

# A Bi-Pedal Humanoid Social Robot

Desai V N S R Sai Prasad<sup>1</sup>, J. Praveena<sup>2</sup>

<sup>1</sup>Student, Department of EEE, Aditya College of Engineering, Madanapalle, Andhrapradesh, India

<sup>2</sup>Assistant Professor, Department of EEE, Aditya College of Engineering, Madanapalle, Andhrapradesh, India

**Abstract:** A social robot is a self-governing robot that cooperates and speaks with people or different self-governing physical specialists by taking after social practices and principles connected to its part. This definition proposes that a social robot must have a physical exemplification. Alex is a social robot that reacts to the orders entered by the client. It utilizes some inherent summons which are handled at backend by Google cloud Speech API. It is absolutely expand on Raspberry pi. It performs activities like essential walk, reaction to the given charges and so on. There are numerous robots which are expand on different supplies yet fabricating a robot utilizing Raspberry pi have many focal points as it is very adjustable, simplicity of control and so on. This model assists in building the robots with cutting edge highlights like multitasking and furthermore a portion of the elements of OS as we can run working framework in Raspberry pi. It makes utilization of the ideas of IoT and furthermore control the IoT based brilliant items with voice orders. There are many points of interest of this robot like individual help, performing huge figuring. The major diligent work of people can be supplanted. It is absolutely easy to use and responds as per the client. Python dialect is utilized to interpret the guidelines by client to control the developments of the robot and execution in light of the voice charges. As python is profoundly utilized and deciphered dialect, making utilization of it in Raspberry pi to control the equipment and programming is the best.

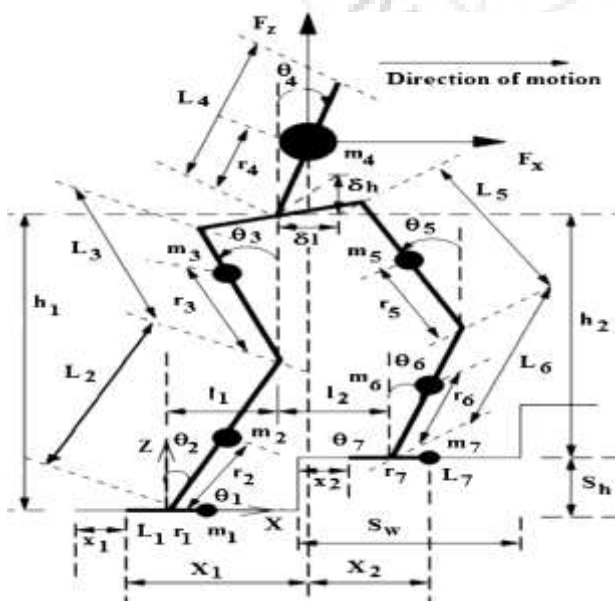
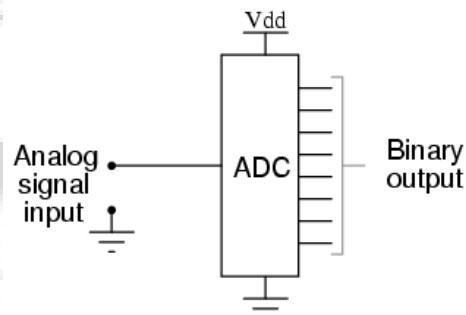
**Keywords:** Internet of Things (IoT), Social Robot, Raspberry pi

## 1. Introduction

### 1.1 Motivation

The main aim of this project is to simulate the basic actions performed by humans like body movements, communication, basic processing etc. As we know, now-a-days there are many robots in the existence, which helps man in working huge tasks like machine designing, Nano material designing etc. Now our task is to build a mini robot which interacts with humans, process the queries given by the humans and finally responding to those queries with a suited answer. It also performs basic human action like walking.

Thus the digital voice is analyzed by the robot, transfers it into its understandable language and processes it. So if he/she asks the robot to say the temperature at that place, it process it the query in the language it was coded, and fetch the result from the global servers for the user.



Consider an example, of a person who is making communication with this robot. The communication with the robot will be in the form of voice. The analogue voice received by the mic of the robot will be converted to digital.

### 1.2 Problem Definition

Though the attempts and the research is done by the robot researchers to match the human intelligence but the result is not achieved. Most of the robots which are existing now still cannot see and are not flexible enough to do multiple functions simultaneously. To achieve the efficient mechanism of the robotics it is very essential to prioritize the ineffectiveness associated in it.

Though the wide use of robots will take way the human jobs and it will increase the unemployment in the society.

The initiation should be done in a systematic order as the use of robotics for every aspect will reduces the jobs for human being. The manufacturing of robots will help in various sectors like military, agriculture, health, education and so on. This will lead to robots as helpers in the work stations with some balance between the what is required and the greed. The society also should encourage for the improvements in

the robotics technology as it will be benefited for the human being and for the various sectors of an economy.

Many tasks which are unable to perform by the human being can be performed with the help of robotics and plays a major role in the war by its operations. The improvement of robotic technology will take away the human jobs so to solve the issues which are related to this the usage of robots should be assigned only for the selected tasks and most preferably should be used where the human is not capable of performing the task.

Recently some robots are developed which uses a screen to display the robots head. Such type of machine is on the borderline of being a robot. If the body only performs the functions as a holder for the screen then it can't be considered as a robot. In the event that the robot has some physical engine and the sensor capacities then it can be considered as a robot.

### 1.3 Objective of the Project

The main objective is to develop and enhance the science of Robotics and to educate about the responsible usage of a robot. Understanding and Identification of the risk in artificial intelligence is a very essential task at hand. This helps in resolving the problems at hand. Careful research and should have more dedication to resolve the programming errors and the cyber-attacks. To achieve the quality of software more attention should be played by the technology companies and the technology industries. Everything which have been created till now in this world is the result of the intelligence. The Artificial Intelligence will augment and empowers the human intelligence. We can help this human civilization if we are successful in keeping technology beneficial.

- To eradicate poverty and hunger from human race.
- Satisfies all the human needs.
- Provides Interaction with the user to fulfil the user requirements through voice commands.

### 1.4 Limitations of The Project

Creation of robot requires more cost as they are very complex. The software in it should be frequently updated to cope up with the needs of the changing environment and the need for the robots to be smarter day by day. It requires long time to recover the codes and to re-install all the software when it undergoes a severe damage. The robots do not have feelings like human beings as it performs what is programmed in it and cannot make the argument of right or wrong. They are unable to take their own decisions when they come across a situation which is not familiar to them so they either process incorrectly or breakdown in such situations.

- Manufacturing a robot requires huge cost as they are complex machines.
- Robots do not have any emotions and moral values.
- There will be no improvement with experience.
- Originality of creative mind as human beings is not possible.

- Destructs the creative mind of the human by high usage of robots.

It stores huge volumes of data but the way the data is accessed and used is very different from the human beings. They are unable to change their responses to the modifying environments. There is nothing like working whole heartedly as of human beings. There is no sense of human touch. They make fail to differentiate between the hardworking and inefficient individual.

They can just help you to design and create but it does not match with the power of thinking as of human intelligence. Humans will become lazy as they prefer to do every task by robots and lose their creative thinking.

#### 1.5.1 Feasibility Study

Feasibility study helps us to know how and so what extent the project will be useful for the organization. The Project will be evaluated based on the operation of the project. Whether the way the project operates are useful to organization. Next project will be based on the economy i.e. the cost to be spend on the project, resources required for successful running of the project.

Based on the below mentioned types of feasibilities the feasibility study for the project will be done. They are

- Operational feasibility
- Technical feasibility
- Economic feasibility

#### 1.5.2 Operational Feasibility

Under the operational feasibility how the project will be operated will be tested. From this we can know to what extent our projects will be operated and what are the important operations will be there in our project will be tested.

- How the project interacts with user  
Our project is very friendly with user. User only needs minimum English skills to work with our project. If any error done by the user it will be shown to the user with an error message.
- Reliability  
The reliability of the project is very high. For every module completed it will be visible to the user and if any error done will also visible to the user and by seeing this he/she can do the necessary modifications.
- Security  
Any user can use this project. But care is taken that he/she cannot modify the internal code or the dataset information.
- Availability  
The project works in offline mode. Hence it can be accessed at any time.

#### 1.5.3. Technical Feasibility

Technical feasibility is mainly concentrated on the technologies used to develop the project. It mainly focuses on the platform used to develop the project. What are hardware and software that the project can run?

- Do the technology used is available in the market?

- Is the technology used can handle all the users working in the organization?
- Whether the technology use is reliable to users?
- If the number of users in the organization increases then do the technology used can be updated?
- Can extra extensions be added to the present technology used?
- Based on the changing time can the technology be updated?

#### 1.5.4. Economic Feasibility

The economic feasibility helps the organization to find how effective the project is based on the economic principles. By the economic feasibility one can answer the following questions:

- What are the economic resources used in the project?
- By using which technology, we can reduce the cost of the project?

## 2. Literature Survey

### 2.1 Existing System

In the existing system, the creation of robot requires huge cost. They need frequent updating in the software to cope up with the needs of the changing environment. In case of the severe breakdowns, the procedure to recover the lost codes and to re-install the system takes more time and cost. It cannot interact with the human beings and they are not yet perfected.

### 2.2 Disadvantages of Existing System

- 1) Huge cost
- 2) No replicating humans
- 3) No improvement with experience

### 2.3 Proposed System

In the proposed system, there is no need of software updating and it automatically get updated according to the changing environment. It can react with humans and responds to them to fulfill their requirements. They can their own decisions in case if they encounter any situation which is unfamiliar to them.



## 3. Hardware Requirement Specification

Requirements specification for a system is a complete description of the behavior of a system to be developed. The requirement specification document enlists all necessary requirements that are required for the project development.

### 3.1 Hardware Requirements

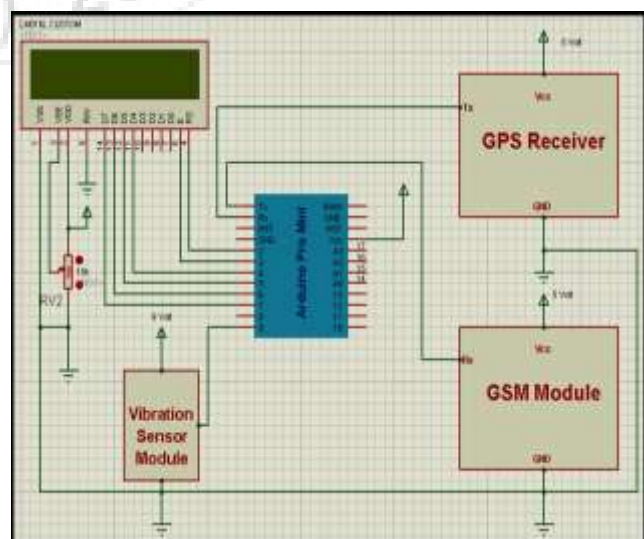
- Arduino Uno R3
- Raspberry pi 3 model B
- Bread Board
- L293 or SN755410 motor driver chip
- Jump cables
- Servo motors
- Power Source Power Bank

### 3.2 Arduino UNO

The Arduino Uno is a microcontroller board in view of the ATmega328 (datasheet). It has 14 computerized input/output pins (of which 6 can be utilized as PWM outputs), 6 simple inputs, a 16 MHz crystal oscillator, a USB association, a power jack, an ICSP header, and a reset catch. It contains everything expected to bolster the microcontroller; just associate it to a PC with a USB link or power it with an AC-to-DC connector or battery to begin. The Uno contrasts from every single going before board in that it doesn't utilize the FTDI USB-to-serial driver chip. Rather, it includes the Atmega8U2 modified as a USB-to-serial converter.

Uno" implies one in Italian and was denoted the arrival of Arduino Software (IDE) 1.0. The Uno board and form 1.0 of Arduino Software (IDE) were the reference adaptations of Arduino, now advanced to more current discharges. The Uno board is the first in a progression of USB Arduino sheets, and the reference demonstrate for the Arduino stage; for a broad rundown of present, past or obsolete sheets see the Arduino list of sheets.

### Circuit Diagram



### Components Used

- Arduino UNO
- Vibration Sensor
- GPS Receiver
- GSM Module
- 12V Battery
- 16\*2 LCD module JHD 162A

### Component Description

#### Technical Specifications

1. Microcontroller - ATmega328
2. Operating Voltage - 5V
3. Input Voltage - 7-12V
4. Input Voltage (limits) - 6-20V
5. Digital I/O Pins - 14
6. Analog Input Pins - 6
7. DC Current per I/O Pin - 40 mA
8. DC Current for 3.3V Pin - 50 mA
9. Flash Memory - 32 KB of which 0.5 KB used by boot loader
10. SRAM - 2 KB
11. EEPROM - 1 KB

### Arduino Physical Structure

The Arduino Uno can be controlled by means of the USB association or with an outer power supply. The power source is chosen consequently. Outside (non-USB) power can come either from an AC-to-DC connector or battery. The connector can be associated by plugging a 2.1mm focus positive connect to the board's power jack. Leads from a battery can be embedded in the Gnd and  $V_{in}$  stick headers of the POWER connector. The board can work on an outer supply of 6 to 20 volts. On the off chance that provided with under 7V, in any case, the 5V stick may supply fewer than five volts and the board might be temperamental. In the event that utilizing more than 12V, the voltage controller may overheat and harm the board. The prescribed range is 7 to 12 volts. The power pins are as per the following:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.
- Each of the 14 digital pins on the Uno can be used as an input or output, using pin Mode (), digital Write (), and digital Read () functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50K ohms. In addition, some pins have specialized functions:
- **Serial:** 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected

to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

- **External Interrupts:** 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach Interrupt () function for details.
- **PWM:** 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog Write () function.
- **SPI:** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- **LED:** 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off. There are a couple of other pins on the board: • AREF. Reference voltage for the analog inputs. Used with analog Reference (). • Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

### 3.3 Raspberry pi 3 model B

The Raspberry pi is a single computer board which is developed to encourage and aid the teaching of programming and computing. For the development of IoT projects it a good starting point. The board is accessible to all and has the lot of connectivity options in it. Raspberry pi is a perfect experimental tool which we can use it as a desktop computer, media centre, server or security device within your home. Linux based operating system can be run on raspberry pi board with having a plenty of access to free software and downloads. The power supply unit is not for general purpose power supply as it has a unique power supply from pi 3 power supply unit. You can get connected to it easily without the help of any external device. The pi 3 brings more processing capability and also saves your time with the development of your applications.



The Raspberry Pi 3 is the third generation Raspberry Pi. It replaced the Raspberry Pi 2 Model B in February 2016. Compared to the Raspberry Pi 2 it has:

- A 1.2GHz 64-bit quad-core ARMv8 CPU
- 802.11n Wireless LAN
- Bluetooth 4.1

- Bluetooth Low Energy (BLE)

Like the Pi 2, it also has:

- 1GB RAM
- 4 USB ports
- 40 GPIO pins
- Full HDMI port
- Ethernet port
- Combined 3.5mm audio jack and composite video
- Camera interface (CSI)
- Display interface (DSI)
- Micro SD card slot (now push-pull rather than push-push)
- Video Core IV 3D graphics core

The Raspberry Pi 3 has an identical form factor to the previous Pi 2 (and Pi 1 Model B+) and has complete compatibility with Raspberry Pi 1 and 2.

### Software Requirements

Coding Languages : Python, Java, C  
Operating System : Windows XP/8, Linux  
API's : Amazon Alexa API, Google Voice API

### Source Code

#### detect\_color.py

```
from pyimagesearch.shapedetector import ShapeDetector
from pyimagesearch.colorlabeler import ColorLabeler
import argparse
import imutils
import cv2
import os

# construct the argument parse and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-i", "--image", required=True,
    help="path to the input image")
args = vars(ap.parse_args())

# load the image and resize it to a smaller factor so that
# the shapes can be approximated better
image = cv2.imread(args["image"])
resized = imutils.resize(image, width=300)
ratio = image.shape[0] / float(resized.shape[0])

# blur the resized image slightly, then convert it to both
# grayscale and the L*a*b* color spaces
blurred = cv2.GaussianBlur(resized, (5, 5), 0)
gray = cv2.cvtColor(blurred, cv2.COLOR_BGR2GRAY)
lab = cv2.cvtColor(blurred, cv2.COLOR_BGR2LAB)
thresh = cv2.threshold(gray, 60, 255,
    cv2.THRESH_BINARY)[1]
cv2.imshow("Thresh", thresh)

# find contours in the thresholded image
cnts = cv2.findContours(thresh.copy(),
    cv2.RETR_EXTERNAL,
    cv2.CHAIN_APPROX_SIMPLE)
cnts = cnts[0] if imutils.is_cv2() else cnts[1]

# initialize the shape detector and color labeler
```

```
sd = ShapeDetector()
cl = ColorLabeler()
# Open a file in write mode
#fo = open("/home/pi/Desktop/PiAUISuite/shape.txt", "w")
# loop over the contours
for c in cnts:
    # compute the center of the contour
    M = cv2.moments(c)
    cX = int((M["m10"] / M["m00"]) * ratio)
    cY = int((M["m01"] / M["m00"]) * ratio)

    # detect the shape of the contour and label the color
    shape = sd.detect(c)
    color = cl.label(lab, c)

    # multiply the contour (x, y)-coordinates by the resize
    ratio,
    # then draw the contours and the name of the shape and
    labeled
    # color on the image
    c = c.astype("float")
    c *= ratio
    c = c.astype("int")
    text = "{} {}".format(color, shape)
    cv2.drawContours(image, [c], -1, (0, 255, 0), 2)
    cv2.putText(image, text, (cX, cY),
        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255,
        255), 2)
    print(text)
    os.system("tts " + text)

cv2.imshow("Image", image)

light_sensor.py
#!/usr/local/bin/python

import RPi.GPIO as GPIO
import time

import os
__author__ = 'Gus (Adapted from Adafruit)'
__license__ = "GPL"
__maintainer__ = "pimylifeup.com"

GPIO.setmode(GPIO.BOARD)

#define the pin that goes to the circuit
pin_to_circuit = 7

def rc_time (pin_to_circuit):
    count = 0

    #Output on the pin for
    GPIO.setup(pin_to_circuit, GPIO.OUT)
    GPIO.output(pin_to_circuit, GPIO.LOW)
    time.sleep(0.1)

    #Change the pin back to input
    GPIO.setup(pin_to_circuit, GPIO.IN)

    #Count until the pin goes high
```

```
while (GPIO.input(pin_to_circuit) == GPIO.LOW):
    cunt += 1
return cunt

#Catch when script is interrupted, cleanup correctly
try:
    # Main loop
    while True:
        print rc_time(pin_to_circuit)
        if rc_time(pin_to_circuit) > 2000:
            os.system("fswebcam -r 640x480 --no-banner
Desktop/out.png")
            os.system("echo ""someone is at your door"" | mail -s
""alert"" desaisaiprasad8@gmail.com")

except KeyboardInterrupt:
    pass
finally:
    GPIO.cleanup()
```

#### Start\_alexash

```
#!/bin/bash
#Start companion service
cd /home/pi/Desktop/alexav-sample-app/samples
cd companionService && npm start&

#Run the sample app
echo "Starting sample app."
cd /home/pi/Desktop/alexav-sample-app/samples
cd javaclient && mvn exec:exec&
echo "When finished "
read -n1 -r -p "Press space to continue..." key

#Run the Wake Word Engine
cd /home/pi/Desktop/alexav-sample-app/samples
cd wakeWordAgent/src && ./wakeWordAgent -e sensory
```

## 4. Future Scope

Humanoid robotics is a developing and testing research field, which has gotten huge consideration during the previous decade and will keep on playing a focal part in mechanical autonomy explore. Regardless the application zone, one of the regular issues handled in humanoid mechanical technology is the comprehension of human-like data preparing and the hidden components of the human cerebrum in managing this present reality.

## 5. Conclusion

Robots are assuming control undertakings which are esteemed dull, grimy and dangerous. The possibility of robots with more noteworthy insight than people is no less than 50 years away, and may never come. It's not the robots we have to stress over, the general population program them. Humanoids can be utilized as laborers at thorough undertaking.

## References

- [1] Asada, H. and Slotine, J.-J. E. (1986). Robot Analysis and Control. Wiley..
- [2] Arkin, Ronald C. (1998). Behavior-Based Robotics. MIT Press.
- [3] Brady, M., Hollerbach, J.M., Johnson, T., Lozano-Perez, T. and Mason, M. (1982), Robot Motion: Planning and Control. MIT Press.
- [4] Horn, Berthold, K. P. (1986). Robot Vision. MIT Press.
- [5] Craig, J. J. (1986). Introduction to Robotics: Mechanics and Control. Addison Wesley.
- [6] Everett, H. R. (1995). Sensors for Mobile Robots: Theory and Application. AK Peters.
- [7] Kortenkamp, D., Bonasso, R., Murphy, R. (1998). Artificial Intelligence and Mobile Robots. MIT Press.
- [8] Poole, D., Mackworth, A. and Goebel, R. (1998), Computational Intelligence: A Logical Approach. Oxford University Press.
- [9] Russell, R. A. (1990). Robot Tactile Sensing. Prentice Hall.
- [10] Russell, S. J. & Norvig, P. (1995). Artificial Intelligence: A Modern Approach. Prentice-Hall. Prentice Hall.

## Author Profile



**Desai.V N S R Sai Prasad** student, currently pursuing B. Tech, in EEE department from Aditya College of Engineering Madanapalle, Chittoor - District, Andhrapradesh, India. Research interest in Embedded Systems.



**J.Praveena** has obtained her B.Tech from JNTUH, Hyderabad in the year 2006. M.Tech in power electronics from college of engineering, JNTUA, Ananthapuramu in the year 2016. She has 9 years of teaching experience. She is Perusing PhD at VIT University. Currently working as Assistant Professor in the Department of EEE. in Aditya College of Engineering Madanapalle, Chittoor district, Andhra Pradesh, Her area of research is Demand side management using IOT, power systems technology