

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

Overview

This implementation presents a hybrid deep learning approach for detecting autism spectrum disorder (ASD) in children through analysis of motor biomarkers extracted from video recordings. The system combines BiLSTM, CNN, and Multi-Head Attention mechanisms in an ensemble architecture to achieve high-accuracy autism detection.

System Requirements

Hardware Specifications

- **Processor:** Intel(R) Core(TM) i5-120U @ 1.40 GHz
- **Memory:** 16.0 GB RAM (15.6 GB usable)
- **Storage:** 423 GB available space
- **Graphics:** Intel(R) Graphics (integrated)
- **System Type:** 64-bit operating system, x64-based processor

Software Environment

- **Operating System:** Windows 11 (64-bit)
- **Python Version:** 3.8+ (recommended 3.9)
- **Platform:** Windows x64

Dependencies and Libraries

Core Deep Learning Framework

Plain Text

```
tensorflow>=2.8.0  
keras>=2.8.0
```

Data Processing and Analysis

Plain Text

```
numpy>=1.21.0  
pandas>=1.3.0  
scikit-learn>=1.0.0  
scipy>=1.7.0
```

Computer Vision and Pose Detection

Plain Text

```
mediapipe>=0.8.9  
opencv-python>=4.5.0
```

Data Balancing

Plain Text

```
imbalanced-learn>=0.8.0
```

Visualization

Plain Text

```
matplotlib>=3.5.0  
seaborn>=0.11.0
```

Utilities

Plain Text

```
pickle  
os
```

Installation

Install dependencies

Bash

```
pip install tensorflow keras numpy pandas scikit-learn scipy mediapipe  
opencv-python imbalanced-learn matplotlib seaborn
```

Project Structure

Plain Text

```
|— feature_engineering.py      # Behavioral biomarker extraction  
|— train_hybrid_model_fixed2.py # Main hybrid model training  
|— evaluate_ensemble_model.py  # Ensemble model evaluation  
|— evaluate_models.py          # Individual model evaluation  
|— preprocess_data.py          # Data preprocessing pipeline  
|— create_pose_landmark_visualization.py # Visualization tools  
|— kare_bazli_ozellik_cikarimi.py # Frame-based feature extraction  
|— models/                    # Trained model files  
|— results/                   # Evaluation results
```

Usage

1. Data Preprocessing

Bash

```
python preprocess_data.py
```

2. Feature Extraction

Bash

```
python feature_engineering.py
```

3. Model Training

Bash

```
python train_hybrid_model_fixed2.py
```

4. Model Evaluation

Bash

```
python evaluate_ensemble_model.py  
python evaluate_models.py
```

5. Visualization

Bash

```
python create_pose_landmark_visualization.py
```

Model Architecture

Hybrid Ensemble Components

1. BiLSTM + CNN + Attention Model (50% weight)

- Bidirectional LSTM layers (64, 32 units)
- 1D CNN with multiple kernel sizes (3, 5, 7)
- Multi-head attention mechanism (4 heads)

2. GRU Model (30% weight)

- Two GRU layers (64, 32 units)
- Dropout regularization

3. CNN Model (20% weight)

- Three 1D convolutional layers (64, 128, 128 filters)
- Global pooling layers

Training Configuration

- **Optimizer:** Adam ($\beta_1=0.9$, $\beta_2=0.999$)
- **Learning Rate:** 0.001
- **Batch Size:** 32
- **Max Epochs:** 50
- **Loss Function:** Binary Cross-entropy
- **Regularization:** Dropout (20-40%),
- **Callbacks:** EarlyStopping, ReduceLROnPlateau

Performance Metrics

Ensemble Model Results

- **Accuracy:** 97.11%
- **Precision:** 98.31%
- **Recall:** 95.87%
- **F1-Score:** 97.07%
- **ROC-AUC:** 98.24%
- **Specificity:** 98.35%

Individual Model Performance

Model	Accuracy	Precision	Recall	F1-Score	AUC
BiLSTM+CNN+Attention	95.0%	96.0%	94.0%	95.0%	97.0%
GRU	92.0%	93.0%	91.0%	92.0%	95.0%
CNN	89.0%	90.0%	88.0%	89.0%	93.0%

Behavioral Biomarkers

Feature Categories

- Movement Parameters:** Velocity, acceleration, jerk
- Statistical Features:** Mean, std, skewness, kurtosis
- Frequency Domain:** Dominant frequency, spectral centroid, bandwidth
- Coordination Indices:** Left-right correlation, synchronization score
- Repetitiveness Measures:** Repetition count, regularity index

Data Requirements

Input Format

- Video Files:** MP4, AVI formats supported
- Frame Rate:** 10 FPS (recommended)
- Duration:** 30 seconds minimum
- Content:** Upper body movements of children (1-10 years)

Dataset Structure

- Training Set:** 80% (1,547 sequences)

- **Test Set:** 20% (484 sequences)
- **Class Balance:** SMOTE applied for balanced training

Reproducibility

Pseudocode Algorithms

Complete pseudocode algorithms are provided in Appendix A of the accompanying research paper, covering:

- Feature extraction pipeline
- Hybrid model architecture
- Ensemble learning strategy
- Cross-validation procedures
- Statistical analysis methods

Contact

For questions or issues, please contact:

- Email: yelda.firat@mudanya.edu.tr