

|  |
| --- |
| **RAJALAKSHMI INSTITUTE OF TECHNOLOGY** |
| (An Autonomous Institution, Affiliated to Anna University, Chennai) |

**DEPARTMENT OF CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)**

**ACADEMIC YEAR 2025 - 2026**

**SEMESTER III**

**ARTIFICIAL INTELLIGENCE LABORATORY**

**MINI PROJECT REPORT**

|  |  |
| --- | --- |
| **REGISTER NUMBER** | 2117240030145 |
| **NAME** | SRI HARISH A |
| **PROJECT TITLE** | AI-Powered Air Writing Recognition System |
| **DATE OF SUBMISSION** |  |
| **FACULTY IN-CHARGE** | **MRS M BHAVANI** |

**Signature of Faculty In-charge**

**INTRODUCTION**

Artificial Intelligence (AI) and Computer Vision have enabled machines to interpret human gestures and movements in real time. One fascinating application of this technology is **Air Writing Recognition**, where a person can write characters in the air, and the system detects and interprets them using a webcam.

This project uses **MediaPipe** and **OpenCV** libraries in Python to track hand movements and draw letters in the air. It also integrates a **Convolutional Neural Network (CNN)** model to recognize the characters written. The project demonstrates how AI, image processing, and machine learning can be combined to enable human-computer interaction without physical contact.

**PROBLEM STATEMENT**

To develop a real-time system that detects hand motion using a webcam and recognizes characters written in the air. The system should accurately track the fingertip, visualize the writing path, and predict the letter drawn using a trained deep learning model.

**THEORETICAL BACKGROUND**

Air writing recognition combines **computer vision** and **machine learning**.

* **Computer Vision:** Uses OpenCV and MediaPipe to detect and track hand landmarks, especially the index fingertip.
* **Machine Learning (CNN):** The CNN model learns to classify the letter drawn by analyzing pixel patterns from the captured air-written images.

**MediaPipe** provides accurate hand tracking by identifying 21 key landmarks on the hand. **OpenCV** helps visualize and record the movement path. The recorded images are used to train a **CNN classifier** that predicts which alphabet letter has been drawn.

This integration demonstrates an application of **gesture recognition and pattern classification**, which are core components of Artificial Intelligence systems.

**ALGORITHM EXPLANATION WITH EXAMPLE**

**Algorithm Steps:**

1. Initialize webcam and set up MediaPipe Hands for hand tracking.
2. Detect the index fingertip (landmark ID 8).
3. Record fingertip movement coordinates and draw a continuous line (writing path).
4. Save the drawn image when the user presses a key (for dataset creation).
5. Train a CNN model using collected images of letters (A–Z).
6. In real-time, capture the user’s drawn letter, preprocess it, and predict using the trained model.
7. Display the predicted letter on screen.

**Example:**

User writes “A” in the air.  
→ System tracks the index finger path.  
→ The drawn pattern is processed by the CNN model.  
→ Output on screen: **Predicted: A**

**IMPLEMENTATION AND CODE**

import cv2

import numpy as np

import mediapipe as mp

mp\_drawing = mp.solutions.drawing\_utils

mp\_hands = mp.solutions.hands

canvas = np.zeros((480, 640, 3), dtype=np.uint8)

cap = cv2.VideoCapture(0)

mp\_hands.Hands(max\_num\_hands=1, min\_detection\_confidence=0.7) as hands:

    prev\_x, prev\_y = 0, 0

while cap.isOpened():

        ret, frame = cap.read()

        if not ret:

            break

       frame = cv2.flip(frame, 1)

        h, w, c = frame.shape

        rgb = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)

        result = hands.process(rgb)

        if result.multi\_hand\_landmarks:

            for hand\_landmarks in result.multi\_hand\_landmarks:

                mp\_drawing.draw\_landmarks(frame, hand\_landmarks, mp\_hands.HAND\_CONNECTIONS)

                x = int(hand\_landmarks.landmark[8].x \* w)

                y = int(hand\_landmarks.landmark[8].y \* h)

                if prev\_x == 0 and prev\_y == 0:

                    prev\_x, prev\_y = x, y

                cv2.line(canvas, (prev\_x, prev\_y), (x, y), (0, 0, 255), 5)

                prev\_x, prev\_y = x, y

        else:

            prev\_x, prev\_y = 0, 0

        combo = cv2.addWeighted(frame, 0.5, canvas, 0.5, 0)

        cv2.imshow("Air Writing", combo)

        key = cv2.waitKey(1)

        if key == ord('c'):

            canvas = np.zeros((480, 640, 3), dtype=np.uint8)

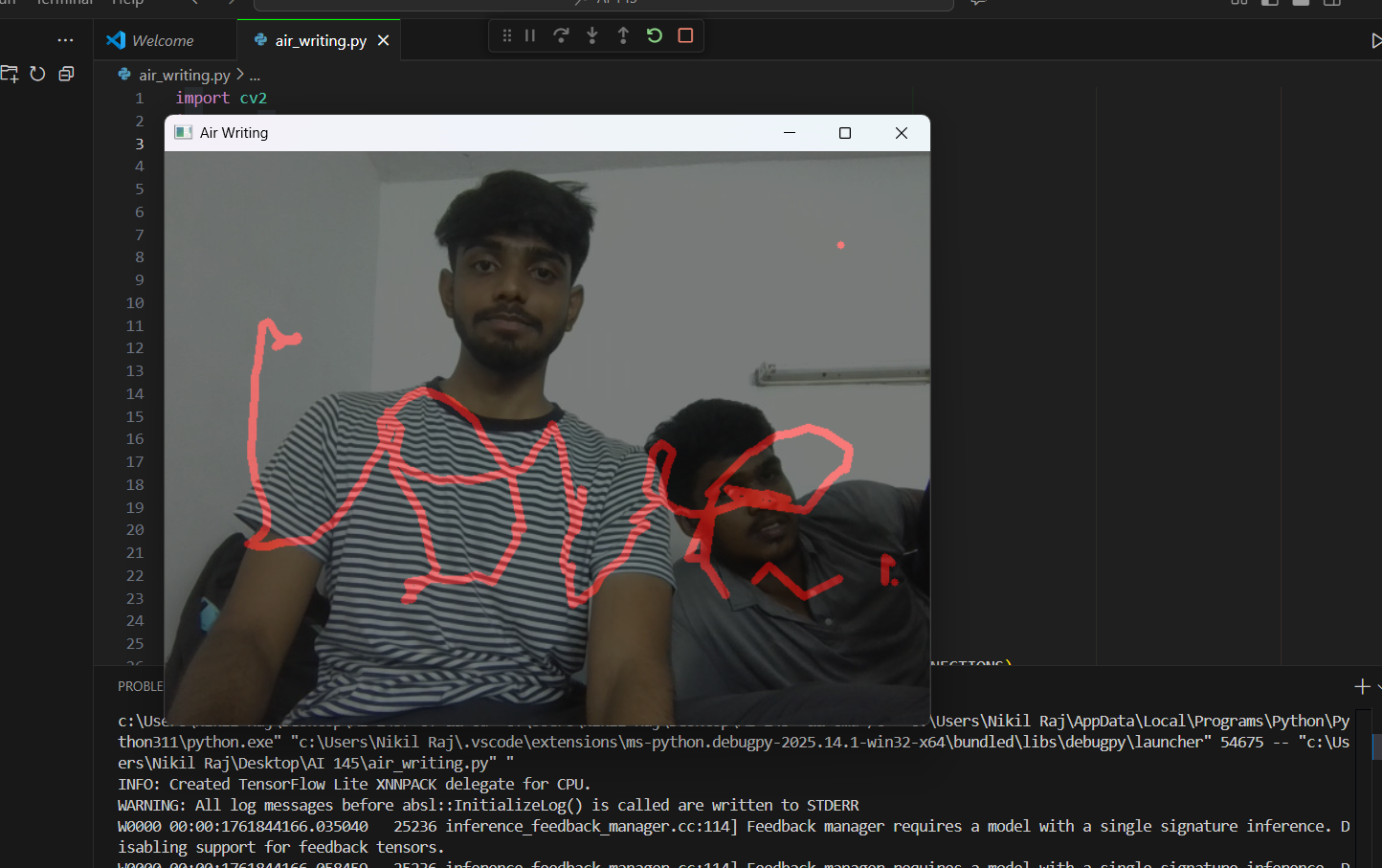
        elif key == ord('q'):

            break

cap.release()

cv2.destroyAllWindows()

**OUTPUT**

****

**RESULTS AND FUTURE ENHANCEMENT**

The Air Writing Recognition System successfully tracks hand movement and recognizes letters drawn in the air. The system uses MediaPipe for accurate hand detection and a CNN model for prediction.

**Future Enhancements:**

* **Recognizing words instead of single letters.**
* **Improving accuracy using Recurrent Neural Networks (RNNs).**
* **Adding gesture-based control for switching modes or clearing the screen.**
* **Developing a mobile or AR-based version for enhanced interactivity.**

|  |  |
| --- | --- |
| **Git Hub Link of the project and report** | https://github.com/sriharisha2006-hash/AI-MINI-PROJECT-2117240030145.git |

**REFERENCES**

* MediaPipe Documentation (2024) – Google Developers.
* OpenCV-Python Tutorials (2024).
* TensorFlow Documentation (2024) – Image Classification Guide.
* GeeksforGeeks (2025) – Hand Tracking using MediaPipe.
* Chat gpt Ai resources