Sense AI Project Documentation

Authors:

Mohammed Al-Madhoun
Layan Khaddash
Shatha Al-Ghariri

Date: October 22, 2025

Abstract

This project presents a real-time multimodal system for autism therapy and screening. It integrates pose estimation, optical flow, and deep learning models (e.g., TCN) to analyze 2D skeleton data during therapy sessions, providing real-time activity recognition and feedback through a lightweight web interface.

Additionally, a Flask-based web application implements behavioral screening using the AQ-10 questionnaire. User responses are processed by a trained Decision Tree model (`.pkl` file) to predict ASD traits, delivering both classification and probability scores. The system is designed for accessibility, enabling caregivers and educators to perform rapid preliminary assessments without clinical supervision.

Introduction

Autism Spectrum Disorder (ASD) is a neurodevelopmental condition that affects communication, social interaction, and motor coordination. Early diagnosis and continuous therapeutic intervention are essential for improving behavioral and social outcomes. However, assessing children's engagement and movement quality during therapy sessions often depends on subjective human observation, which can be time-consuming, inconsistent, and limited in scalability.

To address this challenge, the Sense AI Autism Detection System introduces an intelligent, privacy-preserving platform designed to analyze therapeutic movements of children with ASD using machine learning, computer vision and deep learning. The system leverages the MMASD (Multimodal Dataset for Autism Intervention Analysis) — an open-access benchmark dataset containing over 1,315 video samples, 244,000+ frames, and 11 activity classes recorded from 32 children during real-world therapy sessions. MMASD provides four synchronized, privacy-safe modalities: optical flow, 2D skeletons (OpenPose), 3D skeletons (ROMP), and clinical evaluation scores such as ADOS. These modalities capture the essential motion and behavioral patterns needed to evaluate performance without exposing identifiable personal data.

The proposed system enables real-time pose estimation and activity recognition through a three-phase process:

Exercise Recording: Users select an activity and record a short exercise while the system tracks body movements via a live 2D skeleton overlay.
Caregiver or Clinician Questionnaire: After completing at least six recorded sessions, a caregiver or therapist fills out a behavioral and performance questionnaire, providing qualitative insights that complement the motion data.
Confidence Verification (Optional): If the AI model detects uncertainty, a short follow-up form is triggered to clarify inconsistent or ambiguous responses.

All processing occurs locally, and only skeletal keypoints are stored — ensuring that no identifiable imagery is saved or transmitted. The system also aligns with existing ASD screening tools such as the AQ-10, integrating their structured evaluation logic while maintaining a strictly non-diagnostic, assistive purpose.

System Mechanism

Mechanism of the MMASD project, which includes (1) a real-time video-based activity recognition module and (2) a web-based AQ-10 screening interface for early autism detection.

1. Video-Based Activity Recognition

1. **Video Input:** Therapy session videos are loaded and sampled at 25–30 FPS.
2. **Pose Estimation:** Each frame is processed using OpenPose (2D) and ROMP (3D) to extract skeletal joint coordinates.
3. **Optical Flow:** Motion vectors are computed using the Lucas–Kanade method to capture movement dynamics.
4. **Temporal Modeling:** Sequences of skeleton data are formatted into fixed-length windows and passed to a Temporal Convolutional Network (TCN).
5. **Activity Classification:** The TCN model predicts activity labels (e.g., imitation, turn-taking) for each frame window.
6. **Feedback and Logging:** Predicted labels are overlaid on video frames in real time and logged for therapist review.

2. AQ-10 Web Screening Tool

1. **Frontend Interface:** A Flask-based web application presents the ten AQ-10 screening questions using a clean HTML/CSS interface. Each question offers two response options: *Agree* or *Disagree*.
2. **User Input Handling:** Upon submission, the user's answers are encoded as binary values (1 for Agree, 0 for Disagree).
3. **Model Integration:** The encoded input is passed to a pre-trained Decision Tree classifier loaded from a `.pkl` file.
4. **Prediction Output:** The model returns a binary classification (ASD traits detected or not) along with a probability score. The result is displayed on a separate results page.
5. **Deployment:** The app runs locally via Flask and can be deployed using a WSGI server such as Gunicorn for production use.

Integration Summary

The MMASD system combines multimodal video analysis with accessible behavioral screening. While the video module supports therapists in real-time activity recognition, the AQ-10 web tool enables rapid, self-guided screening for caregivers and educators. Together, they form a hybrid pipeline for early autism support and intervention.

System Architecture

The **Sense AI Autism Detection System** follows a modular, real-time client–server design that connects a pose estimation backend with a lightweight browser-based visualization frontend. It is built using **Flask (Python)** for the backend processing and **JavaScript + HTML + CSS** for the front-end rendering. The architecture is organized into three core layers:

1. Data Input and Pose Extraction

The backend uses the **MediaPipe Pose** model to extract 33 anatomical landmarks from each video frame. These landmarks are then mapped to the **COCO-17** joint format to maintain consistency with standard motion analysis datasets such as MMASD. The camera feed is captured via `OpenCV (cv2.VideoCapture)`, and frames are processed at approximately 25–30 FPS. For each frame, the system:

- Converts the frame from BGR to RGB.
- Runs MediaPipe's neural network to infer 2D joint coordinates and visibility scores.
- Reorders and filters them to match the COCO-17 keypoint indices.
- Applies **Exponential Moving Average (EMA)** smoothing to stabilize joint movement and reduce jitter.

All pose keypoints and confidence scores are temporarily stored in memory for low-latency access.

2. Flask Server and API Endpoints

The Flask server acts as the communication bridge between the vision backend and the web client. It exposes several REST endpoints:

- `/video` – Streams live MJPEG video frames with skeleton overlays for real-time preview.
- `/pose.json` – Provides the most recent COCO-17 keypoints and confidence scores in JSON format for browser polling.
- `/record/start`, `/record/stop`, and `/record/reset` – Manage recording sessions for each selected activity, storing session data as JSON files under organized folders (`sessions/<section_id>/<activity_id>/`).

Each JSON file contains frame-by-frame data, including pixel coordinates, normalized coordinates, timestamps, and confidence values.

3. Front-End Visualization and Interaction

The front end, written in **JavaScript (static/app.js)**, continuously polls the `/pose.json` endpoint (at ∼25 Hz) and renders the live skeleton overlay onto an HTML `<canvas>` element. The rendering pipeline includes:

- Drawing of body joints and connecting edges defined by the COCO-17 skeleton topology.
- Real-time EMA smoothing for motion continuity.

This architecture ensures the system runs entirely **locally** without cloud dependencies, preserving data privacy. All video data remains on-device; only skeletal coordinates and session metadata are saved for analysis.

Dataset

This study utilizes two complementary datasets to support both behavioral and questionnaire-based autism analysis. The first, **MMASD (Multimodal Autism Dataset)**, captures movement and social interaction data during therapy sessions. The second, the **ASD Screening Dataset for Children**, contains behavioral questionnaire responses and demographic data from the UCI Machine Learning Repository. Together, they enable a comprehensive view of autism-related patterns across multimodal and cognitive dimensions.

1. MMASD (Multimodal Autism Dataset)

MMASD is an open-access dataset designed for studying movement and social behavior in therapy sessions of children with Autism Spectrum Disorder (ASD). It provides synchronized, privacy-preserving modalities for motion analysis and clinical research.

Overview

- **Participants:** 32 children with ASD (27 male, 5 female), aged 5–12.
- **Context:** Play-therapy interventions with a triadic setting (child, trainer, model).
- **Clips:** 1,315 curated sessions, ∼244K frames, recorded at 25–30 FPS.
- **Duration:** Over 108 hours of recordings.
- **Resolution:** 320×240 to 1920×1080.

Modalities

- **Optical Flow:** Motion-only frames (Lucas–Kanade), shape $(L-1, H, W, 2)$.
- **2D Skeleton:** OpenPose COCO-17 joints, JSON per frame.
- **3D Skeleton:** ROMP (SMPL/24 joints), NPZ per frame.
- **Clinical Data:** CSV with ID, gender, age, ADOS-2 scores, and severity.

2. ASD Screening Dataset for Children (UCI)

The **ASD Screening Dataset for Children** is an open-access dataset curated by the UCI Machine Learning Repository. It supports early detection of Autism Spectrum Disorder through behavioral screening questionnaires and demographic features. The dataset is suitable for classification and predictive modeling tasks in healthcare and developmental psychology.

Overview

- **Participants:** 292 children (male and female), aged 4–11 years.
- **Context:** Responses to the AQ-10 screening questionnaire, along with demographic and behavioral attributes.
- **Features:** 20 raw features including 10 AQ-10 scores, age, gender, ethnicity, family history, and screening results.
- **Target:** Binary classification label indicating ASD traits (1) or not (0).

Modalities

- **AQ-10 Scores:** Ten binary features (A1–A10) representing responses to the Autism Quotient screening questions.
- **Demographics:** Age (numerical), gender, ethnicity, and parental relation.
- **Behavioral Indicators:** App usage frequency, screen time, and family ASD history.
- **Clinical Label:** Final column `Class/ASD` used as the prediction target.

Preprocessing

- **Missing Data:** Age filled with median value; ethnicity filled with mode.
- **Encoding:** Categorical variables one-hot encoded; binary columns mapped to 0/1.
- **Cleaning:** Removed nulls, standardized column names, and exported cleaned data to `after_prepro_cleaned.csv`.
- **Feature Selection:** AQ-10 columns isolated for specialized model training.

AQ-10 Question Mapping

Each AQ-10 column corresponds to a screening question designed to detect autistic traits. The mapping is as follows:

- `A1_Score`: I often notice small sounds when others do not.
- `A2_Score`: I usually concentrate more on the whole picture, rather than the small details.
- `A3_Score`: I find it easy to do more than one thing at once.
- `A4_Score`: If there is an interruption, I can switch back to what I was doing very quickly.
- `A5_Score`: I find it easy to 'read between the lines' when someone is talking to me.
- `A6_Score`: I know how to tell if someone listening to me is getting bored.
- `A7_Score`: When I'm reading a story, I find it difficult to work out the characters' intentions.
- `A8_Score`: I like to collect information about categories of things.
- `A9_Score`: I find it easy to work out what someone is thinking or feeling just by looking at their face.
- `A10_Score`: I find it difficult to work out people's intentions.

Web Application Integration

To make the screening tool accessible, a Flask-based web application was developed. The app includes:

- **Frontend:** HTML templates with CSS styling present the AQ-10 questions as multiple-choice options ("Agree" or "Disagree").

- **Backend:** Flask handles routing, form submission, and prediction logic.
- **Model Integration:** User responses are converted to binary inputs and passed to a trained Decision Tree model loaded from a `.pkl` file.
- **Output:** The app returns a prediction ("ASD traits detected" or "No ASD traits detected") along with the estimated probability.

Model Performance

Multiple classifiers were evaluated using cross-validation and test set metrics. The top-performing models included Gradient Boosting, Random Forest, and XGBoost. Metrics included:

- **Accuracy:** Up to 100% on test set for Gradient Boosting.
- **F1 Macro Score:** Used to balance precision and recall across classes.
- **Precision and Recall:** Evaluated for both ASD and non-ASD classes.

Source

Dataset available at: https://archive.ics.uci.edu/dataset/419/autistic+spectrum+disorder+screening+data+for+children
Each activity in the MMASD dataset was designed to encourage engagement, coordination, and body awareness during therapy. They include rhythmic arm and body movements, musical interactions like drumming and maracas shaking, and balanced postures such as tree and twist poses. Together, these exercises provide a diverse range of motor patterns used to train and evaluate the real-time pose recognition model, ensuring that the system can accurately analyze body motion while maintaining participant privacy through skeleton-based visualization.

Attribution

When using MMASD, cite:

> Li et al. (2023). *MMASD: A Multimodal Dataset for Autism Intervention Analysis.* Proceedings of ACM ICMI, Paris, France. https://github.com/Li-Jicheng/MMASD-A-Multimodal-Dataset-for-Autism-Intervention-Analysis

Source dataset also referenced from UCI Machine Learning Repository: https://archive.ics.uci.edu/dataset/419/autistic+spectrum+disorder+screening+data+for+children

Future Work

As an enhancement to the current system, we plan to implement a supplementary decision-support module. When the model is uncertain about ASD classification, the user will be redirected to an additional page containing questions about maternal health and psychological factors. These questions follow established clinical criteria used in autism centers, providing an extra diagnostic reference to improve model reliability and decision-making.