SMARTGUARD: AN AI-POWERED DEFENSE SYSTEM

Group 38

Amit Kumar Yadav
Ayush Singh
Sanath Naveen Sharma
Shorya Dixit
Sourav Saini
Vansh Singh

Problem Statement

The current security measures in our high-security region need improvement, particularly in the area of surveillance. The current system lacks the ability to efficiently trace and monitor detected unknown persons.

we propose the integration of Machine Learning (ML) models with robotic surveillance technologies, which includes the incorporation of a Bi-Axial Angular positioning assembly controlled by Arduino code. This approach aims to enhance our defense mechanism by providing an open-ended pointer to efficiently trace and monitor detected unknown persons.

Project Goals

This project aims to develop a real-time intrusion detection system capable of:

- Identifying faces in a video stream using Machine Learning models.
- Recognising known individuals using a face recognition model. Flagging and tracking individuals not recognised in the database (potential intruders).
- Calculating the intruder's location within the monitored area. Directing a camera mounted on a Bi-Axial Angular Positioning System to follow the intruder's movement.

System Overview:

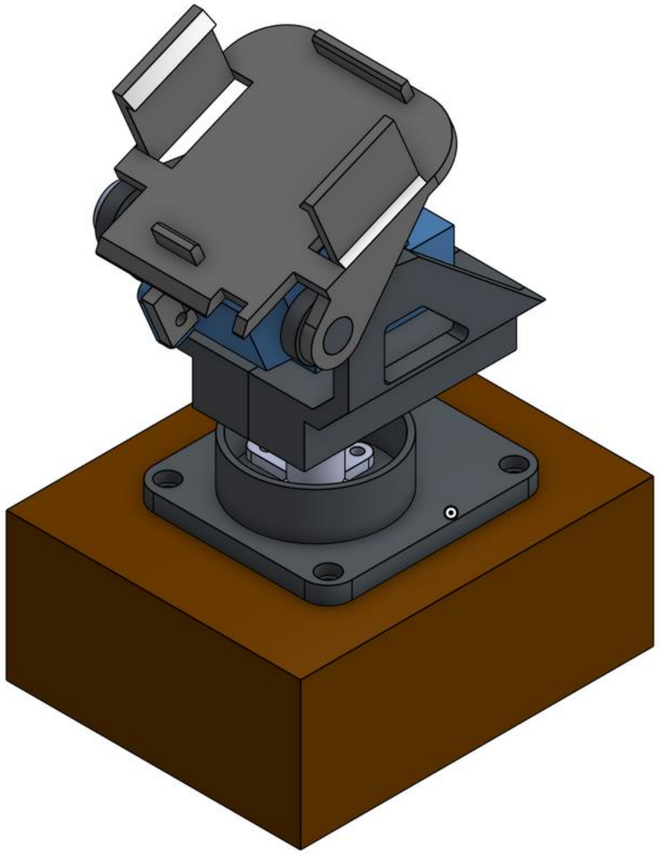
The proposed system comprises three main components:

- Machine Learning Pipeline
- Hardware Assembly
- Integration Algorithm

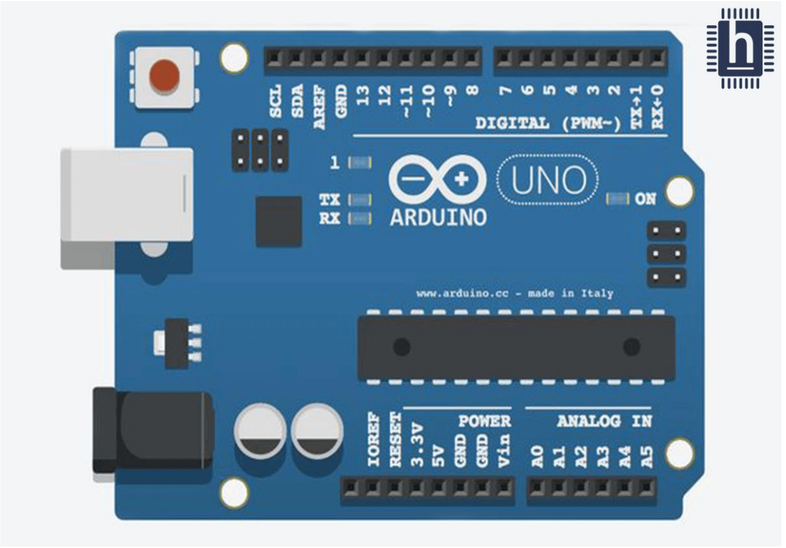
Hardware Assembly

Bi-Axial Angular Positioning System:

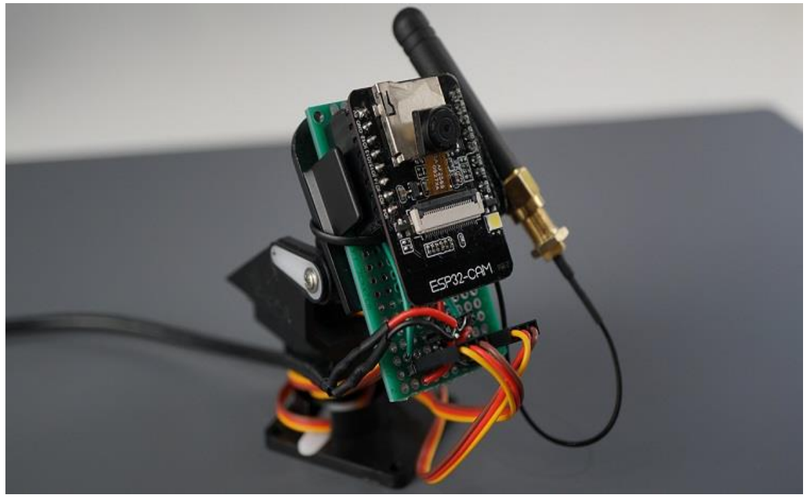
- The Bi-Axial Angular Positioning System comprises a Two-Axis Servo Assembly, controlled by an Arduino Uno.
- The assembly facilitates angular adjustments along both pan and tilt axes, providing precise positioning for the camera.
- The Arduino Uno serves as the control center, orchestrating synchronized movements based on the ML model's angular coordinates.



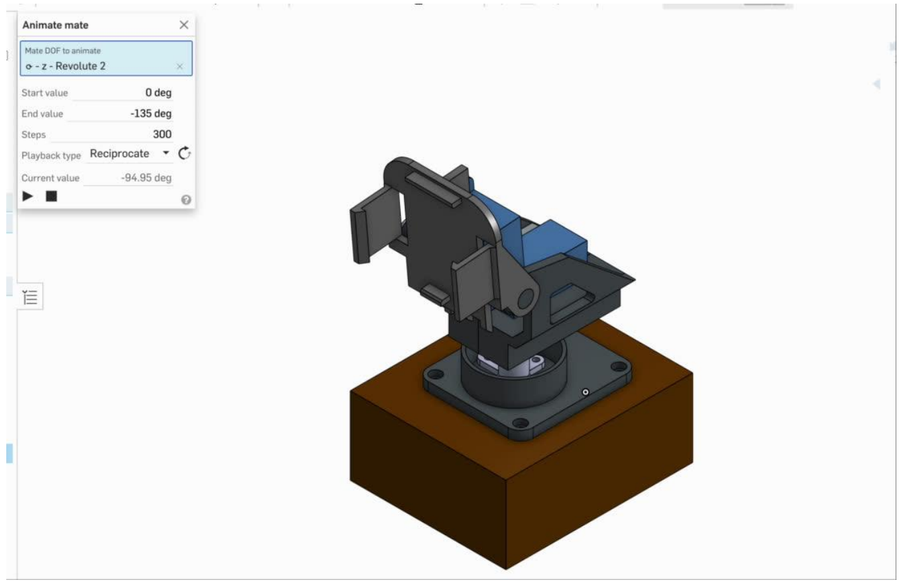
Hardware Assembly

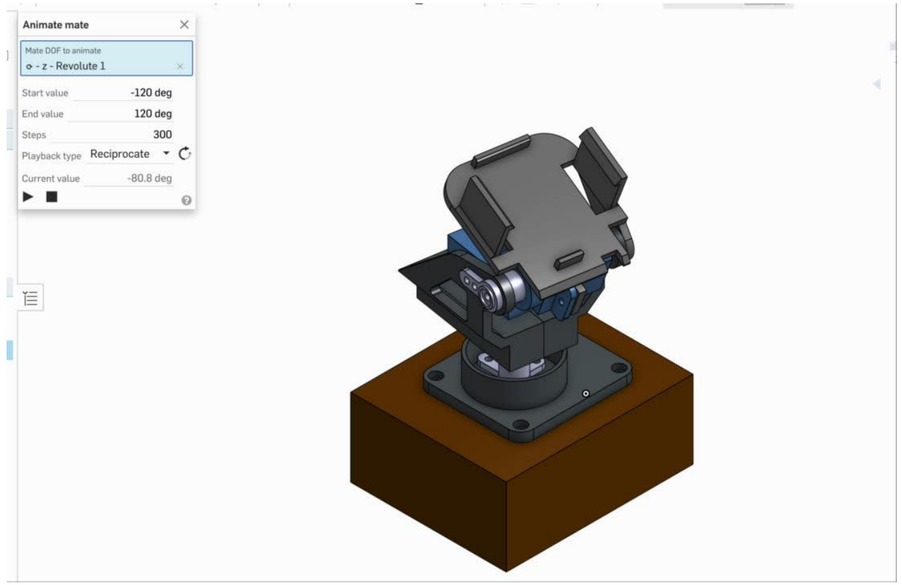


- **Bi-Axial System Design**: Created with two servo motors for angular positioning, controlled by an Arduino Uno.
- **Communication Protocol**: Uses I2C protocol, facilitated by the [insert specific I2C library name] library, for servo-Arduino interaction.
- **Servo Selection**: Chosen based on torque for stable camera movement and speed for rapid adjustments.
- **Camera Integration**: Mounted on the bi-axial system to enable real-time video streaming for face detection.

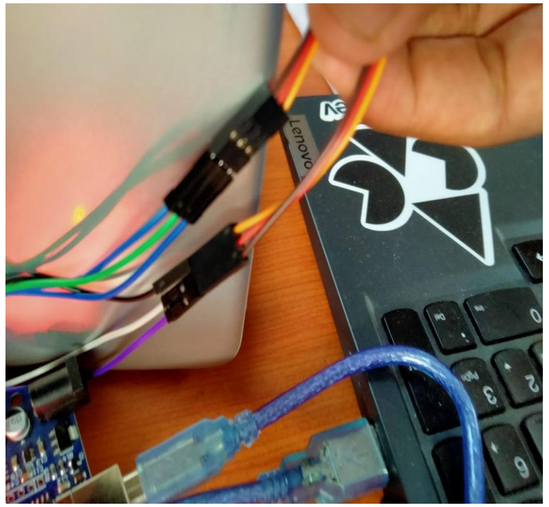


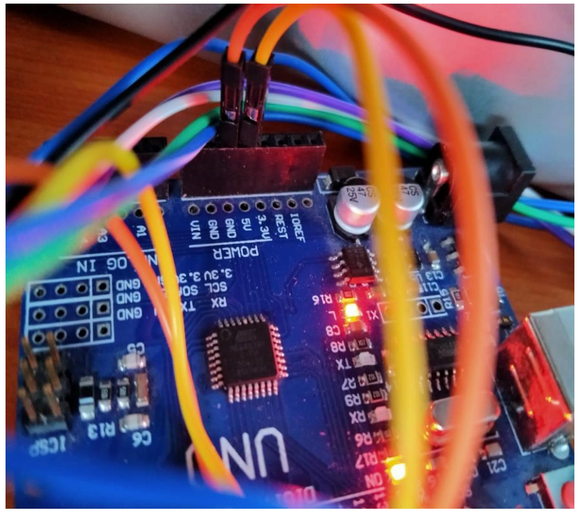
Rotation about x and y axis

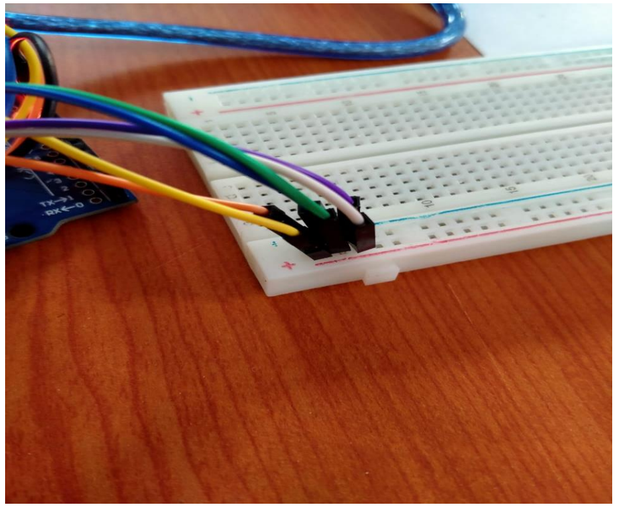


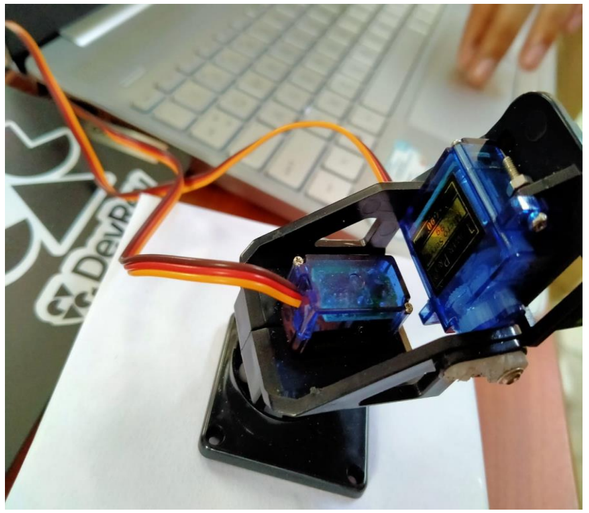


Circuit Design And Assembling









Machine Learning Pipeline

Face Detection:

- **Model Evaluation**: Assessed multiple face detection models including OpenCV, SSD, D lib, MTCNN, Faster MTCNN, Retina Face, Media Pipe, YOLOv8 Face, and Yu Net.
- **Criteria**: Models were evaluated based on their accuracy, speed, and robustness to lighting variations.
- **Comparative Analysis**: Focus on identifying the optimal model for reliable performance in varying environmental conditions.

Face Recognition:

- **Deep Face Integration**: Utilized the Deep Face library for facial recognition tasks.
- **Database Matching**: Implemented to compare and identify faces against a predefined database of known individuals.
- **Purpose**: Enhances security by verifying identities and detecting unauthorized access.

Integration Algorithm

- **Receive Detection Data**: Captures real-time outputs of face locations within the camera frame.
- **Query Recognition Model**: Checks identified faces against the face recognition model to ascertain if the individual is known.
- **Intruder Handling**:
  - Calculates on-screen coordinates if the face is unrecognized.
  - Interpolates these coordinates to real-world angular degrees.
- **Position Transmission**: Sends angular positions to Arduino Uno for processing.
- **Servo Adjustment**: Servo motors adjust the camera to track the intruder’s movement.

Evaluation

Metrics Used: During face recognition model selection, we employed various distance metrics to evaluate the model's performance:

- Cosine similarity
- Euclidean distance
- Euclidean distance with L2 norm (calculates the square root of the sum of squared differences between corresponding elements)

Working Model DEMO

Future Aspects

This project lays a strong foundation for a real-time intrusion detection system. Here are some exciting future aspects to consider:

Object Detection Integration

Enhanced Tracking Algorithms

- Kalman Filtering
- Optical Flow Analysis

Advanced Alerting Systems:

- Zone-based Intrusion Detection
- Audio Integration
- Alert Escalation

.

Conclusion

- **ML and Automation**: Enhances security through face detection, recognition, and robotic cameras for quicker response and improved awareness.
- **Future Development**: Sets the stage for integrating object detection, enhanced tracking, and AI-powered threat assessment.
- **Scalable and Comprehensive**: Designed to evolve into a scalable, intelligent security solution for high-security zones.

Thank You