



SUPERIOR UNIVERSITY

Name: Ahmad Omer

Class: BS-AI

Section: 4-A

Roll No: BSAIM-F23-021

Task: 06

Project Report: Real-Time Webcam Drawing Using Color Detection

1. Introduction

This project implements a **real-time color detection and drawing application** using **OpenCV and Flask**. It captures video from the webcam, detects specific colored objects, and allows users to draw on a virtual canvas by moving the detected objects. The processed video feed is then displayed on a web interface using Flask.

2. Objectives

- Implement a system to detect objects based on color.
- Track the movement of detected objects and visualize drawing.
- Stream the processed video feed to a web-based frontend using Flask.
- Provide a user-friendly interface for interaction.

3. Technologies Used

- **Python** – Programming language.
- **OpenCV** – Computer vision library for color detection and image processing.
- **Flask** – Lightweight web framework for serving video streams.
- **HTML, CSS** – Frontend technologies for the web interface.
- **NumPy** – Used for handling image array operations.

4. Methodology

4.1 Color Detection

- The system captures frames from the webcam.
- Each frame is converted to the **HSV color space** to make color detection more robust.
- Predefined HSV color ranges are used to detect specific colors.
- **Contours are extracted** from the detected regions to find object positions.

4.2 Virtual Drawing Mechanism

- Once a colored object is detected, its **center coordinates are recorded**.
- These points are stored and drawn onto the video frame using **circles**.
- Multiple color categories allow users to draw with different colors.

4.3 Flask Integration for Web Streaming

- The **Flask server** captures and processes video frames.
- The processed video (with detected objects and drawings) is **encoded and streamed** as an MJPEG feed.
- The web page fetches this feed dynamically and displays it in real-time.

5. Results

- The application successfully detects objects of specific colors.
- Users can move a colored object in front of the webcam to draw on the screen.
- The processed video is displayed live on the web interface with minimal lag.
- The project effectively integrates OpenCV for real-time vision processing and Flask for web-based visualization.

6. Future Enhancements

- **Add a Clear Button** – Allow users to reset the drawing.
- **Gesture Controls** – Use hand tracking to start and stop drawing.
- **Multi-User Drawing** – Support multiple webcam feeds for collaborative drawing.
- **Save Drawings** – Provide an option to save the drawn image as a file.

7. Conclusion

This project demonstrates an effective way to combine **computer vision and web technologies** for an interactive user experience. The system enables real-time drawing using colored objects, making it suitable for educational tools, fun applications, and interactive interfaces.

This report provides a concise summary of the project, its working methodology, and potential improvements. Let me know if you need any modifications! 🚀