ISSN: 3107-3883 (Online)

Global E-Journal of Social Scientific Research
Vol. 1| Issue 7| July 2025| pp 8-21

DESIGN AND DEVELOPMENT OF A MACHINE LEARNING-BASED GESTURE-CONTROLLED 3D PRINTED ROBOTIC HAND USING COMPUTER VISION

Author : Ankur Bhargava
University School of Automation & Robotics
Guru Gobind Singh Indraprastha University, Delhi, India
ankurgsb21@gmail.com

Author : Ajay K.S. Singholi
University School of Automation & Robotics
Guru Gobind Singh Indraprastha University, Delhi, India
ajay.s@ipu.ac.in

Author : Deepti Chhabra
Department of Mechanical Engineering
Indira Gandhi Delhi Technical University for Women, Delhi, India
deepti2226@gmail.com

Source: Global E-Journal of Social Scientific Research,
Vol. 1. Issue 7, July 2025, Page Nos. 8-21
Published by: Global Center for Social Dynamic Research

ABSTRACT

This paper introduces a 3D-printed humanoid robotic hand made from a bio-based plastic material, PLA (Polylactic Acid), controlled through human-machine collaboration utilizing machine learning and computer vision algorithms. The system is designed to replicate real-time actions through gesture recognition, a non-invasive technique that enables a wide range of real-time interactions. A Kinect sensor is employed due to its multiple features, allowing for precise and versatile user interaction. The paper suggests using a non-invasive 3D sensor to regulate manipulation in real time. The Kinect sensor can find objects in three dimensions and recognise gestures in a realistic way. Using skeletal data from the Kinect, the status of the user's right arm is captured and sent to the 3D-printed robotic hand. Mimicking human movements in this way is an effective method for controlling robotic operations. The system is calibrated to recognize specific gestures, accurately identifying the user's hand state and angles, streamlining the process and enhancing overall efficiency.

Keywords: 3D Printing, PLA, Computer Vision, Kinect Sensor, Gesture Recognition.

I. INTRODUCTION

Robotics is a rapidly growing field poised to significantly impact society in the coming years. In a variety of real-time and industrial applications, robotic arm manipulation has grown in popularity. Robotic arms are widely used in industrial settings for tasks like cutting, welding, and loading and unloading machines [1]. Outside of the manufacturing floor, they are essential in high-risk, precision-demanding settings like handling radioactive materials, disposing of bombs, and carrying out intricate medical procedures like neurosurgery and cardiology [2]. These robotic systems can be controlled intuitively and in real time thanks to gesture recognition, which offers a strong, non-invasive technique that increases their usability and accessibility in a variety of contexts.The Kinect sensor is advantageous due to its numerous features, enabling precise and versatile user interaction [3]. The authors of this study propose a non-invasive 3D sensor for real-time manipulation control. The Kinect sensor can detect objects in three dimensions and enhance real-time gesture recognition [4]. A robotic hand receives information about the user's right arm's position and motion based on skeletal data collected by the Kinect. Angles are calculated and transmitted to a microcontroller via HC-05 paired Bluetooth modules (Master and Slave). Mimicking human movements with the robotic hand is an effective and beneficial method for performing various operations. The robotic hand, designed to resemble a human hand, is 3D-printed using a 3D printer. Signals from the Arduino are used to actuate the servos controlling the robotic hand. This system can be scaled for automation at various levels, achieving precision in real-time environments. The ability to control the robot from a distance makes it perfect for real-time surgery and bomb disposal, both of which put the safety of people first [2][5].

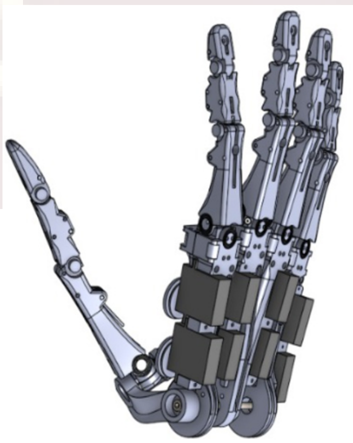

Fig. 1 Robotic Arm basic 3D design

II. DESIGN AND DEVELOPMENT OF HAND

The proposed 3D printed robotic hand consists of four fingers, a thumb, a forearm, and a palm, all of which are 3D printed. Along with the thumb, all four fingers are constructed with the links joined together by pins. The palm also has the links which connect the forearms, thumbs, and fingers to it, with the grooves which allow the tendons to pass from the actuators, which are presented in the forearm, and also consists of different parts which are joined perfectly. A tendon-driven mechanism, which consists of the wires steering over the pulleys, was used. In the overall mechanism, a series of joints support the driving force via pulleys [6].

PLA (Polylactic Acid) is used as a printing material, which is a thermoplastic aliphatic polyester made from tapioca roots, corn starch, etc., which are renewable resources. It is biodegradable in nature and very easy to use compared to ABS (Acrylonitrile Butadiene Styrene) material, as it works in lower temperature ranges, but its lifespan is shorter, and it is pretty fragile.

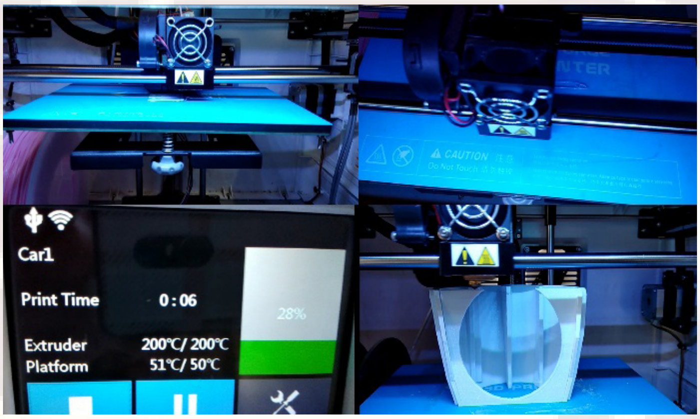

Fig. 2 Construction of different parts of the arm using a 3D Printer

3D Printing technology allows users to develop and manufacture their own prototypes. It is easy to develop new industrial prototypes and applications; more specifically, these printers have improved the 3D model constructions. This robotic hand allows incorporating modifications and updates to improve the design and construction. 3D printing is a faster and simpler technique that can overcome the costs related to the manufacturing.

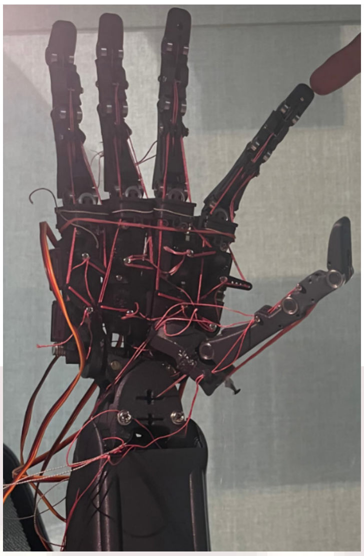

Fig. 3 The final assembled Robotic hand

III. GESTURE RECOGNITION

This section explains how to wirelessly transmit data collected from a Kinect sensor after adjusting the delay and threshold values to standardise the data. The following are necessary for data acquisition: Computer Vision System Toolbox, Simulink Coder, Microsoft Visual Studio, and Microsoft Windows SDK 7.1 for integrating the Kinect sensor with MATLAB [7].

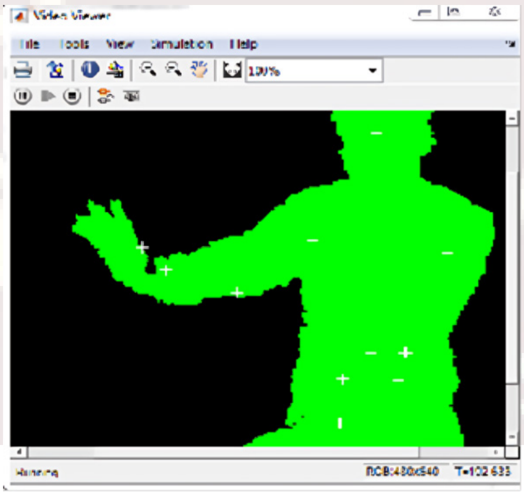

Fig.4. Video viewer’s RGB data display

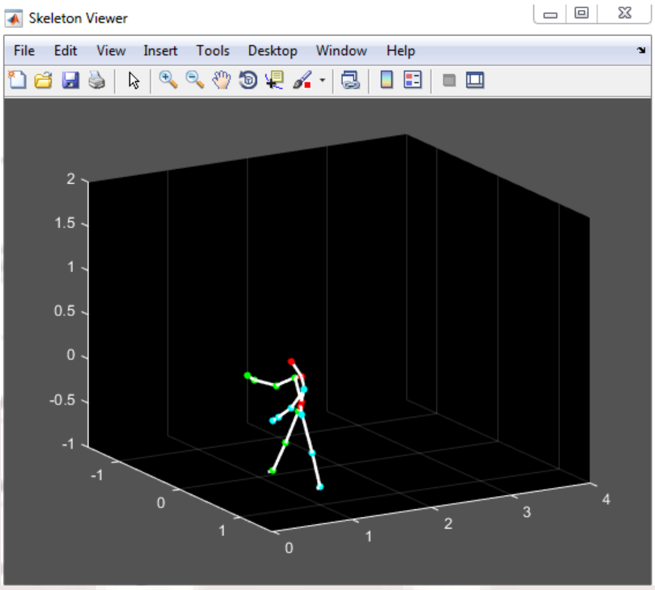

Fig.5. Data visualisation in the skeleton viewer

In this work, human hand tracking is implemented using a Kinect sensor. It is a very effective and straight forward method to coupled man-machine interface efficiently. To control the robotic hand operations in the real-time environment, the human right hand must be recorded in the Kinect, and mapping is done. The main advantage of using Kinect is that there is no need for proper lighting conditions, as it uses the IR light to take the image and sense it after reflection from the object. The hand is controlled by appropriate angles calculated using basic trigonometric equations, i.e., inverse cosine function is used in this case. Simple algorithms are proposed to take data from the real-time environment and send it to the designed system.

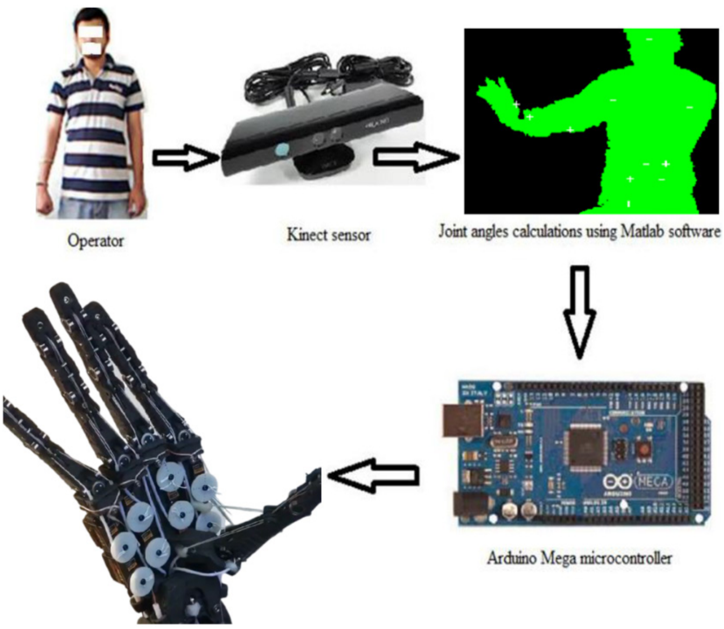

Fig.6. Experimental framework overview

Fig.7 Infrared data view in the infrared viewer with hand gesture recognition

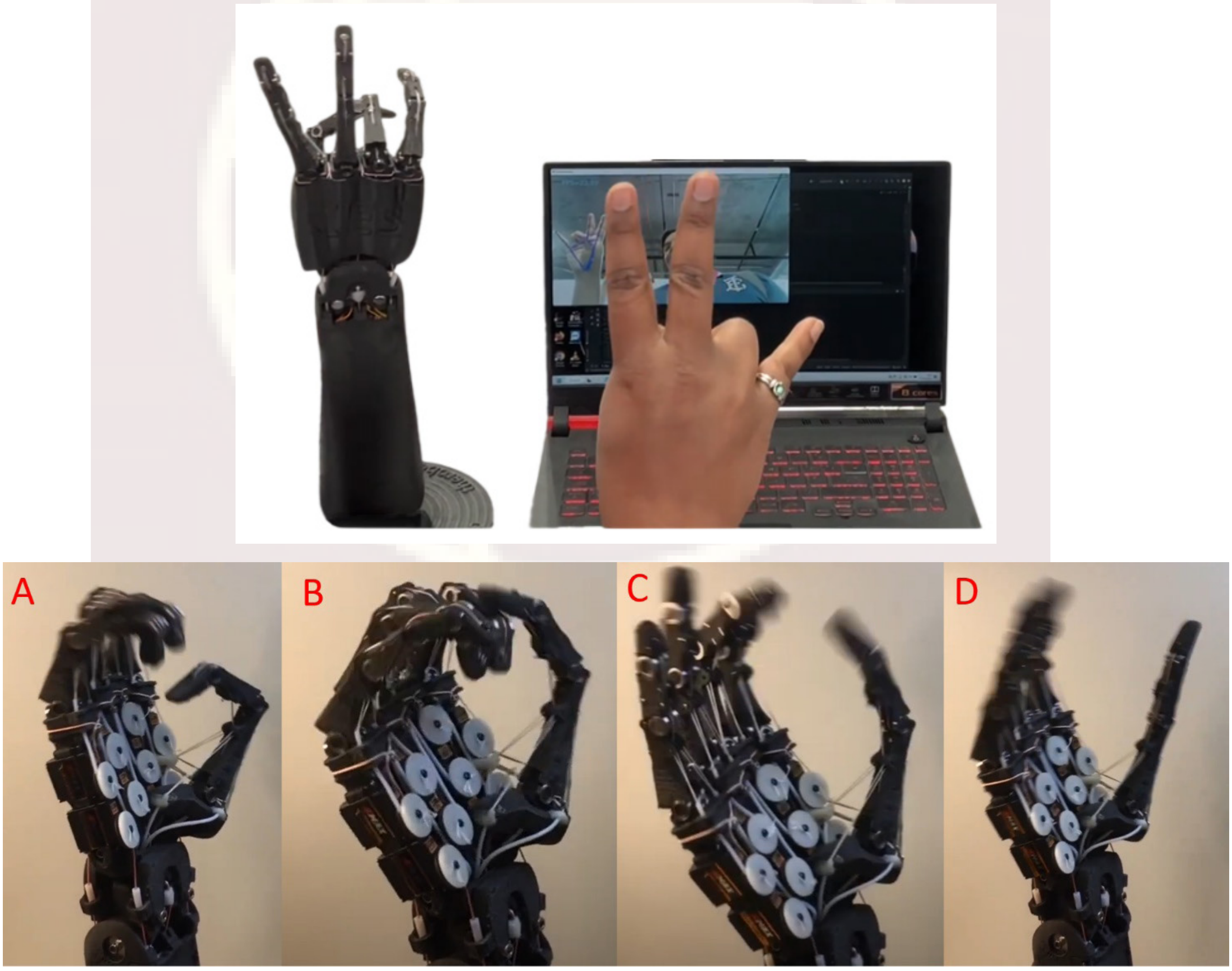

Fig. 8 Controlling of robotic hand using gestures

IV. FRAMEWORK

The operation of the robotic system consists of four separate steps. The first step is to locate and identify skeletons. Using the Kinect sensor, we can then determine the state of the right hand. The second section involves calculating the hand movement's joint angles, which involves figuring out the robot's degree of rotation using vector algebra. In the third section, the microcontroller receives binary values representing the analogue angles converted from their analogue values via connections between the master and slave Bluetooth modules [8]. In addition, the fourth step involves activating the servomotors with the correct delay and specifying the angle data ranges using Arduino programming.

A. Vision Subsystem

A USB to TTL converter, a Bluetooth Controller module, and a PC running Matlab Simulink are the three main parts of a vision system. The Kinect 3D camera is the third part. Kinect keeps track of everything the user does as they move around. The Skeletal Viewer, which is one of the Simulink Natural Interaction Devices (NID) blocks, gives you the X, Y, and Z axis data you need to map the real world process. There are also NID depth, NID image, NID motion tracking, and NID skeleton blocks. Using simulated functional blocks to separate right-hand-related data makes it possible to figure out the angles between different joints, or the angles that different joints make with each other.

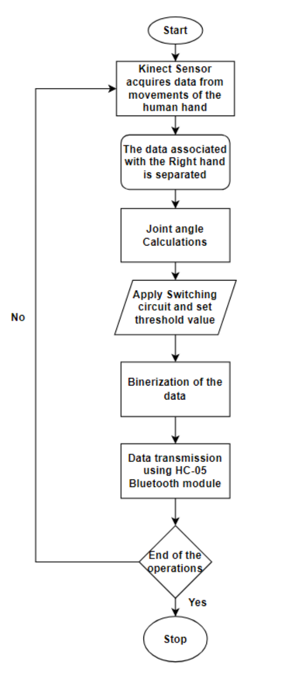

Fig.9 Flow diagram of the vision subsystem

B. Motion Subsystem

An Arduino microcontroller, a robotic hand, an external power supply, and a Bluetooth Slave module make up a motion subsystem. The motion subsystem gets data from the vision subsystem when it sends it to the off-site location using Bluetooth Master-Slave modules. The Arduino microcontroller controls the servos in the robotic hand, and the agent module sends data to the Arduino one at a time. The servomotors need an outside voltage source to work.

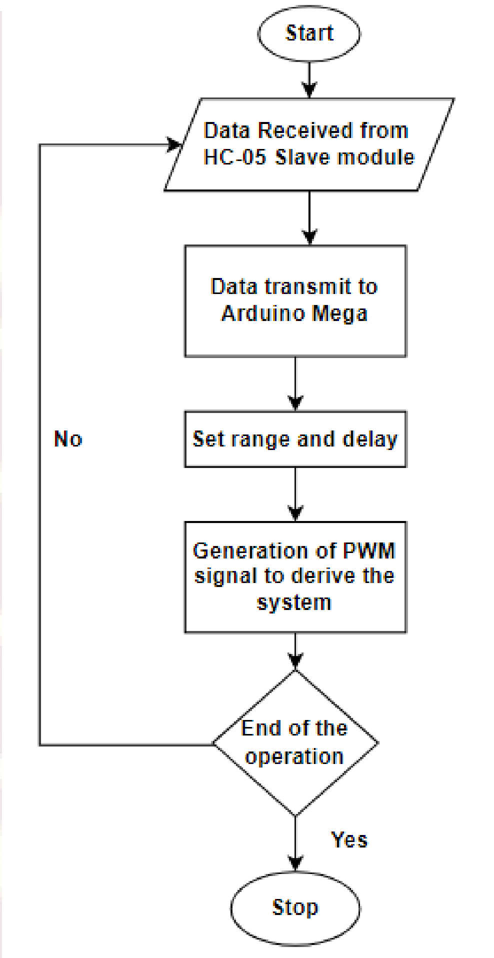

Fig. 10 Flow diagram of the motion subsystem

V. SENSOR AND MICROCONTROLLER DESCRIPTION

A. Kinect Sensor

An RGB camera, a 3D depth sensor (transmitting and receiving), and four microphone channels make up the Microsoft Kinect for Windows, a device that facilitates natural interaction. Using the Kinect with MATLAB, we can detect people, track skeletal data, and realize many novel applications in robotics.

Fig. 11 Schematic diagram of the Kinect Sensor

A depth camera, the Kinect sensor uses infrared light to create a depth picture, which then records the object's distance.The conventional color image is difficult for the computer to understand, whereas a depth image is much easier for the computer to understand. Using the color pixels it is difficult to differentiate between the pixels information provided by the image. The conventional camera gives information about how different things look, but where things are detected is given by the depth camera [9]. In the above figure, the first number shows an infrared projector and receiver, an IR camera. The grids of infrared dots are transmitted by the infrared projector, which is invisible to us, but using an IR camera, an image is captured.

We can detect people and quickly locate their body parts utilising the skeletal information that Kinect offers. With the use of the depth imaging data, a software program may infer the user's skeleton's position, down to the individual joints of the bones that connect them [10]. The IR data, which is obtained using a Kinect sensor has limitations also i.e., any object that is less than or closer to the Kinect's calibration distance will push the dots in one direction out of the position, and thus the displacement of the joints is used to figure out the objects present in that part of the respective scene. In the second limitation, a black shadow is present at the edge of the objects in the IR image, as the Kinect infrared projection dots do not reach that part of the scene. The second camera, a low-quality colour webcam, is attached to the Kinect and is placed at a specific distance from the infrared camera. As a result of the Kinect's infrared camera capturing depth information, the colour image is aligned; thus, any adjustments to the colour image are dependent on the depth data. There is also a possibility of color in the 3D image, which is created using the depth data information, i.e., we can easily create 3D scans or virtual environments with realistic color [11]. There is also another interesting feature in the Kinect that it can move up and down. Inside the Kinect plastic base, a small motor is placed with a series of gears, which enables the Kinect to tilt its camera up and down, but up to a limit of 30 degrees [12], as shown by number three. In addition to the camera and motor, the Kinect has four other sensors, which are microphones shown by number four. Now, a question arises only one microphone is used to capture the sound, so why are there four microphones? The answer is that the Kinect not only captures the sound but also locates that sound within the room [12]. Machine Learning involves the computer learning what the operator is trying to perform without any explicit programming. Pattern recognition research is still at the heart of machine learning. When dealing with the Kinect 3-D sensor, the Random Forest method is employed. Random Forest is an ensemble learning algorithm that includes classification, regression, and other tasks by constructing decision trees. Thus, many body gestures can be sampled, and a training set may be established [13]. The Random decision tree algorithm used in the Kinect can be understood as follows.

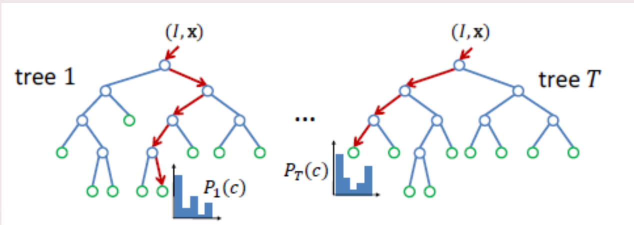

Fig. 12 Machine Learning base decision tree

A decision tree is illustrated in the graphic above. Nodes that are divided are shown in blue, while nodes that are leaves are shown in green. Split nodes make decisions, and leaf nodes display the results. It is important to note that the algorithm's output is a probability of landing up in a particular leaf node.

Propose a set of splitting candidates (split nodes), each of which has two factors on which it depends. The first factor is the "Feature Parameter" and the second is the "Threshold". Dividing the dataset into left and right subplots is the objective. The two subsets are complementary to one other. Determine which branching node yields the most informative result. Before recursing to the left or right, see if the node with the highest gain and lowest tree depth is less than a maximum [13].

B. Arduino MEGA

A board based on the AVR microcontroller, the Arduino integrated development environment (IDE), the programming language, and the Arduino boot loader make up the open-source electronics prototyping platform known as Arduino. Open source hardware describes Arduino because its schematic is available to everybody. Arduino is a platform for building interactive applications that aims to make complicated projects easier to manage. The microcontroller is typically programmed using C and C++ programming language features. Since it is open to everyone, anybody can follow the standard schematic and make their own version of the Arduino board. As an open-source integrated environment, the Arduino IDE is compatible with all Arduino boards and allows users to program them.

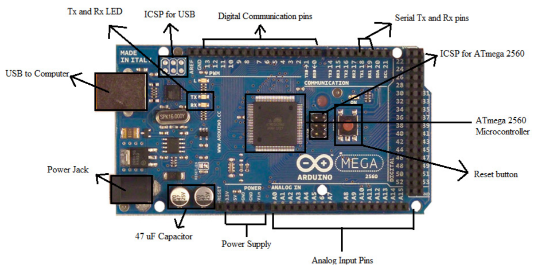

Fig. 13 Schematic diagram of Arduino Mega 2560

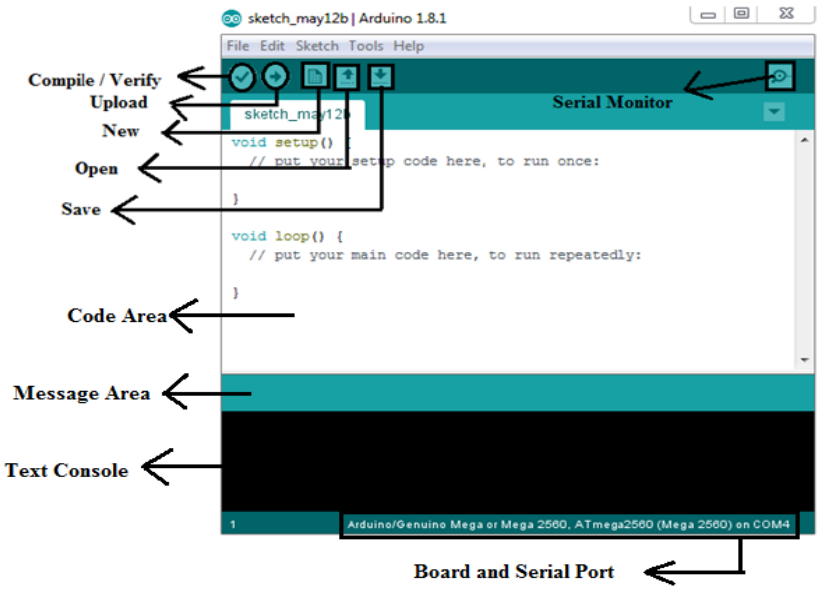

Fig. 14 Layout of the Arduino IDE

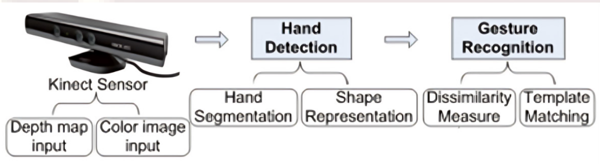

Fig. 15 Framework for Hand Gesture recognition System

C. Bluetooth Modules Pairing

One Bluetooth module must be set as MASTER and the other left as slave in order for them to link [14]. The default state of all HC05 modules is that they are SLAVEs. Modifying the module to conform to the system's actions is possible with the help of AT commands. The slave is configured using an Arduino microcontroller. The process begins with uploading code to the microcontroller, followed by the connection of the HC05 module. Bypassing the boot loader of the microcontroller allows the Arduino to act as a USB-UART converter. Disconnect the microcontroller's USB power source after uploading the sketch to the Arduino board. Then, connect the HC05 Slave and HC05 Master modules in the following order.

VI. CONCLUSIONS

This paper presents an accurate and effective hand gesture recognition system and uses the Kinect sensor as an input device. For hand detection, both the color and depth information are obtained, which provides a robust solution for real-time problems. A 3D-printed robotic hand system that can be controlled in real-time utilising computer vision and machine learning is the fundamental objective of this work. The goal is to solve real-time issues while building it.

This study also comes up with a way for both people and robots to move the robotic hand around freely, which will make it much easier for people and machines to work together. Bluetooth master and agent modules send the calculated joint angles to the Arduino Mega controller. We can see that the system can control the robotic hand with a Kinect sensor by describing how it works in different situations using the user's right hand. It is important to use the right kinematic modelling method, which could be inverse kinematics, forward kinematics, or the D-H parameterisation method. It is also important to be able to control the trajectory and the joints separately. To get the same result no matter what happens to real-time parameters like joint angles and joint coordinates, use estimation with a Kalman filter, an extended Kalman filter, or another method. A haptic interface could make the procedures easier and safer by giving users a better sense of the force they are using to move things around.

ACKNOWLEDGEMENT

The joy and euphoria that accompany the successful completion of the work would be incomplete without the mention of the special person whose guidance and encouragement made my effort successful. I am deeply indebted to my guides whose help, stimulating suggestions, and encouragement helped me throughout to make my effort successful.

REFERENCES

1. R. Rahimi, C. Shao and M. Veeraraghavan, "An Industrial Robotics Application with Cloud Computing and High-Speed Networking", First IEEE International Conference on Robotic Computing, 2017.
2. Yue Ai, Bo Pan, Guojun Niu, Yili Fu et al. "Master-slave Control Technology of Isomeric Surgical Robot for Minimally Invasive Surgery" , International Conference on Robotics and Biomimetics

Qingdao, China, December 3-7, 2016.

3. Shiying Sun, NingAn et al. “Human Recognition for Following Robots with a Kinect Sensor”, International Conference on Robotics and Biomimetics Qingdao, China, December 3-7, 2016.
4. Ming Zeng, Zhengcun Liu et al., “Motion Capture and Reconstruction Based on Depth Information Using Kinect”, 5th International Congress on Image and Signal Processing (CISP 2012).
5. Scott Devine, Karen Rafferty and Stuart Ferguson, “Real-time robotic arm control using hand gestures with multiple end effectors”, UKACC 11th International Conference on Control (CONTROL) Belfast, UK, 31st August to 2nd September, 2016.
6. Bhargava A, Suhaib M, Singholi AKS. An omnidirectional mecanum wheel automated guided vehicle control using hybrid modified A* algorithm. Robotica. 2025;43(2):449-498. doi:10.1017/S0263574724001954
7. Hairina Mohd Jais, Zainal Rasyid Mahayuddin, Haslina Arshad, “A Review on Gesture Recognition Using Kinect “The 5th International Conference on Electrical Engineering and Informatics 2015 August 10-11, 2015, Bali, Indonesia.
8. How Bluetooth technology works, Core system architecture http://www.bluetooth.com/Bluetooth/Learn/.
9. Shiying Sun, NingAn et al.”Human Recognition for Following Robots with a Kinect Sensor”, International Conference on Robotics and Biomimetics Qingdao, China, December 3-7, 2016.
10. Bhargava, A., Suhaib, M. & Singholi, A.S. A review of recent advances, techniques, and control algorithms for automated guided vehicle systems. J Braz. Soc. Mech. Sci. Eng. 46, 419 (2024). https://doi.org/10.1007/s40430-024-04896-w
11. Ming Zeng, Zhengcun Liu et al.,”Motion Capture and Reconstruction Based on Depth Information Using Kinect”, 5th International Congress on Image and Signal Processing (CISP 2012).
12. A. Bhargava, A. S. Singholi and M. Suhaib, “Design and Development of a Visual–SLAM based Automated Guided Vehicle,” 2023 IEEE Fifth International Conference on Advances in Electronics, Computers and Communications (ICAECC), Bengaluru, India, 2023, pp. 1-7, doi: 10.1109/ICAECC59324.2023.10560331.
13. J. Shotton, T. Sharp, A. Kipman, et al., “Real-time human pose recognition in parts from single depth images”, Communications of the ACM, vol. 56, no. 1, p. 116, 2013.

14. A. Bhargava, A. S. Singholi and M. Suhaib, “A study on design and control of Omni-directional mecanum wheels based AGV system,” 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-6, doi: 10.1109/ICCCNT56998.2023.10306665.