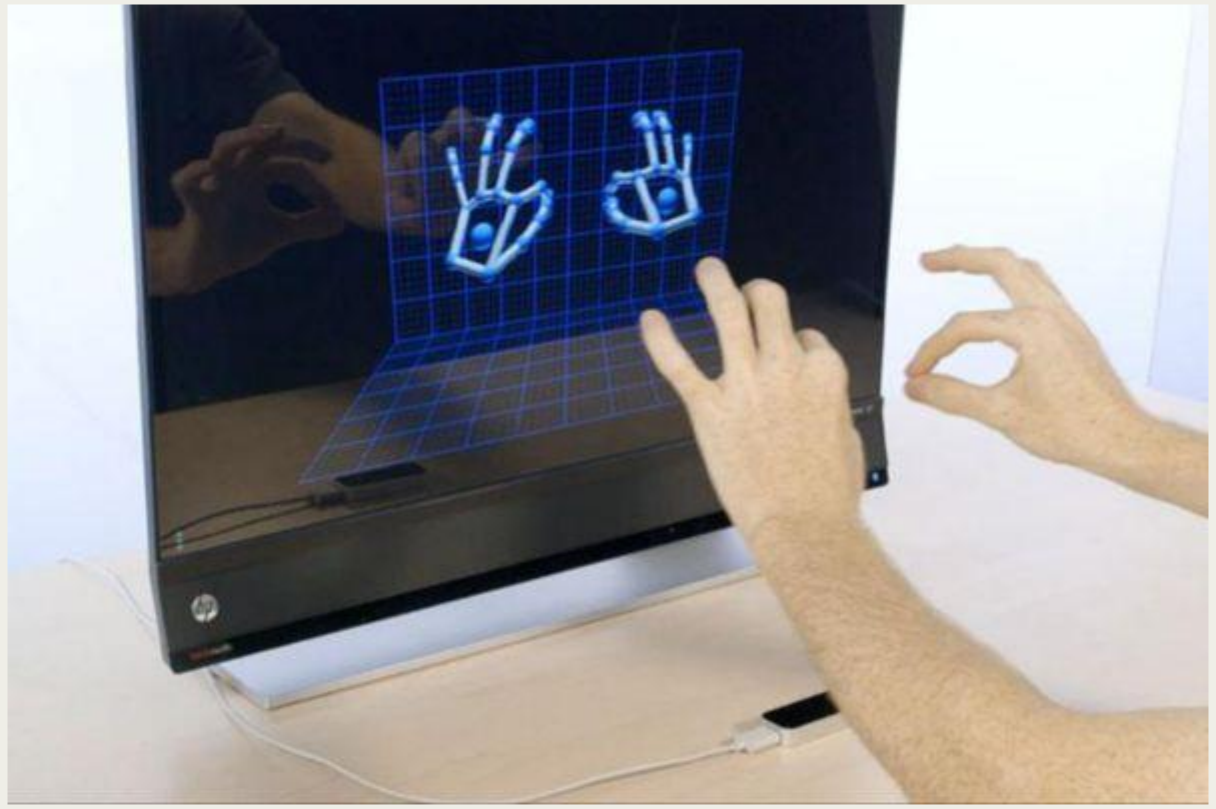
GESTURE VOLUME : VOLUME CONTROL WITH HAND GESTURES

MENTOR : DR.D.BHANU PRAKASH

PRESENTED BY : ANUSUYA B, PRASHANTHI HEBBAR , SHREYAS S N, SIVA SHANKAR SEETHARAMAIAH

INTRODUCTION

- Human–computer interaction is evolving
- Touchless systems are gaining importance
- Hand gestures provide a natural way to interact
- Advances in computer vision make real-time gesture recognition possible



PROBLEM STATEMENT

- Physical controls interrupt workflow
- Buttons can be inconvenient or wear out
- Not suitable for hands-free use
- Hard to manage during presentations
- Need fast, touch-free volume control

OBJECTIVE

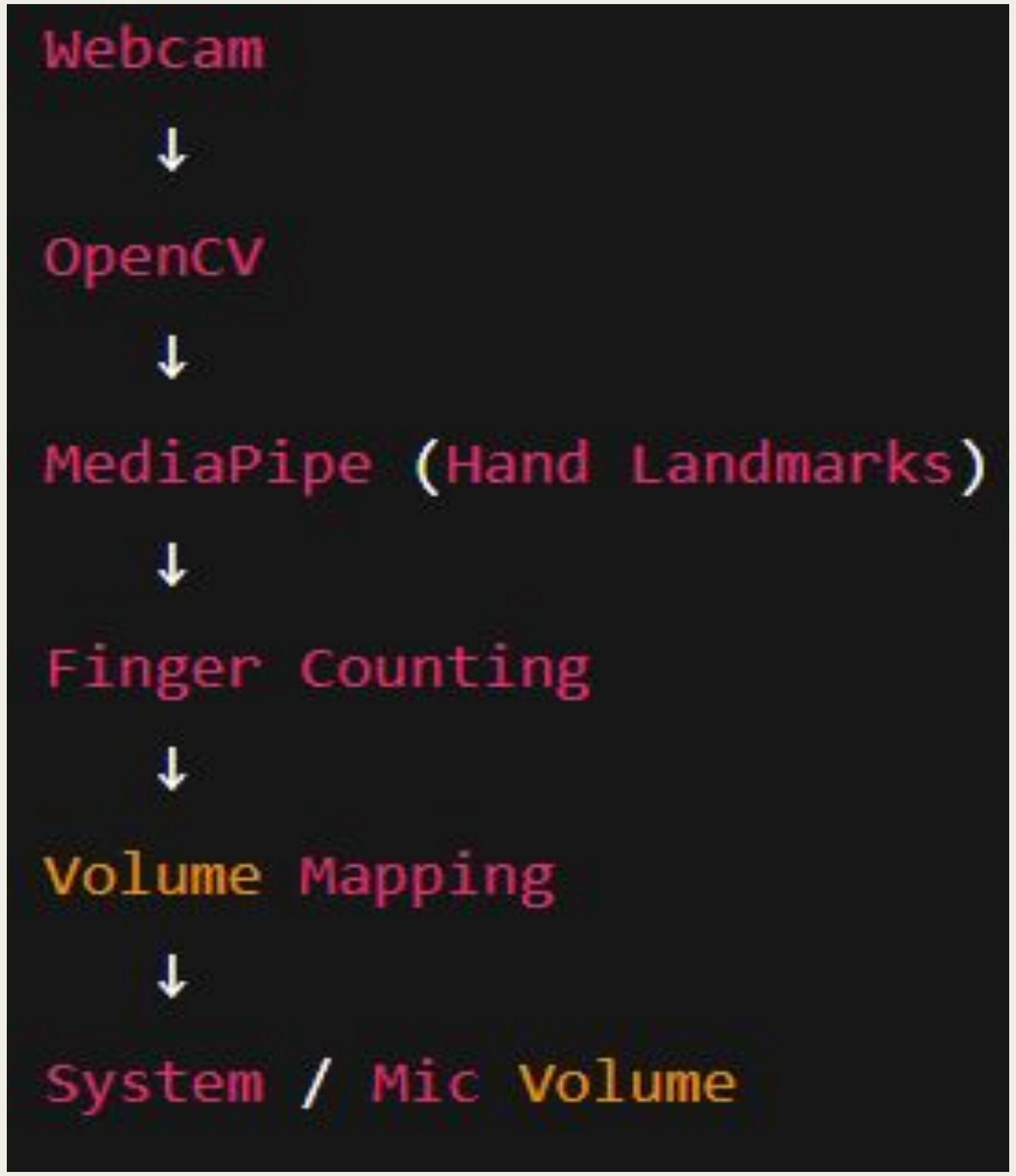
- Detect real-time hand gestures using a webcam
- Accurately count fingers
- Control system and mic volume dynamically
- Provide a touch-free, user-friendly experience

PROPOSED SOLUTION

- Use a standard webcam to capture hand gestures
- Apply computer vision techniques for gesture detection
- Detect and track hand landmarks in real time
- Count fingers accurately using landmark positions
- Map finger count to predefined volume levels
- Control system and microphone volume dynamically

SYSTEM OVERVIEW

- Webcam captures live gestures
- Frames processed with computer vision
- Real-time hand landmark detection
- Finger count from landmark positions
- Continuous real-time interaction



```
Webcam
   ↓
OpenCV
   ↓
MediaPipe (Hand Landmarks)
   ↓
Finger Counting
   ↓
Volume Mapping
   ↓
System / Mic Volume
```

TECHNOLOGICAL STACK OVERVIEW

SOFTWARE	PURPOSE
Windows OS	Supports Pycaw audio control
Python	Core programming language
OpenCV	Webcam access & image processing
MediaPipe	Real-time hand landmark detection & finger tracking
Pycaw	System & microphone volume control
Streamlit	Real-time dashboard / user interface
NumPy, Pandas,Comtypes	Supporting libraries for data handling & stability

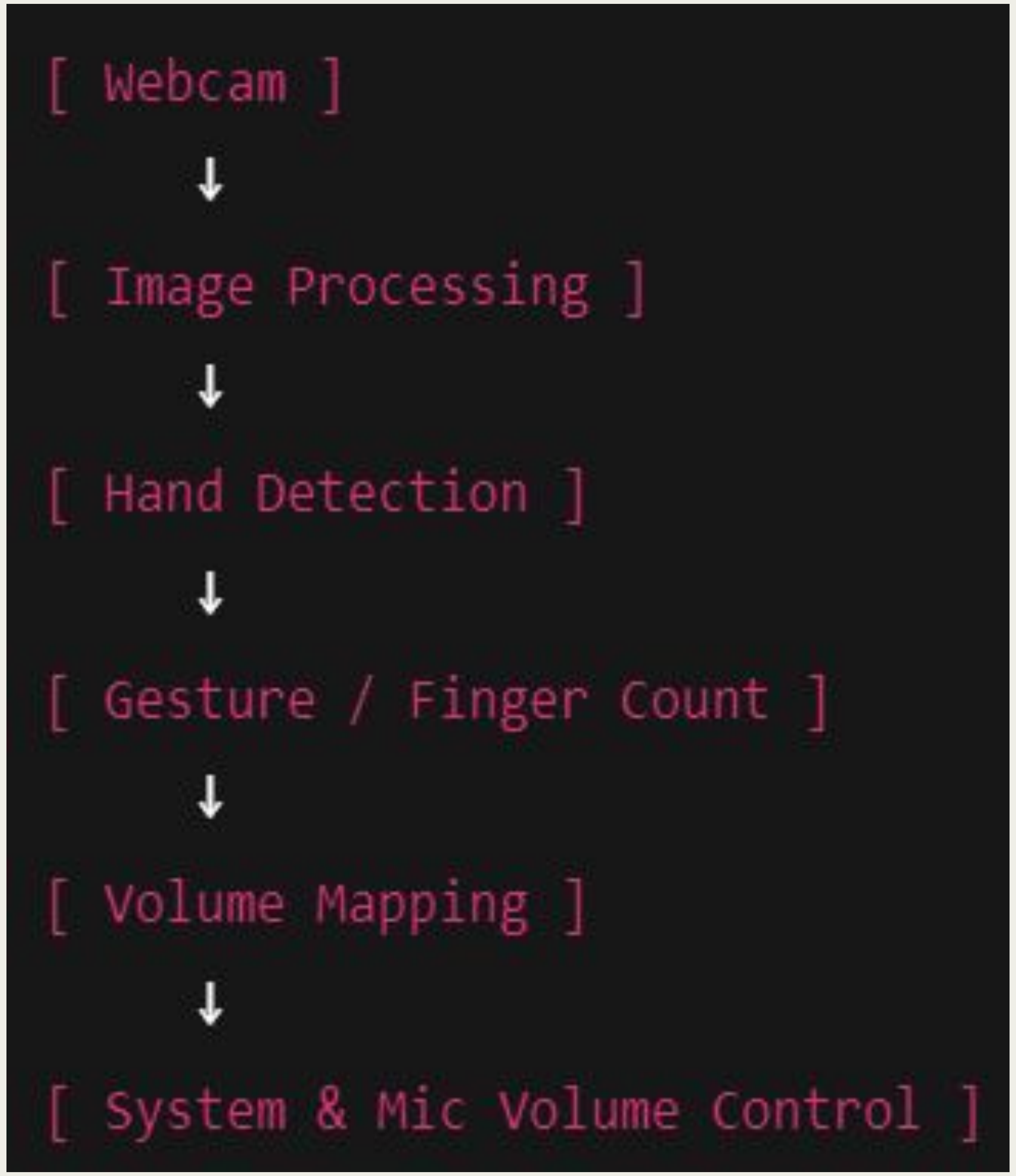
BACKEND TECH STACK

Windows OS & Python	Core platform and programming language
OpenCV	Webcam access & real-time image processing
MediaPipe Hands	Hand landmark detection & finger tracking
Pycaw Library	Controls system and microphone volume via Windows Audio API
COM Objects Integration	Interfaces with IAudioEndpointVolume for audio control
Real-Time Audio Control	Adjusts system volume instantly using gestures
Microphone Control	Supports volume adjustment, mute/unmute with smooth updates
Supporting Libraries	NumPy, Pandas, Comtypes, Streamlit for stability & UI

WORKING FLOW

SYSTEM PROCESSING FLOW

- Webcam captures live frames
- Frames processed using computer vision
- Hand detected & landmarks extracted
- Fingers counted & gesture identified
- Gesture converted into command



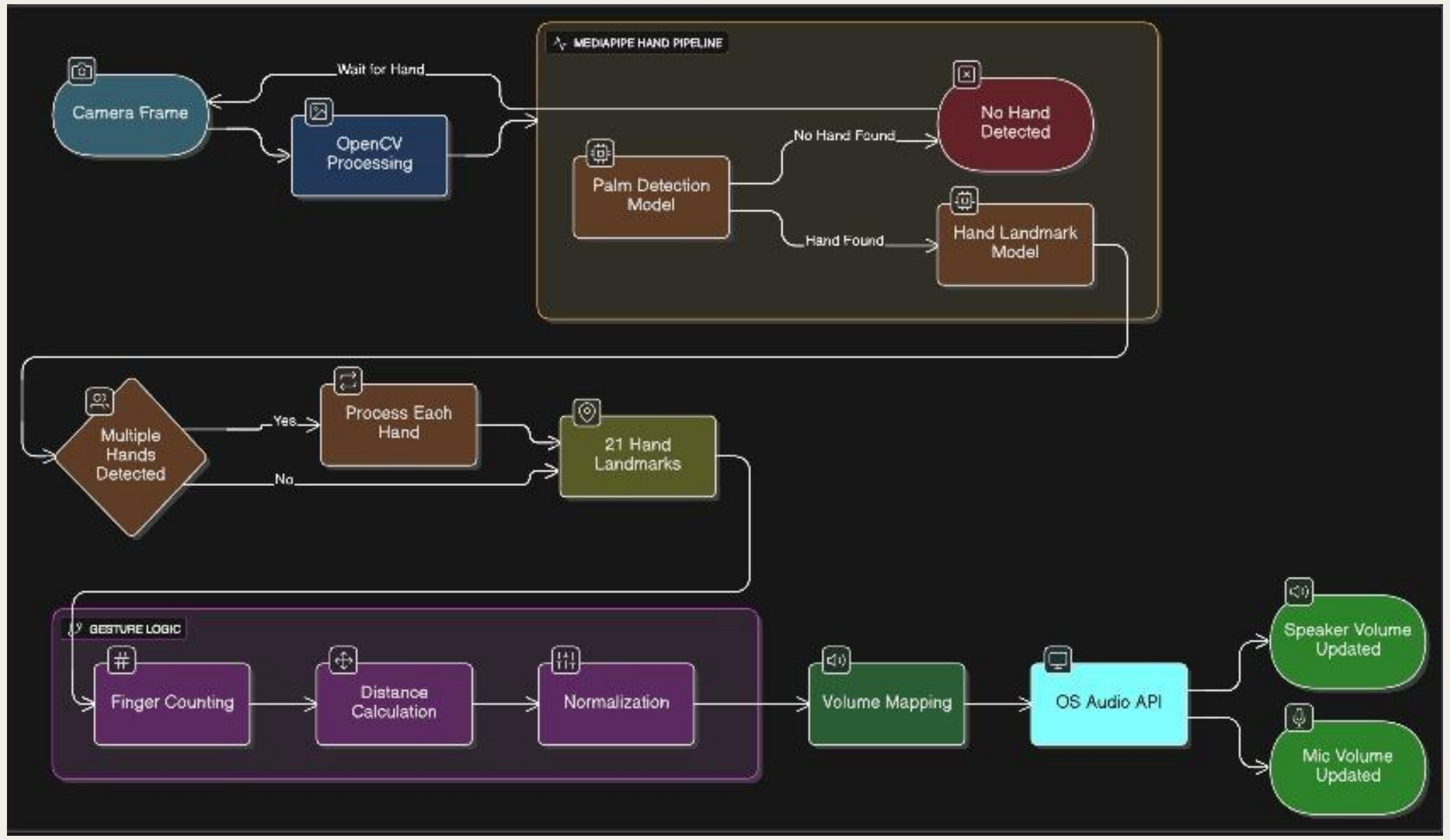
```
[ Webcam ]
    ↓
[ Image Processing ]
    ↓
[ Hand Detection ]
    ↓
[ Gesture / Finger Count ]
    ↓
[ Volume Mapping ]
    ↓
[ System & Mic Volume Control ]
```

VOLUME FLOW

- Finger count → volume value
- Value normalized to system range
- PyCaw applies volume change
- System & Microphone volume updated
- Changes reflected instantly

MEDIAPIPE HAND DETECTION

- Uses **MediaPipe Hands** to detect hand in each video frame
- **Palm Detection Model** first locates hand region for faster processing
- **Hand Landmark Model** then tracks **21 precise hand keypoints**
- Works **in real time on CPU** without external hardware
- Robust to **movement, rotation, and scale changes**
- Extracted landmarks are used for **finger counting & gesture recognition**



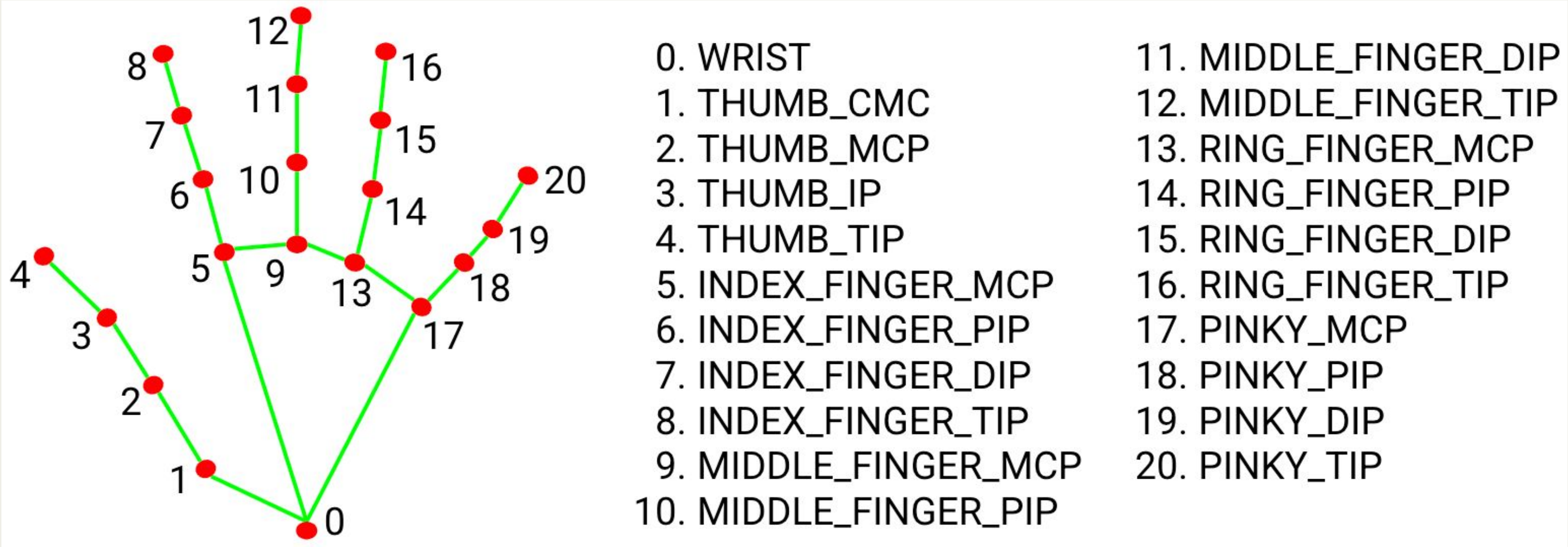
21 HAND LANDMARKS EXPLANATION

- MediaPipe detects **21 fixed hand landmarks**
- Each point represents a **specific joint or fingertip**
- Landmarks are labeled **0 to 20** for identification
- Each landmark has **(x, y, z) coordinates**

  X → Horizontal position

  Y → Vertical position

  Z → Depth / distance from camera
- These landmarks are used for **finger counting, gesture detection, and movement tracking**



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

GESTURE LOGIC

- KEY LANDMARKS USED

  Thumb Tip → **Landmark ID 4**
  Index Finger Tip → **Landmark ID 8**

- DISTANCE CALCULATION

  Distance measured between ID 4 and ID 8
  Uses **Euclidean distance formula**
  Distance value changes as fingers move

- GESTURE INTERPRETATION

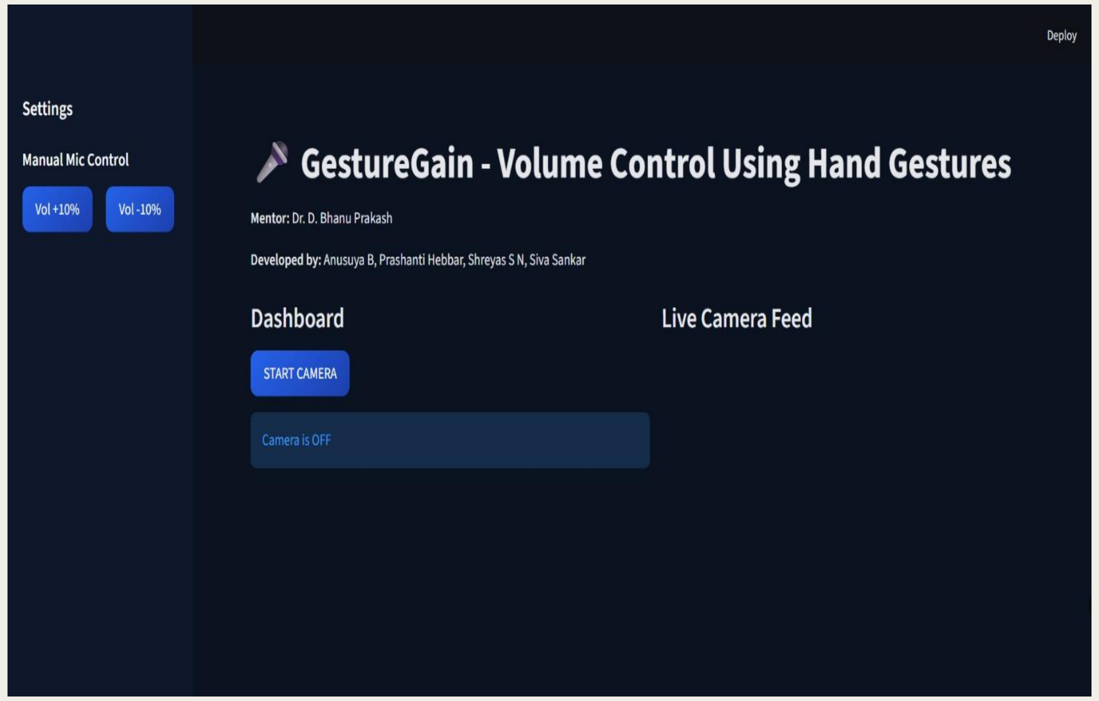
  Small distance → **Low Volume**
  Large distance → **High Volume**

- MAPPING TO VOLUME

  Distance is **normalized** to a fixed range
  Mapped to **System & Microphone volume**
  Volume updates occur **smoothly in real time**

FRONTEND UI

- Displays **live webcam feed** to the user
- Shows **hand landmarks** and detected gestures
- **Current volume level** shown visually
- Provides **real-time feedback** with smooth updates
- Start / Stop control for webcam
- Shows **finger count** and **mute / unmute status**
- Clean, responsive, and easy to use interface



GestureGain - Volume Control Using Hand Gestures

Mentor: Dr. D. Bhanu Prakash

Developed by: Anusuya B, Prashanti Hebbar, Shreyas S N, Siva Sankar

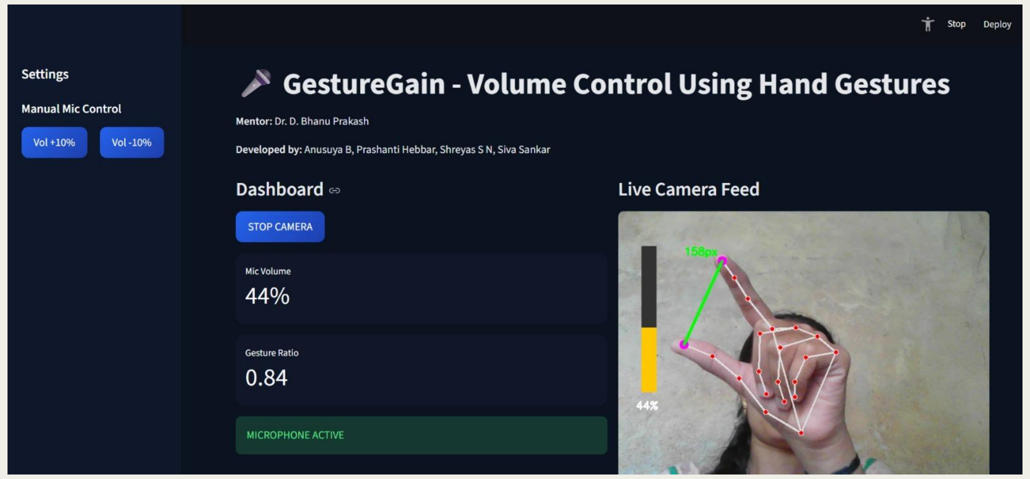
Dashboard

Live Camera Feed

Camera is OFF

WEB DASHBOARD

- Web-based dashboard built using **HTML, CSS, and JavaScript**
- Displays **live webcam feed** for gesture interaction
- Shows **real-time volume indicator** visually
- Updates dynamically based on detected gestures
- Works in browser, providing **simple and user-friendly control**
- Ensures smooth and responsive user experience



GestureGain - Volume Control Using Hand Gestures

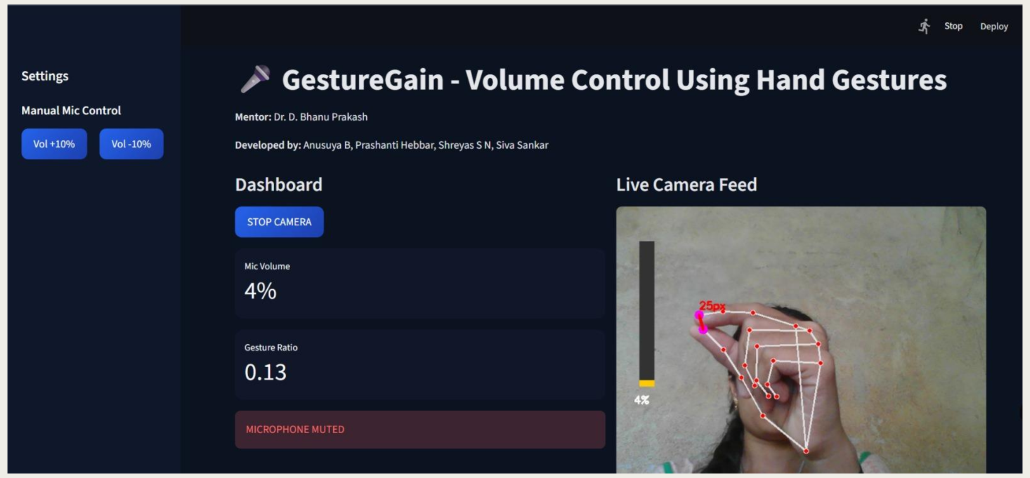
Mentor: Dr. D. Bhanu Prakash

Developed by: Anusuya B, Prashanti Hebbar, Shreyas S N, Siva Sankar

Mic Volume
44%

Gesture Ratio
0.84

MICROPHONE ACTIVE



GestureGain - Volume Control Using Hand Gestures

Mentor: Dr. D. Bhanu Prakash

Developed by: Anusuya B, Prashanti Hebbar, Shreyas S N, Siva Sankar

Mic Volume
4%

Gesture Ratio
0.13

MICROPHONE MUTED

FINAL OUTPUT

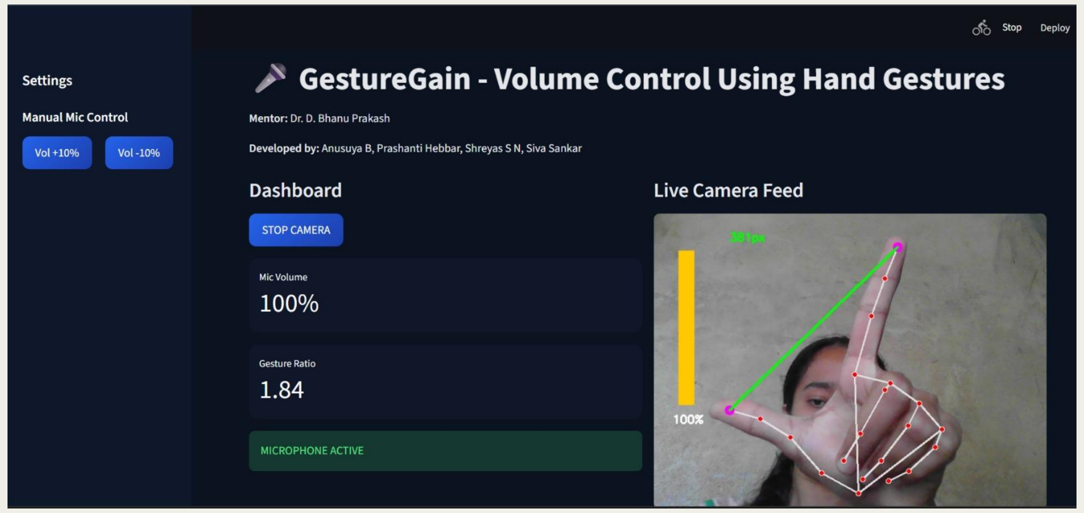
- System successfully detects hand gestures in real time

- Volume level changes smoothly based on gesture distance / finger logic

- System and microphone volume update instantly

- Mute / unmute and volume adjustments work correctly

- Output remains stable and consistent during continuous use



GestureGain - Volume Control Using Hand Gestures

Mentor: Dr. D. Bhanu Prakash

Developed by: Anusuya B, Prashanti Hebbar, Shreyas S N, Siva Sankar

Dashboard

Mic Volume
100%

Gesture Ratio
1.84

MICROPHONE ACTIVE

Live Camera Feed

THANKYOU