Article

Not peer-reviewed version

ASD Recognition through Weighted Integration of Landmark-Based Handcrafted and Pixel-Based Deep Learning Features

Asahi Sekine , Abu Saleh Musa Miah * , Koki Hirooka , Najmul Hassan , Md Al Mehedi Hasan , Yuichi Okuyama , Yoichi Tomioka , Jungpil Shin *

Posted Date: 29 December 2025

doi: 10.20944/preprints202512.2505.v1

Keywords: Autism Spectrum Disorder (ASD); image recognition; Convolutional Neural Network (CNN); Squeeze-and-Excitation (SE)

 doi:10.20944/preprints202512.2505.v1

Article

ASD Recognition Through Weighted Integration of Landmark-Based Handcrafted and Pixel-Based Deep Learning Features

Asahi Sekine [1], Abu Saleh Musa Miah [1,*], Koki Hirooka [1], Najmul Hassan [1], Md Al Mehedi Hasan [2], Yuichi Okuyama [1], Yoichi Tomioka [1] and Jungpil Shin [1,*]

[1] School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu, Fukushima 965-8580, Japan
[2] Research Institute for Electronic Science (RIES), Hokkaido University, Japan
* Correspondence: jpshin@u-aizu.ac.jp, musa@u-aizu.ac.jp

Abstract

Autism Spectrum Disorder (ASD) is a neurological condition that impairs communication skills, with individuals often experiencing mild to severe challenges that may require specialised care. While numerous researchers are developing automated ASD recognition systems, achieving high performance remains challenging due to the lack of effective features. In this study, we propose a novel dual-stream model that combines handcrafted facial-landmark features and pixel-level deep learning features to classify ASD and non-ASD faces. The system processes images through two distinct streams to capture complementary features. In the first stream, facial landmarks are extracted using Mediapipe, initially capturing 478 points and selecting 137 symmetric landmarks. The face position is determined by applying in-plane rotation using the angles calculated from the outer eye corners (landmarks 33 and 263). Geometric features and 52 blendshape features are then fed into Dense layers (128 units) with dropout for regularisation. These features are merged and refined through additional Dense layers (128 and 64 units) to produce the final output for Stream-1. In the second stream, the RGB image is resized, normalised using the preprocessing function corresponding to the chosen backbone (e.g., ResNet50V2, DenseNet121, InceptionV3), and then extracted features using a Convolutional Neural Network (CNN) enhanced with Squeeze-and-Excitation (SE) blocks. Global Average Pooling (GAP) reduces dimensionality, followed by DenseNet (256 units with dropout) and a final Dense layer (64 units) to extract features for Stream-2. The outputs from both streams are concatenated, and a softmax gate with weighted concatenation is applied to combine the features. A final Dense layer (128 units with dropout) refines the features before passing them through a softmax layer to produce the probabilistic classification score. This hybrid approach, integrating landmark-based and RGB-based features, significantly enhances the model's ability to distinguish between ASD and Non-ASD faces. Using the Kaggle dataset, the model achieved an accuracy of 96.43%, with a precision of 97.10%, recall of 95.71%, and an F1 score of 96.40%. On the YTUIA dataset, the accuracy increased to 97.83%, with a precision of 97.78%, recall of 97.78%, and an F1 score of 97.78%. Although these results are promising, they fall short of surpassing the highest reported performance of 95.00% for Kaggle and 95.90% for YTUIA. Future work will focus on optimizing the model's performance to exceed these benchmarks.

Keywords: Autism Spectrum Disorder (ASD); image recognition; Convolutional Neural Network (CNN); Squeeze-and-Excitation (SE)

1. Introduction

ASD (Autism Spectrum Disorder) is a developmental disorder characterized by difficulties in interpersonal and social communication, as well as traits such as rigidity and fixation on specific objects or behaviors, and sensory sensitivity or dullness, which can be observed from early childhood and cause challenges in daily life [1–3]. The diagnostic methods for ASD include "clinical interviews,"

 doi:10.20944/preprints202512.2505.v1

"behavioral observations," and "psychological or intelligence tests." In some cases, physiological tests may also be conducted, and the diagnosis is made when the results meet medical criteria [4]. However, early detection is effective for intervention and support. Just as biases related to gender, ethnicity, and race can affect the screening and diagnosis of ASD, it is believed that methods free from subjectivity and prejudice can be effective. Therefore, Traditional-based and deep learning are utilized. The traditional-based and DL can be used to make unbiased diagnoses without interventions by analyzing children's facial data. Traditional diagnostic methods for ASD, which rely on behavioral assessments and expert interviews, are considered the gold standard. However, these methods often suffer from subjectivity and bias, underscoring the need for more objective and efficient diagnostic tools [4]. Deep learning offers a promising solution, as it can identify complex patterns and data representations that are difficult for humans to detect, thus automating and enhancing the diagnostic process [5]. For instance, deep learning algorithms applied to neuroimaging data such as fMRI [6] and EEG [7] have helped identify neurological differences linked to autism. Despite their precision, these methods are expensive and require specialized professionals. In contrast, facial image analysis provides a simple and effective tool for early ASD screening, eliminating the need for expert intervention and reducing costs. Facial image datasets have proven to be a valuable resource for deep learning algorithms, offering numerous benefits for building accurate ASD diagnostic models [8,9]. While traditional methods are valuable, their subjectivity and limitations highlight the importance of more objective, automated tools for early detection. Active learning, a machine learning technique that selects the most informative data for annotation, is proving to be particularly effective in improving diagnostic models. By identifying subtle facial expressions or behavioral patterns indicative of ASD, active learning can lead to more accurate diagnoses and allow for timely interventions during critical developmental phases [10,11]. However, we were inspired by a deep learning base approach to propose a hybrid model that utilizes the efficacy of deep learning models like CNN, which has increased in capturing morphological differences in the faces of children with ASD compared to those without. There is a growing body of research focusing on Autism Spectrum Disorder (ASD) recognition using various modalities such as EEG, video, facial images, and individual landmarks. However, most of these efforts have struggled to achieve satisfactory performance, largely due to the lack of effective and complementary feature sets. Furthermore, to the best of our knowledge, no existing work has combined both facial image-based pixel features and landmark-based features for ASD recognition. The contributions of the proposed model are outlined as follows:

1. **Dual-Stream Model:** We propose a novel dual-stream model that combines facial landmark-based handcrafted features and pixel-based deep learning features for classifying ASD and Non-ASD faces. This innovative combination integrates landmark-based features and RGB image-based features, capturing complementary information from both feature types. By leveraging both geometric and pixel-based data, the model significantly enhances its ability to distinguish between ASD and Non-ASD faces, improving classification accuracy and robustness.
2. **Two-Stream Description:** The first stream extracts geometric and blendshape features using Mediapipe, processing the extracted landmarks through dense layers. In contrast, the second stream processes RGB images, which are normalised using the preprocessing function corresponding to the chosen backbone (e.g., ResNet50V2, DenseNet121, InceptionV3), then extracted features with a Convolutional Neural Network (CNN) enhanced with Squeeze-and-Excitation blocks and DenseNet layers to capture high-level image features.
3. **Feature Concatenation:** After processing the features from both streams, the outputs are concatenated, merged, and refined. A final softmax layer is used to classify the ASD and Non-ASD faces, ensuring that both feature sets contribute to the final decision.
4. **Experimental Results:** On the Kaggle dataset, the model achieves an accuracy of 96.43%, precision of 97.10%, recall of 95.71%, and an F1 score of 96.40%. For the YTUIA dataset, the model performs even better, reaching an accuracy of 97.83%, precision and recall of 97.78%, and an F1 score of 97.78%.

 doi:10.20944/preprints202512.2505.v1

5. **Novelty and Advantages:** The novelty of this approach lies in the integration of both facial landmark and RGB pixel features, which significantly improves ASD classification. This model enhances classification performance, increases robustness, and provides complementary feature integration, making it a promising solution for ASD recognition.

2. Related Work

Autism Spectrum Disorder (ASD) is a neurodevelopmental disorder characterized by social communication difficulties and restricted, repetitive behaviors. Early diagnosis of ASD, similar to Parkinson's Disease (PD) and Alzheimer's Disease (AD), is crucial for effective intervention. In recent years, the use of facial image datasets has gained attention for ASD detection. Deep learning models, particularly Convolutional Neural Networks (CNNs), have shown promise in analyzing facial images to detect subtle morphological features associated with ASD. For instance, these models can identify facial asymmetry, which is often present in individuals with ASD [12,13]. Similarly, in Parkinson's Disease, changes in facial expressions due to motor impairments have been analyzed using CNNs and recurrent neural networks (RNNs) for early detection [14–17]. For Alzheimer's Disease, facial recognition technologies combined with other data sources have been used to monitor cognitive decline, detecting reduced emotional expression as a sign of progression [18–22]. Multimodal approaches, combining facial features with EEG or speech data, have also been explored to enhance diagnostic accuracy for ASD, PD, and AD [12]. These advances demonstrate the potential of deep learning in the non-invasive diagnosis of neurodevelopmental and neurodegenerative disorders. Studies like those by [23,24] have employed facial asymmetry as a key feature for ASD detection, achieving high accuracy using deep learning algorithms. These studies used 3D blend shapes to analyze facial asymmetry, quantifying differences in key facial regions such as the eyes, eyebrows, and mouth. However, one limitation of existing studies is their reliance on single-domain datasets, which may not generalize well to new datasets or populations. Domain adaptation techniques have been proposed to address this challenge. These techniques allow models to transfer knowledge learned from one dataset to another, enabling the model to perform well on data from diverse sources. Research by [25] demonstrated the effectiveness of domain adaptation in improving model robustness when applied to various medical imaging tasks, including ASD detection. Another approach that has gained popularity is active learning. This method focuses on selecting the most informative data for labeling, reducing the need for large labeled datasets. Active learning has been successfully applied in many medical image classification tasks, including ASD detection, where it helps improve model performance while minimizing the annotation effort. Studies like [10,11] have shown that combining active learning with deep learning models can enhance the accuracy and efficiency of ASD diagnostic systems. Recent work has emphasized combining active learning and domain adaptation in ASD detection. Active learning can help address data variability across different clinical settings by selecting samples that improve model performance [26]. Domain adaptation, on the other hand, allows the model to generalize better to diverse datasets, ensuring its robustness in real-world applications.

Several studies have developed DL–based methods [27–29] for ASD identification using facial images, with most works relying on the Kaggle ASD dataset. Taher M. Ghazal et al. [30] proposed a modified AlexNet-based model, termed ASDDTLA, which reported a comparatively lower accuracy of 87.7%. Furthermore, Alam et al. [27] conducted a systematic ablation study to optimize network architectures, optimizers, and hyperparameters. Using the Xception model with an optimal parameter configuration, they achieved a maximum accuracy of 95%. More recently, Narinder Kaur et al. [31] and M. Ikermane et al. [32] reported accuracies of 70% and 98%, respectively. Across these studies, CNN models pretrained on the ImageNet dataset were predominantly used to extract discriminative facial features from the Kaggle ASD dataset, underscoring the effectiveness of transfer learning for ASD facial image classification.

 doi:10.20944/preprints202512.2505.v1

3. Dataset Description

In this study, we propose a hybrid ASD recognition framework that combines deep features from facial images with handcrafted features derived from facial landmarks detected using the MediaPipe FaceMesh model. We quantify facial asymmetry by calculating the differences between symmetric landmark pairs across key facial regions. The fusion of deep and geometric features is aimed at improving classification performance while enhancing interpretability and robustness. We base our approach on a dataset that includes facial images of individuals with autism and normal controls. The dataset enables us to explore the relationship between facial asymmetry and ASD, which can improve the diagnostic capabilities of the model.

3.1. Dataset 1 (Kaggle)

The first dataset was collected from Kaggle and contains facial images of children with autism. It consists of 2D RGB images of children aged between 2 and 14 years, with most samples belonging to the 2–8 year age group. The dataset shows an overall male-to-female ratio of approximately 3:1; however, the distribution between the ASD group and the normal control (NC) group is balanced at nearly 1:1. The data are divided into training, testing, and validation sets, accounting for 86.38%, 10.22%, and 3.41% of the samples, respectively. Each split preserves an equal proportion of ASD and NC images. The dataset was compiled by Gerry Piosenka using publicly available online sources and does not include additional demographic or clinical information such as ethnicity, socioeconomic background, medical history, or ASD severity.

3.2. YTUIA-YouTube Dataset

The second dataset, named YTUIA, was collected at Universiti Islam Antarabangsa and is based on YouTube videos. It includes videos derived from the Self-Stimulatory Behaviours Dataset (SSBD), a widely used resource for autism research. Although only 50 relevant videos were directly available on YouTube, additional videos were identified with the help of therapists and specialized institutions, resulting in a total of 100 YouTube videos used for frame extraction. For the normal control group, videos showing kindergarten activities within the same age range were selected from YouTube. In the initial stage, faces were detected in each video frame using the MTCNN algorithm. This was followed by a preprocessing process that involved face alignment, cropping, and resizing. The normal control group consisted of 173 unique children, including 117 males and 56 females, aged between 1 and 11 years. The ASD group contained 123 individuals, with 93 males and 30 females, whose ages ranged from 3 to 11 years. To ensure balanced learning, the dataset was divided into a training set of 1,068 samples and a test set of 100 samples, maintaining a 1:1 ratio between ASD and normal control participants.

4. Proposed Methodology

The proposed workflow architecture is shown in Figure 1. The proposed methodology for face-based ASD recognition uses a two-phase approach: training and testing. In the training phase, input images from both ASD and Non-ASD subjects are preprocessed, including resizing and normalization to standardize the image data. Feature extraction is performed through two distinct streams to capture complementary features: Facial landmarks are extracted using Mediapipe, with 478 points initially captured and 137 symmetric landmarks selected. Geometric features and 52 blendshape features are generated from these landmarks. First, we fixed the face position by applying an in-plane rotation using the eye corner angles calculated from the outer eye corners (landmarks 33 and 263).

These features are then processed through Dense layers (128 units) with dropout for regularization. The feature sets are concatenated and refined with additional Dense layers (128 and 64 units) to produce the final output. In the second stream, RGB images are resized and normalized, then processed by a Convolutional Neural Network (CNN) augmented with Squeeze-and-Excitation (SE) blocks. Global Average Pooling (GAP) reduces the dimensionality, followed by DenseNet (256 units with dropout)

 doi:10.20944/preprints202512.2505.v1

and a final Dense layer (64 units) to extract the features for Stream-2. Once features are extracted from both streams, they are concatenated, and a softmax gate with weighted concatenation combines them. A final Dense layer (128 units with dropout) refines the features before passing them through a softmax layer to produce the probabilistic classification score. This hybrid approach, integrating both landmark-based and RGB-based features, significantly enhances the model's ability to distinguish between ASD and non-ASD faces. In the testing phase, the process mirrors the training phase: input images are preprocessed, features are extracted from both streams, modality-gated fusion is applied, and the trained model outputs the prediction. Finally, we evaluate an ensemble of multiple backbones by averaging predicted probabilities or by majority vote with validation-optimized thresholds. This methodology ensures robust classification by combining deep learning with geometric and blendshape facial features for ASD recognition.

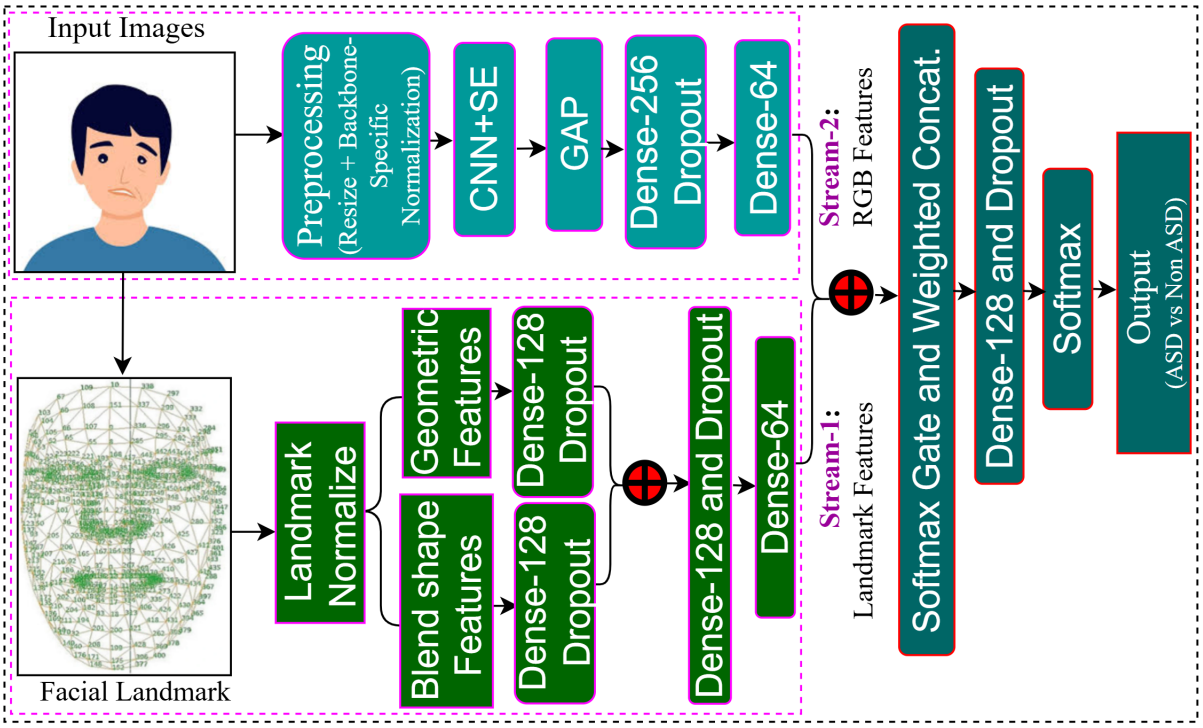

Figure 1. Overview of the proposed two-stream architecture. In Stream-1, face alignment is performed using in-plane rotation based on outer eye corner landmarks (33 and 263) to ensure consistent landmark positioning, followed by extraction and concatenation of geometric and blendshape features. In Stream-2, input images are resized and normalized using backbone-specific preprocessing (ResNet50V2, DenseNet121, and InceptionV3), then processed through a CNN with SE blocks and GAP to obtain deep features. Features from both streams are fused and classified using a softmax layer.

4.1. Preprocessing

In this study, we employed preprocessing for two streams: RGB-based facial images and facial landmarks, which were extracted using Mediapipe. The details of the preprocessing are provided below

4.1.1. RGB Based Preprocessing (Resizing and Normalization)

Given an input image $I \in \mathbb{R}^{H\times W\times C}$, where, H is the height of the image, W is the width of the image, and C = 3 for RGB images. The first step is resizing the image to a standard size of 224 × 224. Each input image is resized to $I' \in \mathbb{R}^{224\times 224\times 3}$, so that the image dimensions are consistent for the network. This resizing ensures that all input images have the same shape, making them compatible for model processing. The image is then normalized using the preprocessing function corresponding to the chosen backbone (e.g., ResNet50V2, DenseNet121, InceptionV3). This step standardizes the pixel

 doi:10.20944/preprints202512.2505.v1

values of the image I', ensuring that they are in a range suitable for the pretrained model's weights. The normalized image I'' is now ready for feature extraction:

$$I'' = \text{Preprocess}_{\text{backbone}}(I')$$

4.1.2. Facial Landmark Based Processing and Selected 137 Symmetric Landmarks

We applied Mediapipe's FaceMesh to extract facial landmarks from the input image. A total of 478 landmarks are initially detected, which includes key facial points such as those around the eyes, eyebrows, nose, and lips. These landmarks provide detailed information about the geometry of the face.

Out of the 478 landmarks, we selected 137 points from symmetric facial areas, such as the eyes with eyebrows, the nose, and the lips. These landmarks were chosen for their relevance in distinguishing subtle features between ASD and Non-ASD faces.

In addition to the facial landmarks, we extract handcrafted features from the detected landmarks and expression-related blendshapes using Mediapipe Face Landmarker v2. These blendshapes model facial expressions and contribute further discriminative information to the feature set.

The facial landmarks are represented as:

$$\mathbf{lms} = \{(x_i, y_i, z_i)\}_{i=1}^{478}$$

where (x_i, y_i, z_i) represent the 3D coordinates of the i-th landmark. After extracting the landmarks, we use the 137 symmetric points to compute geometric features (e.g., distances and angles between points) and blend shape features, which are then used for classification. These features, alongside RGB-based features from the second stream, are merged to form the final feature set for classification.

4.2. Stream-1: Facial Landmark-Based Features

In Stream-1 , first, we fixed the face position by applying an in-plane rotation using the eye corner angles calculated from the outer eye corners (landmarks 33 and 263). This ensures that all facial landmarks are in a consistent position for feature extraction, regardless of initial face orientation. The in-plane rotation angle θ is derived from the eye-line and used to rotate the face. The face alignment is achieved by applying trigonometric functions $\sin(\theta)$ and $\cos(\theta)$ to correct the face position. This alignment step is crucial for ensuring that the extracted features are consistent across various face orientations.

4.2.1. Geometric Features

The face is aligned based on the eye positions by rotating the face with an in-plane rotation angle θ, calculated as:

$$\theta = \text{atan2}(y_{263} - y_{33},\ x_{263} - x_{33}), \tag{1}$$

where (x_{33}, y_{33}) and (x_{263}, y_{263}) are the coordinates of the outer canthi of the eyes. After rotation, the face is aligned, and we append $[\sin(\theta), \cos(\theta)]$ to the feature set to capture the corrected face orientation.

Next, we compute geometric features by calculating the 3D and 2D distances between key facial points. The 3D distance between two points a and b is given by:

$$D_{3D}(a, b) = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2 + (z_a - z_b)^2}, \tag{2}$$

while the 2D distance is calculated as:

$$D_{2D}(a, b) = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}. \tag{3}$$

These distances help capture facial features such as face width and height.

 doi:10.20944/preprints202512.2505.v1

To align facial symmetry, the landmarks are scaled by the inter-outer-canthi distance d, which is calculated as:

$$d = \sqrt{(x_{33} - x_{263})^2 + (y_{33} - y_{263})^2}, \tag{4}$$

After this scaling, the landmarks are rotated to align the eye-line horizontally, and the face is centered by subtracting the mid x-coordinate between the two outer canthi.

For each symmetric pair of landmarks (a, b), we compute the differences in the x- and y-coordinates, as well as their absolute values:

$$[\Delta x, \Delta y, |\Delta x|, |\Delta y|], \tag{5}$$

and when depth information is available, we extend the calculation to the z-coordinates:

$$[\Delta z, |\Delta z|]. \tag{6}$$

For each facial part (eyes, brows, lips, nose, contour), we compute several statistics, including the center, variance, range, and the principal direction (via the leading eigenvector of the 2D covariance matrix). These are used to generate an 8-dimensional feature vector for each part. For the eyes and brows, left-right differences are used, while for the lips, nose, and contour, single-part summaries are computed.

To capture symmetry errors, we reflect the x-coordinates of a subset of points (i.e., $x \to -x$) and calculate the nearest-neighbor distance to the original set. The mean, standard deviation, and 90th percentile of these distances are aggregated to form additional features:

$$\text{Mirror-Symmetry Error} = \text{mean}, \text{std}, \text{90th percentile}. \tag{7}$$

4.2.2. Blendshape Features

In addition to the geometric features, we extract blendshape features that represent facial expressions, such as a smile or a frown, which are indicative of emotional states or behavior. These features are computed from the detected landmarks and capture the dynamics of facial movements. If the blendshape model asset is unavailable, the blendshape features are replaced with a zero vector of the target dimensionality to ensure consistency across all images. The blendshape features are normalized using a reference distance d, which is calculated between landmarks 33 and 263, as described in equation 4. The x- and y-coordinates of the landmarks are normalized by this reference distance to standardize facial position and scale. Additionally, the face is rotated by an angle θ, computed in equation 1, to align the face consistently across different images. Finally, the landmarks are translated so that the midpoint between landmarks 33 and 263 is centered at zero in the x-coordinate, ensuring consistent positioning of facial landmarks. For each pair of symmetric blendshape indices, to quantify expression differences between the left and right sides of the face for each pair of symmetric blendshape indices, we derive from the 52 scores (range 0–1) produced by MediaPipe Face Landmarker v2 the following features: for each left–right pair, (diff, To quantify expression differences between the left and right sides of the face for each pair of symmetric blendshape indices, we derive from the 52 scores (range $0-1$) produced by MediaPipe Face Landmarker v2 the following features: for each left–right pair

$$\text{diff} = s_L - s_R, \qquad |\text{diff}| = |s_L - s_R|, \qquad \text{sum} = s_L + s_R, \qquad \text{max} = \max(s_L, s_R),$$

for each individual key

$$v, \quad v^2, \quad \mathbf{1}[v > 0.5],$$

where $\mathbf{1}[\cdot]$ denotes the indicator function, and group aggregations (mean, max, variance) are computed for the eyes, brows, mouth, jaw, and nose.

 doi:10.20944/preprints202512.2505.v1

4.2.3. Concatenated the Geometric and Blenshape Features

After extracting both geometric and blendshape features, we concatenate these features from the landmark branch and then fuse them with the image branch via a learned modality gate:

$$F_{\text{LM}} = \text{Concatenate}\big(F_{\text{geom}}, F_{\text{blend}}\big), \quad F_{\text{fused}} = \text{Gate}\big(F_{\text{CNN}}, F_{\text{LM}}\big) \tag{8}$$

where the "Gate" refers to a soft attention mechanism that generates modality weights and combines the two modalities into a weighted sum.

After processing the facial landmarks and blendshape features, the resulting features are passed through Dense layers to produce the final feature set for Stream-1. The concatenation and fusion of these features allow the model to leverage both geometric and expression-based information for improved classification.

4.3. Stream-2: RGB-Based Feature Extraction with CNN Backbones

In Stream-2, we process the RGB image I, which is resized and normalized to a standard size and range, as described in the RGB-based preprocessing section. The image I'' after normalization is passed through a convolutional neural network (CNN) backbone, such as ResNet50V2, DenseNet121, or InceptionV3, all of which are pretrained on ImageNet. The top (classification) layers of the pretrained model are removed to extract features from the convolutional layers.

Resizing and Normalization: The first step in Stream-2 involves resizing the image to a fixed size (e.g., 224×224) and normalizing the pixel values. This ensures that the image has a consistent dimension and range, making it suitable for input into the CNN backbone:

$$I'' = \text{Preprocess}(I') \tag{9}$$

where I' is the resized input image and I'' is the normalized image ready for further processing.

CNN Feature Extraction: The normalized image I'' is then passed through a pretrained CNN backbone. The network is used for feature extraction, leveraging the learned filters and weights from ImageNet. This produces feature maps from the convolutional layers:

$$F_{\text{CNN}} = \text{Backbone}(I'') \tag{10}$$

where F_{CNN} represents the raw feature maps obtained from the CNN backbone. These feature maps capture high-level semantic information about the input image.

Squeeze-and-Excitation (SE) Block: To enhance the representational capacity of the CNN, we apply a Squeeze-and-Excitation (SE) block to recalibrate the channel-wise feature responses. The SE block performs global average pooling (GAP) on the feature maps and applies a lightweight attention mechanism to adaptively recalibrate the features:

$$F_{\text{SE}} = \text{SE}(\text{GAP}(F_{\text{CNN}})), \tag{11}$$

where $\text{GAP}(F_{\text{CNN}})$ computes the global average pooling of the feature maps F_{CNN}, and F_{SE} is the recalibrated feature set.

Global Average Pooling (GAP): Next, the output of the SE block undergoes Global Average Pooling (GAP), which reduces the spatial dimensions of the feature maps into a single vector per channel:

$$\text{GAP}(F_{\text{SE}}) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_{\text{SE}}(i, j), \tag{12}$$

where H and W are the height and width of the feature map, and $F_{\text{SE}}(i, j)$ are the values at each spatial location. GAP summarizes the spatial information into a single vector.

 doi:10.20944/preprints202512.2505.v1

Dense Layer Projection: The pooled feature vector is then passed through a Dense layer with 256 units and dropout for regularization:

$$F_{\text{Dense}} = \text{Dense}(F_{\text{GAP}}, 256) \quad \text{with dropout}, \tag{13}$$

where F_{Dense} is the output of the Dense layer, and dropout is applied to prevent overfitting.

Finally, a Dense layer with 64 units is applied to reduce the dimensionality and produce the final output features for Stream-2:

$$F_{\text{Stream-2}} = \text{Dense}(F_{\text{Dense}}, 64), \tag{14}$$

where $F_{\text{Stream-2}}$ is the final feature set representing the processed RGB image, ready to be fused with Stream-1 for the final classification.

In Stream-2, we used a CNN backbone to extract features from the normalized RGB image. The SE block enhances the feature map representation, and GAP summarizes the spatial features into a global feature vector. A Dense layer reduces the dimensionality, and the final Dense layer produces the Stream-2 feature set. These features are then concatenated with the features from Stream-1 for final classification.

4.4. Feature Concatenation and Classification Module

In the classification module, the features from Stream-1 (landmark features) and Stream-2 (RGB features) are concatenated to form a unified feature vector. This concatenated feature vector is then processed through a softmax gate, which applies a weighted combination of the two modalities, allowing the model to learn the optimal contribution of each modality (landmark and RGB features). The fusion of these two complementary feature sets enables the model to better distinguish between ASD and Non-ASD faces.

The fused feature vector is passed through fully connected layers for further refinement, with a Dense layer having 128 units and dropout applied for regularization:

$$\mathbf{y} = \text{Softmax}(WF_{\text{fused}} + \mathbf{b}), \quad \mathbf{y} \in \mathbb{R}^2 \tag{15}$$

where W is the weight matrix, F_{fused} is the concatenated feature vector, and $\mathbf{b}$ is the bias term. The softmax activation produces a probabilistic score for each class, with the two possible output classes being ASD and Non-ASD.

The final class label is determined based on the threshold τ, which is optimized on the validation set:

$$\hat{y} = \begin{cases} \text{ASD} & \text{if } y_{\text{ASD}} \geq \tau \\ \text{Non-ASD} & \text{otherwise} \end{cases} \tag{16}$$

where y_{ASD} represents the probability of the ASD class, and τ is a threshold value chosen to optimize classification performance.

For multi-backbone ensembles, the model outputs from different CNN backbones are aggregated by probability averaging or majority vote, using validation-optimized thresholds to improve robustness and classification accuracy.

5. Experimental Evaluation

We conducted experiments to evaluate the performance of the proposed autism classification model based on facial image analysis. The dataset consists of two classes: ASD (Autistic), which includes facial images of diagnosed individuals, and Non-ASD, which contains facial images of individuals without ASD. The objective was to assess how well the model can differentiate between these two classes using deep learning techniques applied to facial images. The dataset was split into training and test sets at an 80/20 ratio. Each image was resized to 224×224 pixels to ensure consistency in input size. The optimizer used for training was Adam, with binary cross-entropy as the loss function.

 doi:10.20944/preprints202512.2505.v1

A batch size of 32 was selected for training, and the model was trained for a set number of epochs, depending on the dataset used for evaluation.

5.1. Environmental Setting

The experimental evaluation focused on the performance of the proposed autism classification model using facial image analysis. The dataset consisted of two classes: ASD (Autistic) and Non-ASD, where each class contains facial images from individuals diagnosed with ASD and those without the condition, respectively. The dataset was split into training, validation, and testing sets, with an 80/20 split for training and testing. The Kaggle dataset was used with its inherent structure, while the YTUIA dataset was split using a stratified 80/20 split to ensure consistent class distribution across both subsets.

Each image was resized to a consistent input size of 224×224 pixels, and the corresponding RGB image tensor was normalized to a range of $[0, 1]$. Along with the image data, each sample included a vector of facial landmark-based features, which incorporated both geometric features (e.g., distances and angles between key points) and blendshape features (representing facial expressions). The model was trained with the Adam optimizer, using a learning rate scheduler, and checkpoints were saved based on validation accuracy to mitigate overfitting.

5.2. Hyperparameter Setting

The model was trained using the Adam optimizer with an initial learning rate of 1×10^{-4} and a batch size of 32. For training, the Kaggle dataset was run for 100 epochs, while the YTUIA dataset was trained for 30 epochs. Binary cross-entropy was chosen as the loss function to suit the binary classification task. To enhance training efficiency and prevent overfitting, a learning rate scheduler was implemented using the ReduceLROnPlateau callback, which adjusts the learning rate when the validation accuracy plateaus. Additionally, the ModelCheckpoint callback was used to save the model with the best validation accuracy, ensuring that the most optimal model was retained during training. The model also employed feature extraction methods that included both facial landmark-based geometric and blendshape features, as well as CNN-based features derived from the RGB image data.

5.3. Performance Metrics

The performance of the model was assessed using several key metrics, including accuracy, the Receiver Operating Characteristic (ROC) curve, and the Area Under the Curve (AUC). Accuracy measures the proportion of correctly predicted instances relative to the total number of instances. The ROC curve was used to visualize the model's diagnostic performance across classification thresholds, while the AUC score provided a single numerical summary of performance across all thresholds. Threshold optimization was performed by sweeping thresholds on the validation set to determine the optimal classification threshold, particularly for ensemble models. For ensemble predictions, final class labels were determined by combining individual model predictions through probability averaging or majority voting, with the thresholds optimized per model to enhance overall classification performance.

5.4. Experimental Result with Kaggle Dataset

Table 1 presents the results of the binary classification task using the Kaggle dataset. The proposed model was trained for 100 epochs with a threshold of 0.85. The model achieved an impressive accuracy of 96.43%, with a precision of 97.10%, a recall of 95.71%, and an F1 score of 96.40%. These results underscore the importance of proper feature alignment, particularly the inclusion of asymmetry and facial part information, in enhancing the model's performance. The model's ability to efficiently distinguish between ASD and non-ASD faces is evident; however, further optimization through model parameter tuning or additional training epochs may lead to even better performance. In comparison, Stream 1, which used only landmarks for training, achieved a lower accuracy of 73.57%, with precision

 doi:10.20944/preprints202512.2505.v1

at 78.45%, recall at 65.00%, and an F1 score of 71.09%. While the performance was acceptable, it highlights the necessity of incorporating additional features for more accurate classification. Stream 2, which used only landmarks for RGB input, achieved substantial improvements, achieving an accuracy of 95.00%, precision of 95.65%, recall of 94.29%, and an F1 score of 94.96%. This result emphasizes the advantage of integrating RGB input in training, which significantly improved the model's ability to classify ASD and non-ASD faces accurately. These findings collectively demonstrate the effectiveness of the proposed method and the potential improvements that can be made by exploring additional features and training strategies.

Table 1. Experimental Result and SOTA Comparison with Kaggle Dataset

Model	Acc.	Prec.	Recall	F1 Score	Epochs	Threshold	Comments
Stream 1	73.57	78.45	65.00	71.09	100	-	Accuracy achieved with only landmarks for training.
Stream 2	95.00	95.65	94.29	94.96	100	-	Accuracy achieved with only landmarks for RGB input.
Proposed Method	96.43	97.10	95.71	96.40	100	0.85	Efficient in classifying ASD and Non-ASD faces, with potential for further improvement through parameter tuning or increased epochs.

5.5. Experimental Result with YTUIA Dataset

Table 2 presents the results of the binary classification task using the YTUIA dataset. The model was trained for 30 epochs with a threshold of 0.35. The proposed method achieved an impressive accuracy of 97.83%, with precision, recall, and F1 score all reaching 97.78%. This balanced performance across all metrics demonstrates the model's robustness and its ability to efficiently classify ASD and Non-ASD faces. The higher performance on the YTUIA dataset compared to the Kaggle dataset suggests that the model benefits from a more diverse or representative set of facial images, which allows it to capture more discriminative features between ASD and Non-ASD individuals. In comparison, Stream 1, which used only landmarks for training, achieved a lower accuracy of 72.83%, with precision at 76.32%, recall at 64.44%, and an F1 score of 69.88%. While the performance was relatively good, it highlights the potential for improvement by incorporating additional features or adjusting the training strategy. Stream 2, which utilized only landmarks for RGB input, achieved an accuracy of 88.00%, with precision at 91.30%, recall at 84.00%, and an F1 score of 87.50%. This result indicates that while RGB input provides some benefit, it is still less effective compared to the proposed method that uses a combination of features for classification. These results collectively demonstrate the effectiveness of the proposed method in distinguishing between ASD and Non-ASD faces, with potential for further enhancement through the exploration of additional features or more intensive training strategies.

Table 2. Experimental Result with YTUIA Dataset

Model	Acc.	Prec.	Recall	F1 Score	Epochs	Threshold	Comments
Stream 1	72.83	76.32	64.44	69.88	30	-	Accuracy achieved with only landmarks for training.
Stream 2	88.00	91.30	84.00	87.50	30	-	Accuracy achieved with only landmarks for RGB input.
Proposed Method	97.83	97.78	97.78	97.78	30	0.35	Demonstrates balanced performance across all metrics with a higher threshold for improved classification.

5.6. Comparison with State-of-the-Art Methods

As summarized in Table 3, recent state-of-the-art methods for facial image–based ASD classification mainly rely on single-stream convolutional neural networks, active learning–based domain

 doi:10.20944/preprints202512.2505.v1

adaptation, or deep ensemble frameworks to address cross-domain variability between datasets such as Kaggle and YTUIA. Although these approaches report high classification accuracy, performance evaluation is often limited to accuracy alone, providing insufficient insight into class-wise reliability and prediction balance. In contrast, the proposed method explicitly integrates landmark-based geometric and facial asymmetry features with RGB deep representations, enabling complementary structural and appearance modeling. This design yields consistently high and balanced precision, recall, and F1-score on both Kaggle (96.43%) and YTUIA (97.83%) datasets, demonstrating stable and reliable discrimination performance without dependence on complex ensemble architectures or extensive domain adaptation procedures.

Table 3. Comparison with State-of-the-Art (SOTA) Methods for ASD Classification Using Facial Images

Method	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Alam et al. (2024) [26]	Kaggle	95.00	95.00	–	94.00
Alam et al. (2024) [26]	YTUIA	95.90	95.90	–	95.90
Alam et al. (2025) [28]	Kaggle + YTUIA	96.00	96.00	96.00	96.00
Proposed Method (Ours)	**Kaggle**	**96.43**	**97.10**	**95.71**	**96.40**
Proposed Method (Ours)	**YTUIA**	**97.83**	**97.78**	**97.78**	**97.78**

6. Conclusion

In this study, we propose a novel framework for classifying Autism Spectrum Disorder (ASD) from facial images by integrating deep learning with explicitly extracted asymmetry-based features. Leveraging Google MediaPipe, we extracted facial landmarks and computed quantitative asymmetry metrics between the left and right sides of key facial regions, including the eyes, nose, and mouth. These asymmetry features were then combined with high-level visual representations learned via deep learning to enhance both classification performance and interpretability. The proposed dual-stream model processes both RGB images and facial landmarks through separate streams, capturing complementary information. In the first stream, facial landmarks are used to extract geometric and blendshape features, while the second stream processes RGB images using a Convolutional Neural Network (CNN) enhanced with Squeeze-and-Excitation blocks. The outputs from both streams are concatenated and refined for classification through a softmax layer, which ensures robust decision-making. Experimental results on the Kaggle and YTUIA datasets demonstrate that the proposed approach can effectively distinguish between ASD and non-ASD individuals, achieving high accuracy and precision. The incorporation of asymmetry features significantly improved model interpretability and performance, highlighting the effectiveness of integrating landmark-based features with deep learning techniques. The primary contribution of this work lies in introducing a robust method for quantifying facial asymmetry using MediaPipe and incorporating these interpretable features into a deep learning pipeline. While this study focused on specific blendshape regions such as the eyes, mouth, nose, and eyebrows, future work could explore alternative or more comprehensive sets of facial landmarks and other modalities to potentially further improve model performance and expand its applicability in ASD recognition.

Data Availability Statement: Kaggle dataset: https://github.com/mm909/Kaggle-Autism, https://minerva-clinic.or.jp/genetictesting/autismpanel/column/autism-face/#i

References

1. Ghosh, T.; Al Banna, M.H.; Rahman, M.S.; Kaiser, M.S.; Mahmud, M.; Hosen, A.S.; Cho, G.H. Artificial intelligence and internet of things in screening and management of autism spectrum disorder. *Sustainable Cities and Society* **2021**, *74*, 103189.
2. Miah, A.S.M.; Shin, J.; Hasan, M.A.M.; Molla, M.K.I.; Okuyama, Y.; Tomioka, Y. Movie oriented positive negative emotion classification from eeg signal using wavelet transformation and machine learning ap-

 doi:10.20944/preprints202512.2505.v1

proaches. In Proceedings of the 2022 IEEE 15th international symposium on embedded multicore/many-core systems-on-chip (MCSoC). IEEE, 2022, pp. 26–31.
3. Miah, A.S.M.; Shin, J.; Islam, M.M.; Molla, M.K.I.; et al. Natural human emotion recognition based on various mixed reality (MR) games and electroencephalography (EEG) signals. In Proceedings of the 2022 IEEE 5th Eurasian conference on educational innovation (ECEI). IEEE, 2022, pp. 408–411.
4. Khodatars, M.; Shoeibi, A.; Sadeghi, D.; Ghaasemi, N.; Jafari, M.; Moridian, P.; Khadem, A.; Alizadehsani, R.; Zare, A.; Kong, Y.; et al. Deep learning for neuroimaging-based diagnosis and rehabilitation of autism spectrum disorder: a review. *Computers in biology and medicine* **2021**, *139*, 104949.
5. Abdou, M.A. Literature review: Efficient deep neural networks techniques for medical image analysis. *Neural Computing and Applications* **2022**, *34*, 5791–5812.
6. Wang, M.; Xu, D.; Zhang, L.; Jiang, H. Application of multimodal MRI in the early diagnosis of autism spectrum disorders: a review. *Diagnostics* **2023**, *13*, 3027.
7. Esqueda-Elizondo, J.J.; Juárez-Ramírez, R.; López-Bonilla, O.R.; García-Guerrero, E.E.; Galindo-Aldana, G.M.; Jiménez-Beristáin, L.; Serrano-Trujillo, A.; Tlelo-Cuautle, E.; Inzunza-González, E. Attention measurement of an autism spectrum disorder user using EEG signals: A case study. *Mathematical and Computational Applications* **2022**, *27*, 21.
8. Alam, M.S.; Tasneem, Z.; Khan, S.A.; Rashid, M.M. Effect of Different Modalities of Facial Images on ASD Diagnosis Using Deep Learning-Based Neural Network. *J. Adv. Res. Appl. Sci. Eng. Technol* **2023**, *32*, 59–74.
9. Cîrneanu, A.L.; Popescu, D.; Iordache, D. New trends in emotion recognition using image analysis by neural networks, a systematic review. *Sensors* **2023**, *23*, 7092.
10. Jung, S.K.; Lim, H.K.; Lee, S.; Cho, Y.; Song, I.S. Deep active learning for automatic segmentation of maxillary sinus lesions using a convolutional neural network. *Diagnostics* **2021**, *11*, 688.
11. Ammari, A.; Mahmoudi, R.; Hmida, B.; Saouli, R.; Bedoui, M.H. Deep-active-learning approach towards accurate right ventricular segmentation using a two-level uncertainty estimation. *Computerized Medical Imaging and Graphics* **2023**, *104*, 102168.
12. Shin, J.; Miah, A.S.M.; Kakizaki, M.; Hassan, N.; Tomioka, Y. Autism Spectrum Disorder Detection Using Skeleton-Based Body Movement Analysis via Dual-Stream Deep Learning. *Electronics* **2025**, *14*, 2231.
13. Miah, A.S.M.; Hassan, N.; Hossain, M.M.A.; Okuyama, Y.; Shin, J. Multi Class Parkinsons Disease Detection Based on Finger Tapping Using Attention-Enhanced CNN BiLSTM. *arXiv preprint arXiv:2510.10121* **2025**.
14. Miah, A.S.M.; Suzuki, T.; Shin, J. A Methodological and Structural Review of Parkinson's Disease Detection Across Diverse Data Modalities. *IEEE Access* **2025**.
15. Shin, J.; Miah, A.S.M.; Hirooka, K.; Hasan, M.A.M.; Maniruzzaman, M. Parkinson disease detection based on in-air dynamics feature extraction and selection using machine learning. *Scientific Reports* **2025**, *15*, 28027.
16. Matsumoto, M.; Miah, A.S.M.; Asai, N.; Shin, J. Machine Learning-Based Differential Diagnosis of Parkinson's Disease Using Kinematic Feature Extraction and Selection. *IEEE Access* **2025**, *13*, 54090–54104. https://doi.org/10.1109/ACCESS.2025.3553528.
17. Hassan, N.; Miah, A.S.M.; Okuyama, Y.; Shin, J. Neurological Disorder Recognition via Comprehensive Feature Fusion by Integrating Deep Learning and Texture Analysis. *IEEE Open Journal of the Computer Society* **2025**.
18. Hassan, N.; Miah, A.S.M.; Suzuki, T.; Shin, J. Gradual Variation-Based Dual-Stream Deep Learning for Spatial Feature Enhancement With Dimensionality Reduction in Early Alzheimer's Disease Detection. *IEEE Access* **2025**, *13*, 31701–31717. https://doi.org/10.1109/ACCESS.2025.3542458.
19. Hassan, N.; Miah, A.S.M.; Okuyama, Y.; Shin, J. Enhanced Alzheimer's Disease Detection Using Deep Neural Networks with Spatial Feature Enhancement. In Proceedings of the 2024 IEEE 17th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC). IEEE, 2024, pp. 122–128.
20. Hassan, N.; Musa Miah, A.S.; Shin, J. Residual-Based Multi-Stage Deep Learning Framework for Computer-Aided Alzheimer's Disease Detection. *Journal of Imaging* **2024**, *10*, 141.
21. Miah, A.S.M.; Mamunur Rashid, M.; Redwanur Rahman, M.; Tofayel Hossain, M.; Shahidujjaman Sujon, M.; Nawal, N.; Hasan, M.; Shin, J. Alzheimer's disease detection using CNN based on effective dimensionality reduction approach. In Proceedings of the Intelligent Computing and Optimization: Proceedings of the 3rd International Conference on Intelligent Computing and Optimization 2020 (ICO 2020). Springer, 2021, pp. 801–811.
22. Hassan, N.; Miah, A.S.M.; Suzuki, K.; Okuyama, Y.; Shin, J. Stacked CNN-based multichannel attention networks for Alzheimer disease detection. *Scientific Reports* **2025**, *15*, 5815.

 doi:10.20944/preprints202512.2505.v1

23. Derbali, M.; Jarrah, M.; Randhawa, P. Autism spectrum disorder detection: Video games based facial expression diagnosis using deep learning. *International Journal of Advanced Computer Science and Applications* **2023**, *14*.
24. El Mouatasim, A.; Ikermane, M. Control learning rate for autism facial detection via deep transfer learning. *Signal, Image and Video Processing* **2023**, *17*, 3713–3720.
25. Shi, C.; Xin, X.; Zhang, J. Domain adaptation using a three-way decision improves the identification of autism patients from multisite fMRI data. *Brain Sciences* **2021**, *11*, 603.
26. Alam, M.S.; Elsheikh, E.A.; Suliman, F.; Rashid, M.M.; Faizabadi, A.R. Innovative strategies for early autism diagnosis: active learning and domain adaptation optimization. *Diagnostics* **2024**, *14*, 629.
27. Alam, M.S.; Rashid, M.M.; Roy, R.; Faizabadi, A.R.; Gupta, K.D.; Ahsan, M.M. Empirical study of autism spectrum disorder diagnosis using facial images by improved transfer learning approach. *Bioengineering* **2022**, *9*, 710.
28. Alam, M.S.; Rashid, M.M.; Jazlan, A.; Alahi, M.E.E.; Kchaou, M.; Alharthi, K.A.B. Robust Autism Spectrum Disorder Screening Based on Facial Images (For Disability Diagnosis): A Domain-Adaptive Deep Ensemble Approach. *Diagnostics* **2025**, *15*, 1601.
29. Alam, M.S.; Rashid, M.M.; Faizabadi, A.R.; Mohd Zaki, H.F.; Alam, T.E.; Ali, M.S.; Gupta, K.D.; Ahsan, M.M. Efficient deep learning-based data-centric approach for autism spectrum disorder diagnosis from facial images using explainable AI. *Technologies* **2023**, *11*, 115.
30. Ghazal, T.M.; Munir, S.; Abbas, S.; Athar, A.; Alrababah, H.; Khan, M.A. Early detection of autism in children using transfer learning. *Intelligent Automation & Soft Computing* **2023**, *36*, 11–22.
31. Kaur, N.; Gupta, G. Refurbished and improvised model using convolution network for autism disorder detection in facial images. *Indones. J. Electr. Eng. Comput. Sci* **2023**, *29*, 883–889.
32. Ikermane, M.; Mouatasim, A. Web-based autism screening using facial images and a convolutional neural network. *Indones. J. Electr. Eng. Comput. Sci* **2023**, *29*, 1140–1147.