

Design of a Voice Controlled Robotic Arm for Picking and Placing an Object

Shyam.R.Nair¹, Shalini.R.Nair²

¹(Department of Electronics and Instrumentation Engineering, Hindustan University, Chennai, India)

²(Department of Mechanical Engineering, NCERC, Calicut University, Pambadi, Thrissur, India)

ABSTRACT

This paper focuses on design of a voice controlled robotic arm for picking and placing an object. The arm has 2 links and 3 joints. The coordinates from which the object is to be picked and placed is selected by giving proper commands and the robotic arm uses inverse kinematics to execute the task.

Keywords - DOF, Inverse kinematics, Robotics, Robotic arm, Voice control.

1. INTRODUCTION

Robot is a machine to execute different task repeatedly with high precision. Thereby many functions like collecting information and studies about the hazardous sites which is too risky to send human inside. Robots are used to reduce the human interference nearly 50 percent. Robots are used in different types like fire fighting robot, metal detecting robot, etc.

The first robotic arm to be used in an automobile industry was "UNIMATE" in GM motors USA in 1950s. From then there has been tremendous improvement in the research and development in robotics. Now robots are an integral part of almost all industries. Robots have to do different tasks including welding, trimming, picking and placing etc. These robots are controlled in different ways like keypads, optical control, etc.

In this paper, we introduce voice control of the robotic arm. The voice commands are picked up by a PC and by using the speech recognition software and special software developed in Microsoft Visual Studio, the commands are passed to the robot as serial codes. To select the real time object, the corresponding coordinate is selected. Once the robot gets the coordinates, it uses the inverse kinematics to calculate the required rotation.

2. HARDWARE

2.1 Mechanical

The robotic arm has 2 links and 3 joints. It is mounted on the center of a table or the platform on which it is supposed to deal with the objects. The end point of the second link has an electromagnet and all objects are magnetically attractive.

The range of the arm is the total length of the two links. The length of each link can be designed as per requirement. It can be of equal or different lengths. The arm has 3 degrees of freedom. Each joint has a dc geared motor for the link movement.

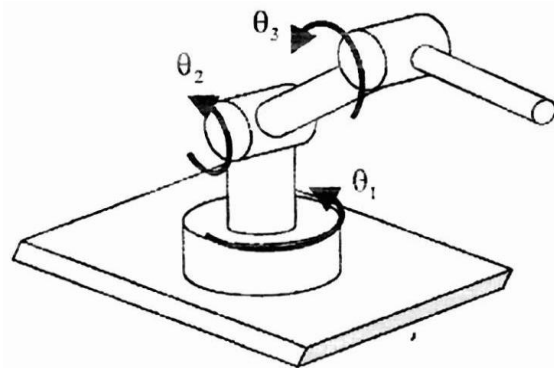


Figure 1 robotic arm

2.2 Electronics

The robot electronics consists of a PIC microcontroller, PC, motor driving unit and a power supply.

The voice commands from human master are fed to the PC with the help of a microphone. The PC uses the speech recognition software and VRobo developed using Microsoft Visual Studio to convert the voice commands to serial codes.

The main commands used are pic, place, x, y, stop, and the numerical values. The required command pic/place is said first and then the corresponding coordinates are told.

For instance, if we need to pic an object from (1, 2) and want to place it at (3, 8), then the following commands are given;

Pic, x, one, y, two, place, x, three, y, eight.

The motors used are dc geared motors with appropriate torque and rpm. Solenoid type relay is used as the motor drive unit.

A lead acid battery is used as power source for the entire system.

3. MATHEMATICAL MODELING

3.1 Inverse kinematics

O is the point to be reached.

'c' and 'a' are lengths of first and second link respectively.

From figure1 and figure3;

$$\theta_1 = \theta$$

$$\theta_2 = A$$

$$\theta_3 = B$$

By Pythagoras theorem,

$$b^2 = x^2 + y^2 + z^2 \quad (1)$$

$$\theta = \tan^{-1}(y/x) \quad (2)$$

$$\phi = \tan^{-1}(z / \sqrt{x^2 + y^2}) \quad (3)$$

We know that area of a triangle is given by;

$$\text{Area} = (s * (s-a) * (s-b) * (s-c))^{1/2} \quad (4)$$

$$\text{Where, } s = (a+b+c)/2 \quad (5)$$

Here, c and a are arm length of link one and two respectively.

We also know that the area of a triangle is given by;

$$\text{Area} = \frac{1}{2} * \text{base} * \text{altitude} \quad (6)$$

From figure (2);

$$\text{Area} = \frac{1}{2} * b * h \quad (7)$$

$$\text{But } h = c * \sin A \quad (8)$$

Now, by substituting (8) in (7), we get;

$$\text{Area} = \frac{1}{2} * b * c * \sin A \quad (9)$$

As we know values of a, b and c, the area is calculated as per equations (4) and (5)

Hence, from equation (9);

$$\sin A = \frac{2 * \text{Area}}{(b * c)}$$

Or

$$A = \sin^{-1} \left(\frac{2 * \text{Area}}{(b * c)} \right) \quad (10)$$

Similarly,

$$B = \sin^{-1} \left(\frac{2 * \text{Area}}{(a * c)} \right) \quad (11)$$

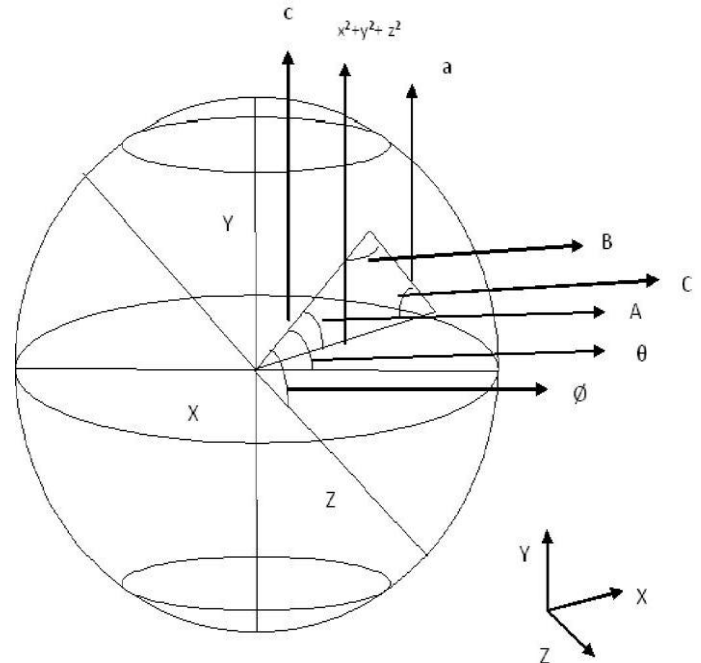


Figure 2 schematic of working environment

3.2 Kinematics

From figure (2),

X coordinates are decided by the arm lengths c, a, angles A, C, θ , ϕ .

As the effective length decreases with increase in angles, A and C, it has a cosine relationship with the x coordinate.

The effective length decreases with increase in the values of θ and ϕ .

Hence, the x coordinate can be written as;

$$X = (c \cos A + a \cos C) \cos \theta \cos \phi$$

Similarly,

Y coordinates are decided by the arm lengths c, a, angles A, C, θ , ϕ .

As the effective length increases with increase in angles, A and C, it has a sine relationship with the y coordinate.

The effective length decreases with increase in the values of θ and ϕ .

Hence, the y coordinate can be written as;

$$Y = (c \cos A + a \cos C) \sin \theta \cos \phi$$

Similarly,

Z coordinates are decided by the arm lengths c, a, angles A, C, θ , \emptyset .

As the effective length increases with increase in angles, A and C, it has a sine relationship with the z coordinate.

The effective length increases with increase in the value of \emptyset .

There is no relation for z with respect to θ .

Hence, the z coordinate can be written as;

$$Z = (c\sin A + a\sin B) \sin \emptyset$$

Hence the kinematic equations for the robotic arm are;

$$X = (c\cos A + a\cos C) \cos \theta \cos \emptyset$$

$$Y = (c\cos A + a\cos C) \sin \theta \cos \emptyset$$

$$Z = (c\sin A + a\sin C) \sin \emptyset$$

Matrix Transformation;

$$\begin{bmatrix} \cos\theta\cos\emptyset & \cos\theta\sin\emptyset & 0 & 0 \\ \sin\theta\cos\emptyset & \sin\theta\sin\emptyset & 0 & 0 \\ 0 & 0 & \sin\emptyset & \cos\emptyset \end{bmatrix} * \begin{bmatrix} c\cos A \\ a\cos C \\ c\sin A \\ a\sin C \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

3.3 Motor Rotation

From equations (2), (9) and (10), the values of θ , A and B are obtained.

\emptyset is the angle of rotation for the base motor. A is the angle of rotation for the motor connecting the first link and B is the angle of rotation for the motor connecting the second link.

Here, we consider that all motors are of 10rpm. And hence, all the motors cover 60° in one second.

To improve accuracy, the motor is turned on only for 1ms in one on loop in the program.

Hence the motor covers an angle of 0.06° in one on loop.

Once the angles are calculated by using the inverse kinematics, the PIC microcontroller decides the number of on loops to be executed for each motor.

For example, if the angle to be covered by the base motor is 24° , then the PIC microcontroller will execute the on loop 400 times.

3.4 Transformation from initial to final point

Once the robot gets the coordinate values, it computes the angles using the inverse kinematics and proper rotations are made by the motors. These angle values are stored in eeprom of the controller.

When the next coordinates are obtained, again the robot calculates the angles, but this time it looks back to the values stored in eeprom and compares it with the newly computed angle values. Then the robot rotates the motors in such a way that the new angles are achieved. And the newly computed angles are overwritten on the previously stored values.

For example;

If the robot has rotated the motors in the following sequence;

A with 15°

B with 20°

\emptyset with 10°

\emptyset with 25°

And if the newly calculated angles are

A with 17°

B with 19°

\emptyset with 10°

\emptyset with 27°

Then the robot will make the following rotations;

A with 2° positive rotation

B with 1° negative rotation

\emptyset with 0°

\emptyset with 2° positive rotation

Thus transformation is done from initial point to the final point.

4. SOFTWARE

Onboard software is mainly developed with micro C. This software interfaces between the PIC microcontroller and the PC. Simulations have been executed both in PIC simulator. The voice control software was developed using Microsoft Visual Studio.

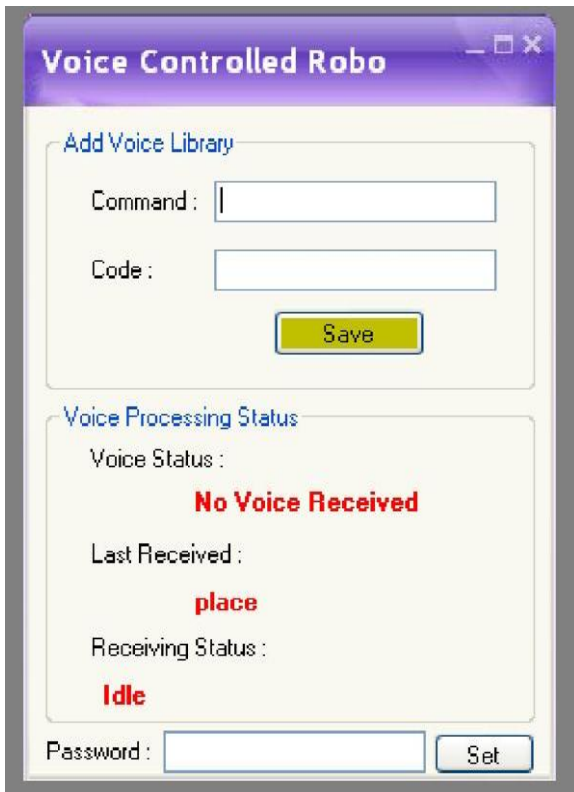


Figure 3 voice control panel

5. CONCLUSION

Voice controlled robotic arm for picking and placing an object was successfully designed. The robot control was found to be user friendly.

In future, brain caps, finger gesture, etc can be implemented.

ACKNOWLEDGEMENT

The research in this paper was carried out at Hindustan Institute of Technology and science, Chennai. This work was supported by the e-MEN Robotic Research Centre Palakkad.

REFERENCES

Journal Papers:

- [1] Shyam.R.Nair, "Design of an Optically Controlled Robotic Arm for Picking and Placing an Object", *International Journal for Scientific and Research Publications, Vol 2, Issue 4, April 2012.*
- [2] Amon Tunwannarux, and Supanunt Tunwannarux "Design of a 5-Joint Mechanical Arm with User-Friendly Control Program" *World Academy of Science, Engineering and Technology 27 2007.*

Books:

- [3] Dr.R.K.Bansal "a textbook of engineering mechanics" page no 195-199,page no 205-209
- [4] John-David Warren, Josh Adams and Harald Molle, "Arduino Robotics"

Proceedings Papers:

- [1] Shyam.R.Nair and Prof. Manjula Pramod, "Self Securable Security Robot Using High Speed Internet for Communication with the Human Master", *IRNet, International Conference on Computational Vision and Robotics (ICCV), March 2012.*
- [2] Redwan Alqasemi and Rajiv Dubey "Control of a 9-DoF Wheelchair-Mounted Robotic Arm System" *Florida Conference on Recent Advances in Robotics, FCRAR 2007.*