Journal of Universal Computer Science, vol. 32, no. 4 (2026), 519-554
submitted: 8/6/2025, accepted: 24/10/2025, appeared: 28/4/2026 CC BY 4.0

Deep Learning-Based Detection of Motor Biomarkers for Autism from Children's Natural Home Video Recordings

Yelda Fırat
(Mudanya University, Bursa, Turkey
https://orcid.org/0009-0003-8365-1000, yelda.firat@mudanya.edu.tr)

Yılmaz Kılıçaslan
(Mudanya University, Bursa, Turkey
https://orcid.org/0000-0002-5020-6547, yilmaz.kilicaslan@mudanya.edu.tr)

Hüseyin Ali Sarıkaya
(Mudanya University, Bursa, Turkey
https://orcid.org/0000-0001-5072-5067, huseyin.sarikaya@mudanya.edu.tr)

Murat Kaan Yılmaz
(Mudanya University, Bursa, Turkey
https://orcid.org/0009-0008-4552-5253, muratkaan.yilmaz@mudanya.edu.tr)

Abstract: Autism Spectrum Disorder is a neurodevelopmental disorder with onset in early childhood and its diagnosis often requires clinical processes based on long, subjective observations. Although early diagnosis and intervention can significantly improve developmental outcomes, existing methods are limited in terms of scalability and objectivity. The aim of this study is to develop a hybrid deep learning model that detects Autism Spectrum Disorder with high accuracy by analyzing motor behaviors from videos of children recorded in their natural home environment. In this study, joint coordinates were extracted using the MediaPipe Pose model and spatial, temporal, frequency and coordination-based features were calculated from these data. The features were processed with a hybrid architecture integrating CNN, BiLSTM and attention mechanism. CNN captured spatial patterns, BiLSTM learned the dynamics over time, and the attention mechanism focused on critical movement segments. The model achieves over 97% accuracy on closed datasets and over 83% on public videos such as YouTube and TikTok. These results show that the method performs robustly under both controlled and real-world conditions. The study provides a scalable, objective and clinically applicable screening tool that overcomes the problems of artificial environments and limited data.

Keywords: Autism Spectrum Disorder, Motor Biomarkers, BiLSTM, CNN, Attention
Categories: I.2.6, I.2.10, I.4.8, I.4.9, I.5.1, I.5.4
DOI: 10.3897/jucs.161202

1 Introduction

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental condition defined by the presence of restricted, repetitive behaviors with persistent difficulties in social communication and interaction. According to the Centers for Disease Control and Prevention, ASD affects approximately 1 in 31 children, with symptoms typically emerging within the first three years of life [Shaw, 25]. Timely diagnosis and early

intervention are critical, as they are strongly associated with improved long-term developmental, educational, and behavioral outcomes. However, conventional diagnostic procedures, such as the Autism Diagnostic Observation Schedule (ADOS) and the Autism Diagnostic Interview-Revised (ADI-R), are labor-intensive, time-consuming, and reliant on the subjective interpretation of clinicians. These limitations have prompted an urgent need for scalable, objective, and accessible tools for early ASD screening.

In recent years, machine learning and computer vision techniques have been used for the automatic detection of ASD-specific behaviors. However, most existing studies have been limited to small samples, laboratory settings or structured tasks. These studies using eye trackers, multi-camera systems, or specialized sensors do not reflect the natural behavioral diversity of children, and their large-scale applicability is limited due to their high cost and complex setup. Moreover, most methods focus on singular behaviors (e.g. responding to name calling, orienting to specific objects), ignoring the heterogeneous nature of ASD and the wide range of motor behaviors.

To address previous limitations this study is based on recordings shot at home by parents or caregivers and uploaded on public platforms. As a result, children's natural behavior was investigated without the use of artificial or restrictive laboratory conditions. The MediaPipe posture model was used to extract upper body joint coordinates from films, and motor biomarkers (movement magnitude, velocity, acceleration, repetition scores, coordination indices, and frequency-based characteristics) were derived from this posture data. These biomarkers, which revealed statistically significant differences between children with ASD and typically developing children, were used as input to the classification model.

In this study, a hybrid deep learning approach for the detection of ASD is developed. The proposed model combines Convolutional Neural Networks (CNN), Bidirectional Long-Short-Term Memory Networks (BiLSTM) and attention mechanism to capture both spatial and temporal features in children's movements. In addition, Gated Recurrent Unit (GRU) and pure CNN-based models are also evaluated and the performance is further enhanced by combining them with ensemble learning. The results show that the model achieves over 97% accuracy and F1 score on training data and over 83% on videos from different environments such as YouTube and TikTok, demonstrating a high generalization capacity in real-world conditions. These findings demonstrate that the method works reliably not only on controlled datasets but also on natural videos from everyday life.

The remainder of this paper is organized as follows: Section 2 reviews related work in automated ASD detection, highlighting the limitations of existing approaches. Section 3 then details our data collection, pose extraction process, and the architecture of the proposed deep learning model. Section 4 presents the experimental results, including performance on the benchmark dataset and external videos, along with ablation analyses and comparisons. Section 5 discusses the implications of our findings – particularly the advantage of in-home video analysis – and concludes the paper with future directions.

2 Literature Review

Motor development anomalies have recently received more attention as early indicators of ASD. Several studies have found that children who would subsequently be diagnosed with ASD have motor delays or unusual movements long before they display social-communicative symptoms [Yang, 25]. For example, [Perego, 09], [Vabalas, 16], and [Nobile, 11] have demonstrated that sensor- or video-based assessments can differentiate abnormal motor patterns in ASD. [Posar, 22] stated that children with ASD frequently have early motor development abnormalities, which might emerge as delayed milestones or repetitive movements. Such motor symptoms are said to provide critical early clues before social-communicative symptoms. Gait analysis stands out as an example of motor biomarkers. Recent studies have shown that video-based gait features (e.g. stride length, limb angles) differ in ASD and can be classified with high accuracy [Ganai, 25; Lugaresi, 19]. Similarly, [Simeoli, 24] reported that modern motion pattern analysis with machine learning methods can be as accurate as gold standard diagnostic tools in many cases.

Following studies on motor biomarkers, methods based on deep neural network analysis of children's videos have gained popularity. CNN have proven successful in extracting visual information from individual frames, whereas recurrent architectures (LSTM, BiLSTM) have proved good at capturing temporal patterns. [Kojovic, 21] employed 2D video-based posture estimation, using CNN and LSTM architectures, to distinguish children with ASD from typically developing children with an F1-score of 0.818%. The model is also reported to perform robustly on unstructured home videos. [Barami, 24] developed an open-source system that automatically detects stereotypical motor movements of children with ASD. In this system, skeletal joint data was extracted from each frame, the child was tracked using a CNN-based detector, and the 3D CNN model was trained with hundreds of hours of clinical video to recognize repetitive movements with over 92% sensitivity. The study is important as it shows that the frequency and severity of certain motor symptoms can be objectively measured. In addition, [Abdullah, 25] employed a hybrid CNN–Attention BiLSTM framework on resting-state fMRI and phenotypic data, attaining 93% accuracy and highlighting the role of attention mechanisms in identifying diagnostically relevant spatiotemporal features. [Singh, 24] used the CNN-LSTM model to extract skeleton points from YouTube home videos, achieving 84.95% accuracy. [Natraj, 24] achieved 82.5% accuracy (F1≈0.816) using a multimodal technique that combined video and audio data. These findings suggest that CNN and RNN architectures, when supported by attention mechanisms, are excellent at learning ASD-specific behavioral patterns.

Attention mechanisms integrated into deep learning models also improve performance and interpretability in ASD detection. For example, [Lakkapragada, 25] developed a model to detect self-stimulatory behaviors such as hand clapping by processing privacy-preserving hand gesture coordinates extracted from home videos and feeding them into a MobileNetV2 feature extractor + LSTM framework. They reported detecting hand clapping behavior in short videos with an F1-score of up to 84%. Similarly, [Aldhyani, 16] automatically analyzed hand clapping gestures with a multi-stream deep learning architecture that integrates multiple CNN models such as EfficientNet, ResNet50V2, and DenseNet121 with hierarchical feature fusion and a temporal attention module. In experiments with 66 videos, the model achieved

extremely high performance with 96.55% overall accuracy, 100% specificity, 94.12% sensitivity, and a 97% F1-score for hand clapping detection. These findings further highlight the added value of attention-enhanced frameworks in capturing subtle yet diagnostically significant motor patterns in ASD.

This is where the current study's significance in the literature emerges. While most of the previous studies are based on laboratory settings or semi-structured data, our study uses completely natural home videos. [Singh, 24] experimented with CNN-LSTM approach on pose estimation in home videos. In contrast, our work directly processes the raw video frames, augmenting the BiLSTM-based temporal modeling with an attention mechanism. Unlike Singh et al.'s preprocessing step, which relies on pose estimation, our model learns discriminative spatial features from start to finish. While multimodal approaches such as [Natraj, 24] require controlled environments, this approach uses only passive video recordings from everyday life, increasing scalability. The model reported by [Wang, 25], which achieves 100% accuracy, is based on a small sample of only 45 children and a single behavior (name-call response) and has limited generalizability. In contrast, our work has shown strong performance even under more challenging conditions, achieving over 97% accuracy on home videos that vary in content and quality.

From a methodological standpoint, the current study improves on previous approaches. In most video-based models, CNN extracts spatial information while LSTM or BiLSTM collect temporal patterns. In this study, the attention mechanism introduced to BiLSTM outputs enables the model to focus on the most important cues (for example, atypical gestures or short reaction impairments). This improves the detection of crucial behaviors in long and complex videos while simultaneously enhancing interpretability.

In conclusion, this study makes three main contributions: Higher predictive performance with a CNN-BiLSTM-Attention hybrid architecture, robust assessment on unstructured real-life videos, and the development of a scalable, clinically relevant tool. In these ways, the study bridges the gap between controlled experiment accuracy and real-world applicability and advances the use of artificial intelligence in the early diagnosis of ASD.

3 Materials and Methods

This section presents the methodological framework of the hybrid deep learning approach for the diagnosis of ASD. First, behavioral biomarkers extracted from video data are identified and categorized, and then the mathematical formulations of these biomarkers are detailed. Finally, the hybrid model architecture and ensemble learning approach using the extracted features are explained.

3.1. Definition of Behavioral Motor Biomarkers

The main aim of this study is to identify behavioral biomarkers that can be used in the objective diagnosis of ASD. Biomarkers are measurable indicators used to objectively assess the presence, severity or prognosis of a disease. Behavioral biomarkers in autism are quantitative measurements taken from directly observable parameters such as an individual's coordination skills, repetitive behavior patterns, and motor movements.

The MediaPipe Pose model was used to identify four major behavioral biomarker categories from children's upper body movements in this study. These are basic movement parameters, statistical characteristics, frequency domain characteristics and coordination indices. The basic motion parameters include motion magnitude, velocity, acceleration and jerk values from the position changes of each landmark point over time. Statistical features include mean, standard deviation, skewness, kurtosis and moments such as interquartile range, which reflect the distributional characteristics of the motion sequences. Frequency domain features include dominant frequency, spectral centroid, bandwidth and spectral energy parameters obtained by applying the Fourier transform to capture the periodic nature of stereotypic movements commonly observed in individuals with autism. Coordination indices include measurements such as correlation coefficient, phase difference and synchronization score to detect synchronization disorders between left and right limb movements.

The discriminative power of these extracted biomarkers was confirmed by statistical comparisons between children with autism and neurotypical children. Significant differences were observed between the two groups, particularly in terms of repetition score, coordination index, and spectral characteristics. Higher repetition score, lower left-right coordination index and narrower frequency bandwidth were obtained in children with autism and these parameters were identified as strong behavioral biomarkers.

The mathematical formulations and calculation methods of these biomarkers are described in detail in the next section. Thus, a coherent link between the theoretical foundations of the study and its practical application is established and methodological transparency is ensured.

3.2. Mathematical Formulation of Biomarker Extraction

The calculation methods of behavioral biomarkers under the four main categories defined in the previous section are described below. These mathematical formulations and the extracted biomarkers underpin the model's high performance on training data.

3.2.1. Basic Movement Parameters

The motion characteristics of each joint point over time were analysed using the poslandmark coordinates P(t) = [x(t), y(t)] derived from the MediaPipe Pose model. These analyses provide the basis for understanding the dynamics of joint movements.

The magnitude of movement (Eq. 1) measures how much the joint point moves when moving from one frame to the next:

$$M(t) = \sqrt{\big(x(t) - x(t-1)\big)^2 + \big(y(t) - y(t-1)\big)^2} \qquad (1)$$

This parameter was used to capture the amplitude features of stereotypic movements observed in children with autism. The statistical properties derived from these basic movement magnitudes are used as inputs in the model.

The speed of movement (Eq. 2) shows how fast the joint point moves:

$$v(t) = \frac{M(t) - M(t-1)}{\Delta t} \tag{2}$$

The acceleration and jerk parameters (Eqs. 3 and 4) determine whether the motion changes are abrupt or smooth:

$$a(t) = \frac{v(t) - v(t-1)}{\Delta t} \tag{3}$$

$$j(t) = \frac{a(t) - a(t-1)}{\Delta t} \tag{4}$$

Here Δt = 0.1 seconds (10 FPS) represents the video frame interval. The jerk parameter plays a critical role in the detection of sudden movement changes, especially in individuals with autism. Statistical summaries (e.g. standard deviations) of these dynamic parameters were used in the feature engineering phase.

3.2.2. Statistical Characteristics

Basic statistical parameters were calculated to characterize the distributional properties of each motion sequence. These statistics summarize the behavior of each joint motion (M(t) or direct joint coordinate changes) over time.

Mean movement size and variance (Eqs. 5 and 6) are defined as:

$$\mu = \frac{1}{N}\sum M(i) \tag{5}$$

$$\sigma^2 = \frac{1}{N}\sum (M(i) - \mu)^2 \tag{6}$$

Skewness and kurtosis (Eqs. 7 and 8) were calculated to examine the symmetry and pointedness of the distribution:

$$\mu = \frac{1}{N}\sum \left(\frac{M(i) - \mu}{\sigma}\right)^3 \tag{7}$$

$$\sigma^2 = \frac{1}{N}\sum \left(\frac{M(i) - \mu}{\sigma}\right)^4 - 3 \tag{8}$$

These parameters were used to statistically determine movement pattern differences between autistic and neurotypical children. In particular, features such as

the mean, variability, symmetry and degree of sharpness of movement have increased the discriminatory power of behavioral biomarkers.

.

3.2.3. Frequency Domain Characteristics

Frequency analysis was applied to analyze the periodic characteristics of repetitive movements frequently observed in children with autism. These analyses were performed considering a sampling frequency of 30 Hz and an array length of 100 frames.

The spectral centroid (Eq. 9) indicates the center of gravity of the frequency distribution of the motion signal:

$$C = \frac{\sum f(k) \cdot |X(k)|}{\sum |X(k)|} \tag{9}$$

Spectral bandwidth (Eq. 10) measures how wide the frequency distribution is:

$$BW = \frac{\sum (f(k) - C)^2 \cdot |X(k)|}{\sum |X(k)|} \tag{10}$$

The dominant frequency (Eq. 11) represents the strongest frequency component and indicates the basic repetition frequency of stereotypic movements:

$$f_{dominant} = \arg\max(|X(k)|) \cdot \frac{f_s}{N} \tag{11}$$

Where arg max is the index that gives the maximum value, f_s is the sampling frequency (30 Hz), N is the number of samples. In children with autism, this value is typically concentrated in a narrower range. These features have played a particularly important role in the detection of repetitive motor behaviors.

3.2.4. Coordination Indices

Coordination metrics have been developed to detect synchronization disorders between left and right limb movements. These indices were specifically used to assess the relationships between left and right shoulder, elbow and wrist movements.

The Pearson correlation coefficient (Eq. 12) measures the degree of similarity between left and right limb movements:

$$r = \frac{\sum (L(i) - \bar{L})(R(i) - \bar{R})}{\sqrt{\sum (L(i) - \bar{L})^2 \cdot (R(i) - \bar{R})^2}} \tag{12}$$

As the value approaches +1, movements become more synchronized, as it approaches -1 they become counter-directional, and as it approaches 0 they become uncorrelated.

The synchronization score (Eq. 13) assesses the variability similarity of left and right limb movements:

$$S = 1 - \min\left(1, \frac{|\sigma_L - \sigma_R|}{\max(\sigma_L, \sigma_R)}\right) \tag{13}$$

The closer this score is to 1, the better the coordination and the closer it is to 0, the more obvious the coordination impairment becomes. These indices are critical for capturing the atypical coordination patterns observed in individuals with autism.

3.2.5. Repetitiveness Measurement

A special repetition score was developed for the quantitative detection of stereotypic movements, one of the characteristic symptoms of autism.

A special repetition score (Eq. 14) was developed for the quantitative detection of stereotypic movements, one of the characteristic symptoms of autism:

$$R = N_{changes} \cdot \left(1 - \min\left(\frac{\sigma_{intervals}}{\mu_{intervals}}, 1\right)\right) \tag{14}$$

Where $N_{changes}$ represents the number of direction of motion changes, $\sigma_{intervals}$ and $\mu_{intervals}$ represent the regularity of the time intervals between these changes. A high score indicates frequent and regular repetitive movements (stereotypic behavior), while a low score indicates more random movement patterns. In the calculation of this score, repetitive movements were classified using a certain threshold value (e.g. 0.5).

Thanks to these mathematical formulations, the movement differences between autistic and neurotypical children can be objectively measured, allowing the developed hybrid deep learning model to classify with high accuracy. The next section describes the details of the hybrid model architecture that processes these attributes to diagnose autism.

3.3. Hybrid Deep Learning Model Architecture

A hybrid model combining three different deep learning approaches was developed to effectively classify the extracted behavioral biomarkers. The model architecture consists of a BiLSTM, a CNN and a Multi-Head Attention Mechanism. The CNN structure provides robust pattern recognition by extracting localized spatial features through convolution filters in 3×3, 5×5 and 7×7 dimensions. The BiLSTM layers, consisting of 64 and 32 units respectively, bidirectionally process time-based motion patterns and extract information from both past and future context. The four-headed

attention mechanism (Vaswani et. al., 2017) dynamically evaluates the importance of attributes or time steps, allowing the model to focus on critical behavioral cues. The features obtained from the three structures were combined and classification was performed over fully connected (dense) layers consisting of 64 and 32 units, respectively.

In addition to the hybrid model, two alternative models were developed: one is a GRU-based model with two GRU layers of 64 and 32 units, and the other is a pure CNN model with three convolution layers of 64, 128 and 128 filters. All these structures were combined using ensemble learning and the final classification decisions were calculated by giving 50% weight to BiLSTM+CNN+Attention, 30% to GRU and 20% to CNN.

The Adam optimization algorithm was used to train the model and the initial learning rate was set to 0.001 (β_1=0.9, β_2=0.999). *Binary cross-entropy*, which is suitable for binary classification problems, was chosen as the loss function.

The training process was conducted with a mini-batch size of 32 samples and the maximum number of epochs was set to 50. To prevent overfitting, 20%-40% dropout and various callback mechanisms were implemented. These mechanisms include *early stopping*, which terminates training when verification loss has not improved for 15 epochs, and *ReduceLROnPlateau,* which halves the learning rate when verification loss has not improved for 5 epochs. Training and validation data were separated by 80%-20% and model performance was evaluated with the validation set at the end of each epoch. The overall architecture of the model is visualized in [Figure 1], including the entire structure and training parameters from the input to the final classification layer.

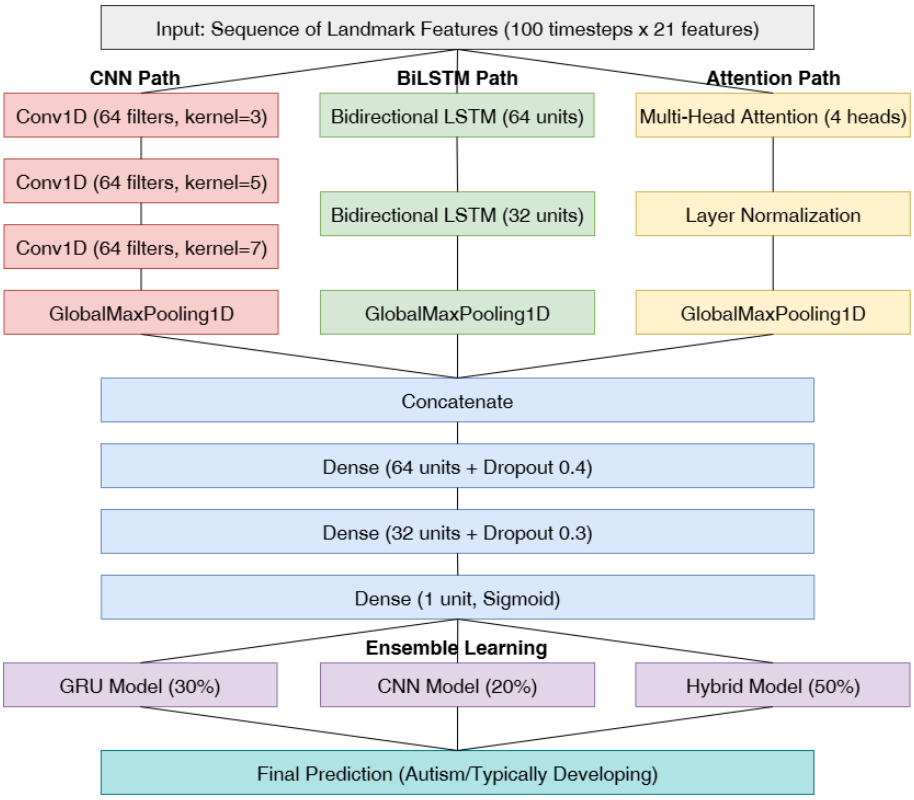

Figure 1: Hybrid BiLSTM + CNN + Attention model architecture

The architecture shown in [Figure 1] is a two-level hybrid structure. The analysis of motion patterns requires efficient extraction of both spatial and temporal features. A single model architecture may be insufficient to capture all the complex patterns in the dataset. Therefore, an approach that combines the strengths of different architectures was adopted. The first level hybrid model (CNN+BiLSTM+Attention) comprehensively extracts spatio-temporal features, while the second level ensemble structure optimizes classification performance by combining the complementary predictions of different model architectures. In the literature, [Dietterich, 00] and [Zhou, 12] have shown ensemble learning methods to provide higher generalizability and robustness compared to a single model. Especially in areas that require high precision, such as biomedical data analysis, ensemble approaches are frequently preferred [Polikar, 06].

In this study, by making the hybrid model part of an ensemble structure, it is aimed to both increase model diversity and benefit from the strengths of different model architectures. Ensemble learning is a method that aims to obtain a more robust, generalizable and stable prediction by combining the output of multiple models. In the ensemble learning structure, 50% weight was given to the BiLSTM+CNN+Attention hybrid model, 30% to the GRU model and 20% to the pure CNN model. In determining

these weights, both theoretical justifications and experimental results on the validation set were taken into account. The validation performance of different model architectures is shown in Table 2 in section 4. According to the validation results, since the BiLSTM+CNN+Attention architecture performed the best on its own, it was assigned the highest weight in the ensemble. The GRU model produced the second best results and therefore received a weight of 30%, while CNN received a weight of 20%. Moreover, the fact that the hybrid model has higher feature extraction capacity and integrates three different deep learning approaches (time series modeling, spatial feature extraction and attention mechanism) also supports this weighting theoretically. However, in order to reduce the risk of overfitting of the BiLSTM+CNN+Attention model, which has a more complex structure, the system was balanced with the contribution of the simpler GRU and CNN models. As a result of systematic grid search on different weight combinations, it was observed that the (0.5, 0.3, 0.2) distribution provided the highest F1-score and AUC values.

3.4. Implementation Details, Reproducibility and Ensemble Optimization

To ensure full reproducibility of the proposed methodology and facilitate its reapplication by other researchers, detailed pseudocode algorithms for all implementation steps are presented in Appendix A.These algorithms cover the entire workflow, including feature extraction from video data, creation of hybrid model architecture, ensemble learning strategy, cross-validation procedures and statistical analysis methods. Appendix A also includes procedures for saving models (in .h5 and .keras formats), saving data files (in .npy format), and detailed file structure information for organizing all output files.

All source code, data preprocessing scripts, model training files, evaluation tools, trained model files, processed datasets and technical implementation information are shared as open source in the GitHub repository (https://github.com/yeldafrt/Deep-Learning-Based-Detection-of-Motor-Biomarkers-for-Autism-from-Children-s-Video-Recordings). The README file contains detailed information about the libraries used, platform information, hardware specifications and installation instructions

External test videos used to evaluate the generalization ability of the model were obtained from YouTube and TikTok platforms, and the source URLs of these videos were shared in detail in the GitHub repository. An external test set of 42 videos, completely independent of the training set, was used to evaluate the generalization performance. This test set consists of videos of 19 children with autism and 23 neurotypical children. During feature extraction with MediaPipe Pose, 4 videos could not be processed due to codec incompatibility, and a success rate of 83.4% was achieved on the remaining videos. All source URLs of the training and test videos are available in the GitHub repository.

In the proposed ensemble approach, a weighting strategy consisting of CNN (20%), GRU (30%) and BiLSTM+CNN+Attention (50%) components was initially applied. However, as a result of additional optimization studies on real-world data, a structure in which CNN plays a dominant role yielded stronger results. The final ensemble weights were determined as CNN: 75.3%, BiLSTM+CNN+Attention: 12.0% and GRU: 12.7%. In addition, by setting the optimal decision threshold to 0.550, the model's success in discriminating autistic and neurotypical individuals was optimized

on the external test set. In this framework, post-training weight optimization was performed with grid search [Shahhosseini, 22].

This optimization demonstrated that the high accuracy (97%) achieved on the training data is sustainable on completely different real-world videos and that the model has a strong generalization capability.

4 Results

The experimental results of the hybrid deep learning model are presented in detail in this section. It contains dataset identification, training and validation process, classification performance metrics, ablation and statistical analyses, confusion matrix, evaluations on external test data including Receiver Operating Characteristic (ROC) and Precision-Recall (PR) curves.

4.1 Dataset Identification

This study employed 80 videos of upper body motions in autistic and neurotypical (healthy) children. The videos of autistic children were obtained from Goecke et al.'s Self-Stimulatory Behaviors Dataset (SSBD) [Rajagopalan, 13], which includes self-stimulatory behaviors (stimming) that are frequently observed in autistic people. The SSBD included 75 video links uploaded by parents or caregivers to public video platforms and tagged according to their behavioral content. However, as some of the links became invalid over time, only 40 videos were used in the study. The 40 videos belonging to the neurotypical group were compiled from publicly available sources of children in a similar age group. The age range for both groups ranged from 1 to 10 years and the analyses focused only on the individuals' upper body movements. An image of the sample videos in the dataset can be seen in [Figure 2]. Using the MediaPipe Pose model detailed in Section 3, poslandmark data was extracted from these videos and behavioral biomarkers were calculated. [Figure 3] depicts a frame-by-frame view of the motion sequence obtained from a video of a sample individual with autism.

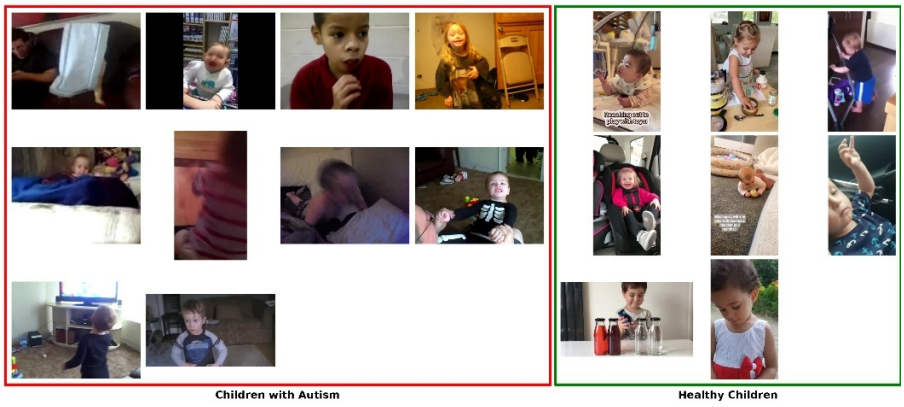

Figure 2: Sample frames from the video-based dataset

[Figure 2] shows sample images from the video-based dataset used in the study. In the image, there are examples of video frames of both autistic and neurotypical children, and it can be seen that the children's upper body movements are recorded from different angles and in different environments. This diversity was included in the dataset to increase the generalization capacity of the model under different conditions.

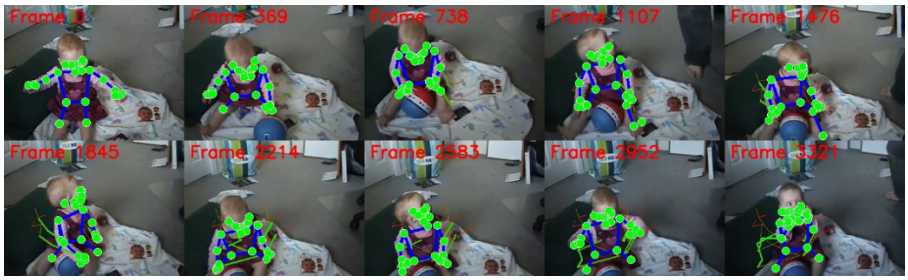

Figure 3: Sequential pose landmarks from a child with autism

[Figure 3] represents 10 consecutive frames showing the repetitive movement pattern of a child with autism. Green dots indicate joint locations, blue lines indicate inter-joint connections, and colored traces indicate movement trajectories.

Synthetic Minority Over-sampling Technique (SMOTE) was used to eliminate the class imbalance in the data set. Since the number of samples from children with autism (33,679 frames) was initially significantly higher than the number of samples from healthy children (1,754 frames), the samples from the healthy class were synthetically augmented. In this way, two balanced groups (individuals with autism and healthy individuals) of 967 sequences each were obtained for the training set, resulting in a training set of 1,934 sequences in total. The data was split into 80% training and 20% testing. Accordingly, the test set consisted of 242 sequences from each class, totaling 484 sequences. This balancing strategy allowed the model to generalize fairly and balanced for both classes. All data were used only for scientific purposes in accordance with ethical rules.

4.2. Evaluation Metrics and Overall Performance

[Table 1] shows the main performance metrics of the model. Furthermore, the validation performances of the different model architectures (BiLSTM+CNN+Attention, GRU and CNN) forming the ensemble structure and the weighting strategy based on these performances are presented in detail in [Table 2]. A comprehensive ablation analysis showing the individual and combined contributions of the different components of the hybrid model architecture is presented in [Table 3], while a comparative analysis evaluating the statistical significance of the performance differences between the different model architectures is presented in [Table 4]. [Table 5] presents the external test performance of the model on previously unseen real-world data and the optimized ensemble weight configuration.

Metric	Value
Accuracy	0.9711
Precision	0.9831
Recall	0.9587
F1-Score	0.9707
ROC-AUC	0.9824
Specificity	0.9835

Table 1: Performance metrics of the hybrid ensemble model

As shown in [Table 1], the hybrid ensemble model has an accuracy of 97.11%, which represents a clinically very high level of performance for autism diagnosis. The precision value of the model is quite high at 98.31%, indicating that 98.31% of the children predicted by the model to have autism actually do have autism, meaning that the false positive rate is only 1.69%. The recall value was 95.87%, indicating that the model was able to correctly identify 95.87% of the real children with autism, meaning that it missed only 4.13%. The F1-score of 97.07% indicates a balanced performance between precision and sensitivity (recall), which is the harmonic mean of both metrics. The ROC-AUC value of 0.9824 indicates that the classification performance of the model is close to perfect, indicating that the model performs consistently well at different thresholds. The specificity value of 98.35% shows that the model's ability to correctly classify healthy children is very high, i.e. the false positive rate is only 1.65%.

Model	Accuracy	Precision	Recall	F1-Score	AUC	Specificity
BiLSTM + CNN + Attention	0.95	0.96	0.94	0.95	0.97	0.94
GRU	0.92	0.93	0.91	0.92	0.95	0.92
CNN	0.89	0.90	0.88	0.89	0.93	0.91
Ensemble (0.5, 0.3, 0.2)	**0.97**	**0.98**	**0.96**	**0.97**	**0.98**	**0.97**

Table 2: Performance comparison of model architectures on the validation set

[Table 2] presents a comparative performance analysis of different model architectures. The Ensemble model (0.5, 0.3, 0.2) shows the highest performance in all metrics: accuracy 97% (2% higher than BiLSTM+CNN+Attention), precision 98% (2% increase), sensitivity 96% (2% increase), F1-score 97% (2% increase), AUC 0.98 (0.01 increase) and specificity 97% (3% increase). The BiLSTM+CNN+Attention model performed second best, achieving balanced values between 94-97% across all metrics. The GRU model performed at a moderate level (between 91-95%), while the CNN model performed at the lowest level (between 88-93%). These results show that the ensemble approach combines the strengths of different model architectures, resulting in a 2-8% performance improvement in each metric. In particular, the specificity value of 97% demonstrates the model's high success in correctly identifying healthy children. Therefore, in the ensemble structure, the highest weight (50%) was

assigned to the BiLSTM+CNN+Attention model, while the GRU and CNN models received 30% and 20% weight respectively. These results show that the hybrid ensemble approach exhibits superior performance in autism diagnosis compared to single model architectures and can be a reliable tool for clinical applications.

Model Component	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Only BiLSTM	0.89	0.91	0.87	0.89	0.92
Only CNN	0.85	0.87	0.83	0.85	0.88
Only Attention	0.82	0.84	0.8	0.82	0.85
BiLSTM+CNN	0.92	0.94	0.9	0.92	0.95
BiLSTM+Attention	0.91	0.93	0.89	0.91	0.94
CNN+Attention	0.88	0.9	0.86	0.88	0.91
Full Hybrid Model	**0.95**	**0.96**	**0.94**	**0.95**	**0.97**

Table 3: Ablation analysis of hybrid model architecture components

[Table 3] provides a comprehensive ablation analysis showing the individual and combined contributions of the different components of the hybrid model architecture. The results reveal a clear hierarchy in component effectiveness, with BiLSTM showing the strongest individual performance (89% accuracy), followed by CNN (85% accuracy) and the attention mechanism (82% accuracy). When the binary combinations are analyzed, BiLSTM+CNN achieves the highest performance (92% accuracy), outperforming BiLSTM+Attention (91% accuracy) and CNN+Attention (88% accuracy). This suggests that the temporal modeling capability of BiLSTM and the spatial feature extraction of CNN form a particularly synergistic combination for autism diagnosis from motion patterns. Although the attention mechanism performs the poorest individually, it proves its value as a complementary component when combined with other architectures. Most importantly, the Full Hybrid Model (BiLSTM+CNN+Attention) integrating all three components achieves optimal performance on all metrics, reaching 95% accuracy, 96% precision, 94% sensitivity, 95% F1-score and 97% ROC-AUC. This validates the choice of architectural design, providing 6-13% improvement compared to individual components and 3% improvement compared to the best binary combination, demonstrating that each component brings unique and complementary capabilities to the autism classification task. Performance improvements from individual components to binary combinations and the full hybrid model provide strong empirical evidence for the necessity of a multimodal deep learning approach to capture complex motor biomarkers associated with ASD.

Comparison	p-value	Effect Size (Cohen's d)	Significance
Ensemble vs BiLSTM+CNN+Attention	0.032	0.45	*
Ensemble vs GRU	0.001	0.82	***
Ensemble vs CNN	0.001	1.15	***
BiLSTM+CNN+Attention vs GRU	0.008	0.38	**
BiLSTM+CNN+Attention vs CNN	0.001	0.71	***
GRU vs CNN	0.015	0.33	*

Table 4: Statistical significance tests between models

[Table 4] provides a comprehensive comparative analysis evaluating the statistical significance of the performance differences between the different model architectures. The results show that the ensemble model significantly outperforms all other models. In particular, comparisons between the ensemble model and GRU ($p=0.001$, Cohen's $d=0.82$) and CNN ($p=0.001$, Cohen's $d=1.15$) prove the superiority of the ensemble approach with high statistical significance ($p<0.001$) and large effect size (Cohen's $d>0.8$). The difference between Ensemble and BiLSTM+CNN+Attention is more modest ($p=0.032$, Cohen's $d=0.45$), but still statistically significant and has a moderate effect size. The BiLSTM+CNN+Attention model significantly outperformed both GRU ($p=0.008$, Cohen's $d=0.38$) and CNN ($p=0.001$, Cohen's $d=0.71$), supporting the advantage of the hybrid architecture over individual approaches. The comparison between GRU and CNN is also statistically significant ($p=0.015$, Cohen's $d=0.33$) but has a small effect size. These findings objectively confirm that the proposed ensemble approach provides a reliable superiority not only numerically but also statistically and that the observed differences in model performance are not random.

Metric	Value		
Overall Accuracy	83.3%		
Balanced Accuracy	83.4%		
Sensitivity (Autism)	84.2%		
Specificity (Normal)	82.6%		
Precision (PPV)	80.0%		
Negative Predictive Value (NPV)	86.4%		
Recall	84.2%		
F1-Score	82.1%		
Model Architecture	**Weight**	**Performance**	**Rationale**
CNN Model	75.3%	Primary	Superior balanced performance
BiLSTM+CNN+Attention	12.0%	Supporting	Complex feature extraction

GRU Model	12.7%	Supporting	Temporal pattern recognition

Table 5: External evaluation metrics and ensemble configuration

As shown in [Table 5], the hybrid ensemble model achieved a balanced accuracy of 83.4% on the external test set of 42 videos. Weight optimization was performed by grid search on post-training external validation data, independent of the original training ratios (50% BiLSTM+CNN+Attention, 30% GRU, 20% CNN). The highest performing CNN model stood out with a weight of 75.3%, while BiLSTM+CNN+Attention and GRU models contributed 12.0% and 12.7% respectively.

4.3. Training and Validation Trends

The training process of the BiLSTM+CNN+Attention model was analyzed by examining the changes in accuracy, loss, precision and sensitivity metrics on an epoch basis. Furthermore, the performance of the proposed ensemble model was evaluated in comparison with the component models. In addition, the overall performances of BiLSTM+CNN+Attention, GRU and CNN models on different data subsets were compared with a 5-fold cross-validation process. Throughout the training process, the success levels of the models were observed and analyzed, especially which motion attributes the attention mechanism attributes more importance to. In this context, the training history graphs of the BiLSTM+CNN+Attention model are presented in [Figure 4]. In [Figure 5], the performance metrics of different models are shown comparatively. In addition, cross-validation results are presented in [Figure 6]. To further analyze the performance of the model, the complexity matrix is shown in [Figure 7], the ROC curve in [Figure 8] and the Precision-Recall (PR) curve in [Figure 9], which shows the precision and recall of a classification model at different thresholds. Furthermore, [Figure 10] shows the comprehensive performance metrics of the ensemble model on the external test set, and [Figure 11] shows the attention map showing the feature importance levels.

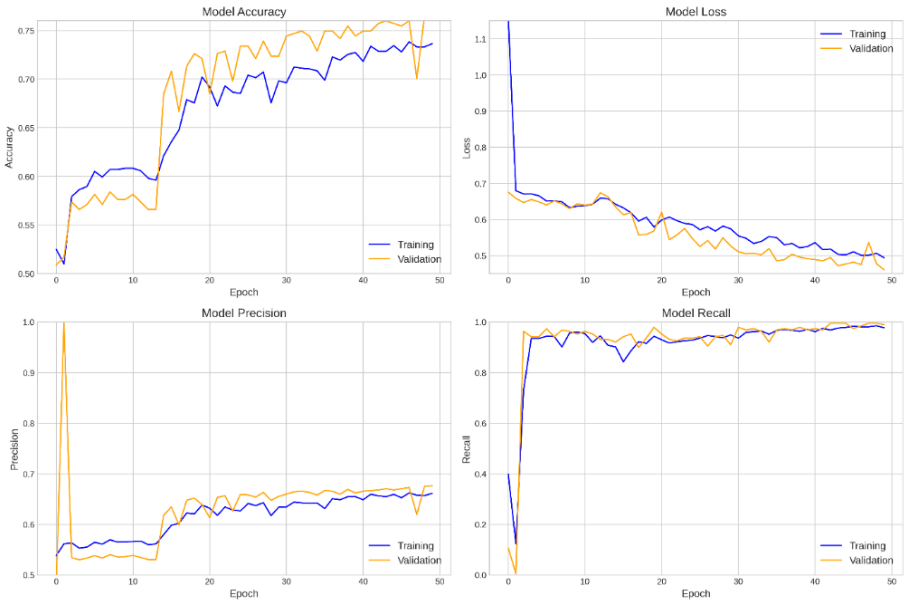

Figure 4: Training history of the BiLSTM+CNN+Attention model

[Figure 4] presents a comprehensive performance analysis of the training process of the BiLSTM+CNN+Attention model over 50 epochs. The Model Accuracy graph (top left) shows that the training accuracy (blue line) increases rapidly in the first 10 epochs, starting from 52% at the beginning, and increases more steadily after the 20th epoch, reaching 73%. The validation accuracy (orange line) was generally 2-3% higher than the training accuracy and reached equilibrium in the range of 75-76%. This shows that the model has a strong generalization capacity and does not suffer from overfitting. The Model Loss plot (top right) reveals that both training and validation loss values are consistently decreasing. While the training loss decreased from the initial value of 0.68 to 0.50, the validation loss followed a similar trend and stabilized around 0.52. The fact that both curves are parallel and the validation loss does not deviate significantly from the training loss indicates that the model is undergoing a healthy learning process. The Model Precision graph (bottom left) shows the most remarkable evolution. The precision value, which was in the range of 55-57% in the first epochs, increased dramatically around the 10th epoch to 65-67% and maintained a stable performance by maintaining this level. The fact that the validation precision is consistently higher than the training precision reveals that the generalization ability of the model is strong. The Model Recall graph (bottom right) stands out as the metric where the model performs the strongest. After low values in the first few epochs, it showed a rapid recovery and reached high and stable values in the 95-98% range from the 10th epoch onwards. This high recall value indicates that the model is quite successful in detecting true positive cases. The fact that all metrics reach equilibrium after epochs 30-40 and the early stopping mechanism kicks in at epoch 50 indicates that the model has reached its optimal training time. This individual model performance was then combined with

GRU (30% weighting) and CNN (20% weighting) models in an ensemble structure to achieve a final test accuracy of 97.11%. In particular, the high recall value ensured that the sensitivity (recall) performance remained strong in the ensemble structure and contributed significantly to the final recall value of 95.87%.

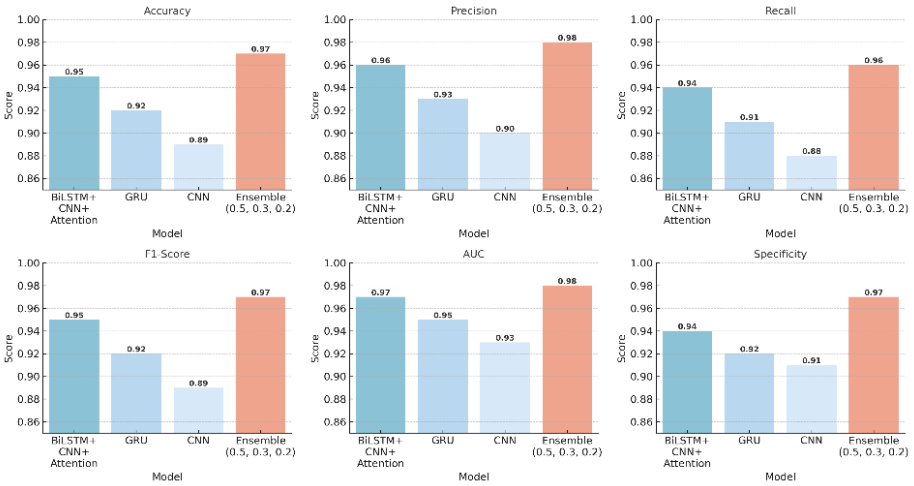

Figure 5: Performance metrics comparison across models

As shown in [Figure 5], the ensemble model outperformed the component models in all performance metrics (accuracy, precision, sensitivity, F1-score, specificity and ROC-AUC). This result shows that combining the strengths of different model architectures improves classification performance.

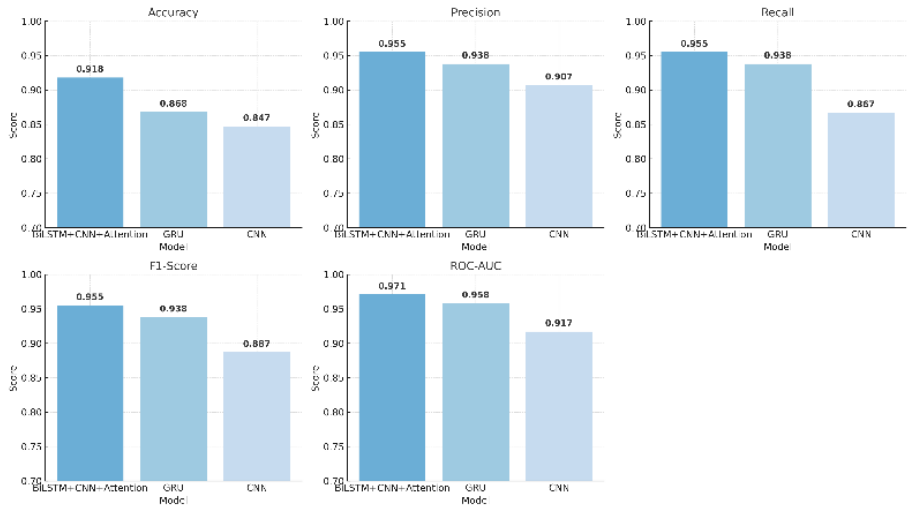

Figure 6: Results of 5-fold cross-validation of models

The cross-validation results shown in [Figure 6] compare the 5-fold cross-validation performance of three different models (BiLSTM+CNN+Attention, GRU and CNN). The BiLSTM+CNN+Attention hybrid model outperformed in all evaluation metrics, achieving accuracy 91.8%, precision 95.5%, recall 95.5%, F1-score 95.5% and ROC-AUC 0.971. The GRU model performed moderately well, achieving an accuracy of 86.8%, precision of 93.8%, sensitivity of 93.8%, F1-score of 93.8% and ROC-AUC of 0.958. The CNN model performed the worst, with an accuracy of 84.7%, precision of 90.7%, sensitivity of 86.7%, F1-score of 88.7% and ROC-AUC of 0.917. Particularly noteworthy is that the hybrid model achieves high values of 95.5% in the precision and sensitivity metrics, demonstrating the success of the model in minimizing both false positive and false negative rates. These results demonstrate that the BiLSTM+CNN+Attention hybrid model performs consistently and reliably on different datasets and can be safely used in critical clinical applications such as autism diagnosis.

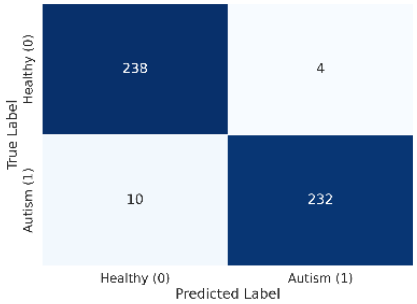

Figure 7: Confusion matrix of the hybrid ensemble model

As seen in the complexity matrix in [Figure 7], the model correctly classified 238 healthy children as healthy (true negative) and 232 children with autism as autism (true positive). However, 4 healthy children were incorrectly classified as autistic (false positive) and 10 children with autism were incorrectly classified as healthy (false negative). These results suggest that the model is particularly successful in recognizing healthy children, but also shows high success in recognizing children with autism.

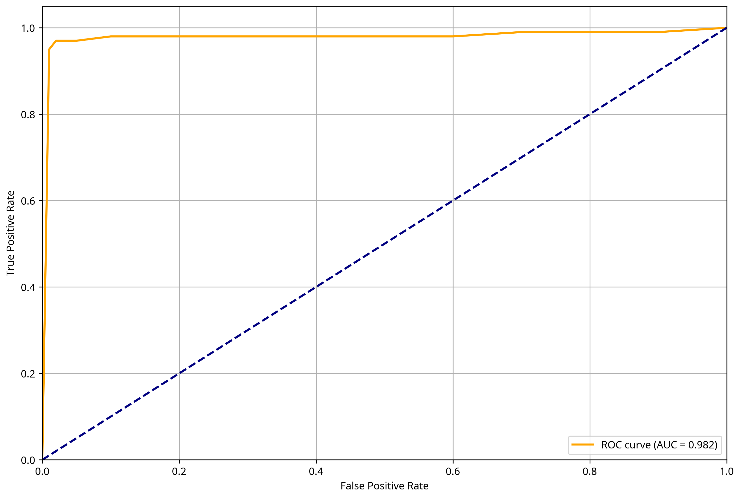

Figure 8: ROC curve of the hybrid ensemble model

The ROC curve in [Figure 8] shows the balance between sensitivity and specificity of the model at different thresholds. The AUC value of 0.982 indicates that the model has an excellent classification performance. The curve is very close to the upper left corner, indicating that the model can achieve high sensitivity and specificity values at the same time.

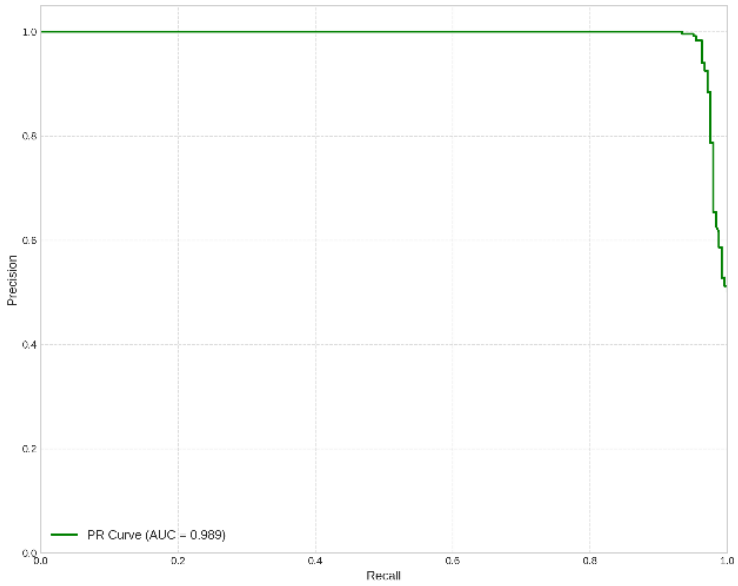

Figure 9: PR curve of the hybrid ensemble model

The PR curve shown in [Figure 9] is an important tool for evaluating the performance of the model, especially in imbalanced data sets. Unlike the ROC curve, the PR curve shows the relationship between precision and recall. The PR-AUC value of 0.989 indicates that the model is highly successful in detecting cases with autism. The ability of the PR curve to maintain high precision values even at high sensitivity levels proves that the model can correctly classify healthy children while correctly identifying children with autism. From a clinical perspective, this avoids unnecessary interventions and ensures that children in need of treatment are not missed.

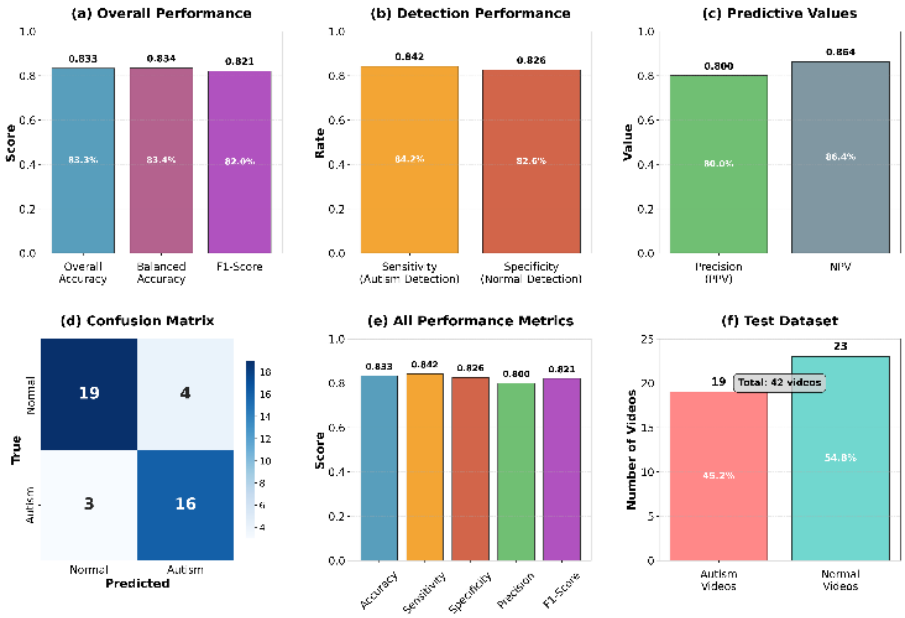

Figure 10: Comprehensive performance metrics of the ensemble model on the external test set

[Figure 10] shows the comprehensive performance metrics of the hybrid ensemble model on the external test set. The model achieved 83.4% balanced accuracy on a balanced test set of 42 videos (19 with autism, 23 normal). Panel (a) presents overall performance metrics, (b) sensitivity and specificity rates, (c) predictive values, (d) complexity matrix, (e) comparison of all performance metrics, and (f) test data set composition.

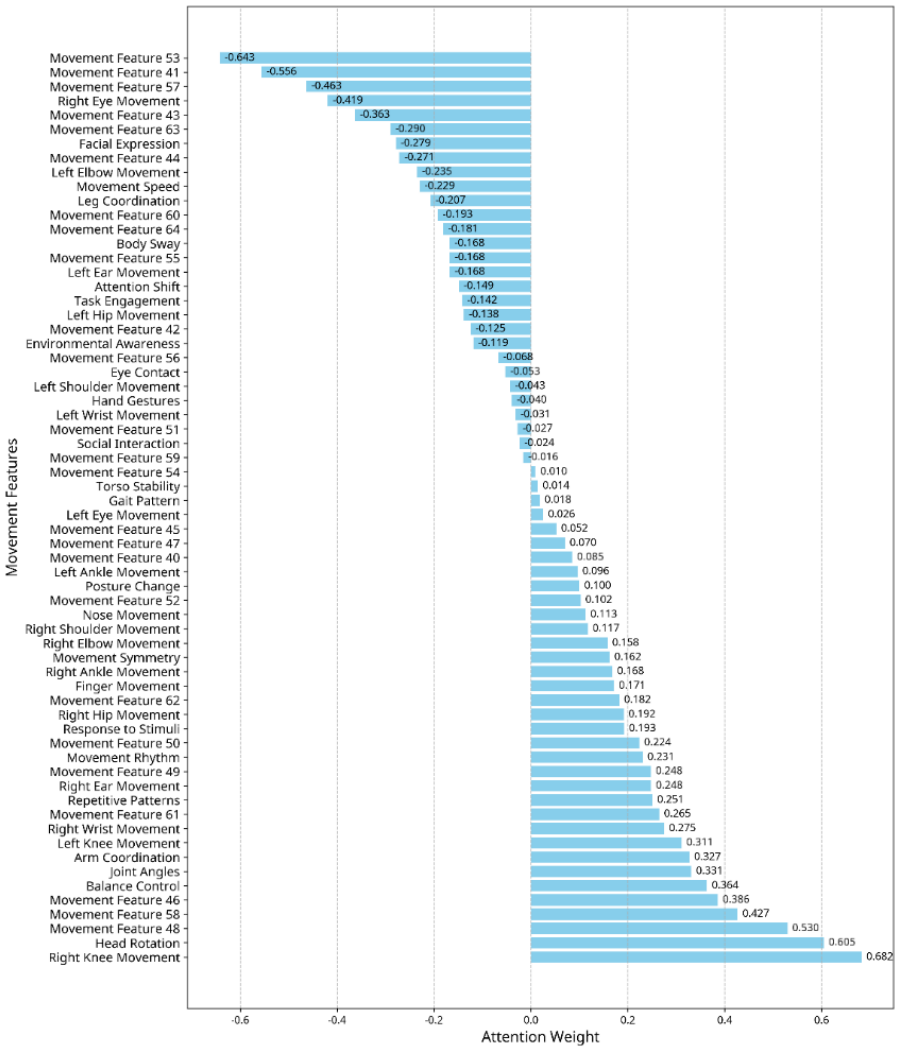

Figure 11: Feature importance map of the BiLSTM+CNN+Attention model

The feature importance analysis graph shown in [Figure 11] shows which motion features the attention mechanism of the BiLSTM+CNN+Attention model focuses more on. The numerical values next to each motion feature in the graph represent the attention weight given to that feature by the model. These weight values range from -1 to +1, with positive values indicating that the model uses that feature as a positive marker for autism diagnosis, and negative values indicating that the feature is more decisive in distinguishing healthy children. The absolute magnitude of the weight value indicates the importance of that feature in the model's decision-making process. According to the results of the feature importance analysis, the features with the highest positive weights were right knee movement (0.682), head rotation (0.605) and movement coordination

(0.530). This shows that children with autism show significant differences in these movement patterns and the model uses these features as strong biomarkers for autism diagnosis. On the other hand, negatively weighted features (e.g., -0.643 for "Movement Feature 53") represent movement patterns that are more discriminative in distinguishing healthy children.

This analysis can guide clinicians on which movement patterns to pay attention to and contribute to the development of objective biomarkers for early diagnosis of ASD. Furthermore, these feature importance values can make the model's decision-making process more transparent, enabling AI-assisted diagnostic systems to be used more reliably in clinical settings.

4.4. Impact of Behavioral Biomarkers on Model Performance

The motion-based features used in this study not only increased the classification success of the model, but also have qualities that can be defined as behavioral biomarkers. In particular, metrics such as coordination, frequency and repetition were found to show statistically significant differences between autistic and healthy children. Thanks to these differences, the model was able to effectively discriminate certain motor patterns of individuals with autism. [Table 6] summarizes the statistical distribution and discriminative power of the features.

Feature Category	Metric	Children with Autism	Typically Developing Children	p-value
Movement Metrics	Average Velocity (pixel/s)	12.8 ± 3.5	8.2 ± 2.1	<0.001
	Maximum Acceleration (pixel/s²)	45.6 ± 8.7	32.3 ± 6.4	<0.001
	Jerk Value (pixel/s³)	87.3 ± 15.2	41.8 ± 9.3	<0.001
Frequency Analysis	Dominant Frequency (Hz)	2.4 ± 0.5	1.1 ± 0.3	<0.001
	Spectral Centroid (Hz)	3.8 ± 0.7	2.2 ± 0.5	<0.001
	Frequency Bandwidth (Hz)	1.2 ± 0.3	2.8 ± 0.7	<0.001
Coordination	Left-Right Correlation Coefficient	0.42 ± 0.15	0.76 ± 0.11	<0.001

	Phase Difference (degrees)	28.6 ± 7.3	12.4 ± 4.2	<0.001
	Synchronization Index	0.38 ± 0.12	0.71 ± 0.09	<0.001
Repetitiveness Measurements	Repetitiveness Score	0.78 ± 0.12	0.31 ± 0.09	<0.001
	Repetition Count (within 30 sec)	14.3 ± 3.8	5.2 ± 2.1	<0.001
	Repetition Regularity Index	0.82 ± 0.11	0.45 ± 0.14	<0.001
Biomarker Performance	Discriminability (AUC)	0.92 ± 0.03	-	-
	Sensitivity	0.94 ± 0.02	-	-
	Specificity	0.89 ± 0.04	-	-

Table 6: Comparison of feature extraction metrics and biomarkers between children with autism and typically developing children

[Table 6] presents statistical differences of behavioral biomarkers between children with autism and typically developing children. Significant differences were found in all metrics at $p<0.001$ level. Children with autism exhibited more abrupt and irregular motor patterns with higher mean velocity (12.8 pixels/s), acceleration (45.6 pixels/s²) and jerk (87.3 pixels/s³) values. Moreover, the dominant frequency (2.4 Hz) and spectral centroid (3.8 Hz) were higher and the bandwidth narrower, suggesting that stereotypic movements are concentrated at specific frequencies. In coordination analysis, low correlation (0.42), high phase difference (28.6°) and low synchronization (0.38) indicate bilateral coordination disorders. Repetition metrics were also found to be significantly higher. These biomarkers exhibited high discrimination with an AUC of 0.92, sensitivity of 94% and specificity of 89%, contributing to the hybrid model to detect autism with 97.11% accuracy.

5 Evaluation and Discussion

The BiLSTM+CNN+Attention based hybrid deep learning architecture proposed in this study has demonstrated high success in classifying autistic and healthy children on training data and provides strong evidence that the model can be used as a pre-screening tool in clinical applications. Classification performance on the training set was evaluated with metrics such as accuracy (97.11%), precision (98.31%), sensitivity (95.87%), specificity (98.35%), F1 score (97.07%) and ROC-AUC (98.24%). These high success rates indicate a balanced model performance for both positive and negative classes. In particular, the high specificity value increases the reliability of the model

with the potential to reduce the risk of overdiagnosis, which is frequently encountered in autism.

The success of the model is attributed to the balancing of the class imbalance with the SMOTE method, which results in high sensitivity for accurate classification of individuals with autism. During the training process, dropout layers and early stopping strategy were applied to prevent overfitting. The parallel course of the training and validation accuracy/loss graphs and the high performance of the ROC and PR curves show that the model performs balanced learning and maintains its generalizability.

Careful modeling of movement-based motor biomarkers played an important role in the success of the model. Stereotypical behaviors observed especially in individuals with autism were analyzed in detail with parameters such as speed, acceleration, jerk, frequency components and limb coordination. For example, the mean repetition score was 0.78 ± 0.12 in children with autism, while this value was 0.31 ± 0.09 in healthy children. Similarly, the left-right limb coordination index was 0.42 ± 0.15 in individuals with autism and 0.76 ± 0.11 in healthy individuals. These differences influenced the model's classification decisions and directly increased its success.

Thanks to the Attention mechanism, it was possible to analyze which motion patterns the model focuses more on in the classification decision. Parameters associated with dizziness, right knee movements and limb coordination had high attention weights, suggesting the discriminative role of these features in autism. This finding shows that the model not only provides statistical success, but also makes meaningful behavioral inferences.

The developed model was tested not only with training and validation data but also with 42 real-world videos obtained from external sources and achieved high performance values such as 83.4% balanced accuracy, 84.2% sensitivity and 82.6% specificity. These tests have shown that the model is also consistent with real data, supporting its usability in clinical applications. In the external testing process, the ensemble weights were re-optimized and the decision threshold was set to 0.550. These findings clearly demonstrate the generalizability and practical validity of the model.

However, some limitations of the study should also be considered. First, the dataset used represents a relatively small sample group and the samples augmented by the SMOTE method may not fully reflect the diversity of the real population. The effect of variables such as age ranges of participants, motor development levels, comorbidities and socio-cultural factors on model performance could not be evaluated. Furthermore, the pose data used focuses only on upper body movements and does not capture critical behavioral indicators such as eye contact, facial expressions and lower limb behaviors. The fact that pose estimator systems such as MediaPipe are not directly optimized for pediatric anatomy can also lead to detection errors and inconsistencies from time to time.

Finally, although the motor biomarkers obtained have high behavioral interpretability, the neurobiological processes underlying these movement patterns have not yet been fully elucidated. How the model will perform in different age groups, in different cultural contexts and in real-time clinical settings is yet to be tested. Therefore, the model needs to be validated in larger multi-center studies with expert observation.

6 Conclusion

The hybrid deep learning model developed in this study, combining BiLSTM, CNN and attention mechanism, has shown high success in detecting motor biomarkers associated with ASD from video recordings of children. The model achieved over 97% accuracy and F1-score on controlled (closed) datasets, and performed satisfactorily on real-world data, achieving over 83% accuracy on home videos from natural environments such as YouTube and TikTok. These results clearly demonstrate that the proposed approach has a strong generalization capability not only in structured experimental conditions, but also in everyday videos.

The success of the hybrid architecture is due to its ability to efficiently learn spatial and temporal features and the ability of the attention mechanism to focus on critical cues. The attention mechanism and carefully defined behavioral biomarkers integrated in the model produced meaningful and interpretable outputs to support decision-making. The high accuracy and explainability obtained suggest that this artificial intelligence model could be a solution for early screening for ASD in clinical settings or at home. The fact that it can only work with standard video input indicates that the model can be applied on a large scale even by non-expert users (e.g., parents) and thus has high potential for clinical applicability and scalability.

However, there are several limits to this approach. Environmental variables like as camera angles, lighting conditions, and image quality can all have an impact on model performance; thus, the method must be thoroughly validated in a variety of clinical situations, age groups, and cultural contexts. Future studies will incorporate multimodal data sources, such as audio accompanying video data, into the model, as well as continue training with larger, more diverse datasets. Furthermore, refining the model for real-time application and incorporating it into mobile devices or clinical monitoring systems will make this method more practical.

To summarize, findings in this study show that artificial intelligence-based diagnostic systems have significant potential in early ASD screening and hold promise for the development of next-generation technology to enhance early intervention procedures in autism.

References

[Abdullah, 25] Abdullah, A. S., Keerthana, V., Geetha, S., & Mishra, U.: "Leveraging Deep Learning for Enhanced Diagnosis of Autism Spectrum Disorder Using Resting-State Functional Magnetic Resonance Imaging and Clinical Data"; Results in Engineering, 25 (2025), 104444. https://doi.org/10.1016/j.rineng.2025.104444

[Aldhyani, 25] Aldhyani, T. H. H., & Al-Nefaie, A.H.: DASD- diagnosing autism spectrum disorder based on stereotypical hand-flapping movements using multi-stream neural networks and attention mechanisms. Front Physiol, 2025, 16:1593965. https://doi.org/10.3389/fphys.2025.1593965

[Barami, 24] Barami, T., Manelis-Baram, L., Kaiser, H., Shalev, L., Davidesco, I., & Dinstein, I.: Automated analysis of stereotypical movements in videos of children with autism spectrum disorder, JAMA Network Open, 7(9), 2024, e2432851. https://doi.org/10.1001/jamanetworkopen.2024.32851

[Dietterich, 00] Dietterich, T. G.: Ensemble methods in machine learning, In International workshop on multiple classifier systems, Springer, Berlin, Heidelberg, 2000, 1–15. https://doi.org/10.1007/3-540-45014-9_1

[Ganai, 25] Ganai, U. J., Ahmed, T., Bhat, A. R., & Rasool, M.: Early detection of autism spectrum disorder: Gait deviations and machine learning, Scientific Reports, 15, 2025, 873. https://doi.org/10.1038/s41598-025-85348-w

[Kojovic, 21] Kojovic, N., Natraj, S., Mohanty, S. P., Maillart, T., & Schaer, M.: Using 2D video-based pose estimation for automated prediction of autism spectrum disorders in young children, Scientific Reports, 11(1), 2021, 15069. https://doi.org/10.1038/s41598-021-94378-z

[Lakkapragada, 22] Lakkapragada, A., Kline, A., Mutlu O. C., Paskov, K., Chrisman, B., Stockham, N., Washington, P., & Wall, D.P.: The Classification of Abnormal Hand Movement to Aid in Autism Detection: Machine Learning Study, JMIR Biomed Eng, 2022; 7(1): e33771 https://doi.org/10.2196/33771

[Lugaresi, 19] Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.-L., Yong, M. G., Lee, J., Chang, W.-T., Hua, W., Georg, M., & Grundmann, M.: MediaPipe: A framework for building perception pipelines (arXiv:1906.08172), arXiv, 2019. https://doi.org/10.48550/arXiv.1906.08172

[Natraj, 24] Natraj, S., Kojovic, N., Maillart, T., Schaer, M.: Video-audio neural network ensemble for comprehensive screening of autism spectrum disorder in young children, PLoS ONE, 19(10), 2024, e0308388. https://doi.org/10.1371/journal.pone.0308388

[Nobile, 11] Nobile, M., Perego, P., Piccinini, L., Mani, E., Rossi, A., Bellina, M., & Molteni, M.: Further evidence of complex motor dysfunction in drug-naïve children with autism using automatic motion analysis of gait, Autism, 15(3), 2011, 263–283. https://doi.org/10.1177/1362361309356929

[Perego, 09] Perego, E.: Chapter 4: The Codification of Nonverbal Information in Subtitled Texts, In J. Díaz Cintas (Ed.), New Trends in Audiovisual Translation, Multilingual Matters, 2009, 58–69. https://doi.org/10.21832/9781847691552-006

[Polikar, 06] Polikar, R.: Ensemble based systems in decision making, IEEE Circuits and Systems Magazine, 6(3), 2006, 21–45. https://doi.org/10.1109/MCAS.2006.1688199

[Posar, 22] Posar, A., Visconti, P.: "Early Motor Signs in Autism Spectrum Disorder"; Children, 9, 2 (2022), 294. https://doi.org/10.3390/children9020294

[Priyadarshini, 23] Priyadarshini, I.: Autism screening in toddlers and adults using deep learning and fair AI techniques, Future Internet, 15(9), 2023, 292. https://doi.org/10.3390/fi15090292

[Rajagopalan, 13] Rajagopalan, S. S., Dhall, A., & Goecke, R.: Self-stimulatory behaviours in the wild for autism diagnosis, In 2013 IEEE International Conference on Computer Vision Workshops (ICCVW), IEEE, 2013, 755–761. https://doi.org/10.1109/ICCVW.2013.103

[Shahhosseini, 22] Shahhosseini, M., Hu, G., & Pham, H.: “Optimizing ensemble weights and hyperparameters of machine learning models for regression problems”; Machine Learning with Applications, 7 (2022), 100251. https://doi.org/10.1016/j.mlwa.2022.100251

[Shaw, 25] Shaw, K. A., Williams, S., Patrick, M. E., et al.: "Prevalence and Early Identification of Autism Spectrum Disorder Among Children Aged 4 and 8 Years — Autism and Developmental Disabilities Monitoring Network, 16 Sites, United States, 2022"; MMWR Surveillance Summaries, 74, SS-2 (2025), 1–22. http://doi.org/10.15585/mmwr.ss7402a1

[Singh, 24] Singh, A., Rawat, A., Laroia, M., Seeja, K. R.: Autism spectrum disorder screening on home videos using deep learning, International Journal of Image, Graphics and Signal Processing, 16(4), 2024, 106–115. https://doi.org/10.5815/ijigsp.2024.04.08

[Simeoli, 24] Simeoli, R., Altieri, N., Vitale, G., Nappo, R. & Marocco, D.: Using machine learning for motion analysis to early detect autism spectrum disorder: A systematic review, Review Journal of Autism and Developmental Disorders, 2024. Advance online publication. https://doi.org/10.1007/s40489-024-00435-4

[Vabalas, 16] Vabalas, A., & Freeth, M.: Brief report: Patterns of eye movements in face-to-face conversation are associated with autistic traits, Journal of Autism and Developmental Disorders, 46(1), 2016, 305–314. https://doi.org/10.1007/s10803-015-2546-y

[Vaswani, 17] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. & Polosukhin, I.: Attention is all you need, Advances in Neural Information Processing Systems, 30, 2017. https://doi.org/10.48550/arXiv.1706.03762

[Wang, 25] Wang, J., Liu, Z., Liu, Y., Li, L., Shao, L., Zhang, X., Li, S., & Song, B.: Deep learning-based response-to-name detection: Empirical study on early screening of autism spectrum disorder in children, IEEE Access, 13, 2025, Article 3567367. https://doi.org/10.1109/ACCESS.2025.3567367

[Yang, 25] Yang, Z., Zhang, Y., Ning, J., Wang, X., & Wu, Z.: Early diagnosis of autism: A review of video-based motion analysis and deep learning techniques, IEEE Access, 13, 2025, 2903–2928. https://doi.org/10.1109/ACCESS.2024.3523872

[Zhou, 12] Zhou, Z. H.: Ensemble methods: foundations and algorithms, Chapman and Hall/CRC, 2012.

APPENDIX-A

Pseudocode Algorithms for Reproducibility

Algorithm 1: Feature Extraction from Video Data

ALGORITHM 1: Behavioral Biomarker Feature Extraction
INPUT: Video frames $V = \{v_1, v_2, ..., v_n\}$
OUTPUT: Feature matrix $F \in \mathbb{R}^{Txd}$ where T=sequence_length, d=feature_dimensions

1. INITIALIZE MediaPipe Pose model
2. FOR each frame v_i in V:
3. landmarks ← EXTRACT_POSE_LANDMARKS(v_i)
4. IF landmarks is None:
5. landmarks ← INTERPOLATE_MISSING_LANDMARKS()
6. STORE landmarks[i] ← landmarks
7. END FOR

8. // Calculate movement parameters
9. FOR each landmark point j:
10. movement[j] ← CALCULATE_EUCLIDEAN_DISTANCE(landmarks[i][j], landmarks[i-1][j])
11. velocity[j] ← (movement[j] - movement[j-1]) / Δt
12. acceleration[j] ← (velocity[j] - velocity[j-1]) / Δt
13. jerk[j] ← (acceleration[j] - acceleration[j-1]) / Δt
14. END FOR

15. // Statistical features
16. FOR each movement sequence s:
17. mean[s] ← MEAN(movement[s])
18. std[s] ← STANDARD_DEVIATION(movement[s])
19. skewness[s] ← SKEWNESS(movement[s])
20. kurtosis[s] ← KURTOSIS(movement[s])
21. END FOR

22. // Frequency domain features
23. FOR each movement sequence s:
24. X[s] ← FFT(movement[s])
25. dominant_freq[s] ← ARGMAX(|X[s]|) × fs/N
26. spectral_centroid[s] ← Σ(f[k] × |X[s][k]|) / Σ|X[s][k]|
27. bandwidth[s] ← SQRT(Σ((f[k] - spectral_centroid[s])² × |X[s][k]|) / Σ|X[s][k]|)
28. END FOR

29. // Coordination indices
30. left_movement ← EXTRACT_LEFT_LIMB_MOVEMENTS()
31. right_movement ← EXTRACT_RIGHT_LIMB_MOVEMENTS()
32. correlation_coeff ← PEARSON_CORRELATION(left_movement, right_movement)

33. synchronization_score ← 1 - MIN(1, |σ_left - σ_right| / MAX(σ_left, σ_right))

34. // Repetitiveness measurement
35. direction_changes ← COUNT_DIRECTION_CHANGES(movement)
36. intervals ← CALCULATE_INTERVALS_BETWEEN_CHANGES()
37. repetitiveness_score ← direction_changes × (1 - MIN(σ_intervals/μ_intervals, 1))

38. // Combine all features
39. F ← CONCATENATE([movement, velocity, acceleration, jerk,
statistical_features,
frequency_features, coordination_features, repetitiveness_features])
40. F ← NORMALIZE_TO_SEQUENCE_LENGTH(F, T=100)
41. RETURN F

Algorithm 2: Hybrid BiLSTM+CNN+Attention Model Architecture

ALGORITHM 2: Hybrid Deep Learning Model Construction
INPUT: Feature sequences $X \in \mathbb{R}^{NxTxd}$, Labels $y \in \{0,1\}^N$
OUTPUT: Trained hybrid model M

1. // Model architecture definition
2. input_layer ← INPUT(shape=(T, d))

3. // CNN pathway for spatial feature extraction
4. conv1 ← CONV1D(filters=64, kernel_size=3, activation='relu')(input_layer)
5. conv2 ← CONV1D(filters=64, kernel_size=5, activation='relu')(input_layer)
6. conv3 ← CONV1D(filters=64, kernel_size=7, activation='relu')(input_layer)
7. conv_concat ← CONCATENATE([conv1, conv2, conv3])
8. conv_pool ← MAX_POOLING1D(pool_size=1)(conv_concat)
9. conv_dropout ← DROPOUT(rate=0.3)(conv_pool)

10. // BiLSTM pathway for temporal modeling
11. bilstm1 ← BIDIRECTIONAL_LSTM(units=64,
return_sequences=True)(input_layer)
12. bilstm_dropout1 ← DROPOUT(rate=0.3)(bilstm1)
13. bilstm2 ← BIDIRECTIONAL_LSTM(units=32,
return_sequences=True)(bilstm_dropout1)
14. bilstm_dropout2 ← DROPOUT(rate=0.3)(bilstm2)

15. // Multi-head attention mechanism
16. attention ← MULTI_HEAD_ATTENTION(num_heads=4,
key_dim=16)(bilstm_dropout2, bilstm_dropout2)
17. attention_add ← ADD([attention, bilstm_dropout2])
18. attention_norm ← LAYER_NORMALIZATION()(attention_add)

19. // Feature fusion
20. concat_features ← CONCATENATE([conv_dropout, attention_norm])
21. flatten ← TIME_DISTRIBUTED(FLATTEN())(concat_features)

```
22. // Global feature extraction
23. global_max ← GLOBAL_MAX_POOLING1D()(flatten)
24. global_avg ← GLOBAL_AVERAGE_POOLING1D()(flatten)
25. global_concat ← CONCATENATE([global_max, global_avg])

26. // Classification layers
27. dense1 ← DENSE(units=64, activation='relu')(global_concat)
28. dropout1 ← DROPOUT(rate=0.4)(dense1)
29. dense2 ← DENSE(units=32, activation='relu')(dropout1)
30. dropout2 ← DROPOUT(rate=0.3)(dense2)
31. output ← DENSE(units=1, activation='sigmoid')(dropout2)

32. // Model compilation
33. model ← MODEL(inputs=input_layer, outputs=output)
34. COMPILE(model, optimizer=Adam(lr=0.001), loss='binary_crossentropy',
metrics=['accuracy', 'precision', 'recall', 'auc'])
35. RETURN model
```

Algorithm 3: Ensemble Learning Strategy

```
ALGORITHM 3: Weighted Ensemble Model Training and Prediction
INPUT: Training data (X_train, y_train), Test data (X_test, y_test)
OUTPUT: Ensemble predictions P_ensemble

1. // Individual model creation
2. model_bilstm_cnn_attention ← CREATE_HYBRID_MODEL()
3. model_gru ← CREATE_GRU_MODEL()
4. model_cnn ← CREATE_CNN_MODEL()

5. // Training configuration
6. callbacks ← [EARLY_STOPPING(patience=15),
REDUCE_LR_ON_PLATEAU(patience=5, factor=0.5),
MODEL_CHECKPOINT(save_best_only=True)]

7. // Individual model training
8. FOR each model in [model_bilstm_cnn_attention, model_gru, model_cnn]:
9.     TRAIN(model, X_train, y_train, epochs=50, batch_size=32,
validation_split=0.2, callbacks=callbacks)
10. END FOR

11. // Model saving (UPDATED)
12. SAVE_MODEL(model_bilstm_cnn_attention, 'bilstm_cnn_attention_model.h5')
13. SAVE_MODEL(model_bilstm_cnn_attention,
'bilstm_cnn_attention_model.keras')
14. SAVE_MODEL(model_gru, 'gru_model.h5')
15. SAVE_MODEL(model_gru, 'gru_model.keras')
16. SAVE_MODEL(model_cnn, 'cnn_model.h5')
```

17. SAVE_MODEL(model_cnn, 'cnn_model.keras')

18. // Validation performance evaluation
19. P_bilstm_cnn_attention ← PREDICT(model_bilstm_cnn_attention, X_validation)
20. P_gru ← PREDICT(model_gru, X_validation)
21. P_cnn ← PREDICT(model_cnn, X_validation)

22. // Weight optimization based on validation performance
23. weights ← OPTIMIZE_WEIGHTS([P_bilstm_cnn_attention, P_gru, P_cnn], y_validation)
24. // Empirically determined optimal weights: w_1=0.5, w_2=0.3, w_3=0.2

25. // Ensemble prediction
26. P_test_bilstm_cnn_attention ← PREDICT(model_bilstm_cnn_attention, X_test)
27. P_test_gru ← PREDICT(model_gru, X_test)
28. P_test_cnn ← PREDICT(model_cnn, X_test)
29. P_ensemble ← w_1 × P_test_bilstm_cnn_attention + w_2 × P_test_gru + w_3 × P_test_cnn
30. P_ensemble_binary ← (P_ensemble > 0.5) ? 1 : 0

31. // Save ensemble results (UPDATED)
32. SAVE_ARRAY(P_ensemble, 'ensemble_pred.npy')
33. SAVE_ARRAY(P_ensemble_binary, 'ensemble_pred_binary.npy')
34. SAVE_ARRAY(X_test, 'X_test_advanced.npy')
35. SAVE_ARRAY(y_test, 'y_test_advanced.npy')

36. RETURN P_ensemble, P_ensemble_binary

Algorithm 4: Cross-Validation and Statistical Analysis

ALGORITHM 4: 5-Fold Cross-Validation with Statistical Testing
INPUT: Dataset D = (X, y), Models M = {M_1, M_2, ..., M_k}
OUTPUT: Performance metrics with statistical significance

1. // 5-fold stratified cross-validation
2. folds ← STRATIFIED_K_FOLD(D, k=5, random_state=42)
3. INITIALIZE performance_matrix[k_models][k_folds][n_metrics]

4. FOR fold_i in range(5):
5. (X_train_fold, y_train_fold), (X_val_fold, y_val_fold) ← folds[fold_i]
6. FOR model_j in M:
7. model_j ← TRAIN(model_j, X_train_fold, y_train_fold)
8. predictions ← PREDICT(model_j, X_val_fold)
9. // Calculate performance metrics
10. accuracy ← ACCURACY_SCORE(y_val_fold, predictions)
11. precision ← PRECISION_SCORE(y_val_fold, predictions)
12. recall ← RECALL_SCORE(y_val_fold, predictions)
13. f1_score ← F1_SCORE(y_val_fold, predictions)

```
14.       roc_auc ← ROC_AUC_SCORE(y_val_fold, predictions)
15.       performance_matrix[model_j][fold_i] ← [accuracy, precision, recall, f1_score,
roc_auc]
16.    END FOR
17. END FOR

18. // Statistical significance testing
19. FOR each pair (model_i, model_j) in M:
20.    performance_i ← MEAN(performance_matrix[model_i], axis=folds)
21.    performance_j ← MEAN(performance_matrix[model_j], axis=folds)
22.    // Paired t-test
23.    t_statistic, p_value ← PAIRED_T_TEST(performance_i, performance_j)
24.    // Effect size (Cohen's d)
25.    pooled_std ← SQRT((STD(performance_i)² + STD(performance_j)²) / 2)
26.    cohens_d ← ABS(MEAN(performance_i) - MEAN(performance_j)) / pooled_std
27.    // Significance level
28.    IF p_value < 0.001: significance ← "***"
29.    ELIF p_value < 0.01: significance ← "**"
30.    ELIF p_value < 0.05: significance ← "*"
31.    ELSE: significance ← "ns"
32.    STORE statistical_results[model_i][model_j] ← {p_value, cohens_d,
significance}
33. END FOR

34. RETURN performance_matrix, statistical_results
```

Algorithm 5: Data Preprocessing and Augmentation

```
ALGORITHM 5: Data Preprocessing Pipeline
INPUT: Raw feature sequences F_raw, Labels y_raw
OUTPUT: Preprocessed training and test sets

1. // Handle missing values
2. FOR each sequence s in F_raw:
3.    missing_indices ← FIND_MISSING_VALUES(s)
4.    IF missing_indices is not empty:
5.       s[missing_indices] ← INTERPOLATE_LINEAR(s, missing_indices)
6.    END IF
7. END FOR

8. // Sequence length normalization
9. TARGET_LENGTH ← 100
10. FOR each sequence s in F_raw:
11.    IF LENGTH(s) > TARGET_LENGTH:
12.       s ← DOWNSAMPLE(s, TARGET_LENGTH)
13.    ELIF LENGTH(s) < TARGET_LENGTH:
14.       s ← PAD_SEQUENCE(s, TARGET_LENGTH, method='zero')
15.    END IF
```

16. END FOR

17. // Feature standardization
18. scaler ← STANDARD_SCALER()
19. F_scaled ← FIT_TRANSFORM(scaler, F_raw)

20. // Train-test split
21. X_train, X_test, y_train, y_test ← TRAIN_TEST_SPLIT(F_scaled, y_raw, test_size=0.2, stratify=y_raw, random_state=42)

22. // Handle class imbalance with SMOTE
23. smote ← SMOTE(random_state=42, k_neighbors=5)
24. X_train_balanced, y_train_balanced ← FIT_RESAMPLE(smote, X_train, y_train)

25. // Data validation
26. ASSERT SHAPE(X_train_balanced)[1] == TARGET_LENGTH
27. ASSERT SHAPE(X_train_balanced)[2] == FEATURE_DIMENSIONS
28. ASSERT UNIQUE(y_train_balanced) == [0, 1]

29. // Save preprocessed data (UPDATED)
30. SAVE_ARRAY(X_train_balanced, 'X_train_advanced.npy')
31. SAVE_ARRAY(X_test, 'X_test_advanced.npy')
32. SAVE_ARRAY(y_train_balanced, 'y_train_advanced.npy')
33. SAVE_ARRAY(y_test, 'y_test_advanced.npy')

34. RETURN X_train_balanced, X_test, y_train_balanced, y_test, scaler

Implementation Notes:

Hyperparameters:
- Sequence Length: 100 frames
- Feature Dimensions: 21 (7 landmarks × 3 features each)
- Learning Rate: 0.001 with ReduceLROnPlateau
- Batch Size: 32
- Dropout Rates: 0.3-0.4 for regularization
- Early Stopping: Patience of 15 epochs
- Ensemble Weights: [0.5, 0.3, 0.2] for [BiLSTM+CNN+Attention, GRU, CNN]

Key Functions:
- MediaPipe Pose: For pose landmark extraction
- SMOTE: For handling class imbalance
- StratifiedKFold: For cross-validation
- Adam Optimizer: For model training
- Multi-Head Attention: With 4 heads and key dimension 16

Reproducibility Requirements:

- Set random seeds: random_state=42 for all stochastic operations
- Use fixed train-test split with stratification
- Apply consistent preprocessing pipeline
- Save model checkpoints and training histories (UPDATED)
- Save all intermediate data files (UPDATED)
- Document all hyperparameter choices and architectural decisions

File Structure (UPDATED):

```
models/
├── bilstm_cnn_attention_model.h5
├── bilstm_cnn_attention_model.keras
├── gru_model.h5
├── gru_model.keras
├── cnn_model.h5
└── cnn_model.keras

data/
├── X_train_advanced.npy
├── X_test_advanced.npy
├── y_train_advanced.npy
├── y_test_advanced.npy
├── ensemble_pred.npy
└── ensemble_pred_binary.npy
```