TOUCHLESS HCL FOR MEDIA CONTROL USING HAND GESTURES ON NVIDIA JETSON NANO

SASTRA DEEMED UNIVERSITY

GUIDANCE UNDER: DR. SIVARAMAN

BHARANITHARAN M -127180011

GOWTAAM M-127180023

HARIKRISHNA R-127004088

1. Abstract

This project presents the design and implementation of a real-time gesture detection system using a Raspberry Pi platform. The system captures hand gestures through a camera module and processes them using computer vision and machine learning techniques. The primary goal is to enable touchless human-computer interaction for applications such as home automation, assistive technology, and smart interfaces. The project demonstrates a cost-effective, portable, and optimized embedded solution capable of recognizing predefined gestures with acceptable accuracy and low latency.

2. Introduction

Gesture recognition has gained significant attention in recent years due to the increasing demand for natural and contactless human-computer interaction. Traditional input methods like keyboards and touchscreens may not always be suitable, especially in hygiene-sensitive environments.

This project focuses on implementing a lightweight gesture detection system using Raspberry Pi, which offers a compact and affordable embedded computing platform. The system integrates image acquisition, preprocessing, gesture classification, and real-time output visualization.

3. Objectives

- Develop a real-time gesture recognition system using Raspberry Pi.
- Implement computer vision techniques for hand detection and tracking.
- Apply machine learning for gesture classification.
- Optimize the system for low-power embedded hardware.
- Demonstrate practical applications such as gesture-based control.

4. System Architecture

The system consists of the following modules:

- **Input Module** – Captures real-time video using a Raspberry Pi camera.
- **Preprocessing Module** – Performs image enhancement and hand segmentation.
- **Feature Extraction** – Identifies key points or contours from the hand region.
- **Classification Module** – Recognizes gestures using a trained model.
- **Output Module** – Displays results and triggers mapped actions.

5. Hardware Utilization

Components Used

- Raspberry Pi (Model 4)
- Raspberry Pi Camera Module / USB Webcam
- MicroSD Card (32GB)
- Power Supply
- HDMI Monitor
- Keyboard and Mouse (initial setup)

Hardware Role

- **Raspberry Pi:** Core processing unit handling image acquisition and inference.
- **Camera Module:** Captures gesture data.
- **MicroSD Card:** Stores OS, datasets, and trained models.

6. Software Requirements

- Raspberry Pi OS (Linux-based)
- Python Programming Language
- OpenCV
- NumPy
- Media pipe

Gesture Recognition Logic

The gesture recognition in this project is implemented using a rule-based approach based on hand landmarks detected by Media Pipe. The Media Pipe Hands model provides 21 landmarks for each detected hand. By analyzing the relative positions of finger

tip landmarks with respect to their lower joints, the system determines whether a finger is extended or folded.

Each finger is evaluated individually, and the total number of extended fingers is used to map gestures to system actions.

Finger Detection Method

Each finger is classified as **extended** or **folded** using landmark comparison:

- If fingertip y-coordinate < lower joint y-coordinate → Finger is extended
- Else → Finger is folded

For the thumb, horizontal comparison (x-axis) is used due to its orientation.

The system works by detecting the number of fingers shown in front of the camera and assigning an action based on the finger count. After the hand is detected, the landmarks of each finger are analyzed to determine whether the finger is open or closed. The total number of open fingers is then calculated. If one finger is detected, the system performs a forward or next action. If two fingers are detected, it performs a backward or previous action.

When three fingers are shown, the system increases the volume. When four fingers are detected, the system decreases the volume. This process runs continuously in real-time, allowing the user to control actions using simple hand gestures.

Applications

- Touchless home automation
- Gesture-based media control
- Smart classroom systems
- Assistive technology
- Robotics interface

Results

The implemented gesture recognition system was successfully tested on a Raspberry Pi using a live camera feed. The system was able to detect hand gestures in real-time and correctly interpret finger-based commands such as forward, backward, volume up, and volume down. The MediaPipe hand tracking framework provided stable landmark detection, enabling accurate identification of extended fingers without requiring a custom trained model.

During testing, the system achieved smooth performance with an average frame rate of approximately 15–25 frames per second under normal lighting conditions. The gesture detection accuracy was observed to be around 90–95% for static hand gestures when the hand was clearly visible within the camera frame. The response time was low, allowing near real-time execution of actions based on detected gestures.

It was observed that the system performed best in well-lit environments with minimal background clutter. In low-light conditions or when the hand was too far from the camera, detection accuracy slightly decreased. Despite these limitations, the system demonstrated reliable and consistent performance for basic gesture-based control, proving the feasibility of implementing real-time gesture recognition on low-cost embedded hardware like Raspberry Pi.