INDUSTRIAL ROBOTICS PROJECT

*Group Members*

~Mohammad Safwan(22BMH1105)

~Vikash Madhava(22BMH1083)

~Charan (22BMH1125)

TITLE:

5-axis robot(TRR:TR)

OBJECTIVE:

The objective of this project is to design and implement a 5-DOF robotic arm that can be intuitively controlled using both **hand gestures** and **voice commands**, providing a natural and contactless human–machine interaction system. The robotic arm uses six servo motors (MG996R and SG90) driven via an Arduino microcontroller, with gesture recognition powered by Python and a webcam using the MediaPipe framework. Voice commands are processed in real-time to trigger specific modes or actions, allowing seamless switching between control methods. The system aims to demonstrate an efficient fusion of computer vision, speech recognition, and embedded control to perform precise, responsive movements of a robotic manipulator.

Methadology:

This project follows a structured approach to develop a 5-DOF robotic arm that can be controlled via both hand gestures and voice commands. The methodology is divided into the following key stages:

**1. Servo Selection and Power Planning**

* **Load Estimation**: Initial calculations were done to estimate the torque required at each joint based on the length of the arm links, expected payload, and gravitational effects.
* **Servo Choice**:
  + **MG996R servo motors** were selected for high-torque joints (base, shoulder, and elbow) due to their ~11 kg·cm torque capability.
  + **SG90 mini servos** were chosen for low-load joints (wrist pitch, wrist roll, and gripper) due to their compact size and ~1.8 kg·cm torque.
* **Power Supply**: A **5V 2A power adapter** was used to power all servos through a **PCA9685 16-channel PWM driver**, ensuring safe and simultaneous actuation without overloading the Arduino.

**2. Mechanical Design in SolidWorks**

* All robotic arm parts—including the **rotating base**, **Arm 1**, **Arm 2**, **wrist housing**, and **gripper with gear mechanism**—were designed using **SolidWorks**.
* **Servo Dimensions Incorporated**: The exact dimensions of the MG996R and SG90 servos were taken into account during CAD modeling to ensure proper fitting within each joint’s casing or mount.
* The design ensures:
  + **Clearance between moving parts**
  + **Avoidance of collisions**
  + **Mounting brackets** that align servo horns with rotational axes
* The gripper was designed with a **gear-driven linkage** that opens and closes symmetrically based on the servo angle.

**3. Software & Control System**

**a. Hand Gesture Control (Python + MediaPipe):**

* Real-time hand tracking was implemented using **MediaPipe** in Python.
* Left hand is for servo selection
* Right hand for increment and decrement

**b. Voice Command Control (Python + SpeechRecognition):**

* **SpeechRecognition** and **PyAudio** libraries were used to detect voice commands.
* Commands such as "start gesture control", "run inverse kinematics", and "reset to home" were implemented to trigger specific modes.
* Allows hands-free transitions between control types.

**c. Inverse Kinematics (Python) end position movement with matplot visualization:**

* IK was applied to control the **end-effector position** based on 2D or 3D coordinates using trigonometric calculations.
* User-defined paths or keyboard inputs (w, a, s, d) allow the arm to follow smooth trajectories.
* Joint limits are enforced to ensure movement remains within servo capability (0°–180°).
* With the use of matplot the the input of theatha 1 and theata 2 for each changes the respective visualization can be formed

**4. Arduino Code**

* The **Arduino Uno** receives serial commands from Python in the format:  
  servo\_index, angle (e.g., 2,135)
* The Arduino parses this data and moves the corresponding servo using the Servo.h library.
* This setup ensures fast and responsive servo actuation while offloading complex logic to the Python side.

Components used

| **S.No** | **Quantity** |  | **Component Name** | **Description** |
| --- | --- | --- | --- | --- |
| 1 | 1 |  | Base 1 | 3D printed base for the arm |
| 2 | 1 |  | base\_link\_upper1 | 3D printed upper base part |
| 3 | 1 |  | Arm link\_1 | 3D printed first arm link |
| 4 | 1 |  | Arm link\_2 | 3D printed second arm link |
| 5 | 1 |  | Arm link\_3 | 3D printed third arm link |
| 6 | 1 |  | Gripper base | 3D printed base of the gripper |
| 7 | 1 |  | Gripper 2 | 3D printed gripper second finger |
| 8 | 1 |  | Gear right | 3D printed gear for gripper finger (right) |
| 9 | 1 |  | Gear left | 3D printed gear for gripper finger (left) |
| 10 | 1 |  | Gripper | 3D printed complete gripper assembly |
| 11 | 4 |  | Link | 3D printed gripper link parts |
| 12 | 2 |  | Spacer | 3D printed gripper joint spacers |
| 13 | 3 |  | MG996R Servo Motor | High torque servo for base and heavy joints |
| 14 | 3 |  | SG90 Micro Servo Motor | Compact servo for gripper and lightweight joints |
| 15 | 3 |  | MG955 Horn | Servo horns for MG996R servos |
| 16 | 3 |  | SG90 Servo Horn | Horns for SG90 servos |
| 17 | 1 |  | Arduino Uno or Mega | Microcontroller for receiving and processing commands |
| 18 | 1 |  | PCA9685 Servo Driver Board | I2C-based PWM driver for controlling multiple servos |
| 19 | 1 |  | Breadboard | For prototyping potentiometer and sensor circuits |
| 20 | 1 |  | 5V 2A Power Supply | External power source for driving all servos reliably |
| 21 | 1 |  | Jumper Wires (male-male) | For connections between Arduino, PCA, and sensors |
| 22 | 1 |  | Jumper Wires (male-female) | For potentiometer and sensor connections |
| 23 | 1 |  | Wooden Board | Base platform for securely mounting the robotic arm |

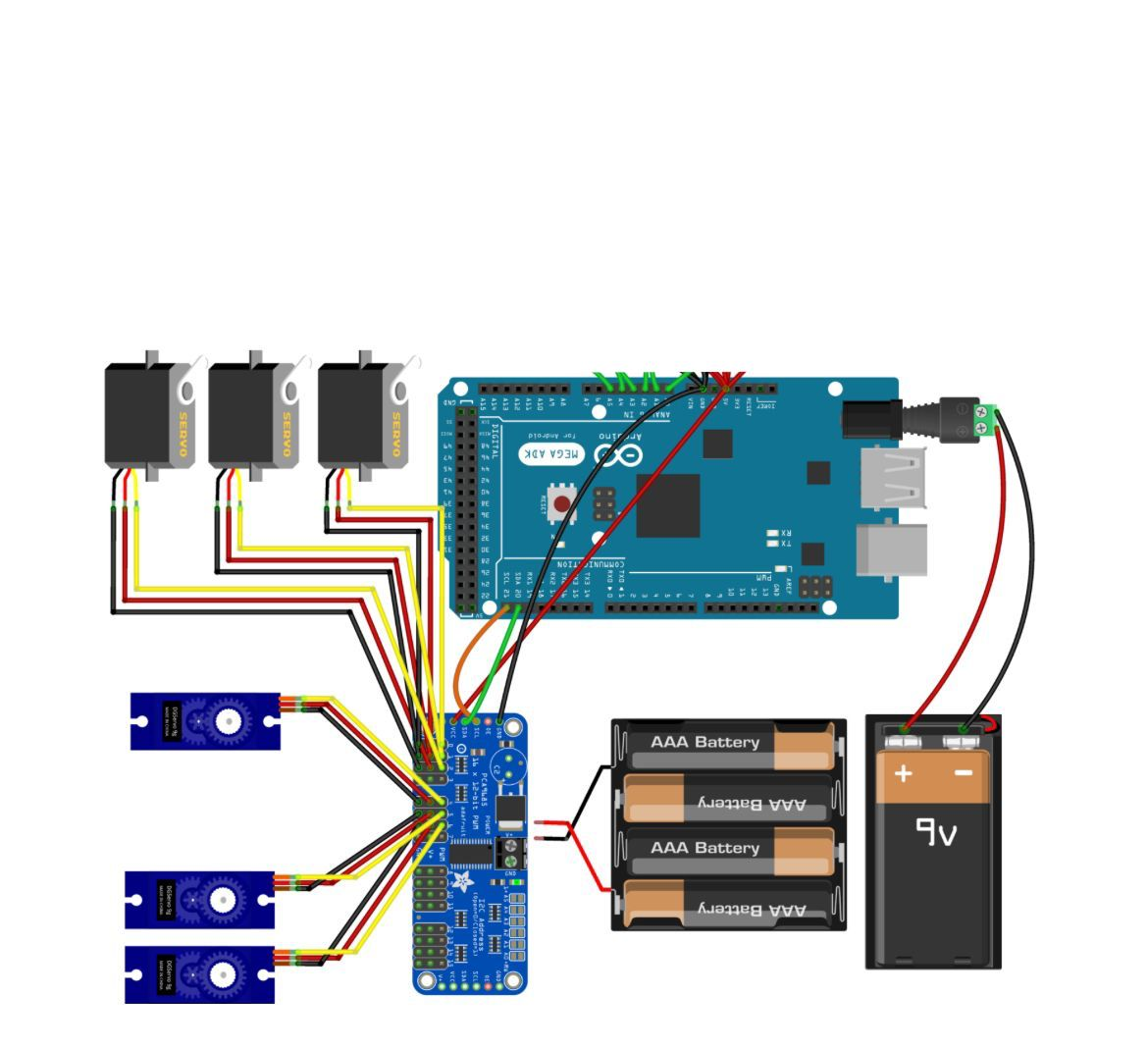
Specifications of servo MG996R:

| **Parameter** | **Value** |
| --- | --- |
| **Operating Voltage** | 4.8V – 7.2V |
| **Typical Voltage** | 5V or 6V |
| **Current (No Load)** | ~200 – 250 mA |
| **Current (Stall)** | ~2.5 A |
| **Torque @ 6V** | ~11 kg·cm |
| **Speed @ 6V** | ~0.14 sec/60° |
| **Step Angle** | Not fixed (analog positional servo) |
| **Control Angle** | ~0° to 180° (sometimes a bit more, ~200°) |
| **PWM Control Signal** | 50 Hz (20 ms period), 1ms = 0°, 2ms = 180° |

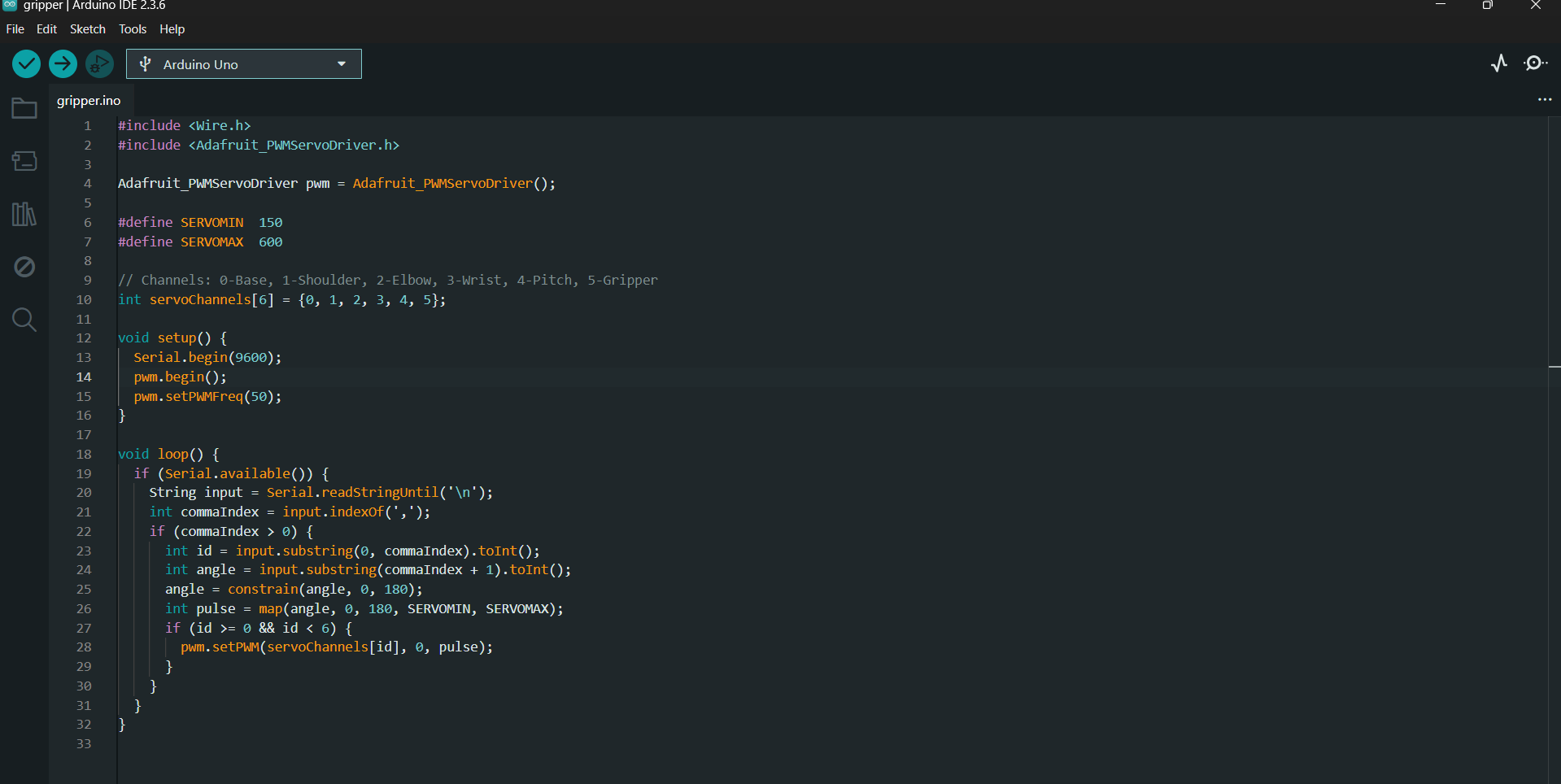
Specifications of servo MG996R:

| **Parameter** | **Value** |
| --- | --- |
| **Operating Voltage** | 4.8V – 5V |
| **Typical Voltage** | 5V |
| **Current (No Load)** | ~100 – 250 mA |
| **Current (Stall)** | ~650 – 800 mA |
| **Torque @ 5V** | ~1.8 kg·cm |
| **Speed @ 5V** | ~0.12 sec/60° |
| **Step Angle** | Not fixed (analog positional servo) |
| **Control Angle** | ~0° to 180° |
| **PWM Control Signal** | 50 Hz, pulse width 1ms to 2ms for 0°–180° |

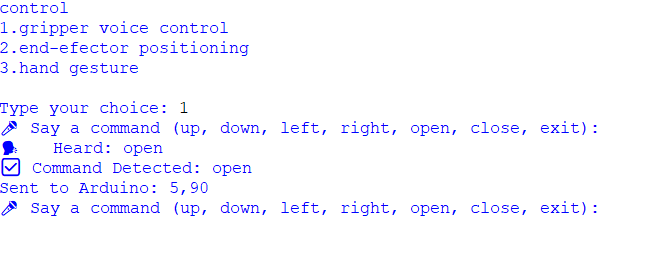
ARDUINO SCHEMATICS:



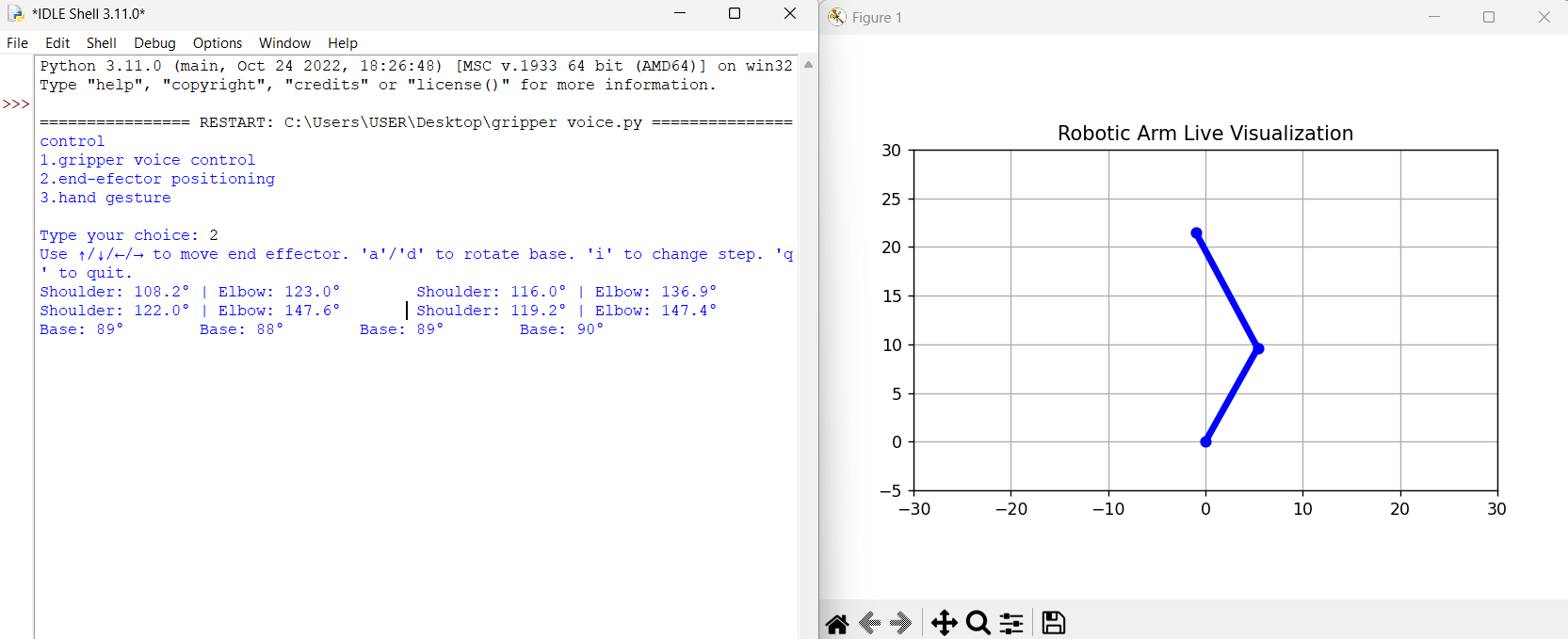
ARDUINO CODE:



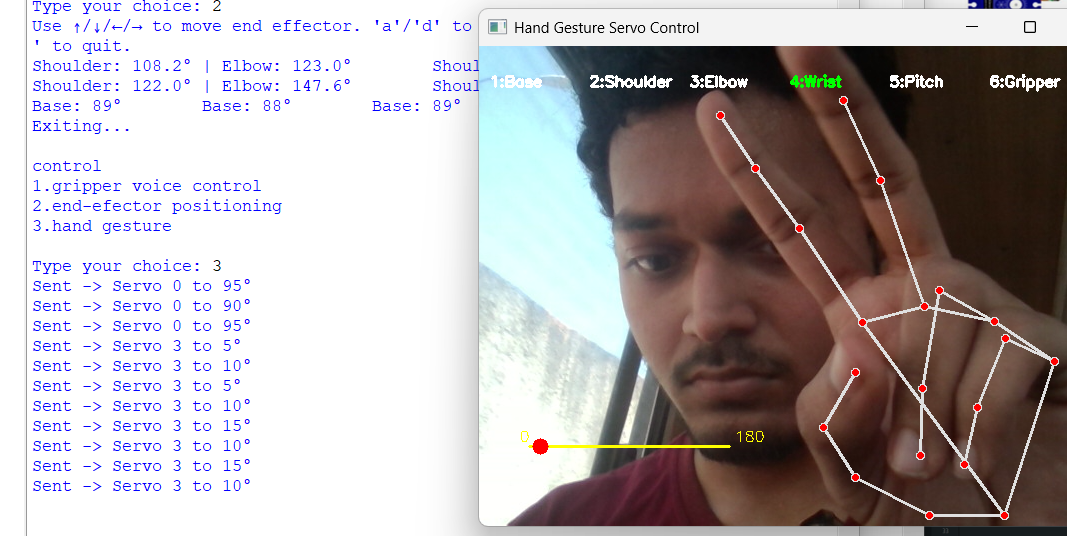
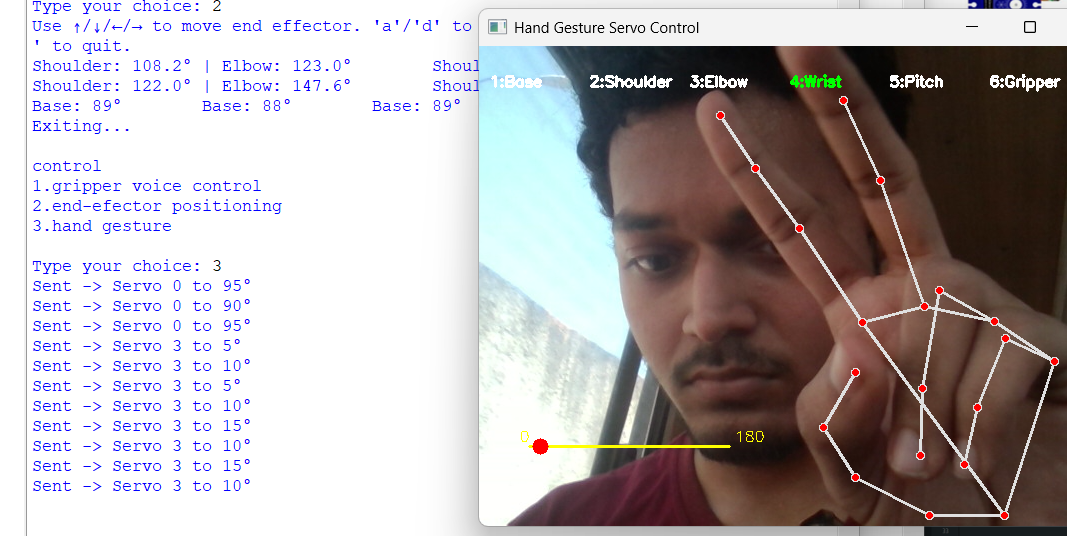
**WORKING AND OUTPUT:**

1)Voice gesture: when voice gesture is choose, u r prompted to give a command to the computer to move the end effector ,the command available for end effector and open, close, right ,left, up, down and exit to go back to menu 

2)End-effector positioning: when this option is chose ,out key board has full control over the robot to move the end effector in x and y plane with the help of up arrow keys ,to move the base we can use the ‘a’ and ‘d’ keys for clockwise and anticlockwise



3)Hand gesture control: this works simply by the webcam input of the computer. with this we are able to control all the 5 axis and the gripper open and close. the hand gesture works in the manner where the left hand is used to choose which servo to rotate and the right hand is used to increment or decrement the chosen servo(middle finger bend to increment, index finger bend to decrement)



MODEL PHOTHO:



**COURSE OUTCOMES:**

**CO1. Understand the fundamentals of robots and their terminologies.**

**CO2. Prepare the program for various robotic applications.**

**CO3. Apply the concept of robots to solve real-world problems.**