlighting clear differences in event distribution. Overall, fixation points were the most frequent event type across all tasks, underscoring their foundational role in visual processing. Saccade points were less frequent but present in all tasks, enabling gaze shifts between visual targets. PSOs were observed in all tasks except Reading, and SPs occurred only in dynamic visual contexts (Video Watching and Moving Dot Tracking).

Reading: Reading involved with two types of eye movement events dominated by fixation points (1.41 million) and saccade points (281k). This pattern reflects the sequential nature of reading, where frequent pauses are required to process text. PSOs and SPs were negligible.

Image Viewing: Combined event counts were much lower (29k total), with fixation points (24k) greatly outnumbering saccades (2.8k) and PSOs (1.7k). This indicates that static image inspection involves relatively fewer eye movements, with a strong emphasis on fixational analysis. SPs were absent.

Video Watching: Eye movements were more balanced, including fixations (6k), saccades

(1.1k), PSOs (529), and SPs (9k). Notably, SPs exceeded saccades in count, reflecting the continuous motion in video stimuli that engages smooth pursuit tracking.

Moving Dot Tracking: SPs (5k) dominated, greatly outnumbering saccades (332) and PSOs (97), while fixations (1.1k) remained low. This aligns with task demands, where following a moving target primarily requires smooth pursuit rather than frequent saccades or prolonged fixations.

Cross-Task Observation: Across tasks, fixation points dominated in static tasks (Reading and Image Viewing), while SPs dominated in dynamic tasks (Video Watching and Moving Dot Tracking). Saccade points occurred in all tasks but were proportionally lower in motion-driven tasks. PSOs were generally low across tasks but slightly more frequent in dynamic tasks, highlighting their role in stabilizing gaze following saccades in visually complex or moving scenes. These patterns confirm that task type strongly shapes the distribution of eye movement events.

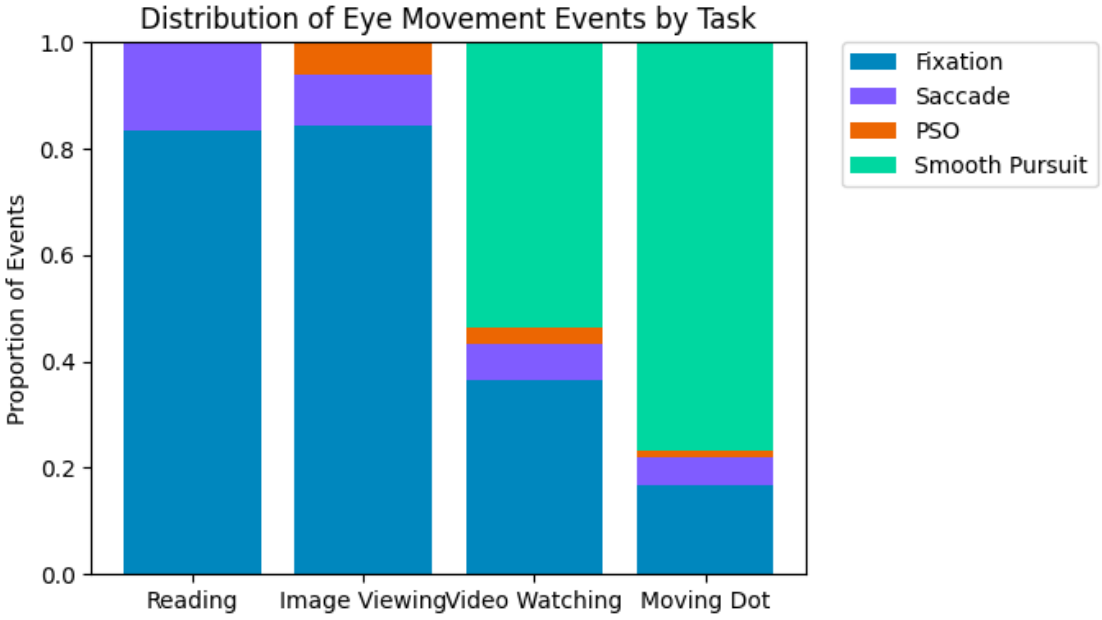


Figure 3.5: The proportion of each eye movement type across tasks.

3.4.3 Overlap in Eye Movement Characteristics and Its Impact on Classification

Although eye movement events are commonly categorized as fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs), their measurable kinematic properties frequently overlap in real data, reducing the separability of these classes. This effect is particularly evident in velocity-based features, which are widely used in both human annotation and automated classification.

Using the empirical velocity distributions (Table 3.1), substantial overlap is evident

between multiple event categories across tasks. For example, fixations and smooth pursuits (SPs) occupy highly similar velocity ranges. In the video task, fixation velocities show upper-tail values (P95 $\approx$ 1.28 px/ms) that overlap directly with the lower-to-median range of SP velocities (which remain below approximately 1.25 px/ms). A similar pattern is observed in the moving-dot task, where fixation and SP velocity percentiles strongly overlap, making these two event types difficult to separate using velocity alone.

Overlap is also present between small-amplitude saccades and PSOs. While saccades are characterized by higher mean velocities overall, the lower percentile ranges of saccades intersect with the upper ranges of PSOs. For example, in the video task, the lower portion of the saccade velocity distribution (P5–P25) overlaps with the upper percentile range of PSOs (P75–P95). This creates a transitional velocity region where short corrective saccades and PSOs become difficult to distinguish.

Even within the same task, the distributions of velocity for different events are not disjoint. In dynamic tasks such as video watching and moving-dot tracking, the tails of fixation, SP, and PSO velocity distributions overlap substantially. This results in ambiguous movement segments, particularly during rapid alternations between pursuit and corrective saccades.

These overlaps arise from both biological variability in oculomotor control and measurement constraints of video-based eye trackers. The limited spatial and temporal resolution of the tracker introduces noise and smoothing effects that widen the empirical velocity distributions and blur event boundaries. As a result, classifiers relying solely on instantaneous velocity or acceleration thresholds are prone to systematic confusion between event classes.

Such overlaps in kinematic and temporal characteristics have been repeatedly identified as a major challenge for both manual annotation and automated eye movement classification. Previous studies have shown that fixation–smooth pursuit and saccade–PSO confusions are particularly prevalent in dynamic tasks, necessitating models that incorporate temporal context and multiple motion features [78, 56].

3.5 Conclusion

In this chapter, we systematically analyzed the behavior of eye movement in four distinct tasks: reading, image viewing, video watching, and moving-dot tracking. The analysis combined statistical metrics, velocity profiles, and scanpath visualizations to characterize the temporal, spatial, and dynamic properties of major eye-movement events.

The results revealed clear task-dependent oculomotor patterns. Reading produced highly structured, linear scanpaths dominated by frequent fixations and short saccades. Image viewing elicited more exploratory behavior, with dispersed fixations and longer saccadic transitions. Video watching involved a mixture of event types, particularly smooth

pursuits, reflecting the demands of tracking continuously moving visual content. Moving-dot tracking was dominated by smooth pursuits, generating continuous trajectories closely aligned with the target motion. Post-saccadic oscillations were observed across tasks and exhibited substantial variability in both frequency and velocity, highlighting their context-sensitive nature.

These results confirm that oculomotor dynamics is shaped by the demands of the visual task and that overlaps between event types present a fundamental challenge for event detection. By providing a detailed statistical and behavioral characterization of task-dependent eye movements, this chapter establishes the empirical foundation for the development of automated eye movement event detection methods.

Chapter 4

Evaluation of Eye Movement Event Detection Algorithms

4.1 Introduction

This chapter presents a comprehensive evaluation of eye movement event detection algorithms, including both classical threshold-based approaches and modern machine learning and deep learning models. While this work builds upon our previously published study [7], it has been extended to include the introduction of a Long Short-Term Memory (LSTM) network to the evaluation, the implementation of leave-one-file-out cross-validation (LOFO-CV) for the Convolutional Neural Network (CNN), LSTM, and Random Forest (RF) classifiers, as well as a comparative analysis of models trained with and without cross-validation. Furthermore, this chapter includes an expanded discussion of how evaluation methodologies influence model performance and generalizability.

The detection of eye movement events is fundamental to understanding gaze behavior in various contexts, such as reading, viewing images, and observation of dynamic scenes [41]. Historically, threshold-based algorithms have served as the dominant approach due to their simplicity and computational efficiency [91, 55]. However, these methods are often limited by their sensitivity to parameter settings and their lack of generalizability across participants and experimental conditions. To overcome these challenges, recent advances have focused on machine learning and deep learning approaches that can automatically learn discriminative patterns directly from data, thereby improving adaptability and robustness.

Despite the growing number of proposed methods, the lack of a standardized evaluation procedure for eye movement event detection continues to pose a major challenge. Previous work has emphasized the importance of evaluating algorithms under consistent conditions and using comparable metrics to ensure fair and interpretable results [99]. Following this principle, the current study employs sample-by-sample comparisons and inter-agreement

metrics against manually annotated data. The manual annotations serve as the ground truth reference for quantifying the classification accuracy of each algorithm.

Five algorithms were implemented and evaluated: Identification by Velocity Threshold (I-VT), Identification by Dispersion Threshold (I-DT), Random Forest (RF), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM) networks. The I-VT and I-DT algorithms, which are designed for binary classification, were used exclusively to identify fixations and saccades. In contrast, the RF, CNN, and LSTM models were able to classify all three event types: fixations, saccades, and post-saccadic oscillations (PSO) or smooth pursuit (SP) [112, 98]. For the threshold-based methods, the influence of varying threshold values was systematically analyzed to determine optimal parameters, while the learning-based models were evaluated both with and without cross-validation to assess their robustness and generalization across participants and recordings.

The dataset used in this evaluation consists of manually annotated image-viewing stimuli, labeled into three event categories: fixations, saccades, and PSOs. Through this comparative analysis, the chapter aims to highlight the strengths and limitations of threshold-based, machine learning, and deep learning approaches, providing a comprehensive perspective on their performance and applicability in eye movement research.

4.2 Manual Human Classification

Manual classification of eye movement events involves trained coders segmenting raw gaze data into fixations, saccades, smooth pursuits, and other event based on subjective threshold values. Manual classification is still a common method for evaluating event detection algorithms and is treated as a "golden standard". Manually classified data are frequently used as training data for machine learning algorithms. However, manual event classification is not an effective way to classify events and it has critical limitations.

4.2.1 Limitations of Manual Human Classification

Manual classification of eye movement data, while often used as a reference standard, presents several limitations:

- **Subjectivity:** Coders may apply different thresholds or interpret ambiguous segments differently, especially in cases of borderline velocities or overlapping signal profiles [41]. This may lead to diffrent event classification. For example, the authors of [43] used twelve experienced but untrained human coders to classify events in six minutes of eye-tracking data and found substantial differences between the classifications when average fixation duration and number of fixations were compared [1].

- **Inter-rater variability:** Studies have shown that agreement between coders can range significantly. Andersson et al. [2] report that Cohen's Kappa values, commonly used to quantify agreement, are often below 0.75 in complex eye-tracking classification tasks.

- **Time-intensive:** Labeling even a few minutes of high-frequency data can take hours, making it inefficient for large-scale datasets [41].

- **Cognitive bias:** Human coders may subconsciously apply expectations about the stimulus, leading to biased segmentation. For instance, in reading tasks, they may incorrectly assume fixations follow a strictly linear progression [2].

Despite these challenges, manual classification remains indispensable for creating high-quality ground truth datasets in eye-tracking research. It is often used to benchmark the accuracy of automated detection algorithms. Tools such as *EyeDoctor* or *EyeTrace* attempt to minimize subjectivity by offering visualization aids and coder agreement metrics, but they cannot fully eliminate the human bottleneck [41].

In this chapter, we used the dataset annotated manually by two human coders, MN and RA discussed in sub Section 3.2.1 and we evaluated to what extent the two coders agreed to classify the same input data into events. We used the eye tracking data recorded during image viewing with the 4988 samples (`UH21_img_Rome_labelled`). Coder MN classified 4282 samples as fixations, 503 as saccades and 203 as PSOs. Coder RA labeled 4173 as fixations, 466 as saccades, 164 as PSOs, 177 as smooth pursuits and eight as undefined. The value of Cohen's kappa was 90% and the confusion matrix between both coders is presented in Table 4.1. It shows that the classifications of the two agreed moderately. The most significant differences could be found in the PSO events with an F1-score as low as 85%. It seems that it is only a minor difference when only some samples between the end of the saccade and the onset of fixation are classified differently. However, such misclassification influences important parameters of eye movement data, like average fixation duration or average saccade length. Such parameters are frequently used in eye movement data analysis [52, 77, 107].

Table 4.1: Confusion matrix between two manual coders.

RA\MN	**Fixation**	**Saccade**	**PSO**
fixation	4111	4	54
saccade	28	444	10
PSO	26	14	297

4.3 Threshold Based Event Detection

4.3.1 Dispersion Threshold-Based Event Detection Methods

Threshold-based methods are historically the first automated eye movement event classification algorithms and are still frequently used nowadays. The I-DT is the most straightforward and obvious eye movement event detection algorithm that classifies fixation points and saccade points based on the dispersion or spread distance of subsequent sample coordinates. The algorithm identifies gaze data as belonging to fixation when the samples are located within a spatially limited area (for example, 0.5°) for minimum allowed fixation duration [97]. It follows that fixation points generally occur near one another. Saccades are then detected implicitly as everything else [91].

The algorithm requires two parameters to identify the events. These are dispersion threshold and duration threshold.The dispersion threshold can be set to 0.5 to 1° of visual angle if the distance from the eye to the screen is known. Otherwise, the dispersion threshold can be estimated from the exploratory analysis of data. The duration threshold is typically set to a value between 100 and 200 ms depending on task processing demands [104]. The algorithm calculates the dispersion of points in a window by simply summing the differences between the points' maximum and minimum X and Y values, as shown in Equation (4.1).

$$D = [max(x) - min(x)] + [max(y) - min(y)] \tag{4.1}$$

However, there are other methods of dispersion estimation methods discussed in [94]. The first method is distance dispersion, an algorithm that classifies every point as fixation if the distance between every point is no further than some threshold D_{max}. It is the most intuitive but less popular measure. Another method is the centroid-distance method, which requires the M of N points to be no further than some threshold C_{max} from the centroid of N points. This algorithm has two versions, a consistent version that recomputes the distance of all points in the fixation to the centroid whenever the fixation is considered and a simpler (and faster) version that only checks the distance of the new point to be added.

The dispersion threshold methods exhibit poor performance detecting fixations and saccades when the signal is noisy [31]. Therefore, choosing the optimum threshold values is the most challenging step in the I-DT event detection algorithms. The impact of varying dispersion threshold values on the classification performance leads to biased results and misclassifications. For example, if the threshold value is set too high, false fixations might be identified and if it is set too low, actual fixations might be missed [10]. Due to this, parameter setting in the I-DT algorithms is crucial and may cause substantial differences in classification performance [10].

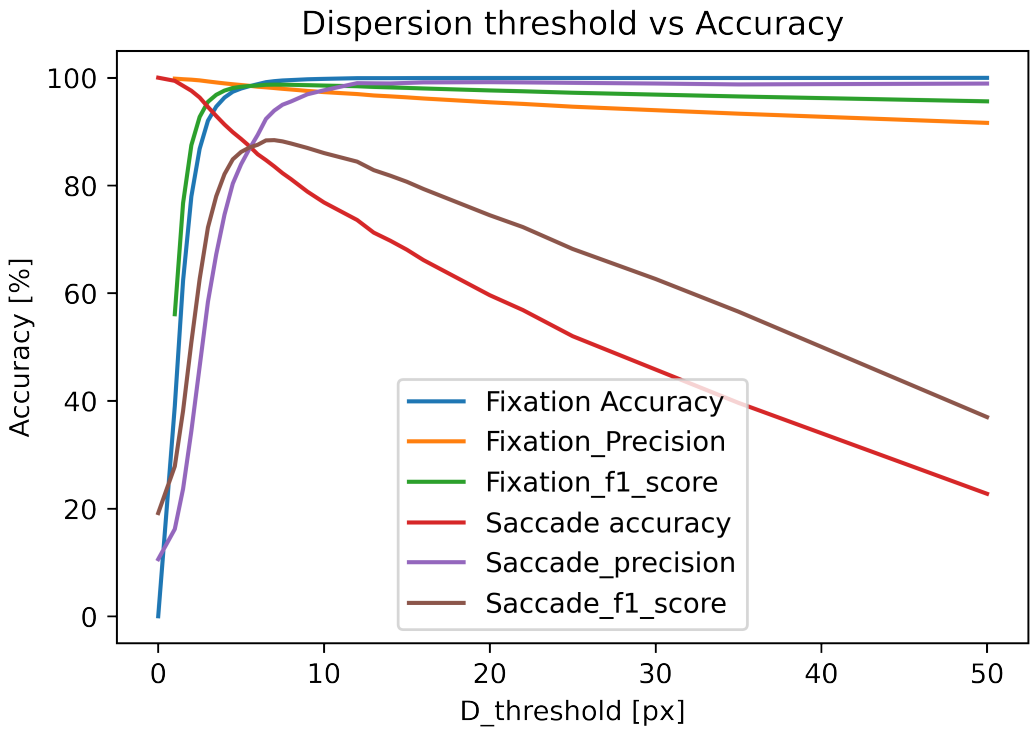


Figure 4.1: The accuracy for fixations and saccades of the I-DT algorithm for different dispersion thresholds.

In this section, we evaluated the I-DT algorithm and the impact of threshold value on the classification performance was examined in a simple experiment. We used the dispersion threshold as a parameter. All input samples were converted into sequences containing the point and four points surrounding the classified point. The algorithm calculated dispersion for each sequence of points using Equation 4.1. We used the data collected from participants viewing images (see Section 7.2.1) and compared the results of the I-DT algorithm with different threshold values with the manual classification.

The results are presented in Figure 4.1, which illustrates the impact of varying dispersion threshold value on the classification performance in the I-DT algorithm. The accuracy for each class is measured by recall, precision and F1-score from a confusion matrix. As shown from the results, the increase of the dispersion threshold value increases the fixation recall but, at the same time, decreases the saccade recall. On the other hand, increasing the threshold decreases fixation precision and increases saccade precision. The F1-score may be considered a good indicator of the correct threshold as it reaches the maximum value for both fixations and saccades for a similar threshold value.

For example, I-DT gives a maximum fixation recall of 99% and a minimum saccade recall of 82% at a dispersion threshold value of 7 px and a maximum saccade accuracy of 99% and a minimum fixation accuracy of 39% at the threshold value of 1 px. The optimum dispersion threshold value for the given example is 3.5 px. At this threshold value the I-DT gives 95% fixation recall value, 93% saccade recall, 98% fixation precision, 51% saccade

precision, 96% fixation F1-score, saccade F1-score 66% and 0.6 Cohen's kappa.

4.3.2 Velocity Threshold-Based Methods

The velocity threshold algorithm is another algorithm and the foundation for an automated/objective standard event detection algorithm. Many studies have adopted this approach [93, 24]. It utilizes the fact that saccadic eye movements are characterized by higher velocity values than fixational movements. The velocity profiles of eye movements show essentially two velocity distributions: low velocities for fixations and high velocities for saccades. The I-VT method identifies events by calculating the point-to-point velocity and then classifies the event as fixation or saccade based on the value of this velocity [91]. The classic I-VT method is designed to classify all eye-tracking input data into fixations and saccades only. The other event types, such as smooth pursuits, post-saccadic oscillations and noises, are not considered.

Figure 4.2 presents the impact of varying velocity thresholds on the classification performance of the I-VT algorithm. The classification accuracy of each class is measured by the recall, precision and F1-score calculated from the confusion matrix. Similarly to the I-DT algorithm, the increase in the velocity threshold increases the fixation classification recall and saccade precision while, at the same time, it decreases fixation precision and saccade recall.

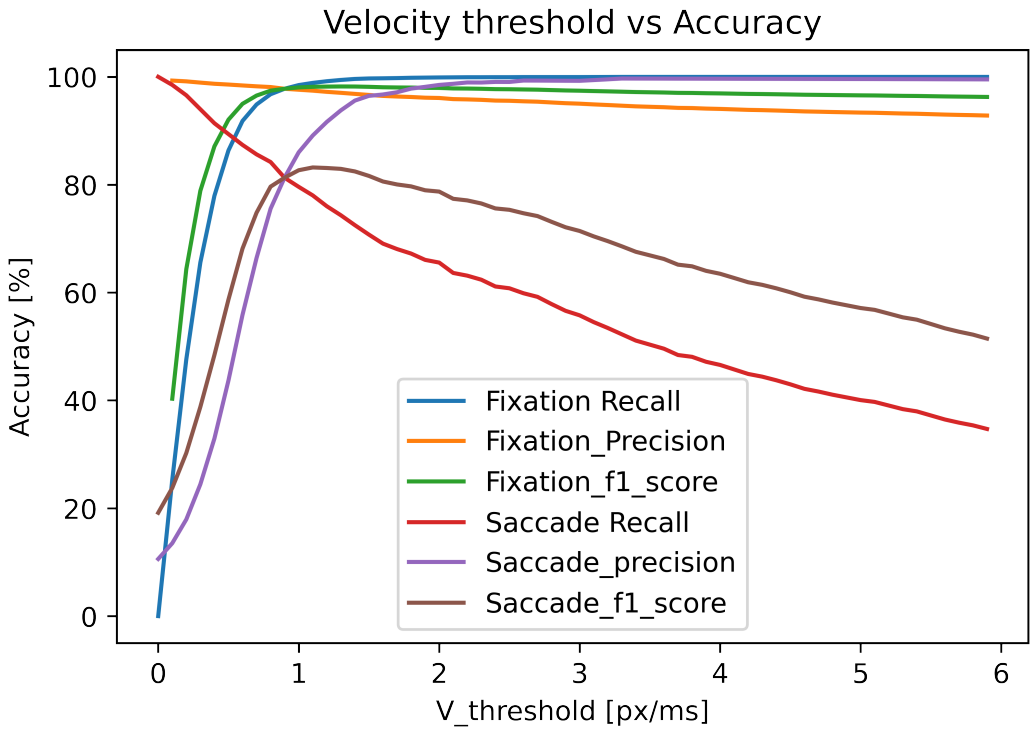


Figure 4.2: The accuracy for fixations and saccades of the I-VT algorithm for different velocity thresholds.

In the given example, I-VT yields a maximum of 99% fixation recall at a threshold velocity of 3.5 px/ms and the saccade recall slightly decreases with the increase in velocity threshold value. The saccade recall reaches 98% and the fixation recall is 25% at the lowest velocity threshold value of 0.1 px/ms because, at this threshold value, most points are classified as saccades. Due to the impact of the threshold value on the classification accuracy of the I-VT algorithm, it is essential to find the optimum threshold value for both fixation and saccade accuracy. Therefore, in this case, the optimum velocity threshold value for I-VT is 0.5 px/ms. At this point, the fixation recall value is 92%, the saccade recall is 87%, the fixation precision is 96%, the saccade precision is 46%, the fixation F1-score is 94% and the saccade F1-score is 60%. The value of Cohen's kappa at the optimum threshold value of 0.6 px/ms is only 0.5, which shows a moderate agreement between the human coders and I-VT classification algorithm.

The main drawback of the algorithm is that it uses only the velocity of the gaze without considering other possibilities like acceleration of the signal, direction of the gaze movement, the distance between the eye and camera, etc. It may result in misclassifications of events because the velocity ranges of the quickest slow eye movements and the slowest parts of saccades may overlap. Therefore, it seems that using other eye movement parameters such as acceleration, amplitude and position of eye movement could improve the results.

There is no standard optimum threshold velocity value and varying the threshold values affects the performance of the event detection algorithms. Due to these reasons, different researchers use different threshold values to develop and evaluate the performance of I-VT algorithms. Due to this variation, it is difficult to compare different studies of threshold-based event detection algorithms [91].

4.4 Machine Learning and Deep Learning Based Event Detection Methods

A major limitation of threshold-based event detection algorithms is the reliance on user-defined parameters that must be fine-tuned depending on the quality of the eye-tracking data. Finding optimal threshold values is often a challenging and subjective task. Moreover, threshold-based algorithms are typically designed for one-step binary classifications such as distinguishing between fixations and saccades and do not extend naturally to the multi-class classification required for detecting more complex eye movement events, such as post saccadic oscillations (PSOs).

Machine learning and deep learning methods address these limitations by learning classification rules directly from labeled data, eliminating the need for manually defined thresholds [112, 98]. These algorithms can automatically classify raw eye-tracking data

into multiple event types such as fixations, saccades, and PSOs based on features extracted from the gaze signal.

Most machine learning algorithms assume that the classification of a specific gaze point depends on its temporal context. Therefore, models typically receive as input a window of gaze samples, containing information from both before and after the point of interest. The input features can include raw coordinates, velocity, acceleration, movement direction, or jerk. The window size is one of the basic parameters for every model.

In this section, we evaluated machine learning and deep learning models, including Random Forest (RF), Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) networks. To assess the generalizability and robustness of these models, we applied a Leave-One-File-Out Cross-Validation (LOFO-CV) strategy. The use of cross-validation ensures that the models are not overfitting to specific data segments and can generalize well to unseen data.

4.4.1 Cross-Validation Strategy

To evaluate the generalization ability and robustness of the proposed models, we employed a Leave-One-File-Out Cross-Validation (LOFO-CV) strategy. This approach is particularly well-suited for time-series data and ensures that the model is tested on completely unseen data while being trained on the rest, thereby simulating real-world variability across sessions or subjects.

For this study, we used six separate eye-tracking files collected during the image viewing task. Each file was treated as an independent data partition, representing one fold in the cross-validation process. In each iteration, five files were used for training, and the remaining one was held out for testing. This process was repeated six times, with a different file used for testing in each fold. The architecture of the cross validation is shown in the Figure 4.3.

This method provides a comprehensive evaluation of model performance across different data distributions and reduces the likelihood of overfitting to any specific file or subject. It also allows us to measure the stability and generalizability of the models across varying input conditions.

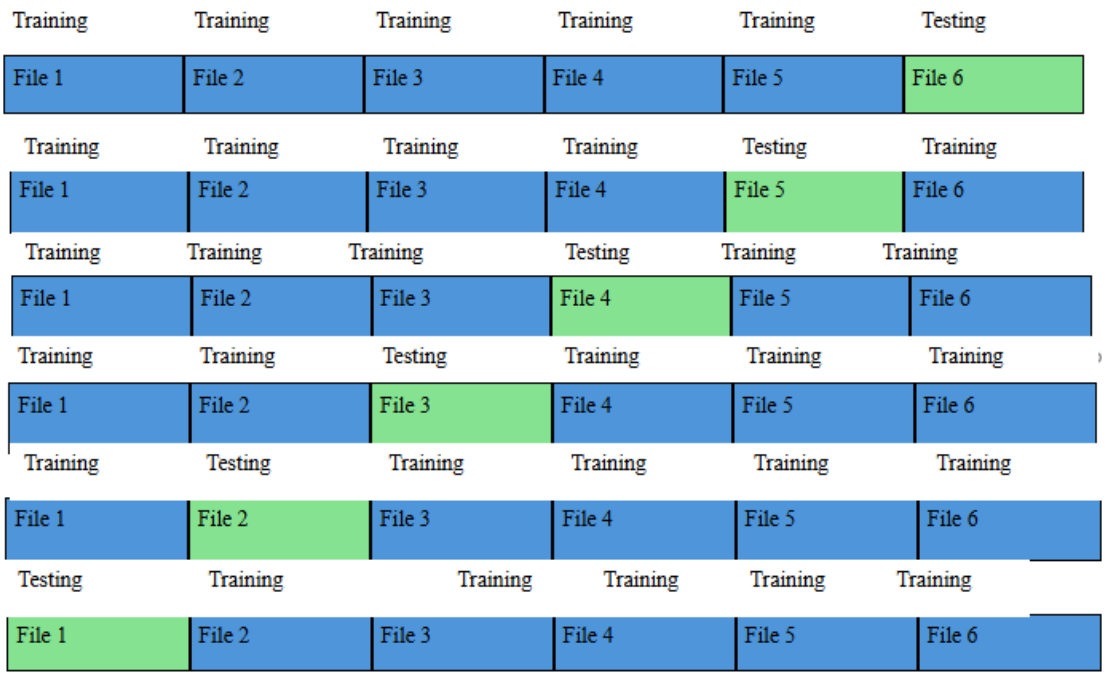


Figure 4.3: Leave One File Out Cross Validation (LOFO-CV) Architecture

4.4.2 Event Classification Using Random Forest Classifier

Fully automated eye movement event classification using a Random Forest classifier was first proposed in [112] to classify fixations, saccades and post-saccadic oscillations. Classification performance was compared with the current state-of-the-art algorithms and manual human coders. The paper stated that the machine learning algorithm outperforms the current state-of-the-art algorithms and almost reaches the performance of manual human experts. However, this performance was only achieved for high-quality data (with low noise levels). In this section, we describe our own implementation of the algorithm that utilizes the Random Forest classification model for event classification.

We implemented the Random Forest classification algorithm to classify eye-tracking data into fixations, saccades and PSOs. We evaluated the classification performance regarding fixation classification accuracy, saccade accuracy and PSO classification accuracy. This algorithm can detect eye movements in the continuous gaze stream and assign labels for all three eye movement types simultaneously. We, therefore, further evaluated the algorithm's classification performance separately for the three-class detection problem by evaluating sample-by-sample predictions, confusion matrices and finally, by evaluating the classification performance of each class.

To build the model, velocity was used as the primary feature. Eye tracker coordinate data were transformed into the velocity domain, and sequences of samples were constructed with a sequence length of 50. Although both shorter and longer sequences were tested, they did not result in a significant difference in performance. Consequently, the input to

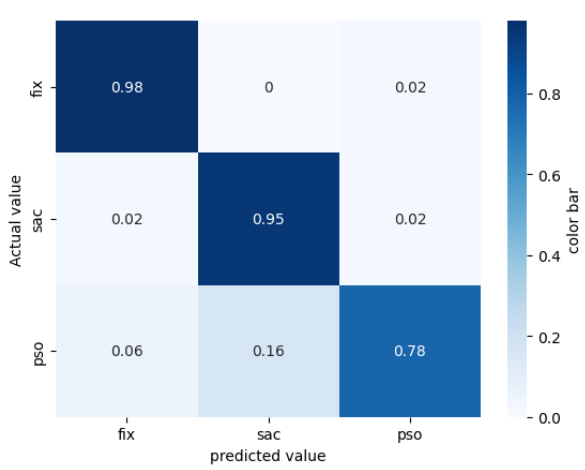
the model consisted of gaze sample sequences with a shape of 50×2.

Additionally, the model was evaluated using a leave-one-file-out cross-validation approach. The performance of the model with and without cross-validation was compared to assess its generalizability.

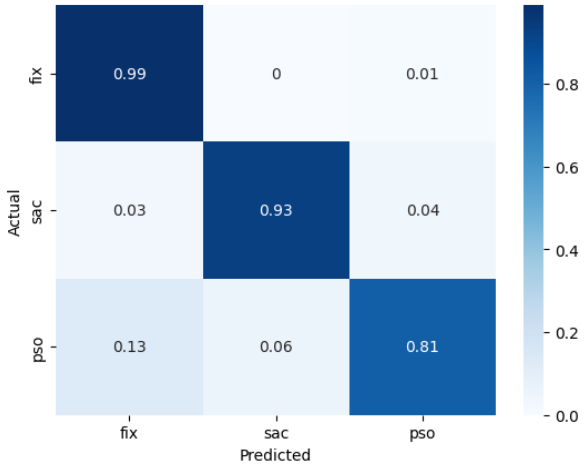
Figure 4.4 presents the confusion matrices of the Random Forest (RF) algorithm for sample-by-sample evaluation, both with and without cross-validation. In the confusion matrix without cross-validation (Figure 4.4(b)), fixations are correctly classified in 98% of cases. However, post-saccadic oscillations (PSOs) exhibit a tendency to be misclassified as saccades (16%), while only 6% of PSOs are misclassified as fixations. Saccades and PSOs are correctly identified in 95% and 78% of frames, respectively. Overall, 7% of PSOs are incorrectly classified as fixations and 16% as saccades.

In the confusion matrix with cross-validation (Figure 4.4(a)), PSOs are more frequently misclassified as fixations (13%), while only 6% are misclassified as saccades. This misclassification pattern is likely due to the class imbalance in the training data, where fixations dominate, causing the model to favor classification of ambiguous events as fixations.

Overall, the results indicate that the RF model performs similarly for fixation and saccade classification with and without cross-validation. However, the use of cross-validation improves PSO detection slightly. Nevertheless, the performance of PSO detection remains relatively low in both scenarios.



The RF confusion matrix without cross validation (a)



The RF confusion matrix with cross validation (b)

Figure 4.4: The confusion matrix of RF with and without cross validation

Tables 4.2 and 4.3 summarize the performance of the RF classifier without and with cross-validation, respectively, in terms of accuracy, precision, recall, and F1 score for each class. The results indicate that the classification of PSO events is the most challenging. Overall, the RF model with cross-validation outperforms the model without cross-validation for saccades and PSOs across all metrics, except for saccade recall, which is

slightly higher without cross-validation. For fixation detection, both approaches yield comparable performance across all evaluation metrics.

Table 4.2: Each class classification performance with RF classifier without cross validation.

Classes	Accuracy	Precision	Recall	F1-Score
Fixation	98%	99%	98%	99%
Saccade	95%	88%	95%	91%
PSO	78%	73%	78%	76%

Table 4.3: Each class classification performance with RF classifier with cross validation.

Classes	Accuracy	Precision	Recall	F1-Score
Fixation	99%	98%	99%	98%
Saccade	93%	92%	93%	92%
PSO	81%	81%	81%	80%

4.4.3 Using Convolutional Neural Networks

Convolutional Neural Networks are good at finding patterns in data, so it is possible to use them in eye movement event detection. One example of such an application is the method proposed in [44], which is based on the deep Convolutional Neural Network that, for each sample, predicts a sequence of probabilities of belonging to a fixation, saccade, or smooth pursuit from a sequence of gaze samples. The method tries to address the drawback of previous methods, which use signal shape and amplitude to determine or to classify the eye movement events, which may be problematic, for instance, for smooth pursuits. The proposed method uses the signal's frequency to classify the data into event types. That means it first converts the raw gaze data into the frequency domain of the raw signal using Fast Fourier Transform (FFT) and then passes the frequency representation of the signal to the CNN network, which in turn gives the output of a three-dimensional activation signal. Each signal represents the probability of each eye movement type (fixation, saccade and SP). Finally, the label with a high probability is assigned to the central sample in the window.

The method is not end-to-end, as the input to the network is the FFT output. It uses hand-crafted features—input data that need to be transformed into the frequency domain. The proposed method classifies fixations, saccades and smooth pursuits without considering other events like PSO. The method outperforms the old algorithms based on simple dispersion and velocity thresholding.

To evaluate the performance of a convolutional neural network (CNN) in classifying eye movement events, we designed a simple network architecture, as illustrated in Figure 4.5. The network processes a continuous stream of two-dimensional gaze samples as input. To generate a prediction for each gaze sample, a sliding window approach is employed, where the window moves over the sequence one sample at a time. The raw gaze coordinate points (x and y) are first converted into horizontal and vertical velocity components by computing the sample-to-sample velocity. To effectively capture relevant eye movement features, the data stream is segmented into overlapping windows of 50 samples, which yielded the best performance in our experiments. The model was further evaluated using leave-one-file-out (LOFO) cross-validation to assess its generalization capability.

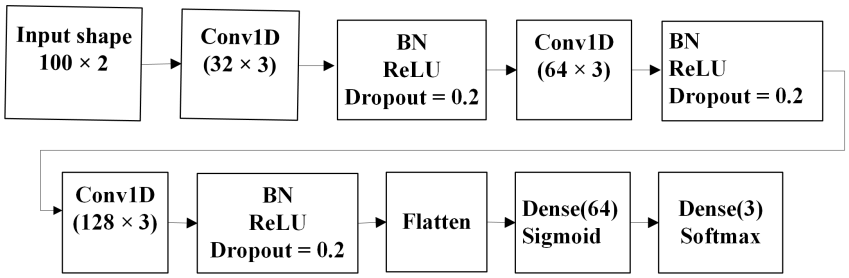


Figure 4.5: The architecture of the CNN used in the experiment.

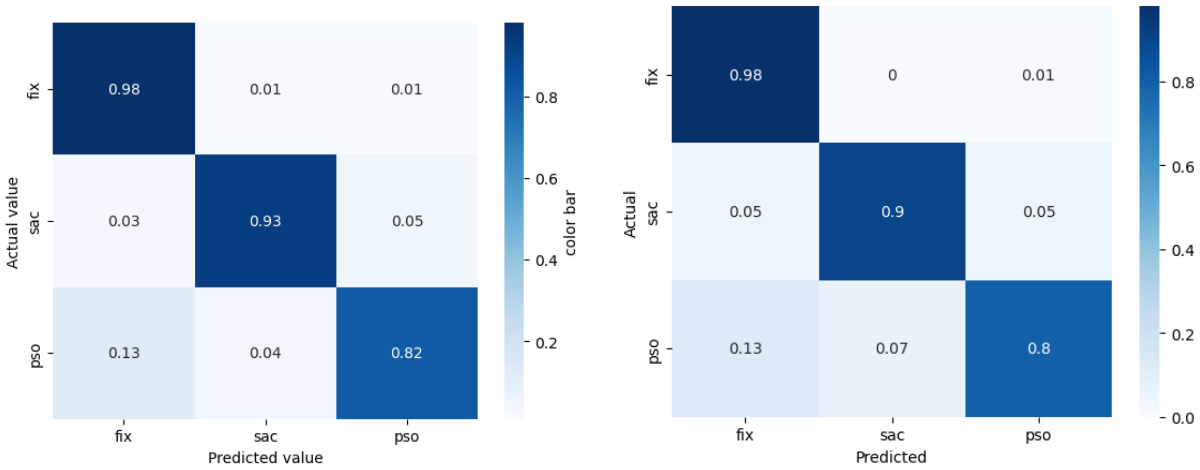
The network is composed of different layers, precisely three convolutional layers with a gradually increasing number of filters (16, 32 and 64) with a kernel size of 3, a batch normalization operation before activation and an output layer. Input to the network is a sequence of gaze samples of shape 50 × 2. The network architecture is shown in Figure 4.5.

Figure 4.6 presents the confusion matrices of the convolutional neural network (CNN) for sample-by-sample evaluation, both with and without cross-validation. In the confusion matrix without cross-validation (Figure 4.6(a)), fixations, saccades, and post-saccadic oscillations (PSOs) are correctly classified in 98%, 93%, and 82% of cases, respectively. With cross-validation (Figure 4.6(b)), 98% of fixations , 90% of saccades, and 80% of PSOs are classified correctly.

However, PSOs tend to be misclassified as saccades in both scenarios. Specifically, 13% of PSOs are misclassified as saccades without cross-validation, while this rate drops slightly to 13% with cross-validation.

Overall, the CNN model demonstrates comparable performance for all event types with and without cross-validation, with a slight improvement observed in the cross-validation condition. Nevertheless, as with the Random Forest model, the detection of PSOs remains

the most challenging, showing relatively lower performance in both cases.



The CNN confusion matrix without cross validation (a)

The CNN confusion matrix with cross validation (b)

Figure 4.6: The confusion matrix of CNN model without and with cross validation

Tables 4.4 and 4.5 summarize the performance of the CNN classifier without and with cross-validation, respectively, in terms of accuracy, precision, recall, and F1 score for each class. The results indicate that the CNN model performs similarly in both settings, with no significant differences observed. However, both precision and recall are slightly higher when cross-validation is applied, suggesting a modest improvement in classification consistency and generalization.

Table 4.4: Each class classification performance with CNN classifier without cross validation.

Classes	**Accuracy**	**Precison**	**Recall**	**F1-Score**
Fixation	98%	99%	98%	99%
Saccade	93%	92%	93%	92%
PSO	82%	79%	82%	81%

Table 4.5: Each class classification performance with CNN classifier with cross validation.

Classes	**Accuracy**	**Precison**	**Recall**	**F1-Score**
Fixation	99%	98%	99%	99%
Saccade	89%	92%	89%	90%
PSO	83%	77%	84%	79%

4.4.4 Using Recurrent Neural Networks

Eye movement recordings form a time series, so it is natural that algorithms proven to operate well on time series could be used for event classification. One of the possibilities is to use Recurrent Neural Networks.

The paper [98] presents an excellent example of such an application. It presents the network that classifies the raw eye movement data into fixations, saccades and smooth pursuits. The network is a combination of the 1D-convolutional network and the BLSTM layer (a classic recurrent layer that preserves information about previous samples). It is built of a one-dimensional temporal convolutional network with one time-distributed dense layer both before and after the BLSTM. Individual feature sets for the model are raw positional coordinates, speed, direction and acceleration. However, the method exhibits poor performance when it takes a combination of parameters. Researchers used a publicly available manually annotated eye-tracking dataset with over four hours of 250 Hz low-frequency recordings done with SR Research EyeLink II and 500 Hz recordings done with SensoMotoric Instruments Hi-Speed 1250 eye tracker. The combination of direction and speed showed a noticeable improvement over using them separately. Acceleration as an additional feature did not improve average detection performance, probably due to its inability to distinguish smooth pursuits from fixations.

Long Short-Term Memory (LSTM) Networks

In this section, we implemented a Long Short-Term Memory (LSTM) network for the classification of fixations, saccades, and post-saccadic oscillations (PSOs). The model was evaluated both with and without cross-validation to assess its generalization performance.

The architecture consists of three stacked LSTM layers with an increasing number of units: 16, 32, and 64, respectively. These are followed by two fully connected (dense) layers: the first with 64 units and a sigmoid activation function, and the second (output layer) with a number of units equal to the number of classes, using a softmax activation function. Batch normalization is applied before the activation function in each layer. The input to the network is a sequence of gaze samples with a shape of 50×2.

The model was trained using the categorical cross-entropy loss function, with the Adam optimizer and accuracy as the evaluation metric. Training was conducted with a batch size of 50 over 20 epochs. The network architecture is illustrated in Figure 4.7.

Figure 4.8(a) and Figure 4.8(b) present the confusion matrices of the LSTM algorithm for sample-by-sample evaluation, without and with cross-validation, respectively. The results demonstrate that the model performs well for fixation and saccade classification in both settings, with 98% of fixations and 91% of saccades correctly identified.

However, the detection of post-saccadic oscillations (PSOs) remains more challenging. Interestingly, the model without cross-validation shows slightly better performance

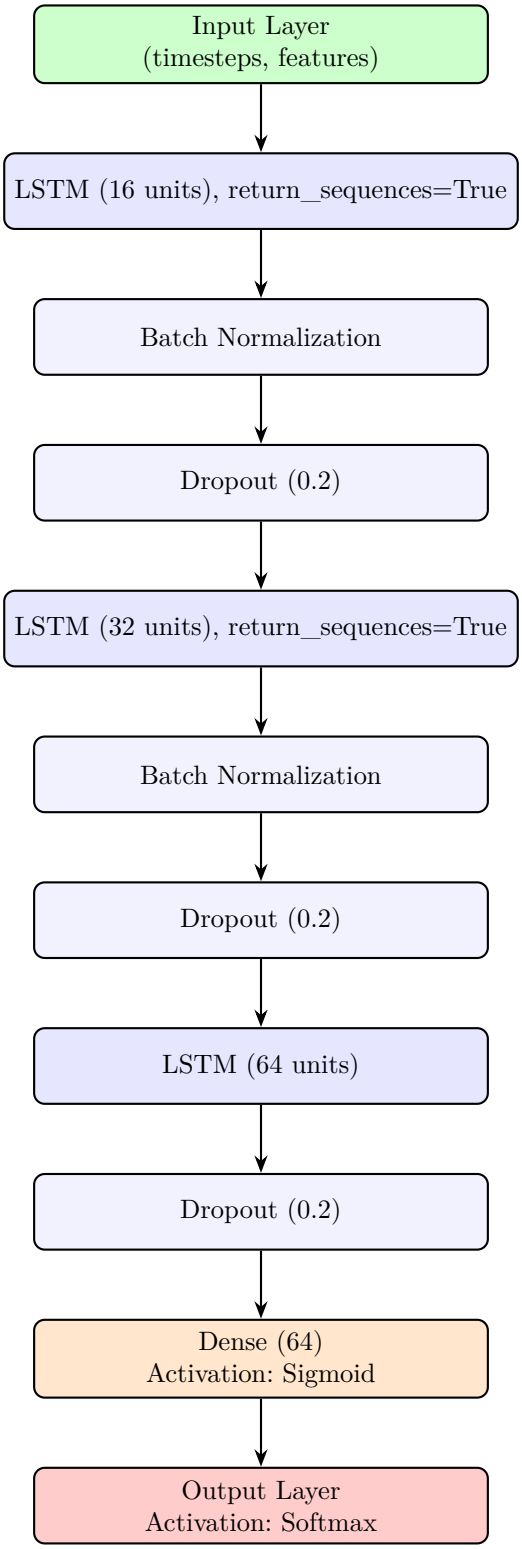
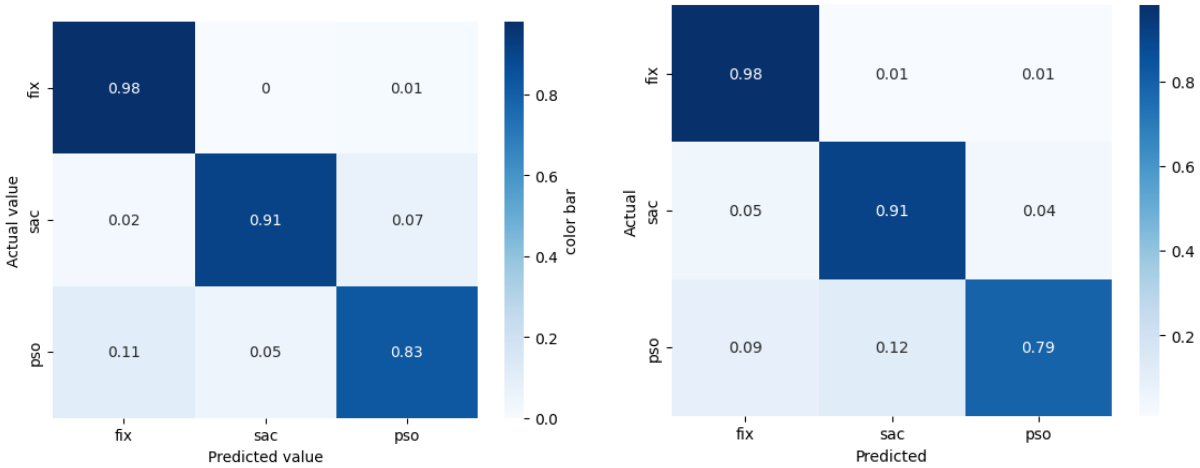


Figure 4.7: The LSTM network architecture

for PSO classification. Specifically, 83% of PSOs are correctly classified without cross-validation, while 11% and 5% are misclassified as fixations and saccades, respectively. With cross-validation, 79% of PSOs are correctly classified, with 9% and 12% misclassified as fixations and saccades, respectively. In both cases, PSOs exhibit a tendency to be misclassified as saccades.

Overall, the LSTM model demonstrates consistent performance for fixation and saccade classification, regardless of the use of cross-validation. While cross-validation generally improves model robustness, PSO detection does not significantly benefit from it in this case. Similar to the CNN and RF models, the classification performance for PSOs remains relatively lower compared to the other classes.



The LSTM confusion matrix without cross validation (a)

The LSTM confusion matrix with cross validation (b)

Figure 4.8: The confusion matrix of LSTM with and without cross validation

Tables 4.6 and 4.7 summarize the performance of the LSTM classifier without and with cross-validation, respectively, in terms of accuracy, precision, recall, and F1 score for each class. The results indicate that the LSTM model without cross-validation outperforms the model with cross-validation for PSO detection across all evaluation metrics. For saccade classification, recall remains similar between the two settings. In the case of fixation detection, both models achieve comparable performance across all metrics.

Table 4.6: Each class classification performance with the LSTM classifier without cross validation.

Classes	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Fixation	98%	99%	98%	99%
Saccade	91%	93%	91%	92%
PSO	83%	74%	83%	78%

Table 4.7: Each class classification performance with the LSTM classifier with cross validation.

Classes	Accuracy	Precision	Recall	F1-Score
Fixation	98%	98%	98%	98%
Saccade	91%	87%	91%	89%
PSO	76%	75%	79%	75%

4.5 Model Performance Across Cross-Validation Folds

To assess the generalizability and robustness of the implemented models, we analyzed their behavior across the seven folds of the Leave-One-File-Out Cross-Validation (LOFO-CV) scheme. This evaluation provides insight into how each model performs on different subsets of the data, simulating the challenge of applying the model to unseen files.

Each of the three models Random Forest (RF), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM) was evaluated using four key metrics: precision, recall, F1-score, and Cohen's Kappa, calculated per fold and per class (Fixation, Saccade, and PSO).

Figure 4.9 illustrates the behavior of the models per-fold across all metrics. The results show that Random Forest consistently performs slightly better overall, particularly standing out in folds 5 and 6, where it achieves the highest values across nearly all metrics: precision, recall, F1 score, and Cohen's kappa. Its performance is both strong and dependable, showing minimal fluctuation between folds. This consistency and balance make Random Forest the best performing model in this comparison.

CNN is very stable across all folds, maintaining high scores across all metrics with little variation. Although it does not always outperform Random Forest, it often comes very close and occasionally leads, for example, in fold 7 for precision. In particular, CNN highly outperforms all other models in fold 1 for recall, achieving a remarkable 0.91, which indicates its strong ability to capture PSO events. Its overall consistency and ability to perform well in PSO detection make it a highly reliable choice.

LSTM has slightly more variations, but still performs quite well, particularly in recall, where it matches or slightly exceeds the other models in some folds. However, its lower precision in some folds affects its overall F1 scores and Cohen's kappa slightly.

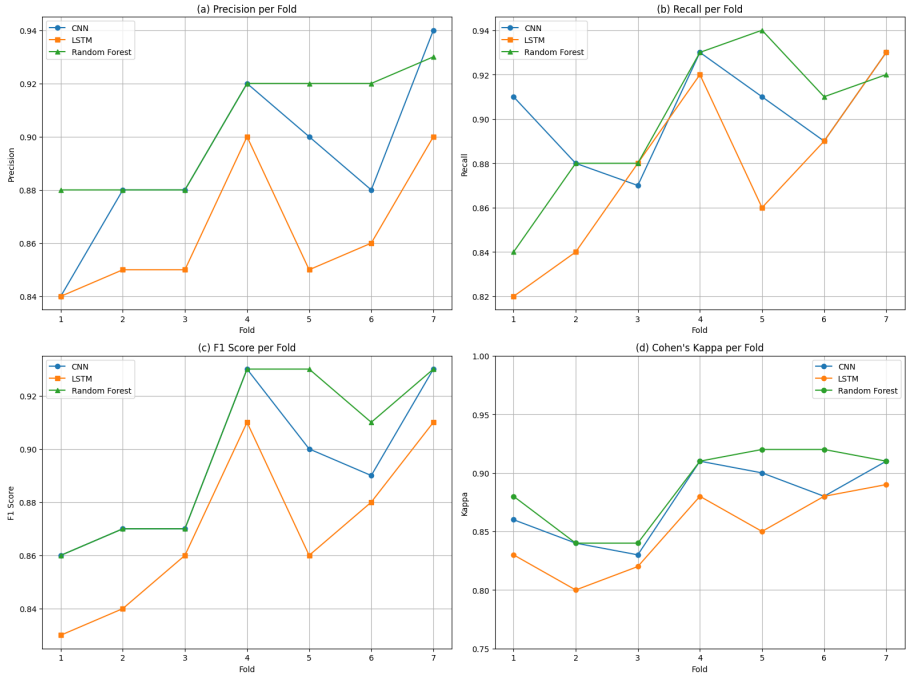


Figure 4.9: The performances of the models per fold across all metrics

The results of the models evaluated across individual folds are summarized in Table 4.8 and Table 4.9, which report the F1-score and precision values, respectively, for each event class (Fixation, Saccade, and PSO). Recall and Cohen's Kappa values across all models and folds are presented in Table 4.10 and Table 4.11. Together, these tables provide a comprehensive overview of model performance across different folds and event types.

4.5.1 Comparative Results and Discussion

Although CNN performed consistently well for fixation and PSO detection, it showed variability in saccade classification across folds. This fluctuation may stem from the model's sensitivity to initial weight settings or from differences in participant behavior across files.

The LSTM model showed promising results, particularly for saccade recall in certain folds. However, its inconsistent performance for PSOs suggests that although LSTM captures temporal dynamics effectively, it may be more sensitive to data distribution shifts between folds.

Among the models, RF demonstrated the most stable and balanced performance across all folds. It achieved higher Cohen's Kappa scores and more consistent recall, especially for saccades. This indicates that RF may be more resilient to class imbalance and inter-subject variability in the dataset.

Table 4.8: F1-Score per Fold and Event

Fold	Event	CNN	LSTM	RF
1	Fixations	0.81	0.82	0.85
	Saccades	0.68	0.65	0.71
	PSO	0.70	0.73	0.72
2	Fixations	0.77	0.84	0.83
	Saccades	0.64	0.62	0.66
	PSO	0.68	0.75	0.70
3	Fixations	0.85	0.87	0.86
	Saccades	0.70	0.67	0.70
	PSO	0.74	0.76	0.75
4	Fixations	0.82	0.83	0.84
	Saccades	0.66	0.67	0.67
	PSO	0.71	0.74	0.73
5	Fixations	0.84	0.89	0.86
	Saccades	0.68	0.68	0.70
	PSO	0.73	0.75	0.74
6	Fixations	0.83	0.85	0.87
	Saccades	0.70	0.66	0.72
	PSO	0.72	0.74	0.75
7	Fixations	0.82	0.85	0.85
	Saccades	0.72	0.65	0.71
	PSO	0.73	0.75	0.74

Table 4.9: Precision per Fold and Event

Fold	Event	CNN	LSTM	RF
1	Fixations	0.80	0.81	0.83
	Saccades	0.69	0.65	0.70
	PSO	0.68	0.72	0.71
2	Fixations	0.76	0.83	0.82
	Saccades	0.63	0.61	0.65
	PSO	0.66	0.74	0.68
3	Fixations	0.84	0.86	0.85
	Saccades	0.69	0.66	0.69
	PSO	0.72	0.75	0.73
4	Fixations	0.81	0.82	0.83
	Saccades	0.65	0.66	0.66
	PSO	0.69	0.73	0.72
5	Fixations	0.83	0.88	0.85
	Saccades	0.67	0.67	0.69
	PSO	0.71	0.74	0.73
6	Fixations	0.82	0.84	0.86
	Saccades	0.69	0.65	0.71
	PSO	0.70	0.73	0.74
7	Fixations	0.81	0.84	0.84
	Saccades	0.71	0.64	0.70
	PSO	0.71	0.74	0.73

Table 4.10: Recall per Fold, Event, and Model

Fold	Event	CNN	LSTM	RF
1	Fixations	0.83	0.85	0.89
	Saccades	0.70	0.66	0.73
	PSO	0.73	0.76	0.75
2	Fixations	0.79	0.86	0.87
	Saccades	0.65	0.64	0.65
	PSO	0.71	0.78	0.73
3	Fixations	0.88	0.89	0.89
	Saccades	0.71	0.69	0.71
	PSO	0.76	0.79	0.77
4	Fixations	0.84	0.86	0.88
	Saccades	0.67	0.69	0.68
	PSO	0.73	0.77	0.75
5	Fixations	0.87	0.91	0.89
	Saccades	0.69	0.70	0.72
	PSO	0.75	0.78	0.77
6	Fixations	0.86	0.88	0.90
	Saccades	0.71	0.68	0.73
	PSO	0.74	0.77	0.78
7	Fixations	0.85	0.88	0.89
	Saccades	0.73	0.67	0.72
	PSO	0.75	0.78	0.76

Table 4.11: Cohen's Kappa Score per Fold and Model

Fold	CNN	LSTM	RF
1	0.72	0.70	0.78
2	0.68	0.71	0.76
3	0.75	0.76	0.79
4	0.73	0.74	0.77
5	0.76	0.78	0.80
6	0.74	0.73	0.79
7	0.75	0.74	0.78

4.5.2 Summary of Fold-Wise Evaluation

Among the three event types, post-saccadic oscillations (PSOs) proved to be the most challenging to classify accurately across all models. Therefore, PSO performance serves as a more informative and discriminative indicator for evaluating and comparing the true capabilities of the models.

In this context, the CNN model demonstrated superior and more stable performance in PSO classification across the folds, making it a strong candidate for detecting complex and ambiguous events. The LSTM model, while effective in capturing temporal patterns, exhibited greater variability across folds in PSO classification, indicating sensitivity to training data distribution. The Random Forest (RF) model showed the most consistent performance overall, particularly for saccades and fixations, but was slightly less effective than CNN in handling the intricacies of PSO events.

These findings reinforce the critical role of cross-validation not only in assessing general model robustness but also in highlighting differences in how models respond to the most difficult classification tasks. Because PSOs are the hardest to detect, how well a model handles them is a good way to judge and improve future models for eye movement event detection.

4.6 Comparative Evaluation

The purpose of this chapter was to compare different eye movement event detection algorithms. This was done by evaluating the performance of five different event classification algorithms drawn from threshold-based, machine learning-based, and deep learning-based approaches, as well as assessing the mutual agreement between two human evaluators. Cross-validation was also applied, and the performance of each model was evaluated per fold to ensure robustness and consistency across different subsets of the data. This fold-wise analysis provided deeper insights into the strengths and stability of each algorithm.

Table 4.12 and 4.13 show the comparative discussion of all the algorithms for f1 score and precission metrics for each event class. The rows in the tables show the algorithms and the columns show the evaluation metrics for each class.

4.6.1 Discussion of Results

The results demonstrate clear differences in performance across the range of evaluated models. particularly in their ability to classify fixations, saccades, and post-saccadic oscillations (PSOs). Manual classification unsurprisingly remains the highest standard for accuracy, particularly for fixation and saccade detection. However, its subjective and time-consuming nature makes automation desirable.

Threshold-based methods such as I-DT and I-VT achieve acceptable performance for fixation and saccade classification but are limited by their dependence on static thresholds. These approaches show poor generalizability and perform particularly poorly in post-saccadic oscillation (PSO) detection due to their inability to adapt to the complex temporal dynamics of eye movements.

In contrast, machine learning-based models, especially Random Forest (RF), Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) networks, outperform threshold-based methods in nearly every evaluation metric. Among these, the CNN model consistently demonstrates superior performance, particularly in terms of F1-score, precision, and confusion matrix analysis.

For fixation detection, all models (RF, CNN, LSTM) perform comparably well, with F1-scores of 99% and nearly perfect precision/recall in most configurations. However, for saccade and PSO detection, CNN clearly stands out. While RF-CV slightly edges out CNN-CV in Cohen's Kappa score (0.887 vs. 0.875), CNN-CV delivers more balanced performance across all three classes, especially in minimizing PSO misclassifications. Notably, CNN-CV achieves F1-scores of 99% (fixation), 90% (saccade), and 79% (PSO) compared to RF-CV's 98%, 92%, and 80%, respectively.

The confusion matrices support these metrics: CNN-CV correctly classifies 98% of fixations, 90% of saccades, and 84% of PSOs, while RF-CV reaches 99%, 93%, and 81%, respectively. However, CNN reduces the PSO-to-saccade misclassification rate slightly more than RF, suggesting better class separation for ambiguous eye events. LSTM, while leveraging the temporal context inherent in gaze sequences, performs slightly lower overall. It achieves good results for fixations and saccades but consistently underperforms in PSO classification compared to CNN and RF, especially when cross-validation is applied.

Overall, CNN with cross-validation (CNN-CV) emerges as the most effective model for general eye movement event detection. It maintains high performance across all classes, handles the complexity of PSOs well, and benefits from robust generalization with cross-validation. RF-CV remains a strong contender, particularly due to its lower computational demands and strong precision on saccades. LSTM may still be valuable for tasks heavily reliant on temporal dependencies but requires further optimization for reliable PSO detection.

4.7 Strengths and weaknesses of event detection algorithms

in this thesis we catagorized Event detection algorithms into three broad categories: threshold-based, machine learning-based, and deep learning-based. Each has distinct strengths and weaknesses. This qualitative comparison complements the quantitative res-

Table 4.12: F1-Score Comparative Evaluation Summary of Eye Movement Event Detection Algorithms

Model	**Fixation F1**	**Saccade F1**	**PSO F1**	**Cohen's Kappa**
Manual MN/RA	0.98	0.95	0.85	1/0.90
I-DT	0.92	0.88	-	0.6
I-VT	0.93	0.89	-	0.5
RF	0.94	0.90	0.76	0.860
CNN	0.95	0.91	0.78	0.870
LSTM	0.93	0.89	0.74	0.845
RF-CV	0.984	0.919	0.798	0.887
CNN-CV	0.984	0.899	0.794	0.875
LSTM-CV	0.981	0.893	0.747	0.850

ults by highlighting practical considerations such as implementation complexity, data requirements, and limitations in event classification capabilities. The overview includes traditional threshold-based algorithms, machine learning, and deep learning approaches, as well as manual classification by human coders

4.7.1 Threshold-Based Methods

These methods rely on preset criteria like velocity (I-VT), dispersion (I-DT), or acceleration to identify events. Fixations are usually defined by low movement variance, while saccades are identified when thresholds are exceeded [91].

Strengths:

- Simple to implement and interpret.
- No training data required.
- Fast and efficient; ideal for real-time applications.

Weaknesses:

- Performance highly depends on optimal threshold tuning, which varies across participants and tasks.

Table 4.13: Per-Class Precision Scores for Eye Movement Event Detection Algorithms

Model	Fixation Precision	Saccade Precision	PSO Precision
Manual	0.99	0.96	0.86
I-DT	0.91	0.87	–
I-VT	0.92	0.88	–
RF	0.95	0.91	0.75
CNN	0.96	0.92	0.77
LSTM	0.94	0.90	0.73
RF-CV	0.983	0.920	0.810
CNN-CV	0.9859	0.9148	0.7687
LSTM-CV	0.9833	0.8662	0.7449

- Poor at detecting subtle events like PSOs or smooth pursuits.
- Not robust to noise, head drift, or variable sampling rates.

4.7.2 Machine Learning-Based Methods

These approaches use classifiers (e.g., Random Forests, SVMs) trained on labeled gaze data with hand-crafted features such as velocity, direction, or gaze dispersion.

Strengths

- Learn task-specific patterns from data.
- Can handle complex feature interactions.
- Good generalization if trained on diverse datasets.

Weaknesses

- Still relies heavily on feature engineering.
- Sensitive to class imbalance and data quality.
- Hard to transfer across very different tasks without retraining.

4.7.3 Deep Learning-Based Methods

Deep learning models like CNNs, RNNs, and hybrids (e.g., 2D-CNN-LSTM) learn spatiotemporal patterns directly from raw or pre-processed data. They can model both local features (e.g., amplitude) and global dependencies (e.g., transitions between fixations and saccades).

Strengths:

- End-to-end learning without manual feature extraction.
- Superior performance in detecting complex or overlapping events (e.g., SP vs. fixation).
- Scalability across tasks and subjects.

Weaknesses:

- Require large, high-quality labeled datasets.
- Black-box nature: limited interpretability.
- Computationally expensive during training and inference.

Table 4.14 summarizes the key strengths and limitations of event detection methods.

Table 4.14: Summary of Comparison of Event Detection Methods

Method Type	Advantages	Limitations
I-VT / I-DT (Threshold-based)	Simple, interpretable, fast.	Needs manual tuning, poor for subtle events.
Random Forest / SVM (ML)	Learns from data, interpretable features.	Requires feature design, moderate generalization.
CNN / LSTM (Deep Learning)	High accuracy, task-adaptive, no feature engineering.	Data hungry, less interpretable, high computation.

It should be emphasized that the presented result takes into account only point-to-point comparisons, so each gaze point is classified as a part of the specific event. In fact, the event itself always takes some time. For instance, the average fixation duration should be about 250 ms [27]. Therefore, the following typical step in event detection is converting a sequence of subsequent points classified as fixations into one fixation with the location calculated as the median of these points' locations. If there is a gap between fixation sequences (several points classified differently), two sequences are classified as two separate fixations. Obviously, this significantly impacts specific measures like the overall number of fixations and average fixation duration. Therefore, we compared the obtained results

after merging subsequent points. For this comparison we computed duration and count of events for IVT, IDT, RF and CNN without cross validation. The results for I-VT and RF are presented in Figures 4.10 and 4.11, respectively. It occurred that the I-VT algorithm found 189 fixations with an average duration of 121 ms while the RF algorithm found only 64 fixations with an average duration of 264 ms. Considering that the manual coder found 91 fixations with average duration 222 ms, it may be concluded that threshold-based algorithms require the additional step of merging subsequent fixations that are located nearby (hence: additional threshold parameters). In contrast, machine learning algorithms deal with this problem internally. It is clearly visible in Figures 4.10 and 4.11.

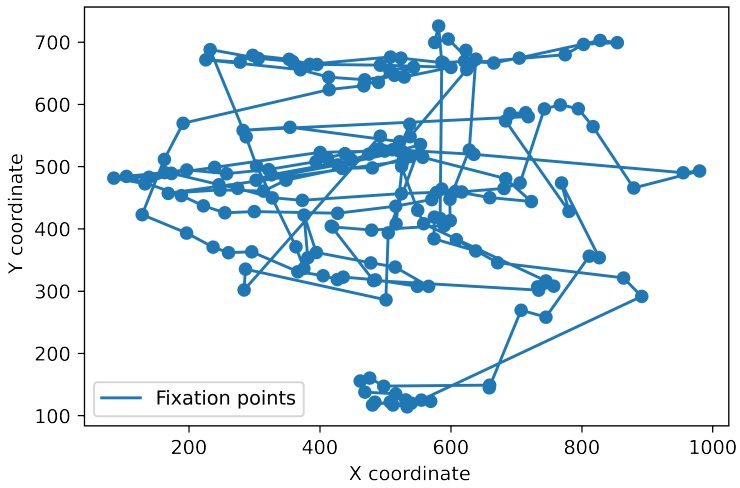


Figure 4.10: Eye fixations obtained from the I-VT algorithm at optimum threshold value of 3.5 px/ms. It is visible that many fixations occur nearby and could probably be combined together.

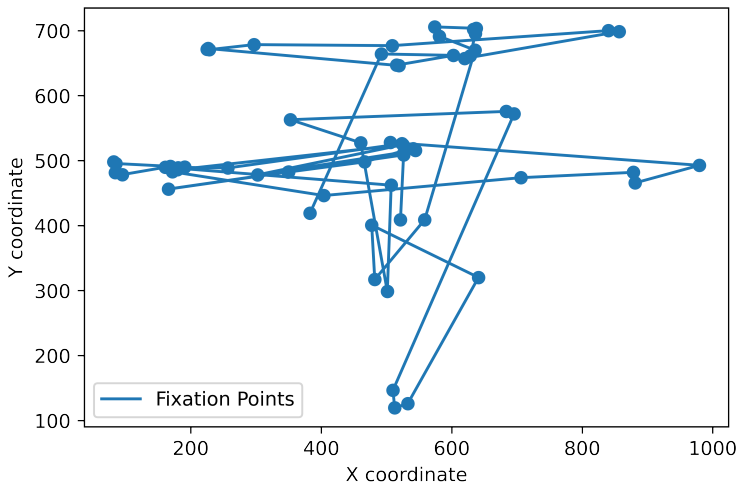


Figure 4.11: Eye fixations obtained from the RF algorithm. Compared to Figure 4.10, there are far fewer fixations.

4.8 Conclusions

This chapter presented a comprehensive evaluation of diffrent eye movement event detection methods, including traditional threshold-based approaches (I-VT, I-DT), machine learning models (Random Forest), and deep learning architectures (CNN, LSTM). Using a unified, expert-labeled dataset and consistent evaluation metrics, including cross-validation and leave-one-fold-out cross-validation (LOFO-CV), the models were assessed on their ability to detect fixations, saccades, and post-saccadic oscillations (PSOs).

The results demonstrate that data-driven methods, particularly CNN and Random Forest, substantially outperform threshold-based approaches in both accuracy and generalizability. These findings support Hypothesis 1 (H1), which stated that traditional threshold-based algorithms are constrained by fixed parameters, binary classification limits, and task-dependent variability, whereas machine learning and deep learning approaches can achieve higher accuracy and robustness. While threshold-based methods provide simple and interpretable heuristics, they are highly sensitive to fixed parameters and perform poorly when detecting events with subtle or overlapping kinematic profiles, such as PSOs. Their limitations are most evident in dynamic visual tasks, where gaze behavior varies widely across users and contexts. Among the machine learning approaches, the CNN model with cross-validation (CNN-CV) shown as the most effective and balanced performer. It achieved high F1-scores across all classes, especially excelling in PSO detection, where it reduced class confusion more effectively than other models. CNN-CV showed strong generalization across folds and minimized misclassification between saccades and PSOs a common challenge due to their temporal adjacency and similar velocity characterstics. Although the Random Forest model slightly outperformed CNN in Cohen's Kappa score, CNN delivered more balanced precision and recall across all classes.

LSTM, while conceptually advantageous due to its ability to model temporal dependencies, lagged behind CNN and RF in overall performance. It proved effective in fixation and saccade classification but showed limitations in handling PSOs, particularly under cross-validation.

Finally, the use of cross-validation LOFO-CV proved critical for assessing model robustness. It exposed performance differences that would be masked by simpler holdout validation, especially in models like LSTM, and emphasized the importance of evaluating generalizability in eye movement event detection systems.

The promising results of the CNN-CV model, along with the potential of recurrent architectures like LSTM to model temporal dependencies in gaze data, motivate further exploration. In the next chapter, we build on these insights by implementing a hybrid CNN-LSTM model that combines spatial feature extraction with temporal sequence modeling. The classification scope is expanded to include smooth pursuits (SPs) in addition to fixations, saccades, and post-saccadic oscillations (PSOs), providing a more compre-

hensive representation of eye movement behaviors in dynamic visual tasks. Furthermore, additional input features such as acceleration, jerk, and direction are introduced to enrich the representation of gaze dynamics. We also investigate how the composition of the target event classes influences model performance, including an in-depth comparison between two alternate configurations: fixation–saccade–PSO (FSPso) and fixation–saccade–smooth pursuit (FSSP). This extension enables a deeper understanding of event separability and the challenges of detecting visually and temporally similar gaze events.

Chapter 5

Eye Movement Event Detection with 2D-CNN-LSTM Networks

5.1 Introduction

This chapter is based on a research paper published in [8], which introduces a hybrid deep learning model combining 2D Convolutional Neural Networks (2D-CNN) with Long-Short-Term Memory (LSTM) networks for the detection of eye movement events. The published work not only proposed this novel model but also extensively analyzed the impact of different feature combinations such as velocity, acceleration, direction, and jerk on classification performance. By evaluating various feature sets, the study provided insights into their contribution to detecting fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs). While the original study focused on video watching datasets, this chapter extends the research by applying the same hybrid model to a new dataset based on image viewing tasks, where it is used to classify fixations, saccades, and PSOs. Furthermore, the model performance is systematically evaluated across different feature sets, allowing for a comparative analysis of how feature selection influences event classification in static image-viewing scenarios. This extension provides valuable insights into the adaptability of the hybrid model across different types of eye-tracking data.

The ability to automatically and simultaneously detect eye movement events such as fixations, saccades, smooth pursuits (SPs), and post saccadic oscillations (PSOs) is fundamental for interpreting gaze behavior. Traditional event detection methods rely on threshold-based algorithms that require manual parameter tuning, making them less adaptive to diverse datasets and experimental conditions. Recent advancements in machine learning and deep learning offer more robust alternatives by leveraging data-driven feature extraction and classification techniques.

Recent advances in deep learning have significantly transformed computer vision and pattern recognition by enabling models to learn hierarchical feature representations dir-

ectly from raw data [59, 95, 35]. In the context of gaze estimation and eye movement analysis, convolutional and recurrent neural networks have demonstrated strong performance in handling appearance variability, temporal dependencies, and noisy signals inherent in eye-tracking data [81, 82, 102]. These developments motivate the adoption of deep learning architectures for eye movement event detection, particularly in scenarios involving complex temporal structure and overlapping event dynamics.

Building on these advances, this thesis proposes a hybrid CNN–LSTM architecture for classifying eye movement events directly from raw gaze data. The convolutional neural network (CNN) component serves as an automatic feature extractor, learning discriminative representations from the input signal, while the long short-term memory (LSTM) component models temporal dependencies to perform event classification. This design is inspired by the work of Startsev, Agtzidis and Dorr [98], who demonstrated the effectiveness of deep learning approaches for eye movement classification, and extends prior work by enabling simultaneous detection of multiple event types within a unified framework.

Before feeding the data into the network, we computed essential motion-related features velocity, acceleration, jerk, and direction from the raw (x,y) gaze coordinates. The model then classifies each data sample into one of four categories: fixations, saccades, PSOs, and SPs. To refine the classifier's output, heuristic measures such as merging nearby fixation points and computing the duration and frequency of detected events were applied, ensuring a more robust event segmentation.

Unlike most event detection methods, which are limited to binary and ternsry classifications, our model simultaneously detects fixations, saccades, PSOs, and SPs. One of the key challenges in event classification is distinguishing PSOs from other events. PSOs often share velocity characteristics with saccades and oscillatory patterns with fixations, making them difficult to identify. To overcome this, we incorporated higher-order velocity derivatives (acceleration, jerk, and direction changes) to improve classification accuracy. Additionally, we evaluated the performance of different combination features, analyzing how individual motion characteristics influence event classification. This systematic feature analysis provides new insights into optimizing deep learning-based event detection across diverse datasets.

Furthermore, this chapter investigates how the structure of the classification task itself influences the performance of the model. Specifically, we conduct an extended evaluation in two alternate three-class configurations: (1) fixation, saccade, and PSO (FSPso), and (2) fixation, saccade, and SP (FSSP). This analysis explores how the inclusion or exclusion of a specific event type affects not only the detection of that event but also the performance on the remaining classes.

This comparison is especially relevant because both PSOs and SPs share signal characteristics with fixations such as overlapping velocity or low positional displacement leading to frequent misclassifications. While SPs tend to be smooth, continuous movements

associated with target tracking, PSOs are brief, oscillatory eye movements that occur immediately after saccades. Their different temporal properties show different modeling challenges. By examining model behavior in these alternate configurations, we aim to gain insight into event-level complexity, inter-class confusion, and model sensitivity to class composition. This extension contributes to a deeper understanding of model generalization and suggests future design considerations for event detection systems.

5.2 Network Architecture

The proposed model is a combination of CNN and LSTM approaches. Such an architecture is frequently referred to as a Long-term Recurrent Convolutional Network (LRCN) [17]. The model takes a sequence of samples and then subsequent convolutional layers are used to extract features that are important for further classification. While the original work [17] worked with images, our model works with eye movement data samples containing preprocessed attributes such as velocity or acceleration. The features are extracted from the input data in convolutional layers are then sent as a sequence to recurrent layers. Recurrent layers work as a network with memory. Their subsequent outputs are generated on the basis of the current input, and all inputs proceeding in a sequence. In contrast to [98], only the last output of the network was used. Thanks to the combination of convolutional and recurrent layers, the network was able to utilize both spatial and temporal features of the signal.

Therefore, the network takes a stream of samples as input. Each sample is a set of different eye movement features, such as direction, velocity, and its higher-order derivatives. The stream of samples is analyzed in windows of 50 samples (which gives a time span of 100 ms in our dataset). To obtain a prediction for each sample, the window moves over the sequence one by one and the sample in the middle of the window is classified according to the neighboring samples. The network comprises different layers, precisely three convolutional layers with 32, 16, and 16 filters and a kernel size of 5. After the convolutional layers, a permute function reshapes the CONV2D output into an LSTM input shape, two LSTM layers with 32 and 16 units, batch normalization layers and one dense output layer.

A sequence of samples of shape $50xN$ where N is the number of features for each combination of features is the input to the network. The network architecture is shown in Figure 5.1.

The model was evaluated using a leave-one-file-out cross-validation approach. In each fold, one file was reserved for testing, while the remaining files were used for training. Sequences of 50 samples were created from the raw data to form input windows for the model. The model was compiled using the RMSprop optimizer with the default learning rate of 0.001, categorical cross-entropy as the loss function, and categorical accuracy as

the evaluation metric. During training, the model was trained for 25 epochs with a batch size of 50. For each training iteration, the labels were binarized using LabelBinarizer to encode the categorical labels into a one-hot format.

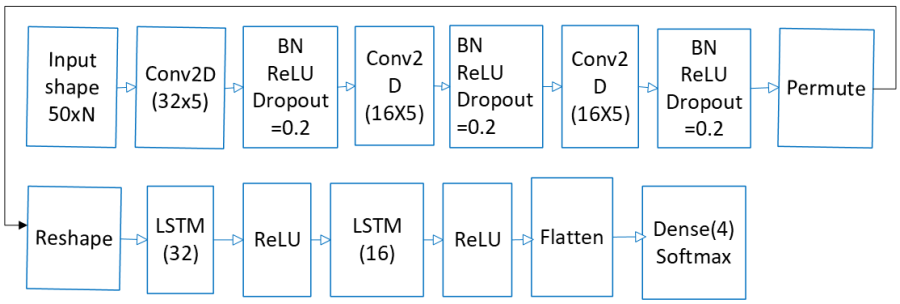


Figure 5.1: The 2DCNN-LSTM Network Architecture.

5.3 Feature Extraction

The main goal was to determine which features and feature combinations would correctly classify fixations, saccades, PSOs, and smooth pursuit events simultaneously. The features used for classification were velocity, acceleration, direction, and jerk. It was hypothesized that velocity and direction alone were insufficient to distinguish PSOs due to the overlapping nature of different event types. For example, velocity and direction were used in [98], and it worked well but only for the detection of fixation, saccade, and smooth pursuit. The velocity feature can distinguish saccades from fixations and smooth pursuits because fixations and smooth pursuits are low-velocity, and saccades are high-velocity movement types. Additionally, direction can distinguish smooth pursuits from fixations because of the uniform distribution of direction in fixations but not in smooth pursuits. However, using only velocity parameters to distinguish fixations from smooth pursuits will not work correctly, as both fixations and SPs are low-velocity movement types. Saccades and PSOs will also be misclassified, as both are high-speed movement types.

Because different features were important in classifying different events, tests were performed to verify which combination of features gave the best event detection results. The velocity, acceleration, jerk and direction are calculated from the raw coordinate points x, y as shown in Figure 5.2. The direction of gaze movement was computed from displacements between consecutive gaze samples, as shown in Equation 5.2.

$$\theta_i = \arctan 2(\Delta y_i, \Delta x_i) \tag{5.1}$$

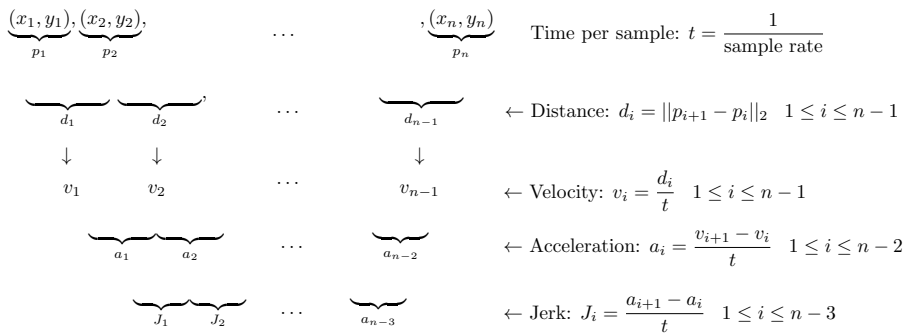


Figure 5.2: Calculation of the features

$$\theta_i = \arctan 2(\Delta x_i, \Delta y_i) \tag{5.2}$$

where $\Delta x_i = x_{i+1} - x_i$ and $\Delta y_i = y_{i+1} - y_i$ denote the horizontal and vertical gaze displacements, respectively. The $\arctan 2$ function ensures that the resulting angle θ_i correctly reflects the direction of motion across all four quadrants of the visual field.

5.4 Performance Evaluation

Currently used fully automated event detection algorithms cannot simultaneously detect all four events (i.e., fixations, saccades, PSOs, and smooth pursuits). Additionally, there is no standard performance metric for eye movement event detectors [99]. So, for a fair comparison, it is recommended that they should be compared to the same metrics for either other algorithms or the same algorithm under different conditions applicable evaluation method [99].

Threshold-based methods were not considered for comparison, as they are statically designed for specific event types based on the threshold value. For example, I-DT and I-VT are designed to detect fixation and saccades only. However, the dataset used for evaluation is annotated with four different event types. Therefore, comparing our method with threshold-based methods would be unfair.

The method was compared with the one most similar to ours, presented in [98]. This method also used a neural network built from a combination of CNN and LSTM layers, but they used 1D-CNN and BLSTM layers. Another difference was that their model output a sequence of event points, while ours classified only one point at a time. In addition, their method was designed to identify only fixations, saccades, and smooth pursuits. It did not consider PSOs, which were tricky to distinguish from fixations, saccades, and smooth pursuits due to their behavioral overlap with these event types.

To compare their model with ours, the model was reimplemented, and the final layer was changed to return only one value, as in ours. In addition, a 50 sample window was used as input to the model. The model was then trained on the same data as used in our

experiments.

However, the primary purpose was to compare the same model for different sets of features. This comparison was made on two levels: sample-level and event-level. For the sample level, the output of the trained model was compared with the ground truth, and classic measures such as F1-score and Cohen's kappa were calculated for each sample. Furthermore, the corresponding confusion matrices for each combination of characteristics—velocity (V), acceleration (A), direction (D) and jerk (J) were analyzed.

Classification of single gaze data points is just a first step in event detection [7]. The next step is to join identically classified neighboring points to form eye movement events. The process is visualized in Figure 5.3. There are many possible measures to compare such results. However, there is no commonly accepted measure for such event-level comparison [99]. Therefore, we decided to compare general events statistics: the number of events and their mean duration. We believe that such information gives a reliable way to compare different results.

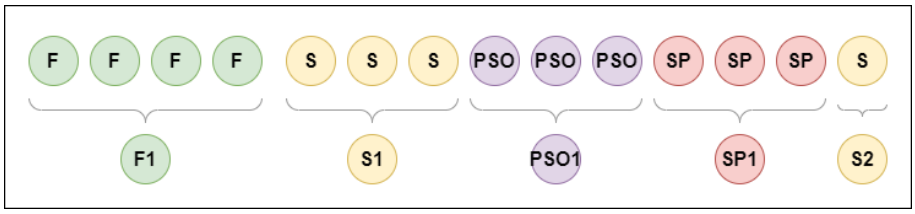


Figure 5.3: Merging neighbouring events. F, S, PSO and SP stand for fixations, saccades, post saccadic oscillations and smooth pursuits respectively.

The train and test dataset split was done using the Leave-One-File-Out (LOFO) cross-validation method to avoid biased and overestimated model performance using a single test set because we repeatedly fit a model to a dataset containing n-1 files. In our case, six files from the dataset described in Section 3 were used.

5.5 Results and Discussions

In this section, the results and discussions on the impact of various feature combinations on the classification performance of different eye movement events are presented. Direction plays a crucial role in identifying smooth pursuit (SP), as it captures the change in the direction of eye movement, which is key to distinguishing SP from other event types. Velocity is essential for distinguishing saccades from fixations and smooth pursuits, as it provides a clear differentiation between high-speed and low-speed movements. Additionally, acceleration and jerk contribute significantly to detecting post-saccadic oscillations (PSOs), as they help to separate PSOs from other movement types based on their rapid, oscillatory nature. Our analysis shows that combinations of velocity, acceleration, jerk, and direction (such as VAJD, VAD, and VJD) improve classification performance, par-

ticularly for PSO detection. We also found that while the absence of jerk in the VAD combination did not significantly affect performance, the lack of velocity in the AJD combination severely impacted the model's ability to correctly classify events.

The performance of different models is presented in Tables 5.1 and 5.2. The models using different feature combinations were compared with ground truth (human coders) as a reference. The baseline model with velocity and direction features, originally used in [98] was also evaluated for fixation, saccades, and smooth pursuit classification. Precision and F1 score were calculated for the proposed models. In order to measure the overall agreement between the manual coders and the proposed methods, Cohen's kappa was calculated between each of the manual coders and the proposed methods. Moreover, confusion matrix analysis was used to analyze the performance of the models.

The experiment started with the combinations of VD and AD feature combinations, which were used in the baseline method [98]. It was discovered that the AD feature combination exhibited very low event detection performance due to its inability to distinguish fixations from smooth pursuits and PSOs from all other event types. A combination of velocity and direction (VD) can identify saccades and PSOs, as velocity can identify saccades from fixations and smooth pursuits. However, the VD combination performs lower for PSO identification because of its behavior similar to saccades with respect to speed.

The results show that using acceleration or jerk in addition to velocity and direction helps the model to distinguish PSOs from other event types. It implies that the model's classification performance with VAJD, VAD, and VJD feature combinations is better than that of AJD, VD, and AD combinations. Not surprisingly, the AJD feature combination cannot classify all event types, especially PSOs. This implies that combining velocity and direction with acceleration and/or jerk shows significant performance improvement for all event types, including PSO.

As the results show, feature selection significantly impacts the event classification results. On the one hand, velocity is crucial for correctly detecting saccades from fixations and smooth pursuits. Hence, these events have low scores for AD and AJD sets. On the other hand, velocity is not sufficient to correctly distinguish between fixations and SPs, as both are low-speed movement types. Therefore, direction is used to detect fixations from SPs. A classic VD set performs better than the former sets for SPs, but its performance is still low for PSOs. Adding acceleration and jerk improves the general performance of the models, especially for PSO detection. A comparison of VAD, VJD, and VAJD sets shows that both acceleration and jerk have a similar impact on the results, and their combination is only marginally better than using only one of them.

Comparison with the results of the 1DCNN-BLSTM model for the same dataset reported in [98] shows that our model performs better and can also distinguish the PSO events. Our modified implementation of the same model does not perform well. However, one of the reasons may be that we changed the model to return only one value instead of

Table 5.1: Comparison of F1-Score(F1) for each event type, mean F1-Score, and mean Cohen's kappa (K)

	F1-Score					
Algorithms	Fixations	Saccades	PSOs	SPs	Mean F1	Mean K
VAJD	0.792	0.918	0.815	0.80	0.83	0.733
VAD	0.803	0.914	0.818	0.803	0.834	0.735
VJD	0.798	0.908	0.795	0.805	0.826	0.723
VD	0.772	0.914	0.797	0.794	0.819	0.708
1DCNN-BLSTM [98] *	0.667	0.72	-	0.663	0.7	0.50
1DCNN-BLSTM(VD) **	0.52	0.79	0.61	0.61	0.63	0.48
AJD	0.644	0.679	0.468	0.641	0.608	0.526
AD	0.37	0.543	0.31	0.464	0.42	0.235

* The 1DCNN-BLSTM row shows the result reported in [98] classifying the three event types: fixation, saccade and SP with VD feature set from the dataset we used in this paper.

** The 1DCNN-BLSTM(VD) is the result of our changed and re-implemented version of their model (classifying four events).

the sequence and skipped the model optimization step.

Misclassifications Analysis

In this section, the misclassifications between events for each feature set are discussed using the confusion matrices, which help to analyze which events were mistaken for others in each feature set.

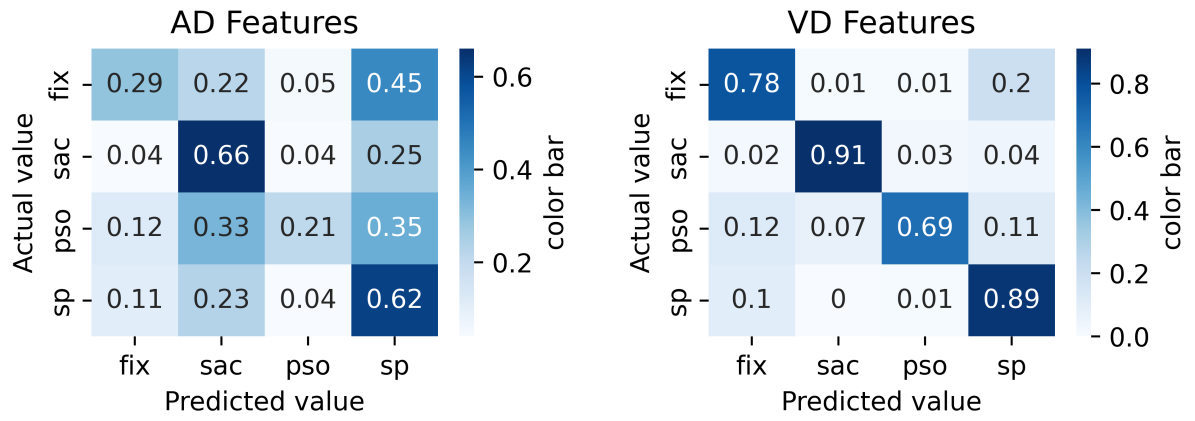
The confusion matrix analysis for the combinations of AD and VD features is shown in Figures 5.4 (a) and 5.4 (b), respectively. Every row in the matrices shows what percent of samples representing the given event was classified as an event given in the corresponding column. The result for the AD combination clearly shows that fixations and SPs are often confused (45% of fixations are classified as smooth pursuits). Moreover, recall of PSOs is very low, and most of the actual PSOs are misclassified (only 21% are classified correctly).

The velocity and direction (VD) combination shows a noticeable improvement over the AD combination. The combination of VD significantly improved the performance of fixation, saccade, and SP identification. However, the performance of the VD combination is lower for correctly identifying PSOs from fixations and smooth pursuits. So, we proposed using combinations of VAJD, VAD, AJD, and VJD to improve the PSO identification performance and reduce misclassifications between PSO and other event types.

Combining velocity and direction with acceleration and/or jerk showed an improvement in the overall identification performance of all event types compared to other feature

Table 5.2: Precision of each event type and features' combination. We compared the performance of the proposed model with different feature combinations.

	Precision Score				
Algorithms	Fixations	Saccades	PSOs	SPs	Mean Precision
VAJD	0.766	0.937	0.935	0.733	0.842
VAD	0.788	0.919	0.935	0.739	0.845
VJD	0.778	0.917	0.921	0.735	0.837
VD	0.7642	0.919	0.932	0.717	0.833
1DCNN-BLSTM(VD)	0.64	0.78	0.79	0.56	0.69
AJD	0.607	0.642	0.804	0.559	0.765
AD	0.517	0.458	0.617	0.371	0.490



(a) (b)

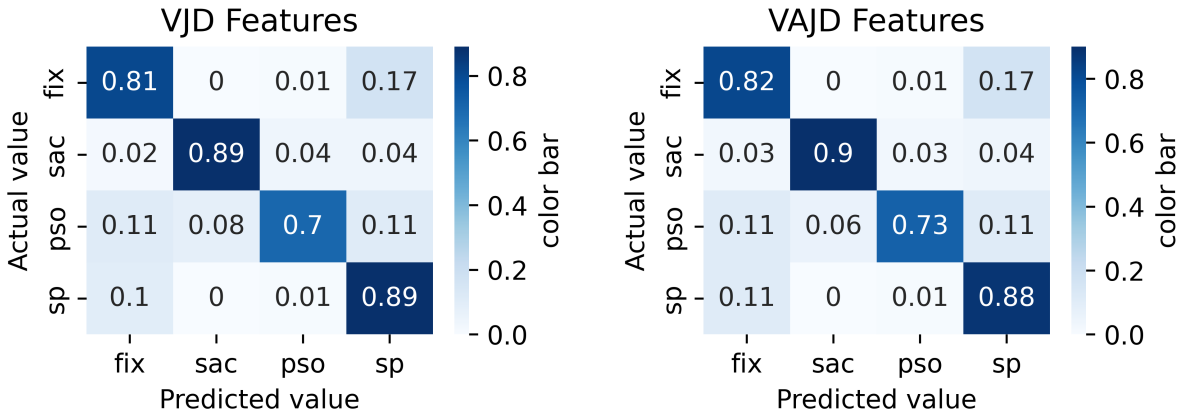
Figure 5.4: The confusion matrix for AD and VD feature combinations.

combinations, especially for PSO detection. However, the performance of PSO identification is still low, and many PSO points are misclassified as fixations and smooth pursuits. The results for these feature combinations are shown in Figure 5.5. For example, the result of the VAJD feature combination in Figure 5.5 (b) shows that 11% of PSOs are misclassified as smooth pursuits and 11% as fixations. The VAD and VJD feature combinations perform approximately the same as VAJD.

In addition, experiments were conducted with combinations without velocity or jerk. A combination of VAD and AJD features was used for the experiment. The result in Figure 5.6 (a) showed that the absence of jerk in the VAD combination does not decrease performance. Regarding the effect of velocity, the AJD in Figure 5.6 (b) has shown that the absence of velocity significantly affects the overall performance. The results imply that combining acceleration and jerk with direction and with the absence of velocity does

Table 5.3: Accuracy for each file and feature combination for Each file in the LOFO Cross validation. We compared the performance of the proposed model with different feature combinations and the state of art appraoch

	Accuracy Score					
Features	File1	File2	File3	File4	File5	File6
VAJD	0.75	0.85	0.78	0.85	0.92	0.92
VAD	0.76	0.85	0.78	0.85	0.92	0.92
VJD	0.75	0.84	0.77	0.85	0.92	0.92
VD	0.72	0.83	0.762	0.84	0.92	0.92
1DCNN-BLSTM(VD)	0.59	0.57	0.55	0.62	0.61	0.66
AJD	0.59	0.74	0.74	0.54	0.79	0.89
AD	0.69	0.47	0.48	0.58	0.18	0.62



(a) (b)

Figure 5.5: The confusion Mmtrix for VJD and VAJD feature combinations.

not improve overall performance for all event types.

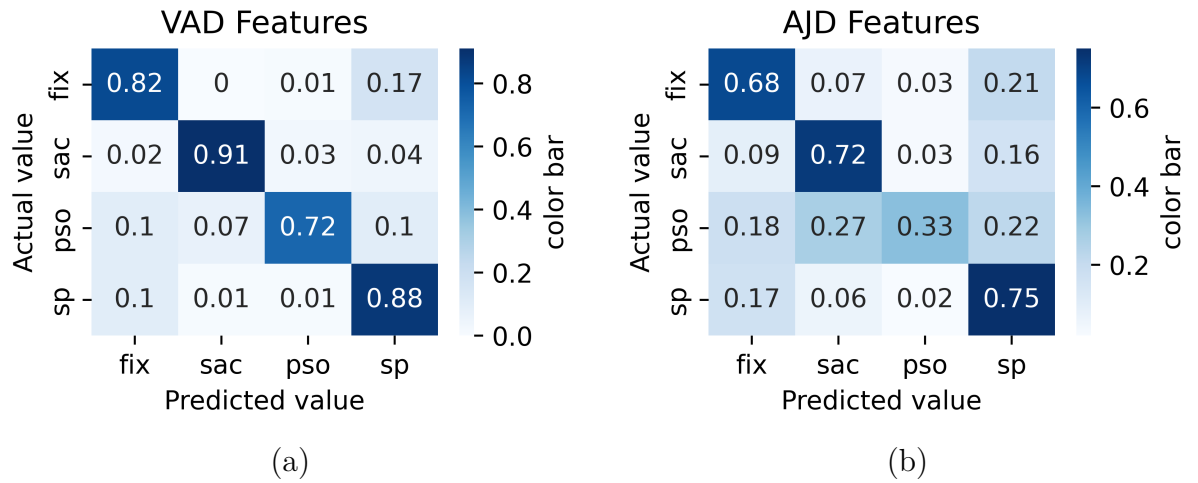
5.6 Event Measures

As stated in Section 4.3, classifying each gaze point as a specific event was just the first step in event detection. The next step was to merge the neighboring points with the same classification into one event, as shown in Figure 5.3. In this way, events with their own properties were created. For instance, fixations had a location (calculated as the mean or median of gaze points), onset, and offset, while saccades and smooth pursuits had start and endpoints.

The main problem of all threshold-based methods is that they are very susceptible to

Table 5.4: Cohen's Kappa for each file and feature combination for Each file in the LOFO Cross validation. We compared the performance of the proposed model with different feature combinations and the state of art appraoch

	Cohens's Kappa					
Features	File1	File2	File3	File4	File5	File6
VAJD	0.60	0.73	0.64	0.73	0.86	0.85
VAD	0.61	0.73	0.63	0.73	0.85	0.85
VJD	0.58	0.72	0.62	0.73	0.86	0.86
VD	0.51	0.70	0.61	0.72	0.85	0.80
1DCNN-BLSTM(VD)	0.31	0.35	0.25	0.33	0.08	0.40
AJD	0.34	0.54	0.57	0.28	0.64	0.80
AD	0.48	0.21	0.19	0.30	0.04	0.19



(a) (b)

Figure 5.6: The confusion matrix for VAD and AJD feature combinations.

outliers. When one gaze point in the middle of a fixation is classified as any other event, it automatically divides the fixation into two separate fixations. The popular solution is to merge subsequent fixations based on their low spatial and temporal distance. However, this introduces two more thresholds that must be manually assigned.

Machine learning-based algorithms are more robust to this problem as they classify gaze points, taking into account not only the sample's properties but also its neighborhood. However, the problem still exists. To analyze this phenomenon for our results, we merged the neighboring event points to form events. Then, for each event type, we calculated two metrics: the mean number of events and the mean event duration.

The results in Table 5.5 show the number of events and mean duration of the events for different combinations of features and for ground truth (manual coding).

Generally, our methods find more fixations than GT, which probably means that some

actual fixations are classified as two or more separate fixations by the model. It is visible, especially for AD and AJD combinations, where the number of detected fixations is about five times larger than it should be. A similar situation can be observed for smooth pursuits. Not surprisingly, when more separate events were detected, the mean durations of events were shorter.

The results show that calculations of event-level statistics, such as the number and duration of events, may reveal additional information about the quality of the model. For instance, the confusion matrix of the VD model in Figure 5.4 (b) suggests that its performance is similar to or even better than VAD, VJD, and VAJD models for smooth pursuit detection. However, event measuring metrics in Table 5.5 show that using the VD model resulted in the creation of much smoother pursuit events than for the VAD, VJD and VAJD models and a shorter duration of these events. So, it may be assumed that the smooth pursuit detection performance of the VD model is lower than that of the VAD, VJD, and VAJD models, even when the sample-level accuracy is better.

As a conclusion from the results, the proposed models with feature combinations VAJD, VAD, and VJD perform approximately the same, outperforming AJD, VD, and AD feature combinations.

Table 5.5: The event measures detected in test data for different feature combinations by manual and the proposed model. The column event measure shows the list of event measuring metrics and the columns VAJD, VJD, VAD, AJD, VD and AD are the results for feature sets.

Metrics	VAJD	VAD	VJD	AJD	VD	AD	GT
No. of Fix	12.5	13.5	14.5	48	18.3	40	10
No of Sacc	14.6	15	15	37	15.3	47.5	15.5
No. of PSOs	11.3	11.3	11.1	19.6	11	26.1	11
No. of SP	9.8	10.8	11.5	58.5	15.3	71	7
Mean FIX-dur	217.2	209.1	203.1	49.3	158.2	23.9	226
Mean Sac-dur	25.1	25.3	25.0	17.4	24.9	25.5	24
Mean PSO-dur	16.8	16.3	16.8	11.0	17.0	11.7	16
Mean SP-dur	403.8	366.2	365.2	60.5	280.3	73.0	508

5.7 Extended Evaluation Using Alternate Event Configurations

The original deep learning model developed in this chapter was trained to classify four types of eye movement events fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs) using a hybrid 2D-CNN-LSTM architecture. This model leveraged spatial-temporal characteristics extracted from multi-dimensional time-series gaze features to capture the dynamics of various eye movement behaviors. While the 4-class setup presents a realistic representation of eye movement dynamics during video watching tasks, its complexity may hinder model performance, especially for subtle or transitional events like PSOs or SPs. These events often share overlapping temporal and velocity characteristics with fixations and saccades, making them more challenging to classify accurately in a multi-class setting.

In the previous section, a core focus was placed on analyzing the effect of feature combinations on model performance. Several configurations were tested, including: VD, VAD, VJD, VAJD feature combinations. After systematically evaluating these combinations, it was observed that the model achieved high and comparable performance when trained on either VAD or VJD features. Therefore, VAD was selected for this extended evaluation.

To explore the effects of different event combinations on classification performance, we conducted additional experiments where the model was trained and tested on three-class classification tasks, by isolating the influence of either PSO or SP alternatively.

This experiment aims to:

- Analyze how the complexity of events affects the detection ability of the model.
- Examine whether reducing class ambiguity improves overall classification.
- Determine which event type PSO or SP is more difficult for the model to detect and distinguish.

The following two configurations were used: Fixation-Saccade-PSO (FSPso) and Fixation-Saccade-Smooth Pursuit (FSSP).

Each experiment was carried out using stratified cross-validation and performance was evaluated based on standard classification metrics: precision, recall, F1 score and precision, as well as Cohen's Kappa.

5.7.1 Performance Comparison and Analysis

Average Classification Performance

The results across six cross-validation folds for both configurations are summarized in Table 5.6. The FSPso setup consistently outperformed the Fixation-Saccade-SP configur-

ation across all metrics, particularly in terms of model agreement (Cohen's Kappa) and classification precision for the fixation class.

The results indicate that while the model shows high classification performance in both cases, it performs significantly better in the FSPso configuration, particularly for fixation detection. The inclusion of SP in the second setup introduces greater overlap with fixation patterns, causing more misclassification and a drop in fixation precision and recall.

Table 5.6: Average precision, recall, F1-score and Cohen's Kappa for FSPso and FSSP Configurations

(a) FSPso configuration

Metric	Fix (FSPso)	Sac (FSPso)	PSO (FSPso)	Accuracy
Precision	0.98	0.90	0.81	0.94
Recall	0.98	0.93	0.78	
F1-score	0.98	0.91	0.79	
Cohen's Kappa	0.88			

(b) FSSP configuration

Metric	Fix (FSSP)	Sac (FSSP)	SP (FSSP)	Accuracy
Precision	0.82	0.91	0.91	0.84
Recall	0.79	0.94	0.94	
F1-score	0.79	0.93	0.93	
Cohen's Kappa	0.72			

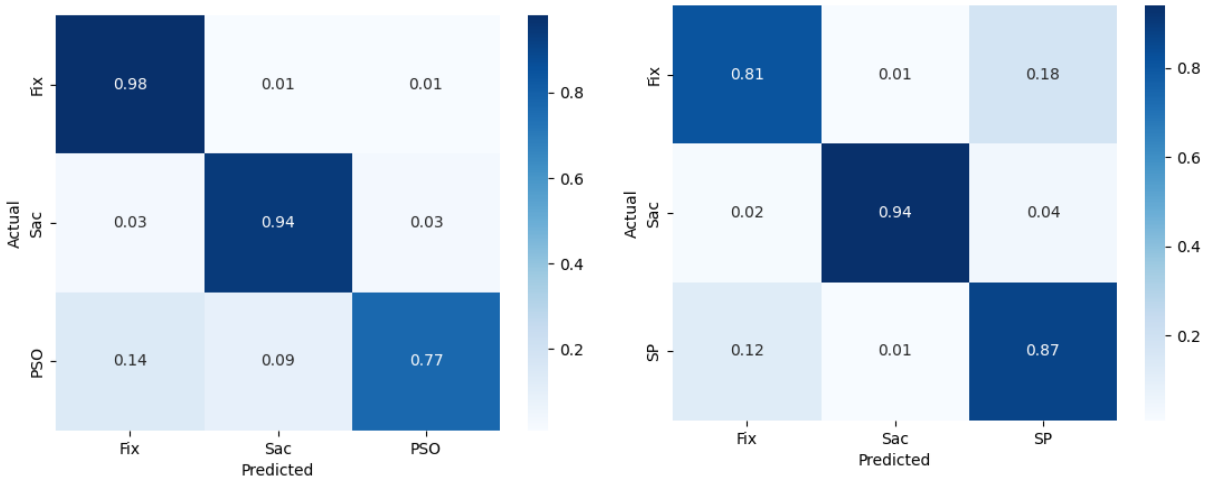
Confusion Matrix Analysis

A detailed analysis of the confusion matrices shown in Figure 5.7(a,b) reveals distinct patterns of misclassification across the two event-set configurations.

In the FSPso configuration (5.7(a)), post-saccadic oscillations (PSOs) exhibit non-negligible misclassification with both saccades and fixations. Specifically, 9% of PSO samples are misclassified as saccades, while 14% are misclassified as fixations. This confusion is likely due to the brief and transitional nature of PSOs, which temporally and kinematically overlap with the termination of saccades, as well as to class imbalance, given that fixations constitute the dominant event type in the dataset. Although PSO detection performance is lower than that of fixations and saccades, its impact on the accurate classification of fixation and saccade events remains minimal.

In contrast, in the FSSP configuration (Figure 5.7(b)), smooth pursuits were more frequently confused with fixations. This confusion is likely due to the similarity in their kinematic profiles, as relatively low velocities and longer durations characterize both event types. This result reflects the inherent difficulty of distinguishing slow eye movements,

particularly smooth pursuits, from fixations using velocity-based features alone



(a) FSPso configuration (b) FSSP configuration

Figure 5.7: The confusion matrices for FSPso and FSSP event configurations

Impact on Agreement and Accuracy

Cohen's Kappa values, were consistently higher for the FSPso configuration (mean: 0.88) compared to FSSP (0.72). This supports the notion that the classification of smooth pursuits introduces more ambiguity into the model, reducing its overall reliability.

The FSSP configuration also demonstrated greater variability in accuracy and class-wise performance across folds, indicating lower robustness in distinguishing SPs from fixations. In contrast, the PSO configuration showed high consistency, although with slightly lower recall for PSOs.

5.7.2 Discussion: Understanding Task Complexity and Model Behavior

These findings provide several insights into the model's strengths and limitations.

Fixations are best detected in the presence of PSOs: In the FSPso setting, fixation detection achieved near-perfect precision and recall (98%). In contrast, fixation detection significantly degraded in the FSSP setup, with precision dropping to 82% and F1-score to 79%. This suggests that SPs, especially when slow and steady, are more likely to be misinterpreted as fixations.

Saccades are consistently detected with high performance across both configurations: Saccades were reliably classified, with F1-scores exceeding 90%. This highlights the effectiveness of temporal convolution and LSTM modeling in capturing the sharp velocity transitions characteristic of saccades.

Smooth pursuits are easier than PSOs to detect but more disruptive: While SPs had higher individual F1-scores (86%) than PSOs (79%), their inclusion negatively impacted fixation detection, reducing overall accuracy and Kappa. This suggests that SPs, while easier to identify on their own, overlap significantly with fixation characteristics, making multi-class discrimination more difficult.

PSOs are subtle and confusable but less harmful to other events: Despite being harder to classify, PSOs did not significantly affect the classification of fixations or saccades. This may be due to their short duration and clear temporal positioning after saccades.

In summary, the model exhibits a high capacity to generalize over complex eye movement patterns, but its performance is strongly affected by the composition of event classes. The choice of which events to include in a classification task is not merely a question of model architecture, but also of inter-class relationships and visual behavior overlap. Future work may explore class-specific strategies to better distinguish between fixations and smooth pursuits in naturalistic settings.

5.8 Conclusion

In this chapter, we proposed a deep learning method to extract eye movement events from raw eye tracking data. The main objective of this study was to find the best feature sets that can be used to build a model capable of classifying four event types simultaneously: fixations, saccades, post-saccadic oscillations (PSOs), and smooth pursuits (SPs). We used velocity, acceleration, jerk, and direction features. Various combinations of these features were evaluated in terms of their impact on classification performance.

We compared the classification performance of different proposed models and a state-of-the-art baseline. The results revealed that combining velocity and acceleration with either direction, jerk, or both leads to significant performance improvements over other feature sets. In particular, the VAJD, VAD, and VJD combinations outperformed AJD, VD, and AD. This finding emphasizes the importance of velocity as a foundational feature, especially when enhanced with higher-order derivatives like acceleration and directionality.

Furthermore, we analyzed the impact of each feature combination on the detection of individual event types. Fixations and saccades were generally well detected using velocity and acceleration alone, whereas SPs and PSOs required additional information such as jerk or direction to be reliably identified. As far as we are aware, this is the first fully automated method capable of simultaneously detecting and classifying all four events: fixations, saccades, SPs, and PSOs. The proposed model was also evaluated using event-level metrics, such as the number of events detected and mean event durations, and these were compared against expert manual annotations.

As with any study, our work coarries certain limitations. Despite achieving competitive

results, classification performance especially for PSOs remains low. PSOs are inherently difficult to detect due to their short duration, low amplitude, and temporal proximity to saccades. Our comparison with related methods was also constrained by the lack of publicly available models capable of detecting all four event types. The closest comparable model did not originally support PSO classification, limiting the fairness of direct comparisons. Furthermore, our evaluation was based on data collected during video viewing tasks; future research should explore the generalizability of these findings to other types of stimuli or tasks, such as reading, static images, or real-world navigation.

he other line of enhancement is based on improving the accuracy of PSO classification in the presence of fixations, saccades, and smooth pursuits. The proposed method improves the recent existing methods by simultaneously classifying fixations, saccade, PSO, and SPs. However, we believe that the proposed approach can be optimized in such a way that the accuracy of the PSO is taken into account

Additionally, applying the proposed method to other datasets with varied sampling rates and gaze behaviors would provide further validation.

Beyond model development and feature engineering, a key contribution of this chapter is the extended evaluation of class composition effects on model performance. Specifically, we investigated how the inclusion or exclusion of PSO and SP events affects detection accuracy and inter-class confusion. Two alternate configurations: fixation-saccade-PSO (FSPso) and fixation-saccade-smooth pursuit (FSSP) were introduced and evaluated using the best-performing VAD feature set. This analysis was motivated by the observation that certain event types overlap in their movement behaviors, particularly in velocity and duration, making them harder to distinguish using automated systems.

The results of this extended analysis revealed that smooth pursuits, although generally easier for the model to classify in isolation, introduced higher confusion rates with fixations when included in the class set. In contrast, PSOs were more difficult to detect directly but did not significantly disrupt classification of fixations or saccades. These findings emphasize that the composition of target classes is a critical factor in the design of event detection models not only in terms of model complexity, but also in how different event types interact and overlap in the feature space.

This detailed understanding of event interaction informs future work on adaptive modeling, class-aware training, and the development of context-sensitive classifiers. Future models may benefit from dynamically adjusting classification criteria based on task demands or temporal context. Overall, the findings in this chapter contribute to both the practical implementation of gaze-based event detection systems and the theoretical understanding of how gaze event definitions influence machine learning outcomes.

Overall, this chapter confirms Hypothesis 2 (H2) by demonstrating that both the selection of appropriate kinematic features and careful consideration of event class composition are critical for robust, simultaneous detection of multiple eye movement events.

Chapter 6

CNN Based Event Detection Model Across Diverse Visual Tasks

6.1 Introduction

Understanding human eye movements is essential for interpreting visual attention, perception, and cognition across a variety of contexts. Automated detection of eye movement events enables large-scale and reproducible analysis in fields such as neuroscience, psychology, and human–computer interaction. However, the variety of oculomotor patterns induced by different types of visual stimuli poses a significant challenge for building robust and generalizable event detection algorithms.

The motivation for the cross-task evaluation conducted in this chapter is grounded in the empirical findings presented in Chapter 3. There, a detailed statistical analysis of eye movement behavior across different visual tasks demonstrated that key event-level characteristics—such as velocity distributions, frequencies, durations, and event proportions vary systematically with task demands. Reading, image viewing, video watching, and moving-dot tracking were shown to elicit distinct oculomotor profiles, with substantial overlap between event types and marked differences in their statistical properties across tasks. These findings highlight that eye movement events are not governed by task-invariant kinematic signatures, but rather reflect task-specific adaptations of the oculomotor system.

In this chapter, we present a two-dimensional convolutional neural network (2D-CNN) based model for eye movement event detection, evaluated across three distinct visual tasks: static image viewing, naturalistic video watching, and controlled moving-dot tracking. Each of these tasks elicits unique oculomotor behaviors due to variations in stimulus motion, attentional demands, and task objectives.

The model is designed to classify three event types: fixations, saccades, and post saccadic oscillations (PSOs). Smooth pursuits (SPs) were excluded from the classification

targets because the image viewing dataset does not contain pursuit events. To ensure fairness and comparability across tasks, SPs were also excluded from the moving dot and video datasets during model training and evaluation.

Although the raw datasets included smooth pursuit (SP) segments, these events were deliberately excluded through a masking procedure applied during label preparation. Specifically, any sequence whose central sample was annotated as SP was reassigned to a placeholder label (-1) and subsequently filtered out prior to training. As a result, SP samples remained present in the input data streams but were not assigned to any output class, and thus did not contribute to the loss function or gradient updates. During evaluation, the same masking procedure was applied so that SP-labeled samples were excluded from accuracy, precision, recall, and F1-score calculations. In other words, SPs were not removed from the datasets entirely, but they were ignored by the classification pipeline. This ensured that the model was trained and evaluated consistently on three event classes, fixations, saccades, and PSOs without being confounded by SP events. The procedure is summarized in Algorithm 1.

The goal of this evaluation is to understand how well the model captures the temporal dynamics of eye movements under varying task conditions, and whether it can generalize beyond the task it was trained on. We examine generalization by training the model on one task and testing it on all three, thereby assessing its robustness across heterogeneous visual and cognitive contexts. We also evaluate performance using multiple metrics: accuracy, Cohen's Kappa, and class-wise precision, recall, and F1-score to obtain a detailed and reliable measure of detection quality. Finally, we interpret differences in model behavior in light of task-specific oculomotor characteristics, linking computational outcomes with underlying physiological and cognitive factors.

Algorithm 1 Masking procedure for excluding smooth pursuits (SPs)

Require: Input sequences X, labels $y \in \{\text{Fix}, \text{Sac}, \text{PSO}, \text{SP}\}$
Ensure: Training and evaluation restricted to {Fix, Sac, PSO}
1: **for** each sequence X_i with central label y_i **do**
2: **if** $y_i = \text{SP}$ **then**
3: $y_i \leftarrow -1$ ▷ Reassign SP to placeholder label
4: Filter out all samples where $y_i = -1$
5: Train model only on remaining classes {Fix, Sac, PSO}
6: During evaluation, apply the same masking so that SP samples do not contribute to accuracy or loss.

6.2 Feature Extraction

Eye movement event classification relies on selecting appropriate kinematic features that effectively capture both transient and sustained oculomotor behaviors. Transient

events such as *saccades* and *post-saccadic oscillations (PSOs)* are characterized by rapid velocity fluctuations and short durations, whereas sustained events such as *fixations* and *smooth pursuits (SPs)* exhibit relatively stable or smoothly varying gaze trajectories.

Features were computed from raw gaze coordinates following standard preprocessing steps, including the computation of velocity using the distance between successive gaze points (see Equation 5.2). From these velocity traces, higher-order derivatives were derived to represent acceleration and jerk,while the movement *direction* was obtained from the arctangent of the horizontal and vertical gaze displacements between consecutive samples (Equation 5.2)

Building on the findings presented in Chapter 5, where feature effectiveness was analyzed in detail for the video-watching dataset, this chapter extends the evaluation to **cross-task generalization**. The model is trained and tested using four distinct feature combinations: VD(Velocity and Direction), VAD(Velocity, Acceleration, and Direction), VJD(Velocity, Jerk, and Direction), VAJD (Velocity, Acceleration, Jerk, and Direction).

The comparative results across these feature sets allow us to assess not only overall classification accuracy but also the robustness of feature representations when applied to new, unseen visual task domains.

Input Representation

To prepare input for the model, features were segmented into fixed-length temporal sequences using a sliding window approach. Specifically, overlapping windows of 50 time steps (stride = 1) were extracted from the continuous gaze recordings. Each window was labeled based on the event occurring at its center frame, allowing the model to learn context around event boundaries. As a result, each input sample had a shape of `(50, 4)`, representing 50 consecutive time steps across the 4 selected feature channels.

6.3 Network Architecture

The proposed eye movement event detection model is a convolutional neural network (CNN) designed to classify short gaze sequences into one of three classes: fixation, saccade, or PSO. Each input is represented as a tensor of shape $(50, N, 1)$, where 50 corresponds to the temporal window length, N denotes the number of kinematic features used (e.g., velocity, acceleration, jerk, direction, in combinations such as VD, VAD, VJD, or VAJD), and 1 represents the channel dimension.

The network is composed of three convolutional blocks, followed by dense layers for classification. Convolutional blocks progressively extract feature–temporal interactions, while dropout and batch normalization are used for regularization. A final softmax layer outputs class probabilities. The architecture is summarized in Algorithm 6.1.

The model was trained with the Adam optimizer using categorical cross-entropy loss, over 10 epochs with a batch size of 32. Labels were one-hot encoded, and training data was prepared using the sliding window method described above.

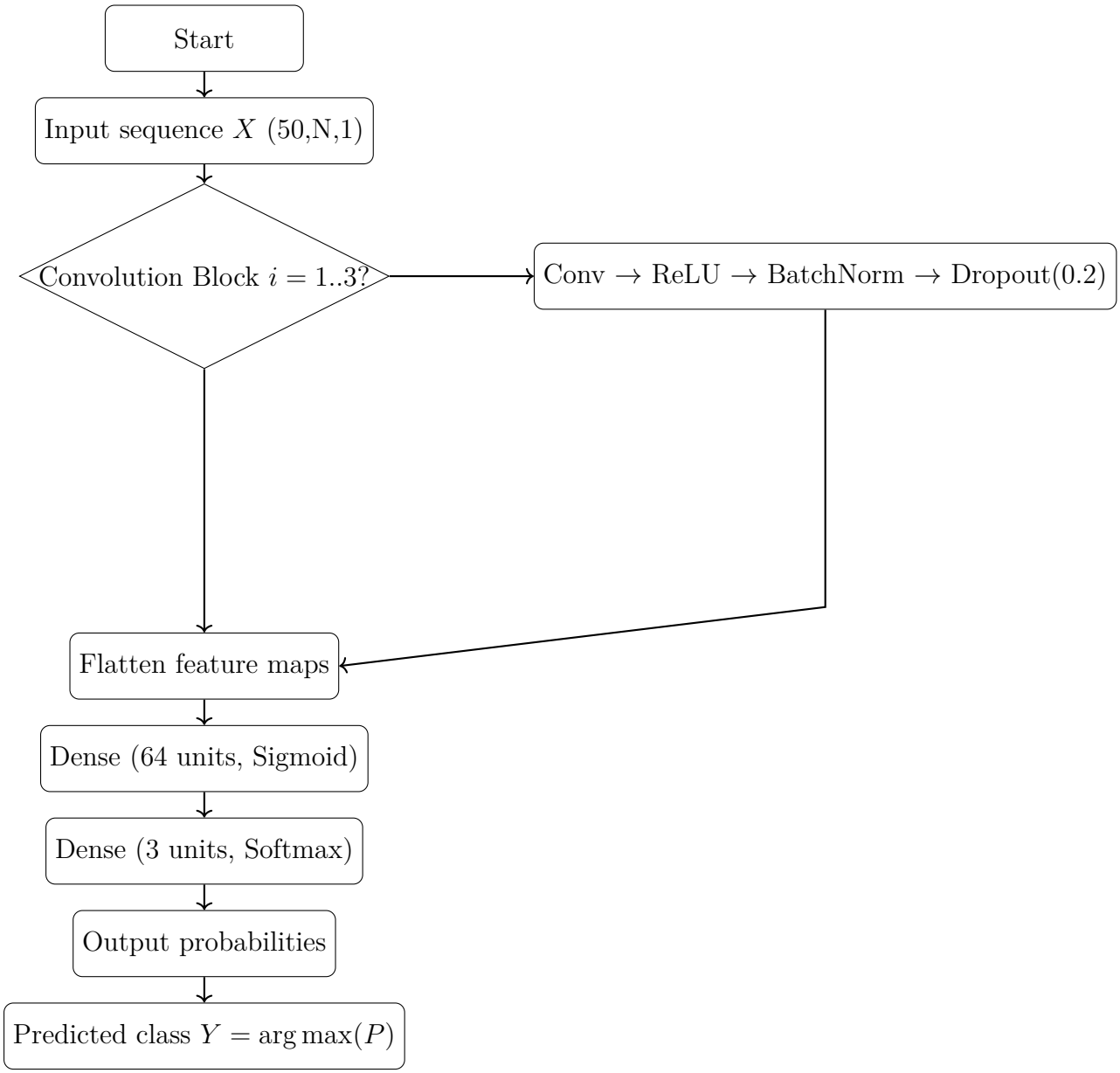


Figure 6.1: Flowchart of the 2D-CNN-based eye movement event detection architecture.

6.4 Dataset Characteristics and Event Distribution

The datasets used for evaluation included three task types: image viewing, video watching, and moving dot tracking. All datasets were manually annotated for fixations, saccades, and PSOs. Smooth pursuits were excluded as explained earlier. Table 6.1 summarizes the distribution of event types across the datasets.

The distribution of eye movement event points across the three datasets reflects the dif-

Table 6.1: Percentage distribution of eye movement events across evaluation datasets.

Dataset	Fixations (%)	Saccades (%)	PSOs (%)	Total Event Points
Image Viewing	84.2%	9.8%	6.1%	28,563
Video Watching	77.8%	15.1%	7.0%	7,545
Dot Tracking	68.2%	25.3%	6.5%	1,312

fering perceptual and oculomotor demands of each visual task. As shown in Table 6.1, fixations dominate all datasets, consistent with their role as the most frequent and stable component of gaze behavior. In contrast, saccades and post-saccadic oscillations (PSOs) occur less frequently but vary systematically across tasks in accordance with task dynamics and stimulus properties.

Saccades were most prevalent in the *moving dot tracking* task (25.3%), intermediate in the *video watching* task (15.1%), and least common during *image viewing* (9.8%). This gradient aligns with the visual control strategies required in each context. In the moving dot task, although the primary oculomotor mode is smooth pursuit, frequent *catch-up saccades* are executed to correct for pursuit lag and maintain foveation on the moving target. The high saccade rate therefore reflects the continual need for rapid positional adjustments during target tracking.

In the video watching condition, the scene's temporal dynamics and semantic richness induce a mixture of pursuit and exploratory behavior. Viewers alternate between smooth pursuit of moving elements and saccadic reorientations toward newly salient regions, producing an intermediate proportion of saccades. Conversely, in the image viewing task, the stimulus is static, and gaze behavior is dominated by prolonged fixations punctuated by occasional exploratory saccades between salient objects or regions, leading to the lowest saccade proportion.

The proportion of PSOs remained relatively consistent across all tasks (approximately 6–7%), suggesting that these micro-oscillatory events are largely task-independent. PSOs arise from the biomechanical properties of the oculomotor plant following saccadic termination and therefore appear with similar frequency whenever saccades occur, regardless of the stimulus context. This stability supports the interpretation that PSOs reflect intrinsic post-saccadic stabilization dynamics rather than task-driven control processes.

6.5 Cross-Dataset Model Evaluation

To test generalization, the model was trained on one dataset (e.g., video) and evaluated on all three (image, video, dot tracking). This design allows us to examine how well the model transfers across task types that differ in stimulus dynamics, gaze patterns, and annotation properties. It also helps identify which training–testing combinations are most compatible and where performance drops occur, thereby providing insights into the task-

dependence of event detection models.

(Table 6.2) below summarizes the classification performance of the event detection model across all train-test combinations for the three visual tasks. The performance of the model was evaluated for all the four feature combinations (VAJD, VAD, VJD and VD) feature combinations. The table presents precision, recall, and F1-scores per class (Fixation, Saccade, PSO), alongside overall accuracy and Cohen's Kappa(K) statistics.

Table 6.2: Per-event-type cross-task generalization performance (F1 / Precision / Recall) for selected feature combinations.

Feature	Train → Test	Fixation (F1 / P / R)	Saccade (F1 / P / R)	PSO (F1 / P / R)	Accuracy	Cohen's Kappa
VAJD						
Image	Image	0.99 / 0.99 / 0.99	0.92 / 0.97 / 0.94	0.91 / 0.81 / 0.86	0.98	0.92
	Video	0.98 / 0.93 / 0.96	0.73 / 0.95 / 0.82	0.65 / 0.53 / 0.58	0.91	0.80
	Dot	0.92 / 0.99 / 0.95	0.88 / 0.72 / 0.79	0.62 / 0.56 / 0.59	0.89	0.75
Video	Image	0.97 / 0.99 / 0.98	0.92 / 0.79 / 0.85	0.65 / 0.60 / 0.62	0.95	0.80
	Video	0.99 / 0.99 / 0.99	0.98 / 0.97 / 0.98	0.92 / 0.91 / 0.91	0.98	0.96
	Dot	0.90 / 0.99 / 0.94	0.89 / 0.64 / 0.74	0.63 / 0.62 / 0.63	0.88	0.74
Dot	Image	0.97 / 0.97 / 0.97	0.76 / 0.75 / 0.75	0.52 / 0.55 / 0.53	0.92	0.76
	Video	0.97 / 0.90 / 0.93	0.64 / 0.77 / 0.70	0.41 / 0.52 / 0.46	0.86	0.67
	Dot	0.99 / 1.00 / 0.99	0.99 / 0.99 / 0.99	0.96 / 0.94 / 0.95	0.99	0.98
VAD						
Image	Image	0.99 / 0.99 / 0.99	0.94 / 0.92 / 0.97	0.89 / 0.90 / 0.87	0.98	0.93
	Video	0.95 / 0.98 / 0.93	0.83 / 0.75 / 0.93	0.63 / 0.62 / 0.65	0.91	0.77
	Dot	0.95 / 0.92 / 0.99	0.78 / 0.91 / 0.69	0.57 / 0.52 / 0.62	0.89	0.75
Video	Image	0.98 / 0.98 / 0.98	0.87 / 0.86 / 0.88	0.67 / 0.68 / 0.65	0.95	0.83
	Video	0.99 / 0.99 / 0.99	0.98 / 0.97 / 0.99	0.91 / 0.91 / 0.91	0.98	0.95
	Dot	0.96 / 0.93 / 0.99	0.79 / 0.84 / 0.75	0.52 / 0.60 / 0.46	0.89	0.76
Dot	Image	0.97 / 0.98 / 0.97	0.76 / 0.78 / 0.73	0.56 / 0.51 / 0.63	0.92	0.73
	Video	0.93 / 0.97 / 0.90	0.68 / 0.60 / 0.77	0.47 / 0.42 / 0.54	0.85	0.63
	Dot	0.99 / 0.99 / 0.99	0.98 / 0.98 / 0.97	0.91 / 0.87 / 0.95	0.96	0.96
VJD						
Image	Image	0.99 / 0.99 / 0.98	0.93 / 0.96 / 0.90	0.81 / 0.70 / 0.96	0.91	0.89
	Video	0.95 / 0.98 / 0.93	0.83 / 0.82 / 0.84	0.61 / 0.50 / 0.79	0.80	0.76
	Dot	0.97 / 0.95 / 0.98	0.80 / 0.92 / 0.70	0.60 / 0.49 / 0.75	0.79	0.78
Video	Image	0.98 / 0.97 / 0.99	0.85 / 0.91 / 0.80	0.66 / 0.67 / 0.65	0.83	0.81
	Video	0.99 / 0.99 / 0.99	0.97 / 0.98 / 0.96	0.91 / 0.92 / 0.90	0.96	0.96
	Dot	0.95 / 0.91 / 0.99	0.76 / 0.86 / 0.69	0.52 / 0.59 / 0.47	0.75	0.73
Dot	Image	0.96 / 0.97 / 0.96	0.72 / 0.65 / 0.87	0.44 / 0.53 / 0.38	0.71	0.69
	Video	0.91 / 0.97 / 0.85	0.64 / 0.50 / 0.89	0.37 / 0.42 / 0.33	0.64	0.57
	Dot	0.99 / 1.00 / 0.99	0.97 / 0.95 / 1.00	0.89 / 0.96 / 0.82	0.95	0.96
VD						
Image	Image	0.99 / 0.99 / 0.99	0.96 / 0.96 / 0.95	0.90 / 0.90 / 0.91	0.98	0.94
	Video	0.95 / 0.98 / 0.92	0.81 / 0.74 / 0.90	0.65 / 0.59 / 0.73	0.90	0.75
	Dot	0.95 / 0.92 / 0.99	0.78 / 0.91 / 0.63	0.60 / 0.52 / 0.71	0.88	0.73
Video	Image	0.98 / 0.98 / 0.98	0.86 / 0.82 / 0.90	0.62 / 0.66 / 0.59	0.95	0.81
	Video	0.99 / 0.99 / 0.99	0.97 / 0.95 / 0.99	0.90 / 0.93 / 0.87	0.98	0.95
	Dot	0.96 / 0.94 / 0.98	0.81 / 0.81 / 0.81	0.43 / 0.66 / 0.32	0.90	0.76
Dot	Image	0.97 / 0.97 / 0.97	0.72 / 0.73 / 0.71	0.51 / 0.49 / 0.53	0.92	0.70
	Video	0.92 / 0.97 / 0.87	0.64 / 0.55 / 0.76	0.49 / 0.42 / 0.57	0.83	0.60
	Dot	0.99 / 0.99 / 1.00	0.98 / 0.98 / 0.97	0.90 / 0.93 / 0.87	0.98	0.96

6.6 Discussion

While all four feature sets were evaluated in the cross-task experiments, VAJD was selected as the primary feature set for the main analyses reported in this chapter. This decision was based on two key observations: First, VAJD consistently yielded the most

robust and balanced performance across different tasks. Second, its feature richness captures both fast transients (saccades, PSOs) and slower, sustained movements (fixations, pursuits). Nonetheless, the comparative results also highlight the usefulness of alternative sets. For example, VAD offers a strong trade-off between performance and computational efficiency; VJD may be preferable when PSO sensitivity is a priority; and VD, though the most minimal, serves as a practical baseline. These findings reinforce VAJD as the most general-purpose feature set for cross-task use, while also clarifying when other sets may be appropriate.

The evaluation results indicate that the proposed event detection model achieves its highest performance when trained and tested within the same visual task domain. In these intra-domain conditions, accuracy consistently exceeds 0.98 and Cohen's Kappa remains above 0.93, reflecting strong agreement between predicted and true labels. For example, when trained and tested on the image dataset, the model achieved an overall accuracy of 0.98 with a Kappa of 0.93, and class-wise F1-scores of 0.99 for Fixations, 0.94 for Saccades, and 0.86 for PSOs. Similarly, training and testing on video data yielded near-perfect results (accuracy = 0.98, Kappa = 0.96, all F1-scores $\geq$ 0.91), and the dot tracking dataset achieved the highest within-domain performance (accuracy = 0.99, Kappa = 0.98, all F1-scores$\geq$ 0.95).

Fixations were classified with consistently high precision, recall, and F1-scores across all experimental settings, rarely falling below 0.90 even in cross-domain evaluations. This robustness likely arises from their stable spatiotemporal characteristics, low velocity, extended duration, and minimal intraclass variability, which are preserved across static and dynamic viewing contexts (see Figure 6.2).

Saccade detection generalized moderately well across tasks, but performance degraded in certain cross-domain scenarios. For instance, when trained on image data and tested on video data, the Saccade F1-score decreased to 0.82 compared to 0.94 in the within-domain case. This reduction may be attributed to differences in saccade amplitude, velocity profiles, and inter-saccadic intervals between static and dynamic scenes. Dynamic viewing conditions, such as videos or moving dots, often elicit anticipatory or catch-up saccades with different temporal dynamics than those observed in static image viewing as shown in (Figure 6.2).

PSO detection was the most affected by cross-domain transfer. While within-domain F1-scores were relatively high (0.86 for images, 0.91 for videos, 0.95 for dots), they dropped significantly in cross-task evaluations, reaching as low as 0.46 when training on dot tracking and testing on video data. This sensitivity reflects the known task-specific variability of PSOs, which are influenced by the preceding saccade's characteristics, luminance changes, and stimulus motion. Such factors vary substantially between image, video, and pursuit-based tasks, limiting the generalizability of PSO representations (see Figure 6.2).

The dot tracking dataset emerged as particularly challenging for cross-domain gener-

alization. Models trained on image or video data achieved relatively low PSO F1-scores on dot tracking (0.59 and 0.63, respectively) and lower Saccade F1-scores (0.79 and 0.74) compared to within-domain results. This difficulty is likely due to the high prevalence of smooth pursuit movements and frequent small-amplitude corrections in the dot tracking task, which disrupt the discrete fixation–saccade–PSO patterns the model learns from other datasets. Conversely, models trained on dot tracking generalized poorly to image or video data, suggesting that the predominance of pursuit dynamics in the training set reduces the model's sensitivity to discrete event boundaries.

Notably, PSO detection tends to show lower accuracy compared to fixations and saccades. This reduced accuracy may partly arise from class imbalance, as PSOs typically appear less frequently and have shorter durations compared to fixations and saccades. Moreover, their overlapping temporal and spatial characteristics with both fixations and saccades increase the likelihood of misclassification, making reliable detection inherently more challenging.

Cohen's Kappa values closely mirrored overall accuracy trends, reinforcing that many observed differences in performance reflect genuine agreement with ground truth annotations rather than merely a class imbalance. However, despite this chance adjustment, the lower detection performance of PSOs likely reflects their physiological properties, relative scarcity, and intrinsic ambiguity in distinguishing them from other event types. (see Figure 6.3)

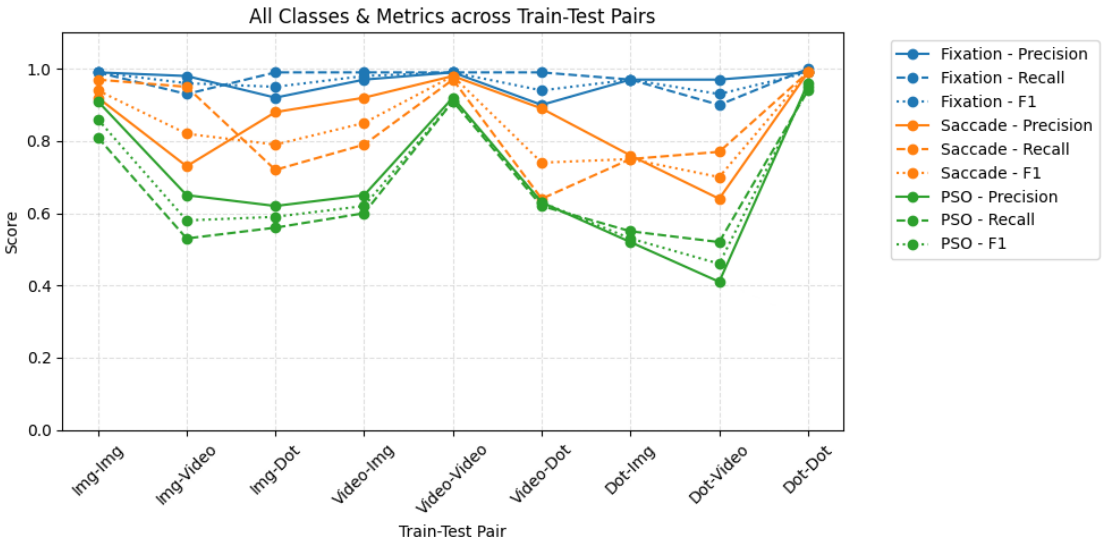


Figure 6.2: The All Classes and Metrics across Train-Test Pairs

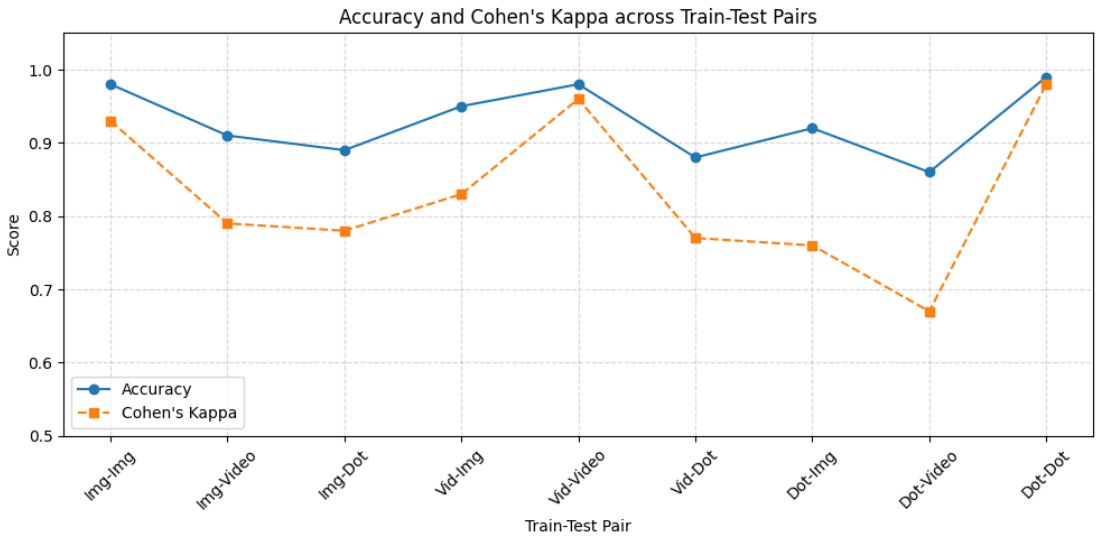


Figure 6.3: Accuracy and Cohenś Kappa across Train-Test Pairs

In summary, the proposed eye movement event detection model demonstrates excellent performance when applied to the same visual task on which it was trained, but its ability to generalize across task domains is more variable and event-type dependent. Fixations, due to their stable and universal temporal–spatial properties, remain robust under domain transfer. Saccades generalize moderately well but show sensitivity to changes in scene dynamics and task demands. PSOs exhibit the greatest domain specificity, reflecting their dependence on the fine-grained dynamics of preceding saccades and stimulus motion. These results highlight the necessity of context-aware model adaptation when deploying event detection systems across diverse viewing conditions and underscore the physiological reality that oculomotor behavior is tightly coupled to the nature of the visual task.

6.7 Conclusion

This chapter systematically assessed the performance of the convolutional neural network based eye movement event detection model across multiple visual tasks. Consistent with Hypothesis 3 (H3), the results demonstrate that the model achieves high accuracy when trained and tested within the same visual task, confirming strong task-specific detection capabilities.

However, when applied across tasks, generalization was more variable, and performance declines were observed, particularly for visually and behaviorally distinct conditions. These declines are consistent with the hypothesis that differences in event frequency, duration, and velocity across tasks (e.g., reading, image viewing, video watching, and moving-dot tracking) influence the statistical properties of eye movement events and affect detection performance.

Across all tasks, detection of PSOs remained lower compared to fixations and saccades.

This is likely due to their lower occurrence rate, shorter duration, and temporal/spatial overlap with other event types, which increases the risk of misclassification. These findings emphasize that, while the CNN model is highly effective within a task, task-dependent variations in oculomotor behavior require task-specific tuning or domain adaptation to maintain robust cross-task performance.

Chapter 7

Examining the influence of PSO Detection on Eye-Tracking During Reading

7.1 Introduction

Post-saccadic oscillations (PSOs) are brief, rapid eye movements that occur immediately after a saccade, before the gaze stabilizes on a fixation. They are commonly observed in eye movement recordings and are typically attributed to overshooting or undershooting of the oculomotor system [3]. Although PSOs are a common feature in eye movement data, they are often overlooked in analysis despite their potential to significantly influence the interpretation of gaze behavior. Neglecting PSOs can lead to misclassifications, particularly between fixations and saccades, compromising the validity of studies that rely on precise eye-tracking measurements. Moreover, ignoring PSOs can distort event statistics-such as average fixation duration, which are widely used to analyze cognitive and linguistic properties of text during reading.

Previous studies have shown that PSO characteristics vary depending on recording techniques and participant populations. Havermann and Lappe [34] and Nyström and Holmqvist [78] demonstrated substantial variability in PSOs across experimental setups, while McConkie and Loschky [71] showed that PSOs can extend the apparent duration of fixations and saccades by up to 20 milliseconds, complicating the interpretation of eye movement measures. These findings suggest that the failure to account for PSOs may systematically bias commonly reported eye-tracking metrics.

Although the destabilizing effect of eye movements immediately after a saccade has been recognized for nearly a century [80], event detection algorithms that explicitly identify PSOs (or glissades) as separate events only began to emerge in the twenty-first century [78, 63]. The early approaches relied on classical methods based on velocity [78]

or acceleration thresholds [63]. More recent work has introduced machine learning techniques, including Random Forest classifiers [112] and neural network–based models [110]. However, many published algorithms still omit PSOs entirely, detecting only fixations, saccades, and smooth pursuits [57, 98]. A similar limitation is present in event detection algorithms provided by major eye-tracker manufacturers, such as SR Research and Tobii, which typically classify eye movement data into fixations and saccades only.

In this chapter, we investigate how explicit detection of PSOs influences eye-tracking measures obtained during reading. We propose an event detection model based on a Convolutional Neural Network (CNN) architecture that classifies gaze data into three event types: fixations, saccades, and PSOs. The model is trained on a dataset manually annotated for all three event classes and is subsequently applied to a separate dataset collected during text reading. We compare the resulting classifications with those produced by the state-of-the-art SR Research algorithm commonly used with the EyeLink 1000 eye tracker, which does not account for PSOs. Our results demonstrate that PSO detection significantly affects fixation statistics commonly used in reading research, including measures related to text complexity, underscoring the importance of incorporating PSOs into eye movement analysis.

This chapter is based on a publication: Examining the Influence of PSO Detection on Eye-Tracking During Reading. Procedia Computer Science, 270, 4204-4212 [9].

7.2 Materials and Methods

This section outlines the methodology used to detect Post Saccadic Oscillations (PSOs) from eye movement. The process involves data collection, preprocessing, feature extraction, and the implementation of a machine learning model to classify eye movements into fixations, saccades, and PSOs.

7.2.1 Datasets

In this chapter, we utilized two datasets discussed in detail in Section 3.2. The first dataset, Lund2013, is a publicly available eye-tracking dataset recorded using a Hi-Speed 1250 eye tracker from SensoMotoric Instruments (Teltow, Germany) at a sampling rate of 500 Hz [2]. Participants in this dataset were presented with static images, text, video clips, and simple moving dot stimuli. The data were manually annotated by two raters, Marcus Nyström (MN) and Richard Andersson (RA), into categories such as fixations, saccades, post-saccadic oscillations (PSOs), smooth pursuit, blinks, and undefined eye movements. For our study, we focused on the image-viewing data labeled with fixations, saccades, and PSOs and we used seven files. The dataset was used to train our CNN model to classify eye movements into three types of events. The dataset can be accessed

from https://github.com/richardandersson/EyeMovementDetectorEvaluation.

The second dataset is the *CopCo* eye-tracking corpus, specifically designed for research in psycholinguistics and natural language processing [39]. Hosted and maintained by the University of Copenhagen, the CopCo corpus analyzes reading behavior in Danish texts across different populations. The eye movement data were collected from native Danish speakers, and the corpus includes two main groups: readers without dyslexia and readers with dyslexia. Additionally, the dataset includes a group of non-native speakers, allowing for a comprehensive analysis of eye movement patterns across varying language proficiencies. In total, the dataset comprises 58 participants - 17 men and 41 women. For this study, we specifically used data from 13 healthy participants. The dataset is preprocessed with SR Research software shipped with the EyeLink 1000 eye tracker and eye movement data are divided into fixations and saccades. The dataset was used to compare the results of the SR Research software with the results of our model. The dataset can be accessed from https://osf.io/ud8s5/.

7.2.2 Data Preprocessing

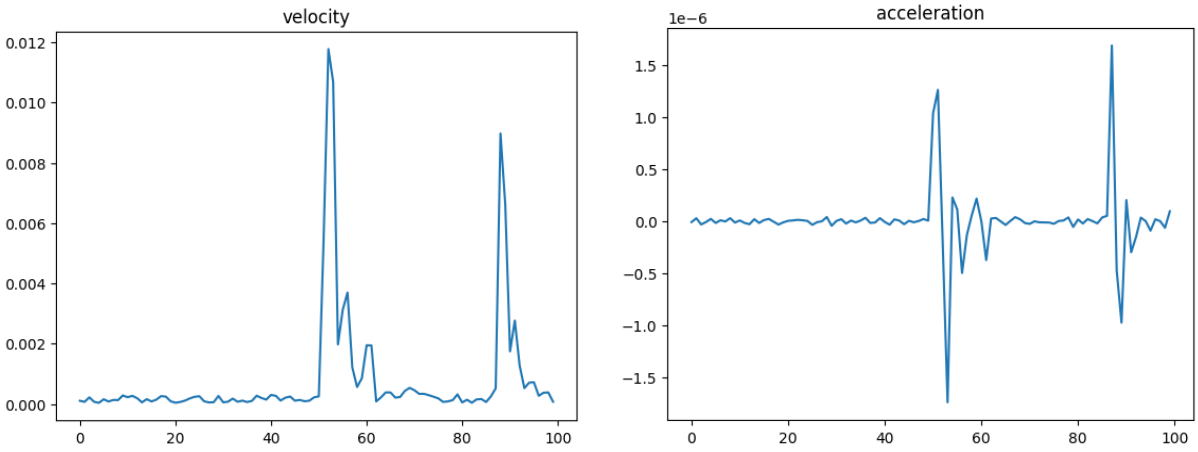
The raw dataset obtained from the CopCo corpus was provided in an ASCII format, that included a significant amount of unnecessary text. To make it suitable for our analysis, several preprocessing steps were necessary to convert it into a usable format. At first all separate recordings from START to END were extracted as separate time series (chunks). Then key columns, including x- and y-coordinates, timestamps, and labels, were extracted. The original raw dataset of eye movements contained markers added by the SR Research software that indicated fixations onsets and offsets (SFIX, EFIX), saccades onsets and offsets (SSACC, ESACC) and blinks (SBLINK, EBLINK). After our preprocessing all samples between SFIX and EFIX were labeled as fixation, between SSACC and ESACC as saccades and all blinks were removed.

7.2.3 Feature Extraction

We used velocity and acceleration features calculated from the raw positional coordinates data points. The velocity of gaze is an obvious and popular choice [91, 56]. Acceleration could aid in saccade detection, as it is also sometimes used in the literature [13, 78, 63] as well as in SR Research's software for the EyeLink trackers.

Velocity and acceleration are calculated from the raw (x, y) coordinate points as shown in Figure 5.2.

Figures 5.2 illustrate the velocity and acceleration signals used as input features for our model. These signals represent the dynamic characteristics of movement over time, where velocity captures the rate of change in position, and acceleration reflects the changes in velocity.



(a). Velocity Pattern (b). Acceleration Pattern

Figure 7.1: Signal pattern for velocity and acceleration.

7.2.4 Convolutional Neural Network

The architecture of our model is based on a 1D Convolutional Neural Network (CNN). The input consists of samples, each sample containing two values: velocity and acceleration of the movement in the specific timestamp. The data stream undergoes processing in sliding windows consisting of 100 samples, corresponding to a temporal duration of 200 ms in our dataset. The prediction for each sample is achieved by sliding the window across the sequence, where the classification of each sample is evaluated based on 50 earlier samples, the sample itself, and 49 later samples. Thus, the network receives input in the form of a sequence of samples structured as $100x2$, with 2 denoting the number of features which are velocity and acceleration.

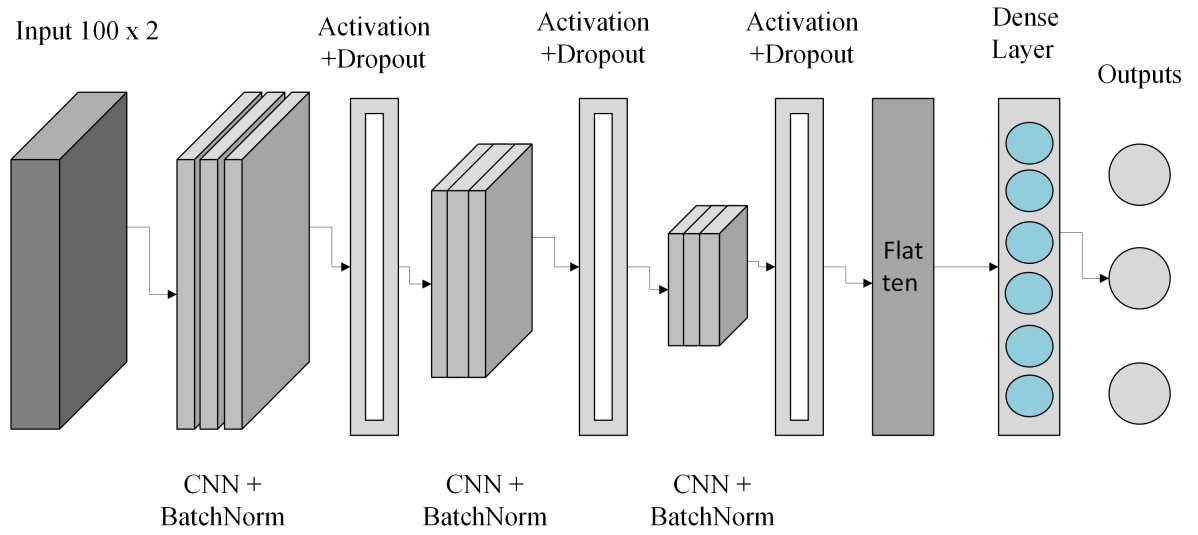


Figure 7.2: Network Architecture

Figure 7.2 represents the CNN architecture used in the study. The architecture is

composed of three convolutional layers, with 32, 64, and 128 filters, respectively, along with a kernel size of 3. The layers are designed to identify and extract both spatial and temporal patterns present in the input data. Following each convolutional layer, batch normalization is implemented to enhance stability and speed up the training process,and dropout layer to mitigate overfitting of the model. The features provided by the convolutional part are then flattened to a vector and sent to the dense layer for actual learning. Finally, a dense output layer is used to generate the final classification for each sample. The output is a vector of three elements representing the probability that the given sample belongs, respectively, to fixation, saccade, or PSO.

The network was trained for 25 epochs, using the Adam optimizer [54]. The categorical cross-entropy loss function was used to measure the error between the predicted and true class labels. Training was performed using a batch size of 100. To evaluate the performance of the model, standard classification metrics were calculated, including precision, precision, recall, F1 score, and Cohen's Kappa coefficient. The confusion matrix was used to analyze misclassifications.

7.3 Results

7.3.1 Building the classification model

The first step in our experiment was to train a model that classifies samples into fixations, saccades, and PSOs. We utilized the data from seven files from the *Lund1013* dataset recorded during static image viewing (therefore, not containing smooth pursuits) and manually labeled into eye movement events (fixations, saccades, and PSOs) by MN-AN as discussed in section 7.2.1.

To evaluate the robustness of the model we used only 80% of available data (23065 samples) to train the model and the rest 20% (5767 samples) to test it. The results presented in Table 7.1 show that the lowest performance was achieved for PSO events. It is not surprising, as it is a class that has the least examples. Moreover, PSOs are difficult to distinguish from saccades and fixations, and even manual annotators differ in their classifications [7].

Table 7.1: The classification performance of the model for test data from Lund2013 dataset

Classes	Precision	Recall	F1-Score	Support
Fixations	0.99	0.99	0.99	4844
Saccades	0.94	0.92	0.93	545
PSOs	0.83	0.83	0.83	378

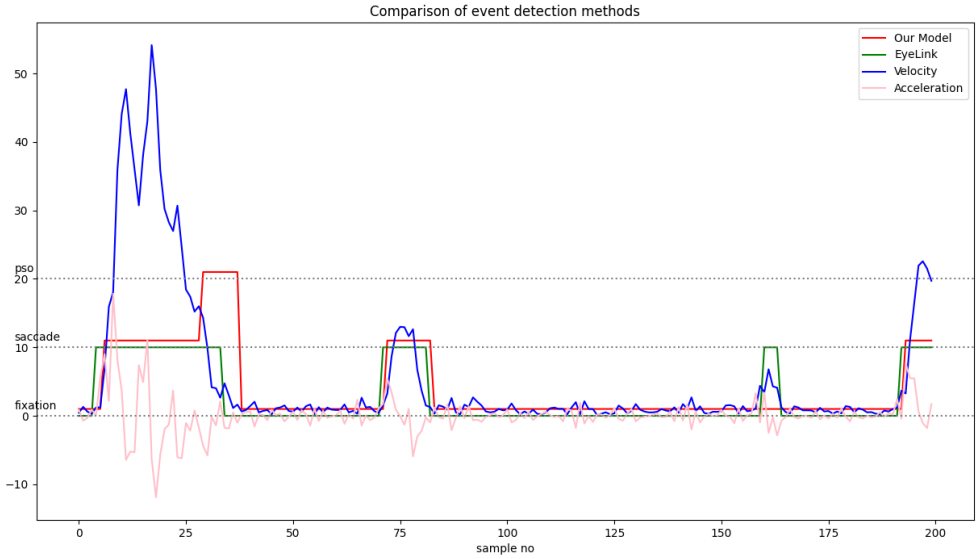


Figure 7.3: Comparison of two event detection methods. Green line shows the results of the SR Research algorithm that classifies events into fixations (low value) or saccades (high value). Red line shows the results of our model that additionally detects PSOs (for better visibility the red line is shifted upward). Blue and pink lines are velocity and acceleration of eye movement added for comparison.

7.3.2 Testing the model

The next step of our research was to use the model described in the previous section to classify raw data from the CopCo dataset into three types of events. Before we could do it, we had to down-sample the data from the *CopCo* dataset from 1000 Hz to 500 Hz because our model was trained on 500 Hz data. Additionally, we added a simple filter that converts samples belonging to fixations shorter than 20 ms to surrounding events.

Figure 7.3 shows an example of the comparison between the classifications of the SR Research state-of-the-art algorithm, named SOTA in the following text (green line) and our CNN model (red line). It is visible that the first saccade (samples 0-30) the onset is a bit later in our model (CNN) and it ends with a PSO that is classified partially as fixation and partially as saccade by the SR Research algorithm (SOTA). The second saccade (samples 70-80) is classified in the same way by both algorithms. The third saccade (samples 160-170) is detected only by the SR Research algorithm.

Another example is presented in Figure **??** which shows how PSOs found by our algorithm relate to the SOTA classification. The first PSO (samples 50-60) is classified by the SOTA algorithm partially as saccade and partially as fixation while the second PSO (samples 190-210) as fixation.

We tested our model on 15 files from the *CopCo* dataset containing 12,829,637 single recordings. Table 7.2 shows comparison of classifications between SR-Research algorithm

(SOTA) and our algorithm. It is visible that about 3% of samples are classified as PSOs. The number of samples classified as fixations by the SOTA algorithm and as PSO by our CNN model is bigger than the number of samples classified as saccades by the SOTA algorithm and as PSO by our CNN model.

Table 7.2: Confusion matrix between the SOTA algorithm that classifies each recording as fixation or saccade and our algorithm that classifies to fixations, saccaded and PSOs

	Classification by our model		
SOTA classification	Fixations	Saccades	PSOs
Fixations	10,476,124 (82%)	124,441 (0.97%)	314,254 (2.4%)
Saccades	279,000 (2.2%)	1,463,021 (11%)	172,797 (1.3%)

7.3.3 Influence on events statistics

The next step was analysis of events calculated from the raw data. All sequences of one type of event were merged and stored as an event with start point and duration. Table 7.3 shows the basic statistics separately for each file. It is visible that the CNN model is characterized with shorter fixations and saccades and the difference is significant.

The duration of fixations decreases on average by 4.91%, which is consistent with the literature [78]. However there are two particppants (P05 and P19) for whom the difference is over 20% and one participant (P016) for whom there is almost no difference.

The duration of saccades also decreases on average by 27% but this value changes for participants from 11% to 49%.

On average 75% of saccades end with PSOs but again these values are significantly different for different files (participants) and range from 51% to 93%.

7.3.4 Analyzing fixation locations shift

Previous steps have shown that the number and durations of fixations differ between the SOTA and CNN models. The next question was whether PSO detection influences the location of fixations. As was shown before, some number of points classified as fixations by the SOTA model are classified as PSOs in the CNN model, so the points in fixations differ and we should expect some distance between these fixations.

To find how big these distances are, we first paired fixations from both models. The fixations were paired if the number of common points (intersection) was greater than 50% of points for both fixations.

We calculated location (x,y) for each fixation by averaging all points belonging to it, and then we calculated the Euclidean distance between both fixations. The results for each file are presented in Table 7.4. It is visible that the distances are quite low.

Table 7.3: Confusion matrix between the SOTA algorithm that classifies each recording as fixation or saccade and our algorithm that classifies to fixations, saccaded and PSOs

file	Fix No		Fix Dur [ms]		Sac No		Sac Dur [ms]		PSO No	PSO Dur [ms]
	SOTA	OUR	SOTA	OUR	SOTA	OUR	SOTA	OUR		
P02	7325	7424	191.68	186.47	7235	7840	38.39	29.79	5754	11.11
P03	11508	11562	190.65	184.01	11384	13148	30.71	25.93	9046	8.30
P04	4641	4666	185.66	181.17	4601	5032	32.79	23.47	4299	11.41
P05	10594	14392	183.99	137.10	10449	15977	41.32	20.48	8252	9.77
P06	9480	9432	173.51	171.64	9378	10125	35.67	27.58	8304	9.78
P07	9545	9638	218.93	210.01	9447	10281	35.33	29.86	8823	10.47
P08	1831	1857	221.54	215.28	1801	1960	32.66	29.23	1029	7.21
P09	2466	2465	184.25	180.13	2442	2751	36.54	25.03	2385	12.88
P10	9101	9120	214.79	210.85	9003	10519	34.27	20.93	9879	12.17
P11	9659	9696	184.22	181.09	9547	10745	34.86	24.93	9145	9.66
P12	6580	6575	214.52	209.35	6500	7329	31.84	23.00	6717	10.93
P15	7651	7705	228.17	224.30	7559	8430	33.79	24.54	6583	10.04
P16	6784	6749	194.08	195.62	6695	7190	34.09	24.67	5392	8.76
P18	6785	8486	188.77	151.30	6712	9425	36.53	20.92	4614	9.73
P19	7527	7549	177.67	172.95	7437	8071	30.43	25.35	5435	9.84
Average			196.83	187.42			34.61	24.05		10.14

Distances are calculated in pixels. According to [39], the size of the display was $590mm \times 335mm$ and the resolution of the screen was 1920×1080. Therefore, the differences in millimeters can be roughly obtained by dividing the distance in pixels by 3. It is therefore visible that average distances between fixations are not higher than two millimeters, which should be sufficient even for demanding fixation-to-word mapping. It occurs that a few points that are added or removed from a fixation do not change its location significantly.

7.4 Discussion

In this chapter, we compared the classification performance of a CNN-based model—which detects fixations, saccades, and PSOs with the traditional SR-Research algorithm that distinguishes only fixations and saccades. Our primary goal was to assess how the inclusion of PSO detection influences common eye-tracking metrics such as event durations and fixation locations.

Our findings indicate that including PSO detection significantly alters the temporal characteristics of eye movement events. Specifically, reclassifying segments as PSOs resulted in shorter measured durations for both fixations and saccades. This observation is in line with previous research; for example, the authors in [78] reported that classify-

Table 7.4: Distances between paired fixations for each file

File	Number of paired fixations	Averaged distance in px
P02	7283	1.38
P03	11439	2.45
P04	4575	2.85
P05	10340	1.67
P06	9375	2.09
P07	9489	1.83
P08	1823	0.98
P09	2431	2.26
P10	9077	0.44
P11	9585	1.06
P12	6546	0.68
P15	7624	1.05
P16	6644	3.89
P18	6516	2.18
P19	7422	0.80

ing PSOs as fixations increases the overall fixation duration by approximately 5%. Our analysis shows that PSO durations typically range between 5 and 15 ms, corroborating findings from Friedman et al [26].

Moreover, our data suggest that long saccades during reading, when eyes return to the beginning of the line, tend to be associated with a higher proportion of PSOs compared to those observed during scene viewing, emphasizing that the dynamic properties of PSOs may depend on the specific visual task [26].

Another important observation from our study is the variability in PSO characteristics across different participants. The differences observed between files suggest that individual oculomotor behavior plays a significant role, making it challenging to draw broad generalizations. This variability calls into question the conclusions of previous studies in which small sample sizes were used: only three participants in [51] and four in [42] underscoring the need for caution when extrapolating results.

Despite the temporal differences, our analysis shows that the spatial distribution of fixations remains largely unaffected by the reclassification process. The average fixation locations, calculated by averaging the coordinates of the gaze points, did not show significant deviations, indicating that while the timing of events changes with PSO detection, the overall spatial mapping of gaze during reading is stable.

Overall, these results highlight both the promise and the challenges of integrating PSO detection into eye-tracking analysis. By providing a more nuanced temporal segmentation of eye movements, our approach contributes to a better understanding of the underlying

oculomotor dynamics during reading. However, individual differences observed and task-specific variations suggest that further research is needed to refine detection algorithms and improve their generalizability in different types of visual tasks

7.5 Conclusions

This chapter focused on the detection and classification of post-saccadic oscillations (PSOs) and their impact on fixation and saccade classification, particularly in reading data eye tracking data. Consistent with Hypothesis 4 (H4), the results demonstrate that explicitly detecting PSOs significantly alters the boundaries between fixations and saccades. Misclassifying PSOs as either fixations or saccades distorts key statistical measures such as event duration, velocity, and frequency. Incorporating PSO detection improves the overall classification accuracy and improves the reliability of eye-tracking analysis in reading research.

In our opinion, PSO detection improves the overall precision of eye-tracking studies and provides more nuanced insights into ocular dynamics insights into ocular dynamics. An important limitation of our study is that the CNN model was trained using eye-tracking data from static image viewing and then applied to data collected during reading. Since eye movement dynamics can vary significantly between these tasks, the performance of the model can be affected when interpreting the reading data. In particular, PSOs are very subtle and hard to detect even in the training data which lowers the accuracy of the model. This means that what the model learned from image viewing data may not fully capture the nuances present during reading, potentially reducing its overall precision in detecting events

To build on the current work, future research can investigate the impact of using different window sizes for data segmentation. Since, our current model processes data using a sliding window of 100 samples (corresponding to 200 ms), which was chosen based on the characteristics of our dataset. Investigating alternative window sizes could reveal more optimal temporal configurations for capturing the dynamic features of eye movements, potentially improving the classification accuracy for PSOs and other events. Second, a comparative study of saccade lengths is recommended. By analyzing how saccade amplitude and duration vary under different conditions, researchers could gain deeper insights into the oculomotor behavior that influences the occurrence and detection of PSO, ultimately leading to a more robust framework for modeling and interpreting eye-tracking data across diverse visual tasks.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

This thesis set out to develop automatic eye movement event detection methods, with the aim of advancing methodological rigor and extending the scope of simultaneously detectable eye movement events.

One of the key contributions of this work is the systematic review and evaluation of eye movement event detection approaches, ranging from manual annotation and threshold-based heuristics to machine learning and deep learning methods. The results presented in Chapter 4 confirm **Hypothesis 1**, which stated that traditional threshold-based algorithms, while widely used, are fundamentally limited by their reliance on manually defined parameters, their binary fixation–saccade classification structure, and their poor generalizability across datasets and tasks, whereas machine learning and deep learning approaches provide more robust alternatives. By conducting all evaluations under consistent experimental conditions using expert-labeled datasets, this thesis provides one of the most comprehensive performance comparisons to date. The results demonstrate that although threshold-based algorithms remain simple and interpretable, they are highly sensitive to parameter choices and perform poorly for subtle or overlapping events. Moreover, they are typically designed to detect only two event types within a single-step framework. In contrast, data-driven approaches-particularly Convolutional Neural Networks (CNNs) and Random Forests-exhibited superior accuracy and robustness, especially for challenging event types such as post-saccadic oscillations (PSOs).

Building upon the findings of Chapter 4, the work presented in Chapter 5 confirms **Hypothesis 2**, which stated that a hybrid CNN-LSTM architecture, when supplied with appropriate kinematic feature combinations, can robustly and simultaneously classify fixations, saccades, post-saccadic oscillations, and smooth pursuits, and that classification performance depends strongly on feature selection. This thesis introduced a hybrid CNN-LSTM framework capable of detecting four core eye movement events-fixations, saccades,

smooth pursuits, and PSOs in a single unified classification step. This represents a novel departure from most prior work, which has predominantly focused on binary or ternary classification schemes. Through systematic analysis of motion-derived features, including velocity, acceleration, jerk, and direction, the results demonstrate that feature choice plays a critical role in separating overlapping event types. For example, acceleration and directional information proved essential for distinguishing PSOs from saccades, while direction was particularly important for differentiating smooth pursuits from fixations. In addition, the thesis provided an extended evaluation of how class composition affects detection performance. Two alternative event configurations-fixation-saccade-PSO (FSPso) and fixation-saccade-smooth pursuit (FSSP) were systematically examined, revealing that smooth pursuits, while relatively easier to detect in isolation, increased confusion with fixations when included, whereas PSOs were more difficult to detect directly but did not substantially interfere with fixation or saccade classification. These findings highlight that robust eye movement detection depends not only on model architecture and feature engineering but also on careful consideration of event set composition.

Another significant contribution is presented in Chapter 6 and confirms **Hypothesis 3**, which proposed that eye movement event detection models perform well within the visual task on which they are trained but exhibit reduced accuracy when applied across different tasks, reflecting task-dependent variability in oculomotor behavior. This chapter examined eye movement behavior across multiple visual tasks—reading, static image viewing, video watching, and moving-dot tracking and evaluated the extent to which detection models generalize across task contexts. While CNN-based models achieved strong within-task performance, cross-task evaluations revealed substantial reductions in accuracy, particularly for post-saccadic oscillations (PSOs). These results highlight the strong task dependence of oculomotor dynamics and underscore the challenges associated with transferring detection models across domains. They further motivate future research into domain-generalized and task-adaptive eye movement detection frameworks.

The final contribution of this thesis is presented in Chapter 7 and confirms **Hypothesis 4**, which proposed that post-saccadic oscillations, though frequently overlooked in eye movement analysis, play a critical role in shaping fixation and saccade statistics, particularly in reading tasks. The results provide clear empirical evidence of the practical impact of PSO detection in reading research. By comparing CNN-based detection results with those obtained using SR Research software, which does not explicitly detect PSOs, the analysis demonstrated that failing to account for PSOs systematically biases commonly used reading measures such as fixation duration. These findings show that PSOs should not be treated as negligible artifacts but rather as meaningful components of gaze behavior, whose misclassification can distort interpretations of cognitive processes during reading.

Overall, this work contributes new insights into both the behavioral characteristics

of eye movements and the computational strategies required for their accurate detection. Specifically, it provides: (i) a comprehensive evaluation of existing eye movement event detection methods, (ii) a novel hybrid deep learning framework with systematic feature contribution analysis, (iii) empirical evidence demonstrating the importance of PSO detection for reading research, and (iv) new findings on the limitations of cross-task generalization. Together, these contributions advance the methodological foundations of automatic eye movement event detection and establish a basis for future systems that are not only accurate within narrowly defined tasks but also robust and adaptable across diverse visual and cognitive contexts.

8.2 Limitations

Despite its contributions, this thesis has several limitations. First, while multiple detection methods were evaluated, comparisons with previous studies are constrained by the lack of standardized benchmarks. Researchers often use different tasks, event definitions, datasets, and annotation schemes, making direct comparisons difficult or potentially misleading. To address this, the evaluations in this thesis were limited to fair comparisons under controlled conditions, either by testing multiple approaches on the same dataset or by systematically varying feature combinations within the same framework.

Second, the datasets used were collected under controlled laboratory conditions with high-quality eye trackers, focusing on tasks such as reading, image viewing, video watching, and moving-dot tracking. While these settings provided well-annotated data for systematic evaluation, they may not fully represent the noise, variability, and ecological complexity of mobile or real-world eye-tracking scenarios.

Third, feature extraction was restricted to kinematic properties derived from gaze coordinates (velocity, acceleration, jerk, and direction). Other potentially informative signals such as pupil size, eyelid movements, or multimodal data like EEG were not explored, even though they can help disambiguate events such as fixations and pursuits.

Finally, although post-saccadic oscillations were a central focus of this thesis, a key limitation was the absence of manually annotated PSOs in the reading dataset. Manual annotation of PSOs is extremely time-consuming and requires expert knowledge, which limited the depth of validation possible in this context. As a result, the CNN-based detection results for PSOs in reading could not be directly benchmarked against ground-truth human labels, which would have strengthened the analysis.

8.3 Future Work

The findings of this thesis open several promising directions for future research:

- Developing larger manually annotated reading datasets including PSOs is a critical next step. Although time-consuming, such datasets would provide a stronger foundation for evaluating models and understanding the role of PSOs in reading behavior.
- Domain adaptation for cross-task transfer results revealed significant limitations in cross-task generalization. Future work should investigate domain adaptation, transfer learning, or task-agnostic representations that allow models to generalize across diverse visual and cognitive contexts.
- Extending evaluation beyond laboratory-controlled conditions to mobile and naturalistic environments will improve ecological validity and test the robustness of detection models in everyday contexts.
- Incorporating complementary signals such as pupil size, EEG, or head movement could improve classification of ambiguous events (e.g., PSOs vs. saccades) and enable richer interpretations of oculomotor behavior.
- Dedicated work on PSO detection is warranted, given their subtlety and overlap with saccades. Future approaches may benefit from advanced feature engineering, adaptive temporal windows, or attention-based deep learning models designed to capture fine-grained oscillatory dynamics.
- While accuracy was the primary focus here, optimizing detection algorithms for computational efficiency and low-latency performance will be crucial for real-world applications in human-computer interaction, assistive technologies, and neurology.

Bibliography

[1] Richard V Abadi, David Carden and John Simpson. 'Listening for eye movements'. In: *Ophthalmic and Physiological Optics* 1.1 (1981), pp. 19–27.

[2] Richard Andersson, Linnea Larsson, Kenneth Holmqvist, Martin Stridh and Marcus Nyström. 'One algorithm to rule them all? An evaluation and discussion of ten eye movement event-detection algorithms'. In: *Behavior research methods* 49 (2017), pp. 616–637.

[3] A Terry Bahill, Michael R Clark and Lawrence Stark. 'Glissades—eye movements generated by mismatched components of the saccadic motoneuronal control signal'. In: *Mathematical Biosciences* 26.3-4 (1975), pp. 303–318.

[4] A Terry Bahill, Michael R Clark and Lawrence Stark. 'The main sequence, a tool for studying human eye movements'. In: *Mathematical biosciences* 24.3-4 (1975), pp. 191–204.

[5] Werner Becker. 'Metrics'. In: *Eye Movements: Cognition and Visual Perception*. Ed. by R. H. S. Carpenter. Lawrence Erlbaum Associates, 1989, pp. 13–67.

[6] Sylwester Białowąs and Adrianna Szyszka. 'Eye-tracking in marketing research'. In: *Managing Economic Innovations–Methods and Instruments* 1.69 (2019), pp. 91–104.

[7] Birtukan Birawo and Pawel Kasprowski. 'Review and evaluation of eye movement event detection algorithms'. In: *Sensors* 22.22 (2022), p. 8810.

[8] Birtukan Adamu Birawo and Pawel Kasprowski. 'Performance Analysis of Eye Movement Event Detection Neural Network Models with Different Feature Combinations'. In: *Applied Sciences* 15.11 (2025), p. 6087.

[9] Birtukan Adamu Birawo, Pawel Kasprowski and Mohd Faizan Ansari. 'Examining the Influence of PSO Detection on Eye-Tracking During Reading'. In: *Procedia Computer Science* 270 (2025), pp. 4204–4212.

[10] Pieter Blignaut. 'Fixation identification: The optimum threshold for a dispersion algorithm'. In: *Attention, Perception, & Psychophysics* 71 (2009), pp. 881–895.

[11] Maria Borgestig, Jan Sandqvist, Richard Parsons, Torbjörn Falkmer and Helena Hemmingsson. 'Eye gaze performance for children with severe physical impairments using gaze-based assistive technology—A longitudinal study'. In: *Assistive technology* 28.2 (2016), pp. 93–102.

[12] Christian Braunagel, David Geisler, Wolfgang Stolzmann, Wolfgang Rosenstiel and Enkelejda Kasneci. 'On the necessity of adaptive eye movement classification in conditionally automated driving scenarios'. In: *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*. 2016, pp. 19–26.

[13] Han Collewijn and Ernst P Tamminga. 'Human smooth and saccadic eye movements during voluntary pursuit of different target motions on different backgrounds.' In: *The Journal of physiology* 351.1 (1984), pp. 217–250.

[14] Morten la Cour and Berndt Ehinger. 'The retina'. In: *Advances in Organ Biology* 10 (2005), pp. 195–252.

[15] Heiner Deubel and Bruce Bridgeman. 'Fourth Purkinje image signals reveal eye-lens deviations and retinal image distortions during saccades'. In: *Vision research* 35.4 (1995), pp. 529–538.

[16] Leandro L Di Stasi, Rebekka Renner, Peggy Staehr, Jens R Helmert, Boris M Velichkovsky, José J Cañas, Andrés Catena and Sebastian Pannasch. 'Saccadic peak velocity sensitivity to variations in mental workload'. In: *Aviation, space, and environmental medicine* 81.4 (2010), pp. 413–417.

[17] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko and Trevor Darrell. 'Long-term recurrent convolutional networks for visual recognition and description'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 2625–2634.

[18] Mehmet Donmez. 'The Use of Eye-tracking Technology in Education'. In: (2023).

[19] Michael Dorr, Thomas Martinetz, Karl R Gegenfurtner and Erhardt Barth. 'Variability of eye movements when viewing dynamic natural scenes'. In: *Journal of vision* 10.10 (2010), pp. 28–28.

[20] Andrew T Duchowski and Andrew T Duchowski. *Eye tracking methodology: Theory and practice*. Springer, 2017.

[21] Maria K Eckstein, Belén Guerra-Carrillo, Alison T Miller Singley and Silvia A Bunge. 'Beyond eye gaze: What else can eyetracking reveal about cognition and cognitive development?' In: *Developmental cognitive neuroscience* 25 (2017), pp. 69–91.

[22] Niels Ehlers and Jesper Hjortdal. 'The cornea: epithelium and stroma'. In: *Advances in organ biology* 10 (2005), pp. 83–111.

[23] Carlos Elmadjian, Candy Gonzales, Rodrigo Lima da Costa and Carlos H Morimoto. 'Online eye-movement classification with temporal convolutional networks'. In: *Behavior Research Methods* (2022), pp. 1–19.

[24] Casper J Erkelens and Ingrid MLC Vogels. 'The initial direction and landing position of saccades'. In: *Studies in Visual Information Processing*. Vol. 6. Elsevier, 1995, pp. 133–144.

[25] Lee Friedman, Vladyslav Prokopenko, Shagen Djanian, Dmytro Katrychuk and Oleg V Komogortsev. 'Factors affecting inter-rater agreement in human classification of eye movements: a comparison of three datasets'. In: *Behavior Research Methods* 55.1 (2023), pp. 417–427.

[26] Lee Friedman, Ioannis Rigas, Evgeny Abdulin and Oleg V Komogortsev. 'A novel evaluation of two related and two independent algorithms for eye movement classification during reading'. In: *Behavior Research Methods* 50 (2018), pp. 1374–1397.

[27] Niels Galley, Dirk Betz and Claudia Biniossek. 'Fixation durations-Why are they so highly variable'. In: *Das Ende von Rational Choice? Zur Leistungsfähigkeit der Rational-Choice-Theorie* 93 (2015), pp. 83–106.

[28] Kerstin Gidlöf, Annika Wallin, Richard Dewhurst and Kenneth Holmqvist. 'Using eye tracking to trace a cognitive process: Gaze behaviour during decision making in a natural environment'. In: *Journal of eye movement research* 6.1 (2013).

[29] Alessandro Grillini, Daniel Ombelet, Rijul S Soans and Frans W Cornelissen. 'Towards using the spatio-temporal properties of eye movements to classify visual field defects'. In: *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. 2018, pp. 1–5.

[30] Dan Witzner Hansen and Qiang Ji. 'In the eye of the beholder: A survey of models for eyes and gaze'. In: *IEEE transactions on pattern analysis and machine intelligence* 32.3 (2009), pp. 478–500.

[31] Katarzyna Harężlak and Paweł Kasprowski. 'Evaluating quality of dispersion based fixation detection algorithm'. In: *Information Sciences and Systems 2014: Proceedings of the 29th International Symposium on Computer and Information Sciences*. Springer. 2014, pp. 97–104.

[32] Christopher M. Harris and Daniel M. Wolpert. 'Signal-dependent noise determines motor planning'. In: *Nature* 394 (1998), pp. 780–784.

[33] Hamilton Hartridge and LC Thomson. 'Methods of investigating eye movements'. In: *The British journal of ophthalmology* 32.9 (1948), p. 581.

[34] Katharina Havermann and Markus Lappe. 'The influence of the consistency of postsaccadic visual errors on saccadic adaptation'. In: *Journal of Neurophysiology* 103.6 (2010), pp. 3302–3310.

[35] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. 'Deep Residual Learning for Image Recognition'. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[36] JM Henderson and A Hollingworth. 'High-level scene perception. annual review of psychology'. In: (1999).

[37] John M Henderson. 'Human gaze control during real-world scene perception'. In: *Trends in cognitive sciences* 7.11 (2003), pp. 498–504.

[38] Roy S Hessels, Richard Andersson, Ignace TC Hooge, Marcus Nyström and Chantal Kemner. 'Consequences of eye color, positioning, and head movement for eye-tracking data quality in infant research'. In: *Infancy* 20.6 (2015), pp. 601–633.

[39] Nora Hollenstein, Marina Björnsdóttir and Maria Barrett. 'CopCo: The Copenhagen Corpus of Eye-Tracking Recordings from Natural Reading'. In: (2022).

[40] Nils Holmberg, Helena Sandberg and Kenneth Holmqvist. 'Advert saliency distracts children's visual attention during task-oriented internet use'. In: *Frontiers in Psychology* 5 (2014), p. 51.

[41] Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka and Joost Van de Weijer. *Eye tracking: A comprehensive guide to methods and measures*. OUP Oxford, 2011.

[42] Ignace Hooge, Marcus Nyström, Tim Cornelissen and Kenneth Holmqvist. 'The art of braking: Post saccadic oscillations in the eye tracker signal decrease with increasing saccade size'. In: *Vision research* 112 (2015), pp. 55–67.

[43] Ignace TC Hooge, Diederick C Niehorster, Marcus Nyström, Richard Andersson and Roy S Hessels. 'Is human classification by experienced untrained observers a gold standard in fixation detection?' In: *Behavior Research Methods* 50 (2018), pp. 1864–1881.

[44] Sabrina Hoppe and Andreas Bulling. 'End-to-end eye movement detection using convolutional neural networks'. In: *arXiv preprint arXiv:1609.02452* (2016).

[45] Sabrina Hoppe, Tobias Loetscher, Stephanie A Morey and Andreas Bulling. 'Eye movements during everyday behavior predict personality traits'. In: *Frontiers in human neuroscience* 12 (2018), p. 105.

[46] Laurent Itti and Christof Koch. 'A saliency-based search mechanism for overt and covert shifts of visual attention'. In: *Vision research* 40.10-12 (2000), pp. 1489–1506.

[47] Robert JK Jacob and Keith S Karn. 'Eye tracking in human-computer interaction and usability research: Ready to deliver the promises'. In: *The mind's eye*. Elsevier, 2003, pp. 573–605.

[48] Qiang Ji and Xiaojie Yang. 'Real-time eye, gaze, and face pose tracking for monitoring driver vigilance'. In: *Real-time imaging* 8.5 (2002), pp. 357–377.

[49] Antony William Joseph and Ramaswamy Murugesh. 'Potential eye tracking metrics and indicators to measure cognitive load in human-computer interaction research'. In: *J. Sci. Res* 64.1 (2020), pp. 168–175.

[50] Marcel A Just and Patricia A Carpenter. 'A theory of reading: from eye fixations to comprehension.' In: *Psychological review* 87.4 (1980), p. 329.

[51] ZA Kapoula, DA Robinson and TC Hain. 'Motion of the eye immediately after a saccade'. In: *Experimental Brain Research* 61 (1986), pp. 386–394.

[52] Pawel Kasprowski, Katarzyna Harezlak and Sabina Kasprowska. 'Development of diagnostic performance & visual processing in different types of radiological expertise'. In: *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. 2018, pp. 1–6.

[53] Eun Yi Kim, Sin Kuk Kang, Keechul Jung and Hang Joon Kim. 'Eye mouse: mouse implementation using eye tracking'. In: *2005 Digest of Technical Papers. International Conference on Consumer Electronics, 2005. ICCE.* IEEE. 2005, pp. 207–208.

[54] Diederik P Kingma and Jimmy Ba. 'Adam: A method for stochastic optimization'. In: *arXiv preprint arXiv:1412.6980* (2014).

[55] Oleg V Komogortsev, Denise V Gobert, Sampath Jayarathna, Do Hyong Koh and Sandeep M Gowda. 'Standardization of automated analyses of oculomotor fixation and saccadic behaviors'. In: *IEEE Transactions on biomedical engineering* 57.11 (2010), pp. 2635–2645.

[56] Oleg V Komogortsev and Alex Karpov. 'Automated classification and scoring of smooth pursuit eye movements in the presence of fixations and saccades'. In: *Behavior research methods* 45 (2013), pp. 203–215.

[57] Rakshit Kothari, Zhizhuo Yang, Christopher Kanan, Reynold Bailey, Jeff B Pelz and Gabriel J Diaz. 'Gaze-in-wild: A dataset for studying eye and head coordination in everyday activities'. In: *Scientific reports* 10.1 (2020), p. 2539.

[58] Eileen Kowler. 'Eye movements: The past 25 years'. In: *Vision research* 51.13 (2011), pp. 1457–1483.

[59] Alex Krizhevsky, Ilya Sutskever and Geoffrey E Hinton. 'Imagenet classification with deep convolutional neural networks'. In: *Advances in neural information processing systems* 25 (2012), pp. 1097–1105.

[60] Michael F Land and Peter McLeod. 'From eye movements to actions: how batsmen hit the ball'. In: *Nature neuroscience* 3.12 (2000), pp. 1340–1345.

[61] Linnéa Larsson. 'Event detection in eye-tracking data for use in applications with dynamic stimuli'. PhD thesis. Lund University, 2016.

[62] Linnéa Larsson, Marcus Nyström, Richard Andersson and Martin Stridh. 'Detection of fixations and smooth pursuit movements in high-speed eye-tracking data'. In: *Biomedical Signal Processing and Control* 18 (2015), pp. 145–152.

[63] Linnéa Larsson, Marcus Nyström and Martin Stridh. 'Detection of saccades and postsaccadic oscillations in the presence of smooth pursuit'. In: *IEEE Transactions on biomedical engineering* 60.9 (2013), pp. 2484–2493.

[64] R John Leigh and David S Zee. *The neurology of eye movements*. Oxford university press, 2015.

[65] Larry S Liebovitch. 'Why the eye is round'. In: *Advances in Organ Biology* 10 (2005), pp. 1–19.

[66] Stephen G Lisberger, Edward Joseph Morris and Lawrence Tychsen. 'Visual motion processing and sensory-motor integration for smooth pursuit eye movements.' In: *Annual review of neuroscience* 10 (1987), pp. 97–129.

[67] Casimir JH Ludwig and Iain D Gilchrist. 'Measuring saccade curvature: A curve-fitting approach'. In: *Behavior Research Methods, Instruments, & Computers* 34 (2002), pp. 618–624.

[68] Robert Gabriel Lupu, Radu Gabriel Bozomitu, Alexandru Păsărică and Cristian Rotariu. 'Eye tracking user interface for Internet access used in assistive technology'. In: *2017 E-Health and Bioengineering Conference (EHB)*. IEEE. 2017, pp. 659–662.

[69] Päivi Majaranta and Andreas Bulling. 'Eye tracking and eye-based human–computer interaction'. In: *Advances in physiological computing*. Springer, 2014, pp. 39–65.

[70] Susana Martinez-Conde, Stephen L Macknik and David H Hubel. 'The role of fixational eye movements in visual perception'. In: *Nature reviews neuroscience* 5.3 (2004), pp. 229–240.

[71] George W McConkie and Lester C Loschky. 'Perception onset time during fixations in free viewing'. In: *Behavior Research Methods, Instruments, & Computers* 34.4 (2002), pp. 481–490.

[72] Maria Laura Mele and Stefano Federici. 'Gaze and eye-tracking solutions for psychological research'. In: *Cognitive processing* 13 (2012), pp. 261–265.

[73] Craig H Meyer, Adrian G Lasker and David A Robinson. 'The upper limit of human smooth pursuit velocity'. In: *Vision research* 25.4 (1985), pp. 561–563.

[74] Parag K Mital, Tim J Smith, Robin L Hill and John M Henderson. 'Clustering of gaze during dynamic scene viewing is predicted by motion'. In: *Cognitive computation* 3 (2011), pp. 5–24.

[75] Rizwan Ali Naqvi, Muhammad Arsalan, Ganbayar Batchuluun, Hyo Sik Yoon and Kang Ryoung Park. 'Deep learning-based gaze detection system for automobile drivers using a NIR camera sensor'. In: *Sensors* 18.2 (2018), p. 456.

[76] Rizwan Ali Naqvi, Muhammad Arsalan and Kang Ryoung Park. 'Fuzzy system-based target selection for a NIR camera-based gaze tracker'. In: *Sensors* 17.4 (2017), p. 862.

[77] Shivsevak Negi and Ritayan Mitra. 'Fixation duration and the learning process: An eye tracking study with subtitled videos'. In: *Journal of Eye Movement Research* 13.6 (2020), pp. 10–16910.

[78] Marcus Nyström and Kenneth Holmqvist. 'An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data'. In: *Behavior research methods* 42.1 (2010), pp. 188–204.

[79] Marcus Nyström, Ignace Hooge and Kenneth Holmqvist. 'Post-saccadic oscillations in eye movement data recorded with pupil-based eye trackers reflect motion of the pupil inside the iris'. In: *Vision research* 92 (2013), pp. 59–66.

[80] K Orzechowski. 'De l'ataxie dysmetrique des yeux. Remarques sur l'ataxie des yeux dite myoclonique (opsclonie, opsochorie)'. In: *J Psychol Neurol* 35 (1927), p. 37639.

[81] Cristina Palmero, Javier Selva, Mohammad Ali Bagheri and Sergio Escalera. 'Recurrent cnn for 3d gaze estimation using appearance and shape cues'. In: *arXiv preprint arXiv:1805.03064* (2018).

[82] Seonwook Park, Emre Aksan, Xucong Zhang and Otmar Hilliges. 'Towards End-to-End Video-based Eye-Tracking'. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2020.

[83] Pramodini A Punde, Mukti E Jadhav and Ramesh R Manza. 'A study of eye tracking technology and its applications'. In: *2017 1st International Conference on Intelligent Systems and Information Management (ICISIM)*. IEEE. 2017, pp. 86–90.

[84] Cyril Rashbass. 'The relationship between saccadic and smooth tracking eye movements'. In: *The Journal of physiology* 159.2 (1961), p. 326.

[85] Keith Rayner. 'Eye movements in reading and information processing: 20 years of research.' In: *Psychological bulletin* 124.3 (1998), p. 372.

[86] Keith Rayner. 'Eye movements in reading: Models and data'. In: *Journal of eye movement research* 2.5 (2009), p. 1.

[87] Erik D Reichle, Keith Rayner and Alexander Pollatsek. 'The EZ Reader model of eye-movement control in reading: Comparisons to other models'. In: *Behavioral and brain sciences* 26.4 (2003), pp. 445–476.

[88] Constantin A Rothkopf and Jeff B Pelz. 'Head movement estimation for wearable eye tracker'. In: *Proceedings of the 2004 symposium on eye tracking research & applications*. 2004, pp. 123–130.

[89] Michele Rucci and Jonathan D Victor. 'The unsteady eye: an information-processing stage, not a bug'. In: *Trends in neurosciences* 38.4 (2015), pp. 195–206.

[90] Dario D Salvucci and John R Anderson. 'Automated eye-movement protocol analysis'. In: *Human-computer interaction* 16.1 (2001), pp. 39–86.

[91] Dario D Salvucci and Joseph H Goldberg. 'Identifying fixations and saccades in eye-tracking protocols'. In: *Proceedings of the 2000 symposium on Eye tracking research & applications*. 2000, pp. 71–78.

[92] Dario D Salvucci and John R Anderson. 'Tracing eye movement protocols with cognitive process models'. In: (1998).

[93] Tayyar Sen and Ted Megaw. 'The effects of task variables and prolonged performance on saccadic eye movement parameters'. In: *Advances in Psychology*. Vol. 22. Elsevier, 1984, pp. 103–111.

[94] Frederick Shic, Brian Scassellati and Katarzyna Chawarska. 'The incomplete fixation measure'. In: *Proceedings of the 2008 symposium on Eye tracking research & applications*. 2008, pp. 111–114.

[95] Karen Simonyan and Andrew Zisserman. 'Very Deep Convolutional Networks for Large-Scale Image Recognition'. In: *arXiv preprint arXiv:1409.1556* (2014).

[96] Tim J Smith and Parag K Mital. 'Attentional synchrony and the influence of viewing task on gaze behavior in static and dynamic scenes'. In: *Journal of vision* 13.8 (2013), pp. 16–16.

[97] Lawrence Stark. 'Scanpaths revisited: Cognitive models, direct active looking'. In: *Eye movements: Cognition and visual perception* (1981), pp. 193–226.

[98] Mikhail Startsev, Ioannis Agtzidis and Michael Dorr. '1D CNN with BLSTM for automated classification of fixations, saccades, and smooth pursuits'. In: *Behavior Research Methods* 51 (2019), pp. 556–572.

[99] Mikhail Startsev and Raimondas Zemblys. 'Evaluating eye movement event detection: A review of the state of the art'. In: *Behavior Research Methods* (2022), pp. 1–62.

[100] Enkelejda Tafaj, Thomas C Kübler, Gjergji Kasneci, Wolfgang Rosenstiel and Martin Bogdan. 'Online classification of eye tracking data for automated analysis of traffic hazard perception'. In: *Artificial Neural Networks and Machine Learning–ICANN 2013: 23rd International Conference on Artificial Neural Networks Sofia, Bulgaria, September 10-13, 2013. Proceedings 23*. Springer. 2013, pp. 442–450.

[101] Benjamin W Tatler, Roland J Baddeley and Iain D Gilchrist. 'Visual correlates of fixation selection: Effects of scale and time'. In: *Vision research* 45.5 (2005), pp. 643–659.

[102] Kang Wang, Hui Su and Qiang Ji. 'Neuro-inspired Eye Tracking with Eye Movement Dynamics'. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 9831–9840.

[103] Michel Wedel and Rik Pieters. 'A review of eye-tracking research in marketing'. In: *Review of Marketing Research* (2017).

[104] Heino Widdel. 'Operational problems in analysing eye movements'. In: *Advances in psychology*. Vol. 22. Elsevier, 1984, pp. 21–29.

[105] EP Widmaier, H Raff and KT Strang. 'Cardiovascular physiology'. In: *Vander's Human Physiology the Mechanism of Body Function. 10th ed. McGraw-Hill* (2006), pp. 452–454.

[106] Harry J Wyatt and Jordan Pola. 'Smooth pursuit eye movements under open-loop and closed-loop conditions'. In: *Vision research* 23.10 (1983), pp. 1121–1131.

[107] Ying Yan, Huazhi Yuan, Xiaofei Wang, Ting Xu and Haoxue Liu. 'Study on driver's fixation variation at entrance and inside sections of tunnel on highway'. In: *Advances in Mechanical Engineering* 7.1 (2015), p. 273427.

[108] Dong Hyun Yoo and Myung Jin Chung. 'A novel non-intrusive eye gaze estimation using cross-ratio under large head motion'. In: *Computer Vision and Image Understanding* 98.1 (2005), pp. 25–51.

[109] Raimondas Zemblys, Niehorster Diederick and Kenneth Holmqvist. 'End-to-end eye-movement event detection using deep neural networks'. In: *Journal of Eye Movement Research: vol. 10, iss. 6: Abstracts of the 19th European Conference on Eye Movements, August 20-24, 2017, Wuppertal, Germany*. Vol. 10. 6. International Group for Eye Movement Research. 2017.

[110] Raimondas Zemblys, Diederick C Niehorster and Kenneth Holmqvist. ‘gazeNet: End-to-end eye-movement event detection with deep neural networks’. In: *Behavior research methods* 51 (2019), pp. 840–864.

[111] Raimondas Zemblys, Diederick C Niehorster, Oleg Komogortsev and Kenneth Holmqvist. ‘" Using machine learning to detect events in eye-tracking data": Correction.’ In: (2019).

[112] Raimondas Zemblys, Diederick C Niehorster, Oleg Komogortsev and Kenneth Holmqvist. ‘Using machine learning to detect events in eye-tracking data’. In: *Behavior research methods* 50 (2018), pp. 160–181.

List of Figures

List of Tables

List of Abbreviations

AI	Artificial Intelligence
CNN	Convolutional Neural Network
LSTM	Long Short-Term Memory
CNN-LSTM	Hybrid Convolutional Neural Network–Long Short-Term Memory model
RF	Random Forest
ML	Machine Learning
DL	Deep Learning
HCI	Human-Computer Interaction
PSO	Post-Saccadic Oscillation
SP	Smooth Pursuit
Fix	Fixation
Sac	Saccade
IVT	Identification by Velocity Threshold
IDT	Identification by Dispersion Threshold
VD	Velocity and Direction feature set
VAD	Velocity, Acceleration, and Direction feature set
VJD	Velocity, Jerk, and Direction feature set
VAJD	Velocity, Acceleration, Jerk, and Direction feature set
FSPso	Fixation-Saccade-Post Saccadic Oscillation classification scheme
FSSP	Fixation-Saccade-Smooth Pursuit classification scheme
Hz	Hertz
px	Pixel
ms	Millisecond
ROC	Receiver Operating Characteristic
AUC	Area Under the Curve
F1	F1-score
CNN-2D	Two-Dimensional Convolutional Neural Network

SP Masking	Procedure for excluding smooth pursuit events from classification
SR Research	Commercial eye-tracking analysis software
GT	Ground Truth