Communication and Computer Engineering Research Magazine, Vol (1), No (1), 2024.

Deep Learning-based Eye-Tracking Biomarker Analysis for Computer-Aided ASD Diagnosis

Dalia I. Elewaily[1,*], Abdulrahman S. Elsayed[2], Abdallah M. Eldaw[2], Ahmed Y. Elgraihy[2], Hazem H. Ibrahim[2], Moaz E. Tawfik[2], Mohammad K. Abulkhair[2], Omnia H. Azzam[2], Safaa W. Mohamed[2], Sara Y. Nour[2], Yousef K. Sedky[2], Sarah M. Ayyad[1]

[1] Computer and Control Systems Depart mment, Faculty of Engineering, Mansoura University, Egypt
[2] Communication and Computer Engineering Program, Faculty of Engineering, Mansoura University, Egypt

*: Corresponding Author

Research Paper Data:

- *Paper ID:*
- *Submitted:*
- *Revised:*
- *Accepted:*

Abstract

Autism Spectrum Disorder (ASD) presents challenges in early detection and intervention, prompting novel diagnostic approaches. This study employs machine learning and deep learning on eye-tracking data to uncover ASD-related patterns. Utilizing traditional algorithms and Convolutional Neural Networks (CNNs), we analyze longitudinal eye-tracking biomarker datasets. Logistic Regression, KNN, SVC, Decision Tree, and Random Forest algorithms are trained and evaluated alongside CNN models.

Results demonstrate efficacy in discerning ASD-related patterns, with SVC and Random Forest showing robust generalization. Integration of an augmented CNN model into the "TAYF" desktop application marks progress towards clinical deployment, offering clinicians a tool for early ASD detection. This research highlights the potential of computational methods to enhance understanding and intervention for ASD-related visual processing abnormalities, aiming to improve outcomes for individuals on the autism spectrum.

Keywords: *Autism Spectrum Disorder, Accuracy, Convolutional Neural Networks, Diagnosis, Eye-Tracking, Machine Learning Classifiers*

1. Introduction

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental condition characterized by persistent challenges in social communication, and repetitive behaviors. Early diagnosis and intervention are crucial for improving outcomes and enhancing the quality of life for individuals with ASD. However, diagnosing ASD in early childhood can be challenging due to the heterogeneity of symptoms, variability in presentation, and the lack of objective diagnostic tools. Traditional diagnostic methods often rely on clinical observation and standardized assessments, which can be time-consuming and subject to variability in interpretation.

In recent years, there has been growing interest in leveraging machine learning (ML) and deep learning (DL) techniques to aid in the early diagnosis of ASD [22]. ML and DL offer powerful computational approaches for analyzing diverse datasets and identifying patterns that may be indicative of ASD. By harnessing the potential of these advanced algorithms, researchers and clinicians aim to develop accurate, efficient, and objective methods for early ASD diagnosis.

The utilization of ML and DL approaches in ASD diagnosis [2, 8] encompasses a multifaceted analysis of diverse data types. This inclusive examination incorporates behavioral observations, developmental assessments, neuroimaging data, genetic information, and sensor-based data, each contributing pivotal insights into the ASD diagnostic landscape. The taxonomy delineated in Figure (1) classifies ML and DL-driven ASD diagnostic and screening methodologies into six principal categories.

Primarily, behavioral analysis-based approaches represent a foundational category, concentrating on scrutinizing behavioral data to unveil patterns indicative of ASD [17, 18]. Employing ML and DL algorithms features such as social interactions, communication patterns, repetitive behaviors, and sensory sensitivities are scrutinized to discern ASD-related traits with precision and efficacy.

Similarly, neuroimaging analysis-based methodologies [14, 15] harness advanced brain imaging techniques such as structural magnetic resonance imaging (MRI), functional MRI (fMRI), and diffusion tensor imaging (DTI) [24] to elucidate the intricate brain structure and function in ASD individuals. Through ML and DL algorithms, neuroimaging data is meticulously analyzed to unveil biomarkers, structural abnormalities, and functional connectivity patterns intricately associated with ASD.

Furthermore, genetic analysis-based strategies [9,10] pivot towards the dissection of genetic data to pinpoint genetic markers or variations linked to ASD susceptibility. Here, ML and DL algorithms scrutinize genetic sequences, single nucleotide polymorphisms (SNPs) [24], gene expression profiles [25], and epigenetic modifications [26], unraveling the genetic underpinnings contributing to ASD pathogenesis.

Conversely, language and communication analysis-based methodologies scrutinize linguistic and communication data, delving into speech patterns, language use, and verbal/non-verbal communication to discern language deficits indicative of ASD. Employing ML and DL algorithms, natural language processing (NLP) [11] data, including interview transcripts, conversations [12], or written text, is meticulously analyzed to delineate linguistic features intricately associated with ASD.

Moreover, sensor-based monitoring approaches [19, 20] deploy wearable sensors or smart devices to amass data on movement, activity levels, physiological signals, and environmental factors, facilitating the identification of behavioral patterns characteristic of ASD. Through ML and DL algorithms, sensor data, including accelerometry, heart rate variability, or sleep patterns [27], is meticulously analyzed to unearth characteristic features indicative of ASD.

Fig. 1. Categories of ML and DL-based ASD Diagnosis Approaches

Lastly, eye-tracking analysis-based methodologies emerge as a promising frontier, endeavoring to pinpoint biomarkers associated with ASD [16]. By leveraging eye-tracking biomarkers, these methodologies analyze eye movement patterns during social interactions, visual tasks, or attentional tasks to discern disparities in gaze behavior indicative of ASD. Through the adept analysis of fixation patterns, saccadic movements, or pupil dilation, ML and DL algorithms unveil abnormalities in visual attention intricately associated with ASD.

Each methodology-based ASD diagnosis possesses distinctive strengths, potential applications, and contributions to the advancement of ASD. However, ML and DL-based ASD diagnosis methodologies encounter numerous limitations, encompassing challenges about data quality and quantity, model interpretability, overfitting risks, the necessity for manual feature engineering, data imbalance, generalization to real-world contexts, ethical and privacy considerations, validation and reproducibility constraints, and impediments to clinical integration. These constraints underscore the imperative for interdisciplinary collaboration and sustained research endeavors aimed at mitigating these obstacles and cultivating robust, interpretable, and ethically grounded ASD diagnosis methodologies capable of seamless translation into clinical settings.

The main objective of this research is to employ traditional machine learning algorithms and deep learning techniques for the accurate classification of eye-tracking data related to ASD. Through comprehensive evaluation, the proposed system aims to understand the accuracy, interpretability, efficiency, generalization, and feature engineering requirements of the trained models. Additionally, the system seeks to integrate the trained augmented CNN model into a desktop application ("TAYF") to facilitate clinical ASD diagnosis, offering clinicians a valuable tool for early detection and intervention strategies employing rigorous methodology, utilizing datasets effectively, conducting thorough model training, and evaluating performance meticulously, the proposed system strives to close the divide between research findings and clinical application, with the overarching goal of enhancing ASD diagnosis and intervention outcomes.

The primary contributions of this research paper are as follows:

1. Reviewing and classifying the predominant approaches to ASD diagnosis based on machine learning and deep learning methodologies.
2. Employing traditional machine learning and deep learning techniques to analyze an eye-tracking biomarker dataset for ASD diagnosis.
3. Evaluating the performance metrics, dataset utilization, and methodology of the trained models to enhance ASD diagnosis and intervention efficacy.
4. Integrating the trained model into the desktop application "TAYF" for clinical ASD diagnosis, thereby facilitating its practical application in real-world scenarios.

The subsequent sections of the paper are structured as follows: Section 2 offers an overview of the most related research studies on ASD diagnosis utilizing machine learning (ML) and deep learning (DL) approaches, elucidating their methodologies and limitations. In Section 3, the methodology of the proposed deep learning model slated for integration into clinical diagnosis is discussed. Section 4 entails the evaluation of the performance of both trained ML and DL models. Section 5 delves into the integration of the trained DL model within the developed desktop application 'TAYF'. Finally, Section 6 concludes the paper and delineates avenues for future research.

2. Literature review

The field of Autism Spectrum Disorder (ASD) diagnosis and screening has witnessed significant advancements driven by interdisciplinary research efforts across various domains. In this section, an overview of the most relevant literature is presented, that pertains to ML and DL-based methodologies for ASD diagnosis, encompassing behavioral analysis, neuroimaging analysis, genetic analysis, language and communication analysis, sensor-based monitoring, and eye-tracking analysis. The methodologies, utilized approaches, performance metrics, and the limitations of the most related work are summarized in Table 1.

Neuroimaging analysis-based methodologies [13-15] have received significant attention in the pursuit of biomarkers for ASD, with ML and DL algorithms playing a pivotal role in analyzing both structural and functional brain imaging data. As in [14], where authors employed deep learning models on MRI data to discern structural and strategic markers of ASD, delineating specific brain regions associated with the disorder. The objective was to streamline ASD diagnosis by furnishing clinicians with efficient tools for classification and analysis. Additionally, a study presented the ASD-SAENet model, which integrates a Sparse Autoencoder and Deep-Neural Network to classify ASD from fMRI data with high accuracy. This model exhibits notable specificity and robust generalizability across diverse imaging sites, underscoring its potential for early and precise ASD detection.

Additionally, there has been a thorough exploration of behavioral analysis-based methodologies for ASD diagnosis, employing machine learning (ML) and deep learning (DL) algorithms to scrutinize behavioral data and identify patterns suggestive of ASD. In a study referenced in [17], interaction analysis within home videos coupled with explainable artificial intelligence was employed to recognize potential indicators of infant development impairment, with a specific focus on ASD-related features. Additionally, in another study proposed in [18], a Composite AI framework amalgamating Deep Learning and rule-based systems was introduced for behavior analysis, exhibiting high accuracy in discerning communication cues. This framework adeptly interprets both verbal and non-verbal cues, showcasing the potential of Composite AI in augmenting social interaction analysis.

Furthermore, Genetic analysis-based methods have shed light on the genetic underpinnings of ASD, with ML and DL algorithms utilized to analyze genetic data and identify genetic markers or variations associated with ASD risk. In [10], ML algorithms were harnessed to discern gene biomarkers associated with ASD and Alzheimer's disease, achieving remarkable accuracy rates of 97.62% for ASD prediction and 92.95% for Alzheimer's discrimination. The methodology involved feature selection and genetic algorithm optimization. Similarly, [9] leveraged gene expression data from toddlers with ASD and controls to identify crucial biomarkers using LASSO regression and neural networks. The study aimed to formulate a predictive model for early ASD diagnosis based on immune-related biomarkers, achieving an accuracy of 86% and an AUC of 0.88.

Moreover, Language and communication analysis-based methodologies have leveraged ML and DL techniques to analyze linguistic and communication data for ASD diagnosis. In [12], ADOS-2 recordings were employed to extract prosodic features using a harmonic model, with a specific focus on pitch and loudness, aiming to discern between individuals with ASD and typically developing controls. This study underscored the efficacy of prosodic measures over articulation features in distinguishing ASD cases. Additionally, [11] employed natural language processing (NLP) techniques to translate saccadic eye movements into sequences resembling text, to ASD. The approach exhibited promising accuracy in classification models, suggesting the potential of sequence-based modeling for ASD diagnosis.

Contrastingly, sensor-based monitoring methodologies have emerged as a promising avenue for ASD diagnosis, employing wearable sensors and smart devices to gather data on movement, activity levels, physiological signals, and environmental factors. In [19], authors introduced a platform based on wearable sensors to recognize gestures in children with ASD, utilizing machine learning algorithms. The system's objective is to enhance communication by monitoring and categorizing gestures exhibited by individuals with autism. Additionally, in [20], the presented study aimed to diagnose ASD by employing AI and biosensors to analyze brain connectivity patterns. Through the utilization of functional MRI data and innovative algorithms, the research endeavors to enhance accuracy in ASD detection and deepen understanding of functional abnormalities.

On the other hand, methodologies based on eye-tracking analysis have exhibited considerable potential in discerning biomarkers linked to ASD. For instance, a study conducted by [1] employed eye-tracking technology in conjunction with machine learning algorithms to scrutinize gaze behavior, aiming at early detection of ASD in pediatric populations. The findings of this investigation underscored the efficacy of leveraging visual representations and deep learning models to achieve objective ASD diagnosis. Additionally, the work referenced in [7] delves into the shortcomings inherent in prevailing datasets and conventional screening techniques, accentuating the promise held by deep neural networks in enhancing diagnostic efficacy.

Further, authors in [3] proposed an automated technique for detecting ASD, termed ETASD-CBODL, which amalgamated U-Net segmentation, Inception v3 feature extraction, CBO hyperparameter optimization, and LSTM classification methodologies. The primary objective was to augment the accuracy of ASD identification through the integration of eye-tracking data and deep-learning models. Meanwhile, [4] introduced an Involution Fused ConvNet architecture tailored for the analysis of eye-tracking patterns observed in children diagnosed with ASD, resulting in commendable accuracy rates and performance metrics. Through the amalgamation of disparate datasets, augmentation techniques, and the utilization of sophisticated neural network architectures, the investigat-

Table 1

Summary of the most related ML and DL-based ASD diagnosis Studies.

Category	Reference	Year	Methodology	Dataset	Modelling algorithms	Performance Metrics	Limitations
Neuroimaging	[13]	2020	utilizing deep learning models, including CNN, STN, CAM, RNN, RAM, and FC, to analyze structural and strategic evidence of ASD using MRI data without human feature extraction.	• YUM dataset: 84 subjects diagnosed with ASD based on DSM-V criteria • ABIDE dataset: Over 1000 images from multiple institutions	• CNN (Convolutional Neural Network) • STN (Spatial Transformer Network) • CAM (Class Activation Mapping) • RNN (Recurrent Neural Network) • RAM (Recurrent Attention Model) • FC (Fully-Connected Network)	• The 2D/3D CNN achieved an Accuracy of 89% for the YUM dataset • A simple 3D CNN achieved an Accuracy of 90% for the ABIDE dataset	• Data variability due to multiple institutions and scanners in the ABIDE dataset • Small sample size in the YUM dataset compared to ABIDE • Structural heterogeneity in the ABIDE dataset. • Challenges in achieving consistent accuracy across different datasets • Interpretability and comprehensibility of deep learning models in diagnosing psychiatric disorders
	[14]	2021	The study proposed the ASD-SAENet model, combining a Sparse Autoencoder and Deep-Neural Network, trained using k-fold cross-validation on fMRI data to classify ASD. Fine-tuning was done using the Adam optimizer, with model evaluation on both the whole dataset and individual sites.	• ABIDE dataset containing 1,035 subjects.	• ASD-SAENet model combining Sparse Autoencoder and Deep-Neural Network.	• Accuracy: 70.8% • Sensitivity: 62.2% • Specificity: 79.1%	• Modest sample size for training and evaluation. • Lack of interpretability of deep-learning model features. • Potential group differences due to head movement in fMRI data. • Uncertainty in distinguishing neurological differences from noise.
	[15]	2021	The study combines deep feature selection (DFS) to identify critical functional connections related to ASD and utilizes graph convolutional networks for classification, demonstrating improved accuracy in distinguishing ASD from typically developing individuals.	• ABIDE database	• Deep Feature Selection (DFS). • Graph Convolutional Networks (GCN). • Multi-Layer Perceptron (MLP). • Logistic Regression, Gaussian Process (GP). • Support Vector Machine (SVM)	• Accuracy improvement by DFS: 8% to 15% • Classification accuracy comparison with traditional methods: Superior performance	The study is limited by the availability and quality of data from the ABIDE database. The generalizability of the proposed method to other datasets and populations needs further validation. The interpretability of the identified functional connections and their direct relevance to ASD pathology requires additional investigation.
Behavioural analysis	[17]	2023	A Composite AI framework combining Deep Learning methods for feature extraction with rule-based systems for detecting key episodes, followed by classifiers for interpreting activities in social interactions.	• not explicitly mentioned.	• Deep Learning methods for feature extraction, • Rule-based systems for episode detection. • Classifiers for activity interpretation.	• Achieved 87% accuracy for verbal requests and 89% accuracy for non-verbal requests.	• Reliance on predefined rules may limit adaptability to diverse social interaction scenarios. • Lack of real-time processing capabilities Limited scalability to large-scale applications.
	[18]	2023	Utilizing interaction analysis in home videos and explainable artificial intelligence to identify potential indicators of infant development impairment, focusing on specific features related to ASD	• Home videos of infants-caregivers interactions, including children later diagnosed with ASD and typically developing children.	• Explainable artificial intelligence using the SHAP approach.	• Sensitivity and specificity for identifying potential red flags for ASD were high, indicating promising results for early detection.	• Limited sample size with a small number of children. • Disproportion between male and female children in the dataset. • Inconsistency in the most rated features identified by observers.
Genetic Analysis	[10]	2020	• Utilize machine learning algorithms to identify candidate gene biomarkers for ASD and Alzheimer's disease by analyzing gene expression profiles and applying feature selection techniques. • Employing a workflow involving signal-to-noise ratio analysis, logistic threshold function, HSIC-Lasso algorithm, and Regularized Genetic Algorithm to select optimal gene subsets for disease prediction.	• Autism dataset: GSE26415 with 42 samples (21 autistic, 21 control) and 19,194 gene expression probes. • Alzheimer's dataset: GSE1297 with 31 samples (9 control, 22 AD-affected).	• Bayes net (BN), logistic regression (LR), support vector machine (SVM), multilayered perceptron neural network (MLP-NN), extreme gradient boosting (XGBoost).	• SVM model:92.95% accuracy for Alzheimer's samples. • Extreme Gradient Boosting algorithm: 97.62% for ASD	• Limited sample size in the datasets. • Complexity of high-dimensional genomic data affecting computational efficiency. • Optimal results are yet to be achieved with more sophisticated mathematical models.
	[9]	2023	Utilizing gene expression data from ASD and control samples, performing differential gene expression analysis, and employing machine learning techniques like LASSO regression, logistic regression, and nomogram construction for early ASD diagnosis.	• Gene expression data from 128 ASD and 126 control toddlers, were obtained from the Gene Expression Omnibus (GEO) database (GSE111175 and GSE42133).	• Least Absolute Shrinkage and Selection Operator (LASSO) regression. • Binary logistic regression.	• Accuracy achieved by LASSO regression: 86%. • Accuracy achieved by neural network models: 88%. • Area Under the Curve (AUC) for the neural network model: 0.88	• Limited sample size • Lack of external validation in larger, heterogeneous populations. • Reliance on peripheral blood biomarkers may not fully capture the complexity of ASD. • Potential overfitting of machine learning models.
Language and Communication	[11]	2020	NLP techniques were applied to transform saccadic eye movements into text-like sequences for detecting ASD using ConvNet and LSTM models. Evaluation is done through a 3-fold cross-validation procedure, employing the SMOTE over-sampling technique, with performance peaking at L=400 for sequence length.	• Eye-tracking data for individuals with and without ASD.	• ConvNet • LSTM	• ConvNet Model: Precision 0.79, Recall 0.71. • LSTM Model: Precision 0.72, Recall 0.48.	• Lack of a benchmark dataset in ASD literature for objective comparison with other ML approaches. • Limited exploration of ML approaches integrating eye-tracking data. • Performance decline observed beyond sequence length L=400.
	[12]	2021	Utilizing ADOS-2 recordings from conversational tasks to extract prosodic features using a harmonic model, focusing on pitch and loudness to differentiate between individuals with ASD and typically developing (TD) controls.	• ADOS-2 recordings from conversational tasks Modelling	• Support Vector Machine (SVM)	Accuracy: • AUC-ROC 88.27% • FMC 77.6%	• Small sample size and lack of gender and IQ balance. • Preliminary results not adjusted for multiple comparisons. • Need for replication in larger, more diverse samples. •

Cont. Table 1

Summary of the most related ML and DL-based ASD diagnosis Studies.

Category	Reference	Year	Methodology	Dataset	Modelling algorithms	Performance Metrics	Limitations
Sensor-based monitoring	[19]	2021	Extracting features from sensor data in time and frequency domains, evaluating various classifiers for gesture recognition, and implementing real-time gesture monitoring using Raspberry Pi.	9 ASD children were used for training and testing, with each gesture performed 7-12 times.	• K-Nearest Neighbors (KNN), • Decision Tree, • Random Forest.	• Accuracy of about 91% in recognizing gesture movements of children with ASD.	• Limited sample size of ASD children used for training and testing. • Lack of public availability of the dataset. • Dependency on specific sensors and data acquisition setup.
	[20]	2023	Utilizing AI algorithms and functional MRI data to diagnose ASD, incorporating multiple brain atlases and a low estimated rank tensor approach to analyze functional connectivity patterns.	ABIDE dataset	• Linear SVM • 2D CNN	• Accuracy - 76.61% (AAL atlas), • 63.74% (Pearson correlation FCN) • AUC - 0.82	• Complexity in estimating ideal connections in fMRI data. • Challenges in dataset availability and data heterogeneity from multiple sites. • Subject variability impacting brain region identification and connectivity analysis.
Eye-tracking analysis	[1]	2021	recruited 59 school-aged children, including those with ASD and typically developing children, to analyze gaze behavior using eye-tracking technology [T3]	59 school-aged participants	• Convolutional Neural Network (CNN)	• Accuracy ≈71% [T5]	• Small number of participants affecting generalizability. • Lack of access to all standardized test scores for clinical diagnosis. • Short duration of video scenarios impacting data richness [T6].
	[7]	2022	Utilizing eye-tracking scan path images to train machine learning models for autism screening, comparing traditional algorithms with a deep neural network approach.	• Original Dataset: 547 ETSP images • Augmented Dataset: 2566 ETSP images	• Decision Stump (DSVM) • Decision Jungle (DJ) • Boosted Decision Tree (BDT) • Deep Neural Network (DNN)	• Sensitivity: 78.57 • Specificity: 75.47 • (PPV): 87.12 • (NPV): 62.50 • (AUC): 78.00	• Limited availability of datasets suitable for machine learning research. • Challenges in obtaining an extensive dataset representing the complete spectrum of ASD symptoms. • Dependence on consistent and high-quality training data • Potential impact of the environment. distractions on the child's attention.
	[3]	2023	Devloping an automated ASD detection technique, ETASD-CBODL, utilizing U-Net segmentation, Inception v3 feature extraction, CBO hyperparameter optimization, and LSTM classification.	• ASD dataset with 547 instances, including 219 ASD class images and 328 typically developing (TD) class images	• ETASD-CBODL • FFNN • ANN • google net • Res-Net18 • Google-Net SVM	ACCURACY: • ETASD-CBODL:99.29 • FFNN:99.00 • ANN: 98.86 • Google-Net: 94.63 • Res-Net18: 98.56 • Google-Net SVM: 96.70	• Data quality impact on model accuracy. • Technical requirements for specialized tools. • Ethical concerns regarding data privacy. • Generalizability across diverse populations and settings.
	[5]	2023	Utilizing eye-tracking data from children with ASD and TD children, applied deep learning models (LSTM, CNN-LSTM, Bi-LSTM, GRU), and evaluated model performance using various metrics.	• Eye-tracking data was collected from 29 children with ASD and 30 typically developing children.	• LSTM, • CNN-LSTM, • Bi-LSTM, • GRU.	• BiLSTM: Test Accuracy 96.44%, Sensitivity 93.50%, Specificity 98.17%, AUC 97%, F1 Score 97.20%. • GRU: Test Accuracy 97.49%, Sensitivity 95.89%, Specificity 98.40%, AUC 97%, F1 Score 98.04%. • CNN-LSTM: Test Accuracy 97.94%, Sensitivity 96.44%, Specificity 98.79%, AUC 98%, F1 Score 98.39%. • LSTM: Test Accuracy 98.33%, Sensitivity 97.25%, Specificity 98.94%, AUC 98%, F1 Score 98.70%.	• Small Sample Size: A limited number of participants in the dataset may impact the generalizability of the findings. • Computational Complexity: Deep learning models used in the study may require significant computational resources and expertise. • Potential Bias: The study's findings may be influenced by biases inherent in the dataset or modeling techniques.
	[4]	2024	utilizing two datasets of typically developed children and those with ASD, merging them for a larger sample, and implementing an Involution Fused ConvNet architecture to analyze eye-tracking patterns, achieving high accuracy and performance metrics.	• Dataset 1: Contains 547 images with 328 from typically developed participants and 219 from ASD diagnosed participants. • Dataset 2: Includes 300 images with eye movement data from 14 children diagnosed with ASD and 14 typically developed children. • Combined Dataset: Merged Dataset 1 and Dataset 2 to create a larger dataset with 628 images from typically developed participants and 519 from ASD participants.	• Involution Fused ConvNet: A proposed hybrid model based on involution and convolution layers for analyzing eye-tracking patterns in children with ASD.	• Accuracy: 97.41% • F1 Score: 97.83% for grayscale photos and 97.12% for RGB photos.	• Unable to classify sub-types of ASD due to limitations in the research scope. • Did not measure computational complexity using FLOPs and MACCs for the new architectures. • Lack of optimized libraries for calculating complexity metrics for Involution and Transformer-based models.
	[6]	2024	Utilized a CNN model to analyze eye-tracking scan path images for predicting ASD severity based on a dataset of 59 children. Conducted experiments with eye-tracking technology on children viewing videos with visually attractive components to classify ASD severity.	• Public dataset with 59 children (29 ASD-diagnosed, 30 Typically Developing)	• Convolutional Neural Network (CNN)	• Accuracy of 95.59% in classifying ASD	• Relatively small dataset size • Study conducted in a controlled environment, potentially not reflecting real-world scenarios. • Focus on a specific age group • Lack of exploration on individual differences in eye-tracking patterns.

-ion underscored the efficacy of location-specific methodologies in the classification of ASD.

The collective literature review highlights the wide range of methodologies utilizing ML and DL for ASD diagnosis. However, these methodologies encounter obstacles relating to data accessibility, interpretability, and bias. Constraints such as limited datasets and interpretability hinder the transparency of models, while biases within training data can result in distorted predictions. Additionally, challenges regarding generalization, computational resource requirements, and ethical considerations contribute to the scalability and ethical application of these methodologies.

3. The Proposed System Methodology

The primary goal of this study is to utilize both traditional machine learning algorithms and deep learning techniques to accurately classify eye-tracking data associated with Autism Spectrum Disorder (ASD). The proposed system aims to comprehend the accuracy, loss, precision, sensitivity, and generalizability of the trained models. Moreover, the system endeavors to integrate the trained augmented CNN model into a desktop application named "TAYF," to aid clinical ASD diagnosis. This integration provides clinicians with a valuable tool for early detection and intervention strategies. An overview of the proposed system is depicted in Figure (2).

3.1 Dataset

The primary dataset utilized in this proposed system is publicly available in [21]. Comprising eye-tracking data obtained from a cohort of 60 participants, the dataset encompasses a total of 545 distinct images. Notably, participants' eye movements were meticulously tracked during the viewing of specific videos, incorporating individuals of varying genders and aged between 2 to 12 years. Furthermore, the dataset includes individuals diagnosed with ASD as well as typically developing counterparts, with each participant uniquely identified by an assigned ID to facilitate efficient data management and analysis.

As depicted in Figure (3), the longitudinal nature of the dataset, spanning various recording intervals over approximately one year, offers valuable insights into the temporal dynamics of eye movement patterns across developmental trajectories. Additionally, the dataset features visual representations showcasing eye movements superimposed on a black background, with distinct markers indicating participants diagnosed with ASD (TS) and typically developing individuals (TC). Serving as a pivotal resource, this dataset enables an in-depth exploration of eye movement patterns in the context of ASD compared to typically developing peers, shedding light on the underlying mechanisms of visual attention in neurodevelopmental disorders.

The inclusion of longitudinal data and a diverse participant cohort facilitates nuanced investigations into developmental trajectories and potential biomarkers associated with ASD-related visual processing abnormalities. Furthermore, the clear and intuitive visualization of eye-tracking data enhances interpretability, facilitating robust comparative analyses across participant groups. Thus, this dataset holds significant promise in advancing ASD research, informing diagnostic refinement, and the development of targeted intervention strategies to address visual attention deficits in affected individuals.

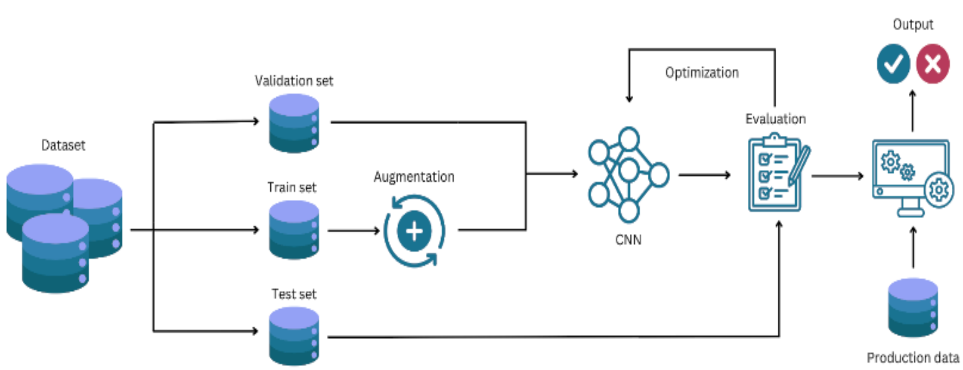
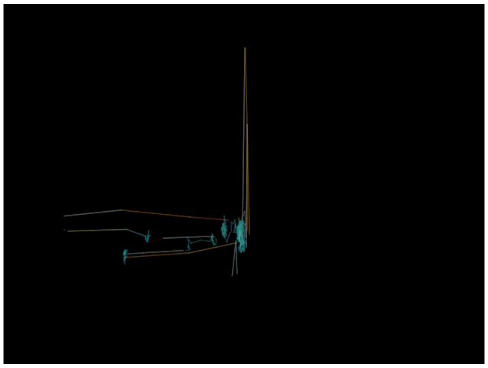
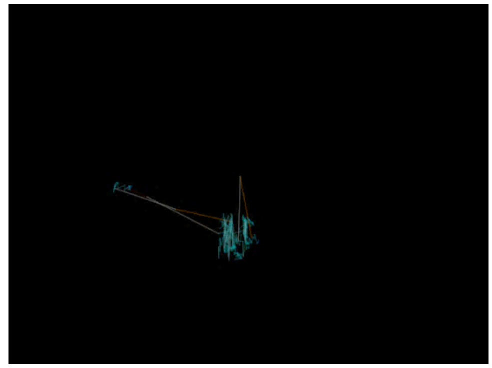
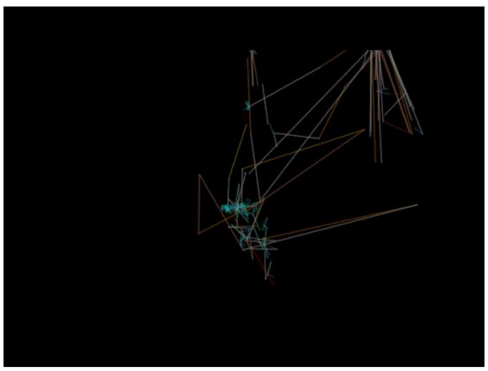
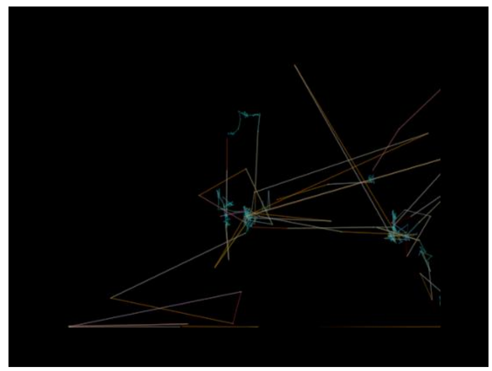


Fig. 2 Overview of the proposed system.

(a) Tc001_39 (b) Tc002_39

(c) TS001_11 (d) TS002_11

Fig. 3. Sample Images From The DataSet

3.2 Data Analysis

3.2.1 Traditional ML techniques

The analysis of the eye-tracking biomarker dataset using traditional machine learning techniques involves a systematic approach to train and evaluate models such as Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Classifier (SVC), Decision Tree, and Random Forest. With a dataset comprising 545 images, it is essential to partition the data into distinct subsets for training, validation, and testing, ensuring the integrity of the model development process.

The dataset splitting typically involves allocating 70% of the data for training, 20% for validation, and 10% for testing. This partitioning scheme enables the models to learn patterns from the training data, optimize hyperparameters using the validation set, and evaluate performance on unseen data via the testing set, thereby providing a reliable assessment of generalization capabilities.

Each traditional machine learning algorithm is then trained on the training subset using the eye-tracking biomarker data. Logistic Regression learns the relationship between the biomarkers and ASD diagnosis by fitting a logistic curve to the data. KNN classifies samples based on the majority class among its nearest neighbors, employing a distance metric such as Euclidean distance.

SVC aims to find the optimal hyperplane that separates classes in the feature space, while Decision Tree recursively partitions the feature space based on informative features to create a tree-like structure for classification. Random Forest, on the other hand, constructs multiple decision trees and aggregates their predictions to enhance robustness and accuracy.

After training, the models are validated using the validation subset to fine-tune hyperparameters and optimize performance metrics such as accuracy, precision, recall, and F1 score. Hyperparameter tuning involves adjusting parameters such as regularization strength, number of neighbors (K), kernel type, maximum tree depth, and number of trees in Random Forest to achieve optimal performance. The performance evaluation and results of the trained models will be discussed in Section 4.

3.3.2 The Deep Learning Model

The meticulous elucidation of the Convolutional Neural Network (CNN) tailored for ASD classification is detailed herein. The process encompasses various stages, including data collection, preprocessing, model development, training, evaluation, and testing, thereby shedding light on the intricate nuances of each step involved in crafting a dependable and resilient ASD classification model. The architecture of the CNN model is illustrated in Figure (4).

The TensorFlow library is utilized as the cornerstone for model generation, accompanied by NumPy, renowned for its diverse array of data structures and methods, facilitating seamless data manipulation. Matplotlib is instrumental in visualizing various figures throughout the model development process. Also, Optimal GPU memory utilization is ensured by the TensorFlow GPU memory growth configuration, dynamically allocating memory as needed to mitigate memory-related issues.

I. Data Collection & Preprocessing

A diverse dataset is meticulously assembled, forming the bedrock upon which our model is built. The dataset, comprising 545 images meticulously classified into two classes: TC images denoting Non-ASD and TS images signifying ASD serves as the foundation for model training. By Utilizing NumPy, data is converted into NumPy arrays, subsequently grouped into batches of 32 to form tensor shapes. This step is accompanied by rigorous plotting to ensure the integrity of data processing.

Additionally, Data was structured as [32, 256, 256, 3], representing batches of 32 images, each delineated by a 256x256 pixel matrix with RGB elements. Pixel values are scaled down from the range [0 – 255] to [0 – 1] to ensure consistency and numerical stability, followed by rigorous testing and plotting to validate proper rescaling. Moreover, data batches are meticulously partitioned into 70% training, 20% validation, and 10% testing subsets, employing take and skip functions alongside initial functions to ensure proper shuffling.

II. Building the CNN Model

The crux of our methodology lies in the design of the CNN architecture, which is depicted in Figure (hh) meticulously crafted to extract intricate features while mitigating overfitting. Leveraging Sequential for a linear stack of layers, the model architecture encompasses convolutional layers for feature extraction, pooling layers for representation downsampling, and dense layers for final output. Also, compilation utilizes the Adam algorithm, tracking model accuracy across training epochs.

III. Training & Testing the CNN Model

With the model architecture defined, the training phase ensues, guided by TensorBoard callbacks offering invaluable insights into the training process. The model learns from the labeled dataset, iteratively adjusting internal parameters to optimize predictions. Also for testing and performance evaluation of the trained models on the 10% of the dataset, precision, recall, and binary accuracy metrics are employed for rigorous model evaluation. The confusion matrix and classification reports that illuminate the capability of ASD classification are presented and discussed in Section 4.

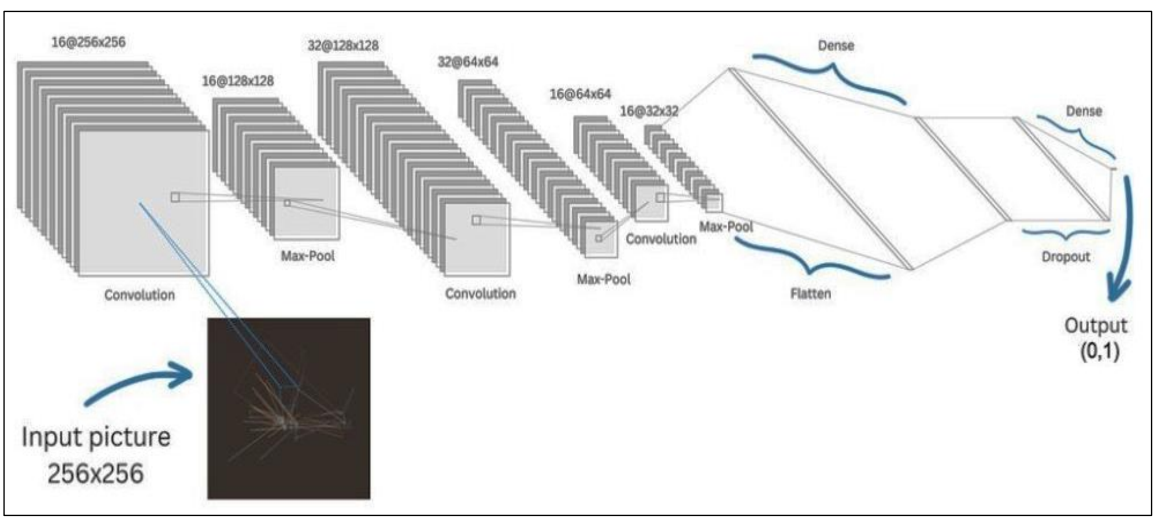


Fig. 4. CNN model Architecture.

IV. The Augmented the CNN Model

As a way to improve the model generalization, the dataset augmentation process was involved in expanding the original dataset by 40%, thereby enriching it with additional instances. This augmentation enhances the diversity and variability of the dataset, which can lead to improved model generalization and performance. Subsequently, the augmented dataset is utilized to train the CNN model tailored for ASD classification. The augmented model performance compared to the CNN model is discussed in Section 4. Also, The integration process of the augmented model into the TAYF application is presented in Section 5.

4. Performance Evaluation & Results

4.1 The Evaluation of the Traditional ML Models

The performance of traditional machine learning algorithms was evaluated as an image classification task using 10% of the dataset separated from the original dataset of eye-tracking biomarker images. The following algorithms were tested: Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Classifier (SVC), Decision Tree, and Random Forest. Each algorithm was trained and evaluated using 5-fold cross-validation, and the average accuracy across folds was calculated.

In the context of eye-tracking-based ASD diagnosis, the performance metrics of trained models serve as crucial indicators of their effectiveness in accurately identifying ASD-related patterns and abnormalities in eye movement data. These metrics provide quantitative insights into how well the models distinguish between individuals with ASD and typically developing individuals based on their eye-tracking scan paths. Key performance metrics in this domain include accuracy, precision, recall, and F1 score.

Accuracy (Eq. 1) measures the overall correctness of the model's predictions, it shows how well the model identifies ASD-related eye movement patterns. While precision (Eq. 2) quantifies the proportion of true positive predictions (correctly identified ASD cases) out of all positive predictions made by the model. It highlights the model's ability to avoid false positives, which is crucial in minimizing misdiagnosis in ASD screening. TP stands for True Positive, While TN stands for True Negative. In contrast, FP stands for Folse Positive, While FN stands for False Negative.

Further, Recall (Eq.3), also known as sensitivity, measures the proportion of true positive predictions out of all actual positive instances in the dataset. It indicates the model's capability to capture all relevant ASD-related eye movement patterns, thereby minimizing false negatives. While, the F1 score (Eq. 4) combines precision and recall into a single metric, providing a balanced assessment of the model's performance. It calculates the harmonic mean of precision and recall, offering a comprehensive evaluation of both false positives and false negatives in ASD diagnosis based on eye-tracking data.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1\ score = 2\ X\ \frac{Precision\ X\ Recall}{Precision + Recall} \quad (4)$$

The performance of the trained models was assessed through rigorous evaluation across multiple folds and metrics. Table 2 shows the average accuracy for each trained model during 5-fold cross-validation (CV). While Table 3 illustrates the performance metrics for each trained model. Also, their corresponding Confusion matrixes are illustrated in Figure (5).

Logistic Regression exhibited an average accuracy of 72.26% across the five folds, with a precision of 80% for class 0 and 54% for class 1. KNN achieved an average accuracy of 60.74%, demonstrating varying precision between classes, notably achieving 100% precision for class 0 but only 15% for class 1. SVC outperformed the other models with an average accuracy of 72.46%, displaying balanced precision across both classes and achieving 100% precision for class 0. Decision Tree and Random Forest models attained average accuracies of 59.18% and 70.90%, respectively. However, both models showed varying performance metrics across classes.

Furthermore, in terms of the 10% test set evaluation, SVC emerged as the top performer with an accuracy of 78.78%, indicating robust generalization capabilities. Random Forest also demonstrated favorable performance on the test set with an accuracy of 75.76%. Logistic Regression and Decision Tree models performed comparatively lower on the test set, with accuracies of 69.70% and 60.61% respectively, suggesting some degree of overfitting or limitations in generalization. KNN, while performing well in cross-validation, showed a lower accuracy of 66.67%

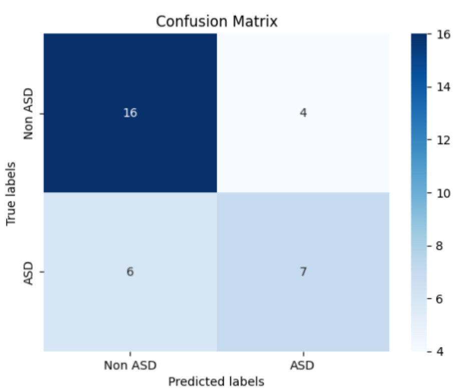
on the test set, indicating potential issues with generalization or suitability to unseen data. Overall, the evaluation highlights SVC and Random Forest as promising candidates for ASD classification due to their balanced performance across different evaluation metrics and robustness in generalization to unseen data.

Table 2
The Accuracy for each trained model during 5-fold cross-validation (CV)

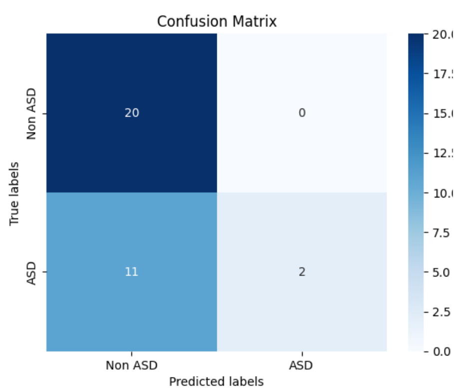
No.	ML Algorithm	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Avg Accuracy of folds	Avg accuracy on the test set
1	**Logistic Regression**	0.75728	0.72816	0.76471	0.64706	0.71569	0.72258	0.69697
2	**KNN**	0.60194	0.62136	0.62745	0.59804	0.58824	0.60741	0.66667
3	**SVC**	0.74757	0.70874	0.76471	0.69608	0.70588	0.72460	0.78788
4	**Decision Tree**	0.75728	0.63107	0.72549	0.71569	0.71569	0.59183	0.60606
5	**Random Forest**	0.73529	0.60784	0.72549	0.64706	0.76471	0.70904	0.75758

Table 3
Performance of the ML algorithms' tests on the original dataset

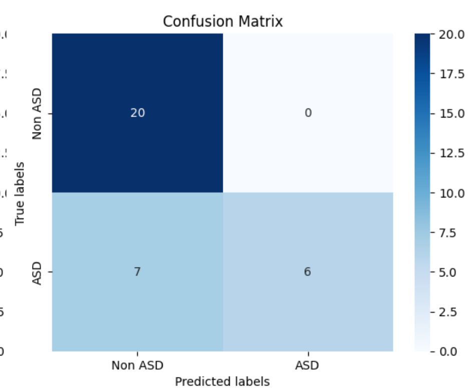
No.	ML Algorithm	Class	Support	Performance Metrics			
				Precision	Recall	F1-score	Accuracy
1	**Logistic Regression**	**0**	22	0.80	0.73	0.76	0.70
		1	11	0.54	0.64	0.58	
2	**KNN**	**0**	31	1.00	0.65	0.78	0.67
		1	2	0.15	1.00	0.27	
3	**SVC**	**0**	27	1.00	0.74	0.85	0.79
		1	6	0.46	1.00	0.63	
4	**Decision Tree**	**0**	21	0.70	0.67	0.68	0.61
		1	12	0.46	0.50	0.48	
5	**Random Forest**	**0**	22	0.85	0.77	0.81	0.76
		1	11	0.62	0.73	0.67	



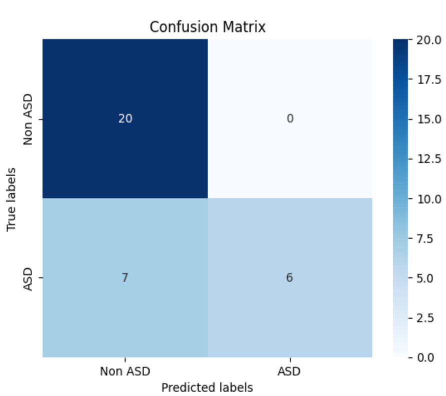
(a) Logestic Regrission



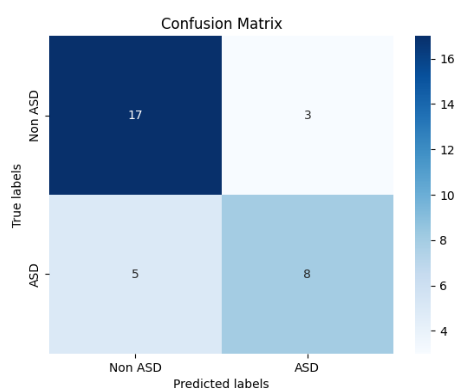
(b) KNN



(c) SVC



(d) Decision Tree



(e) Random Forest

Fig . 5. Confusion Matrix of the traditional ML models

4.2 The Evaluation of the Original CNN Model

The evaluation process of the CNN-trained model involved rigorous monitoring of key performance metrics on a subset comprising 10% of the dataset consisting of eye-tracking biomarkers. The CNN model underwent training for 25 epochs. Throughout the training phase, the model's performance was continuously assessed by tracking both training and validation loss, along with accuracy, at each epoch. This iterative process allowed for the observation of the model's learning progress and provided insights into its convergence behavior.

Following the initial training phase, the CNN model underwent fine-tuning to optimize its performance further. This fine-tuning process involved continued training until a predefined metric, such as validation accuracy, demonstrated minimal variation across consecutive epochs. By fine-tuning the model in this manner, efforts were made to enhance its ability to generalize well to unseen data and improve its overall performance on the classification task.

Table 4 and Figure (6) illustrate the training progress and validation performance of the CNN model across the 50 epochs. Initially, both the training and validation losses exhibit relatively high values, indicating significant discrepancies between predicted and actual values. However, as training progresses, both losses gradually decrease, suggesting an improvement in the model's performance and its ability to make accurate predictions. Correspondingly, the accuracy metrics, both for training and validation sets, increase steadily over epochs, indicating enhanced performance and a better fit to the data.

Notably, as the number of epochs increases, the training and validation losses converge to low values, while the accuracies approach 100%, indicating a high degree of alignment between predicted and actual values. This convergence suggests that the model has effectively learned the underlying patterns in the data and can make highly accurate predictions. Overall, the progression of metrics over epochs demonstrates the iterative process of model training and refinement, culminating in a well-performing neural network model with high accuracy and low loss on both training and validation datasets.

Table 4
Training progress and validation performance of the original CNN model

Epoch No.	Train Loss	Validation Loss	Train accuracy	Validation accuracy
1	0.618534	0.495032	0.661458	0.78125
2	0.572243	0.528925	0.697917	0.729167
3	0.52423	0.464259	0.747396	0.802083
4	0.490082	0.417983	0.744792	0.791667
5	0.412643	0.368568	0.799479	0.8125
6	0.406817	0.430676	0.799479	0.770833
7	0.361036	0.268364	0.820313	0.885417
8	0.342061	0.246872	0.851563	0.916667
9	0.289622	0.30532	0.86979	0.84375
10	0.27873	0.242421	0.867188	0.895833
11	0.229229	0.234496	0.908854	0.885417
12	0.255683	0.298521	0.888021	0.864583
13	0.216497	0.148398	0.911458	0.947917
14	0.129601	0.074943	0.955729	0.979167
15	0.079275	0.086742	0.971354	0.979167
16	0.067875	0.119244	0.981771	0.96875
17	0.085294	0.021911	0.96875	0.989583
18	0.066256	0.090504	0.96875	0.96875
17	0.054194	0.036544	0.979167	1
20	0.027889	0.02223	0.99219	1
21	0.03091	0.015118	0.989583	1
22	0.022305	0.012841	0.994792	1
23	0.007788	0.009223	1	1
24	0.006362	0.003719	1	1
25	0.005397	0.013103	1	1

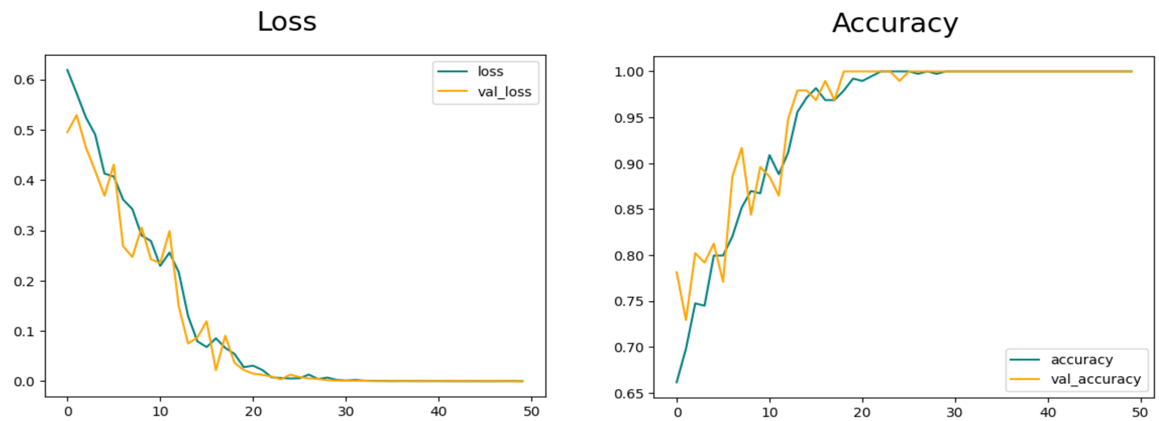


Fig. 6. Loss & Accuracy of the trained CNN model

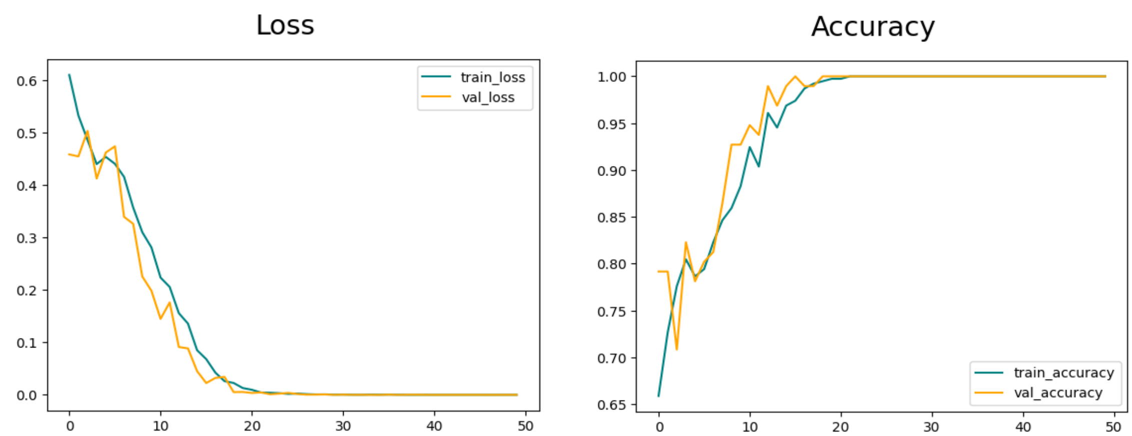
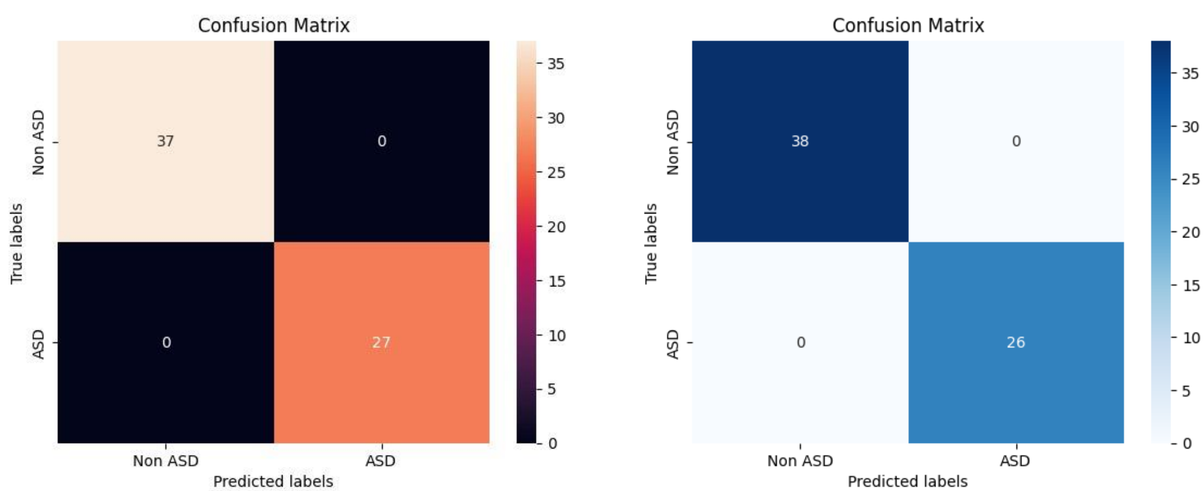


Fig. 7. Loss & Accuracy of the Augmented CNN model



(a) CNN Model **(b) The Augmented CNN Model**

Fig. 8. Confusion matrix of the original CNN Model VS the Augmented Model

4.3 The Evaluation of the Augmented CNN Model

The augmented CNN model was trained using data augmentation techniques to enhance its generalization ability, specifically on the training data. Throughout the training process, both training and validation metrics were closely monitored, revealing insightful trends in the model's performance. The validation accuracy of the augmented CNN model reached an impressive 100%, indicating that the model achieved optimal performance on the validation dataset. This suggests that the original CNN model likely wasn't overfitting to the training data, as further data augmentation did not significantly improve performance. Table 5 and Figure (7) illustrate the training progress and validation performance of the augmented CNN model. Also, Figure (8) shows the confusion matrices of the original and augmented CCN Model.

5. Clinical Integration (The Desktop Application 'TAYF')

This section delves into the intricacies of implementing and integrating the desktop application and transforming the augmented CNN model for deployment.

5.1 Converting the Model

The conversion process was initiated to transition the TensorFlow model into a TFLite file, optimized for on-device inference on mobile devices. Through the utilization of the tflite_converter, a series of steps were undertaken to load the model, execute requisite functions, and ultimately save it as a TFLite file. Subsequently, the integration of the TFLite model into our flutter application necessitated reliance on the "tflite_flutter" library, which is the sole repository offering updated support for desktop applications. However, the compilation of a Dynamic Link Library (DLL) proved to be a daunting task, owing to the outdated guidelines and discrepancies between systems and library updates. These challenges were addressed by procuring specific versions of Microsoft C++ build tools and CMake, cloning the TensorFlow source code from GitHub, and isolating the TFLite C API for compilation using CMake, tailored to Windows environments.

5.2 DDL construction

The construction of the DLL involved navigating through the integration process with a need for improvisation, given the dated guidelines. Specific versions of Microsoft C++ build tools and CMake were obtained, followed by the cloning of the TensorFlow source code, and the isolation of the TFLite C API for compilation via CMake.

Table 5
Training progress and validation performance of the augmented CNN model

Epoch	Train Loss	Validation Loss	Train accuracy	Validation accuracy
1	0.610516	0.458808	0.658854	0.791667
2	0.533055	0.455064	0.726563	0.791667
3	0.486312	0.503732	0.776042	0.708333
4	0.440375	0.412755	0.804688	0.822917
5	0.454172	0.46243	0.786458	0.78125
6	0.44093	0.474301	0.794271	0.802083
7	0.416042	0.339733	0.822917	0.8125
8	0.357526	0.326436	0.846354	0.864583
9	0.310339	0.225828	0.859375	0.927083
10	0.281315	0.198682	0.882813	0.927083
11	0.22377	0.145097	0.924479	0.947917
12	0.205932	0.176522	0.903646	0.9375
13	0.155941	0.091387	0.960938	0.989583
14	0.136085	0.088728	0.945313	0.96875
15	0.085262	0.045295	0.96875	0.989583
16	0.068302	0.022629	0.973958	1
17	0.042572	0.032427	0.986979	0.989583
18	0.026224	0.034359	0.992188	0.989583
19	0.022774	0.005338	0.994792	1
20	0.013123	0.005644	0.997396	1
21	0.009804	0.003717	0.997396	1
22	0.004549	0.004739	1	1
23	0.004173	0.001293	1	1
24	0.003314	0.002429	1	1
25	0.001897	0.003822	1	1

Crafting the appropriate commands for Windows environments entailed considerable trial and error, ultimately leading to the successful compilation of the DLL.

5.3 Model Integration

The integration of the TFLite model into the application necessitated adherence to guidelines provided by the "tflite_flutter" library, coupled with the importation of the TFLite library and model files to facilitate seamless integration within the application.

5.4 Converting Image Format

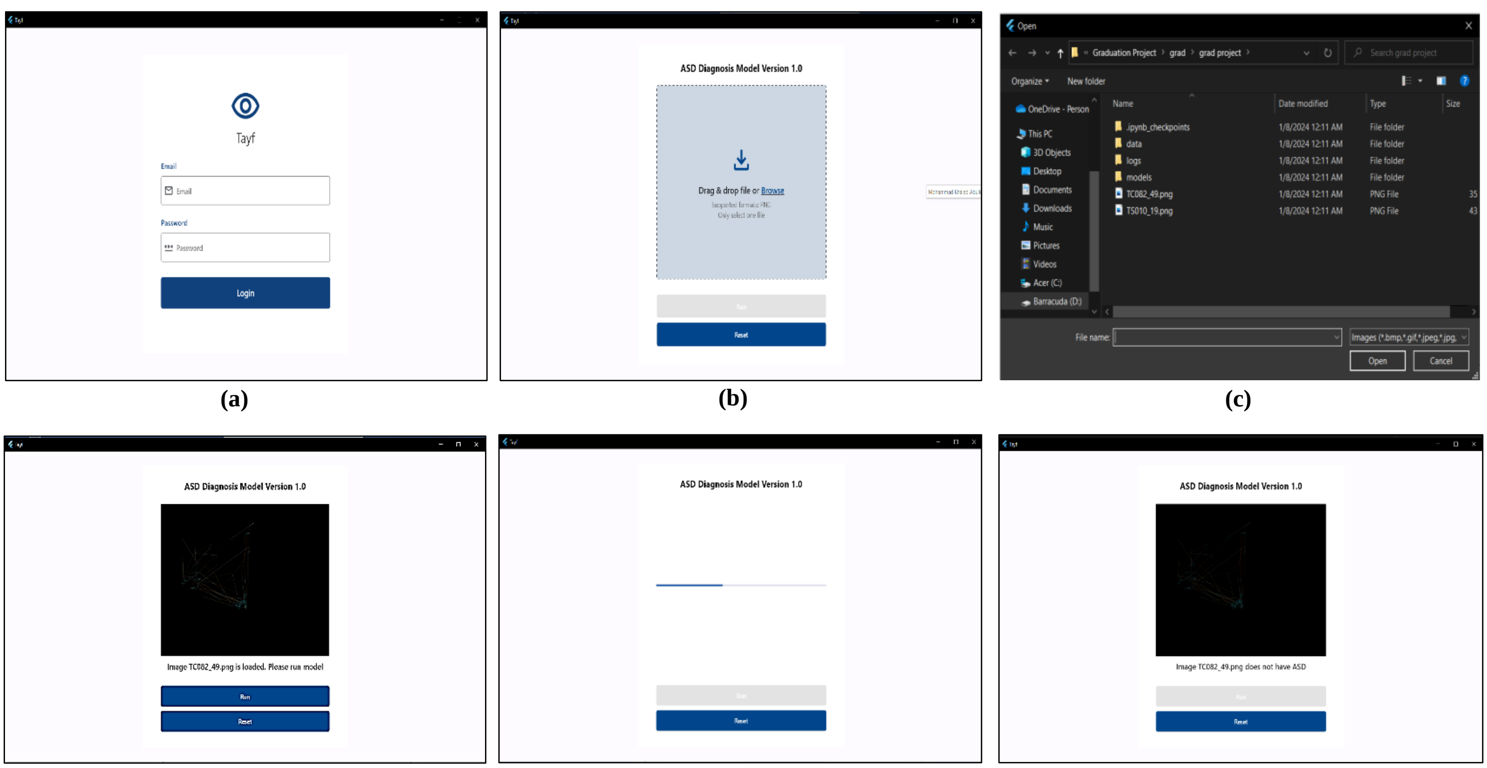
The adaptation of the image format for model compatibility presented substantial challenges. Implementation of a file-picking method and drag-and-drop functionality was undertaken, leveraging external libraries to enhance user experience. Additionally, the utilization of an Image library for PNG image decoding, resizing, and manipulation was employed to align with the requisite model dimensions. Subsequent normalization and manipulation of RGB values facilitated the creation of the necessary input tensor shape for model inference.

5.5 User Interface (UI)

The UI component encompassed a login screen and an interface for image input to the model. Customized window configurations were implemented within the main function to ensure optimal display and functionality.

The login screen, as illustrated in Figure (9.a) showcases a simplistic design devoid of a sign-up function, tailored for research model interface purposes. Access to the model for testing and usage is restricted solely to preconfigured accounts stored within the database.

Upon successful login, users are presented with a model interaction screen facilitating drag-and-drop functionality for image upload (Figure (9.b)) or browsing of the file system (Figure (9.c)). Notably, the run button remains disabled until an



(a) (b) (c)

(d) (e) (f)

Fig. 9. The Discktop Application with the Integrated Augmented CNN model

image is successfully loaded to prevent potential errors. Furthermore, a reset button is incorporated to allow for quick model reset for subsequent testing.

After successful image upload, the image name, and the run button become enabled, as depicted in the figure below (Figure (9.d)). Subsequently, upon initiating model execution, a progress bar is displayed on the screen (Figure (9.e)), with the run button disabled to prevent errors during the execution process. The resulting output is directly presented on the screen, as depicted in the figure below (Figure (9.f)). Once again, the run button is disabled to mitigate potential errors, leaving only the reset button operational for initiating a fresh model execution cycle.

6. Conclusion& Future Work

This research endeavor embarked on a comprehensive exploration of machine learning and deep learning methodologies for the classification of eye-tracking data associated with Autism Spectrum Disorder (ASD). Through the integration of traditional machine learning algorithms and Convolutional Neural Networks (CNNs), this study aimed to discern patterns indicative of ASD-related visual processing abnormalities, thereby contributing to early diagnosis and intervention strategies.

The proposed system methodology encompassed the utilization of meticulously curated datasets, including a longitudinal compilation of eye-tracking biomarker images obtained from participants spanning various developmental stages. Leveraging both traditional ML techniques and deep learning frameworks, our investigation unveiled nuanced insights into the predictive capabilities of these models.

In the realm of traditional ML, Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Classifier (SVC), Decision Tree, and Random Forest algorithms were meticulously trained and evaluated, shedding light on their efficacy in discerning ASD-related patterns. While SVC and Random Forest emerged as top performers, exhibiting robust generalization capabilities, KNN demonstrated notable performance variations, indicating potential areas for improvement.

Furthermore, the exploration of CNN models unveiled the intricate nuances of deep learning methodologies in ASD classification. The training and evaluation of CNN models underscored their ability to discern complex patterns within eye-tracking data, with both original and augmented models exhibiting remarkable performance, as evidenced by high validation accuracies.

The integration of the augmented CNN model into the desktop application "TAYF" marks a significant step towards clinical deployment, offering clinicians a valuable tool for early ASD detection. Despite challenges encountered during model conversion and integration, the successful deployment of the TFLite model within the application heralds promising prospects for real-world implementation.

Overall, this research illuminates the potential of machine learning and deep learning techniques in enhancing our understanding of ASD-related visual processing abnormalities. By leveraging advancements in computational methodologies, we stand poised to revolutionize early ASD diagnosis and intervention strategies, ultimately fostering improved outcomes for individuals on the autism spectrum.

References

[1] Cilia, F., Carette, R., Elbattah, M., Dequen, G., Guérin, J. L., Bosche, J., ... & Le Driant, B. (2021). Computer-aided screening of autism spectrum disorder: eye-tracking study using data visualization and deep learning. JMIR human factors, 8(4), e27706.

[2] Jeyarani, R. A., & Senthilkumar, R. (2023). Eye Tracking Biomarkers for Autism Spectrum Disorder Detection using Machine Learning and Deep Learning Techniques. Research in Autism Spectrum Disorders, 108, 102228.

[3] Thanarajan, T., Alotaibi, Y., Rajendran, S., & Nagappan, K. (2023). Eye-Tracking Based Autism Spectrum Disorder Diagnosis Using Chaotic Butterfly Optimization with Deep Learning Model. Computers, Materials & Continua, 76(2).

[4] Islam, M. F., Manab, M. A., Mondal, J. J., Zabeen, S., Rahman, F. B., Hasan, M. Z., ... & Noor, J. (2024). Involution Fused ConvNet for Classifying Eye-Tracking Patterns of Children with Autism Spectrum Disorder. arXiv preprint arXiv:2401.03575.

[5] Ahmed, Z. A., Albalawi, E., Aldhyani, T. H., Jadhav, M. E., Janrao, P., & Obeidat, M. R. M. (2023). Applying eye tracking with deep learning techniques for early-stage detection of autism spectrum disorders. Data, 8(11), 168.

[6] Alsaidi, M., Obeid, N., Al-Madi, N., Hiary, H., & Aljarah, I. (2024). A Convolutional Deep Neural Network Approach to Predict Autism Spectrum Disorder Based on Eye-Tracking Scan Paths. Information, 15(3), 133.

[7] Kanhirakadavath, M. R., & Chandran, M. S. M. (2022). Investigation of eye-tracking scan path as a biomarker for autism screening using machine learning algorithms. Diagnostics, 12(2), 518.

[8] Mengi, M., & Malhotra, D. (2022). Artificial intelligence based techniques for the detection of socio-behavioral disorders: a systematic review. Archives of Computational Methods in Engineering, 29(5), 2811-2855.

[9] Tang, H., Liang, J., Chai, K., Gu, H., Ye, W., Cao, P., ... & Shen, D. (2023). Artificial intelligence and bioinformatics analyze markers of children's transcriptional genome to predict autism spectrum disorder. Frontiers in Neurology, 14, 1203375.

[10] Sekaran, K., & Sudha, M. (2021). Predicting autism spectrum disorder from associative genetic markers of phenotypic groups using machine learning. Journal of Ambient Intelligence and Humanized Computing, 12(3), 3257-3270.

[11] Elbattah, M., Guérin, J. L., Carette, R., Cilia, F., & Dequen, G. (2020, December). Nlp-based approach to detect autism spectrum disorder in saccadic eye movement. In 2020 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 1581-1587). IEEE.

[12] Asgari, M., Chen, L., & Fombonne, E. (2021). Quantifying voice characteristics for detecting autism. Frontiers in Psychology, 12, 665096.

[13] Ke, F., Choi, S., Kang, Y. H., Cheon, K. A., & Lee, S. W. (2020). Exploring the structural and strategic bases of autism spectrum disorders with deep learning. Ieee Access, 8, 153341-153352.

[14] Almuqhim, F., & Saeed, F. (2021). ASD-SAENet: a sparse autoencoder, and deep-neural network model for detecting autism spectrum disorder (ASD) using fMRI data. Frontiers in Computational Neuroscience, 15, 654315.

[15] Shao, L., Fu, C., You, Y., & Fu, D. (2021). Classification of ASD based on fMRI data with deep learning. Cognitive Neurodynamics, 15(6), 961-974.

[16] Kollias, K. F., Syriopoulou-Delli, C. K., Sarigiannidis, P., & Fragulis, G. F. (2021). The contribution of machine learning and eye-tracking technology in autism spectrum disorder research: A systematic review. Electronics, 10(23), 2982.

[17] Dos Santos Melicio, B. C., Xiang, L., Dillon, E., Soorya, L., Chetouani, M., Sarkany, A., ... & Lorincz, A. (2023, October). Composite AI for Behavior Analysis in Social Interactions. In Companion Publication of the 25th International Conference on Multimodal Interaction (pp. 389-397).

[18] Paolucci, C., Giorgini, F., Scheda, R., Alessi, F. V., & Diciotti, S. (2023). Early prediction of Autism Spectrum Disorders through interaction analysis in home videos and explainable artificial intelligence. Computers in Human Behavior, 148, 107877.

[19] Siddiqui, U. A., Ullah, F., Iqbal, A., Khan, A., Ullah, R., Paracha, S., ... & Kwak, K. S. (2021). Wearable-sensors-based platform for gesture recognition of autism spectrum disorder children using machine learning algorithms. Sensors, 21(10), 3319.

[20] Parui, S., Samanta, D., Chakravorty, N., Ghosh, U., & Rodrigues, J. J. (2023). Artificial intelligence and sensor-based autism spectrum disorder diagnosis using brain connectivity analysis. Computers and Electrical Engineering, 108, 108720.

[21] Visualization of Eye-Tracking Scanpaths in Autism Spectrum Disorder: Image Dataset. Available online: https://figshare.com/articles/dataset/Visualization_of_EyeTracking_Scanpaths_in_Autism_Spectrum_Disorder_Image_Dataset/7073087/1 (accessed on 28 Feb. 2024).

[22] Rahman, M. M., Usman, O. L., Muniyandi, R. C., Sahran, S., Mohamed, S., & Razak, R. A. (2020). A review of machine learning methods of feature selection and classification for autism spectrum disorder. Brain sciences, 10(12), 949.

[23] Helmy, E., Elnakib, A., ElNakieb, Y., Khudri, M., Abdelrahim, M., Yousaf, J., ... & El-Baz, A. (2023). Role of Artificial Intelligence for Autism Diagnosis Using DTI and fMRI: A Survey. Biomedicines, 11(7), 1858.

[24] Li, Y., Qiu, S., Zhong, W., Li, Y., Liu, Y., Cheng, Y., & Liu, Y. (2020). Association between DCC polymorphisms and susceptibility to autism spectrum disorder. Journal of autism and developmental disorders, 50, 3800-3809.

[25] Bao, B., Zahiri, J., Gazestani, V. H., Lopez, L., Xiao, Y., Kim, R., ... & Courchesne, E. (2023). A predictive ensemble classifier for the gene expression diagnosis of ASD at ages 1 to 4 years. Molecular psychiatry, 28(2), 822-833.

[26] Yoon, S. H., Choi, J., Lee, W. J., & Do, J. T. (2020). Genetic and epigenetic etiology underlying autism spectrum disorder. Journal of clinical medicine, 9(4), 966.

[27] Rad, N. M., & Marchiori, E. (2021). Machine learning for healthcare using wearable sensors. In Digital Health (pp. 137-149). Academic Press.