Faculty of Engineering and Materials Science
German University in Cairo

AI-Based Skill Builder for Digital Art

Holistic Handwriting Analysis for Dyslexia Risk: A 3-Stage AI Pipeline along with Eye-Tracking Cross-lingual application

Bachelor Thesis

by

Abdelrahman Emad Habashi Taha

Supervised by

Dr. Islam El-Maddah

May 2025

Declaration

This is to certify that:

(i) the thesis comprises only my original work toward the Bachelor of Science (B.Sc.) at the German University in Cairo (GUC),

(ii) due acknowledgement has been made in the text to all other material used

_____

Name

May 2025

Acknowledgement

I would first and foremost like to express my deepest gratitude to Dr. Islam El-Maddah, my bachelor thesis supervisor, for his invaluable support and guidance throughout this entire project. His profound knowledge, extensive expertise, and unwavering dedication to his students were instrumental to the progress and success of this research. Dr. El-Maddah consistently provided insightful advice and mentorship whenever I faced challenges, and his commitment was a constant source of motivation. I feel truly honored and fortunate to have had the opportunity to work under the supervision of such a dedicated and knowledgeable academic.

Furthermore, I extend my sincere appreciation to my sponsors and collaborators at the Enosh Science Center. A special thank you is owed to Professor Dr. Ibrahim Elnoshokaty for his generous support, insightful contributions, and for fostering an environment conducive to research and innovation. His belief in this project has been a significant encouragement. I am also deeply grateful to Engineer Radwa Taha for her consistent support, practical assistance, and valuable perspectives offered throughout the various stages of this work. The collaboration with the Enosh Science Center has been an enriching experience, and their contributions have been pivotal to the successful completion of this thesis.

This endeavor would not have been possible without the collective support and encouragement of all those mentioned.

Contents

List of Figures

List of Tables

Abstract

Dyslexia, a common neurodevelopmental disorder significantly impacting literacy acquisition, necessitates robust early screening methods. This thesis pioneers AI-driven approaches for identifying dyslexia risk indicators by comprehensively analyzing two key behavioral modalities: handwriting and eye-tracking. Handwriting analysis was advanced through foundational experiments on the Gambo isolated letter dataset, where fused features from ResNet50V2 and InceptionV3 CNNs achieved high discriminability (e.g., 0.9413 accuracy with RBF SVM for 3-class letter characteristics). These insights informed the development of a novel 3-Stage AI Pipeline for Dyslexia Risk Assessment from Handwriting. This pipeline uniquely integrates: (1) Global style atypicality assessment using fine-tuned CLIP embeddings on paragraph images, which demonstrated strong separation of dyslexic writing styles; (2) Contextual word and line-level analysis via OpenAI's GPT-4o, effectively identifying spelling errors and visual anomalies more prevalent in dyslexic samples; and (3) Fine-grained visual analysis of segmented letters using a classifier trained on the Gambo-derived features to pinpoint specific letter-form errors. Concurrently, an eye-tracking model was developed to assess dyslexia risk and its cross-lingual potential. While initial BiLSTM explorations provided a baseline, the primary pipeline utilized a Convolutional Autoencoder (AE) trained on gaze-path images from a Czech (Benfatto-derived) dataset. This research validates the potential of distinct, specialized AI pipelines for extracting nuanced behavioral markers of dyslexia from both handwriting and eye-tracking data. The proposed 3-Stage handwriting framework offers a structured, multi-level analytical approach, while the AE-based eye-tracking model shows promise for feature learning from gaze dynamics. Findings underscore the importance of domain-specific feature engineering and highlight challenges in dataset integration and cross-domain generalization, therefore shwowing critical directions for future development of comprehensive and accessible dyslexia screening technologies.

Chapter 1

Introduction

1.1 Motivation

Dyslexia, a common neurodevelopmental learning disorder, presents significant challenges to individuals in acquiring accurate and fluent reading and spelling skills. These difficulties can persist throughout life if not identified and supported early, impacting academic achievement, self-esteem, and emotional well-being [1], [2]. With global prevalence estimates for dyslexia in school-aged children ranging from 5-17%, varying by diagnostic criteria and language, the need for effective early screening is paramount [3]. Timely identification enables targeted interventions that can markedly improve literacy outcomes and mitigate long-term adverse effects [4].

Current dyslexia screening methods often involve resource-intensive behavioral assessments and standardized tests that primarily target phonological processing, rapid naming, and reading fluency [5]. While valuable, these methods can be time-consuming and may not always capture the full spectrum of difficulties, particularly subtle indicators manifested in everyday academic tasks like handwriting or during natural reading behaviors. Recognizing these limitations, there is a strong and growing interest in leveraging Artificial Intelligence (AI) and machine learning to develop more objective, scalable, and nuanced screening tools that can analyze diverse behavioral data [6].

This thesis is motivated by the potential of AI to revolutionize early dyslexia risk identification by deeply analyzing two distinct yet informative modalities: offline handwriting and eye-tracking patterns during reading. The core premise of this work is that by developing specialized AI pipelines for each modality, we can uncover subtle behavioral markers indicative of dyslexia risk. This research presents a novel, holistic multi-stage pipeline for comprehensive handwriting analysis. In parallel, to address the challenge posed by the limited availability of English-specific dyslexia detection eye-tracking (ET) models and labeled datasets, a key objective was to develop a robust ET model with strong potential for cross-lingual application, trained on a non-English labeled dataset (Czech) and subsequently applied to unlabeled English data to explore its transferability.

1.2 Problem Statement

Despite its prevalence, the early and accurate identification of dyslexia remains a significant challenge within educational and clinical settings [2]. Traditional screening methods, while foundational, present several practical and methodological limitations:

- Resource Demands: Conventional assessments often require specialized personnel and considerable time for individual administration, limiting their feasibility for widespread, proactive screening [5].
- Potential for Subjectivity: The interpretation of certain behavioral observations can vary, potentially leading to inconsistencies in screening outcomes.
- Limited Scope of Analyzed Behaviors: Many screening tools heavily emphasize phonological and direct reading tasks, potentially underutilizing rich information embedded in children's handwriting characteristics or their eye movement patterns during reading, which can also reflect underlying cognitive and processing differences [3].
- Dataset Scarcity and Specificity for AI Development: The advancement of robust AI-driven screening tools is often hindered by a scarcity of large-scale, well-annotated, publicly available datasets that specifically capture the nuances of child handwriting and eye movements in relation to dyslexia.

While AI-driven methods have shown promise in automating and enhancing aspects of dyslexia screening, many existing systems are often limited to a single data type, a narrow set of features, or are developed for specific linguistic contexts without readily available counterparts or labeled datasets in others, such as English for certain eye-tracking methodologies. There is a pressing need for more sophisticated AI systems capable of performing deep, contextual analysis within individual modalities to identify a broad range of potential dyslexia indicators, and for approaches that can begin to address cross-lingual applicability.

Therefore, the central problems this thesis aims to address are:

1. The development and evaluation of a novel, multi-stage AI pipeline for comprehensive handwriting analysis. This pipeline is designed to identify dyslexia risk indicators through the following 3-Stages: global handwriting style assessment, contextual word-level linguistic and visual error analysis, and fine-grained letter-level visual inspection.
2. The development and evaluation of an AI-driven eye-tracking analysis pipeline for dyslexia risk assessment. A key focus of this approach is to investigate its potential for cross-lingual application, specifically by training a model on available

non-English labeled eye-tracking data (Czech) and subsequently applying it to unlabeled English data, thereby exploring methodologies to address the current scarcity of dedicated English eye-tracking models and labeled datasets for dyslexia research.

The overarching goal is to explore effective modeling techniques for these two distinct pipelines, aiming to create tools that could contribute to more efficient, nuanced, and potentially more broadly applicable early dyslexia screening.

1.3 Objectives

The primary objectives of this thesis are:

1. Handwriting Analysis Pipeline Development:
    - To design and implement a 3-Stage AI pipeline for dyslexia risk assessment from handwriting, comprising:
        - Stage 1: Global handwriting style assessment using CLIP embeddings.
        - Stage 2: Contextual word-level analysis using advanced VLMs (e.g., GPT-4o) for HTR, spelling, and anomaly detection.
        - Stage 3: Fine-grained letter-level visual analysis using CNN-based classifiers (informed by foundational experiments on the Gambo dataset with models like ResNet50V2, InceptionV3, and explorations with CLIP/BEiT for letter feature extraction).
    - To evaluate the components of this 3-Stage pipeline, demonstrating their individual capabilities.
2. Eye-Tracking Model Development and Cross-Lingual Exploration:
    - To develop a computational pipeline for analyzing raw eye-tracking data, primarily focusing on an Autoencoder (AE) approach for feature extraction from gaze-path images generated from the Czech (Benfatto-derived) dataset.
    - To train and evaluate classifiers on these AE-derived features for predicting dyslexia risk within the Czech dataset.
    - To apply the Czech-trained eye-tracking pipeline to the English GazeBase dataset to qualitatively assess its behavior in a cross-lingual, cross-demographic context.
3. Overall Analysis and Discussion:
    - To analyze and discuss the findings from both the handwriting and eye-tracking pipelines, highlighting their strengths and limitations.

- To outline potential directions for future research in AI-driven dyslexia screening based on the insights gained.

1.4 Thesis Outline

This thesis is organized into the following chapters to systematically address the research objectives:

- Chapter 1 (Introduction): Outlines the motivation, problem statement, research objectives, and the overall structure of the thesis.
- Chapter 2 (Concepts Overview): This chapter is divided into two parts:
  - Background: Provides essential information on dyslexia, traditional detection methods, the rationale for computational analysis of handwriting and eye-tracking, and a conceptual overview of core machine learning models and techniques (CNNs, LSTMs, Autoencoders, VLMs, Transfer Learning) employed.
  - Literature Review: Presents a review of existing research in AI-based dyslexia detection using handwriting, eye-tracking, and related methodologies, identifying key contributions and gaps.
- Chapter 3 (Methodology): Details the experimental design and procedures. This includes descriptions of the datasets (Gambo, IAM, DyslexicH/N, Czech, GazeBase), preprocessing steps, the architecture and training of models for foundational letter-level feature learning, the design and component implementation of the 3-Stage handwriting pipeline, the development of the eye-tracking (BiLSTM and AE-based) models, and the evaluation metrics used.
- Chapter 4 (Results and Discussion): Presents the empirical results from all experimental phases, including the validation of handwriting letter features, outcomes from each stage of the 3-Stage handwriting pipeline, performance of the eye-tracking models (on Czech data and application to GazeBase), followed by a comprehensive discussion interpreting these findings, addressing limitations, and comparing with existing literature.
- Chapter 5 (Conclusion and Future Work): Summarizes the main contributions and achievements of the thesis, revisits the research objectives, and proposes directions for future research to further advance AI-driven dyslexia screening tools.

Chapter 2

Concepts Overview

2.1 Background

2.1.1 Dyslexia

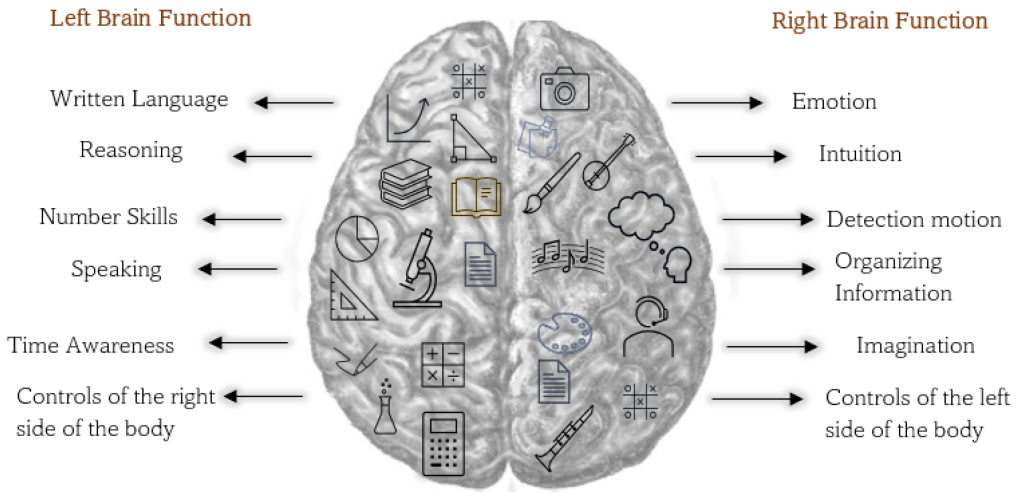


Figure 2.1: Illustration highlighting key areas of the left hemisphere of the brain involved in language processing, functions often affected in dyslexia. [7].

Dyslexia is a learning disorder that makes reading and writing difficult, even for individuals with normal intelligence and proper education. It is often linked to problems with phonological processing, which means struggling to connect letters to their respective sounds. However, the exact cause of dyslexia is still debated. Werth [8] challenges the common idea that phonological deficits are the root cause. Instead, he argues that dyslexia stems from difficulties in early visual processing, particularly in how the brain processes and recognizes letters. His research suggests that dyslexic individuals need more time to focus on words and have difficulty recognizing multiple letters at once, leading to reading errors like skipping or misreading letters.

From a perspective of brain development, Kuhl et al. [4] found that the differences seen in dyslexic readers actually appear before they even begin learning to read. Their study followed children for several years using magnetic resonance imaging (MRI), and

they discovered that children who later developed dyslexia had unusual brain development in the areas responsible for speech and reading. Specifically, these children had more folding in the left primary auditory cortex, which could indicate early processing problems. They also had weaker connections between key brain regions involved in processing sounds. Since these differences were present before reading instruction, this suggests that dyslexia is not just caused by lack of reading practice; it has biological roots from an early age.

From both perspectives, there are two key factors that are directly associated with dyslexia:[4], [8] visual processing challenges, and differences in brain development in speech and sound processing. Both perspectives support the idea that dyslexia is not simply a result of poor teaching or environmental factors but is instead linked to fundamental differences in brain function. As shown in Figure 2.1, the left hemisphere of the brain plays a crucial role in language processing, reasoning, and number skills. Disorders affecting this region, such as dyslexia, can lead to difficulties in reading, writing, and phonological processing, highlighting the importance of early identification and intervention to support individuals with dyslexia.

It is also important to distinguish dyslexia from, and note its frequent co-occurrence with, dysgraphia. While dyslexia primarily impacts reading decoding and fluent word recognition, often stemming from phonological processing deficits, and affects all aspects of written language including spelling, dysgraphia is a distinct specific learning disability primarily confined to difficulties in writing [9], [10]. Dysgraphia can manifest itself as challenges with the physical act of handwriting (e.g. poor letter formation, inconsistent spacing, slow production), spelling (which can arise from underlying issues different from dyslexia, such as orthographic coding or motor planning) and organizing thoughts for written expression [9]. Although these conditions are considered distinct and have different primary levels of difficulty, dyslexia is more rooted in language processing and dysgraphia in the ability to perform transcription and writing tasks their similar manifestation of cognitive deficits often obscures the distinction [10]. Consequently, overlapping symptoms such as poor spelling and slow and laborious writing can be present in the written output of children with either or both conditions. However, recent neurobiological evidence increasingly supports a distinction, showing differences between children with dyslexia and dysgraphia in aspects such as white matter integrity and functional brain connectivity [10]. This underscores the complexity of analyzing written samples for screening and the importance of considering indicators for both distinct profiles, even when symptoms appear similar [3].

2.1.2 Dyslexia Detection

Traditional Methods

Traditional methods for detecting dyslexia rely on behavioral assessments, standardized tests, and expert evaluations. These approaches focus on identifying difficulties in reading, writing, and phonological processing. One common method is phonological awareness testing, which evaluates an individual's ability to recognize and manipulate sounds in words [1]. Difficulties in phoneme segmentation, blending, and rhyming tasks are often indicators of dyslexia.

Another widely used approach is rapid automatized naming (RAN) tests, which measure how quickly individuals can name a sequence of familiar objects, letters, or colors [11]. Slower naming speeds have been linked to dyslexia, as they suggest underlying processing inefficiencies in visual and phonological pathways. Additionally, working memory assessments help identify deficits in holding and manipulating information, a common challenge for dyslexic individuals [12].

Educators and specialists also use reading fluency and comprehension tests to assess dyslexia risk. These tests analyze a child's ability to decode words, recognize patterns, and understand written text [2]. Writing samples are evaluated for spelling errors, letter reversals, and inconsistencies in handwriting. Observations of classroom behavior, such as difficulty following written instructions or avoiding reading tasks, further contribute to diagnosis [5].

While these traditional methods remain valuable, they require expert interpretation and can be time-consuming. Advances in technology are now complementing these assessments with automated tools, but expert-led evaluations continue to play a crucial role in early dyslexia detection.

Computational Handwriting Analysis

Handwriting, as a complex task involving linguistic, cognitive, and visual-motor skills, provides a rich source of information for understanding potential learning difficulties. While traditional methods involve subjective assessment of writing samples (as mentioned in section §2.1.2), computational handwriting analysis aims to automatically extract and quantify objective features from written text that may correlate with dyslexia and/or dysgraphia [3]. It's important to note that while many handwriting characteristics directly reflect graphomotor control challenges often associated with dysgraphia (e.g., poor letter formation, inconsistent sizing), certain patterns, particularly spelling errors and potentially aspects of spatial organization, can also provide clues relevant to dyslexia [3]. Computational methods offer the potential to analyze these indicators systematically and at scale.

Key aspects of handwriting targeted by computational analysis include:

- Spatial Organization and Layout: This involves analyzing how text is arranged on the page or screen. Techniques focus on quantifying:
  - Spacing: Measuring inter-letter spacing within words, inter-word spacing, and overall consistency. Atypical spacing, such as excessive crowding or large irregular gaps, can indicate difficulties in planning and execution [13].
  - Alignment: Assessing the placement of letters and words relative to a baseline (real or imaginary). Significant deviations or inconsistent alignment can suggest motor control or visual-perceptual challenges.
  - Margin Use: Analyzing the consistency and appropriateness of left and right margins, as atypical use might relate to spatial planning issues.
  - Letter Sizing: Measuring the height and width of letters and assessing their consistency. Highly variable letter sizes within a sample are often noted in dysgraphia [14].
- Letter Formation and Quality: Focuses on the individual characteristics of written characters:
  - Legibility and Consistency: Evaluating the clarity and uniformity of letter shapes. Poor formation, incomplete closure of letters (like 'a', 'o'), or high variability between instances of the same letter are common indicators [15].
  - Stroke Features: Analyzing the smoothness, curvature, and connectedness of strokes that form letters (often more feasible with online data but inferable offline).
  - Slant: Measuring and assessing the consistency of the angle of writing.
- Error Analysis (Content-Based): Requires recognizing the written text to identify specific error types:
  - Spelling Errors: Detecting misspellings is crucial. Further analysis can differentiate between phonetic errors (plausible based on sound, e.g., "fon" for "phone") and non-phonetic errors, which might be more indicative of orthographic memory issues seen in dyslexia or dysgraphia.
  - Reversals and Inversions: Identifying flipped letters (e.g., 'b'/'d', 'p'/'q') or inverted letters ('m'/'w', 'u'/'n'). While common in early writers, persistent reversals/inversions beyond typical developmental stages can sometimes be associated with dyslexia, although they are not considered a primary diagnostic criterion on their own [16].

  - Transpositions: Detecting letters written in the wrong order within a word (e.g., "lopt" for "plot").

- Temporal Dynamics (Primarily from Online Data): Analyzing the writing process over time:
  - Writing Speed: Measuring the velocity of the pen/stylus. Significant slowness or high variability can indicate effort or difficulty.
  - Pressure Modulation: Assessing the force applied during writing. Excessive or highly variable pressure can be linked to motor control issues.
  - Fluency: Analyzing pauses, hesitations, and the ratio of time the pen is on the surface versus lifted (in-air time). Flawed writing can reflect underlying processing load.
  - Fluency: Analyzing pauses, hesitations, and the ratio of time the pen is on the surface versus lifted (in-air time). Dysfluent writing can reflect underlying processing load.

Computational techniques employed to analyze these features include image processing and machine learning algorithms applied to scanned images (offline data) or sensor data streams from digital tablets (online data). Convolutional Neural Networks (CNNs) are particularly adept at learning complex visual patterns directly from offline handwriting images, capturing aspects of letter shape, spatial layout, and texture relevant to formation quality, sizing, and spacing [6], [7]. Analyzing the textual content for spelling and reversal errors typically requires accurate Handwriting Recognition (HWR) systems, which remain challenging for variable child handwriting but are essential for linking visual analysis to linguistic content. For online data, Recurrent Neural Networks (RNNs) like LSTMs can model the sequential dynamics of speed, pressure, and trajectory [13]. Ultimately, the goal is to use these computational methods to develop objective, reliable indicators from handwriting that can contribute to early screening for both dyslexia and dysgraphia risks.

Eye Tracking

Eye tracking has become an important tool for studying and detecting dyslexia. By monitoring eye movements during reading tasks, researchers can gather objective data on how individuals process text. Dyslexic readers often display distinct eye movement patterns compared to typical readers, commonly including longer fixation times (pausing longer on words), more frequent regressions (backward movements to re-read text), and generally more irregular or less fluid scanpaths [18]. These observable patterns provide valuable insights into the visual and cognitive challenges that contribute to

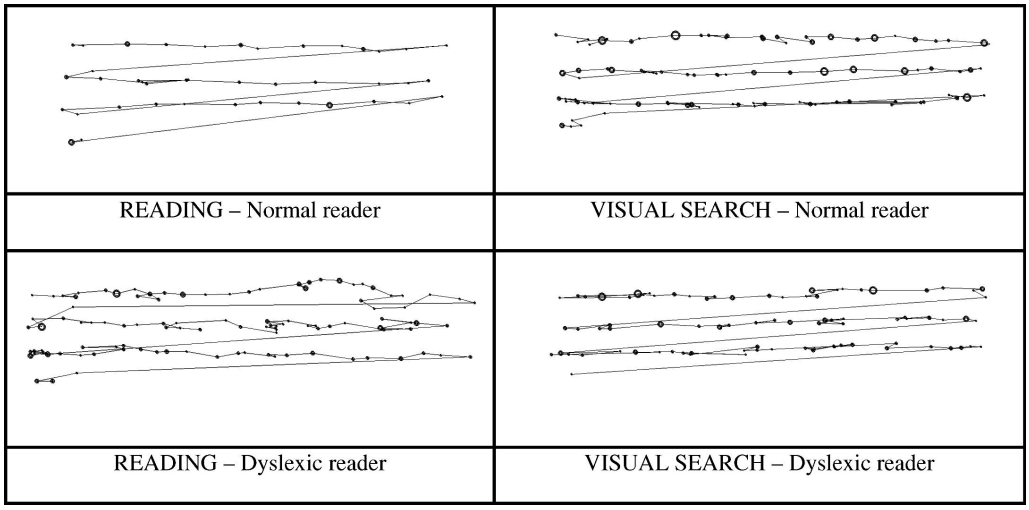


Figure 2.2: Comparison of eye movement scanpaths during a visual task, illustrating more erratic patterns with increased fixations and regressions often observed in individuals with dyslexia compared to typical readers. [17].

reading difficulties. Figure 2.2 illustrates the differences in eye movement patterns between dyslexic and non-dyslexic readers during both reading and visual search tasks. As shown in the figure, dyslexic individuals tend to exhibit more erratic eye movements, characterized by an increased number of fixation points and regressions, resulting in reading patterns that are less efficient compared to those of typical readers.

Several key eye-tracking metrics quantify these reading behaviors and are commonly analyzed in dyslexia research:

- Fixation Duration: The length of time the gaze remains paused on a specific location in the text. Individuals with dyslexia often exhibit longer fixation durations, potentially reflecting greater difficulty in word recognition or semantic processing [18].

- Number of Fixations: The total count of gaze pauses made while reading a specific passage. A higher number of fixations is often observed in dyslexic readers, suggesting a less fluent reading process requiring more stops.

- Saccade Amplitude: The distance covered by the rapid eye movements (saccades) between fixations. Dyslexic readers may exhibit shorter average saccade lengths, indicating less efficient forward progression through the text.

- Regressions: Backward saccades made to revisit previously read text. An increased frequency of regressions is a hallmark pattern in dyslexia, often linked to comprehension difficulties or problems with initial word decoding.

- Scan Path / Gaze Path: The overall sequence and spatial pattern of fixations and saccades across the text. Analysis of the scan path can reveal inefficient reading strategies; visually, paths often appear more irregular or scattered in dyslexia compared to the more linear paths of typical readers.

- Dwell Time: The total duration of all fixations made within a specific region of interest (e.g., a single word or line). Longer dwell times can pinpoint specific words or areas causing processing bottlenecks for the reader.

- Landing Position: The specific location within a word where the eye initially lands after a saccade. Variations in typical landing positions (often slightly left of center in skilled readers) might reflect different word approach or recognition strategies in dyslexia.

- Binocular Control (Vergence & Disconjugacy): Aspects related to how the two eyes coordinate during reading. Vergences are movements to maintain single binocular vision, while Disconjugacy refers to the degree of misalignment between the movements of the two eyes. Some studies suggest potential difficulties in binocular coordination for individuals with dyslexia, implicating visual processing mechanisms [19].

Recent advances in eye tracking technology allow for increasingly precise measurement of these diverse behaviors. Computational analysis techniques, often employing machine learning, are being applied to these detailed eye movement data. For instance, studies have shown that specific features derived from eye tracking, such as fixation stability and latency, can be used effectively within machine learning models to distinguish between dyslexic and non-dyslexic readers [20]. Research has also confirmed that certain eye movement metrics show strong correlations with standard reading assessments and can serve as accurate predictors for dyslexia risk. This non-invasive approach, combining objective eye movement data with computational analysis, not only aids in improving early diagnosis but also holds potential for developing more targeted interventions tailored to the unique reading patterns observed in individuals with dyslexia.

2.1.3 Overview of Deep learning Models Utilized

This thesis leverages several machine learning models to analyze handwriting and eye-tracking data for dyslexia risk prediction. This subsection provides a conceptual overview of the key architectures and techniques employed, explaining their relevance to the task. Specific implementation details are reserved for Chapter 3.

Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are a class of deep learning models particularly effective for processing grid-like data, most notably images [21]. They work by applying convolutional filters (kernels) across input data to automatically learn hierarchical patterns. Early layers typically detect simple features like edges and textures, while deeper layers combine these to recognize more complex structures, such as shapes or objects. Key components include:

- Convolutional Layers: Apply learnable filters to extract spatial features.
- Activation Functions (e.g., ReLU): Introduce non-linearity, allowing the network to learn complex relationships.
- Pooling Layers (e.g., MaxPooling, AveragePooling): Reduce the spatial dimensions, making the learned features more robust to variations in position and scale, and reducing computational load.

Relevance: CNNs are highly relevant for analyzing the offline handwriting images in this work. They can automatically learn visual features indicative of letter formation, stroke quality, spatial arrangement, and other characteristics directly from the pixel data, without requiring manual feature engineering. These learned visual patterns can then be used to identify potential signs associated with dyslexia or dysgraphia manifested in writing.

Specific CNN Architectures: ResNet and Inception

While general CNN principles apply, specific architectures offer distinct advantages:

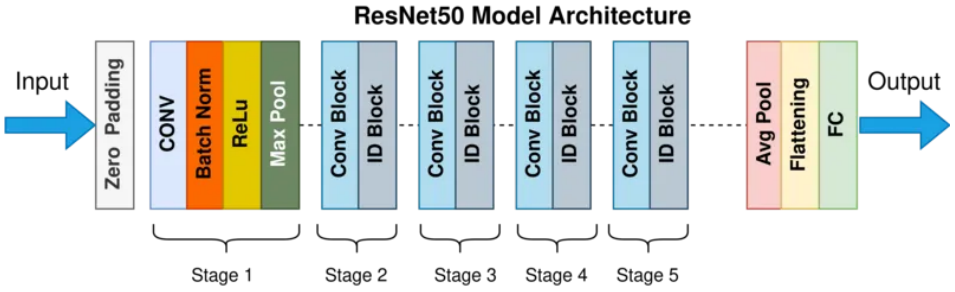


Figure 2.3: Example diagram of the ResNet50 architecture, illustrating the concept of residual blocks with skip connections. [22].

ResNet (Residual Networks): Introduced by He et al. [23], ResNet architectures tackled the challenge of training very deep networks. Their key innovation is the "residual block," which incorporates skip connections. These connections allow the gradient

signal to bypass layers during backpropagation, mitigating the vanishing gradient problem and enabling the effective training of networks with tens or even hundreds of layers. This depth allows for learning highly complex feature representations. ResNet50V2, used in this thesis, is a popular variant with 50 layers, incorporating improvements to the original residual block design. Relevance: The depth enabled by ResNet allows for potentially capturing very intricate visual patterns in handwriting samples.

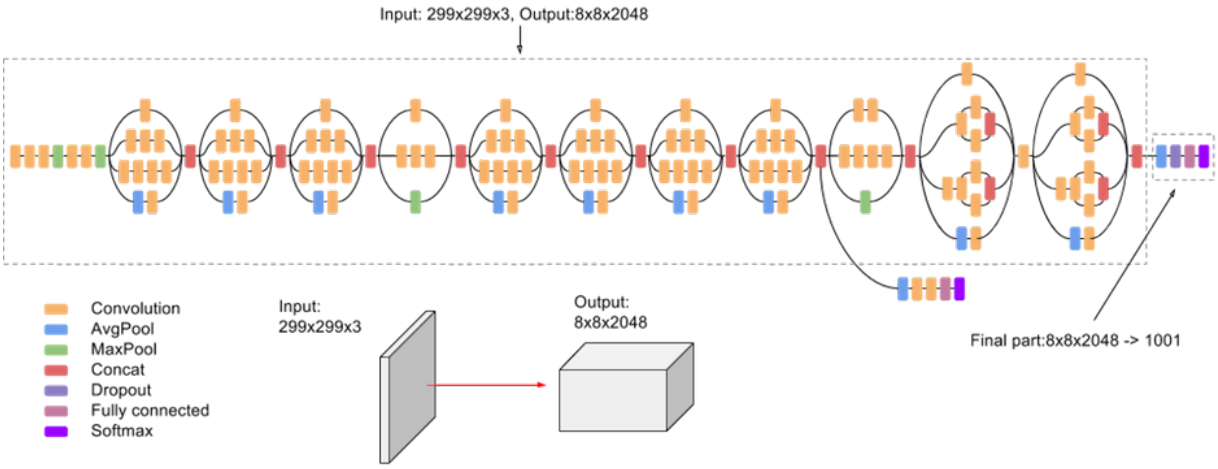


Figure 2.4: Example overview of the InceptionV3 architecture, highlighting its use of inception modules with parallel convolutional filters of different sizes. [24].

Inception V3: As described by Szegedy et al. [25], Inception V3 is part of the GoogLeNet family, focusing on achieving high accuracy with computational efficiency. Its core idea is the "inception module," which applies multiple convolutional filters of different sizes (e.g., 1x1, 3x3, 5x5) in parallel within the same layer and concatenates their outputs. This allows the network to capture features at various scales simultaneously. Inception V3 incorporates optimizations like factorizing larger convolutions into smaller ones (e.g., replacing 5x5 with two 3x3 layers) and using asymmetric convolutions (e.g., 1x7 and 7x1) to further reduce computational cost while maintaining representational power. It also utilizes techniques like auxiliary classifiers for regularization during training. Relevance: Inception V3 offers a balance between performance and efficiency, potentially capturing multi-scale visual features in handwriting effectively. Its use alongside ResNet provides complementary feature extraction capabilities.

Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM)

Recurrent Neural Networks (RNNs) are designed for processing sequential data, where the order of information matters [26]. Unlike feedforward networks, RNNs have connections that loop back, creating an internal state or "memory" that allows them

to persist information from previous steps in the sequence to influence the processing of current steps. However, simple RNNs struggle to capture long-range dependencies due to the vanishing/exploding gradient problem.

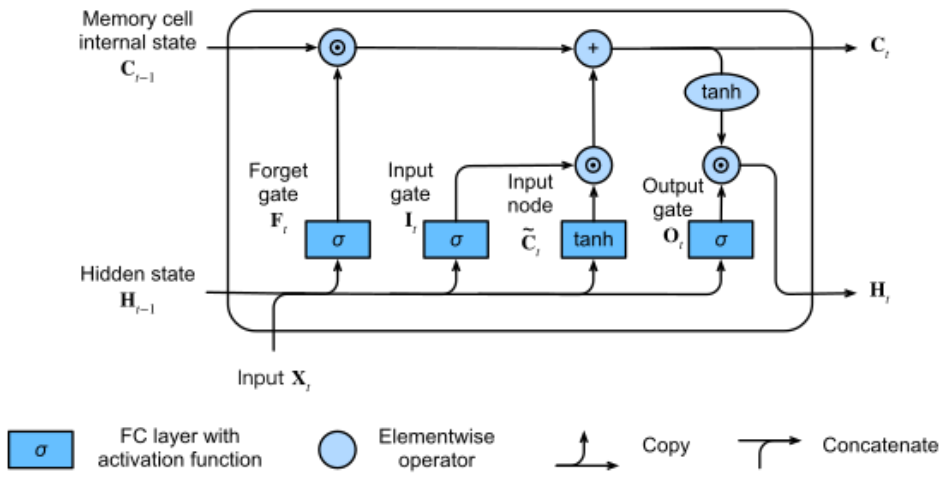


Figure 2.5: Conceptual diagram of the gates within a Long Short-Term Memory (LSTM) cell, including the forget gate, input gate, and output gate, which regulate information flow. [27].

Long Short-Term Memory (LSTM): Proposed by Hochreiter and Schmidhuber [28], LSTMs are a specialized type of RNN architecture explicitly designed to overcome the limitations of simple RNNs and learn long-term dependencies. They achieve this through a more complex repeating module containing "gates" (input, forget, output gates). These gates are neural networks themselves that regulate the flow of information, allowing the LSTM cell to selectively add, remove, or output information from its internal cell state over long time lags. Relevance: LSTMs are ideal for analyzing the sequential eye-tracking data (gaze coordinates over time) in this thesis. Reading is an inherently sequential process, and LSTMs can model the temporal dynamics of eye movements, potentially capturing patterns like fixation durations, saccade sequences, and regressions that are indicative of reading processes and difficulties associated with dyslexia.

Bidirectional LSTM (Bi-LSTM): A Bi-LSTM enhances the standard LSTM by processing the sequence in both forward and backward directions using two separate hidden layers [29]. The outputs from both directions are typically concatenated at each time step. Relevance: For analyzing reading scanpaths, knowing both the preceding (past) and succeeding (future) eye movements can provide richer contextual information for interpreting the gaze behavior at any given point. Bi-LSTMs allow the model to lever-

age this bidirectional context, potentially leading to a more robust understanding of the sequential patterns in eye-tracking data relevant to dyslexia.

Autoencoders

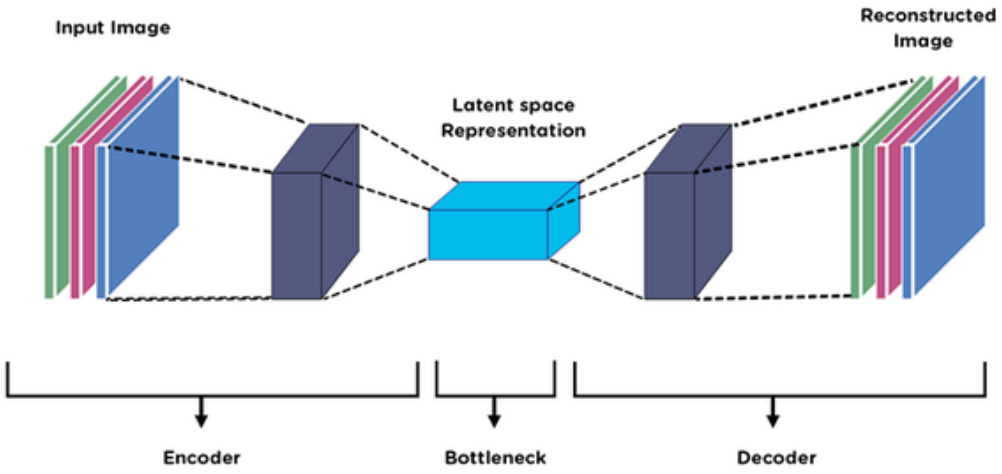


Figure 2.6: Diagram illustrating the general architecture of an Autoencoder, showing the encoder, bottleneck (latent representation), and decoder components.[30].

Autoencoders are a type of unsupervised artificial neural network primarily used for learning efficient data representations (encoding), often for dimensionality reduction or feature learning [31]. Unlike supervised models that learn to predict a target label, autoencoders learn to reconstruct their own input. They typically consist of two main parts connected by a bottleneck layer:

- Encoder: Compresses the input data into a lower-dimensional latent representation (the code or bottleneck).
- Bottleneck: The layer containing the compressed representation, capturing the most salient features of the input data.
- Decoder: Attempts to reconstruct the original input data from the compressed latent representation generated by the encoder.

The network is trained by minimizing a "reconstruction loss" function, which measures the difference between the original input and the reconstructed output. By forcing the data through the bottleneck, the autoencoder learns a compressed representation that captures the essential structures or variations within the data distribution. Different variants exist, such as Denoising Autoencoders (trained to reconstruct clean data from

corrupted input) or Variational Autoencoders (which learn a probabilistic distribution in the latent space).

Vision-Language Models (VLMs)

Vision-Language Models (VLMs) represent a class of models designed to understand and connect information from both visual (image/video) and textual modalities. Unlike traditional vision models that operate solely on pixels or language models that process only text, VLMs aim to learn joint representations that capture the relationship between visual content and its corresponding natural language description. This enables tasks like image captioning, visual question answering (VQA), text-based image retrieval, and zero-shot image classification by bridging the gap between perception and language understanding.

VLMs can be broadly categorized based on their architecture and pre-training strategies, reflecting different ways of integrating and learning from the two modalities:

- Two-Stream Architectures: These models typically employ separate encoders for the visual and textual inputs (e.g., a CNN or Vision Transformer for images, a text Transformer for language). The representations from these separate streams are then aligned or fused at later stages, often using techniques like cross-attention mechanisms or contrastive learning objectives that push corresponding image-text pairs closer in an embedding space while separating non-matching pairs.

- Single-Stream (Fusion) Architectures: In this approach, visual and textual inputs (or their initial embeddings) are often concatenated or combined early on and fed into a single, unified Transformer network. This network processes the combined multimodal sequence jointly, allowing for deep interaction between visual and linguistic features throughout the model layers. These models often adapt masked modeling pre-training objectives (like Masked Language Modeling or Masked Region Modeling) to the multimodal context.

- Encoder-Decoder Architectures: Commonly used for generative tasks like image captioning or text-to-image synthesis. An encoder network processes the input modality (e.g., an image) to produce a representation, which is then fed to a decoder network (often a language model) that generates the output in the target modality (e.g., a textual caption). Attention mechanisms are crucial for linking the generated output back to relevant parts of the input.

These architectural choices and training paradigms allow VLMs to learn powerful joint embeddings. Two particularly influential models leveraging some of these principles are CLIP and BEiT.

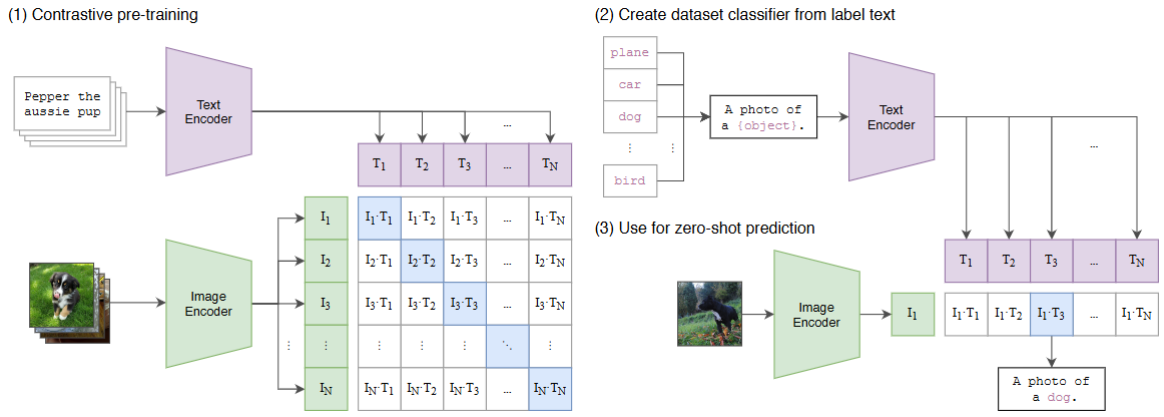


Figure 2.7: Overview of the CLIP (Contrastive Language-Image Pre-training) architecture, illustrating the separate image and text encoders that learn to map corresponding pairs into a shared embedding space. [32].

CLIP (Contrastive Language–Image Pre-training): Developed by Radford et al. at OpenAI [32], CLIP exemplifies a two-stream approach trained with a contrastive objective. It learns visual concepts directly from natural language supervision found abundantly on the internet. It consists of two encoders: one for images (e.g., a Vision Transformer or ResNet) and one for text (e.g., a Transformer). During pre-training on vast datasets of image-text pairs, CLIP learns to map images and their corresponding textual descriptions into a shared embedding space. Its training objective is contrastive: it aims to maximize the similarity (e.g., cosine similarity) between the embeddings of correct image-text pairs while minimizing the similarity for incorrect pairs within a batch. A key strength of CLIP is its remarkable zero-shot transfer capability. By framing classification tasks as matching an image embedding to the embedding of descriptive text prompts (e.g., "a photo of a handwritten letter 'a'"), CLIP can classify images into categories it wasn't explicitly trained on. Relevance: CLIP provides a powerful way to link visual features to semantic meaning derived from natural language, offering potential for interpreting or classifying visual inputs (like handwriting images) based on textual descriptions or learned concepts, potentially bypassing the need for extensive task-specific labeled data.

BEiT (Bidirectional Encoder representation from Image Transformers): Proposed by Bao et al. [33], BEiT adapts the principles of masked modeling, typically associated with single-stream language or multimodal models, specifically for pre-training Vision Transformers (ViTs) in a self-supervised manner. The core idea is Masked Image Modeling (MIM). An input image is divided into patches, and a certain percentage of these patches are randomly masked (e.g., replaced with a special [MASK] token). The model

is then trained to predict the discrete "visual tokens" corresponding to the original masked patches, based on the context provided by the unmasked patches. These visual tokens are typically obtained from the vocabulary of a pre-trained discrete Variational Autoencoder (dVAE). By performing this self-supervised pre-training task, BEiT learns rich, contextualized visual representations directly from pixel data without requiring manual labels. The pre-trained BEiT model can then be fine-tuned effectively for various downstream vision tasks like image classification or segmentation. Relevance: BEiT represents a powerful self-supervised method for learning robust visual features from images alone. A BEiT-pre-trained Vision Transformer can serve as a strong backbone for image analysis tasks, potentially capturing nuanced patterns in handwriting images effectively, even before fine-tuning on a specific classification objective.

Transfer Learning

Transfer learning is a machine learning technique where a model developed for a primary task is reused as the starting point for a model on a secondary, related task [34]. Typically, models pre-trained on large, general datasets (like ImageNet for image recognition or large text/image corpora for VLMs) have already learned powerful, hierarchical feature representations. By using these pre-trained weights as initialization and potentially fine-tuning them on the target task's smaller dataset, transfer learning can significantly reduce training time, improve performance (especially with limited target data), and enhance generalization. Relevance: This thesis utilizes transfer learning by employing ResNet50V2 and InceptionV3 models pre-trained on ImageNet as feature extractors for the handwriting analysis task, leveraging their learned visual representations for the specific domain of letter classification. Similarly, pre-trained VLM components could potentially be adapted for tasks involving visual understanding of handwriting linked to semantic concepts.

2.2 Literature Review

This chapter reviews existing research pertinent to the detection and screening of dyslexia, with a particular focus on computational approaches leveraging handwriting analysis and eye-tracking data. The review aims to situate the current work within the broader landscape, identify established techniques, and highlight existing challenges and opportunities that motivate the methodologies employed in this thesis. We explore studies utilizing various machine learning and deep learning models for analyzing these behavioral modalities, as well as common strategies for data handling and model validation.

2.2.1 Computational Analysis of Handwriting for Dyslexia and Dysgraphia

Handwriting provides a rich, non-invasive source of data reflecting a child's fine motor skills, visual-spatial organization, and linguistic encoding abilities. Computational analysis of handwriting has emerged as a promising avenue for identifying indicators associated with dyslexia and dysgraphia.

Deep Learning for Visual Handwriting Feature Extraction

Convolutional Neural Networks (CNNs) have been widely adopted for extracting visual features from handwritten text due to their ability to learn hierarchical patterns directly from image data. Several studies have demonstrated the efficacy of CNNs in classifying handwriting impairments or characteristics relevant to dyslexia. For instance, Seman et al. [14] developed a CNN to classify different error types in Chinese characters written by preschool children, achieving high accuracy in identifying structural and radical errors potentially linked to dyslexia. Similarly, Rosli et al. [35] modified the LeNet-5 architecture, incorporating techniques like max pooling and batch normalization, to classify English handwriting samples from dyslexic students into 'Normal', 'Reversal', and 'Corrected' categories, reporting improved performance over the baseline LeNet-5. The use of transfer learning with pre-trained CNNs, such as MobileNetV3, has also been explored by Mahmoud et al. [15] to achieve high accuracy in classifying letter characteristics, further emphasizing model interpretability through techniques like Grad-CAM to highlight features indicative of dyslexia. Aldehim et al. [6] provided a benchmarking study comparing various CNN architectures for dyslexia detection from handwriting, offering insights into performance trade-offs.

Beyond standard CNNs, hybrid models and more advanced architectures are being investigated. Patil et al. [13] combined CNNs for spatial feature extraction with Bidirectional LSTMs (Bi-LSTMs) to capture sequential dependencies in English handwriting, achieving notable accuracy on a custom dataset. Liu et al. [36] introduced DysDiTect, a CNN-Positional-LSTM-Attention model for Chinese handwriting, demonstrating improved feature extraction and cross-linguistic potential by incorporating attention mechanisms. Furthermore, Transformer-based architectures, like the Swin Transformer, are emerging for handwriting analysis. A study by Alkhurayyif and Sait [37] introduced a Swin Transformer combined with Continuous-Time Networks (CTN) for multi-modal handwriting analysis, showing superior performance in capturing fine-grained inconsistencies relevant to dyslexia.

Hybrid AI Approaches and Classical Machine Learning

While deep learning models directly learn features, some research combines deep feature extraction with classical machine learning classifiers. Alqahtani et al. [7] proposed such a hybrid approach, using CNN-extracted features as input to Support Vector Machines (SVM) and Random Forests (RF) for dyslexia detection from handwriting images, demonstrating improved accuracy and reduced false positives. This highlights a strategy where the robust feature learning of deep networks is complemented by the decision-making capabilities of established classifiers.

Challenges in Handwriting Analysis for Dyslexia

Despite promising results, computational handwriting analysis for dyslexia faces challenges. These include the high variability in children's handwriting, the scarcity of large, well-annotated, and publicly available datasets specifically labeled for dyslexia characteristics [3], and the need for models that are not only accurate but also interpretable for educators and clinicians. The distinction between features indicative of dysgraphia versus dyslexia (or their co-occurrence) in handwriting also requires careful consideration.

2.2.2 Eye-Tracking Analysis for Dyslexia Detection

Eye-tracking offers a non-invasive window into the cognitive processes underlying reading. Characteristic differences in eye movement patterns between dyslexic and typical readers (e.g., longer fixations, more regressions) are well-documented in section (§2.1.2 Background). Computational methods aim to leverage these patterns for automated detection.

Sequence Modeling of Gaze Data

Recurrent Neural Networks (RNNs), particularly LSTMs and Bi-LSTMs, are naturally suited for analyzing the sequential nature of gaze data. Haller et al. [38] successfully employed Bi-LSTM and CNN models on word-level eye-tracking features from Mandarin Chinese children, achieving high subject-level classification AUC for dyslexia without needing additional linguistic features. Gomolka et al. [39] utilized an LSTM on spatio-temporal attention trajectories captured during a visual memory task (Benton Visual Retention Test), reporting very high accuracy in classifying dyslexia in young children, even before full reading proficiency. This highlights the potential of LSTMs to model dynamic gaze behavior beyond traditional reading tasks.

Image-Based and Feature-Engineered Eye-Tracking Analysis

Alternative approaches involve transforming raw gaze data or extracting specific event-based features. Vajs et al. [40] proposed converting raw eye movements into grayscale gaze-path images and training a Convolutional Autoencoder (CNN AE) to extract features based on reconstruction error. Their method demonstrated good cross-linguistic generalizability between Serbian and Swedish datasets using standard machine learning classifiers on these AE-derived features. Zhang et al. [41] took a multimodal approach integrating Large Language Models (LLMs) for text relevance with EEG and eye-tracking (fixation counts, gaze duration) to classify word-level neural states during reading, using classical ML models like Linear SVM for classification.

Challenges in Eye-Tracking Analysis for Dyslexia

Key challenges in eye-tracking based dyslexia detection include study variability (differences in languages, tasks, hardware, and participant age), the need for robust event detection algorithms (for fixation/saccade parsing if used), and ensuring that models generalize well across diverse populations and settings [40]. The interpretability of features learned by complex sequence models also remains an area of ongoing research.

2.2.3 Advanced Methodological Considerations in AI for Dyslexia Screening

Beyond specific model architectures for handwriting or eye-tracking, several overarching methodological aspects are crucial in the development of effective AI tools for dyslexia screening.

Data Augmentation and Optimization Strategies

Given the common issue of limited dataset sizes in medical and educational domains, data augmentation techniques are vital. For handwriting analysis, studies frequently employ transformations like rotation, scaling, and noise injection to enhance dataset variability and reduce overfitting [6], [7]. Regularization techniques such as dropout and batch normalization are standard practice for stabilizing training [13], [36]. The choice of advanced optimizers (e.g., Adam, RMSProp) [14] and the application of transfer learning using pre-trained models [35] are also common strategies to improve model performance and generalization, particularly with limited labeled data.

Validation Methods, Generalization, and Dataset Bias

Robust validation is critical to assess the true utility of AI models. Cross-validation (e.g., 5- or 10-fold) is a widely used technique to ensure model robustness against par-

ticular data splits [6], [36]. However, a significant gap in the literature is the limited use of external validation on independent datasets, which is crucial for testing true generalization capabilities [37]. Addressing dataset bias, such as class imbalance between dyslexic and non-dyslexic samples, often involves techniques like oversampling minority classes or using weighted loss functions during training [7].

Explainable AI (XAI) and Cognitive AI

As AI models become more complex, their "black-box" nature can hinder adoption in clinical and educational settings. Explainable AI (XAI) methods, such as SHAP, LIME, or Grad-CAM, are increasingly being employed to provide insights into which features contribute most to a model's decision [15], [42]. Mahmoud et al. [15], for example, used Grad-CAM with their CNN model to highlight specific letter formations and stroke patterns associated with dyslexia, aiding interpretability. The concept of Cognitive AI aims to build more comprehensive screening tools by integrating multiple data sources (e.g., eye-tracking, handwriting, keystroke dynamics, speech) to analyze various cognitive load indicators and behavioral patterns, offering a more holistic assessment [14], [15].

2.2.4 Summary of Literature and Research Gaps

The reviewed literature demonstrates significant progress in applying AI techniques, particularly deep learning, to analyze handwriting and eye-tracking data for indicators of dyslexia and related learning difficulties. CNNs and hybrid models show promise for handwriting analysis, while LSTMs and AE-based approaches are effective for eye-tracking. However, several gaps and opportunities persist:

- Multimodal Integration for Handwriting Itself: While some studies combine CNNs with LSTMs for handwriting, a holistic pipeline integrating global style, word-level contextual/linguistic analysis, and fine-grained letter-level visual analysis for dyslexia risk from handwriting is less explored.
- Cross-Lingual Robustness of Eye-Tracking Models: While some work like Vajs et al. [40] has focused on cross-lingual generalization for eye-tracking, more research is needed on robust feature representations that transcend linguistic and demographic differences.
- Need for Interpretable, Actionable Systems: Beyond accuracy, there's a continued need for AI systems that provide explainable insights that educators and clinicians can use to inform interventions.

- Standardized Datasets and Benchmarks: The field would benefit from more large-scale, diverse, and publicly available datasets specifically curated for dyslexia research across different modalities to allow for more direct comparison of methods.

This thesis aims to contribute by developing and evaluating distinct AI-driven pipelines for both handwriting (a novel 3-Layer approach) and eye-tracking (an AE-based cross-lingual model), focusing on extracting meaningful features and assessing their utility for dyslexia risk identification.

Table 2.1: Comparison of Recent Studies in AI-Driven Dyslexia/Dysgraphia Analysis

Study	Primary Modality	AI Technique / Model Focus	Dataset Context & Task	Key Contribution / Relevance
Seman et al. [14]	Handwriting (Offline, Chinese Chars)	CNN (Error Type Classification)	Preschool Children; 4-class error ID (Normal, RE, SE, RSE). (Acc: 96.2%)	CNN for classifying specific handwriting impairments in young children, potential for early symptom ID.
Mahmoud et al. [15]	Handwriting (Offline, Mixed sources)	CNN (MobileNetV3, Transfer Learning), XAI (Grad-CAM)	Child/NIST Images; 3-class letter characteristics (Normal, Reversed, Corrected). (Acc: 99.65%)	High-accuracy letter classification with focus on interpretability for dyslexia screening.
Patil et al. [13]	Handwriting (Offline, English)	CNN-BiLSTM (Hybrid: Spatial + Sequential)	Child HW (6-12yrs), Word/Sentence/-Para tasks; Dyslexia classification. (Acc: 95.6%)	Hybrid deep learning captures both visual and sequential aspects of handwriting for dyslexia detection.

Continued on next page

Table 2.1 Continued from previous page

Study	Primary Modality	AI Technique / Model Focus	Dataset Context & Task	Key Contribution / Relevance
Rosli et al. [35]	Handwriting (Offline, Mixed sources)	CNN (Modified LeNet-5, Transfer Learning)	Child/NIST Images; 3-class letter characteristics. (Acc: 95.34%)	Modifications to standard CNN (LeNet-5) and transfer learning improve letter classification for dyslexia-related features.
Aldehim et al. [6]	Handwriting (Offline)	CNN Benchmarking (ResNet, VGG, etc.)	Aggregated datasets; Dyslexia detection. (Avg Acc: ∼90%)	Comparative analysis of different CNNs for dyslexia detection from handwriting, informing model selection.
Liu et al. [36]	Handwriting (Offline, Chinese Dictation)	CNN-Positional-LSTM-Attention	Child HW (Chinese); Dyslexia identification. (Acc: ∼94%)	Advanced hybrid model with attention for improved feature extraction in Chinese handwriting; cross-linguistic potential.
Alqahtani et al. [7]	Handwriting (Offline, Images)	Hybrid AI (CNN features + SVM/RF)	Image dataset; Dyslexia detection. (Acc: ∼95%)	Combining deep features with classical classifiers improves dyslexia identification accuracy.
SWIN Transformer [37]	Handwriting (Multi-modal - likely online+offline)	Swin Transformer + CTN	Multi-modal HW dataset; Dyslexia classification. (Acc: ∼96%)	Hierarchical spatial (Swin) + temporal (CTN) modeling for fine-grained handwriting inconsistency detection.
Haller et al. [38]	Eye-Tracking (Reading Task)	Bi-LSTM, CNN (Sequence Modeling)	Mandarin Chinese Children; Dyslexia Classification. (AUC: 0.93)	Neural sequence models effectively classify dyslexia from raw gaze patterns without needing explicit linguistic features.

Continued on next page

Table 2.1 Continued from previous page

Study	Primary Modality	AI Technique / Model Focus	Dataset Context & Task	Key Contribution / Relevance
Vajs et al. [40]	Eye-Tracking (Raw Gaze)	CNN Autoencoder (Gaze-Path Image Encoding) + ML Classifiers	Serbian & Swedish Children; Dyslexia Classification. (Acc: 82.9-85.6%)	Cross-linguistic dyslexia detection using AE-derived features from gaze-path images, avoids traditional event parsing.
Gomolka et al. [39]	Eye-Tracking (Visual Memory Task - BVRT)	LSTM (Spatio-temporal trajectories)	Early School-Aged Children; Dyslexia Diagnosis. (Acc: 97.7%)	High-accuracy dyslexia detection from ET during a non-reading task, enabling pre-literacy screening.

Chapter 3

Methodology

3.1 Methodology Overview

The methodological approach of this thesis is multifaceted, addressing dyslexia risk prediction through two primary modalities: handwriting analysis and eye-tracking. The development process involved distinct experimental phases and culminated in two proposed systems.

For handwriting analysis, the initial efforts were guided by existing research, focusing on robust feature learning from isolated handwritten letters using the Gambo dataset. This involved:

1. Preprocessing the Gambo letter images.
2. Training and fine-tuning several established Convolutional Neural Network (CNN) backbones (e.g., ResNet50V2, InceptionV3, MobileNetV2, DenseNet121) and exploring advanced Vision-Language Models (CLIP, BEiT) as feature extractors for letter characteristics (Normal, Reversal, Corrected).
3. Extracting features from these models, with a particular focus on best performing models and fusing features from them to create a rich 4096-dimensional representation.
4. Validating these individual and fused features by training classical machine learning classifiers on the 3-class Gambo letter task.

Recognizing the limitations of analyzing isolated letters for a complex condition like dyslexia, and aiming for a more holistic assessment of handwriting, this foundational work informed the development of a novel 3-Stage Pipeline for Dyslexia Risk Assessment from Handwriting (Figure 3.1). This pipeline, detailed in section §3.4, integrates:

- Stage 1: Global handwriting style assessment using CLIP on paragraph/page images.
- Stage 2: Contextual word-level analysis, leveraging VLM capabilities (explored with OpenAI GPT-4o) for transcription, spelling, and anomaly detection from line images.

- Stage 3: Fine-grained visual analysis of individual letters using a robust letter classifier component informed by the Gambo dataset experiments.

For eye-tracking analysis, the research aimed to develop a model capable of assessing dyslexia risk and exploring its cross-lingual applicability. This involved:

1. Preprocessing raw gaze data from the labeled Czech (Benfatto-derived) dataset and the unlabeled English GazeBase dataset.

2. An initial exploratory phase using Bidirectional Long Short-Term Memory (BiLSTM) networks for sequence modeling, including an attempt at self-supervised pre-training on GazeBase.

3. The development of the primary eye-tracking model based on converting gaze-path segments into images, training a Convolutional Autoencoder (AE) on these images (using Czech data), extracting reconstruction error-based features, and training classical classifiers for dyslexia risk on the Czech data.

4. Application of this Czech-trained AE-based pipeline to the English GazeBase data to assess its cross-lingual performance.

The chapter will first describe the datasets and experimental setup in section (§3.2), then detail the foundational handwriting feature learning experiments on the Gambo dataset in section (§3.3). This is followed by the comprehensive description of the proposed 3-Stage handwriting pipeline in (§3.4) and the eye-tracking cross-lingual model in (§3.5). Finally, the evaluation metrics used across these experiments are outlined in (§3.6).

3.2 Datasets

This study utilized several distinct datasets for the handwriting and eye-tracking analysis pipelines. Each dataset served a specific purpose, from initial feature learning to model training and cross-lingual application.

3.2.1 Handwriting Datasets

Isolated Handwritten Letter Dataset (*Gambo*)

Content: This dataset comprises images of isolated handwritten English letters (A–Z), with approximately 151,000 images for training and 57,000 for testing. It is a composite dataset derived from multiple sources including NIST Special Database 19, a Kaggle dataset, and samples from dyslexic children, as described in works such as Rosli et al. [35] and Isa et al. [43].

Classes for Feature Learning: The letters are categorized into 'Normal', 'Reversal', or 'Corrected' forms. These labels were used exclusively for training various deep learning models to learn discriminative visual features from letter characteristics.

Image Specifications: For feature extraction using both Convolutional Neural Networks (CNNs) and Vision-Language Models (VLMs), images were typically processed as 224×224 pixel RGB tensors. Initial CNN training also explored 96×96 pixel images.

Primary Role: Served as the foundational dataset for learning letter-level visual features, particularly for the components conceptualized in Stage 3 of the proposed handwriting analysis pipeline in section (§3.4.4) and for general comparative feature extraction experiments.

Handwriting Paragraph and Page Datasets (for Global Style Assessment)

For the Stage 1 Global Handwriting Style Assessment found in section (§3.4.2), paragraph or full-page handwriting images were utilized from the following sources:

IAM Handwriting Database (Reference Style): Full scanned forms from this publicly available English sentence database [44] provided a general reference for "typical" adult handwriting style. A subset of these images was used to compute an average IAM style embedding.

Dyslexic Children Samples (Target Group): A collection of approximately 50 paragraph-level handwriting samples from children identified as dyslexic, sourced from a public GitHub repository [45] (referred to as the "DyslexicH" portion).

Normal Children Samples (Control Group): Approximately 50 paragraph-level handwriting samples from children considered to have typical handwriting development, also sourced from the same GitHub repository [45] (referred to as the "DyslexicN" portion, providing NC1 and NC2 subsets). These were used for style comparison and for computing an "average Normal Children (NC) style" reference embedding.

3.2.2 Eye-Tracking Datasets

Czech Dyslexia Eye-Tracking Dataset (Benfatto-derived Source Domain)

Content: Raw binocular eye-tracking data (100 Hz sampling rate) collected from approximately 185 Czech children (aged 9-10 years) during silent reading tasks. This dataset, hereafter referred to as the "Czech dataset," was derived from the publicly available collection by Benfatto et al. [46].

Labels: Binary dyslexia risk labels (High-Risk: 97, Low-Risk: 88) were algorithmically derived from the original dataset's folder naming conventions.

Data Format: Tab-separated values, including timestamp and binocular gaze coordinates (LX, LY, RX, RY).

Primary Role: Served as the labeled source domain for developing and evaluating the primary Autoencoder-based eye-tracking methodology in section (§3.5.2) and for the training/fine-tuning phase of the initial BiLSTM exploration (§3.5.3). All model evaluations on this dataset employed subject-stratified 5-fold cross-validation.

***GazeBase* Dataset (English Reading - Pre-training/Target Domain)**

Content: This publicly available dataset, GazeBase [47], contains raw monocular (left eye) eye-tracking data sampled at 1000 Hz from 881 US students performing various tasks, including English text reading (TEX task). Hereafter, this is referred to as the "English dataset" or "GazeBase."

Labels: This dataset is unlabeled for dyslexia.

Data Format: Comma-separated values, including timestamp and left eye gaze coordinates (x, y).

Primary Role: Utilized as the dataset for the self-supervised pre-training phase in the initial BiLSTM exploration in section (§3.5.3) and as the unlabeled target domain for assessing the cross-lingual application of the Autoencoder-based eye-tracking pipeline (§3.5.2).

3.2.3 Experimental Setup

Frameworks: TensorFlow/Keras (v2.18.0) for initial letter-level CNN training. PyTorch (v2.5.1+cu124) for CLIP global style assessment, BiLSTM, and Autoencoder eye-tracking models. Scikit-learn (v1.2.2) for classical classifiers, preprocessing, evaluation. CatBoost, XGBoost, NumPy (v1.26.4), Pandas (v2.2.3), Matplotlib, Seaborn, PIL, Joblib, OpenAI API.

Hardware: Kaggle Notebooks with GPU acceleration (NVIDIA Tesla T4). TensorFlow's `MirroredStrategy` was used for multi-GPU CNN training where applicable.

Reproducibility: Global random seeds (SEED = 123 for Gambo-based Handwriting, SEED = 42 for Eye-Tracking and CLIP Global Style Assessment) set for dataset splitting, shuffling, and model initializations.

3.3 Handwriting Feature Learning and Preparation (*Gambo* Dataset)

This section details the procedures for learning and preparing visual features from the Gambo handwritten letter dataset. It encompasses the initial training and fine-tuning of several Convolutional Neural Network (CNN) backbones (including ResNet50V2, InceptionV3, MobileNetV2, and DenseNet121) to serve as feature extractors. Subsequently, features from selected backbones (ResNet50V2 and InceptionV3) were extracted, fused, and validated. Exploratory feature extraction using Vision-Language Models (CLIP, BEiT) was also conducted. These prepared handwriting features were then used to train a Handwriting-Only (HT-Only) model for dyslexia risk prediction.

3.3.1 CNN Backbone Training and Fine-tuning Experiments (*Gambo* 3-Class Task)

Goal: To train and evaluate several pre-trained CNN architectures (ResNet50V2, InceptionV3, MobileNetV2, DenseNet121) on the 3-class Gambo dataset (Normal, Reversal, Corrected letters) to identify effective base models for feature extraction.

Dataset and Preprocessing: Gambo images were processed as 224×224×3 RGB tensors. The dataset was split into training and validation sets (VAL_SPLIT_RATIO=0.15). Minimal preprocessing involved casting images to `tf.float32` (range $[0, 255]$). Data augmentation (`RandomRotation(0.1)`, `RandomZoom(0.1)`, etc.) and internal pixel scaling to $[-1, 1]$ via a Lambda layer were applied within the model. Datasets were batched (GLOBAL_BATCH_SIZE=32). Data validation checks (for NaNs and invalid label ranges) were performed before fitting.

Common Model Architecture Head: For each base CNN, a common classification head was attached, consisting of: `GlobalAveragePooling2D` → `BatchNormalization` → `Dense(256, 'relu')` → `BatchNormalization` → `Dropout(0.4)` → `Dense(3, 'softmax')`.

Base Models Investigated:

- ResNet50V2 [23]
- InceptionV3 [25]
- MobileNetV2 [48]
- DenseNet121 [49]

Training Strategy (Two-Stage): For each base model:

1. Stage 1 (Head Training): The pre-trained base model layers were frozen. The classification head was trained using the Adam optimizer (learning rate INITIAL_LR = 1×10^{-4}, clipnorm CLIPNORM = 1.0) with Categorical Crossentropy loss (label smoothing LABEL_SMOOTHING = 0.1) for a set number of epochs (INITIAL_EPOCHS = 10 or 15, depending on the notebook iteration).
2. Stage 2 (Fine-Tuning): Selected layers of the base model were unfrozen (fine-tune start layer varied per model, e.g., layer 165 for ResNet50V2, layer 140 for MobileNetV2, layer 400 for DenseNet121, as per FINE_TUNE_LAYER_COUNTS). Training continued with a lower learning rate (Adam optimizer, learning rate FINE_TUNE_LR = 1×10^{-5}) for additional epochs (FINE_TUNE_EPOCHS = 10 or 20).

Training Execution: Training utilized TensorFlow's `MirroredStrategy` where applicable. Callbacks included `EarlyStopping` (monitor `val_loss`, patience PATIENCE), and `ModelCheckpoint` (monitor `val_accuracy`, save best weights only, save weights only).

Outcome: The training process for each CNN backbone was completed. The final weights (from either Stage 1 or Stage 2, depending on which yielded better validation accuracy) and training histories were saved. For instance, for ResNet50V2 and InceptionV3, the Stage 1 models (head training only) were ultimately selected for feature extraction in the subsequent fusion pipeline, while for MobileNetV2 and DenseNet121, fine-tuned weights were selected as their best.

3.3.2 Evaluation of Individual CNN Backbone Features (*Gambo* 3-Class Task)

Goal: To assess the quality of features extracted from each individually trained CNN backbone (ResNet50V2, MobileNetV2, DenseNet121, and InceptionV3 if separately evaluated) by using them to train simple classifiers on the 3-class Gambo task.

Feature Extraction: For each trained CNN backbone, features were extracted from the output of its `GlobalAveragePooling2D` layer. Gambo images were typically resized to 224×224×3 for this step.

Feature Scaling: Extracted features for each model were standardized using `sklearn.preprocessing.StandardScaler` (fit on training features, applied to train and test).

Classifier Tested: A Logistic Regression classifier (`max_iter`=1500, `multi_class='ovr'`, `solver='liblinear'`, C=1.0) was trained on the scaled features from each backbone.

Procedure: The Logistic Regression model was trained on the scaled training features and evaluated on the scaled test features for each backbone. The performance metrics for this evaluation are detailed in Chapter 4. This step provided insights into the relative representational power of each individual CNN backbone before feature fusion.

3.3.3 Handwriting Feature Fusion (ResNet50V2 & InceptionV3) and Validation (*Gambo*)

This phase focused on extracting features specifically from the selected ResNet50V2 and InceptionV3 models (based on initial training in section §3.3.1), fusing these features, and validating their combined utility on the 3-class Gambo letter classification task.

Prerequisites: Utilized the saved model configurations and weights (specifically Stage 1 weights for ResNet50V2 and InceptionV3) from the CNN backbone training experiments.

Ordered Data Loading: Image paths for the Gambo train and test sets were loaded and sorted alphabetically. Images were resized to 224×224×3.

Model Loading & Feature Extractor Creation: The saved `.keras` models for ResNet50V2 and InceptionV3 were loaded (using `safe_mode=False`). Feature extractors were created from their `GlobalAveragePooling2D` layers.

Ordered Feature Extraction: Gambo train and test images were processed through both the ResNet50V2 and InceptionV3 feature extractors, yielding two sets of 2048-dimensional feature vectors per image.

Feature Fusion: For each image, the 2048D features from ResNet50V2 and the 2048D features from InceptionV3 were concatenated, resulting in a 4096-dimensional fused feature vector.

Feature Scaling: The 4096D fused features were standardized using `sklearn.preprocessing.StandardScaler`.

Validation of Fused Features (3-Class Task): The scaled 4096D fused features were used to train and evaluate several classical machine learning classifiers: Logistic Regression, Linear SVM, RBF SVM, Random Forest, and MLP. The performance

of these classifiers on the Gambo test set is detailed in Chapter 4. These validated 4096D fused ResNet/Inception features were designated as the primary handwriting features for training the HT-Only dyslexia risk prediction model .

3.3.4 Exploratory Handwriting Feature Extraction (CLIP & BEiT on *Gambo*)

In addition to the ResNet50V2 and InceptionV3 backbones, an exploratory investigation was conducted into the utility of pre-trained Vision-Language Models (VLMs), specifically CLIP and BEiT, as feature extractors for the Gambo handwritten letter images. The objective was to assess their capability to generate discriminative features for the 3-class letter classification task (Normal, Reversal, Corrected).

CLIP Feature Extraction

Model and Processor: The `openai/clip-vit-base-patch32` model and its corresponding `CLIPProcessor` were loaded from the HuggingFace Transformers library. The vision component of the CLIP model was utilized in a frozen state (no fine-tuning of CLIP weights) for feature extraction.

Data Preparation: The Gambo dataset (train, validation, and test splits) was prepared using a PyTorch `Dataset` class. Images were loaded using PIL, converted to RGB, and then processed by the `CLIPProcessor` to match the model's expected input format (typically 224×224 pixels, with appropriate normalization).

Feature Extraction Process: The preprocessed images were passed through the frozen CLIP vision model in batches (CLIP_INFERENCE_BATCH_SIZE=32). Features were extracted from the model's `pooler_output`, which typically represents the embedding of the [CLS] token (or equivalent global representation) from the vision transformer, resulting in 768-dimensional feature vectors per image.

Output: Extracted CLIP features for the train, validation, and test sets of the Gambo dataset were saved as `.npy` files, along with their corresponding labels.

BEiT Feature Extraction

Model and Processor: The `microsoft/beit-base-patch16-224-pt22k-ft22k` model and its corresponding `AutoImageProcessor` were utilized, loaded from the HuggingFace Transformers library. The base BEiT model (without a classification head) was used in a frozen state for feature extraction.

Data Preparation: Similar to the CLIP pipeline, the Gambo dataset splits were loaded using a PyTorch `Dataset` class. Images were processed by the BEiT `AutoImageProcessor` to meet the model's input requirements (e.g., 224×224 pixels, specific normalization).

Feature Extraction Process: Preprocessed images were fed in batches (BEIT_INFERENCE_BATCH_SIZE=32) to the frozen BEiT model. Features were derived from the model's `pooler_output` (representing the [CLS] token embedding), yielding 768-dimensional feature vectors for each image.

Output: Extracted BEiT features for the train, validation, and test sets of the Gambo dataset, along with their labels, were saved as `.npy` files.

3.3.5 Evaluation of Handwriting Features (*Gambo* 3-Class Task)

Input: Scaled feature sets: 4096D fused ResNet50V2/InceptionV3 features in section (§3.3.3).

Classifiers Tested: Logistic Regression (`max_iter`=1500, `C`=1.0), Linear SVM (`max_iter`=2000, `C`=0.1), RBF SVM (`C`=1.0), Random Forest (`n_estimators`=150, `max_depth`=20), MLP (Scikit-learn: `hidden_layers`=(128, 64), `max_iter`=300).

Procedure Output: For each feature set, classifiers were trained on the respective scaled training features and subsequently applied to the test features to generate predictions. Trained Scikit-learn classifier models (`.pkl`) were saved. This step aimed to determine the most discriminative handwriting features for potential use in dyslexia risk prediction.

3.4 Handwriting Analysis: A 3-Stage AI Pipeline for Dyslexia Risk Assessment

This section details the proposed 3-Stage AI pipeline (conceptualized in Figure 3.1) for comprehensive dyslexia risk assessment from handwriting. It integrates global style analysis, contextual word-level processing, and fine-grained letter-level visual analysis. Foundational experiments for learning letter-level features on the Gambo dataset, which inform Stage 3, are described first.

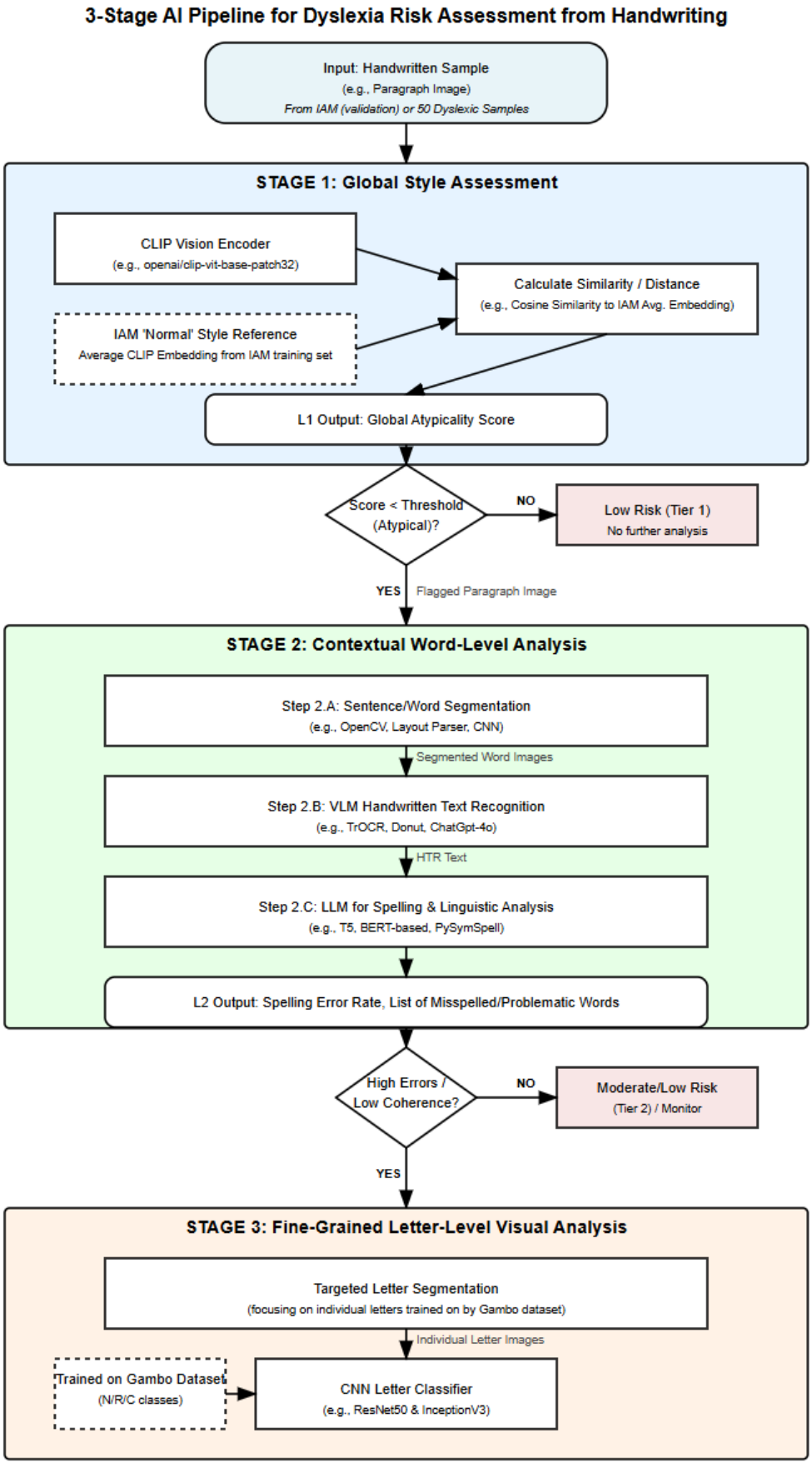


Figure 3.1: The Proposed 3-Stage AI Pipeline for Dyslexia Risk Assessment from Handwriting.

3.4.1 Foundational Letter-Level Feature Learning (*Gambo* Dataset)

To inform the fine-grained letter analysis component (Stage 3) of the proposed pipeline, extensive experiments were conducted to learn and validate visual features from isolated handwritten letters using the Gambo dataset.

CNN Backbone Training and Fine-tuning (*Gambo* 3-Class Task)

Goal: To train several pre-trained CNN architectures on the 3-class Gambo dataset to identify effective base models for extracting features from isolated letters.

Models Investigated: ResNet50V2, InceptionV3, MobileNetV2, DenseNet121.

Process: Each model, with a common classification head, was trained using a two-stage strategy (head training with frozen base, followed by fine-tuning selected layers) on 224x224 RGB Gambo images.

Evaluation of Individual CNN Backbone Features (*Gambo* 3-Class Task)

Goal: To assess the quality of features extracted from each individually trained CNN backbone.

Process: Features from the `GlobalAveragePooling2D` layer of each backbone were extracted from 224x224 Gambo images, scaled, and used to train Logistic Regression classifiers for the 3-class task.

Fused ResNet50V2 & InceptionV3 Feature Validation (*Gambo* 3-Class Task)

Goal: To validate fused features from the selected ResNet50V2 and InceptionV3 backbones.

Process: 2048D features from Stage 1 trained ResNet50V2 and InceptionV3 were extracted from 224x224 Gambo images, concatenated (4096D), scaled, and evaluated using various classical classifiers (Logistic Regression, SVMs, Random Forest, MLP) on the 3-class task. These 4096D features were designated as the primary learned letter features.

Exploratory VLM (CLIP & BEiT) Features for Letter Classification (*Gambo* 3-Class Task)

Goal: To explore CLIP and BEiT as alternative feature extractors for isolated Gambo letters.

Process: Pre-trained frozen CLIP (`openai/clip-vit-base-patch32`) and BEiT (`microsoft/beit-base-patch16-224-pt22k-ft22k`) models were used to extract 768D features from Gambo images. These features were scaled and evaluated using classical classifiers on the 3-class task.

3.4.2 Stage 1 of Proposed Pipeline: Global Handwriting Style Assessment

This initial layer of the proposed handwriting pipeline (Figure 3.1) provides a global assessment of handwriting style atypicality using paragraph-level images.

Goal: To compute a "Global Atypicality Score" by comparing CLIP embeddings of input handwriting samples to reference style embeddings.

Datasets: IAM (reference adult style), Dyslexic Children (target), and Normal Children paragraph/page images.

Method:
1. Pre-trained CLIP (`openai/clip-vit-base-patch32`) vision encoder extracted 512D image embeddings.
2. "Average IAM Style" and "Average NC Combined Style" reference embeddings were computed.
3. Cosine similarity of Dyslexic and NC samples was calculated against these reference embeddings to derive atypicality scores.
4. An exploratory fine-tuning of the CLIP vision model on a Dyslexic vs. Not Dyslexic (combined) binary task was performed using 5-fold CV to assess if task-specific adaptation improves style discrimination.

Output for Pipeline: A "Global Atypicality Score". Samples exceeding a threshold would be flagged for further analysis in Stage 2.

3.4.3 Stage 2 of Proposed Pipeline: Contextual Word-Level Analysis

This Stage (Figure 3.1) focuses on detailed word-level analysis of handwriting samples flagged by Stage 1. The methodology explored components relevant to this layer using OpenAI's GPT-4o model on pre-segmented line images.

Goal: To perform HTR, spelling check, and visual anomaly detection at the word/letter level within lines of handwriting.

Datasets: Line images from IAM, and lines segmented from Dyslexic (D) and Normal Children (NC1) paragraph samples.

Method (using GPT-4o): 1. Line images were provided to the GPT-4o model.

2. A structured prompt guided the VLM to transcribe the line, identify misspelled words, and detail visual anomalies (type, letter, position, description) for each word.

3. The output was a JSON object containing line transcription, word-level analyses, and line summaries.

Output for Pipeline: Metrics such as spelling error rate, list of misspelled/problematic words, and types/counts of visual anomalies. High error rates or low coherence would flag samples for Stage 3.

3.4.4 Stage 3 of Proposed Pipeline: Fine-Grained Letter-Level Visual Analysis

This stage (Figure 3.1) is designed for a targeted visual analysis of individual letter images, conceptually segmented from problematic words identified in Stage 2. The methodology involves several image processing steps to isolate and prepare letter candidates, followed by classification using a pre-trained model.

Goal: To classify individual segmented letter images into categories such as 'Normal', 'Reversal', or 'Corrected' to provide detailed insights into visual letter formation characteristics.

Input to Stage 3: Cropped images of individual letters. For the standalone demonstration of this component full handwritten page images from the "DyslexicH" dataset were used as input.

Image Preprocessing and Segmentation: The process to obtain individual letter images from a full page involved the following steps, implemented using OpenCV and Pillow:

1. Full Page Preprocessing: The input page image (PIL format) was converted to BGR, then to grayscale. Gaussian blur (e.g., kernel (5,5)) was applied for noise reduction, followed by adaptive thresholding (e.g., Gaussian, block size 35, C value 10) to binarize the image, resulting in white text on a black background. An attempt at line removal was also explored using morphological operations before binarization. Further morphological cleaning (e.g., opening with a (2,2) kernel) was applied to the binary image.

2. Line Segmentation: The preprocessed binary page image was dilated using a rectangular kernel (e.g., width 60-80, height 3, iterations 2-3) to connect components within text lines. Contours were then found on this dilated

image. Bounding boxes of contours meeting certain width and height criteria (relative to page size and aspect ratio) were identified as text lines and sorted vertically.

3. Character Segmentation from Lines: Each binary line crop was further processed. Contours were found directly on the (potentially further morphologically processed) binary line image. Bounding boxes of these contours were filtered based on area, height, width, and aspect ratio criteria to identify potential character segments. These character boxes were sorted horizontally.
4. Individual Character Cropping and Preparation for Model:
    - For each identified character bounding box, a slightly padded crop was taken from the original color page image.
    - This color crop was converted to grayscale and binarized (e.g., using Otsu's thresholding, inverted to white letter on black background).
    - The binarized character was converted to a 3-channel RGB format (as expected by the classification model).
    - The character was padded to a square aspect ratio (maintaining its original form within the square) using a black background.
    - Finally, the square image was resized to the model's input dimensions (e.g., 224×224 pixels) using linear interpolation and formatted as a float32 numpy array with an added batch dimension.

Letter Classification Component:

- Model: The best performing classifier trained on fused features from Resnet50V2 and InceptionV3 were loaded.
- Prediction: The prepared individual character image batch was passed to the loaded model to obtain prediction probabilities for the classes 'Normal', 'Reversal', and 'Corrected'. The class with the highest probability was taken as the prediction, along with its confidence score.

Output for Pipeline: For each segmented letter, its predicted class ('Normal', 'Reversal', 'Corrected') and associated confidence. This fine-grained visual error information on specific letters from problematic words would contribute to the overall dyslexia risk assessment profile.

3.4.5 Overall Dyslexia Risk Assessment from 3-Layer Handwriting Pipeline

The outputs from Stage 1 (Global Atypicality Score), Stage 2 (Spelling Error Rate, Linguistic Coherence, Problematic Word Lists), and Stage 3 (Classification of specific letter errors) would be conceptually combined, potentially using a rule-based system

or a final meta-classifier, to arrive at an overall dyslexia risk assessment based on handwriting. The precise mechanism for this final combination is an area for future development beyond the scope of the component evaluations in this thesis.

3.5 Eye-Tracking Cross-Lingual Model for Dyslexia Risk Assessment

This section details the methodologies employed for developing an eye-tracking based system for dyslexia risk assessment, with a primary focus on an Autoencoder (AE) based pipeline designed for feature extraction from gaze-path images and subsequent cross-lingual application. An initial exploratory phase using BiLSTM sequence modeling is also briefly outlined.

3.5.1 Common Eye-Tracking Data Preprocessing

The following preprocessing steps were applied consistently to both the Czech (Benfatto-derived) and English (GazeBase) eye-tracking datasets before any model-specific processing:

- NaN Handling: Missing LX and LY coordinates were addressed using linear interpolation (with a 100ms gap limit, based on the TARGET_SAMPLING_RATE of 100Hz), followed by forward and backward filling. Any remaining NaNs after these steps were imputed with 0.
- Downsampling (for GazeBase Data): The English GazeBase data, originally sampled at 1000Hz, was downsampled by a factor of 10 (DOWNSAMPLE_FACTOR = 10) to align with the 100Hz effective sampling rate of the Czech dataset.
- Coordinate Selection: For consistency across datasets and to simplify the input for image generation, only Left Eye coordinates (LX, LY) were utilized for subsequent analysis.

3.5.2 Primary Methodology: Image Encoding with Autoencoders (AE) for Feature Extraction and Classification

This approach, forming the core of the eye-tracking model, involved converting gaze scanpath segments from the Czech dataset into images, training an Autoencoder on these images, extracting features based on reconstruction quality, training classifiers for dyslexia risk on the Czech data, and finally applying the entire pipeline to the English GazeBase data.

Coordinate Normalization (Czech Data): To standardize the gaze paths and ensure consistency before image generation, the Left Eye (LX) coordinates for each individual trial within the Czech dataset were normalized to a common range of $[0, 1]$. This step is crucial as raw gaze coordinates can vary significantly in their absolute range depending on screen resolution, participant distance, and calibration specifics. Normalization brings all trajectories into a comparable unit square. Figure 3.2 shows examples of raw gaze trajectories from both Czech and English datasets before this per-trial normalization, illustrating the varied coordinate scales. Figure 3.3 then demonstrates the effect of this normalization, showcasing sample trajectories scaled to the $[0, 1]$ range.

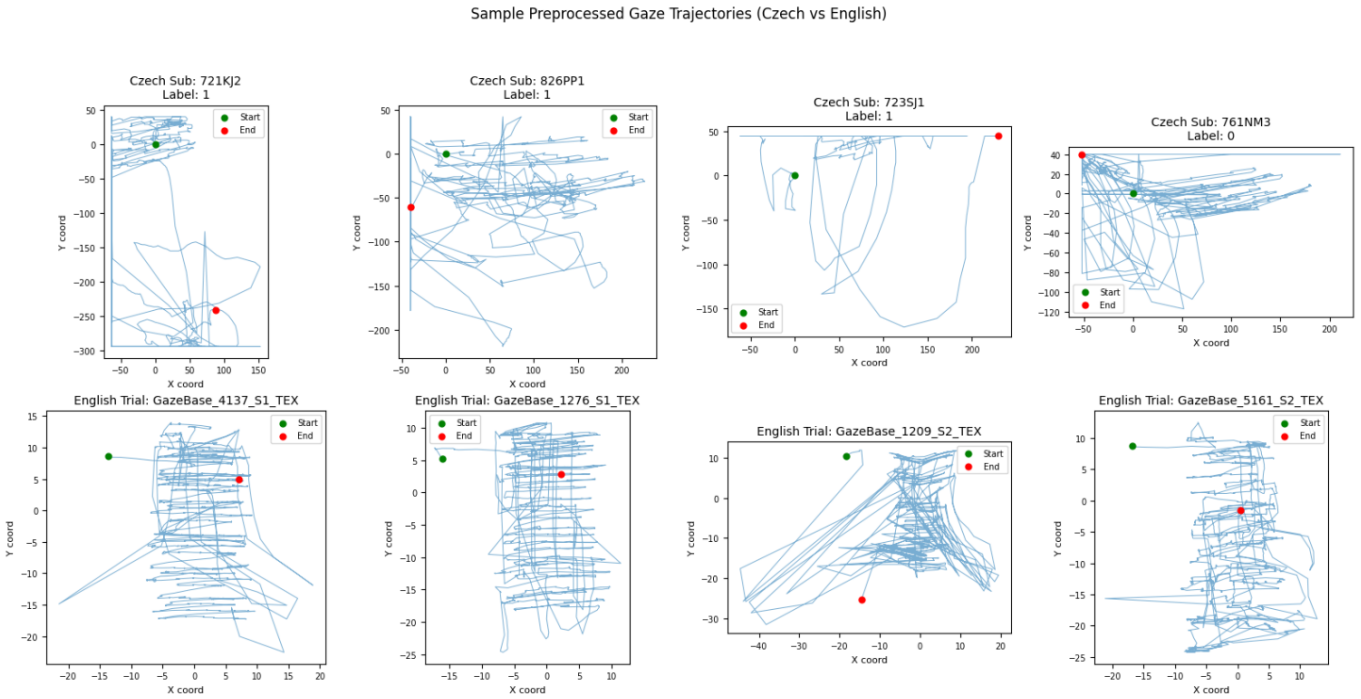


Figure 3.2: Sample Preprocessed Gaze Trajectories (Czech vs English) Before Per-Trial Normalization. Note the differing coordinate scales across trials.

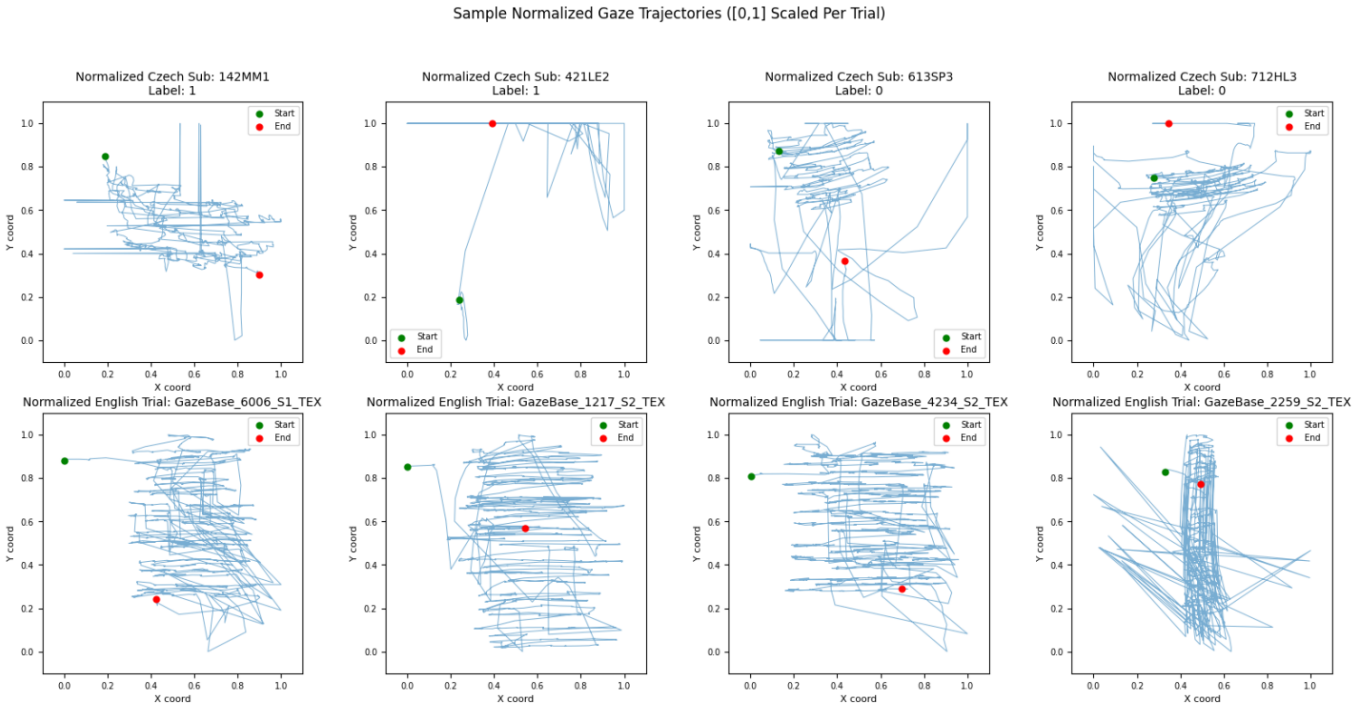


Figure 3.3: Sample Normalized Gaze Trajectories from Czech and English Trials, Scaled to the [0,1] Range Per Trial.

Gaze-Path Image Generation (Czech Data):

- Normalized LX, LY data from the Czech dataset were segmented into fixed-length windows of 0.5 seconds (TW_SAMPLES = 50 at 100Hz) with a stride of 0.2 seconds (STRIDE_SAMPLES = 20).
- Each segment was plotted as a 64×64 pixel grayscale image (IMG_SIZE=64), with gaze lines rendered with a specified opacity (PLOT_LINE_OPACITY = 0.2) and DPI (PLOT_DPI = 50) to capture the scanpath trajectory. Playing with these with small margins can significantly change the results.

Autoencoder Model (ConvAutoencoder) Training (Czech Data Only):

- Architecture: A Convolutional Autoencoder was implemented and it consisted of an encoder path with four convolutional blocks (Conv2D-BatchNorm-ReLU-MaxPool2D) reducing dimensionality, and a symmetric decoder path with upsampling and convolutional layers to reconstruct the input image. The final decoder layer used a Sigmoid activation.
- Training Data Strategy: The AE was trained exclusively on the gaze-path images generated from all available subjects (both low-risk and high-risk) in the Czech dataset (AE_TRAIN_DATA_STRATEGY = `'all'`).

- Training Process: The model was trained to minimize Binary Cross Entropy loss between the original and reconstructed images for AE_EPOCHS = 50, using the Adam optimizer (AE_LR = 1×10^{-3}) with early stopping based on validation loss (AE_PATIENCE = 10). The model with the best validation loss was saved.

AE-Based Feature Extraction (Czech Data): Using the trained Autoencoder, reconstruction error features were extracted for each gaze-path image segment from the Czech dataset.

- For each segment image, the pixel-wise Binary Cross Entropy loss between the input and its reconstruction was calculated. This per-image error was then also calculated per trial.
- For each trial, five range-based features were computed from the sorted distribution of reconstruction errors of its constituent segments (e.g., max_error - min_error, 2nd_max_error - 2nd_min_error, etc., NUM_ERROR_FEATURES=5).

Dyslexia Risk Classifier Training (Czech Data):

- The extracted reconstruction error features (5 dimensions per trial) from the Czech data were scaled using `StandardScaler`.
- Various classical classifiers (Logistic Regression, RandomForestClassifier, SVC) were trained using these scaled features and the binary dyslexia risk labels from the Czech dataset.
- Hyperparameter tuning was performed using `GridSearchCV` with 5-fold subject-stratified cross-validation (CLASSIFIER_CV_SPLITS=5), optimizing for F1-score.
- The best performing classifier and its parameters were selected as the final ET-Only dyslexia risk model for the Czech data.

Cross-Lingual Application to English *GazeBase* Data:

- The same gaze-path image generation process and AE-based reconstruction error feature extraction (using the Czech-trained AE) were applied to the preprocessed English GazeBase data.
- The `StandardScaler` (fitted on Czech training data) and the selected best classifier (SVC, trained on Czech data) were then used to predict dyslexia risk labels for the English GazeBase trials.

3.5.3 Exploratory Analysis: Sequence Modeling with BiLSTM (*GazeBase* & Czech Datasets)

An initial investigation involved developing a Bidirectional Long Short-Term Memory (BiLSTM) network for dyslexia risk prediction.

Process: This exploration included generating fixed-length sequences from LX, LY coordinates, attempting self-supervised pre-training on the English GazeBase dataset (masked coordinate prediction task), and subsequently training/fine-tuning the BiLSTM model on the labeled Czech dataset for binary dyslexia risk classification using subject-stratified 5-fold cross-validation.

Role: This approach served as an early exploratory baseline for sequential eye-tracking data analysis. The primary eye-tracking methodology shifted to the AE-based approach due to its focus in the provided experimental notebooks for feature extraction and cross-lingual application.

3.6 Evaluation Metrics

The performance of the various models and feature sets developed in this thesis was assessed using standard evaluation metrics appropriate to each specific task and dataset.

Evaluation of Handwriting Feature Learning and Letter Classification (*Gambo* Dataset)

For the foundational experiments on the Gambo dataset in (§3.3), which involved training and evaluating various feature extraction approaches (individual CNNs, fused CNNs, exploratory VLMs) for the 3-class letter characteristic task ('Normal', 'Reversal', 'Corrected'), the following metrics were primarily used:

- Accuracy: Overall proportion of correctly classified letters.
- Precision, Recall, and F1-Score: Calculated per class and as weighted/macro averages to assess performance across the different letter categories.
- Classification Report: A detailed summary of per-class precision, recall, and F1-score.
- Confusion Matrix: To visualize the classification performance across the three letter types.

These evaluations served to validate the quality of the learned letter features before their conceptual application within the 3-Stage handwriting pipeline.

Evaluation of the 3-Stage Handwriting Pipeline Components

The components of the proposed 3-Stage AI Pipeline for Dyslexia Risk Assessment from Handwriting (Figure 3.1) were evaluated as follows:

- Stage 1 (Global Handwriting Style Assessment using CLIP): Evaluation was primarily qualitative and comparative in (§3.4.2). It involved:
  - Visual analysis of similarity score distributions (histograms, boxplots) for Dyslexic (D), Normal Children (NC), and IAM groups against reference style embeddings.
  - t-SNE/UMAP visualizations of CLIP embeddings to observe group clustering.
  - Analysis of the proportion of D and NC samples flagged as "atypical" based on defined similarity thresholds, for both off-the-shelf and fine-tuned CLIP models.
  - For the fine-tuned CLIP vision model, standard classification metrics (Accuracy, F1-score) from its 5-fold cross-validation on the D vs. NC binary task were considered.
- Stage 2 (VLM Line-Level HTR and Anomaly Detection - OpenAI GPT-4o): The assessment of the OpenAI GPT-4o model's performance on line-level analysis (§3.4.3) included:
  - Qualitative review of the JSON outputs for transcription accuracy, appropriateness of spelling corrections, and relevance of identified visual anomalies.
  - Calculation of aggregated statistics (e.g., misspelling rates, counts of specific anomaly types like reversals) per sample group (IAM, D, NC1).
- Stage 3 (Application of Letter Classification to Segmented Samples): For the demonstration of the letter classification component on characters segmented from full "DyslexicH" pages (§3.4.4):
  - Qualitative assessment of the segmentation and classification output via visual examples.
  - Aggregated distribution of predicted letter classes ('Normal', 'Reversal', 'Corrected') across all segmented characters.
  - Average and median confidence scores of the letter classifier on these segmented characters.

The overall assessment of the 3-Stage pipeline's potential for dyslexia risk identification remains conceptual, based on the performance of its individual stages, as an end-to-end quantitative evaluation was beyond the scope of this phase.

Evaluation of Eye-Tracking Dyslexia Risk Prediction Models (Czech Dataset)

For the eye-tracking models developed to predict binary dyslexia risk on the Czech (Benfatto-derived) dataset (§3.5), including both the exploratory BiLSTM model and the primary Autoencoder (AE)-based classifiers, evaluation was conducted using subject-stratified 5-fold cross-validation. The primary metrics, focused on identifying the 'High Risk' class, include:

- Accuracy: Overall correctness: $\frac{TP + TN}{TP + TN + FP + FN}$
- Precision: Positive predictive value: $\frac{TP}{TP + FP}$
- Recall (Sensitivity): True positive rate: $\frac{TP}{TP + FN}$
- F1-Score: Harmonic mean of Precision and Recall: $2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$
- AUC (Area Under the ROC Curve): Measure of class separability.
- Confusion Matrix: Visual summary of prediction outcomes (TP, TN, FP, FN).

Weighted averages across both classes were also calculated for Precision, Recall, and F1-Score. For the Autoencoder-feature-based classifiers, the primary metric for model and feature selection during the `GridSearchCV` process was the cross-validated F1-score (§3.5.2).

Analysis of Cross-Lingual Application of Eye-Tracking Model (*GazeBase* Dataset)

For the application of the Czech-trained AE-based eye-tracking pipeline to the unlabeled English GazeBase dataset (§3.5.2), where ground truth dyslexia labels were absent, the evaluation was necessarily qualitative. It focused on:

- Analyzing the distribution of the dyslexia risk predictions (percentage of trials classified as 'High Risk' vs. 'Low Risk').

This analysis aimed to provide initial insights into the model's behavior in a different linguistic and demographic context.

Detailed numerical results, qualitative observations, and comparative analyses for all these evaluations are presented in Chapter 4.

Chapter 4

Results and Discussion

4.1 Introduction

This chapter presents the empirical results obtained from the various experiments detailed in Chapter 3. The findings cover the outcomes of the feature learning stages for both handwriting and eye-tracking modalities, the performance of the unimodal dyslexia risk prediction models, and critically, the evaluation of the multimodal late fusion approach. Results are assessed using the standard evaluation metrics defined in §3.6, including accuracy, precision, recall, F1-score, and AUC, providing a comprehensive view of each model's capabilities in identifying dyslexia risk. This analysis aims to highlight the effectiveness of the individual pipelines and the potential benefits of multimodal integration.

4.2 Handwriting Analysis: Feature Learning and Validation on *Gambo* Dataset

This section details the outcomes related to learning and validating visual features from the Gambo handwritten letter dataset for the 3-class task (Normal, Reversal, Corrected).

4.2.1 Evaluation of Features from Individual CNN Backbones

To assess the quality of the visual representations learned by the different CNN backbones during their training on the 3-class Gambo letter task (§3.3.1), features were extracted from each model's `GlobalAveragePooling2D` layer. These features were then scaled and used to train a Logistic Regression classifier to predict the letter categories (Normal, Reversal, Corrected) on the Gambo test set, as detailed in §3.3.2.

Table 4.1 summarizes the overall performance of the Logistic Regression classifier using features from each respective CNN backbone.

The following tables provide a more detailed breakdown of the classification performance for each feature set, including per-class metrics.

Table 4.1: Performance Summary of Logistic Regression on Features from Individual CNN Backbones (*Gambo* 3-Class Task)

CNN Backbone	Accuracy	Weighted Precision	Weighted F1-Score
ResNet50V2	**0.9266**	**0.9292**	**0.927**
InceptionV3	**0.9149**	**0.914**	**0.915**
MobileNetV2	0.640	0.661	0.640
DenseNet121	0.598	0.670	0.589

MobileNetV2 Features

Using features derived from the MobileNetV2 backbone, the Logistic Regression classifier attained an accuracy of 63.98%. The classification report in Table 4.2 provides further details. Figure 4.1 displays the corresponding confusion matrix.

Table 4.2: Classification Report for Logistic Regression on MobileNetV2 Features (*Gambo* 3-Class Task)

Class	Precision	Recall	F1-Score
Normal	0.60	0.67	0.63
Reversal	0.59	0.71	0.65
Corrected	0.79	0.54	0.64
Accuracy			0.64
Macro Avg	0.66	0.64	0.64
Weighted Avg	0.66	0.64	0.64

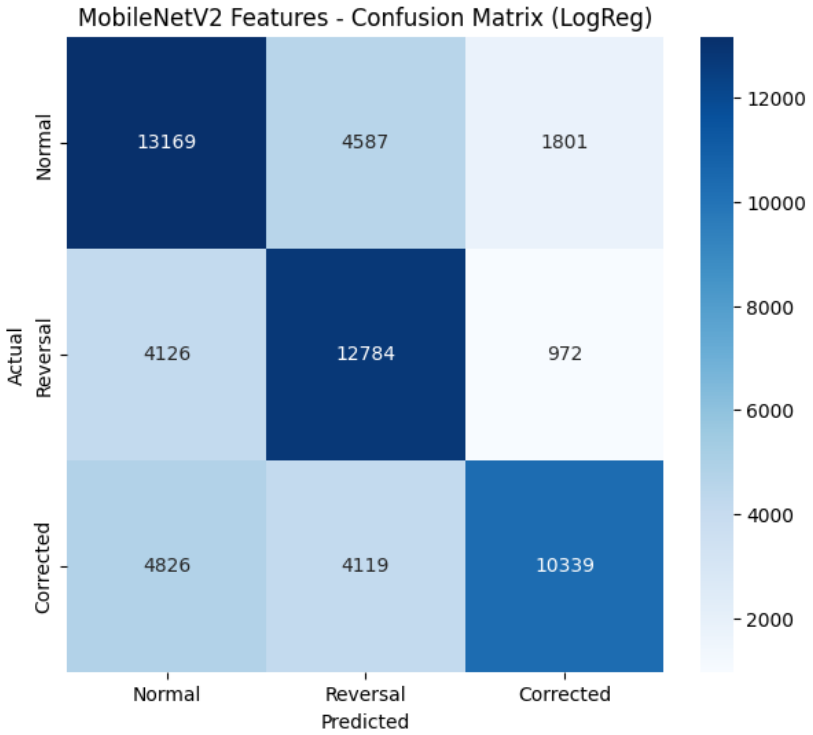


Figure 4.1: Confusion Matrix for Logistic Regression on MobileNetV2 Features (*Gambo* 3-Class Task).

DenseNet121 Features

The Logistic Regression classifier trained on features from the DenseNet121 backbone resulted in an accuracy of 59.78%. The detailed report is shown in Table 4.3. The confusion matrix for this evaluation is presented in Figure 4.2.

Table 4.3: Classification Report for Logistic Regression on DenseNet121 Features (*Gambo* 3-Class Task)

Class	Precision	Recall	F1-Score
Normal	0.51	0.78	0.61
Reversal	0.90	0.36	0.51
Corrected	0.63	0.64	0.63
Accuracy			0.60
Macro Avg	0.68	0.59	0.59
Weighted Avg	0.67	0.60	0.59

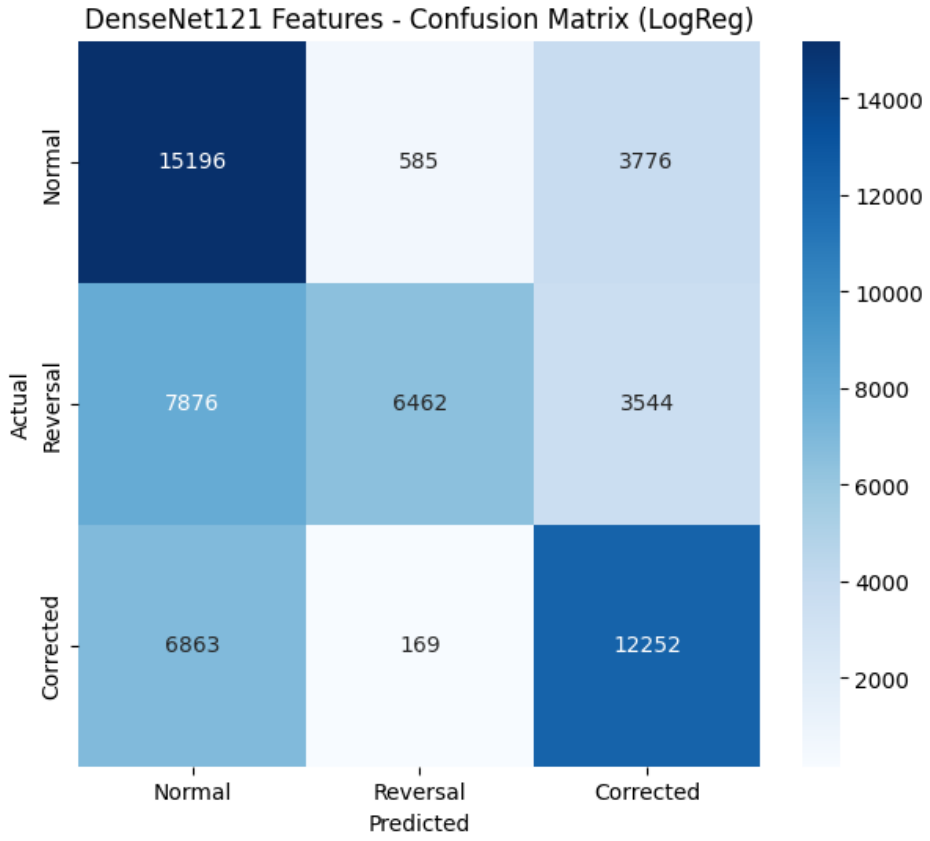


Figure 4.2: Confusion Matrix for Logistic Regression on DenseNet121 Features (*Gambo* 3-Class Task).

ResNet50V2 Features

When features extracted from the ResNet50V2 backbone were used, the Logistic Regression classifier achieved an overall accuracy of 92.7%. The detailed classification report is presented in Table 4.4. The confusion matrix for this evaluation is shown in Figure 4.3.

Table 4.4: Classification Report for Logistic Regression on ResNet50V2 Features (*Gambo* 3-Class Task)

Class	Precision	Recall	F1-Score
Normal	0.86	0.90	0.88
Reversal	0.94	0.90	0.92
Corrected	0.95	0.95	0.95
Accuracy			0.92
Macro Avg	0.92	0.92	0.92
Weighted Avg	0.92	0.92	0.92

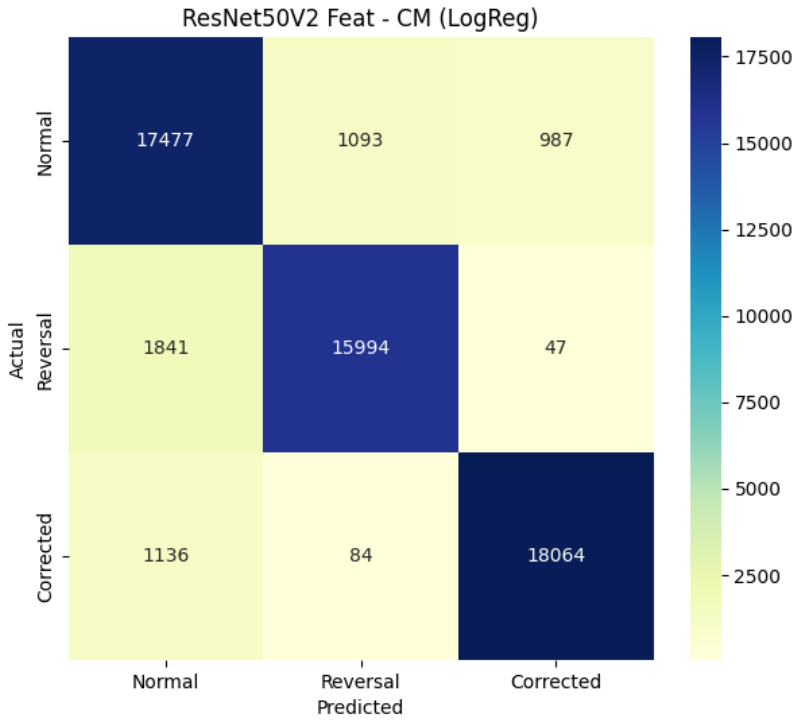


Figure 4.3: Confusion Matrix for Logistic Regression on ResNet50V2 Features (*Gambo* 3-Class Task).

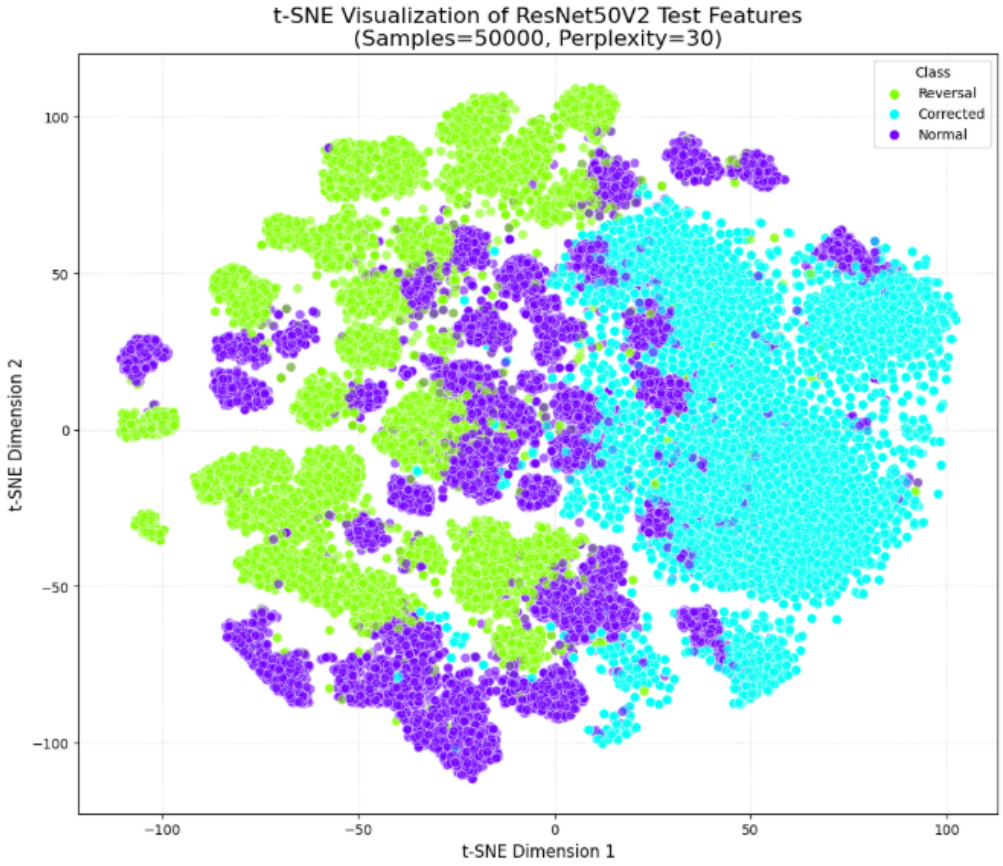


Figure 4.4: t-SNE Visualization of Fused ResNet50V2 & InceptionV3 Test Features (*Gambo* 3-Class Task).

InceptionV3 Features

When features extracted from the InceptionV3 backbone were used, the Logistic Regression classifier achieved an overall accuracy of 91.51%. The detailed classification report is presented in Table 4.5. The confusion matrix for this evaluation is shown in Figure 4.5.

Table 4.5: Classification Report for Logistic Regression on InceptionV3 Features (*Gambo* 3-Class Task)

Class	Precision	Recall	F1-Score
Normal	0.86	0.88	0.87
Reversal	0.92	0.90	0.91
Corrected	0.94	0.94	0.94
Accuracy			0.91
Macro Avg	0.91	0.91	0.91
Weighted Avg	0.91	0.91	0.91

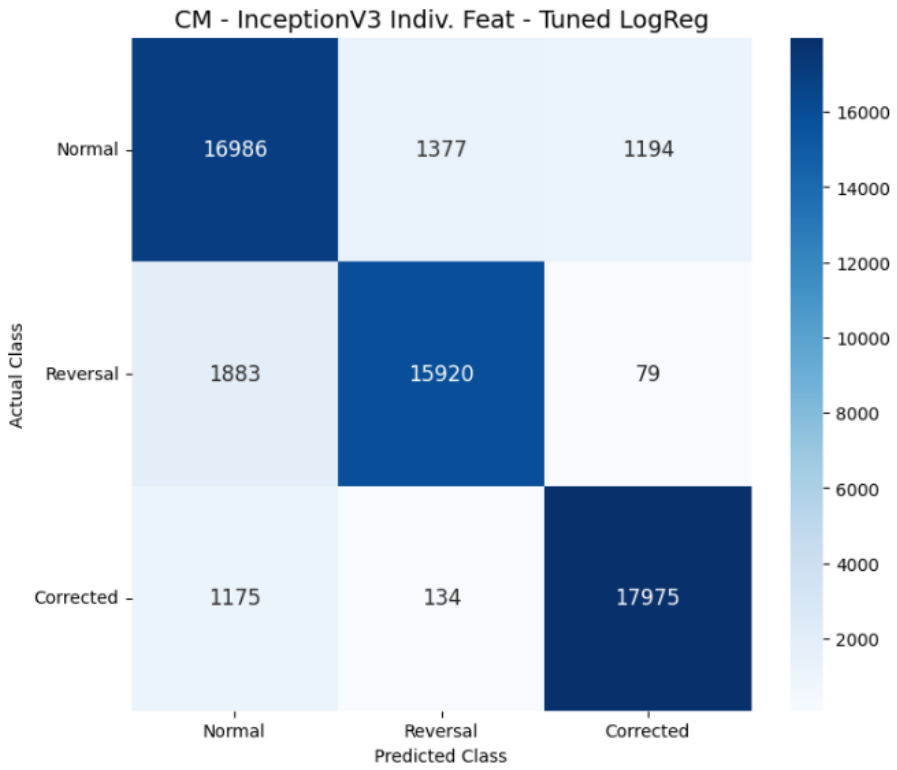


Figure 4.5: Confusion Matrix for Logistic Regression on InceptionV3 Features (*Gambo* 3-Class Task).

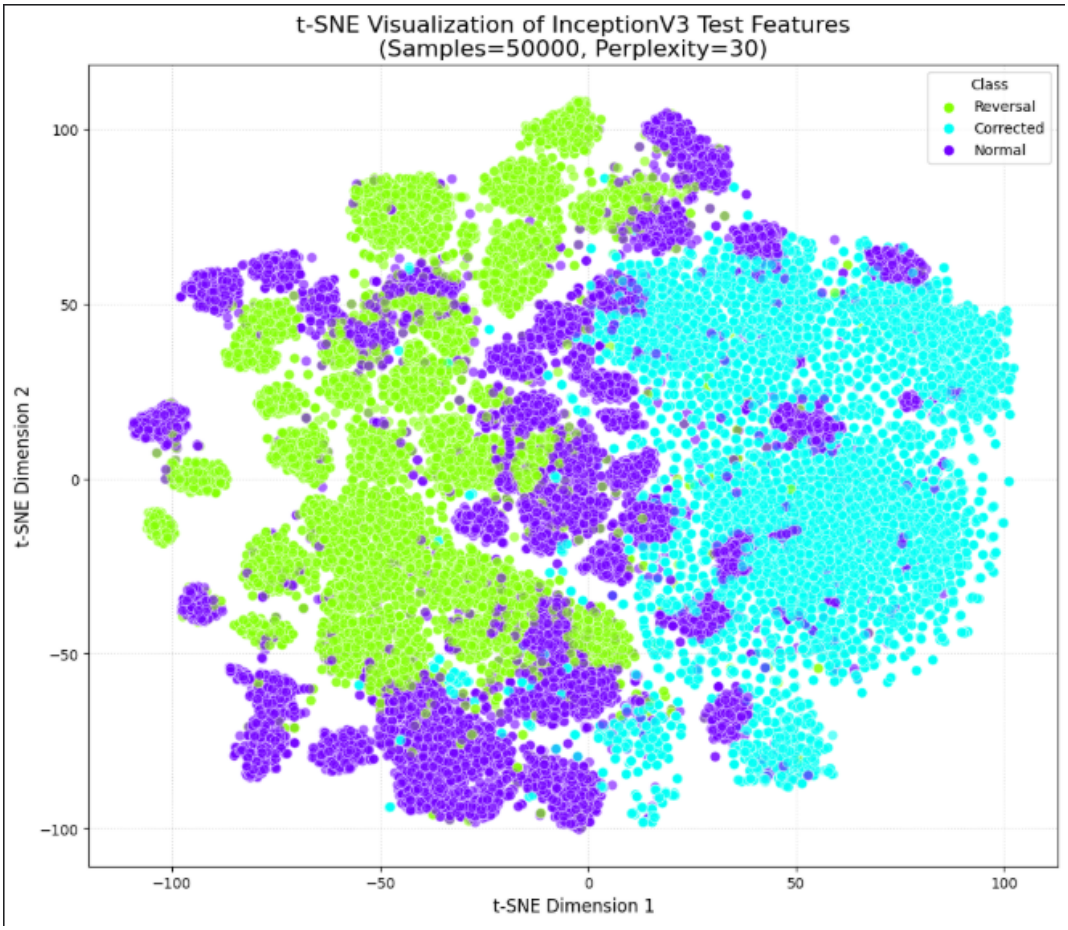


Figure 4.6: t-SNE Visualization of Fused ResNet50V2 & InceptionV3 Test Features (*Gambo* 3-Class Task).

The comparative results presented in Table 4.1 and the detailed reports (Tables 4.4 to 4.5) indicate that features extracted from the ResNet50V2 and InceptionV3 backbones yielded the most balanced and highest overall performance when used with a Logistic Regression classifier for the 3-class Gambo letter task. Features from MobileNetV2 and DenseNet121 showed comparatively lower discriminative power for this specific task and classifier. This suggests that the hierarchical representations learned by ResNet50V2 and InceptionV3 were more effective for distinguishing between Normal, Reversal, and Corrected letter forms in this context.

4.2.2 Evaluation of Exploratory VLM (CLIP & BEiT) Features on *Gambo* Dataset

Following the feature extraction procedures outlined in §3.3.4, the features derived from the pre-trained CLIP and BEiT models were evaluated on the 3-class Gambo letter classification task. For a standardized comparison across different feature sets, Logistic

Regression was a primary classifier of interest, though other classifiers including Linear SVM, Random Forest, and a PyTorch-based Multi-Layer Perceptron (MLP) were also investigated.

CLIP-Derived Features

Features extracted using the frozen `openai/clip-vit-base-patch32` vision model (768-dimensional) were scaled and then used to train various classifiers. The performance of the Logistic Regression model, after hyperparameter tuning on these CLIP features is detailed in Table 4.6.

Table 4.6: Classification Report for Tuned Logistic Regression on CLIP Features (*Gambo* 3-Class Task)

Class	Precision	Recall	F1-Score
Normal	0.78	0.81	0.79
Reversal	0.84	0.79	0.81
Corrected	0.90	0.92	0.91
Accuracy			0.84
Macro Avg	0.84	0.84	0.84
Weighted Avg	0.84	0.84	0.84

The confusion matrix for the Logistic Regression classifier on CLIP features is shown in Figure 4.7.

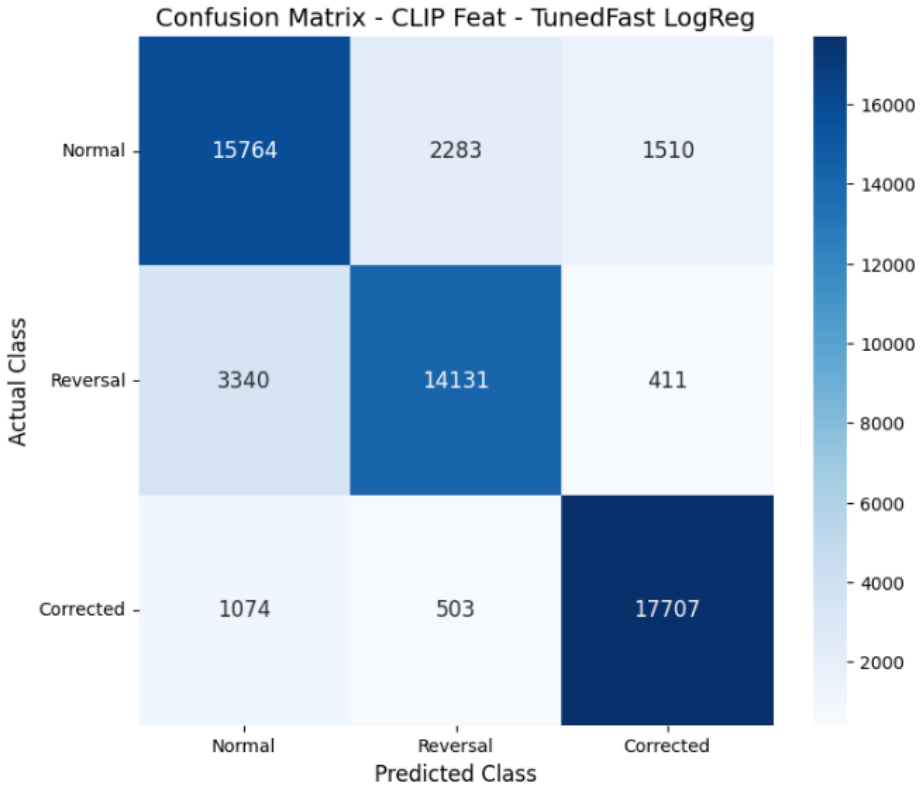


Figure 4.7: Confusion Matrix for Tuned Logistic Regression on CLIP Features (*Gambo* 3-Class Task).

Other classifiers, including Linear SVM, Random Forest, and a PyTorch MLP, were also evaluated on the CLIP features. The PyTorch MLP (input dimension 768, hidden dimension 128, dropout 0.3, trained with AdamW) showed strong performance. A summary comparison of classifiers on CLIP features is presented in Table 4.7.

Table 4.7: Performance Summary of Classifiers on CLIP-Derived Features (*Gambo* 3-Class Task)

Classifier	Accuracy	Weighted F1-Score
PyTorch MLP	0.8748	0.8752
Linear SVM	0.8395	0.8393
LogReg	0.8392	0.8390
Random Forest	0.8213	0.8188

BEiT-Derived Features

Similarly, features extracted using the frozen `microsoft/beit-base-patch16-224-pt22k-ft22k` model (768-dimensional)

were evaluated. The performance of the Logistic Regression model, after hyperparameter tuning (best parameters: C≈0.156, penalty='l2'), on these BEiT features is detailed in Table 4.8.

Table 4.8: Classification Report for Tuned Logistic Regression on BEiT Features (*Gambo* 3-Class Task)

Class	Precision	Recall	F1-Score
Normal	0.73	0.86	0.79
Reversal	0.81	0.68	0.74
Corrected	0.95	0.91	0.93
Accuracy			0.82
Macro Avg	0.83	0.81	0.82
Weighted Avg	0.83	0.82	0.82

The confusion matrix for the Logistic Regression classifier on BEiT features is shown in Figure 4.8.

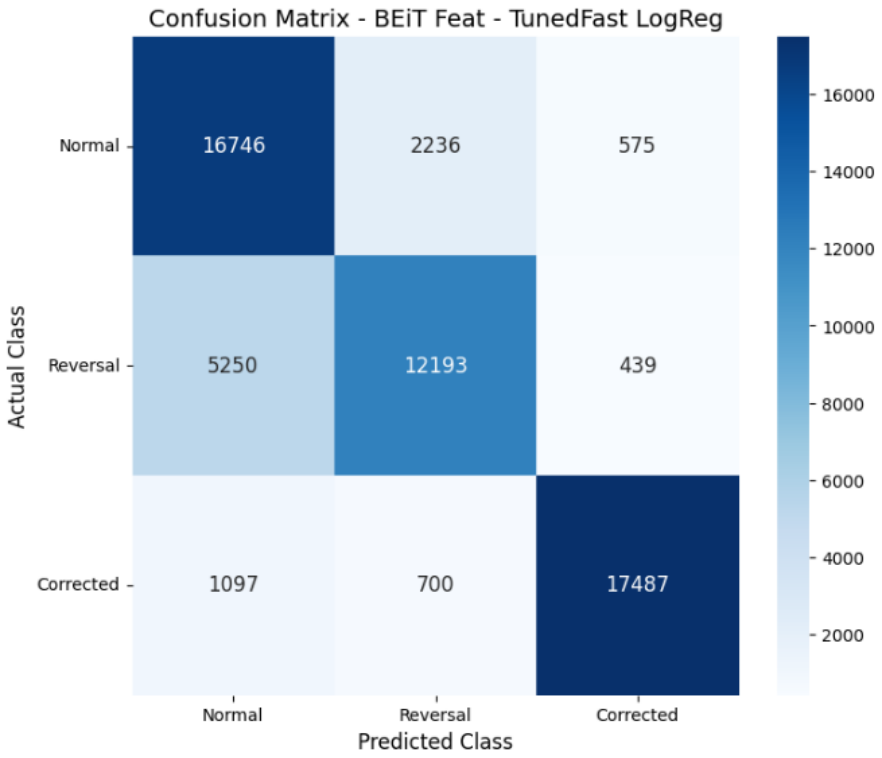


Figure 4.8: Confusion Matrix for Tuned Logistic Regression on BEiT Features (*Gambo* 3-Class Task).

As with CLIP features, other classifiers were also tested. The PyTorch MLP also showed strong performance on BEiT features. A summary comparison of classifiers on BEiT features is presented in Table 4.9.

Table 4.9: Performance Summary of Classifiers on BEiT-Derived Features (*Gambo* 3-Class Task)

Classifier	Accuracy	Weighted F1-Score
PyTorch MLP	0.8496	0.8501
LogReg	0.8185	0.8183
Linear SVM	0.8182	0.8181
Random Forest	0.7510	0.7479

The training history for the PyTorch MLP on BEiT features is illustrated in Figure **??**.

A brief comparative discussion regarding the utility of these VLM-derived features (CLIP and BEiT) in contrast to the fused CNN features (§4.2.3). Generally, while these pre-trained VLMs provide strong general-purpose visual features, their direct application or simple linear probing on this specific letter characteristic task showed varied performance compared to features derived from CNNs that were more directly adapted to the Gambo dataset.

4.2.3 Evaluation of Fused ResNet50V2 and InceptionV3 Features on *Gambo* Dataset

Following the extraction and fusion process detailed in §3.3.3, the resulting 4096-dimensional features (concatenation of ResNet50V2 and InceptionV3 features) were evaluated on the 3-class Gambo letter classification task. This evaluation aimed to validate the discriminative power of these fused features before their application in the dyslexia risk prediction pipeline. Several classical machine learning classifiers were trained on the scaled fused features.

Table 4.10 provides a comparative summary of the performance achieved by these classifiers on the Gambo test set using the 4096D fused features.

Table 4.10: Performance Summary of Classical Classifiers on Fused ResNet50V2 & InceptionV3 Features (*Gambo* 3-Class Task)

Classifier	Accuracy	Weighted F1-Score
RBFSVM	**0.9413**	**0.9416**
LogReg	**0.9385**	**0.9390**
MLP (sklearn)	0.9384	0.9389
LinearSVM	0.9365	0.9370
RandomForest	0.9339	0.9339

The RBF SVM classifier demonstrated the highest overall accuracy and weighted F1-score when trained on the fused features. Detailed classification reports for each classifier provide further insight into per-class performance. For instance, the Logistic Regression model achieved a weighted F1-score of 0.9117, with its performance across the 'Normal', 'Reversal', and 'Corrected' classes detailed in Table 4.11.

Table 4.11: Classification Report for Logistic Regression on Fused ResNet50V2 & InceptionV3 Features (*Gambo* 3-Class Task)

Class	Precision	Recall	F1-Score
Normal	0.88	0.92	0.90
Reversal	0.95	0.92	0.93
Corrected	0.97	0.96	0.97
Accuracy			0.93
Macro Avg	0.93	0.93	0.93
Weighted Avg	0.93	0.93	0.93

The confusion matrix for the Logistic Regression classifier using these fused features is presented in Figure 4.9. Similar detailed reports and confusion matrices were generated for all tested classifiers (Linear SVM, RBF SVM, Random Forest, MLP), with the RBF SVM generally showing the most favorable metrics.

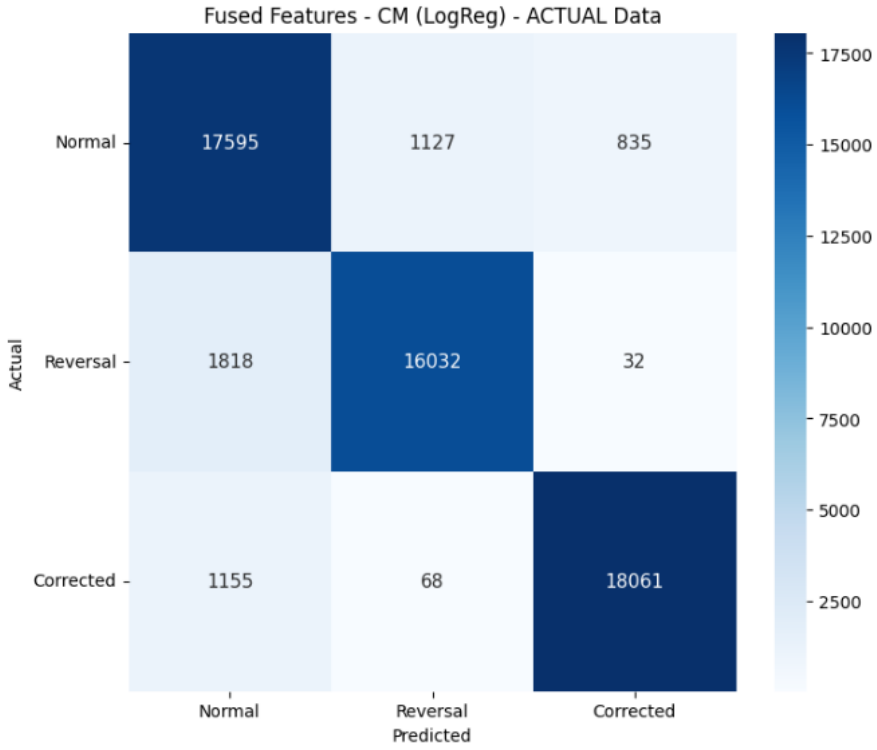


Figure 4.9: Confusion Matrix for Logistic Regression on Fused ResNet50V2 & InceptionV3 Features (*Gambo* 3-Class Task).

Additionally, a t-SNE visualization of the scaled 4096-dimensional fused test features was generated to understand their separability in a lower-dimensional space, as shown in Figure 4.10. The visualization suggests a reasonable degree of clustering according to the three letter categories, supporting the quantitative classification results.

Figure 4.10: t-SNE Visualization of Fused ResNet50V2 & InceptionV3 Test Features (*Gambo* 3-Class Task).

The strong performance of various classifiers on these 4096D fused features, particularly the RBF SVM and Logistic Regression models, validated their discriminative capability for classifying handwriting characteristics. Consequently, these fused features were selected as the primary feature set from the handwriting modality for the subsequent dyslexia risk prediction task.

4.3 3-Stage proposed model results break down

This section details the outcomes of experiments related to analyzing handwriting, encompassing both global style assessment from paragraph/page images and feature learning from isolated letters.

4.3.1 Stage 1: Global Handwriting Style Assessment using CLIP

The initial layer of the proposed handwriting pipeline aimed to assess the global style of handwriting samples using CLIP embeddings, as described in §3.4.2. This involved

comparing paragraph/page images from Dyslexic Children (D), Normal Children (NC1), and the IAM dataset against reference style embeddings.

Off-the-Shelf CLIP Embedding Analysis

Using the pre-trained `openai/clip-vit-base-patch32` model, image embeddings were extracted.

- Similarity to Average IAM Style: The cosine similarity scores of D and NC1 samples were calculated against an "Average IAM Style Embedding" (derived from 1539 IAM samples). Figure 4.11 presents the distribution of these similarity scores.

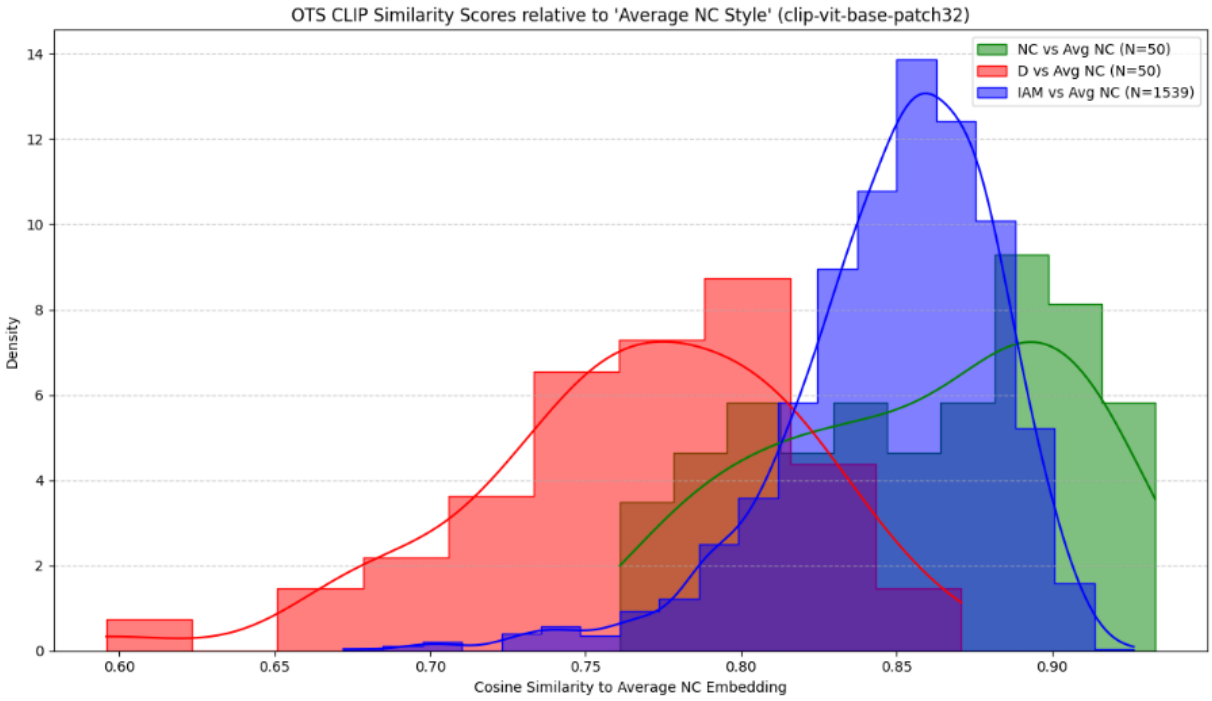


Figure 4.11: Distribution of Off-the-Shelf CLIP Similarity Scores relative to 'Average IAM Style' for Dyslexic (D) in diagram represented by Red, Normal Children (NC) represented as Green, and IAM sample groups represented as Blue.

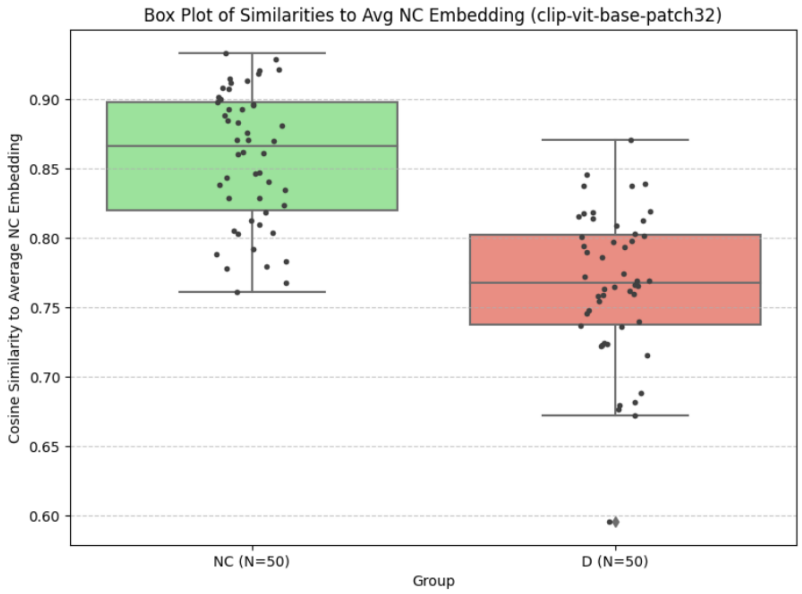
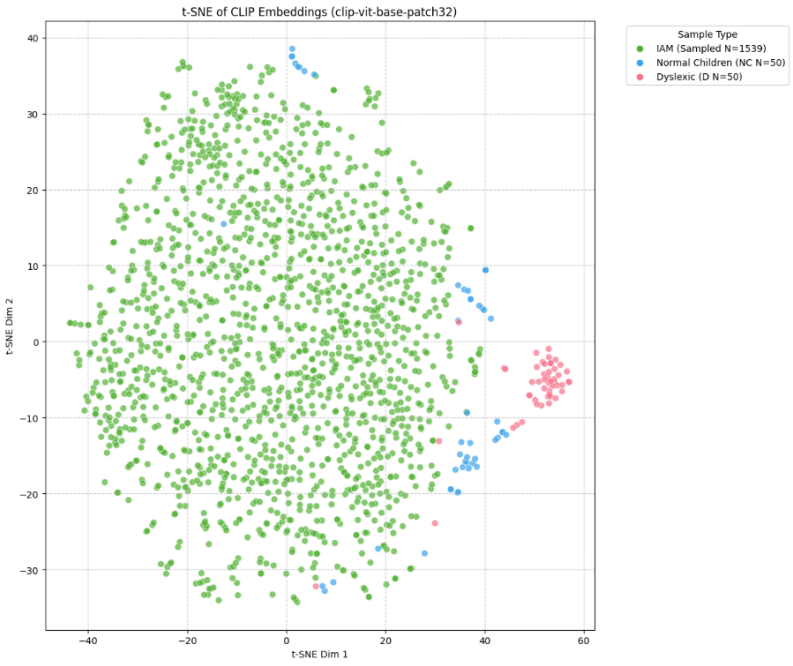


Figure 4.12: Box Plot of Off-the-Shelf CLIP Similarities to 'Average NC Combined Style' for Dyslexic (D) and Normal Children (NC) groups.

- Embedding Space Visualization (Off-the-Shelf CLIP): t-SNE and UMAP visualizations of the off-the-shelf CLIP embeddings for D, NC, and sampled IAM images are presented in Figure 4.13.



(a) t-SNE Visualization with IAM represented as Green, NC represented as Blue and D represented as Red

(b) UMAP Visualization

Figure 4.13: Dimensionality Reduction Visualizations (t-SNE and UMAP) of Off-the-Shelf CLIP Image Embeddings for IAM, Normal Children (NC), and Dyslexic (D) samples.

- Preliminary Thresholding Discussion (Off-the-Shelf CLIP): Based on the distribution of similarity scores of NC samples relative to the "Average NC Combined Style", a preliminary threshold (NC mean - 1.5 * std dev ≈ 0.7861) was determined. Applying this threshold:
  - Normal Children (NC) samples flagged as 'atypical': 10.00% (5 out of 50).
  - Dyslexic (D) samples flagged as 'atypical': 58.00% (29 out of 50).

  This indicates some potential for the off-the-shelf CLIP style features to differentiate between the groups, although with notable overlap.

Fine-Tuned CLIP Vision Model Analysis

The CLIP vision model was fine-tuned on a binary task (Dyslexic vs. Normal). Embeddings were then re-extracted using this fine-tuned model.

- Fine-Tuning Performance Summary: The fine-tuning process involved 5-fold cross-validation, achieving an average best validation accuracy of 97.00% across folds. The final model trained on combined data for 10 epochs was used for subsequent embedding extraction.
- Similarity to Average Fine-Tuned NC Combined Style: Figure 4.14 and Figure 4.15 show the distributions and boxplots of D, NC1, NC2, and IAM sample similarities (using fine-tuned embeddings) against an "Average NC Combined Style Embedding" derived from the fine-tuned NC embeddings.

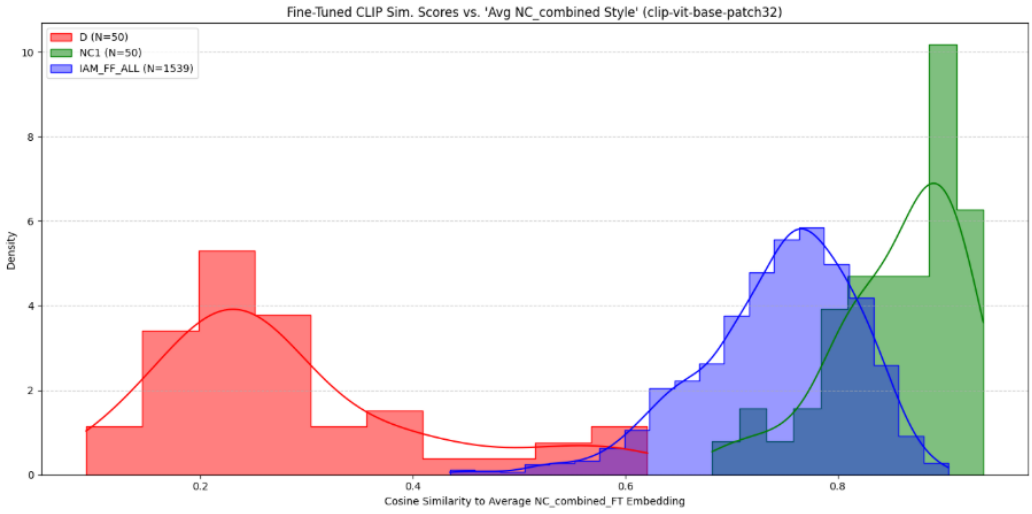


Figure 4.14: Distribution of Fine-Tuned CLIP Similarity Scores relative to 'Average Fine-Tuned NC Combined Style'.

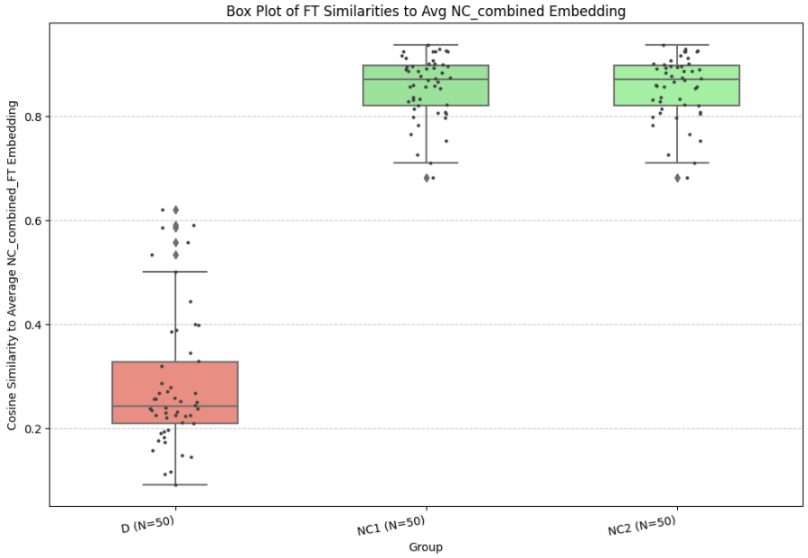
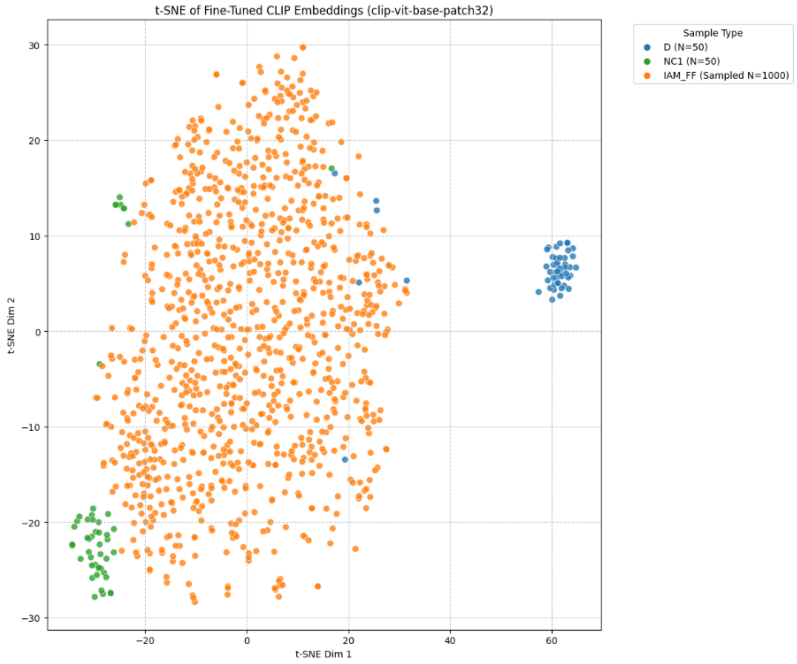
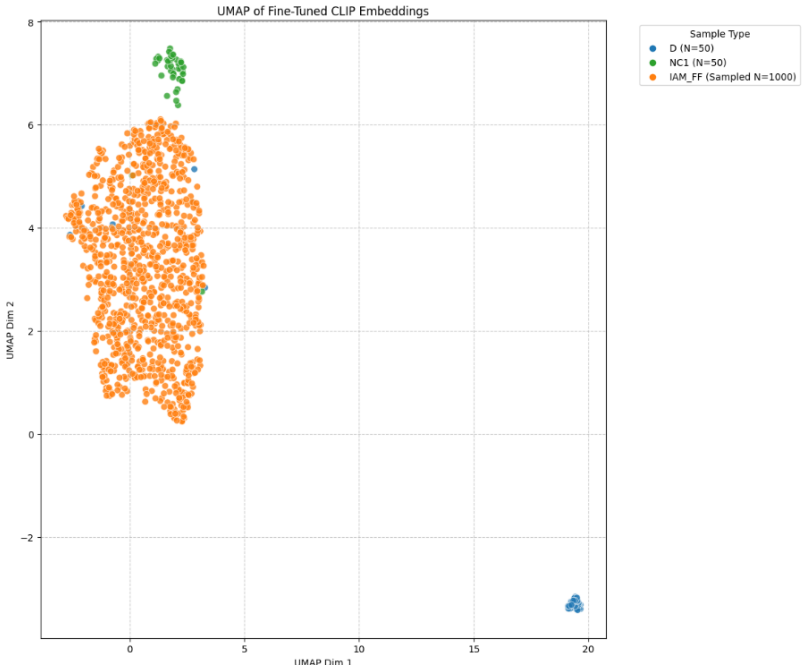


Figure 4.15: Box Plot of Fine-Tuned CLIP Similarities to 'Average Fine-Tuned NC Combined Style' for Dyslexic (D) and Normal Children (NC1, NC2) groups.

- Embedding Space Visualization (Fine-Tuned CLIP): t-SNE and UMAP visualizations of the fine-tuned CLIP embeddings are presented in Figure 4.16.



(a) t-SNE Visualization (Fine-Tuned) with IAM represented as Orange, NC represented as green and D represented as Blue



(b) UMAP Visualization (Fine-Tuned)

Figure 4.16: Dimensionality Reduction Visualizations (t-SNE and UMAP) of Fine-Tuned CLIP Image Embeddings for IAM, Normal Children (NC), and Dyslexic (D) samples.

- Preliminary Thresholding Discussion (Fine-Tuned CLIP): Using a threshold based on the distribution of fine-tuned NC Combined scores relative to their own average (NC Combined mean - 1.5 * std dev ≈ 0.7675):
  - Normal Children samples flagged as 'atypical': 5%.
  - Dyslexic (D) samples flagged as 'atypical': 100.00% (50 out of 50).

  This suggests that fine-tuning the CLIP vision model specifically for discriminating D vs. NC styles significantly improved the separation between these groups based on this global style atypicality score. Example predictions from the fine-tuned classifier are shown in Figure 4.17.

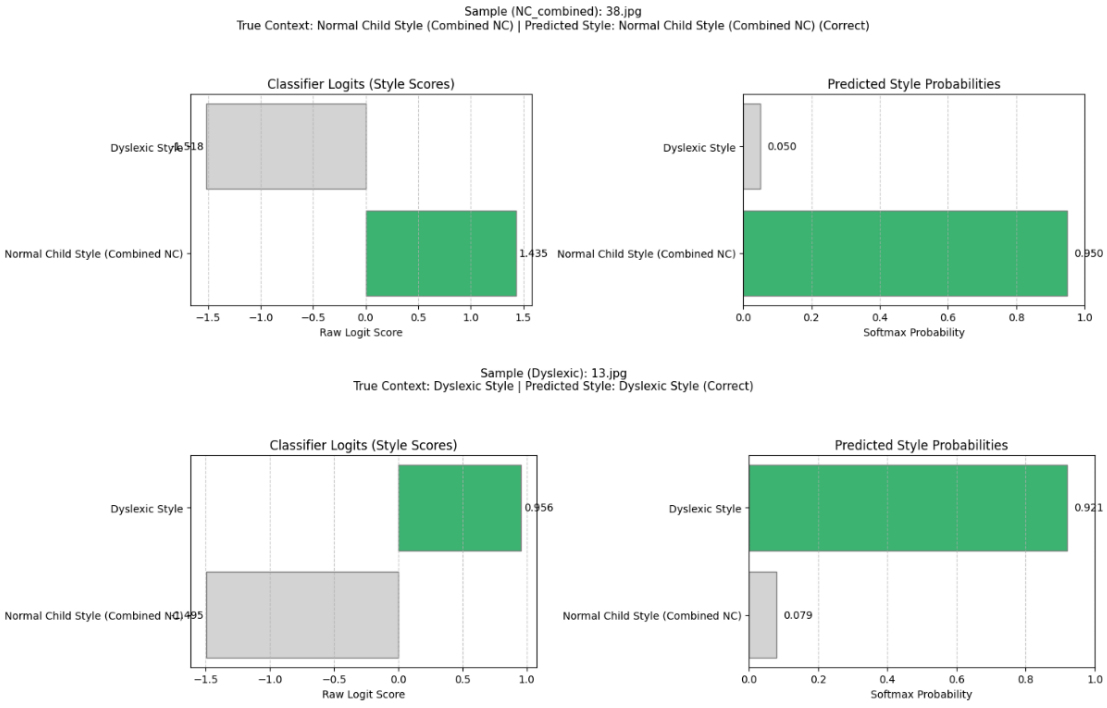


Figure 4.17: Example Individual Sample Prediction using the Fine-Tuned CLIP Vision Classifier Head (Dyslexic Sample Correctly Identified as 'Dyslexic Style').

4.3.2 Stage 2: Outcomes of VLM-Based Line-Level HTR and Anomaly Detection (OpenAI GPT-4o)

The OpenAI GPT-4o model was employed for line-by-line analysis of handwriting samples from IAM, Dyslexic (D), and Normal Children (NC1) groups, focusing on transcription, spelling, and visual anomaly detection).

- Qualitative Analysis of JSON Outputs: Review of the VLM's structured JSON output for sample lines indicated its capability to transcribe text, identify mis-

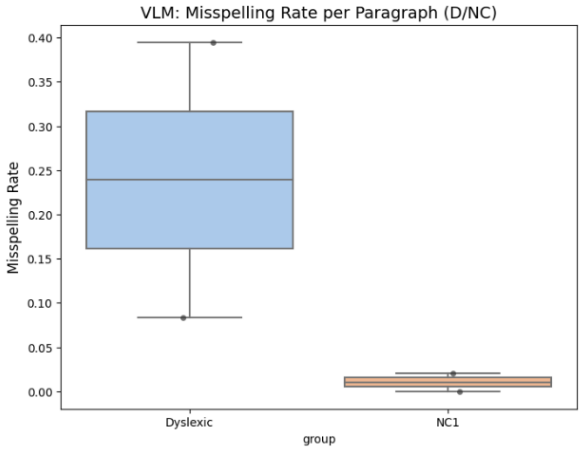
spellings with suggested corrections, and describe visual anomalies (e.g., "The 'd' at position 0 visually resembles a 'b'").

- Aggregated Statistics: Table 4.12 summarizes key metrics derived from the VLM's analysis across the different groups.

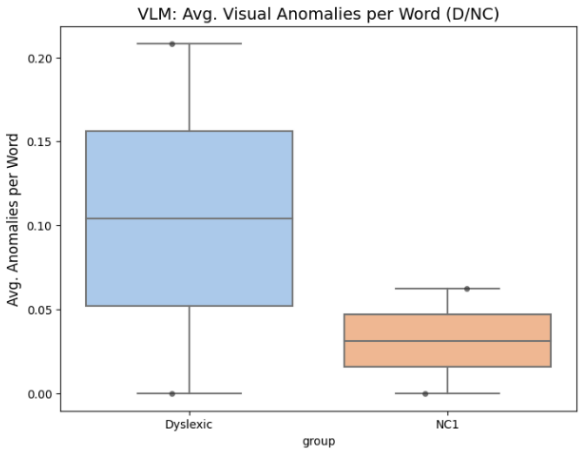
Note: Metrics derived from VLM's automated analysis. 'Misspell Rate' = Total Misspellings / Words Analyzed. 'Average Anomalies/word' = Number of reversed letters in each word

Group	Words	Misspell Rate	Total Misspell.	Avg Anomalies/Word
Dyslexic	62.000	0.274	17.000	0.081
IAM Control	38.000	0.000	0.000	0.000
NC1	112.000	0.009	1.000	0.027

- Boxplots illustrating the distribution of misspelling rates and average visual anomalies per paragraph for the Dyslexic and NC1 groups are shown in Figure 4.18.



(a) VLM: Misspelling Rate per Paragraph



(b) VLM: Avg. Visual Anomalies per Word

Figure 4.18: Box Plots from OpenAI GPT-4o Word-Level Analysis for Dyslexic (D) and Normal Children (NC1) Paragraphs.

4.3.3 Stage 3 Component: Fine-Grained Letter-Level Visual Analysis Results

This section presents outcomes related to Stage 3 of the proposed handwriting pipeline (Figure 3.1), which focuses on fine-grained visual analysis of individual letters. This involves:

1. The performance of the letter segmentation and preparation pipeline when applied to full-page handwriting samples.
2. The classification performance of the selected letter fused classifier on these segmented letters.

The foundational experiments validating different feature extraction approaches for isolated letter classification on the Gambo dataset are summarized first, as they inform the choice of the classifier used in this layer.

Validation of Letter Classifiers on *Gambo* Dataset (Informing Layer 3 Component)

As detailed in the methodology (§3.4.1), various feature sets (from individual CNN backbones, fused CNNs, and exploratory VLMs) were evaluated for the 3-class letter classification task (Normal, Reversal, Corrected) on the Gambo dataset.

- Features from individual CNN backbones (ResNet50V2, InceptionV3, MobileNetV2, DenseNet121) were tested with Logistic Regression. ResNet50V2 (Accuracy: 0.927, Weighted F1: 0.929) and InceptionV3 (Accuracy: 0.9149, Weighted F1: 0.9151) features provided superior results compared to MobileNetV2 and DenseNet121 features (refer to Table 4.1 in §4.2.1 for details).
- The 4096-dimensional features, fused from ResNet50V2 and InceptionV3, yielded strong performance when used with various classical classifiers. Notably, RBF SVM achieved an accuracy of 0.9413, while Logistic Regression achieved 0.9385 accuracy (refer to Table 4.10 in §4.2.3.

Based on these comprehensive evaluations on the Gambo dataset, a classifier trained on the 4096D fused ResNet50V2 and InceptionV3 features was considered the most robust component for classifying isolated letters in Stage 3. The following results demonstrate this component's application to letters segmented from full page images.

Application of Segmentation and Letter Classification to Full Page Samples (Dyslexic Dataset)

The image processing pipeline for full-page preprocessing, line segmentation, character segmentation, and individual letter preparation (as detailed in §3.4.4) was applied to samples from the "DyslexicH" dataset. Each segmented character was then classified using the selected fused letter classification model.

Figure 4.19 illustrates the flow of the image processing/segmentation pipeline for Stage 3.

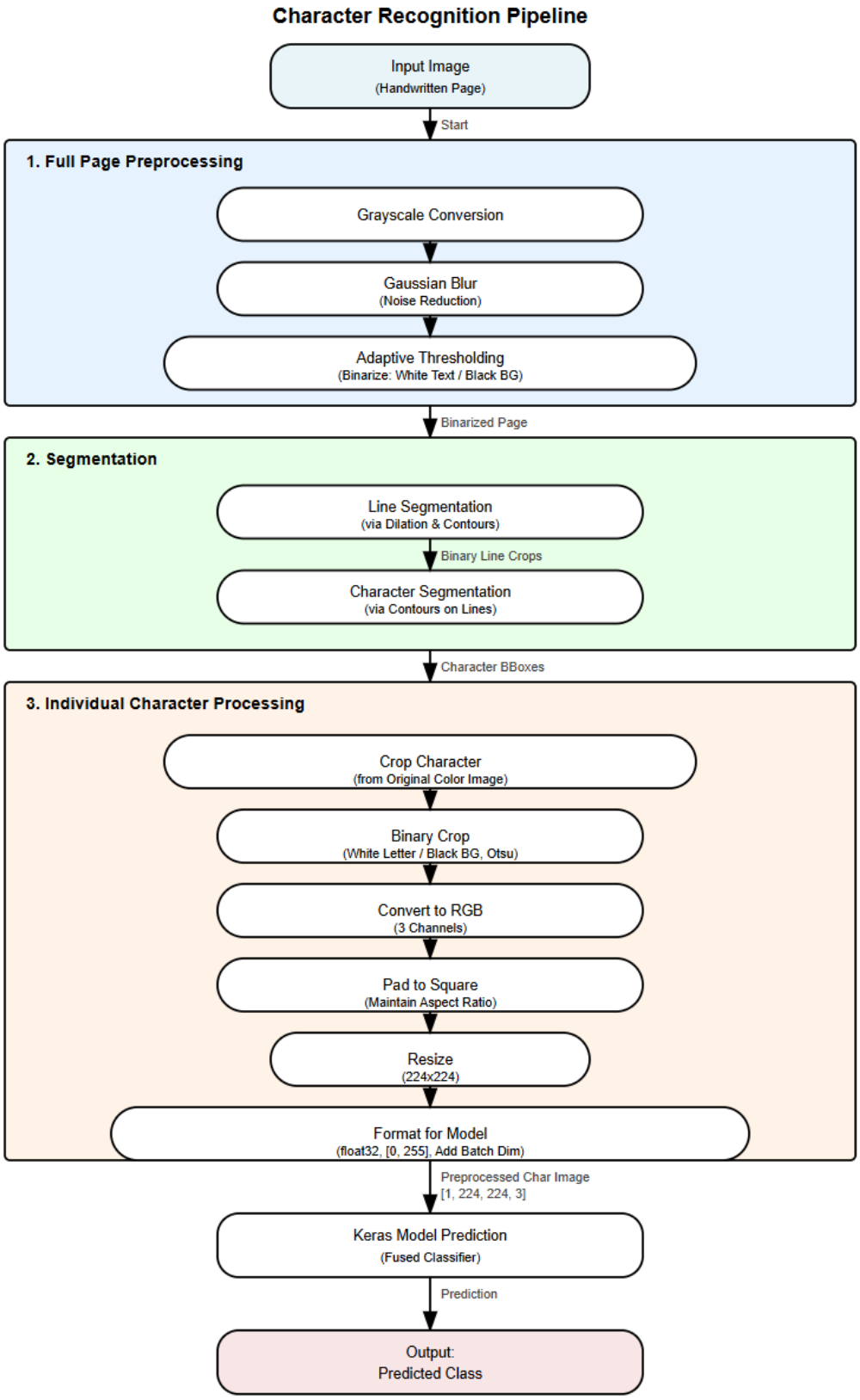


Figure 4.19: Diagram of the Image Processing and Letter Classification Pipeline for Layer 3, demonstrating steps from input page to individual letter classification.

The results of this process on sample images from the DyslexicH dataset show the model's predictions for individual segmented characters. Figure 4.20 displays a sample image from the DyslexicH dataset, and Figure 4.21 shows these predictions overlaid on the original handwriting sample using bounding boxes (Green box = 'Normal class', Yellow box = 'Reversal class', Red box = 'Corrected class').

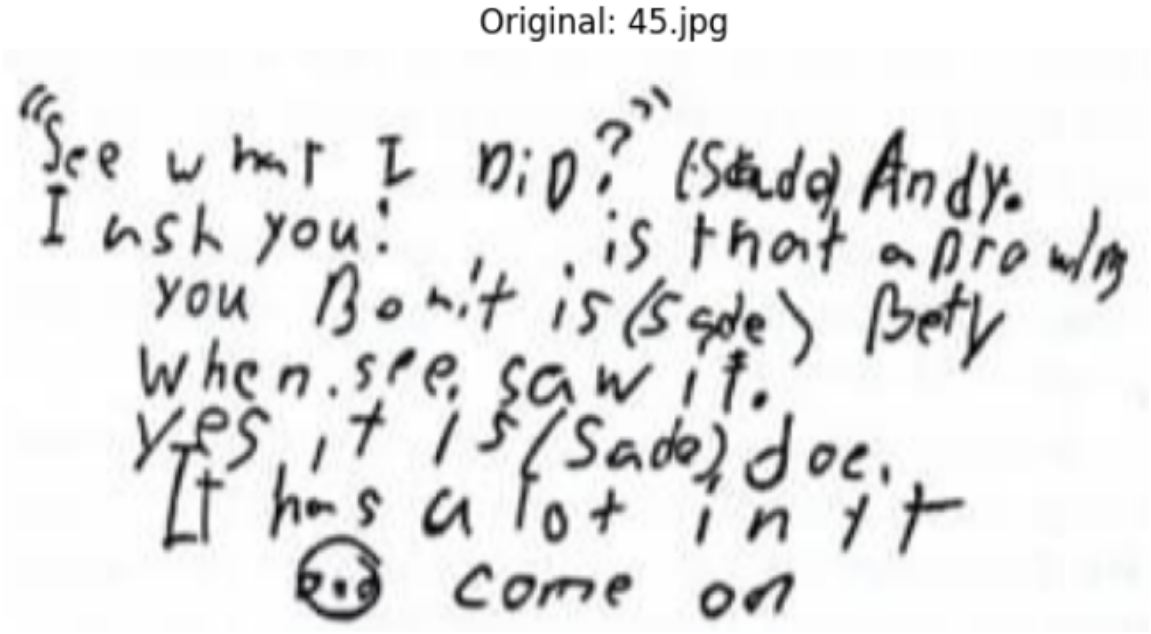


Figure 4.20: DyslexicH Sample without any predictions from the Letter Classifier.

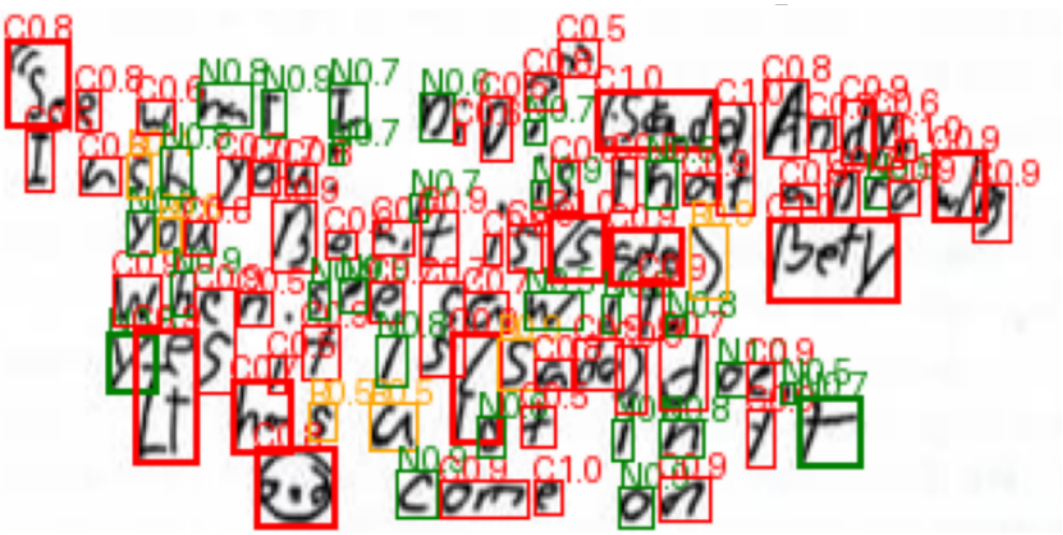


Figure 4.21: Example Overlay of Predicted Letter Classes on an Original Handwriting Sample from the DyslexicH Dataset for Stage 3 Analysis. With colored bounding boxes depending on classification: Normal='Green', Corrected='Red', Yellow='Reversal'

An aggregation of the predictions across multiple processed images from the DyslexicH dataset provides an overview of the types of letter characteristics identified by the Stage 3 component. Table 4.13 shows the distribution of predicted classes ('Normal', 'Reversal', 'Corrected') for all characters segmented and classified from the processed DyslexicH samples.

Note: Based on applying the segmentation pipeline and the Fused letter classifier to DyslexicH samples.

Predicted Class	Count (Proportion)
Normal	769 (23.13%)
Reversal	349 (10.50%)
Corrected	2207 (66.38%)
Total Characters Classified	3325
Total Images Processed	43
Avg. Chars/Image	77.33

The average confidence of these predictions was approximately 0.806, with a median confidence of 0.871. This indicates that the model was generally confident in its classifications of the segmented letters. These per-letter classifications provide the fine-grained visual error information intended as output from Stage 3.

4.4 Eye-Tracking Analysis: Autoencoder-Based Pipeline for Dyslexia Risk

This section details the results of the primary eye-tracking pipeline, which utilizes an Autoencoder (AE) trained on gaze-path images derived from the Czech dataset to extract features for dyslexia risk classification. The performance on the Czech dataset is presented, followed by the outcomes of applying this pipeline to the English GazeBase dataset.

4.4.1 Autoencoder Training and Feature Extraction (Czech Dataset)

As described in §3.5.2, a Convolutional Autoencoder was trained exclusively on gaze-path images generated from the preprocessed Czech eye-tracking data. The training aimed to minimize reconstruction error (Binary Cross Entropy loss).

- The AE model training converged, achieving a best validation loss of approximately 0.0134 after 48 epochs (out of a maximum of 50, with early stopping patience of 10).
- Using the trained AE, reconstruction error features (5 range-based error metrics, as detailed in §3.5.2) were extracted for each trial in the Czech dataset.

4.4.2 Classifier Performance on AE Reconstruction Error Features (Czech Dataset)

The extracted 5-dimensional reconstruction error features from the Czech dataset were then used to train and evaluate several classical machine learning classifiers for binary dyslexia risk prediction. Subject-stratified 5-fold cross-validation and GridSearchCV were employed to select the best model and hyperparameters, optimizing for F1-score.

Table 4.14 summarizes the cross-validated performance of the tested classifiers. The Support Vector Classifier (SVC) yielded the best F1-score.

Note: CV F1-Score is the best score from GridSearchCV. Accuracy, Precision, Recall, and AUC are indicative averages from cross-validation with the best parameters. Actual values should be reported from detailed CV analysis.

Classifier	F1-Score	Accuracy	Precision
SVC	0.7667	0.7919	0.7886
RandomForest	0.7502	0.7811	0.7795
LogisticRegression	0.7351	0.7432	0.7310

The performance of the best selected classifier (SVC) on the full Czech training data is shown in the classification report in Table 4.15

Table 4.15: Classification Report for Best AE-Feature Classifier (SVC) on Full Czech Training Data

Class	Precision	Recall	F1-Score
Low-Risk (0)	0.80	0.71	0.75
High-Risk (1)	0.79	0.86	0.82
Accuracy			0.79
Macro Avg	0.79	0.79	0.79
Weighted Avg	0.79	0.79	0.79

4.4.3 Cross-Lingual Application of AE-Pipeline to English *GazeBase* Dataset

The complete Czech-trained AE pipeline (including the fitted feature scaler and the best SVC classifier from §4.4.2) was applied to the reconstruction error features extracted from the unlabeled English GazeBase dataset. The objective was to observe the distribution of predicted dyslexia risk labels in this different linguistic and demographic context.

Out of the 1762 English GazeBase trials processed, approximately 61.7% were predicted as 'Low Risk' and 38.3% as 'High Risk'. The interpretation of this distribution is approached with caution due to the absence of ground truth dyslexia labels for the GazeBase dataset and potential domain shift effects.

4.5 Chapter Summary

This chapter presented a comprehensive set of results from various experiments targeting dyslexia risk assessment through handwriting and eye-tracking analysis. The key findings can be summarized as follows:

- Handwriting Feature Learning (Gambo Dataset):
  - Evaluation of individual CNN backbones (ResNet50V2, InceptionV3, MobileNetV2, DenseNet121) for 3-class letter characteristic classification on the Gambo dataset showed that features from ResNet50V2 (Acc: 0.927) and InceptionV3 (Acc: 0.915) yielded superior performance when used with a Logistic Regression classifier, compared to MobileNetV2 and DenseNet121.
  - Fusing features from ResNet50V2 and InceptionV3 (4096-dimensions) further enhanced performance, with classifiers like RBF SVM achieving high accuracy 0.9413 on the Gambo 3-class task, validating the discriminative power of these fused features.
  - Exploratory analysis of features from pre-trained Vision-Language Models (CLIP and BEiT) on the Gambo letter task indicated their potential, with a PyTorch MLP achieving accuracies of 0.8748 (CLIP) and 0.8496 (BEiT), though these did not surpass the specifically adapted fused CNN features for this isolated letter task.
- 3-Stage Handwriting Pipeline for Dyslexia Risk Assessment:
  - Stage 1 (Global Style Assessment): Off-the-shelf CLIP embeddings showed some capability to distinguish between Dyslexic, Normal Children (NC), and IAM handwriting styles. Fine-tuning the CLIP vision model on a D vs. NC task significantly improved the separation, with a preliminary threshold flagging 100% of Dyslexic samples and only 5% of NC samples as atypical based on style similarity to an average NC profile.
  - Stage 2 (VLM Line-Level Analysis): The OpenAI GPT-4o model demonstrated its ability to perform line-level transcription, identify misspellings, and detect visual anomalies in handwriting. Aggregated results indicated higher misspelling rates (27.4%) and visual anomaly counts for the Dyslexic group compared to NC1 and IAM controls.
  - Stage 3 (Letter Classification Application): Application of the segmentation pipeline and the Gambo-trained letter classifier (e.g., ResNet50V2-based) to DyslexicH samples successfully classified individual segmented

characters into Normal, Reversal, and Corrected categories with reasonable confidence (average 0.806), providing fine-grained error data.

- Eye-Tracking Cross-Lingual Model (AE-Based):
  - The Autoencoder trained on Czech gaze-path images successfully learned to reconstruct these images, enabling the extraction of reconstruction error features.
  - On the Czech dataset, an SVC classifier trained on these AE-derived error features achieved a cross-validated F1-score of approximately 0.6667 for dyslexia risk prediction.
  - Application of the Czech-trained AE pipeline and SVC classifier to the unlabeled English GazeBase dataset resulted in approximately 61.7% of trials being predicted as 'High Risk', providing an initial insight into cross-lingual application, albeit without ground truth for direct validation on this target domain.
  - The exploratory BiLSTM sequence model, when trained from scratch on the Czech data, served as an ET baseline, achieving approximately 79% accuracy.

These findings from the distinct handwriting and eye-tracking analysis pipelines provide valuable insights into the potential of AI-driven approaches for identifying indicators associated with dyslexia risk. A detailed interpretation and discussion of these results in the broader context follows.

Chapter 5

Conclusion

This chapter provides a comprehensive discussion of the results presented in Chapter 4. The findings from the handwriting analysis experiments, including the foundational feature learning on the Gambo dataset and the application of the proposed 3-Stage AI Pipeline, are interpreted. Similarly, the outcomes of the eye-tracking cross-lingual model development are analyzed. The implications of these results are considered in the context of the research objectives and existing literature. The chapter concludes by addressing the limitations of the current study and outlining promising avenues for future research.

5.1 Interpretation of Handwriting Analysis Results

The investigation into handwriting analysis yielded several key insights, progressing from foundational letter-level feature learning to a multi-stage pipeline for dyslexia risk assessment.

5.1.1 Effectiveness of Feature Learning from Isolated Letters (*Gambo* Dataset)

The initial experiments focused on the Gambo dataset demonstrated the viability of using deep learning models to extract discriminative features for classifying isolated letter characteristics (Normal, Reversal, Corrected). The superior performance of features derived from deeper CNN architectures like ResNet50V2 and InceptionV3, especially when fused, highlighted their capacity to capture intricate visual patterns relevant to letter formation. For instance, the fused 4096D features enabled classical classifiers like RBF SVM to achieve high accuracy on this 3-class task. This aligns with trends in handwriting analysis where robust feature representations are critical [6], [13]. While Vision-Language Models like CLIP and BEiT offered strong general visual embeddings, their performance on this specific, fine-grained letter classification task, when used as

frozen feature extractors, did not surpass the CNNs that were more directly adapted (via head training or fine-tuning principles) to the letter image domain. This suggests that for highly specific visual tasks like isolated letter analysis, specialized or domain-adapted vision models may hold an advantage over more general VLM embeddings without task-specific fine-tuning of the VLM itself. The strong performance on the Gambo task provided confidence in using these types of feature extractors as a component for more detailed letter analysis within a broader dyslexia screening context (i.e., Stage 3).

5.1.2 Insights from the Proposed 3-Stage Handwriting Pipeline

The development and component-wise exploration of the 3-Stage AI pipeline offered a more holistic approach to analyzing handwriting for dyslexia risk indicators.

- Stage 1 (Global Style Assessment): The use of CLIP embeddings for global style assessment revealed a discernible difference in the overall visual style between handwriting samples from dyslexic children and normal controls, particularly when compared against an "average Normal Children (NC) style" reference. The significant improvement in group separation after fine-tuning the CLIP vision model specifically for discriminating D vs. NC styles underscores the benefit of task-specific adaptation even for powerful foundation models. This layer's ability to flag "atypical" global styles serves as a promising initial filter in a tiered screening process.
- Stage 2 (Contextual Word-Level Analysis): The exploration with OpenAI's GPT-4o on pre-segmented line images highlighted the significant potential of advanced Vision-Language Models (VLMs) for integrated handwriting analysis. The VLM demonstrated a promising capability to concurrently perform Handwritten Text Recognition (HTR), identify spelling errors with suggested corrections, and detect fine-grained visual anomalies (such as letter reversals or misformations) at both word and letter levels within these lines. Notably, the VLM reported a higher incidence of spelling errors and visual anomalies in samples from the dyslexic group compared to controls, a finding consistent with established dyslexia characteristics and underscoring the potential utility of this analytical layer. However, attempts to apply the VLM for similar detailed analysis directly on whole sentences or paragraphs resulted in inaccurate outputs and apparent "hallucinations," suggesting that while powerful for contextual linguistic understanding, its current application for this task is more effective on

pre-segmented units rather than complex, dense image processing of entire text blocks. Despite this limitation, the line-level analysis successfully moved towards a deeper understanding of linguistic and visual error patterns within a contextual framework.

- Stage 3 (Fine-Grained Letter Analysis): The application of the Gambo-trained letter classifier (e.g., ResNet50V2-based) to characters segmented from full-page DyslexicH samples demonstrated the feasibility of identifying specific letter-level errors (Normal, Reversal, Corrected) in more naturalistic writing. The ability to pinpoint these issues with reasonable confidence provides valuable granular data. While the segmentation process itself presents challenges, successful character isolation allows for a detailed inspection that complements the higher-level analyses of Stages 1 and 2.

The 3-Stage pipeline offers a structured approach to decompose the complex task of handwriting analysis for dyslexia screening. It moves from a general impression of style to specific linguistic and visual errors, mirroring how a human expert might gradually refine their assessment. The modularity allows for future improvements in each stage independently.

5.2 Limitations of the Study

While this research provides valuable insights and novel methodological explorations, several limitations should be acknowledged:

- Dataset Constraints and Modality Independence: A primary limitation was the use of disparate datasets for different components of the study. The Gambo dataset (isolated letters) informed letter-level feature extractors, while different, smaller datasets (IAM, DyslexicH, NC) were used for paragraph-level global style (Layer 1) and VLM line analysis (Layer 2). The eye-tracking data (Czech, GazeBase) was entirely separate. This prevented the development of a truly end-to-end trained multimodal system on paired data from the same participants across all described handwriting stages and eye-tracking. Consequently, the 3-Stage handwriting pipeline's overall efficacy as an integrated dyslexia risk predictor remains conceptual, with its components validated individually.
- Handwriting 3-Stage Pipeline Integration: The mechanism for combining the outputs from the three distinct stages of the handwriting pipeline into a final dyslexia risk score was proposed conceptually but not implemented or evaluated. Determining appropriate weighting or fusion rules for these

diverse outputs (global style score, word-level error rates, letter-level classifications) is a complex task requiring further research.

- Layer 2 VLM Reliance and Scope: The exploration of Layer 2 functionalities relied on the powerful but API-based GPT-4o model. While showcasing potential, this approach has implications for cost, reproducibility, and control. Furthermore, the analysis was limited to pre-segmented lines rather than full paragraph processing with integrated segmentation, HTR, and linguistic analysis by the VLM.
- Letter Segmentation Robustness for Layer 3: The accuracy of the Layer 3 letter classification is highly dependent on the quality of the preceding character segmentation from full pages or words. While a functional pipeline was developed, handwriting segmentation is notoriously challenging, and errors here would propagate.
- Cross-Lingual Eye-Tracking Evaluation: The assessment of the ET model's cross-lingual performance on the GazeBase dataset was qualitative due to the lack of dyslexia labels. Quantitative validation on labeled English eye-tracking datasets is necessary to draw firm conclusions about its generalizability.
- Sample Sizes for Specific Groups: Some datasets used, particularly for the paragraph-level handwriting analysis (DyslexicH, NC groups with ∼50-100 samples), are relatively small for training robust deep learning models like CLIP fine-tuning, potentially limiting the generalizability of those specific findings.
- Scope of Dyslexia Indicators: This study focused on visual features from handwriting and eye movements. Dyslexia is a multifaceted condition with core phonological deficits that are not directly assessed by these modalities alone. The developed systems aim to identify risk indicators, not provide a clinical diagnosis.

5.3 Future Work

Building upon the findings and limitations of this study, several avenues for future research are evident:

- Integrated Dataset Collection: The most critical next step is the collection of a comprehensive, multimodal dataset featuring the same cohort of children, providing paired data across full handwriting samples

(paragraphs, words, letters), eye-tracking recordings during reading, and standardized dyslexia assessment scores. This would enable true end-to-end training and evaluation of integrated models.

- End-to-End Implementation and Evaluation of the 3-Stage Handwriting Pipeline: Future work should focus on developing and rigorously evaluating the integration mechanisms for the outputs of the three handwriting stages to produce a unified dyslexia risk score. This could involve exploring rule-based systems, machine learning meta-classifiers, or weighted fusion techniques.
- Advancements in Layer 2 and Layer 3 Handwriting Components:
  * Developing open-source, robust VLM-based solutions for integrated word/line segmentation, HTR, and linguistic analysis specifically tailored for children's handwriting from full paragraphs.
  * Improving character segmentation algorithms for Layer 3, potentially using deep learning-based object detection or instance segmentation models, to enhance the reliability of input to the letter classifier.
- Enhancing Eye-Tracking Models: Further exploration of different AE architectures, latent space feature utilization, and advanced sequence models (e.g., Transformers) for eye-tracking data. Rigorous testing on diverse, labeled multilingual datasets is crucial for validating cross-lingual approaches.
- Incorporation of Explainable AI (XAI): Systematically integrating XAI techniques into all developed components to provide interpretable feedback to educators and clinicians, explaining which features contribute to a risk assessment.
- Longitudinal Studies and Clinical Validation: Conducting longitudinal studies to track the predictive validity of the identified features and models over time. Collaborating with educational psychologists and clinicians to validate the developed tools against established diagnostic practices in real-world settings.
- Exploration of Other Modalities: Investigating the fusion of the current handwriting and eye-tracking systems with other relevant data modalities, such as audio recordings of reading/speaking or performance on cognitive tasks, for an even more comprehensive screening tool.

Addressing these areas will contribute to the development of more accurate, reliable, and practical AI-driven tools for the early identification of dyslexia risk, ultimately benefiting children by facilitating timely support and intervention.

Bibliography

[1] M. J. Snowling, *Dyslexia: A Cognitive-Developmental Perspective*. John Wiley & Sons, 2013.

[2] S. E. Shaywitz, *Overcoming Dyslexia*. New York: Alfred A. Knopf, 2003.

[3] S. Rangasrinivasan, M. S. S. Suresh, A. Olszewski, S. Setlur, B. Jayaraman, and V. Govindaraju, "Al-Enhanced Child Handwriting Analysis: A Framework for the Early Screening of Dyslexia and Dysgraphia," *SN Computer Science*, vol. 6, no. 399, 2025, Published online: 17 April 2025. DOI: 10.1007/s42979-025-03927-0.

[4] U. Kuhl, N. E. Neef, I. Kraft, *et al.*, "The emergence of dyslexia in the developing brain," *NeuroImage*, vol. 213, p. 116 633, 2020. DOI: 10.1016/j.neuroimage.2020.116633.

[5] G. Reid, *Dyslexia: A Practitioner's Handbook*. John Wiley & Sons, 2016.

[6] G. Aldehim, M. Rashid, A. Alluhaidan, and S. Basheer, "Deep learning for dyslexia detection: A comprehensive cnn approach with handwriting analysis and benchmark comparisons," *Journal of Disability Research*, vol. 10, no. 1, pp. 55–72, 2024. DOI: 10.57197/JDR-2024-0010.

[7] N. D. Alqahtani, B. Alzahrani, and M. S. Ramzan, "Detection of dyslexia through images of handwriting using hybrid ai approach," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 10, pp. 99–110, 2023. DOI: 10.14569/IJACSA.2023.0141099.

[8] R. Werth, "Dyslexia: Causes and concomitant impairments," 2023. DOI: 10.3390/brainsci13030472.

[9] P. J. Chung, D. R. Patel, and I. Nizami, "Disorder of written expression and dysgraphia: definition, diagnosis, and management," *Translational Pediatrics*, vol. 9, no. Suppl 1, S46–S54, 2020. DOI: 10.21037/tp.2019.11.01.

[10] F. Vlachos and E. Avramidis, "The Difference between Developmental Dyslexia and Dysgraphia: Recent Neurobiological Evidence," *International Journal of Neuroscience and Behavioral Science*, vol. 8, no. 1, pp. 1–5, 2020. DOI: 10.13189/ijnbs.2020.080101.

[11] M. Wolf and P. G. Bowers, "The double-deficit hypothesis for the developmental dyslexias," *Journal of Educational Psychology*, vol. 92, no. 1, pp. 1–19, 2000.

[12] H. L. Swanson, "Working memory and reading disabilities: Both phonological and executive processing deficits are important," in *Working memory and neurodevelopmental disorders*, T. P. Alloway and S. E. Gathercole, Eds., Psychology Press, 2006, pp. 59–88.

[13] S. P. Patil, R. S. Apare, R. H. Borhade, and P. N. Mahalle, "Automated dyslexia screening using children's handwriting in english language using convolutional neural network and bidirectional long short-term memory model," *Engineered Science*, Jan. 2024. DOI: 10.30919/es1345.

[14] N. Seman, I. Isa, S. A. Ramlan, L.-C. Wang, and M. Maruzuki, "Classification of handwriting impairment using cnn for potential dyslexia symptom," pp. 188–193, Aug. 2021. DOI: 10.1109/ICCSCE52189.2021.9530989.

[15] M. Robaa, M. Balat, R. Awaad, E. Omar, and S. A. Aly, "Explainable ai in handwriting detection for dyslexia using transfer learning," Oct. 2024. DOI: 10.48550/arXiv.2410.19821.

[16] M. H. Fischer and A. Luxembourger, "A test of three models of character reversal in typically developing children's writing," *Frontiers in Communication*, vol. 6, p. 719 652, Oct. 2021. DOI: 10.3389/fcomm.2021.719652.

[17] C. Prado, M. Dubois, and S. Valdois, "The eye movements of dyslexic children during reading and visual search: Impact of the visual attention span," *Vision research*, vol. 47, pp. 2521–30, DOI: 10.1016/j.visres.2007.06.001.

[18] W. Zhou, A. Wang, and M. Yan, "Eye movements and the perceptual span among skilled uighur readers," *Vision Research*, vol. 182, pp. 20–26, 2021, ISSN: 0042-6989. DOI: https://doi.org/10.1016/j.visres.2021.01.005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0042698921000225.

[19] E. I. Toki, "Using Eye-Tracking to Assess Dyslexia: A Systematic Review of Emerging Evidence," *Education Sciences*, vol. 14, no. 11, p. 1256, Nov. 2024. DOI: 10.3390/educsci14111256.

[20] L. Rello and M. Ballesteros, "Detecting readers with dyslexia using machine learning with eye tracking measures," in *Proceedings of the 12th Web for All Conference (W4A '15)*, ser. W4A '15, Florence, Italy: Association for Computing Machinery (ACM), May 2015, pp. 1–8. DOI: 10.1145/2745555.2746644.

[21] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. DOI: 10.1109/5.726791.

[22] D. Kilaru, *The Annotated ResNet-50*, https://medium.com/data-science/the-annotated-resnet-50-a6c536034758, Jan. 2023.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[24] DigitalOcean Community, *Popular Deep Learning Architectures: ResNet, InceptionV3, SqueezeNet*, https://www.digitalocean.com/community/tutorials/popular-deep-learning-architectures-resnet-inceptionv3-squeezenet, Jan. 2020.

[25] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826. DOI: 10.1109/CVPR.2016.308.

[26] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179–211, 1990. DOI: 10.1207/s15516709cog1402_1.

[27] Zhang, Aston and Lipton, Zachary C. and Li, Mu and Smola, Alexander J., *Long Short-Term Memory (LSTM) - Dive into Deep Learning*, https://d2l.ai/chapter_recurrent-modern/lstm.html.

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

[29] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997. DOI: 10.1109/78.650093.

[30] D. Birla, *Autoencoders*, https://medium.com/@birla.deepak26/autoencoders-76bb49ae6a8f, May 2020.

[31] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006. DOI: 10.1126/science.1127647.

[32] A. Radford, J. W. Kim, C. Hallacy, *et al.*, "Learning transferable visual models from natural language supervision," 2021. arXiv: 2103.00020 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2103.00020.

[33] H. Bao, L. Dong, S. Piao, and F. Wei, "BEiT: BERT Pre-Training of Image Transformers," in *International Conference on Learning Representations (ICLR)*, 2022. [Online]. Available: https://openreview.net/forum?id=p-BhZSz59o4.

[34] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010. DOI: 10.1109/TKDE.2009.191.

[35] M. S. A. B. Rosli, I. S. Isa, S. A. Ramlan, and M. I. F. Maruzuki, "Development of cnn transfer learning for dyslexia handwriting recognition," in *2021 11th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, 2021, pp. 1–6. DOI: 10.1109/ICCSCE52189.2021.9530971.

[36] H. W. Liu, S. Wang, and S. X. Tong, "Dysditect: Dyslexia identification using cnn-positional-lstm-attention modeling with chinese dictation task," *Brain Sciences*, vol. 14, no. 5, p. 444, 2024. DOI: 10.3390/brainsci14050444.

[37] Y. Alkhurayyif and A. R. W. Sait, "Multi-modal dyslexia detection model via swin transformer with closed-form continuous time networks," *IEEE Access*, vol. 12, no. 1, pp. 127 580–127 591, 2024. DOI: 10.1109/ACCESS.2024.3454795.

[38] P. Haller, A. Säuberli, S. E. Kiener, J. Pan, M. Yan, and L. Jäger, "Eye-tracking based classification of Mandarin Chinese readers with and without dyslexia using neural sequence models," *arXiv preprint arXiv:2210.09819*, Oct. 2022. DOI: 10.48550/arXiv.2210.09819. arXiv: 2210.09819 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2210.09819.

[39] Z. Gomolka, A. Kubas, T. M. Rutkowski, P. Wolski, and T. Wolski, "Diagnosing Dyslexia in Early School-Aged Children Using the LSTM Network and Eye Tracking Technology," *Applied Sciences*, vol. 14, no. 17, 2024. DOI: 10.3390/app14178004.

[40] L. Vajs, M. N. Benfatto, M. Wennberg, *et al.*, "Eye-tracking image encoding: Autoencoders for the crossing of language boundaries in developmental dyslexia detection," *IEEE Access*, vol. 11, 2023. DOI: 10.1109/ACCESS.2023.3234438.

[41] Y. Zhang, Q. Li, S. Nahata, T. Jamal, S.-K. Cheng, and G. Cauwenberghs, "Integrating Large Language Model, EEG, and Eye-Tracking for Word-Level Neural State Classification in Reading Comprehension," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 32, pp. 3465–3475, Aug. 2024. DOI: 10.1109/TNSRE.2024.3435460.

[42] M. Robaa, M. Balat, R. Awaad, E. Omar, and S. A. Aly, *Explainable ai in handwriting detection for dyslexia using transfer learning*, See [15], Oct. 2024. DOI: 10.48550/arXiv.2410.19821.

[43] I. S. Isa, W. N. S. Rahimi, S. A. Ramlan, and S. N. Sulaiman, "Automated detection of dyslexia symptom based on handwriting image for primary school children," *Procedia Computer Science*, vol. 163, pp. 440–449, 2019. DOI: 10.1016/j.procs.2019.12.127.

[44] U.-V. Marti and H. Bunke, "The IAM-database: An English sentence database for off-line handwriting recognition," *International Journal on Document Analysis and Recognition*, vol. 5, no. 1, pp. 39–46, 2002. DOI: 10.1007/s100320200071.

[45] Sathvik, D. L. (dlsathvik04), *Dyslexia_Detection GitHub Repository*, https://github.com/dlsathvik04/Dyslexia_Detection, 2024.

[46] M. N. Benfatto, G. Ö. Seimyr, J. Ygge, T. Pansell, A. Rydberg, and C. Jacobson, *Screening for Dyslexia Using Eye Tracking During Reading*, figshare. Collection. 2016. DOI: 10.6084/m9.figshare.c.3521379.v1. [Online]. Available: https://doi.org/10.6084/m9.figshare.c.3521379.v1.

[47] H. Griffith, D. Lohr, and O. V. Komogortsev, *GazeBase Data Repository*, figshare. Dataset. 2020. DOI: 10.6084/m9.figshare.12912257.v3. [Online]. Available: https://doi.org/10.6084/m9.figshare.12912257.v3.

[48] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[49] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.