

# IoT based Low-cost Posture and Bluetooth Controlled Robot for Disabled and Virus Affected People

Tajim Md. Niamat Ullah Akhund<sup>1</sup>  
Department of CSE  
Daffodil International University  
Dhaka, Bangladesh

Mosharof Hossain<sup>2</sup>  
Department of CSE  
Daffodil International University  
Dhaka, Bangladesh

Khadizatul kubra<sup>3</sup>  
Department of CSE  
Daffodil International University  
Dhaka, Bangladesh

Nurjahan<sup>4</sup>  
Department of ICT  
Bangabandhu Sheikh Mujibur Rahman  
Digital University, Bangladesh

Alistair Barros<sup>5</sup>  
School of Information Systems  
Queensland University of Technology  
Brisbane, Australia

Md Whaiduzzaman<sup>6</sup>  
School of Information Systems  
Queensland University of Technology  
Brisbane, Australia

**Abstract**—IoT-based robots can help people to a great extent. This work results in a low-cost posture recognizer robot that can detect posture signs from a disabled or virus-affected person and move accordingly. The robot can take images with the Raspberry Pi camera and process the image to identify the posture with our designed algorithm. In addition, it can also take instructions via Bluetooth from smartphone apps. The robot can move 360 degrees depending on the input posture or Bluetooth. This system can assist disabled people who can move a few organs only. Moreover, this system can assist virus-affected persons as they can instruct the robot without touching it. Finally, the robot can collect data from a distant place and send it to a cloud server without spreading the virus.

**Keywords**—Internet of Things (IoT); Raspberry Pi; Pi camera; Pi robots; hospital robots; posture recognition robots

## I. INTRODUCTION

The Fourth Industrial Revolution (or Industry 4.0), a term coined by Klaus Schwab, refers to the industrial and manufacturing activities automation through using emerging technologies. Robotics as well as the Internet of Things (IoT) is a key element among such modern technologies. In recent times, robots and robotics is the part and parcel of modern science. Robots are working in lots of sectors including the dangerous, monotonous as well as restricted areas where there is a huge risk to human lives. It is a gift of modern artificial Intelligence. People implemented different types of robots for different works. IoT is an ecosystem where things have a unique identifier, embedded with sensors, and actuators, and can send or receive data over the internet. A fusion of both IoT and robotics or in individual manner, these advanced technologies are being used pervasively in many sectors, for example, smart home, agriculture- paddy cultivation [1], smart campus, smart city, environmental weather monitoring, industry automation-poultry farm automation [2], hotel management [3], health areas- patient management [4] as well as taking care of patient by providing nursing facilities [5], remote sensing and monitoring, virus affected and disabled people management [6]. However, the interaction between humans and robots is

still a challenging task. Apart from verbal communication, humans use non-verbal signs like postures and gestures to communicate with each other. Non-verbal communication is an instinctive way of communication. It is done by facial expression, gestures/postures, gait, or head nods. It is noteworthy that there is a slight difference between gestures and postures. For example, hand gesture refers to a dynamic state of hand movements whereas hand posture refers to the static state of the hand. In this work, we have proposed an IoT-based system to command a robot with Bluetooth through mobile phone and hand postures, especially beneficial for physically challenged as well as virus-affected people. Apart from this, another system is integrated to measure the body temperature of the person through a contact-less IR temperature sensor and another sensor to measure the temperature and humidity of the room. The collected data will be stored in a secure thingspeak cloud server. Here, hand posture is used as an input to the robot and according to the input, the robot is designed to move towards 360 degrees of movement to left/right/forward/backward and to stop. This robot takes hand posture as input by a pi camera, then converts the posture into commands and performs accordingly.

The objectives of this work are as follows:

- 1) People can call the robot by hand postures and Bluetooth command to give the robot 360 degrees of movement. As this robot is controlled in two ways, namely, hand posture taken by a pi camera and also mobile Bluetooth, the system is reliable because if one type of controlling fails, the other type may work.
- 2) This robot may work in virus-affected areas with virus-affected people without making harm to healthy people eventually helping to stop community transmission.
- 3) The robot can collect data and send them to a cloud server for future monitoring.

The rest of the paper is arranged as follows: Section II discusses the related works associated with this paper, Section III

demonstrates the system model, algorithms, circuit diagrams and flowcharts of the work, Section IV shows the experiment results, comparisons of our work with some other works, limitations and finally Section V provides the concluding remarks and future scope of this work.

## II. LITERATURE REVIEW

Researchers explored several approaches to detect hand postures in their works. In literature, the recognition systems are designed based on numerous methods such as deep-learning, 3D Model, depth, skeleton, motion, appearance as well as color [7]. Boyali et al. [8] proposed six different hand gesture and posture (1- Fist, 2- Hand Relax, 3- Fingers Spread, 4- Wave In, 5- Wave Out, 6-Double Tap) recognition system with an accuracy of 97% after receiving electromyography (EMG) signal which is acquired by eight myo armband attached to a person's arm. Here, sparse subspace clustering (SSC) and collaborative representation based classification (CRC) are used to train and recognize patterns. Nguyen et al. [9] demonstrated a hand posture recognition framework consisting of hand detection through traditional hand detector viola jone using haar-like and cascaded adaBoost, low level feature extraction on hand region through image conversion as well as pixel-based extraction, a combination of three kernel descriptors (KDES), namely, gradient, pixel value and texture KDES for hand representation on HSV, RGB, lab color channel and finally multi class support vector machine (SVM) for classification. They have reported an average of 97.3% accuracy on the NUS-2 data-set and 85% on their data-set within 1m to 3m distances. IoT based systems and robots are helping mankind in remote monitoring [10], remote sensing [11], disabled patient management [12], virus affected people management [13] and so on in all times [14] in secure way [15]. Embedded systems and IoT are helping hospitals [16], Agricultural Systems [17], energy generation [18], electronic voting [19] and bio-metric[20] security systems. Akhund et al. [6] reported a 97.9% success rate to recognize hand gestures of disabled as well as virus-affected people to operate the robot. They have developed a robotic agent consisting of an MPU6050 accelero-meter gyroscope sensor, Arduino nano, 433KHz radio wave receiver, L293D motor driver IC. The sensor is responsible for tracking the movement of the hand. Alam et al. [21] designed a hand gesture controlled robot which can recognize five different gestures (left, right, forward, backward, stop) with an accuracy rate of 93.8%. The system uses MPU 6050 module, Arduino nano, HT12E encoder IC, decoder chip HT12D, RF transmitter and receiver. Fakhrurroja et al. [22] developed a system to control electronic devices like fan, light, TV, AC of a smart home through the detection of hand gestures. In their study, kinect v2 sensor is used to track nine different hand state combinations along with raspberry pi and relay module. They have achieved 87% accuracy to detect the gestures. Chen et al. [23] investigated compact CNN named EMGNet to classify hand gestures after analyzing surface electromyography (sEMG) signals obtained through myo armband. The model is tested on Ninapro DB5 Dataset and validate on the myo Dataset. The model can identify 5 different hand gestures with 99.81% recognition accuracy. Maharani et al. [24] implemented two traditional machine learning algorithms namely SVM with directed acyclic graph and K means clustering for the classification of hand postures

receiving through Kinect v2. The four gestures (forward, right, left and stop) are used to operate the bioloid premium robot consisting of IC 74LS241N, Arduino mega, kinect v2. They tested hand gesture recognition from the distance of 2, 3 and 4 meters where the range of the body slope was 45, 0 and 45 degrees. They have reported 95.15% accuracy in 10 ms and 77.42% accuracy in 4.45 ms using SVM and k-means clustering algorithm respectively. Meghana et al. [25] designed a system to move a robot towards forward, backward, right and left direction through identifying hand gesture as well as voice especially for the people who are unable to see or hear. The hardware used in the system is MPU6050, L293D driver, LCD, HC05 Bluetooth module, Arduino uno. Abed et al. [26] experimented with a vision-based hand gesture recognition system that can identify five different hand gestures to move a mobile robot towards forward, backward, left, right as well as stop. Raspberry Pi camera module is used to track hand gestures. The hardware that is used to process and provide commands to the robot along with a pi camera is Raspberry Pi 3 model B, L298N motor driver, power supply, camera board, 5 inches 800\*480 resistive HD touch screen, rover 5 two-wheel drive platform. They have used Open cv library of python programming language to perform the task. The system showed 98% recognition accuracy. Su designed a 10 types of hand gestures recognition system based on depth vision and surface EMG signals. Here, leap motion controller collects depth vision data finally labeled by hierarchical K means clustering. In addition, myo armband is used to receive and transmit EMG signals. Then, preprocessing is done by a band-stop filter, and a band-pass filter. After feature extraction, finally, multiclass SVM is used to classify the signal. Adithya et al. [27] employed a deep CNN with rectified linear unit (Relu) activation function to automatically recognize hand gestures. They have trained and tested the model using 2 datasets namely the National University of Singapore (NUS) and the American fingerspelling dataset. The model showed  $94.7 \pm 0.80\%$  accuracy,  $94.96 \pm 1.20\%$  precision,  $94.85 \pm 1.30\%$  recall,  $94.26 \pm 1.70\%$  f1-score in case of NUS dataset and  $99.96 \pm 0.04\%$  accuracy,  $99.96 \pm 0.04\%$  precision,  $99.96 \pm 0.04\%$  recall,  $99.96 \pm 0.04\%$  f1-Score in case of American fingerspelling dataset. Chansri et al. [28] presented a skin color technique consisting of an RGB camera along with Raspberry Pi to control hand gestures. They have experimented on the American sign language dataset with 12 gestures and achieved 90.83% accuracy. Mondal at el. [29] developed a temperature detector product module for measuring the body temperature of covid 19 affected patients with 98% accuracy. The hardware part consists of LoLin NodeMCU V3 with ESP8266 module, DS18B20 temperature sensor probe, passive buzzer, LED and flat vibrator motor whereas programming is done in Arduino IDE. This system creates an alarm using a buzzer when the temperature exceeds 100.4°F or 38°C temperature. To store the data, they have used ThingSpeak cloud server. IoT based systems are helping people to make cheap irrigation systems [30], Seamless Microservice Execution [31], Big Data Processing [32] and blockchain technology [33]. IoT is helping lung cancer diagnosis [34]. Fog computing plays a vital role for augmenting resource utilization [35]. Cloud computing is helping to analyse people sentiment [36] and dynamic task scheduling algorithms can make it more efficient [37]. So, we should make proper use of IoT and robotics in medical science.

### III. METHODOLOGY AND SYSTEM MODEL

#### A. Requirements

To implement the prototype we used Raspberry Pi 3B+, pi camera 5MP and python 3 programming language for the image processing part. The robot motion controlling part consists of a robotic chassis, breadboard DC gear motor, motor driver L298N, battery, wires, battery charger. Instead, the Arduino UNO Micro-controller, HC-05 Bluetooth Sensor, DHT11 sensor, IR temperature sensor MXL90614, Node MCU ESP8266, Wi-Fi, ThingSpeak cloud server and C++ programming language for Bluetooth controlling and data sending part.

#### B. System Model

The diagram of the system methodology of the robot movement is depicted in Figure 1. For recognising the posture with image processing we have used Raspberry Pi and Pi camera. The robot can be controlled via Bluetooth also. We could include Bluetooth module with the Raspberry Pi too but we have done it with Arduino uno to make the system more durable and easy. If Raspberry Pi fails any time to detect posture then Arduino will work with following the instructions from mobile.

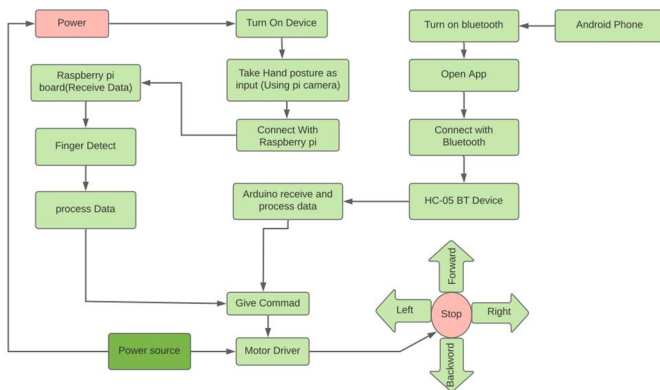


Fig. 1. System Methodology

1) *Posture Detection Part:* Posture detection is done with image processing using python programming language on Raspberry Pi 3 B+. Pycharm IDE community version was installed on Raspberry Pi. At first, we need to install two library function numpy and opencv. Before plugin library function we updated the pip version 2 to version 3. Then (while (cap.isOpened)) executes video capturing. We crop correct image size. Then we convert color in gray scale and create window for mapping the picture. After that conditions to find motion from the window will be executed with our image processing algorithm. If the motion found then motion pixel detect. We also plot detected motion in window map to simulate the detection in computer screen before applying it in Raspberry Pi and robot. Then the motion data and detected posture creates movement command for the robot to move 360 degrees. The movement commands goes to the GPIO ports of Raspberry Pi and send signal to the motor driver and the motor driver moves the 2 motor of the robot. Finally the robot moves by following the posture signal. If the motion not found then the robot will stop movement and

again executes new video reading command. Algorithm for Movement controlling with posture recognition by Raspberry Pi and image processing is as mentioned in Algorithm 1.

#### Algorithm 1 Posture Recognition with Raspberry Pi and Image Processing

```

1: import: cv2; numpy ← np; math; cap =
  cv2.VideoCapture(0)
2: while (cap.isOpened()) do
3:   ret, img = cap.read()
4:   cv2.rectangle(img, (300, 300), (100, 100), (0, 255, 0),
  0)
5:   crop_img = img[100 : 300, 100 : 300]
6:   grey = cv2.cvtColor(crop_img, cv2.COLOR_BGR2GRAY)
7:   value = (35, 35)
8:   blurred = cv2.GaussianBlur(grey, value, 0)
9:   thresh1 = cv2.threshold(blurred, 127, 255, cv2.T.B.I+
  cv2.T.O)
10: ▷ T.B.I=THRESH.BINARY.INV ▷ T.O=THRESH.OTSU
11:   cv2.imshow('Thresholded', thresh1)
12:   (version) = cv2.version.split('.')
13:   if version == '3' then
14:     image, contours, hierarchy =
  cv2.findContours()
15:   else if version == '4' then
16:     contours, hierarchy = cv2.findContours()
17:     cnt = max(contours, key=lambda x:
  cv2.contourArea(x))
18:     x, y, w, h = cv2.boundingRect(cnt)
19:     cv2.rectangle(crop_img, (x, y), (x + w, y + h), (0,
  0, 255), 0)
20:     hull = cv2.convexHull(cnt)
21:     drawing = np.zeros(crop_img.shape, np.uint8)
22:     cv2.drawContours(drawing, [cnt], 0, (0, 255, 0), 0)
23:     cv2.drawContours(drawing, [hull], 0, (0, 0, 255),
  0)
24:   end if
25:   hull = cv2.convexHull(cnt, returnPoints=False)
26:   defects = cv2.convexityDefects(cnt, hull)
27:   count.defects = 0
28:   cv2.drawContours(thresh1, contours, -1, (0, 255, 0), 3)
29:   for i in range(defects.shape[0]) do
30:     s, e, f, d = defects[i, 0]
31:     start = tuple(cnt[s][0])
32:     end = tuple(cnt[e][0])
33:     far = tuple(cnt[f][0])
34:     a = math.sqrt((end[0] - start[0]) ** 2 + (end[1] -
  start[1]) ** 2)
35:     b = math.sqrt((far[0] - start[0]) ** 2 + (far[1] -
  start[1]) ** 2)
36:     c = math.sqrt((end[0] - far[0]) ** 2 + (end[1] -
  far[1]) ** 2)
37:     angle = math.acos((b ** 2 + c ** 2 - a ** 2) / (2
  * b * c)) * 57
38:   end for
39:   if angle <= 90 then
40:     count.defects += 1
41:     cv2.circle(crop_img, far, 1, [0, 0, 255], -1)
42:     cv2.line(crop_img, start, end, [0, 255, 0], 2)
43:   end if
  
```

```

44:  if count.defects == 1 then
45:      str = "Detect 1 Finger"
46:      cv2.putText(img, str, (50, 50), cv2.F.H.S, 0.5, (0,0,
255), 2)
47:          ▷ F.H.S = FONT.HERSHEY.SIMPLEX
48:  else if count.defects == 2 then
49:      str = "Detect 2 Finger"
50:      cv2.putText(img, str, (50, 50), cv2.F.H.S, 0.5, (0,0,
255), 2)
51:  else if count.defects == 3 then
52:      str = "Detect 3 Finger"
53:      cv2.putText(img, str, (50, 50), cv2.F.H.S, 0.5, (0,0,
255), 2)
54:  else if count.defects == 4 then
55:      str = "Detect Entire Hand"
56:      cv2.putText(img, str, (50, 50), cv2.F.H.S, 0.5, (0,0,
255), 2)
57:  else
58:      str = "Detect No Finger"
59:      cv2.putText(img, str, (50, 50), cv2.F.H.S, 0.5, (0,0,
255), 2)
60:      cv2.imshow('Gesture', img)
61:      all.img = np.hstack((drawing, crop.img))
62:      cv2.imshow('Contours', all.img)
63:      k = cv2.waitKey(10)
64:  end if
65:  if k == 27 then
66:      break
67:  end if
68: end while
    
```

Flow chart of Hand posture recognition and finger detection is mentioned in Figure 2.

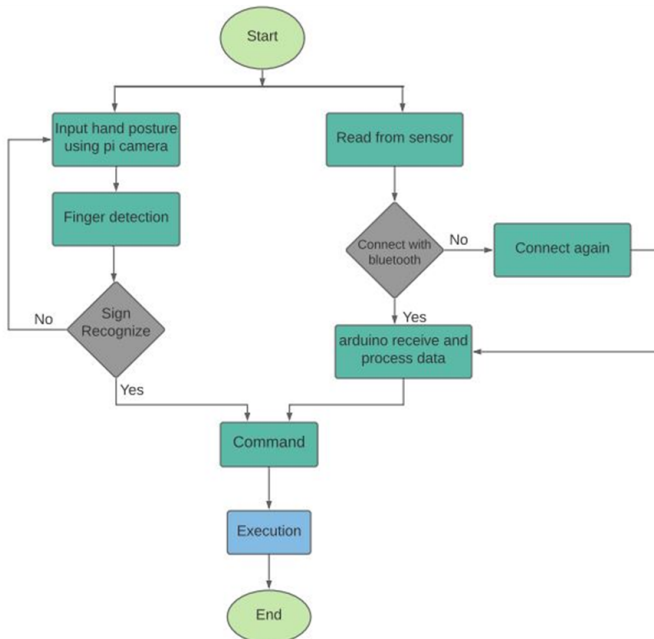


Fig. 2. Flow Chart of Hand Posture Recognition and Finger Detection

The circuit diagram of posture detection and robot controlling with Raspberry Pi is mentioned in Figure 3.

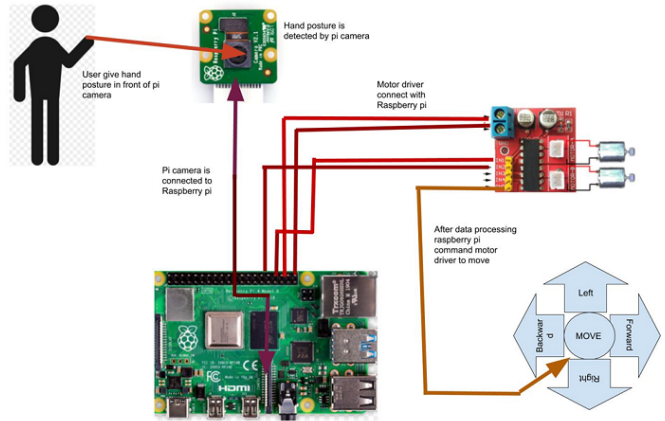


Fig. 3. Circuit Diagram of Posture Detection and Robot Controlling with Raspberry Pi

2) Bluetooth Sensing and Controlling Part: Movement with Bluetooth sensing from mobile phone is another part of this project. We used the Arduino IDE to write the C++ programming language for Arduino UNO Micro-controller. The Arduino will receive the signal sent from mobile phone with Bluetooth sensor HC-05. Android or IOS mobile phones can send any data via any Bluetooth data sender mobile app (we used Bluetooth RC controller available in app store and play store) to HC-05 after pairing. Arduino can receive that data with serial communication (Tx and Rx). After receiving the data in Arduino, conditions for 360 degrees movement will be executed. Then the signals for forward, backward, left, right and stop will be sent to the motor driver L298N. Then the motor driver will move the 2 DC motors of robot by following the commands. Algorithm for Movement Controlling with Bluetooth is mentioned in Algorithm 2.

The circuit diagram of robot controlling with Bluetooth is mentioned in Figure 4.

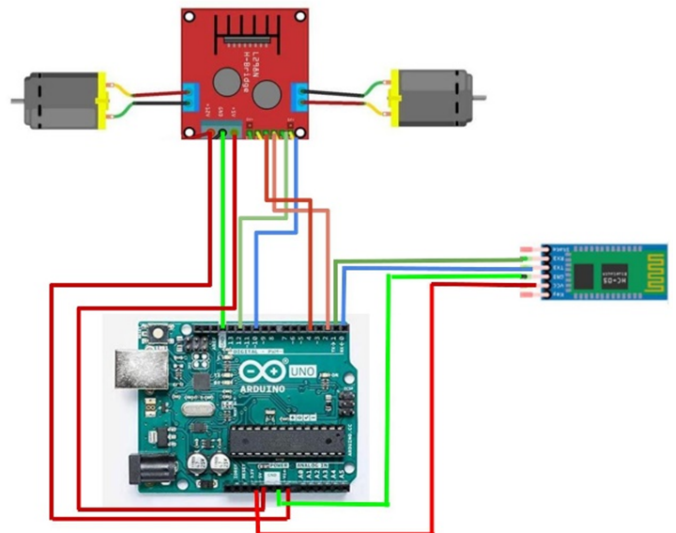


Fig. 4. Circuit Diagram of Robot Controlling with Bluetooth

---

**Algorithm 2** Movement Controlling with Bluetooth

---

```
1: Include Libraries.
2: m11 ← 5; m12 ← 6; m21 ← 9; m22 ← 10; char
   str[2],i;    ▷ Defining Arduino pins for motor driver and
   taking variables.
3: procedure SETUP()                ▷ Setup for Arduino
4:   Serial.begin(9600);
5:   pinMode(m11, OUTPUT);
6:   pinMode(m12, OUTPUT);
7:   pinMode(m21, OUTPUT);
8:   pinMode(m22, OUTPUT);
9: end procedure
10: while Serial.available() do    ▷ Establishing connection
11:   char ch=Serial.read();        ▷ Receiving Bluetooth data
12:   if (ch=='F') then
13:     digitalWrite(m11, HIGH);
14:     digitalWrite(m12, LOW);
15:     digitalWrite(m21, HIGH);
16:     digitalWrite(m22, LOW);    ▷ Going Forward
17:   else if (ch=='B') then
18:     digitalWrite(m21, LOW);
19:     digitalWrite(m22, HIGH);
20:     digitalWrite(m11, LOW);
21:     digitalWrite(m12, HIGH);  ▷ Going Backward
22:   else if (ch=='L') then
23:     digitalWrite(m11, HIGH);
24:     digitalWrite(m12, LOW);
25:     digitalWrite(m21, LOW);
26:     digitalWrite(m22, LOW);   ▷ Going Left
27:   else if (ch=='R') then
28:     digitalWrite(m11, LOW);
29:     digitalWrite(m12, LOW);
30:     digitalWrite(m21, HIGH);
31:     digitalWrite(m22, LOW);   ▷ Going Right
32:   else if (ch=='S') then
33:     digitalWrite(m11, LOW);
34:     digitalWrite(m12, LOW);
35:     digitalWrite(m21, LOW);
36:     digitalWrite(m22, LOW);   ▷ Stop Movement
37:   end if
38: end while
```

---

3) *Data Sensing and Sending Part*: The system will be able to collect data from the patient with IR temperature sensor MXL90614 without touching people and collect the room temperature and humidity with DHT11 sensor. Then it will upload the collected data to a cloud server of ThingSpeak via Wi-Fi. The collected data will be shown in an Oled display and can be monitored from anywhere in the world with that cloud platform. The circuit concept of data sensing and sending part is shown in Figure 5. We used Node-MCU ESP8266 as micro-controller unit for this.

Algorithm for Data sensing and sending with NodeMCU is mentioned in 3.

#### IV. RESULTS

We have got experimental result of this hand posture recognition robot. When it detects one finger it goes forward. When the pi camera detects two fingers of hand then it goes

---

**Algorithm 3** Remote Data Sensing with Node MCU

---

```
1: Include Libraries: SPI, Wire, Adafruit-GFX, Adafruit-
   SSD1306, DHT, ESP8266WiFi, WiFiClient and ThingS-
   peak.
2: oledRESET ← ledBUILTIN; DHTPIN ← D4;
   DHTTYPE ← DHT11; ssid ← wifiName;
   password ← wifiPassword; myChannelNumber ←
   thingspeakChannelNumber; myWriteAPIKey ←
   thingspeakWriteApiKey; dataState ← false;    ▷
   Defining pins and variables.
3: AdafruitSSD1306 display(oledRESET);
4: DHT dht(DHTPIN, DHTTYPE);
5: procedure SETUP()
6:   Serial.begin(115200);
7:   dht.begin();
8:   if (SSD1306LCDHEIGHT != 64) then
9:     print("Height incorrect, please fix
   AdafruitSSD1306.h!");
10:  end if
11:  display.begin(SSD1306.SWITCHCAPVCC, 0x3C);
12:  display.clearDisplay();
13:  display.display();
14:  display.setTextSize(1);
15:  display.setTextColor(WHITE); ▷ making oled display
   ready
16:  WiFi.begin(ssid, password);
17:  while (WiFi.status() != WLCONNECTED) do
18:    delay(500);
19:    Serial.print(".");
20:  end while
21:  print(WiFi.localIP());    ▷ Connect to WiFi network
22:  ThingSpeak.begin(client); ▷ start cloud server
   communication
23: end procedure
24: procedure VOID LOOP()
25:   temperature = dht.readTemperature();
26:   humidity = dht.readHumidity();    ▷ getting the data
27:   display.clearDisplay();
28:   display.setCursor(0, 0);
29:   display.print(temperature, " ");
30:   display.print(humidity);
31:   display.display();    ▷ printing the data in oled display
32:   Serial.println(temperature);
33:   Serial.print(humidity);    ▷ printing the data in serial
   monitor
34:   if (temperature < 100 and humidity < 150) then
35:     if dataState then
36:       ThingSpeak.writeField(myChannelNumber, 1,
   temperature, myWriteAPIKey);
37:       dataState = false;
38:     else
39:       ThingSpeak.writeField(myChannelNumber, 2,
   humidity, myWriteAPIKey);
40:       dataState = true;
41:     end if
42:   end if
43:   delay(15000);    ▷ uploading the data to the cloud
   server
44: end procedure
```

---

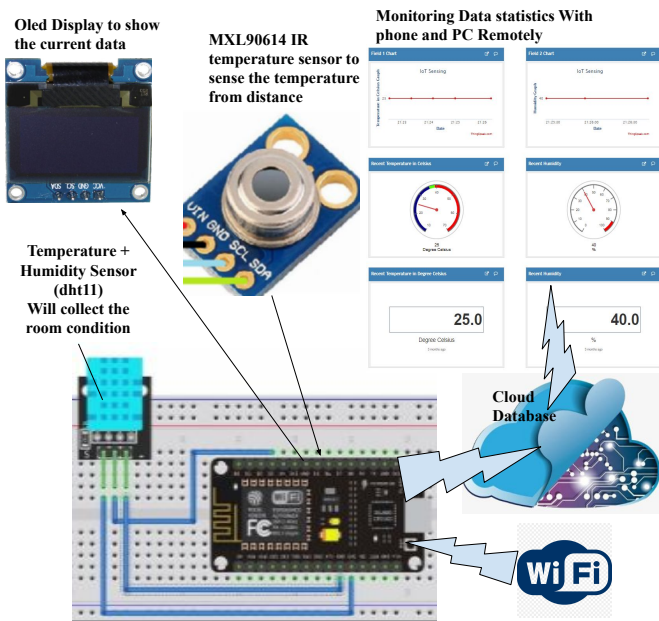


Fig. 5. Circuit Concept of Data Sensing and Sending Part

backward. According to the algorithm, when it detects three finger then the robot move to its right side and after detecting four fingers the robot moves to its left side. Finally when the camera detects five fingers of hand then the robot stops its movement. We use python programming language to detect fingers. Figure 6 shows detection of one finger simulation.

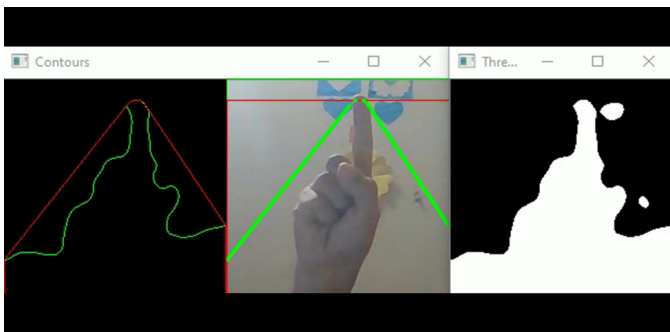


Fig. 6. Detect One Finger

The proposed system can detect two finger successfully. Figure 7 shows our system detect two fingers successfully.

Our system can detect two finger successfully. Figure 8 shows our system detect 3 fingers successfully.

The system can also detect two finger successfully. Figure 9 shows our system detect 4 fingers successfully.

Finally, our system detects five fingers form human hand. Figure 10 shows our system can detect 5 fingers accurately.

The system shows the collected data in the ThingSpeak server (Location: <https://thingspeak.com/channels/739817>). Figure 11 shows Collected data visualisation in cloud server.

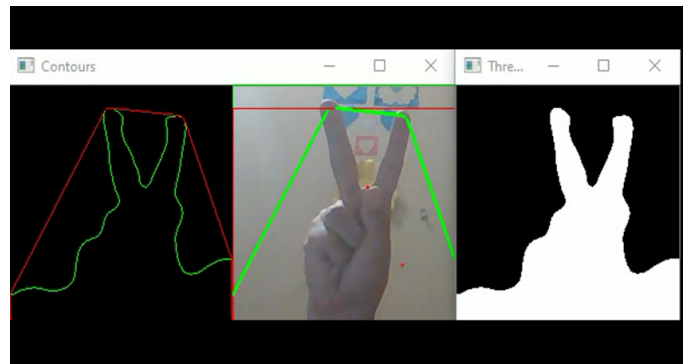


Fig. 7. Detect Two Fingers

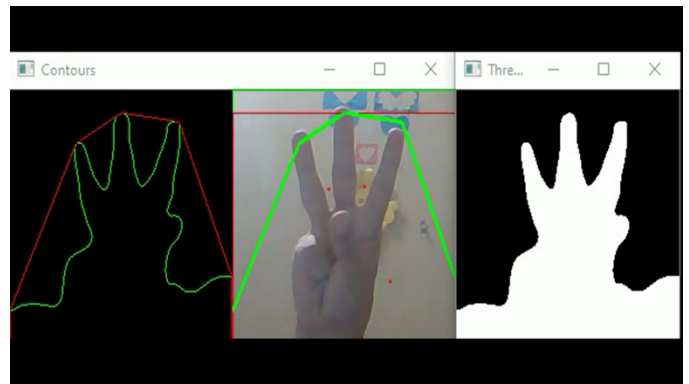


Fig. 8. Detect Three Finger

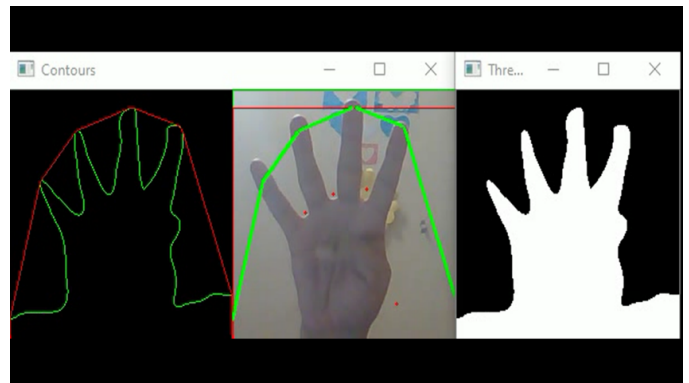


Fig. 9. Detect Four Finger

#### A. Prototype Output

Final view of the Prototype Robot is mentioned in Figure 12, Figure 13 and Figure 14.

#### B. Obtained Features and Results

We tested our system for posture detection, Bluetooth controlling and data sending to cloud server. We got desired results. The features obtained from the prototype are as follows:

- 1) The robot is able to recognize the posture and fingers of a patient and move towards the patient. It showed 95% success rate in 500 tests.

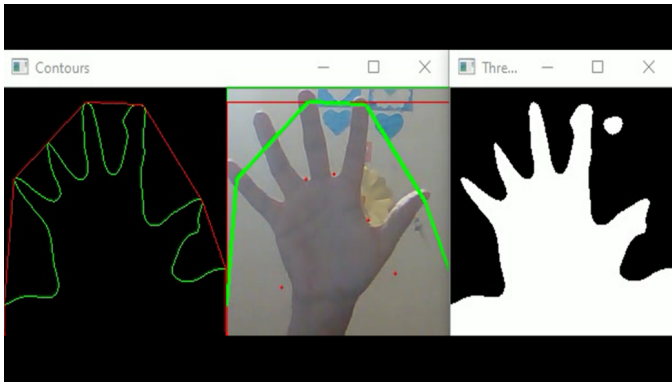


Fig. 10. Detect Five Finger

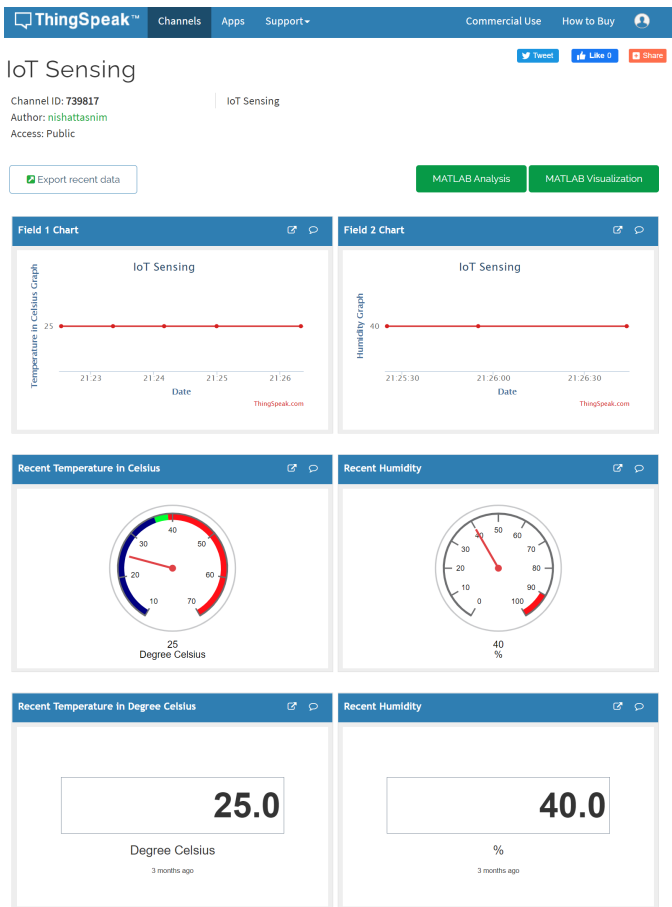


Fig. 11. Collected Data Visualisation in Cloud Server

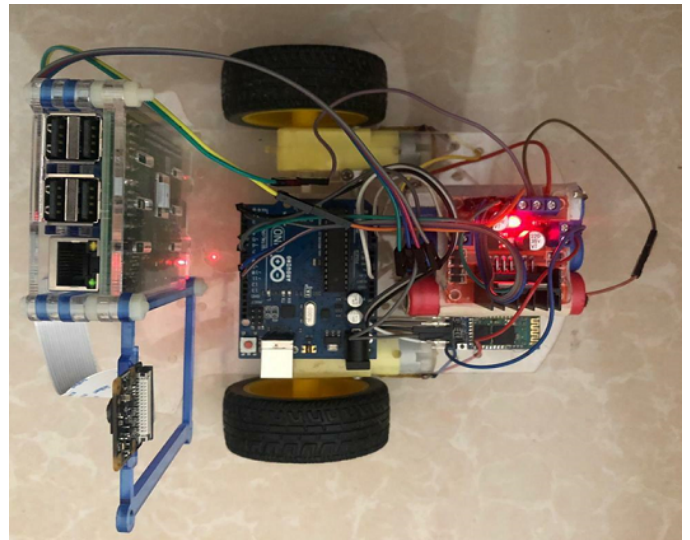


Fig. 12. Final Look (Side 1)

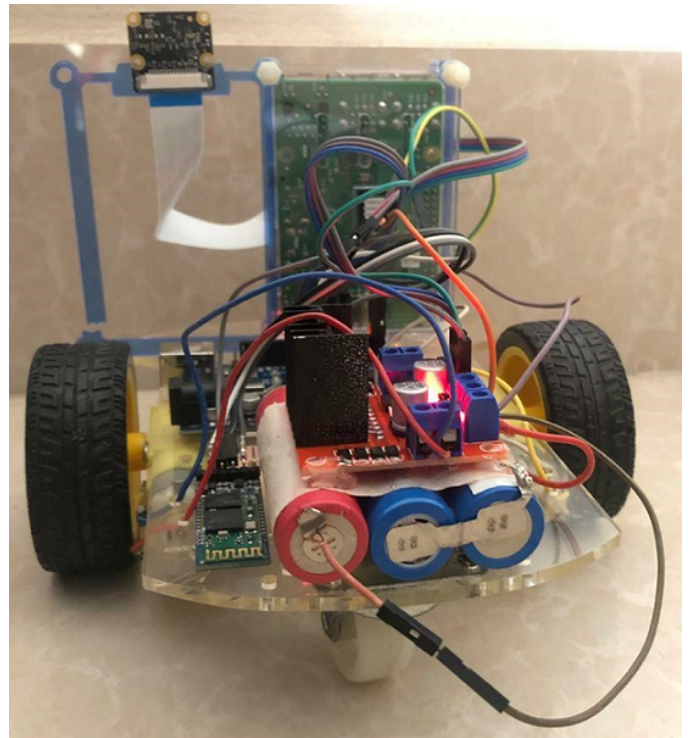


Fig. 13. Final Look (Side 2)

- 2) The robot can move 360 degree by following the instructions from posture. It showed 95% success rate in 500 tests.
- 3) The robot also works using Bluetooth sensor. It showed 98% success rate in 500 tests.
- 4) It can also make movement by the instructions from mobile phones via Bluetooth with 97% success rate in 500 tests.
- 5) The robot successfully collected temperature and humidity data from people and environment and showed

- 6) in oled display with 96% success rate in 500 tests.
- 6) The robot successfully sent the collected data to cloud database with 94% success rate in 500 tests.
- 7) The system is cost effective. The prototype costs less than 100 USD. Where some existing systems may cost more than 500 USD without having all of the features obtained in this work.

C. Limitations

The system has some limitations also which can be mitigated in future development. The limitations we have found

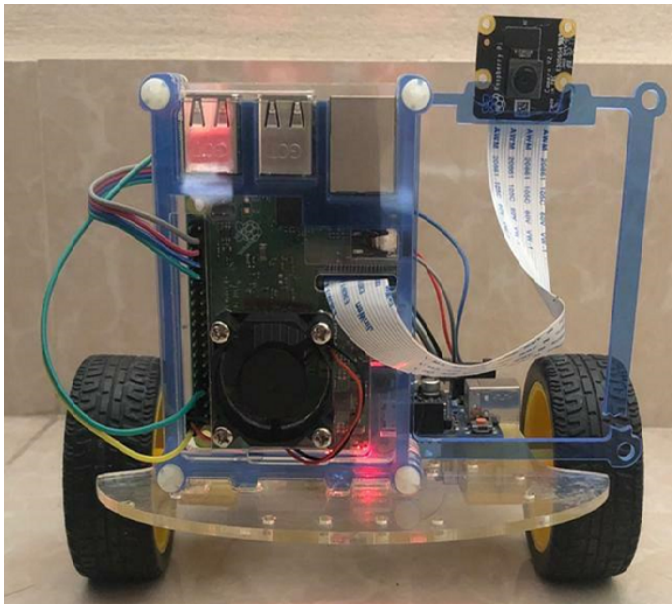


Fig. 14. Final Look (Side 3)

are as follows:

- 1) The system is not water proof. Water may damage the full or partial system.
- 2) For making the system low cost we used low cost sensors that may make incorrect results some time.
- 3) The robot should be maintained carefully. Otherwise it can be broken.

#### D. Discussion

The main objective of this paper is to operate a robot using hand gestures or a mobile phone through Bluetooth and after then, collect data and store them on a cloud server. The process of making hand gestures as input has been performed through an embedded pi camera, processed to recognize correctly as well as respond accordingly. The whole process takes on an average of 1500 milliseconds at the rate of 97% accuracy. Alam et al.[21] designed hand gesture-controlled robot using MPU 6050 gyroscope module and Arduino nano with an accuracy rate of 93.8%. Maharani et al.[24] performed 1080 number of tests in total by 6 different people with four gestures (forward, right, left, and stop), from three distances (2m, 3m, 4m), and at three slopes position (450, 00, -450). According to them, SVM showed superior results than K-means clustering with 95.15% recognition accuracy in 10ms. Hand gestures are used to control the home appliances[22] with the accuracy rate 87% in 3 meter distance. They utilized a Kinect sensor to take hand gestures with three-hand states(open, close, and lasso), and processing is performed through Raspberry Pi. Meanwhile, Xing et al.[38] achieved 83.23% accuracy using a little modified CNN to surface electromyographic signal for hand gestures recognition. Hence, it is evident that our proposed work provides superior the result compared to other existing methods summarised in table I.

Our robot also contains a non-contact temperature sensor MXL90614 which can detect Covid-19 affected patient's body

TABLE I. COMPARISON WITH STATE-OF-THE ART METHOD OF HAND POSTURE RECOGNITION

Method	Accuracy
Alam et al.[21]	93.8%
Fakhrurroja et al.[22]	87%
Nazzi et al.[38]	95.01%
Maharani et al.[24]	95.15%
<b>Our Proposed Model</b>	<b>97%</b>

temperature at a distance and send it to the cloud without wearing any devices in the body like the work of [29]. These above discussion shows the advantages of this work.

#### V. CONCLUSION

In the modern era, robots are engaged in a lot of sectors. This work implemented an IoT-based posture recognizer remote sensing robot. Hospital patients can call the robot with hand posture and smartphone via Bluetooth; then the robot can go to the patient by following posture or Bluetooth command. Then it can collect data with sensors and send them to a cloud database. Therefore, any disabled or virus-affected person can control it and be monitored with this system remotely without affecting any healthy people. We got around 95-97% success rate among all the features. In future work machine learning features can be added to this robot to predict the patient and environmental conditions.

#### ACKNOWLEDGMENT

This research is partially supported through the Australian Research Council Discovery Project: DP190100314, "Re-Engineering Enterprise Systems for Microservices in the Cloud".

#### REFERENCES

- [1] Atia Sultana, Md. Abul Hasan, and Tajim Md. Niamat Ullah Akhund, "An approach to Create IOT based Automated Smart Farming System for Paddy Cultivation," 2019. [Online]. Available: <http://rgdoi.net/10.13140/RG.2.2.13443.04647>
- [2] T. M. N. U. Akhund, S. R. Snigdha, M. S. Reza, N. T. Newaz, M. Saifuzzaman, and M. R. Rashel, "Self-powered IoT-Based Design for Multi-purpose Smart Poultry Farm," in *Information and Communication Technology for Intelligent Systems*, T. Senjyu, P. N. Mahalle, T. Perumal, and A. Joshi, Eds. Singapore: Springer Singapore, 2021, vol. 196, pp. 43-51. [Online]. Available: [http://link.springer.com/10.1007/978-981-15-7062-9\\_5](http://link.springer.com/10.1007/978-981-15-7062-9_5)
- [3] T. M. N. U. Akhund, M. A. B. Siddik, M. R. Hossain, M. M. Rahman, N. T. Newaz, and M. Saifuzzaman, "IoT Waiter Bot: A Low Cost IoT based Multi Functioned Robot for Restaurants," in *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*. Noida, India: IEEE, Jun. 2020, pp. 1174-1178. [Online]. Available: <https://ieeexplore.ieee.org/document/9197920/>
- [4] T. M. Niamat Ullah Akhund, M. J. N. Mahi, A. N. M. Hasnat Tanvir, M. Mahmud, and M. S. Kaiser, "ADEPTNESS: Alzheimer's Disease Patient Management System Using Pervasive Sensors - Early Prototype and Preliminary Results," in *Brain Informatics*, S. Wang, V. Yamamoto, J. Su, Y. Yang, E. Jones, L. Iasemidis, and T. Mitchell, Eds. Cham: Springer International Publishing, 2018, vol. 11309, pp. 413-422. [Online]. Available: [http://link.springer.com/10.1007/978-3-030-05587-5\\_39](http://link.springer.com/10.1007/978-3-030-05587-5_39)
- [5] Tajim Md. Niamat Ullah Akhund, "Study and Implementation of Multi-Purpose IoT Nurse-BoT," 2019. [Online]. Available: <http://rgdoi.net/10.13140/RG.2.2.14674.09924>



- [6] T. M. Niamat Ullah Akhund, W. B. Jyoty, M. A. B. Siddik, N. T. Newaz, S. A. Al Wahid, and M. M. Sarker, "IoT Based Low-Cost Robotic Agent Design for Disabled and Covid-19 Virus Affected People," in *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*. London, United Kingdom: IEEE, Jul. 2020, pp. 23–26. [Online]. Available: <https://ieeexplore.ieee.org/document/9210389/>
- [7] M. Oudah, A. Al-Naji, and J. Chahl, "Hand Gesture Recognition Based on Computer Vision: A Review of Techniques," *Journal of Imaging*, vol. 6, no. 8, p. 73, Jul. 2020. [Online]. Available: <https://www.mdpi.com/2313-433X/6/8/73>
- [8] A. Boyali, N. Hashimoto, and O. Matsumoto, "Hand posture and gesture recognition using MYO armband and spectral collaborative representation based classification," in *2015 IEEE 4th Global Conference on Consumer Electronics (GCCE)*. Osaka, Japan: IEEE, Oct. 2015, pp. 200–201. [Online]. Available: <http://ieeexplore.ieee.org/document/7398619/>
- [9] V.-T. Nguyen, T.-L. Le, T.-H. Tran, R. Mullot, and V. Courboulay, "Hand Posture Recognition Using Kernel Descriptor," *Procedia Computer Science*, vol. 39, pp. 154–157, 2014. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1877050914014410>
- [10] T. M. Akhund, N. Ullah, N. T. Newaz, M. Rakib Hossain, and M. Shamim Kaiser, "Low-cost smartphone-controlled remote sensing iot robot," in *Information and Communication Technology for Competitive Strategies (ICTCS 2020)*. Springer, 2021, pp. 569–576.
- [11] T. Akhund, I. A. Sagar, and M. M. Sarker, "Remote temperature sensing line following robot with bluetooth data sending capability," in *International Conference on Recent Advances in Mathematical and Physical Sciences (ICRAMPS)*, 2018.
- [12] T. M. Akhund, N. Ullah, G. Roy, A. Adhikary, A. Alam, N. T. Newaz, M. Rana Rashel, M. Abu Yousuf *et al.*, "Snappy wheelchair: An iot-based flex controlled robotic wheel chair for disabled people," in *Information and Communication Technology for Competitive Strategies (ICTCS 2020)*. Springer, 2021, pp. 803–812.
- [13] T. Akhund, N. T. Newaz, and M. M. Sarker, "Posture recognizer robot with remote sensing for virus invaded area people," *Journal of Information Technology (JIT)*, vol. 9, pp. 1–6, 2020.
- [14] F. I. Suny, M. R. Fahim, M. Rahman, N. T. Newaz, T. M. Akhund, N. Ullah *et al.*, "Iot past, present, and future a literary survey," in *Information and Communication Technology for Competitive Strategies (ICTCS 2020)*. Springer, 2021, pp. 393–402.
- [15] N. T. Newaz, M. R. Haque, T. M. N. U. Akhund, T. Khatun, M. Biswas, and M. A. Yousuf, "Iot security perspectives and probable solution," in *2021 Fifth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4)*. IEEE, 2021, pp. 81–86.
- [16] A. H. Himel, F. A. Boby, S. Saba, T. M. Akhund, N. Ullah, and K. Ali, "Contribution of robotics in medical applications a literary survey," in *Intelligent Sustainable Systems*. Springer, 2022, pp. 247–255.
- [17] M. Suny, F. Islam, T. Khatun, Z. Zaman, M. Fahim, M. Roshed, M. Islam, R. Jesmin, T. M. Akhund, N. Ullah *et al.*, "Smart agricultural system using iot," in *Intelligent Sustainable Systems*. Springer, 2022, pp. 73–82.
- [18] M. R. Rashel, M. Islam, S. Sultana, M. Ahmed, T. M. Akhund, N. Ullah, J. N. Sikta *et al.*, "Internet of things platform for advantageous renewable energy generation," in *Proceedings of International Conference on Advanced Computing Applications*. Springer, 2022, pp. 107–117.
- [19] M. M. Sarker and T. M. N. U. Akhund, "The roadmap to the electronic voting system development: a literature review," *International Journal of Advanced Engineering, Management and Science*, vol. 2, no. 5, p. 239465, 2016.
- [20] M. M. Sarker, M. A. I. Shah, T. Akhund, and M. S. Uddin, "An approach of automated electronic voting management system for bangladesh using biometric fingerprint," *International Journal of Advanced Engineering Research and Science*, vol. 3, no. 11, p. 236907, 2016.
- [21] M. Alam and M. A. Yousuf, "Designing and Implementation of a Wireless Gesture Controlled Robot for Disabled and Elderly People," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*. Cox'sBazar, Bangladesh: IEEE, Feb. 2019, pp. 1–6. [Online]. Available: <https://ieeexplore.ieee.org/document/8679290/>
- [22] H. Fakhurroja, A. Abdillah, U. Nadiya, and M. Arifin, "Hand State Combination as Gesture Recognition using Kinect v2 Sensor for Smart Home Control Systems," in *2019 IEEE International Conference on Internet of Things and Intelligence System (IoT&IS)*. BALI, Indonesia: IEEE, Nov. 2019, pp. 74–78. [Online]. Available: <https://ieeexplore.ieee.org/document/8980390/>
- [23] L. Chen, J. Fu, Y. Wu, H. Li, and B. Zheng, "Hand Gesture Recognition Using Compact CNN via Surface Electromyography Signals," *Sensors*, vol. 20, no. 3, p. 672, Jan. 2020. [Online]. Available: <https://www.mdpi.com/1424-8220/20/3/672>
- [24] D. A. Maharani, H. Fakhurroja, Riyanto, and C. Machbub, "Hand gesture recognition using K-means clustering and Support Vector Machine," in *2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*. Penang: IEEE, Apr. 2018, pp. 1–6. [Online]. Available: <https://ieeexplore.ieee.org/document/8405435/>
- [25] M. Meghana, C. Usha Kumari, J. Sthuthi Priya, P. Mrinal, K. Abhinav Venkat Sai, S. Prashanth Reddy, K. Vikranth, T. Santosh Kumar, and A. Kumar Panigrahy, "Hand gesture recognition and voice controlled robot," *Materials Today: Proceedings*, vol. 33, pp. 4121–4123, 2020. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S2214785320350343>
- [26] A. Ali A. and R. Sarah A., "Python-based Raspberry Pi for Hand Gesture Recognition," *International Journal of Computer Applications*, vol. 173, no. 4, pp. 18–24, Sep. 2017. [Online]. Available: <http://www.ijcaonline.org/archives/volume173/number4/abed-2017-ijca-915285.pdf>
- [27] A. V. and R. R., "A Deep Convolutional Neural Network Approach for Static Hand Gesture Recognition," *Procedia Computer Science*, vol. 171, pp. 2353–2361, 2020. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1877050920312473>
- [28] C. Chansri, J. Srinonchat, E. G. Lim, and K. L. Man, "Low Cost Hand Gesture Control in Complex Environment Using Raspberry Pi," in *2019 International SoC Design Conference (ISOCC)*. Jeju, Korea (South): IEEE, Oct. 2019, pp. 186–187. [Online]. Available: <https://ieeexplore.ieee.org/document/9027669/>
- [29] M. S. Mondal, K. Roy, and S. Sarkar, "Design and Development of Wearable Remote Temperature Monitoring Device for Smart Tracking of COVID-19 Fever," *SSRN Electronic Journal*, 2020. [Online]. Available: <https://www.ssrn.com/abstract=3735919>
- [30] T. M. Akhund, N. Ullah, N. T. Newaz, Z. Zaman, A. Sultana, A. Barros, and M. Whaiduzzaman, "Iot-based low-cost automated irrigation system for smart farming," in *Intelligent Sustainable Systems*. Springer, 2022, pp. 83–91.
- [31] M. Whaiduzzaman, A. Barros, A. R. Shovon, M. R. Hossain, and C. Fidge, "A resilient fog-iot framework for seamless microservice execution," in *2021 IEEE International Conference on Services Computing (SCC)*. IEEE, 2021, pp. 213–221.
- [32] R. Hossen, M. Whaiduzzaman, M. N. Uddin, M. J. Islam, N. Faruqui, A. Barros, M. Sookhak, and M. J. N. Mahi, "Bdps: An efficient spark-based big data processing scheme for cloud fog-iot orchestration," *Information*, vol. 12, no. 12, p. 517, 2021.
- [33] M. Whaiduzzaman, M. J. N. Mahi, A. Barros, M. I. Khalil, C. Fidge, and R. Buyya, "Bfim: Performance measurement of a blockchain based hierarchical tree layered fog-iot microservice architecture," *IEEE Access*, vol. 9, pp. 106 655–106 674, 2021.
- [34] N. Faruqui, M. A. Yousuf, M. Whaiduzzaman, A. Azad, A. Barros, and M. A. Moni, "Lungnet: A hybrid deep-cnn model for lung cancer diagnosis using ct and wearable sensor-based medical iot data," *Computers in Biology and Medicine*, vol. 139, p. 104961, 2021.
- [35] M. R. Hossain, M. Whaiduzzaman, A. Barros, S. R. Tuly, M. J. N. Mahi, S. Roy, C. Fidge, and R. Buyya, "A scheduling-based dynamic fog computing framework for augmenting resource utilization," *Simulation Modelling Practice and Theory*, vol. 111, p. 102336, 2021.
- [36] M. Mahi, J. Nayeem, K. M. Hossain, M. Biswas, and M. Whaiduzzaman, "Sentrac: A novel real time sentiment analysis approach through twitter cloud environment," in *Advances in Electrical and Computer Technologies*. Springer, 2020, pp. 21–32.
- [37] A. M. R. Mazumder, K. A. Uddin, N. Arbe, L. Jahan, and M. Whaiduzzaman, "Dynamic task scheduling algorithms in cloud computing," in *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*. IEEE, 2019, pp. 1280–1286.

- [38] K. Xing, Z. Ding, S. Jiang, X. Ma, K. Yang, C. Yang, X. Li, and F. Jiang, "Hand Gesture Recognition Based on Deep Learning Method," in *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*. Guangzhou: IEEE, Jun. 2018, pp. 542–546. [Online]. Available: <https://ieeexplore.ieee.org/document/8411908/>