Nat. Volatiles & Essent. Oils, 2021; 8(5): 3133-3143

Gesture Based Robot Arm Control

[1]Rathika P D, [2]Jai Gowtham Sampathkumar, [3]Aravinth Kumar T, [4]Shri Ram Senthilkumar

[1]Robotics and Automation
PSG College of Technology
Coimbatore, India
pdr.rae@psgtech.ac.in

[2]Robotics and Automation
PSG College of Technology
Coimbatore, India
jaigowsam@gmail.com

[3]Robotics and Automation
PSG College of Technology
Coimbatore, India
aravinthkumar.thangavelu@gmail.com

[4]Robotics and Automation
PSG College of Technology
Coimbatore, India
shriramtg34@gmail.com

Abstract

Robotics is a constantly evolving field nowadays. A robot is a mechanical device that can perform physical tasks while being controlled and supervised by humans. Several robots have been created to perform risky tasks that are impossible for humans to perform directly. The robot arm is one of the most popular types. In this project, it is proposed to control a robot arm using finger and hand movements. However, the traditional way of gesture control employs sensor-based gloves to track motion, which is time and energy intensive due to its weight. To overcome this problem, using vision for motion tracking can be the most suited method at hand currently. The ML based computer vision method provides us with the real-time landmark on different points of the hand. With these landmarks we can calculate various mathematical parameters that can be used for the control of the robot arm. Movements that are landmarked, are processed and formulated using the mathematical parameters. The final output of mathematical parameters fed to the microcontroller. Signals from the microcontroller will be the desired input signal to the motors. The rotation of the robot arm is controlled using movement of the human arm. The end application of this project is to be as similar as human operation in dull, dirty and dangerous environments. The project's major goal is to protect those who come into direct contact with these environments. Bomb defusing, painting, welding, and other applications are a few examples of where this project is useful.

***Keywords*—** Robot Arm, Media Pipe, Gesture, OpenCV, Arduino, Servo.

I. Introduction

People nowadays have a constant need for extra assistance systems to improve productivity and safety. This will necessitate the use of automation systems. Standard automation systems, skilled and well-trained employees are required to produce the high-quality products. With the advancement of technology, these machines that formerly required human assistance to work have been made to operate spontaneously without the need for human power. Robots are one of the most common components of automation systems. Robot arm are machines which are configured to do a given activity as well as duty fastly, effectively, but also precisely. They're generally motor actuated and used for doing heavy or highly repetitive operations quickly and consistently over long durations, and they're particularly useful in the machining, manufacturing, production and assembling industries. A standard industrial robot arm consists of a manipulators, articulations and a series of joints that works with each other to mimic the motion and capabilities of a human arm as precisely as feasible. A gesture-controlled robot arm is a sort of action that functions by detecting and responding to hand gestures by robots. Gesture-based computing interfaces where the human body interacts with digital content, such as computer vision-based approaches and a gesture-controlled robot arm is a type of robot that works based on signals supplied by hand gestures. The robotic arm moves and executes the work based on human hand gestures, and this technology mimics human hand motions. A vision-based method makes it simple and easy to manage a robot arm for a variety of operations such as palletizing, painting, welding, and so on. Articulated robots are those that have rotating joints. Axes are the term used in robotics to describe these joints. Servo motors are commonly used to power articulated robots. Servo motors can be two axis design or up to ten axis or more. In industrial robots, four to six axes are common. In industrial applications, six axes are the most typical.

II. RELATED WORK

A lot of researches have been conducted in the area related to Gesture Based Robot Arm Control

Poltak Sihombing et al. [1] focus on the robot arm controlling is based on the movement of fingers and hands. The research work is aimed at the brief description of control of a simple robot arm by movement of human fingers and human hand. This project has created a robot arm using a Fuzzy logic technique to control the movement of robot arm. Yanmin Zhu et al. [2] focus in using basic background subtraction, skin color detection, hand reach detection, palm detection. The detection of a complete palm is used to determine whether the hand extends beyond the camera's range of vision. The ergonomics idea is used to assess if the hand is outside of the camera's field of view.

There has also been some more study done in this area. Using RGB cameras, PramodKumar et al. [3] investigated numeric and subjective comparisons of algorithms and approaches. A collection of 13 metrics developed from various algorithm attributes and the experimental methodologies used in algorithm assessment are utilized to evaluate algorithms quantitatively. The project emphasizes the need of taking these factors into account, as well as the algorithm's identification accuracy, when predicting its success in real-world applications. Sakshi Sharma et al. [4] described how he used accelerometers to control 5 servo motors with gestures. The use of an accelerometer to drive a robot arm is explored and implemented in the hardware.

III. SYSTEM COMPONENTS

In recent years, technology has provided various sources of basic items that assist people in meeting their requirements. As a result, the software's and algorithms assist developers in using the package, which is either an open-source platform or a commercial product, to build their own goods, which is helpful for various industries. Such software's and algorithms will be updated periodically to make a better component. The following software and algorithms are used in this project. They are PyCharm Edu 2021.1.2 - Software, Arduino IDE - Software, Machine Learning Pipeline - Algorithm, MediaPipe – Library. The following components are used in this project. They are Arduino Uno, Servo motors, 5V Battery.

A. PyCharm IDE

PyCharm is an integrated development environment (IDE) for computer programming that focuses on the Programming in Python Language. It was created by the JetBrains Corporation (previously identified as IntelliJ). PyCharm IDE supports website design and data science with code analyzing, a visual debugging, an integrated unit tester, version control system integration, and Django and Anaconda integration. PyCharm is compatible with Linux, Mac OS X, Windows versions. PyCharm includes coding aid and inspection, auto - completion, syntax and error highlights, linter integration, and rapid fixes.

B. Arduino IDE

The Arduino Integrated Development Environment (IDE) is a software application that functions on computers running Windows, Mac OS X, and Linux. It's used to compile and execute programs on Arduino boards and other development boards that enable third-party cores. By connecting to the Arduino and Genuino devices and uploading code, it communicates with them. The Arduino IDE includes a text editor, a serial monitor, a text editor, a toolbar with activity buttons, and a number of menus.

C. Machine Learning Pipeline

A machine learning pipeline codifies and automates the process of developing a machine learning model. Data extraction, pre-processing, model training, and deployment are all handled by machine learning pipelines. MediaPipe Hands uses a machine learning pipeline that comprises a number of interconnected models. A palm detection model which works on the entire frame comes back as an aligned hand boundary. From the palm detector-cropped picture area, a hand landmark model extracts high-fidelity 3-dimensional hand key points. Moreover, utilizing the hand landmark identified in the previous frame, crops may be made in the pipelines, and palm detection is only required to re-localize the hand when the landmark models never longer detects its presence.

D. MediaPipe

Recognizing the motion and form of hands could enhance the user experience in a wide range of technical areas and interfaces. In augmented reality, it enables the texture of digital material and information on top of the real world, as well as the interpretation of sign language and the control of hand movements. Because palms frequently entirely cover themselves or one another (for instance, hand occlusions and hand shaking), and there are no high contrast styles, hand perception in real time is a particularly difficult computer vision task. MediaPipe's Hands is a high-resolution tracking device for hands and fingers. 21 3-dimensional landmarks of a hand are determined using machine learning. A single image is used to determine 21 3D landmarks of a hand using machine learning. While most current framework systems rely on complex desktop environments for inference, our approach works in real time on a cell phone and can be scaled to many hands. By making these hand recognition abilities open to the rest of the research and innovation community, the designers expect that new applications and research routes will emerge, leading to new applications and research.

E. Arduino

The ATmega328P is used in the Arduino Uno, which was created by Arduino.cc. There are numerous digital and analog I/O pins. This allows the arduino to be connected to external circuits. As an output, this board can run and control relays, LEDs, servos, and motors, and it can connect to other Microcontrollers. The board has 6 analog and 14 digital input/output ports (six of which are utilized for PWM output). The Arduino Uno is powered by a 9 V battery or a USB wire. It can handle voltages ranging from 7 to 20 volts. The Arduino Uno is shown in fig. 1.

.

Fig. 1. Arduino UNO

F. Servo Motors

A servo motor is a rotary actuator that enables accurate angular position control. A motor and a position feedback sensor are used in it. To complete the system, a servo drive is necessary. The feedback sensor is used by the drive to accurately regulate the motor's rotational position. Fig. 2 depicts a DC Servo motor.

Fig. 2. Servo Motor (DC)

G. Power Supply

A power supply is an electrical device that stores and transmits electrical energy in order to convert it into a various forms of energy. A power supply or battery delivers the necessary electric energy to power the load at the needed voltage, current, and frequency. The Battery is shown in fig 3.

Fig. 3. 5 V 1500 mAh Battery

IV. METHODOLOGY

A systematic methodology is used with the ultimate goal of creating a fully functional human following load carrier in mind. This project takes a top-down, decentralized approach. There are numerous stages to the project. Step-by-step procedures were followed, starting with various literature surveys and then questioning about their concerns. After determining the most common issues, the problem statement was written. A solution was suggested for the aforementioned issues, and the best solution was chosen. The main concept for the design has been chosen, and numerous existing publications and patents have been evaluated.

A significant concept was raised following an examination of the numerous publications. As a result, a conceptual design with estimated parameters was developed. A detailed study was conducted after the necessary softwares and electronics were selected, and accurate parameters were established. After that, the detailed design was modelled and used to create prototypes. Finally, the prototype was put to the test using hand gestures to produce the desired response in physical environments. Fig. 4 depicts the methodology that was developed.

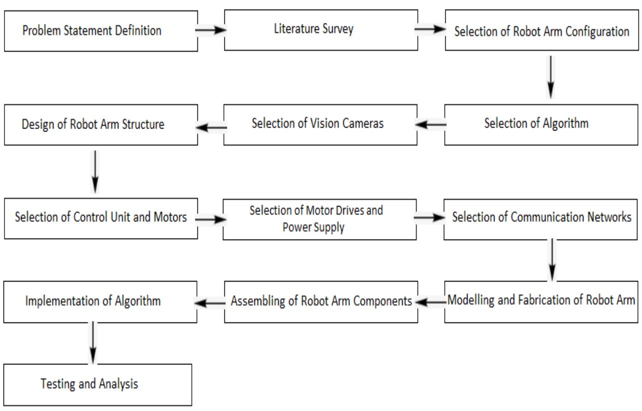

Fig. 4. Process flowchart

V. CONCEPTUAL DESIGN

The project is primarily concerned with software architecture, with the software being implemented through interfacing it with hardware for physical testing and analysis. The algorithm's creation and implementation may control any type of articulated robot for any industrial application. The programming languages utilized are Python and embedded C. The Arduino Uno is the microcontroller that operates and controls the actuators. The actuators used in the work are DC Servo motors. A web camera captures the image, which is then processed using a pre-trained machine learning pipeline. The capturing and processing program is written in PyCharm (Python IDE) and is used to acquire data from the image and send it to Arduino. The servo motor is commanded to make the desired gestures. The block diagram of the proposed model of this project is mentioned in the Fig. 5.

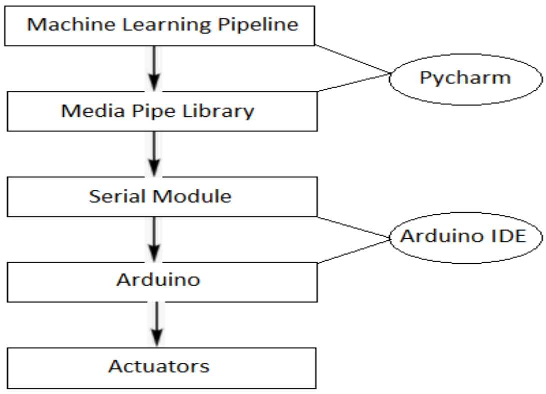

Fig. 5. Block Diagram of the Proposed Model

VI. DETAILED DESIGN

In gesture-based robot arm control, computer vision is employed to control the hardware. As a result, the project focuses mostly on software design, with visual testing conducted using hardware via triggering motors with required motion.

A single-shot detection model for mobile real-time use is built to recognize beginning postures of hands, comparable to the facial recognition model in MediaPipe Face Mesh. Hand identification is a difficult problem since the model must distinguish between occluded and self-occluded hands despite operating with a diverse rsize range of the hands and a wide scale time frame in relation to a captured image.

Regression is utilized to accomplish exact localized key point of 21 3D hand-knuckle points inside the identified hand regions, i.e. direct point predictions, after palm identification across the whole picture.

To get ground truth data, 30 thousand actual pictures were manually tagged with 21 3 dimensional coordinates, as shown in Fig. 6. A best quality artificial hand model was created across different backdrops and mapped to the appropriate 3 dimensional coordinates to better cover the possible hand postures and provide more oversight on the nature of hand geometry. The 21 point hand landmarks are shown in Fig.7.

The media pipe hands algorithm analyzes real-time images and determines which palm belongs to which hand. 21 point coordinates are mapped after palm detection. The top three coordinate points are taken in all fingers from the tip to the bottom, a line is drawn connecting the three spots, and the angle between the three points is measured.

The five-finger angle is recorded in the list values and communicated to the Arduino via serial communication. According to our gesture, the arduino controls the servo motor in real time. The control circuit and actuators are connected as per circuit design. The circuit is designed in TinkerCad software is shown in Fig.8.

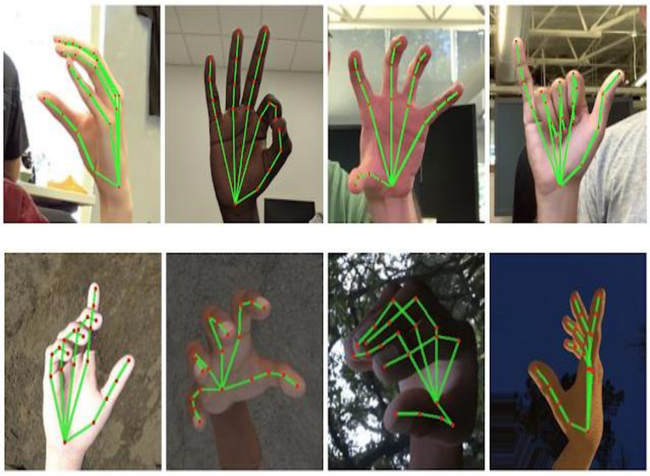

Fig. 6. Various Hand Tracking Coordinated

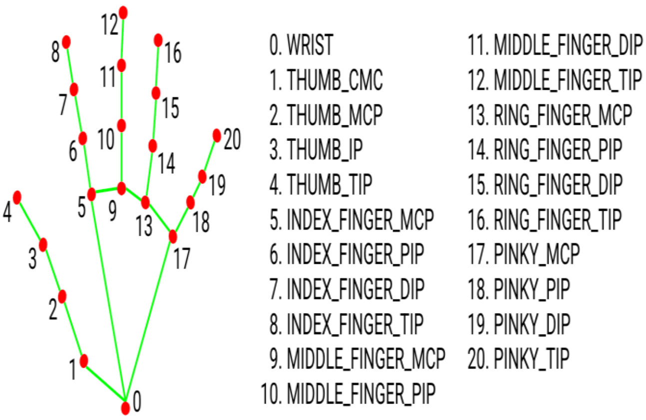

Fig. 7. Hand Landmarks

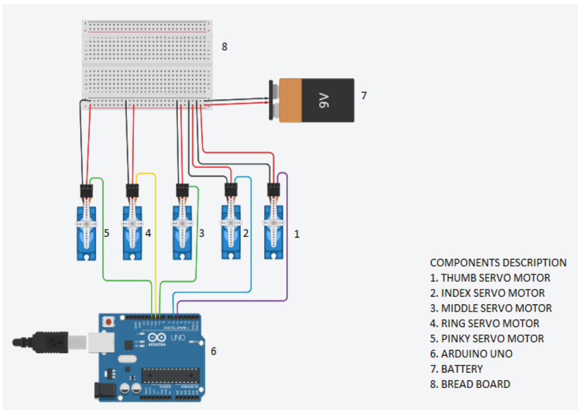

Fig. 8. Circuit Design

VII. IMPLEMENTATION

The code is written in Python and executed in the PyCharm IDE, where a Python script captures and processes video in real time. The processed data is then sent to an Arduino Uno microcontroller, which uses commands from the Arduino IDE to control the servo motors. The base, shoulder, elbow, wrist, hand, and gripper are the six joints that make up a robot arm. The five joints are taken into account to actuate the motors in this project. The human hand's five fingers and palm are employed for recognition and detection. Each of the joints in the robot arm is controlled by the five fingers thumb, index, middle, ring, and little.

The base is actuated by the thumb finger, the shoulder is actuated by the index finger, the elbow is actuated by the middle finger, the wrist is actuated by the ring finger, and the hand is actuated by the little finger.

The hand is recognized using 21 landmarks in the hand. The angle is found by using three points on a finger, and the value is then processed and measured. Through serial module communication, the measured value will be listed and delivered to the Arduino IDE.

The actuation signal pulse for each servo motor will be the processed output for the appropriate hands. All articulated robots with six joints can benefit from this software architecture.

The software is mainly developed in this project to perfectly connect the user and the robot. The final physical hardware is installed in order to evaluate the system's functionality, actuation precision, and practicality.

To acquire a video capture object for the camera, call cv2.VideoCapture(). Set up an unlimited while loop and read the frames using the above-mentioned object's read () function.

To view the frames in the movie, use the cv2.imshow() function. When the user presses a certain key, the loop is broken. The Fig. 6.1 shows Video Capturing.

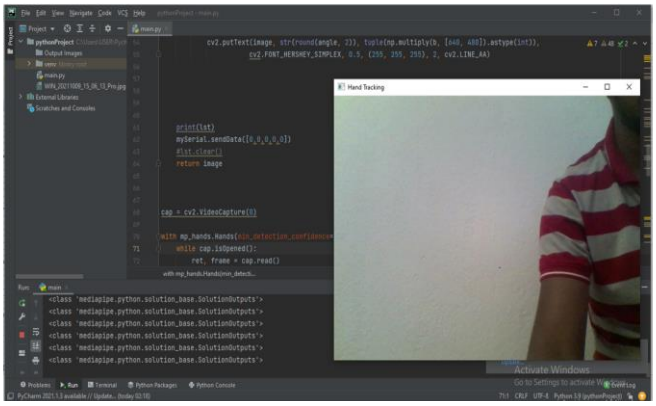

Fig. 9. Video Capturing

To extract actionable data from the video captured, use the libraries described in the prior section to process it. The following command is used to create a function that calculates the angle.

def draw_finger_angles(image, results, joint_list):

The command below is to extract the coordinates of the predetermined points on MediaPipe hands solutions within the function.

a=np.array([hand.landmark[joint[0]].x,hand.landmark[joint[0]].y])

The following command is then used to calculate coordinates in radian values.

radians = np. arctan2(c[1] - b[1], c[0]-b[0]) - np.arctan2(a[1]-b[1], a[0]-b[0])

The angle is then calculated using the formula below.

angle = np.abs(radians*180.0/np.pi)

Using the below instruction, this function is used to provide a list of angle data to Arduino. The Fig. 10 shows processing the Real-time Video.

mySerial.sendData(lst)

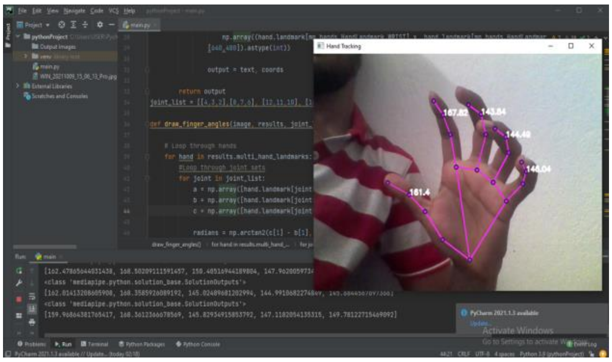

Fig. 10 Processing the Real-time Video

Data is sent from Python to Arduino through serial communication. It is a serial module that allows access to the serial port.

Creating a serial object and assigning configurations to it (port number, baud rate, digits sent per value)

mySerial: SerialObject=SerialObject("COM6", 9600, 1)

Below is the command used to send the desired data to Arduino. Lists of value sent through serial communication are shown in Fig. 11.

mySerial.sendData()

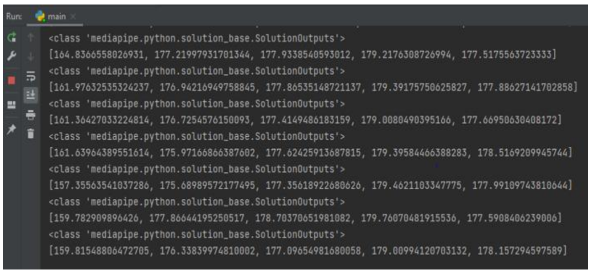

Fig. 11.   Lists Values sent to Serial Communication

Hardware must be interfaced with software to view physical movements and actuation. The data transfer from the computer to the Arduino Uno board takes place when the Arduino Uno board is connected to the computer through USB communication cable. Since the Arduino Uno has six PWM pins, the servo motors may be attached directly to the board. The 9V Battery provides the necessary power to operate the servo motor. Breadboard and jumper wires are used to make the connections. The Fig. 12 shows the installation of hardware components.

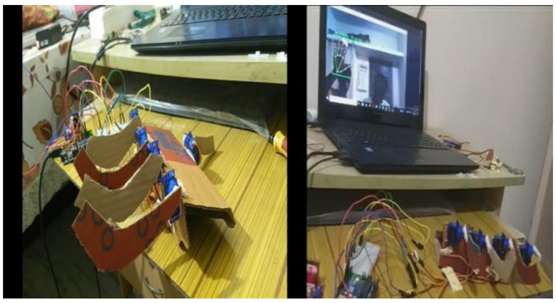

Fig. 12 The Installation of Hardware Components

VIII. TESTING

All of the components are connected during the testing phase, and the Arduino board's USB serial connection is connected to the computer. The PyCharm IDE and Arduino IDE are configured to run the program script. All of the joints in the program worked as predicted, and the motor actuation was able to deliver the appropriate output. Fig. 13 shows processed gesture action with angle values. The output from the hardware is shown in Fig. 14. The Table 1 shows test results.

INPUT / OUTPUT		FINGERS				
TRIALS		THUMB	INDEX	MIDDLE	RING	LITTLE
TRIAL 1	INPUT VALUE	180	160	160	170	180
	ACCEPTABLE VALUE	170-180	150-170	150-170	160-180	170-180
	OUTPUT VALUE	178.2	165.8	164.6	175.6	176.4
	ERROR	-1.8	+5.8	+4.6	+5.6	-3.6
TRIAL 2	INPUT VALUE	160	180	180	10	160
	ACCEPTABLE VALUE	150-170	170-180	170-180	0-20	150-170
	OUTPUT VALUE	165.8	177.3	178.9	17.2	150.3
	ERROR	-5.8	-2.7	-1.1	+7.2	-9.7
TRIAL 3	INPUT VALUE	150	100	60	40	120
	ACCEPTABLE VALUE	140-160	90-110	50-70	30-50	110-130
	OUTPUT VALUE	155.2	95.8	66.2	42.6	118.8
	ERROR	-5.2	-4.2	+6.2	+2.6	-1.2

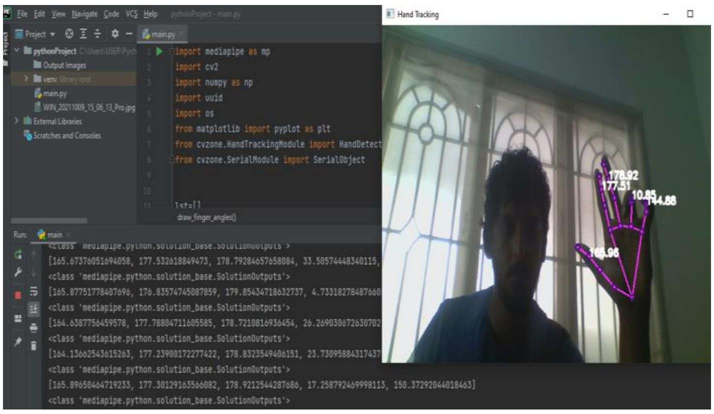

Fig. 13. Processed Gesture Action with Angle Values

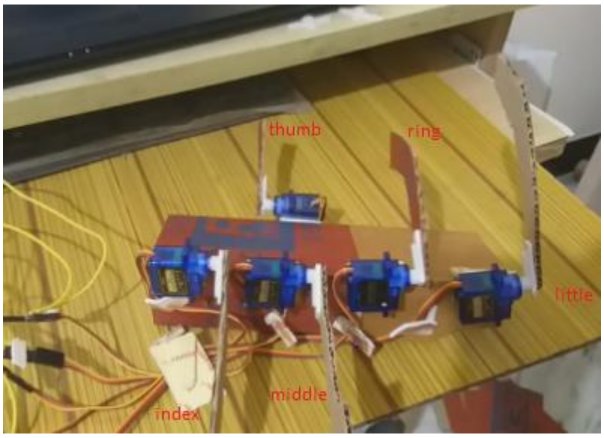

Fig. 14. Output from the Hardware

IX. APPLICATIONS

Human involvement in hazardous environments can be avoided if this project is successfully implemented in the market with high precision.

The Gesture Based Robot Arm Control is intended to be included as an integral part of the project. In future, a whole arm can be controlled by mimicking a human arm using gestures. This project can be implemented in real world applications like bomb diffusing, segregating hazardous wastes, etc.

X. CONCLUSION

Gesture Based Robot Arm Control using OpenCV has been implemented and developed in this project. The interfacing of software and hardware is done and the desired output is achieved according to the given input. This project shows the accurate and feasible approach to control the robot arm using gestures. More packages have been installed to increase the efficiency of the actuations performed.

REFERENCES

1. Poltak Sihombing, Rifky B. Muhammad, Herriyance, Elviwani Elviwani, "Robotic Arm Controlling Based on Fingers and Hand Gesture", IEEE Xplore, 14 August 2020.
2. Yanmin Zhu, Zhibo Yang, Bo Yuan, "Vision Based Hand Gesture Recognition", IEEE Xplore, 27 May 2013.
3. PramodKumar, Pisharady, Martin Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review", ELSEVIER, 1 November 2015.
4. Sakshi Sharma, Shubhank Sharma, Piyush Yadav, "Design and Implementation of Robotic Hand Control Using Gesture Recognition", International Journal of Engineering Research & Technology(IJERT), 04 April 2017.
5. Ss, Dheeban & Dhanasekaran Velayutha Rajan, Harish & A, Hari & Marimuthu, Prasanna & N, Senthil Kumar, "Gesture Controlled Robotic Arm", The International Journal Of Science & Technoledge, 2016.