LOW COST EYE GESTURE COMMUNICATION SYSTEM FOR PEOPLE WITH MOTOR DISABILITIES

A PROJECT REPORT

Submitted by

DEENADAYALAN R	**(611821104006)**
DINESHKUMAR V	**(611821104008)**
SURIYA A	**(611821104051)**

In partial fulfillment for the award of the degree

of

BACHELORE OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

P.S.V. COLLEGE OF ENGINEERING AND TECHNOLOGY

(Approved by AICTE, New Delhi and Affilicated to Anna University,chennai)

(Accredited by the NAAC with ‘A’ Grade)

(An ISO 9001:2015 Certified Institution)

KRISHNAGIRI

ANNA UNIVERSITY : CHENNAI 600025

MAY 2025

ANNA UNIVERSITY : CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this Project report **“LOW COST EYE GESTURTE COMMUNICATION SYSTEM FOE PEOLE WITH MOTOR DISABILITIES”** is the bonafide work of “**DEENADAYALAN R (611821104006), DINESHKUMAR V (611821104008), SURIYA A (611821104051)”** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Mr.C.PRAKASH NARAYANAN,M.E.,	**Mrs.R.MONISHA,M.E.,**
HOD and PROFESSOR	**ASSISTANT PROFESSOR**
Department of Computer Science	Department of Computer Science
And Engineering.	And Engineering.
P.S.V College of Engineering	P.S.V College of Engineering
and Technology,	and Technology,
Krishnagiri-635108	Krishnagiri-635108

Submitted for the University Viva-Voce Project examination held on
…....................... at P.S.V College of Engineering and Technology, Krishnagiri.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

At the very outset, we wish to express our sincere thanks to all those who were involved in the completion of this project.

Our most sincere salutations go to Anna University, Chennai that gave us an opportunity to have sound base of Computer Science and Engineering.

We acknowledge our profound intellect of gratitude to our beloved chairman **Dr.P.SELVAM,M.A.,B.Ed.,M.Phil.,Ph.D.,** for his support and coordination.

We also express our gratitude to our honourable principal **Dr.P.LAWRENCE,M.E.,Ph.D.,** for his timely advice and his encouragement.

We offer our sincere thanks to **Mr.C.PRAKASH NARAYANAN,M.E.,**Head of the Department of Computer Science and Engineering for giving this prospect and his full support.

We consider it as a great privilege to place a record of our deep sense of gratitude to Internal Guide **Mrs.R.MONISHA,M.E.,**Assistant Professor, Computer Science and Engineering for being the great inspiration to us.

We also express our thanks to our Department Faculty members, our parents and our beloved friends.

DEENADAYALAN.R

DINESHKUMAR.V

SURIYA.A

ABSTRACT

Individuals with severe motor disabilities, such as those caused by ALS, cerebral palsy, or spinal cord injuries, often face significant challenges in communication and interaction with their environment. This paper presents a low-cost eye gesture-based communication system designed to enhance accessibility and independence. The system utilizes a webcam or an infrared-based eye-tracking module to detect and interpret eye movements, which are then mapped to predefined commands for speech synthesis, device control, and text input. Machine learning algorithms process real-time eye gestures, ensuring high accuracy and responsiveness. Unlike expensive commercial alternatives, this system leverages affordable hardware and open-source software, making it accessible to a wider population. Potential applications include speech assistance, smart home control, wheelchair navigation, and human-computer interaction. The proposed system significantly improves the quality of life for individuals with motor impairments, enabling seamless communication and interaction with digital and physical environments.

KEYWORDS- Eye gesture recognition, blink detection, computer vision, assistive technology, motor disabilities, Arduino, real-time communication, facial landmark detection, low-cost system.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Communication is a fundamental aspect of human interaction, yet individuals suffering from cerebral palsy, or spinal cord injuries—often lose the ability to speak or use their limbs. Despite these limitations, many retain control over their eye movements, which can serve as a powerful means of communication.

Modern assistive technologies like eye-tracking devices exist, but they are often prohibitively expensive, complex, or require specialized hardware, making them inaccessible to people in low-income or resource-constrained environments. This highlights the need for a more affordable, open-source, and user-friendly solution.

This project aims to develop a low-cost eye gesture communication system that leverages common and inexpensive hardware such as webcams, Arduino microcontrollers, and servo motors. By using computer vision algorithms to detect eye blinks and gaze direction, the system translates these gestures into specific commands. These commands can then be used to activate physical devices, generate voice outputs, or interact with a graphical interface.

The goal is to empower individuals with motor impairments to express themselves, perform basic tasks, and regain a level of independence through a system that is both accessible and customizable.

1.2 OVERVIEW

This project introduces a low-cost communication system designed for individuals with motor disabilities who are unable to speak or move their limbs. The system relies on eye gestures—such as blinks and gaze direction—as input signals to enable communication.

Using a standard webcam, the system captures real-time video of the user’s face. It processes this input using computer vision techniques (OpenCV and Dlib in Python) to detect specific eye movements. These gestures are then interpreted as commands and transmitted to an Arduino microcontroller, which activates a servo motor or other hardware output, such as a buzzer, LED, or even a speech module. The project is modular and designed to be cost-effective, with total expenses kept under ₹5000 (~$60). It supports customization, scalability, and integration with future enhancements like smart home control, predictive text input, or multilingual voice output.

This system offers an accessible alternative to expensive commercial eye-tracking devices and enables people with severe disabilities to communicate independently and effectively.

1.3 OBJECTIVE

The primary objective of this project is to develop a low-cost, eye gesture-based communication system that enables individuals with severe motor disabilities to communicate and interact with their environment using only their eye movements.

Specific goals include:

1. To **detect eye gestures** (such as blinks and gaze direction) using a standard webcam and computer vision techniques.
2. To **interpret these gestures** and convert them into digital commands in real time.
3. To **control external devices** (e.g., servo motors, LEDs, buzzers) using an Arduino microcontroller based on the recognized gestures.
4. To provide an **affordable and accessible solution** that can be used in low-resource settings.

5. To create a **modular system** that can be extended with additional features like text-to-speech output or smart home integration.

1.4 AVAILABLE FEATURES

1. Eye Gesture Detection

- Detects **blinks** and **gaze direction** (left, right, center).
- Uses **OpenCV** and **Dlib** for real-time facial landmark tracking.
- Calculates **Eye Aspect Ratio (EAR)** to identify blinks reliably.

2. Gesture-to-Command Mapping

Eye gestures are mapped to specific commands:

  - **Single Blink** – Select
  - **Double Blink** – Confirm
  - **Look Left/Right** – Navigate between options

3. Arduino-Based Control

- Sends gesture commands from Python to **Arduino UNO** via serial communication.
- Arduino activates a **servo motor**, **LED**, or **buzzer** based on the received input.

4. Physical Output (Servo Motor)

- The servo motor simulates physical actions like pressing a switch or pointing to a message.
- Enables **non-verbal communication or interaction** with physical devices.

5. Optional Text-to-Speech (TTS) Output

- Converts selected commands into **spoken messages** using Python libraries like pyttsx3.

- Supports **voice feedback** for the user or for communication with others.

6. Cost-Effective Design

Entire system built with **readily available, low-cost components**:

- Webcam
- Arduino Uno
- Servo Motor
- Basic electronic components

7. Modular and Expandable

- System is designed to be **easily modified** or extended.
- Can be integrated with a GUI, virtual keyboard, or IoT devices.

1.5 COMPONENTS OF EYE GESTURE SYSTEM

1. Processing Unit

Laptop/PC with Python installed

- Runs the computer vision code.
- Handles real-time eye gesture detection and command processing.

2. Webcam (USB or Built-in)

- Captures live video of the user's face.
- Used as the primary input device for detecting eye gestures.
- Resolution: Minimum 640x480 recommended.

3. Arduino UNO (or compatible microcontroller)

- Receives serial commands from the PC.
- Controls connected hardware like servo motors or LEDs based on received input.
- Acts as the output controller for real-world interactions.

4. Servo Motor (SG90 or equivalent)

- Executes physical actions like rotating a pointer or triggering a button.
- Simulates selection or response actions from the user.

5. USB Cable

- Used to connect the Arduino UNO to the PC.
- Also provides power and serial communication.

6. Jumper Wires and Breadboard (Optional)

- For testing or extending the system with LEDs, buzzers, or relays.
- Used in prototyping the hardware control section.

7. Optional Components

- **Buzzer / LED** – for simple alerts or feedback.
- **Text-to-Speech Speaker** – if audio output is included.
- **Power Supply (Battery Pack)** – for standalone Arduino operation (optional).

8. Software Requirements

- **Python 3.x**
- **OpenCV** – for image processing.
- **Dlib** – for facial landmark detection.

- **pyttsx3** (optional) – for voice output.
- **Arduino IDE** – for uploading code to the Arduino board.

1.6 BENEFITSOFEYE GESTURE SYSTEM

1. Affordable and Low-Cost

Built using inexpensive, easily available components.

Total cost is significantly lower than commercial eye-tracking systems (under ₹5000 or ~$60).

2. Non-Invasive and User-Friendly

No physical contact or wearable devices needed.

Uses a standard webcam, ensuring user comfort and hygiene.

3. Improves Quality of Life

Allows people with severe motor disabilities to express their needs, emotions, or choices independently.

Supports basic interaction with caregivers or automated systems.

4. Customizable and Scalable

System can be tailored to individual users' needs and abilities.

New commands, gestures, or devices can be added easily.

5. Real-Time Performance

Processes eye movements in real-time for immediate feedback and response.

Low latency enhances usability and responsiveness.

6. Compatible with Assistive Technologies

Can be extended to include speech output (TTS), smart home control, or GUI interfaces.

Compatible with existing assistive communication devices and systems.

7. Encourages Independence

Reduces dependency on caretakers for basic communication.

Gives users more control over their environment.

1.7 APPLICATIONSOFEYE GESTURE SYSTEM

1. Assistive Communication for People with Disabilities

- Enables individuals with motor neuron diseases (e.g., ALS), paralysis, or spinal cord injuries to communicate using only their eye movements.
- Allows selection of words, phrases, or actions through simple eye gestures.

2. Human-Computer Interaction (HCI)

- Can be used as a hands-free input method for controlling computers.
- Useful in scenarios where physical input devices (mouse, keyboard) are not practical.

3. Smart Home Control

- Eye gestures can be used to control smart appliances such as lights, fans, or emergency alarms.
- Offers physically challenged individuals greater control over their surroundings.

4. Hospital or Care Home Environments

- Useful in ICU or post-surgical recovery, where patients may be conscious but unable to speak or move.
- Helps medical staff understand patient needs (e.g., pain, thirst, discomfort).

5. Rehabilitation and Therapy

- Can be used as part of cognitive or physical therapy for patients relearning communication or control skills.

- Helps assess motor control and attention in a non-invasive manner.

6. Education for Special Needs

- Assists children or adults with special needs in participating in learning activities.
- Enables interaction with educational tools through gaze or blink-based commands.

1.8 PROBLEM STATEMENT

Individuals with severe motor disabilities—such as those caused by ALS (Amyotrophic Lateral Sclerosis), spinal cord injuries, or advanced neuromuscular disorders—often lose the ability to move their limbs or speak. As a result, they are unable to perform basic communication or interact with their environment, leading to extreme dependence on caregivers and a diminished quality of life.

While advanced eye-tracking technologies exist, they are typically expensive, require specialized hardware, and are not accessible in low-resource or rural settings. There is a significant gap in the availability of affordable, user-friendly, and reliable communication systems that can be operated by users with only eye movements.

This project addresses the problem by developing a low-cost, camera-based eye gesture communication system that detects blinks and gaze direction to generate commands, enabling users with motor impairments to communicate and interact with external devices.

CHAPTER 2

LITERATURE SURVEY

2.1 A Servey about Eye Gesture Communication System

The development of assistive communication systems using eye gestures has gained significant interest in the fields of biomedical engineering, computer vision, and human-computer interaction. This section reviews key research efforts and technologies related to eye-tracking, blink detection, and low-cost communication systems.

1. **Hansen, D.W., Ji, Q. (2010).***In the Eye of the Beholder: A Survey of Models for Eyes and Gaze* (IEEE TPAMI).

This paper provides a comprehensive survey of gaze tracking and eye modeling techniques, identifying the trade-offs between accuracy and cost. It highlights the feasibility of using appearance-based and feature-based gaze estimation models for low-cost applications.

2. **San Agustin, J., et al. (2009).***Evaluation of a low-cost open-source gaze tracker* (ETRA Conference).

This study evaluated an open-source gaze tracking solution using standard webcams. It demonstrates that pupil detection and corneal reflection-based tracking methods can achieve acceptable accuracy levels with low-cost hardware.

3. **Krafka, K., et al. (2016).***Eye Tracking for Everyone* (CVPR).

Krafka et al. introduced a deep learning-based model, iTracker, capable of predicting gaze direction from mobile and webcam images. This work shows that convolutional neural networks trained on large datasets can offer robust gaze estimation without the need for expensive hardware.

4. OpenGazer Project – University of Cambridge

OpenGazer is an open-source software project that performs real-time gaze tracking using a webcam. It applies computer vision techniques to estimate the user's gaze direction, enabling hands-free interaction with digital interfaces.

5. ITU GazeTracker – IT University of Copenhagen

ITU GazeTracker is another open-source eye tracking platform designed for accessibility research. It enables gaze tracking using low-cost webcams and provides a customizable interface for integrating gaze-based input into various applications.

CHAPTER 3

EXISTING SYSTEM

3.1 EXISTING SYSTEM

Several eye-tracking and gesture-based communication systems are currently in use, primarily aimed at helping individuals with motor disabilities communicate or control their environment. These systems can be broadly classified into high-end commercial systems and basic DIY/academic prototypes.

1. Commercial Eye-Tracking Systems

- Examples: **TobiiDynavox**, **EyeTech Digital Systems**, and **EyeGaze Edge**.
- These devices use **infrared-based eye tracking** to accurately monitor pupil movement.
- Paired with specialized software, they allow users to select words, control computers, or even speak via synthesized voice output.

Limitations:

- **High cost**: Often priced above ₹1,00,000 (~$1,200+).
- Require **calibration** and are **not portable** or usable in all lighting conditions.
- Require **specialized support** and training for use.

2. DIY and Academic Prototypes

- Projects built using **Arduino**, **Raspberry Pi**, or **OpenCV** have shown promising results.
- These systems use webcams and open-source code to detect **blinks**, **gaze direction**, or **facial gestures**.

- Some integrate speech synthesis or control of home appliances (lights, fans).

Limitations:

- Many of these systems are **not real-time** or have **low reliability**.
- Limited **gesture mapping** and **hardware control** features.
- Often lack a **user-friendly interface** and **robust error handling**.

Key Gaps in Existing Systems

- Cost-effective systems usually **lack output integration** (like motors or speech).
- High-end systems are **not accessible** in low-resource settings.
- Few systems offer a **complete solution** combining real-time gesture recognition with hardware output.

3.2 ADVANTAGES OF EXISTING SYSTEMS

1. **High Accuracy (Commercial Systems)**

   Infrared-based systems like **Tobii Eye Tracker** provide very precise eye-tracking and gesture recognition.

2. **Wide Range of Features**

   Allow complex communication, virtual keyboard access, smart home integration, and voice synthesis.

3. **Proven Reliability**

   Commercial systems are tested, medically certified, and used in clinical environments worldwide.

4. Comfortable and Non-Invasive

Most systems are contactless and allow hands-free operation, enhancing user comfort.

3.3 DISADVANTAGES OF EXISTING SYSTEMS

1. High Cost

Prices often exceed ₹1,00,000 (~$1,200), making them unaffordable for most users in low-income or rural areas.

2. Hardware Dependency

Many systems require specialized **infrared cameras**, **head-mounted gear**, or **eye-tracking sensors**, which are not readily available.

3. Complex Setup

Require calibration and often depend on controlled lighting conditions for optimal performance.

4. Limited Customizability (Commercial Systems)

Proprietary systems do not allow modification, limiting flexibility for specific user needs or integration with local devices.

5. DIY Systems Lack Stability

Many open-source projects are experimental and may suffer from issues like poor real-time performance, inaccurate detection, or unstable output.

CHAPTER 4

PROPOSED SYSTEM

4.1 PROPOSED SYSTEM

The proposed system is a low-cost, eye gesture-based communication platform designed to help individuals with motor disabilities interact with their environment or caregivers using only eye movements. It uses a standard webcam to detect eye blinks and gaze direction in real-time, powered by computer vision algorithms (OpenCV and Dlib in Python). These detected gestures are mapped to specific commands, which are then sent to an Arduino microcontroller via serial communication. The Arduino responds by controlling physical outputs such as servo motors, buzzers, or LEDs.

This system serves as an affordable alternative to commercial eye-tracking systems and is tailored for home use, hospitals, or care centers where cost-effective and quick deployment is essential.

4.2 ADVANTAGES OF THE PROPOSED SYSTEM

1. **Low Cost**
    - Uses affordable components like Arduino UNO, SG90 servo motor, and a basic webcam.
    - Total cost remains under ₹5000 (~$60), making it highly accessible.
2. **Non-Invasive & Contactless**

    Requires no wearable gear or physical contact; only needs a webcam.

3. **Real-Time Processing**

   Detects and interprets gestures with minimal delay for seamless communication.

4. **Hardware Integration**

   Can control real-world devices (like motors or buzzers), unlike many DIY systems which are software-only.

5. **Customizable and Scalable**

   Easily upgradable to include voice output, IoT device control, or additional gestures.

6. **Open-Source & Flexible**

   Built using Python and Arduino, allowing full control and modification for specific use cases.

4.3 DISADVANTAGES OF THE PROPOSED SYSTEM

1. **Lighting Sensitivity**

   Performance may degrade under poor or inconsistent lighting conditions.

2. **Limited Gesture Set**

   Currently supports only basic gestures (e.g., blink, look left/right); complex commands require more development.

3. **Accuracy Can Vary**

   Detection may be less reliable for users with involuntary eye movements or specific medical conditions.

4. **Dependent on External Computer**

   Requires a computer to run the Python detection script, which may limit mobility unless a compact processor like Raspberry Pi is used.

CHAPTER 5

SYSTEM REQUIREMENTS

5.1 SYSTEM REQUIREMENT

5.1.1 Hardware Requirements

Component	Description
Webcam	Captures real-time video of the user's eyes and face
Laptop/PC	Runs the Python code and processes the video input
Arduino UNO	Receives commands from PC and controls physical devices
Servo Motor (SG90)	Simulates physical interaction (e.g., pointing, pressing)
USB Cable	Connects Arduino to PC for power and serial communication
Breadboard & Wires	For connecting components if needed
LED/Buzzer (optional)	For signaling output or alerts
Power Source	USB from computer or external battery (for portable Arduino setups)

5.1.2 Software Requirements

Software/Tool	Purpose
Python 3.x	Main programming language for computer vision and gesture detection
OpenCV	For image processing and real-time face/eye detection
Dlib	For facial landmark detection (used to compute Eye Aspect Ratio - EAR)
imutils	Utility library for OpenCV-based processing (resizing, drawing, etc.)
pyserial	To send serial data from Python to Arduino
pyttsx3 (optional)	For speech output (text-to-speech)
Arduino IDE	Used to write and upload code to the Arduino UNO board
Operating System	Windows, macOS, or Linux (compatible with all components and libraries)

CHAPTER 6

SYSTEM ARCHITECTURE

6.1 ARCHITECTURAL DIAGRAM

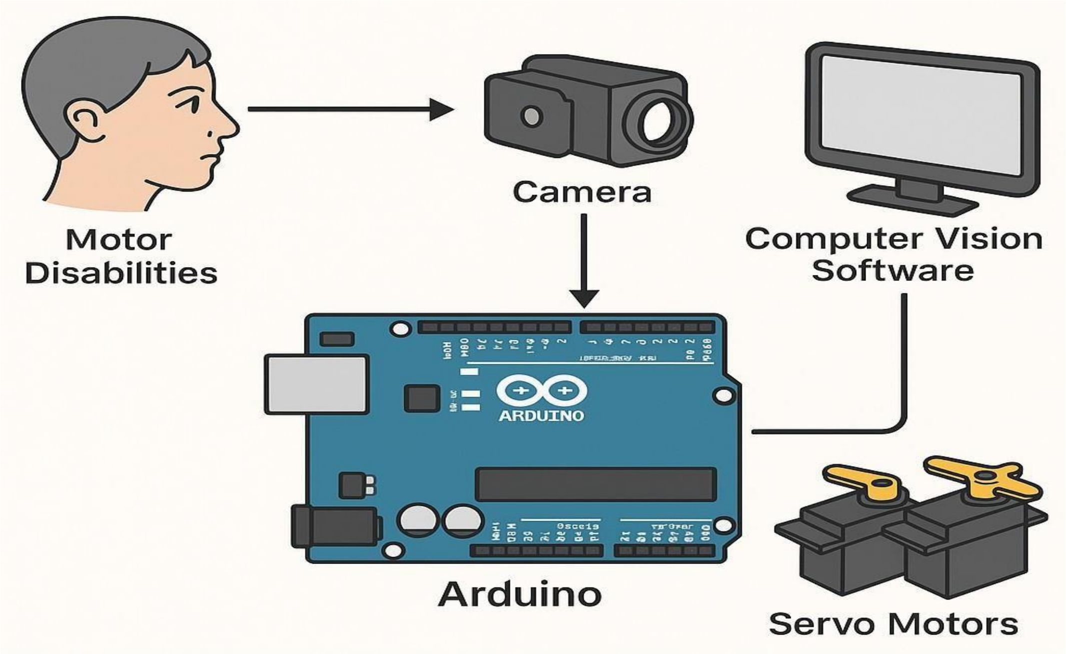


Figure 6.1 Architectural diagram

The eye gesture communication system for individuals with motor disabilities is designed to interpret eye movements and blinks as control commands for assistive devices like wheelchairs. At the input layer, a webcam or camera module captures real-time video of the user's face, focusing on the eyes. This video stream is processed by a computing device such as a laptop or Raspberry Pi, which runs an eye gesture detection algorithm using libraries like MediaPipe for face and iris tracking, and OpenCV for image processing. The system identifies key gestures such as looking left, right, up, down, and blinking. A calibration module can be included to personalize the system for each user by recording their natural eye positions and movements, ensuring accurate detection.

6.2 CIRCUITDIAGRAM

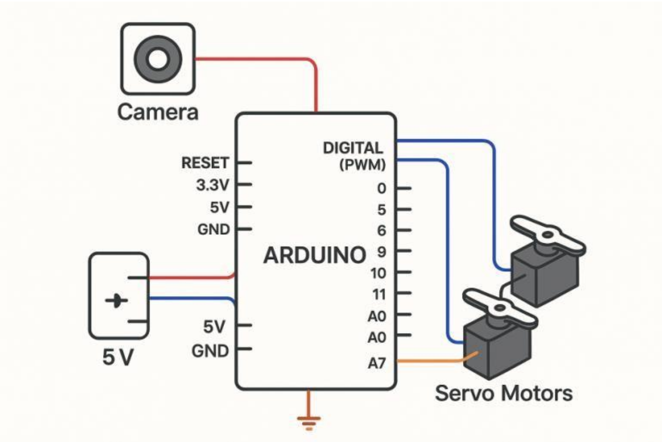


Figure 6.2 Circuit diagram

The circuit for the eye gesture communication system is centered around an Arduino Uno, which acts as the main control unit. It receives numeric control commands from a computer via USB serial communication. These commands correspond to specific eye gestures detected by the software (e.g., looking left, right, up, or blinking). Based on the received input, the Arduino controls various actuators connected to it.

6.3 FLOW CHART

The flow chart of the eye gesture communication system outlines the step-by-step process that begins with real-time visual input and ends with physical movement. The process starts when the camera captures live video of the user's face, focusing on the eyes. This video is passed to the processing unit (computer or Raspberry Pi), where MediaPipe and OpenCV are used to detect eye landmarks, track iris movement, and identify blink patterns. If a calibration step is included, the system first collects personalized gaze data to improve gesture accuracy.

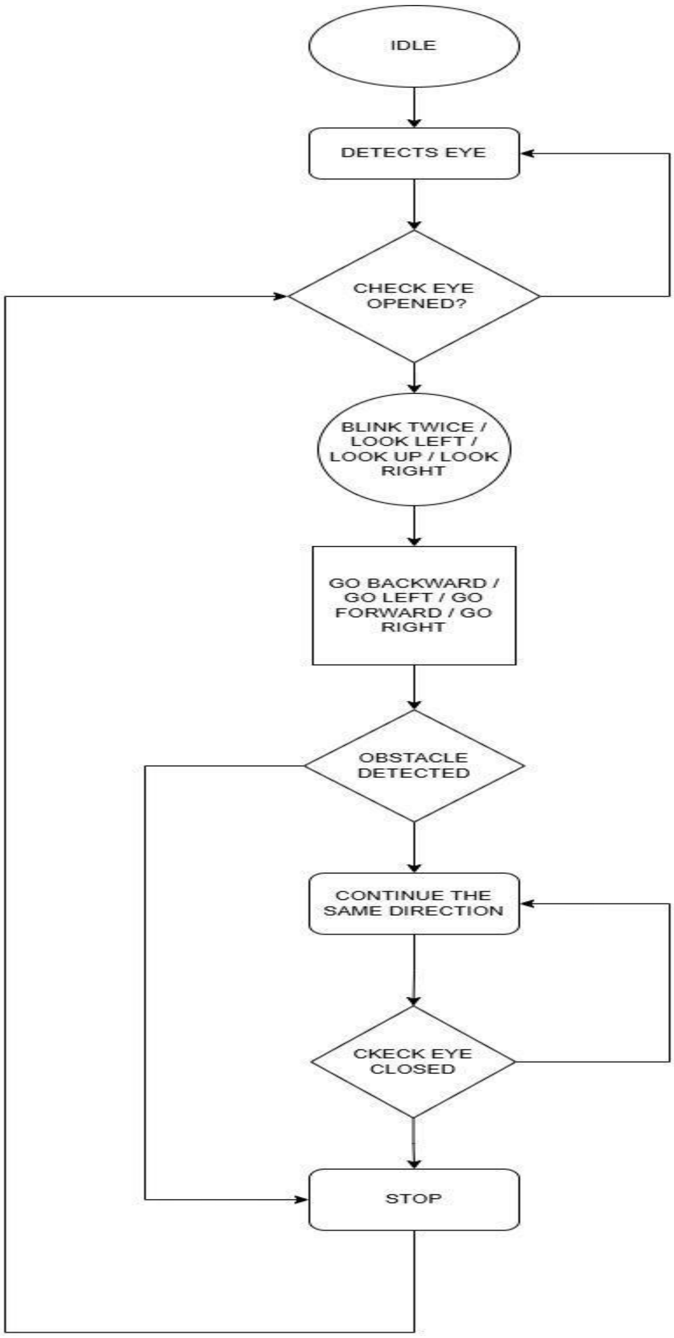


Figure 6.3 Flow chart

CHAPTER 7

MODULES DESCRIPTION

7.1 MODULE DESCRIPTION

1. Video Capture Module

Function:

Captures real-time video stream of the user's face using a webcam.

Tools Used:

- OpenCV (cv2)
- Standard USB or built-in webcam

Description:

This module initializes the camera, captures frames in real time, and sends each frame to the processing module for analysis.

2. Facial Landmark Detection Module

Function:

Detects facial features, especially eyes, and tracks eye positions.

Tools Used:

- Dlib (pre-trained 68 facial landmarks model)
- OpenCV for visualization

Description:

This module uses Dlib to identify and isolate facial landmarks (like eye corners, eyelids), which are essential for blink detection and gaze tracking.

3. Eye Gesture Recognition Module

Function:

Analyzes eye movements (blink, left, right) using geometric relationships.

Tools Used:

- Eye Aspect Ratio (EAR) formula
- Python logic for thresholding gestures

Description:

Calculates the Eye Aspect Ratio from landmark points. If the ratio drops below a threshold, it detects a blink. Eye direction is estimated based on iris positioning within the eye area.

4. Command Mapping Module

Function:

Maps detected eye gestures to predefined commands.

Tools Used:

- Python dictionaries or logic statements

Description:

Each eye gesture is assigned a specific action (e.g., blink = “Select”, look left = “Option 1”, look right = “Option 2”). This module translates gestures into text commands or control signals.

5. Serial Communication Module

Function:

Sends commands to Arduino via serial port.

Tools Used:

- PySerial library

Description:

Once a command is generated (like "move servo" or "activate buzzer"), it is sent to the Arduino over USB using serial communication.

6. Arduino Control Module

Function:

Receives commands from the PC and performs physical actions.

Tools Used:

- Arduino UNO
- Servo Motor / LED / Buzzer

Description:

The Arduino reads incoming serial data and activates the appropriate hardware (e.g., rotating servo, turning on buzzer).

7. Feedback Module (Optional)

Function:

Provides audio or visual confirmation of user input.

Tools Used:

- pyttsx3 for speech
- LEDs or on-screen GUI

Description:

Confirms detected gesture or action via speech or lights, giving users real-time feedback to enhance usability.

CHAPTER 8

ALGORITHM AND MODELS

8.1 DESCRIPTION OF ALGORITHM

1. EYE ASPECT RATIO (EAR)

Eye Aspect Ratio (EAR) is a geometric measure derived from facial landmarks around the eyes. It helps determine whether an eye is open or closed using a simple formula based on the vertical and horizontal distances between specific eye landmarks.

2. RULE-BASED (IRIS RATIO)

A rule-based (iris ratio) system is another method for tracking eye gaze direction or eye gestures, typically based on the position of the iris (pupil) within the eye region. This approach is particularly useful for determining left, right, up, or center eye movements in a communication system.

3. FINITE STATE MACHINE (FSM)

- A Finite State Machine (FSM) is a computational model used to design logic where the system can be in one state at a time, and transitions between states occur based on inputs or conditions.

- In an eye gesture communication system, FSMs are commonly used to interpret sequences of eye movements (e.g., blinks, gaze directions) into commands or messages.

4. TEXT-TO-SPEECH (TTS)

Text-to-Speech (TTS) is a technology that converts written text into spoken audio using synthetic or pre-recorded voices. It's widely used in assistive communication systems, especially for people with motor or speech impairments.

5. SERIAL COMMAND

- A Serial Command Algorithm is a logical sequence of steps used to:
- Generate, format, and transmit commands (as strings or bytes) via serial communication (e.g., USB/UART),
- Receive, parse, and act upon these commands on the receiving device (e.g., Arduino, microcontroller).

CHAPTER 9

CONCLUSION AND FUTURE ENHANCEMENT

9.1 CONCLUSION

The proposed Eye Gesture Communication System provides an affordable, accessible solution for individuals with severe motor disabilities to interact and communicate using simple eye movements. By integrating computer vision with Arduino-based hardware control, the system enables users to perform meaningful actions such as triggering a buzzer, activating a servo motor, or selecting commands—all through blinks and gaze direction. Unlike costly commercial eye-tracking solutions, this system emphasizes low cost, ease of use, and customizability, making it suitable for deployment in homes, hospitals, and care facilities. The modular approach and use of open-source tools ensure that the system is both scalable and adaptable for future needs.

9.2 FUTURE ENHANCEMENT

To increase usability and extend functionality, several enhancements can be incorporated into future versions of the system:

1. **Speech Output Integration**

   Add **text-to-speech (TTS)** functionality to convert selected commands into audible speech for non-verbal communication.

2. **IoT and Smart Home Integration**

   Enable control of smart home devices (lights, fans, alarms) using eye gestures via **Wi-Fi or Bluetooth modules** (e.g., ESP8266).

3. Advanced Gesture Recognition

Expand gesture set to include **long blink, double blink, up/down gaze**, and custom sequences for more complex communication.

4. Portable and Wireless Setup

- Replace the PC with a **Raspberry Pi** to make the system compact, mobile, and standalone.
- Add a **battery-powered module** for full portability.

5. AI-Based Personalization

Incorporate **machine learning models** to adapt gesture recognition to individual users for better accuracy.

6. Graphical Interface

Develop a **GUI** that visually displays available commands and the gesture detected, improving user feedback and ease of use.

7. Medical Integration

Link system with **emergency response alerts** or patient monitoring systems in hospital environments.

APPENDICES

A. SOURCE CODE (PYTHON)

Eye gesture.py

```
import cv2
import mediapipe as mp
import pyttsx3
import serial
import time
import numpy as np

# Serial setup (update COM port if needed)
arduino = serial.Serial('COM7', 9600, timeout=1)
time.sleep(2)

# Voice feedback
engine = pyttsx3.init()
def speak(cmd):
    engine.say(cmd)
    engine.runAndWait()

# MediaPipe setup
mp_face_mesh = mp.solutions.face_mesh
face_mesh = mp_face_mesh.FaceMesh(static_image_mode=False,
max_num_faces=1, refine_landmarks=True)

# Command map
command_values = {
    "RIGHT": 1,
    "LEFT": 2,
    "FORWARD": 3,
    "BACKWARD": 4,
    "STOP": 5
}

# Blink and debounce
blink_counter = 0
blink_threshold = 5
last_command = ""
last_time = time.time()
```

```
# Eye calibration data
calibration_data = {
    "LEFT": None,
    "RIGHT": None,
    "UP": None,
    "NEUTRAL": None
}
calibrated = False

# Eye landmarks for pupils
LEFT_PUPIL = 468
RIGHT_PUPIL = 473

# Calibration function
def calibrate_position(label, timeout=4):
    print(f"Look {label} for {timeout} seconds...")
    coords = []

    start_time = time.time()
    while time.time() - start_time < timeout:
        ret, frame = cap.read()
        if not ret:
            continue
        frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
        results = face_mesh.process(frame_rgb)
        if results.multi_face_landmarks:
            mesh = results.multi_face_landmarks[0].landmark
            left_pupil = mesh[LEFT_PUPIL]
            right_pupil = mesh[RIGHT_PUPIL]
            coords.append([(left_pupil.x + right_pupil.x) / 2, (left_pupil.y + 
right_pupil.y) / 2])
        cv2.putText(frame, f"Calibrating: Look {label}", (30, 40), 
cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
        cv2.imshow("Calibration", frame)
        if cv2.waitKey(1) & 0xFF == 27:
            break

    if coords:
        calibration_data[label] = np.mean(coords, axis=0)
        print(f"{label} position calibrated:", calibration_data[label])

# Command sender with debouncing
def send_command(cmd):
```

```
    global last_command, last_time
    if cmd != last_command and (time.time() - last_time) > 1.5:
        value = command_values.get(cmd, 0)
        if value != 0:
            arduino.write(f"{value}\n".encode())
            print(f"Sent: {cmd} ({value})")
            speak(cmd)
            last_command = cmd
            last_time = time.time()

# Start webcam
cap = cv2.VideoCapture(0)

# Calibration phase
for direction in ["NEUTRAL", "LEFT", "RIGHT", "UP"]:
    calibrate_position(direction)
cv2.destroyAllWindows()
calibrated = all(v is not None for v in calibration_data.values())

if not calibrated:
    print("Calibration failed. Exiting.")
    cap.release()
    arduino.close()
    exit()

# Thresholds
POSITION_TOLERANCE = 0.02
EYE_OPEN_THRESHOLD = 0.015

# Main loop
while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        break

    h, w = frame.shape[:2]
    frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    results = face_mesh.process(frame_rgb)

    if results.multi_face_landmarks:
        mesh = results.multi_face_landmarks[0].landmark
        left_pupil = mesh[LEFT_PUPIL]
        right_pupil = mesh[RIGHT_PUPIL]
```

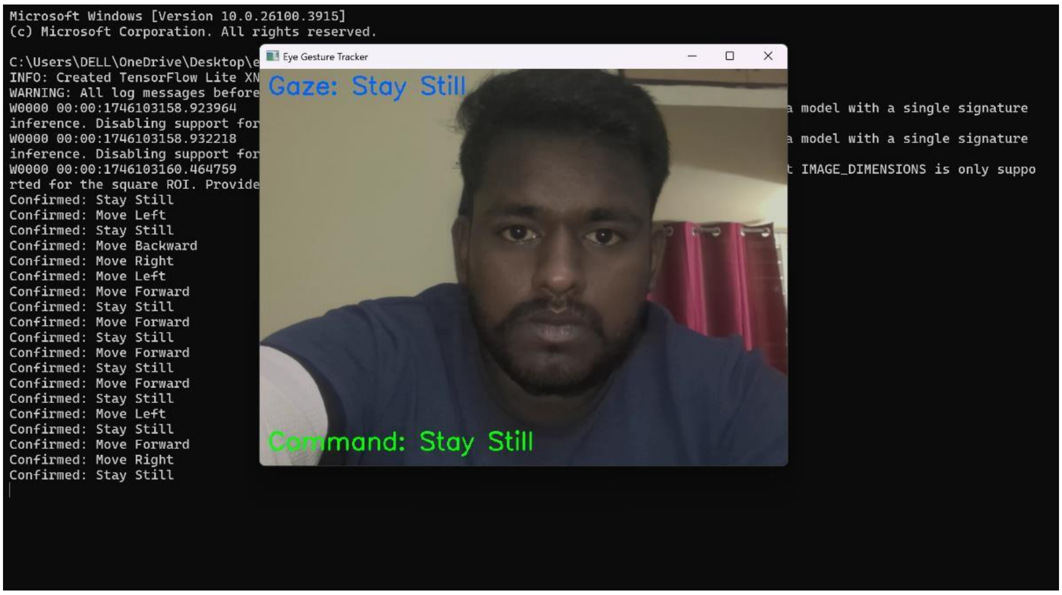
```python
        # Blink detection
        eye_top = mesh[159].y
        eye_bottom = mesh[145].y
        eye_open = eye_bottom - eye_top

        if eye_open < EYE_OPEN_THRESHOLD:
            blink_counter += 1
        else:
            if 1 <= blink_counter <= blink_threshold:
                send_command("STOP")
            elif blink_counter > blink_threshold:
                send_command("BACKWARD")
            blink_counter = 0

        # Gaze detection using Euclidean distance
        pupil_avg = np.array([(left_pupil.x + right_pupil.x) / 2, (left_pupil.y +
right_pupil.y) / 2])
        distances = {key: np.linalg.norm(pupil_avg -
np.array(calibration_data[key])) for key in calibration_data}

        # Find closest match
        command = min(distances, key=distances.get)
        if distances[command] < POSITION_TOLERANCE:
            send_command(command)

    cv2.imshow("Eye Gesture Control", frame)
    if cv2.waitKey(1) & 0xFF == 27:
        break

cap.release()
cv2.destroyAllWindows()
arduino.close()
```

B. ARDUINO CODE

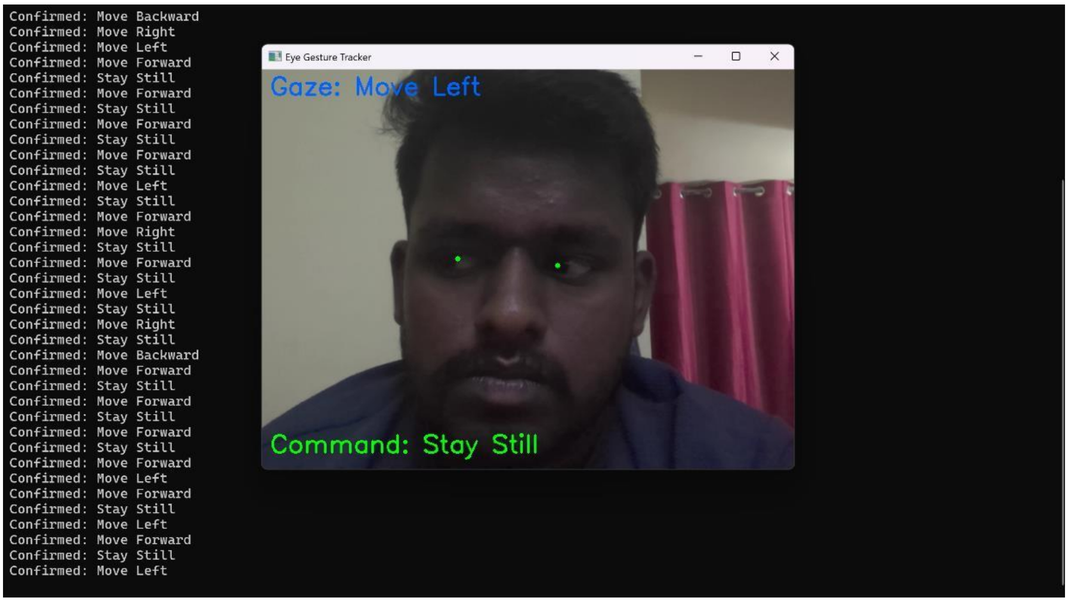
```cpp
#include <Servo.h>
int motorPin1 = 5;
int motorPin2 = 6;
int servoPin = 9;
Servo myServo;  // Create a Servo object to control the servo motor
```

```
void setup() {
  Serial.begin(9600);
  pinMode(motorPin1, OUTPUT);
  pinMode(motorPin2, OUTPUT);
  myServo.attach(servoPin);  // Attach the servo motor to the pin
}

void loop() {
  if (Serial.available()) {
    int command = Serial.parseInt();

    if (command == 1) {  // RIGHT
      // Rotate servo to a specific angle for "RIGHT"
      myServo.write(10);  // Rotate servo to 90 degrees (example for right)
    } else if (command == 2) {  // LEFT
      // Rotate servo to a specific angle for "LEFT"
      myServo.write(170);  // Rotate servo to 0 degrees (example for left)
    }
     else if (command == 3) {  // FORWARD
      // Control motors for forward movement
     digitalWrite(motorPin1, HIGH);
     digitalWrite(motorPin2, LOW);
    } else if (command == 4) {  // BACKWARD
      // Control motors for backward movement
      digitalWrite(motorPin1, LOW);
      digitalWrite(motorPin2, HIGH);
    } else if (command == 5) {  // STOP
      // Stop the motors and set servo to neutral position
      digitalWrite(motorPin1, LOW);
      digitalWrite(motorPin2, LOW);
      myServo.write(90);  // Set servo to neutral position (e.g., 90 degrees)
    }
```

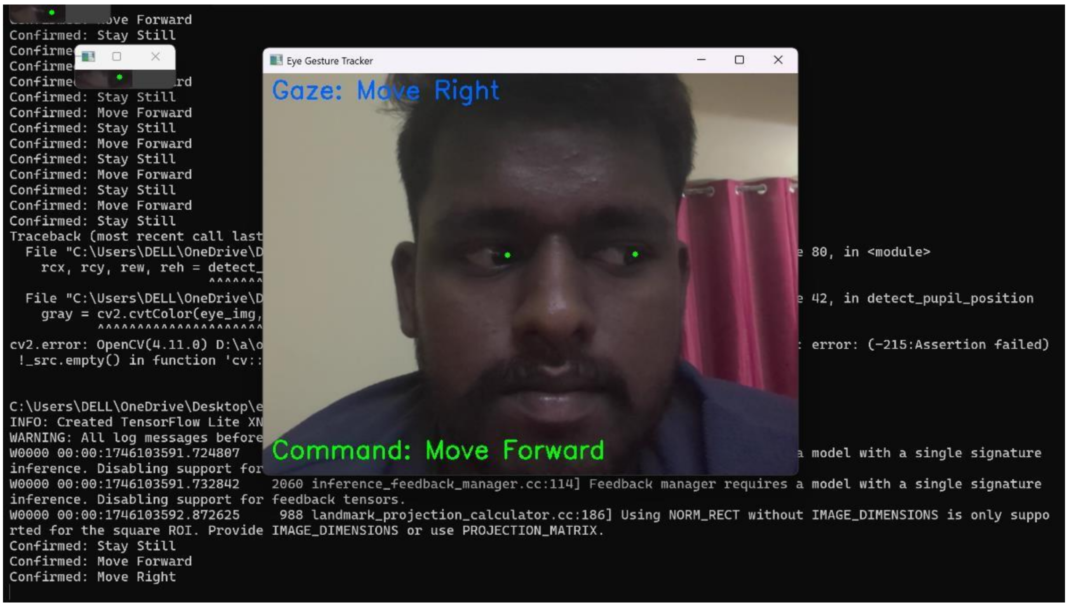
C. SCREENSHOTS



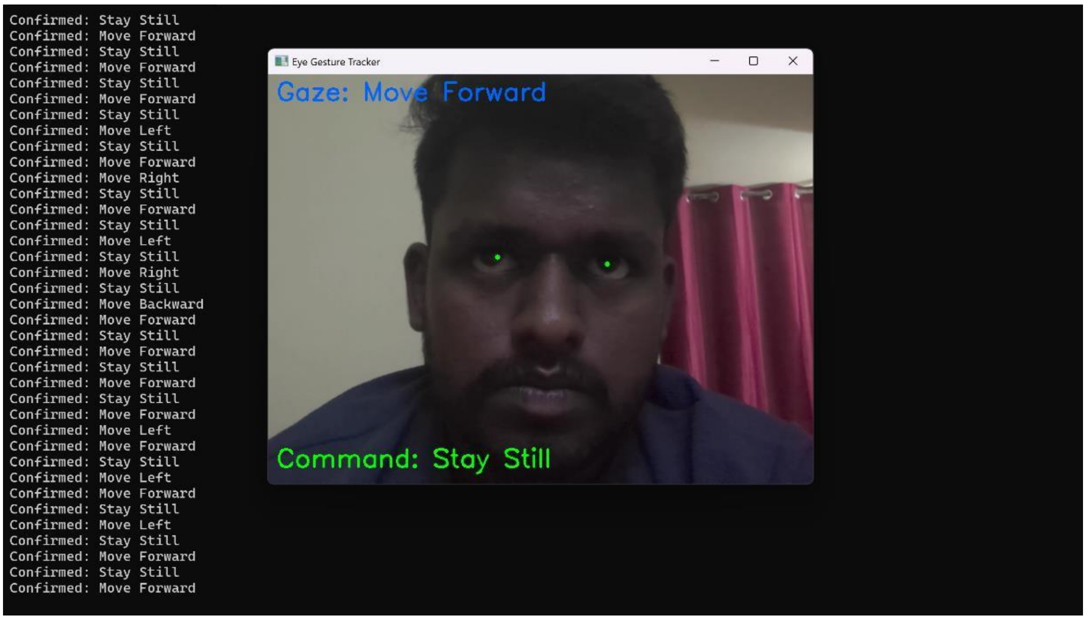
Scr.No.1:Stay still



Scr.No. 2:Move Left



Scr.No.3:Move Right



Scr.No.4: Move forward

REFERENCES

[01] NWanluk, S Visitsattapongse, A Juhong (2016). Smart wheelchair based on eye tracking. 2016 9th Biomedical Engineering International Conference (BMEiCON). doi: 10.1109/BMEiCON.2016.7859594

[02] MA Eid, N Giakoumidis, A El Saddik(2016). A novel eye-gaze-controlled wheelchair system for navigating unknown environments: case study with a person with ALS. IEEE Access (Volume:4). doi: 10.1109/ACCESS.2016.2520093

[03] A Ishizuka, A Yorozu(2017). Motion control of a powered wheelchair using eye gaze in unknown environments. 2017 11th Asian Control Conference (ASCC). doi: 10.1109/ASCC.2017.8287148

[04] AniwatJuhong, T Treebupachatsakul, C Pintavirooj (2018). Smart eye-tracking system. 2018 International Workshop on Advanced Image Technology (IWAIT). doi: 10.1109/IWAIT.2018.8369701

[05] N Oda (2018). Development of Power Assist Wheelchair Control with Driving Assistance using Fish-eye Vision. 12th France-Japan and 10th Europe-Asia Congress on Mechatronics. doi: 10.1109/MECATRONICS.2018.8495884

[06] A Rajesh, M Mantur (2017). Eyeball gesture controlled automatic wheelchair using deep learning. 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC). doi: 10.1109/R10-HTC.2017.8288981

CONFERENCE:

[1] Prof.R.Monisha, Deenadayalan R (611821104006), Dineshkumar V (611821104008), Suriya A (611821104051) attended a International Level Conference of Innovative Research in Science, Engineering and Management at P.S.V. College of Engineering and Technology, Presented a paper entitled **“LOW COST EYE GESTURTE COMMUNICATION SYSTEM FOE PEOLE WITH MOTOR DISABILITIES”** on April 2025 (ICNTSETM’25).

CONFERENCE CERTIFICATE:


INSTITUTION'S INNOVATION COUNCIL

P.S.V

COLLEGE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)
(Approved by AICTE & Affiliated to Anna University, Chennai)
(Inclusion Under Section 2(f) & 12(B) of the UGC Act, 1956)
Chennai - Bangalore Highway (NH-46), Krishnagiri - 635108.

2nd INTERNATIONAL CONFERENCE ON NEW TRENDS IN SCIENCE, ENGINEERING, TECHNOLOGY AND MANAGEMENT - 2025
(ICNTSETM'25)-HYBRID MODE
(BOTH ONLINE & OFFLINE)
ISBN-978-93-48505-24-8

Certificate Of Appreciation

This is to certify that Dr./Mr./Ms./Mrs. Deenadayalan R of P.S.V College of Engineering and Technology has participated / presented a paper entitled Low cost eye gesture communication system for people with motor disabilities in the 2nd International Conference on New Trends in Science, Engineering, Technology & Management (ICNTSETM'25) held on 25th & 26th April 2025 at P.S.V College of Engineering and Technology, Krishnagiri, Tamilnadu.

Convener | Principal

INSTITUTION'S INNOVATION COUNCIL

P.S.V COLLEGE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)
(Approved by AICTE & Affiliated to Anna University, Chennai)
(Inclusion Under Section 2(f) & 12(B) of the UGC Act, 1956)
Chennai - Bangalore Highway (NH-46), Krishnagiri - 635108.

2nd INTERNATIONAL CONFERENCE ON NEW TRENDS IN SCIENCE, ENGINEERING, TECHNOLOGY AND MANAGEMENT - 2025

(ICNTSETM'25)-HYBRID MODE
(BOTH ONLINE & OFFLINE)
ISBN-978-93-48505-24-8

Certificate Of Appreciation

This is to certify that Dr./Mr./Ms./Mrs. Dinesh Kumar. V of P.S.V College of Engineering and Technology has participated / presented a paper entitled Low cost eye gesture communication system for people with motor disabilities in the 2nd International Conference on New Trends in Science, Engineering, Technology & Management (ICNTSETM'25) held on 25th & 26th April 2025 at P.S.V College of Engineering and Technology, Krishnagiri, Tamilnadu.

Convener

Principal

INSTITUTION'S INNOVATION COUNCIL

P.S.V COLLEGE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)
(Approved by AICTE & Affiliated to Anna University, Chennai)
(Inclusion Under Section 2(f) & 12(B) of the UGC Act, 1956)
Chennai - Bangalore Highway (NH-46), Krishnagiri - 635108.

2nd INTERNATIONAL CONFERENCE ON NEW TRENDS IN SCIENCE, ENGINEERING, TECHNOLOGY AND MANAGEMENT - 2025

(ICNTSETM'25)-HYBRID MODE
(BOTH ONLINE & OFFLINE)
ISBN-978-93-48505-24-8

Certificate Of Appreciation

This is to certify that Dr./Mr./Ms./Mrs. Suriya A of P.S.V College of Engineering and Technology has participated / presented a paper entitled low cost eye gesture communication system for people with motor disabilities in the 2nd International Conference on New Trends in Science, Engineering, Technology & Management (ICNTSETM'25) held on 25th & 26th April 2025 at P.S.V College of Engineering and Technology, Krishnagiri, Tamilnadu.

Convener

Principal

PUBLICATION:

[1] Prof.R.Monisha, Deenadayalan R (611821104006), Dineshkumar V (611821104008), Suriya A (611821104051) Published a paper in a journal called "International Journal Innovative Research in Computer and Communication Engineering (IJIRCCE) under a topic **"LOW COST EYE GESTURTE COMMUNICATION SYSTEM FOE PEOLE WITH MOTOR DISABILITIES"**. It is published on May, 2025.Issue-ISSN:2320-9801.