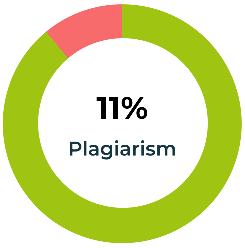
Plagiarism Detector .net

Plagiarism Report



🟩 Unique	89%
🟥 Exact Match	11%
🟦 Partial Match	0%

Primary Sources

1 https://cs231n.stanford.edu/2... 11%

, where N is the total number of labels, k is the number of classes, and nc is the number of samples for class c. How-ever, the resulting weight as shown in Table 1 is too extreme where an event class weight that is 320 times higher than the non-event class.

Excluded URL (s)

01 None

Content

5 Proposed Methodology

The proposed framework integrates eye-tracking scanpath images with participant metadata to create a multimodal predictive model for Autism Spectrum Disorder (ASD) diagnosis. The workflow is divided into several stages: feature extraction, fusion strategy, model architecture, and training and evaluation.

5.1 Feature Extraction

Feature extraction is performed separately for the two modalities: scanpath images and participant metadata.

Image branch: Each scanpath is converted into a grayscale image of resolution 128×128 pixels, encoding temporal gaze dynamics such as fixation density and saccade trajectories. Images are replicated into three channels to match the ImageNet pre-trained CNN (MobileNetV2) input:

$I' \in R128\times128\times3$

The CNN produces a feature embedding:

$fimg \in Rd$

Metadata branch: Features include Age, Gender, and Childhood Autism Rating Scale (CARS) scores. Standardization is applied using z-score normalization:

Z-score Normalization

$z = x - \mu$

σ (3)

where x is the raw feature, μ is the mean, and σ is the standard deviation. The standardized vector $f_{meta} \in R^3$ is passed through dense layers with ReLU activations to create a low-dimensional embedding.

This approach allows the framework to capture both behavioral (scanpath) and demographic-clinical (metadata) characteristics.

5.2 Fusion Strategy

An attention-based fusion mechanism is employed to adaptively weight the modalities. First, features are projected to a common latent space:

Feature Projection

$$p_{img} = \phi(W_{img} f_{img} + b_{img}) \quad (4)$$

$$p_{meta} = \phi(W_{meta} f_{meta} + b_{meta}) \quad (5)$$

where $\phi(\cdot)$ is ReLU, and W, b are trainable parameters.

Attention weights for each modality are computed:

Attention Weights

$$\alpha_{img} = \sigma(w_{img}^T p_{img}) \quad (6)$$

$$\alpha_{meta} = \sigma(w_{meta}^T p_{meta}) \quad (7)$$

Weighted features are obtained as:

5

Weighted Features

$$\hat{f}_{img} = \alpha_{img} \odot p_{img} \quad (8)$$

$$\hat{f}_{meta} = \alpha_{meta} \odot p_{meta} \quad (9)$$

Finally, the fused representation is:

Fused Representation

$$f_{fusion} = [\,\hat{f}_{img} \parallel \hat{f}_{meta}] \quad (10)$$

where $\parallel$ denotes concatenation.

5.3 Architecture

The architecture (Figure 2) has two parallel branches:

1. Image branch: Grayscale scanpath → replicate to 3 channels → MobileNetV2 (frozen) → global average pooling → dropout.
2. Metadata branch: Normalized features → dense layer (16 units, ReLU, L2 regularization) → dropout.

The fused representation is passed through fully connected layers (Dense-32, ReLU, Dropout) before a final sigmoid unit for binary classification (ASD vs. Control). This multimodal approach leverages CNNs for visual features and MLPs for tabular features, integrated via attention fusion.

5.4 Training and Evaluation

The model is trained with the Adam optimizer at learning rate 1×10^{-4}. Binary cross-entropy loss is defined as:

Binary Cross-Entropy Loss

$$L = -\frac{1}{N} \sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \quad (11)$$

Class imbalance is addressed by computing class weights:

Class Weights

$$w_c = \frac{N}{C \times n_c} \quad (12)$$

Data augmentation (flips, rotations, zooms) is applied during training, with early stopping (patience = 5) to prevent overfitting. An 80-20 stratified split is used for training and validation. Evaluation metrics include accuracy, sensitivity, specificity, and AUC.

Figure 2: Proposed Multimodal Framework for ASD Diagnosis. The architecture integrates eye-tracking scanpath images and participant metadata through attention-based fusion. 1) Inputs: Grayscale scanpath images (replicated to 3 channels) and clinical metadata (Age, Gender, CARS). 2) Feature Extraction: MobileNetV2 extracts high-level image embeddings, while dense layers process metadata. 3) Attention-Based Fusion: Modality-specific attention weights (α_{img} and α_{meta}) are learned, producing weighted embeddings. 4) Fusion and Classification: Weighted embeddings are concatenated, passed through fully connected layers with dropout, and a sigmoid activation outputs the ASD probability. This framework adaptively emphasizes the most informative modality, enabling robust and accurate ASD prediction.

Experimental results demonstrate that the multimodal attention-based fusion outperforms unimodal baselines, highlighting the benefit of integrating behavioral and clinical features.

6 Experimental Setup

To rigorously evaluate the proposed multimodal framework for ASD diagnosis, a comprehensive experimental setup was designed. This section details the tools, frameworks, hyperparameters, evaluation metrics, and other implementation specifics.

6.1 Tools and Frameworks

The experiments were conducted using the following software and hardware:

- Programming Language: Python 3.10
- Deep Learning Framework: TensorFlow 2.12 / Keras
- Data Processing and Analysis: NumPy, Pandas, Scikit-learn

- Visualization: Matplotlib, Seaborn
- Hardware: NVIDIA GPU (e.g., RTX 3090), 32GB RAM, Intel Core i9 CPU

These tools provided efficient handling of both image and tabular data for multimodal learning.

6.2 Data Preprocessing

- Scanpath Images: Resized to 128×128 pixels, converted to grayscale, and replicated to 3 channels.
- Metadata: Features (Age, Gender, CARS scores) standardized using z-score normalization.
- Data Augmentation: Random flips, rotations ($\pm 15^{\circ}$), and zooms (±10%) applied on training images to enhance generalization.

6.3 Hyperparameters

The model was trained with the following hyperparameters:

Hyperparameters

- Optimizer: Adam
- Learning Rate: 1×10^{-4}
- Batch Size: 32
- Epochs: 50 (with early stopping pa-

tience = 5)

- Dropout Rate: 0.3 for both branches
- L2 Regularization: 0.01 on dense layers

6.4 Train-Test Split

A stratified 80-20 split was employed to divide the dataset into training and validation sets, ensuring proportional representation of ASD and control classes. Class weights were computed to handle class imbalance:

Class Weights

$$w_c = \frac{N}{C \times n_c} \qquad (13)$$

where N is the total number of samples, C is the
1. number of classes, and nc is the number of samples in class c.

6.5 Evaluation Metrics

The performance of the proposed model was assessed using standard metrics for binary classification:

- Accuracy: Overall correctness of predictions.
- Sensitivity (Recall): Ability to correctly identify ASD cases.
- Specificity: Ability to correctly identify control cases.
- Area Under ROC Curve (AUC): Measures the model's ability to distinguish between classes.

6.6 Implementation Details

- Pre-trained MobileNetV2 weights (ImageNet) were used for feature extraction; convolutional layers were frozen.
- Attention-based fusion layers were trained from scratch.
- Early stopping and dropout were applied to mitigate overfitting.
- Experiments were repeated 5 times with different random seeds to ensure robustness.

References