Design of a 32 DOF Andro Humanoid Robot using Cascaded Computational Intelligence

S Shyam.R.Nair, Prof. Manjula Pramod

Dept. of EIE, Hindustan University, Chennai

Abstract- This paper focuses on the design of a 32 DOF andro humanoid robot using cascaded computational intelligence. The robot consists of hip, neck, Lip, shoulder, elbow, and wrist and finger joints. The robot secures itself from hacking and power problems. The intelligence of the robot is a combination of human intelligence, computational intelligence and sensor fusion unit data.

Index Terms- Robot, Kinematics, Inverse kinematics, Andro Humanoid, DOF, Cascaded computational intelligence, sensor fusion unit.

I. 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. Humanoids are robots which resembles human joints. Humanoids are nowadays used in industrial purpose, military, research etc.

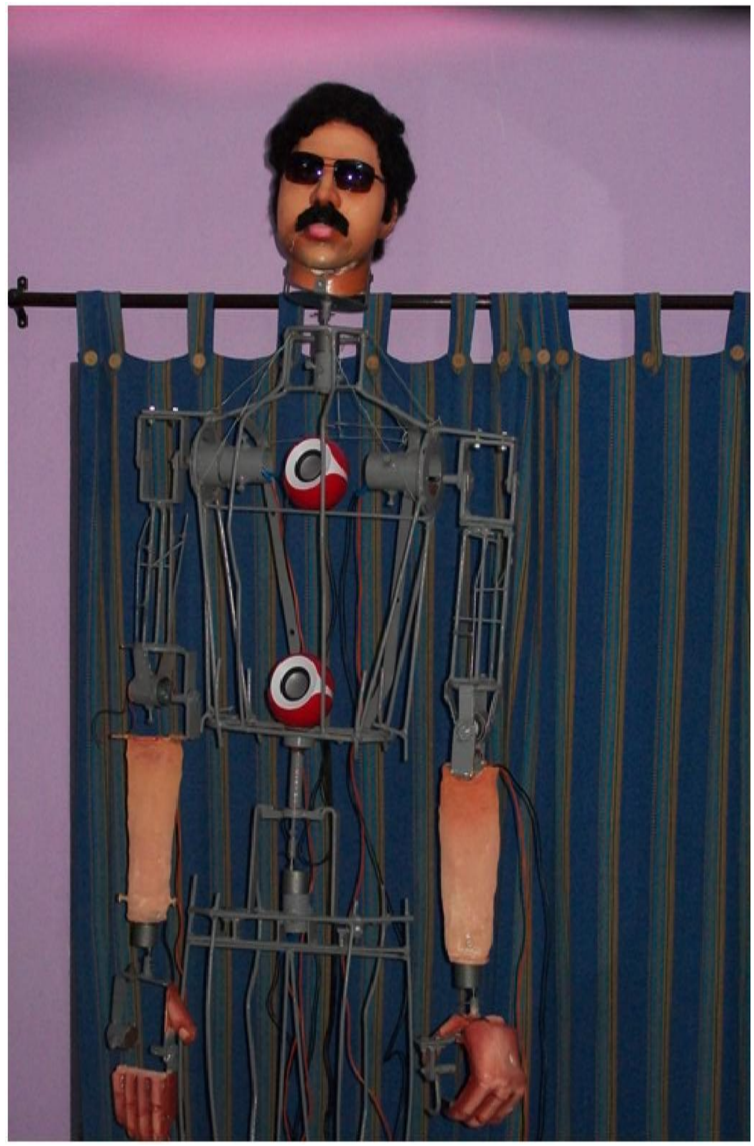
This paper focuses on designing a 32 DOF andro humanoid robot. The robot uses cascaded computational intelligence and hence it has a self securing feature. The robot is controlled by a human master and a local master controller which is connected to a sensor fusion network.

II. Hardware Concept

A. Mechanical Part

The humanoid consists of a hip, neck, Lip, shoulder, elbow, and wrist and finger joints. The humanoid has 32 degrees of freedom.

The humanoid has a platform on which it is mounted. The platform helps the humanoid to move around in a plane surface. The platform consists of four wheels each of which is connected to dc geared motors. The hip joint of the humanoid has one degree of freedom and allows the upper body of the humanoid to rotate in horizontal direction. The neck joint helps to rotate the head in horizontal direction. A web camera is mounted in the eyes of the humanoid.



The humanoid has shoulder, elbow, wrist and finger joints on both hands. The shoulder has 2 DOF, elbow has 3DOF, wrist has 4 DOF and the finger unit has 5 DOF. The hip and neck has one DOF and the lip has 2 DOF. All these togather makes 32 DOF andro humanoid.

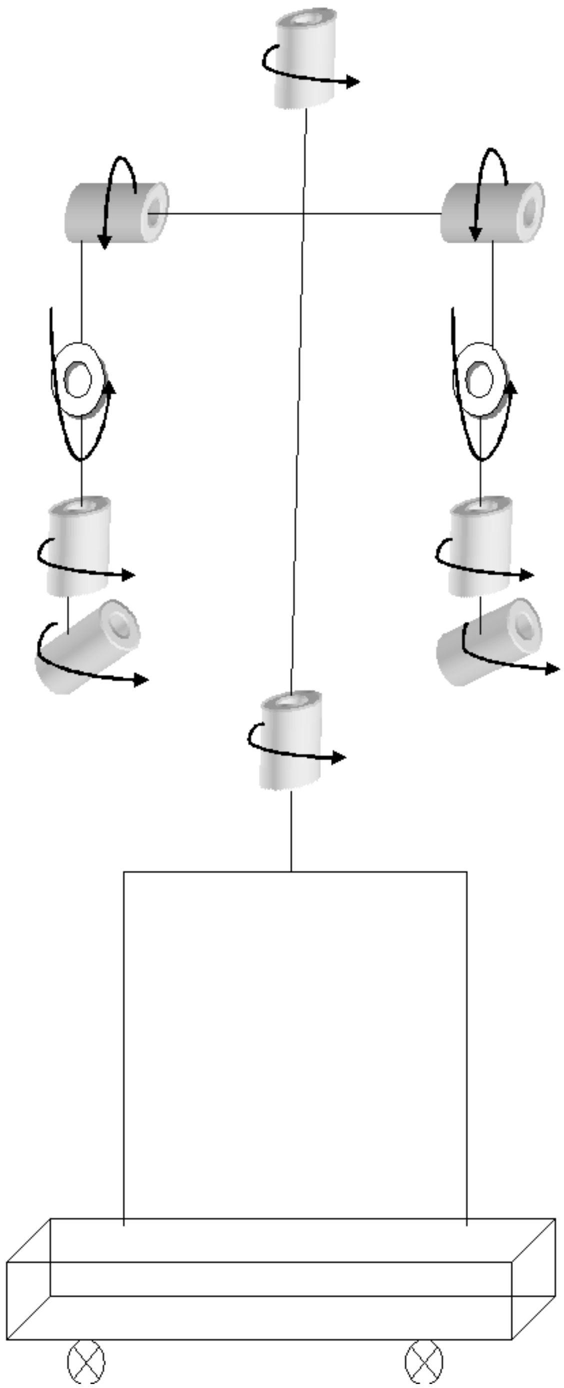


Figure 2. Schematic of motor arrangement

B. Torque Calculations

There are 14 motors with 10 rpm each including the four in the platform which are connected to the wheels.
Neck motor:
The motor is placed at neck which will make the head rotate in the horizontal plane.
Weight of the head including the web camera is 0.5 Kg.
The distance from the shaft of the motor is 25 cm.
Hence torque required to make proper movement is 0.5*6 = 3 Kgcm
Required power = 2*π*N*T/60
= 3.12 Watts
Where N is rpm of the motor and T is torque

Hip motor:
The motor is placed at hip which will make the upper body rotate in the horizontal plane.
Weight of the upper body is 5.5 Kg.
The distance from the shaft of the motor is 30 cm.
Hence torque required to make proper movement is 5.5*30 = 165 Kgcm
the required torque is 50% of the calculated torque due to the symmetry of the upper body.
Hence actual torque = 82.5 Kgcm.
Required power = 2*π*N*T/60
= 85.8 Watts.

Shoulder motor:
The motor is placed at the shoulder which will make the hand rotate in the vertical plane.
Weight of the hand is 1.5 Kg.
The distance from the shaft of the motor is 70 cm.
Hence torque required to make proper movement is
1.5*70 = 105 Kgcm
the required torque is 50% of the calculated torque as one of the end is hinged which supports the rotation.
Hence actual torque = 52.5 Kgcm.
Required power = 2*π*N*T/60
= 54.95 Watts.

Elbow motor:
The motor is placed at elbow which will make the arm rotate in the horizontal plane.
Weight of the arm is 0.6 Kg.
The distance from the shaft of the motor is 20 cm.
Hence torque required to make proper movement is
0.6*20 = 12 Kgcm
the required torque is 50% of the calculated torque as one of the end is hinged which supports the rotation.
Hence actual torque = 6 Kgcm.
Required power = 2*π*N*T/60
= 6.24 Watts.

Wrist motor:
The motor is placed at the wrist which will make the palm yoke.
Weight of the palm is 0.3Kg.
The distance from the shaft of the motor is 20 cm.
Hence torque required to make proper movement is
0.3*20 = 6 Kgcm
Required power = 2*π*N*T/60
= 6.24 Watts.

Finger motor:
The motor is placed at the finger which will make the finger grasp.
Weight of the finger unit is 0.05Kg.
The distance from the shaft of the motor is 10 cm.

Hence torque required to make proper movement is
0.05*10 = 0.5 Kgcm
Required power = 2*π*N*T/60
= 0.52 Watts.

Wheels motor:
The motors are coupled to the wheels of the platform for free movement in a plane surface.
The weight of the body is equally distributed to four wheels. Thus each wheel will bear a load of 10Kg.
Distance of the wheel from axis of the motor= 2cm.
Hence torque required for 1 wheel is 10*2 = 20Kgcm.
Required power = 2*π*N*T/60
= 20.8 Watts.

C. Inverse Kinematics

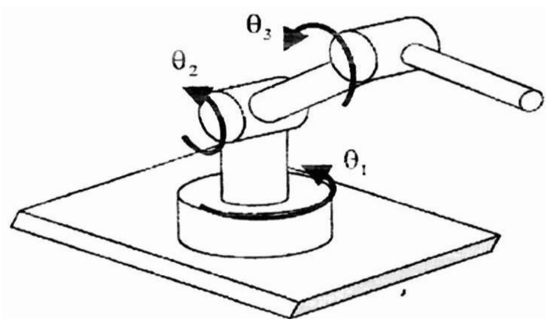


Figure 3. Arm Schematic

O is the point to be reached.
'c' and 'a' are lengths of first and second link respectively.
From figure3 and figure 4;
θ1 = θ
θ2 = A
θ3 = B

By Pythagoras theorem,

$$b^2 = x^2+y^2+ z^2 \quad (1)$$
$$\theta = \tan^{-1}(y/x) \quad (2)$$
$$Ø = \tan^{-1}(z/ x^2+y^2) \quad (3)$$

We know that area of a triangle is given by;

$$Area = (s*(s-a)*(s-b)*(s-c))^{½} \quad (4)$$
$$\text{Where, } s = (a+b+c)/2 \quad (5)$$

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;

$$Area = ½*base*altitude \quad (6)$$

From figure (4);

$$Area = ½*b*h \quad (7)$$
$$\text{But } h = c*\sin A \quad (8)$$

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

$$Area = ½*b*c*\sin A \quad (9)$$

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

Hence, from equation (9);

$$\sin A = 2*Area/ (b*c)$$

Or

$$A = \sin^{-1} (2*Area/(b*c)) \quad (10)$$

Similarly,

$$B = \sin^{-1} (2*Area/ (a*c)) \quad (11)$$

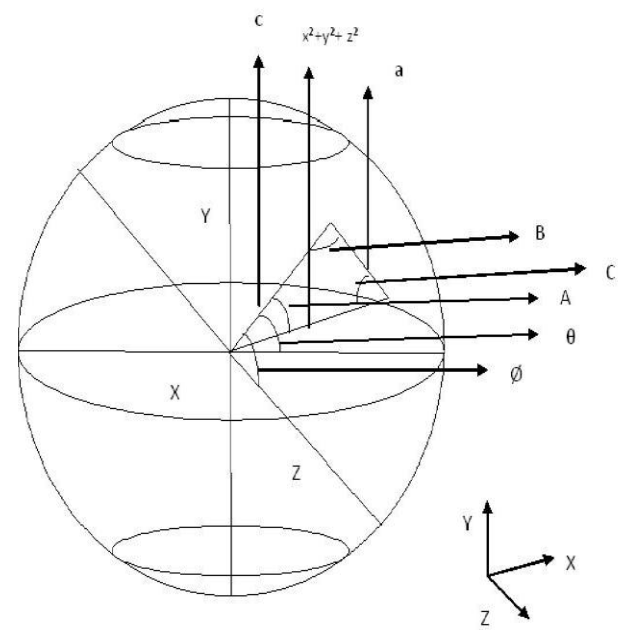


Figure 4. Schematic of working environment

D. Kinematics

From figure (4),
X coordinates are decided by the arm lengths c, a, angles A, C, θ, Ø.
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 Ø.

Hence, the x coordinate can be written as;
X = (cCosA+aCosC) Cos θ Cos Ø

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

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 Ø.

Hence, the y coordinate can be written as;
Y = (cCosA+aCosC) Sin θ Cos Ø

Similarly,
Z coordinates are decided by the arm lengths c, a, angles A, C, θ, Ø.
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 Ø.
There is no relation for z with respect to θ.

Hence, the z coordinate can be written as;
Z = (cSinA+aSinB) Sin Ø

Hence the kinematic equations for the robotic arm are;
X = (cCosA+aCosC) Cos θ Cos Ø
Y = (cCosA+aCosC) Sin θ Cos Ø
Z = (cSinA+aSinC) Sin Ø

Matrix Transformation;

$$\begin{bmatrix} cos\theta cos\emptyset & cos\theta cos\emptyset & 0 & 0 \\ sin\theta cos\emptyset & sin\theta cos\emptyset & 0 & 0 \\ 0 & 0 & sin\emptyset & sin\emptyset \end{bmatrix} * \begin{bmatrix} cCosA \\ aCosC \\ cSinA \\ aSinC \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

E. Motor Rotation

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

F. 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°
Θ with 10°
Ø with 25°

And if the newly calculated angles are

A with 17°
B with 19°
Θ with 10°
Ø with 27°

Then the robot will make the following rotations;

A with 2° positive rotation
B with 1° negative rotation
Θ with 0°
Ø with 2° positive rotation

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

III. MODES OF OPERATION

Manual

In manual mode, the robot is controlled by the human master. The task that is to be executed is instructed by the human master as voice, finger gesture, eye ball movement, brain cap etc.

The instruction is fed to the client PC as characters using appropriate human machine interface. These characters are fed serially to the server laptop using high speed internet. The server laptop receives these instructions through high speed internet and feeds it to the master PIC microcontroller.

The visuals are obtained through the camera in the robot by using the Gmail video chat or Skype software. This also introduces a dual channel communication, ie, the instructions are given through one channel and the visuals are obtained through other channel. So if one of the channels is hacked we get access to the robot using the other channel.

Suppose we lose the video signals, we can command the robot to come back to the source point. Otherwise if we lose the instruction channel, the video chat helps us to locate the robot.

In this mode, the sensor fusion unit is not activated. This mode is usually used for training purpose.

Automated

In this mode, the robot does not get the instructions from the human master. Instead, it will use the data in eeprom to execute task or it will use neural scheme for executing task.

In this mode, the sensor fusion unit is activated and it is used to analyze the environmental conditions.

Cascaded

In certain places, we cannot expect the robot to execute task by itself and neither can we expect human commands enough to execute the task like surveying of an unknown area or in warfare. In such area, we use both human intelligence and sensors for executing a task. In such situations the cascaded mode will be of great use.

In this mode the instructions from master is received in the client node as explained in manual mode.

Once the instruction is received at the master controller it verifies the environmental conditions. This master controller is connected to sensor fusion unit which is used to observe the environmental conditions. The sensor fusion unit consists of ultra sonic sensors for identifying and locating obstacles, temperature sensor for analyzing the surrounding temperature, etc. The master controller verifies the environmental conditions and executes the task if and only if the conditions are ideal to be executed.

In this mode the robot continuously verifies the communication system security. This is done by transmitting and receiving a peculiar code in a definite interval of time. If the code is not received in either of the ends at proper time then the robot disconnects all the communication channels and returns back to the source point by using the data stored in eeprom of the controller.

In this mode the robot also verifies the battery level using a battery monitoring circuit. If the robot identifies a drain in battery charge, it automatically switches to the secondary power source. Once the robot switches to the last battery, it will stop all the task and returns back to the source point using the data in eeprom.

IV. ELECTRONICS

The platform is designed to fit a laptop, sensors, and four 12V 7aH lead acid batteries. The platform also contains the electronics hardware including a relay based circuit for driving the dc geared motors, a pic microcontroller circuits, and ultrasonic sensors.

The task to be executed is given as voice commands[14] by the human master at the client pc. A serial code is generated for each word and is transmitted to the laptop on the robot platform. For generating, transmitting and receiving the code software's known as Roboclient and Roboserver is used. Both these software's are developed in Microsoft Visual Studio. The client pc and the server laptop is connected to internet using high speed 3G network. The serial code is transmitted from the laptop to the microcontroller circuit by a RS-232 cable. When the microcontroller receives this signal, it checks for the environmental conditions using the ultrasonic sensors. The environmental condition, here, refers to any obstacle for the task execution. Now if there are no obstacles, then, the robot executes the task. But if there is any obstacle, it sends the human master a serial code indicating its problem and waits for further instruction from human master. The human master can view the area under survey with the help of the camera mounted on the robot platform. This is made possible by using the Gmail video chat or the Skype software.

The robot platform also contains a battery monitoring circuit. If this circuit identifies a power crisis, then, the circuit switches to the next power source. The number of power sources can be fixed according to the need and availability. If the last battery is switched, then, the robot closes all its applications and comes back to the source point by using the data in the eeprom. The robot also returns to the source point if it identifies a security issue in the communication link.

A. Identification of a security issue in communication link

The client pc and the server laptop continuously exchange a serial code. If the pic microcontroller does not receive this code in a specified delay, it comes to a conclusion that the communication link is lost or hacked. In that case, the robot closes all its communication links and other activities and returns back to the source point. By using the data stored in eeprom.

B. Returning of robot to the source point using the data in eeprom

The robot stores all the movement codes in the internal eeprom of the pic microcontroller. The first location contains the movement code and the next location contains the time delay for the code. This continues for all the movement codes. When the robot faces a security or power problem, the robot executes the data in eeprom in last in first out method. The controller swaps the forward and backward instructions. That is if there is a forward movement in the eeprom location, the robot moves backward and vice versa. The rest of the instructions remain the same. This helps robot from protecting itself from exploitation from unauthorized access.

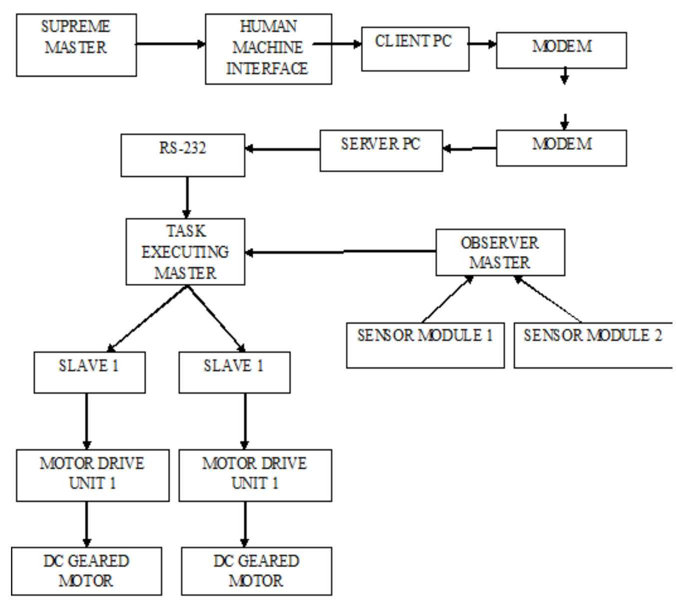


Figure 5. Block Schematic

C. Visual sensor

This part is the place to install sensors which are used for searching vital sign of the victims like camera [11]. The camera can be made to continuously rotate as per the command from human master. The visuals obtained by the camera are transmitted through the Gmail video chat or the Skype software.

ISSN 2250-3153

D. Monitoring and navigation of robot

The robot is monitored using the camera in the robot platform. The robot uses ultrasonic sensors for identifying any obstacle in the path of its motion. The sensors are connected to the controller circuit for making appropriate decision. When the robot navigates automatically, it uses the data in eeprom. For the patrol robot navigation several methods are used like sensing the paths, graphical user interface, map following algorithms and compass, other Kalman filtering methods[6][5][7][8][9]. All these approaches are somehow complex. This can be overcome by using the EEPROM of the microcontroller. PIC16F877A has 256 bytes of EEPROM inside it. So memory can use it to store data that need on a permanent basis and we can read it back from it. There are two functions to accomplish the task. Eeprom_Read and Eeprom_Write.

Eeprom_Read function returns an integer and takes a parameter of the address from which the data has to be fetched. Eeprom_Write takes two parameters the first one is the address and the second one is the data.

unsigned short Eeprom_Read(unsigned int address);
void Eeprom_Write(unsigned int address, unsigned short data);

Most of the robot will navigate with different algorithm.[10][11][12][13] The robot will navigate with respect to the commands from the user (initially from A to B) as shown in fig.7. All the running commands are stored in the memory and also a counter is set to determine how long the commands are executed. Now the current position of the robot is at B. Whenever it needs to return, the last stored command will execute first (LIFO). If the last executed command is forward, that swaps it into backward command.
Similarly,

Backward ≈ Forward

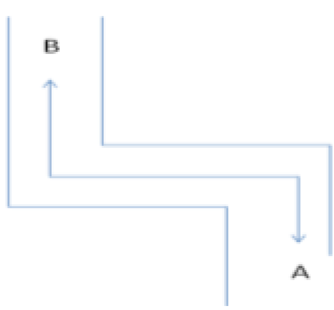


Figure 6. Path Schematic

V. SOFTWARE

Onboard software is mainly developed with micro C. This software interfaces between the operator station software and the robot by receiving operator's command to control all robot functions. Simulations have been executed both in Mat lab and PIC simulator. In Mat lab the approaches were implemented under ideal hypothesis, more realistic settings. The commands are send through serial communication with the help the software Roboclient developed in Microsoft Visual studio. The data is received using the software Roboserver developed in Microsoft Visual studio. Real time visuals can be captured and displayed on the window with the help of Gmail video chat or Skype software.

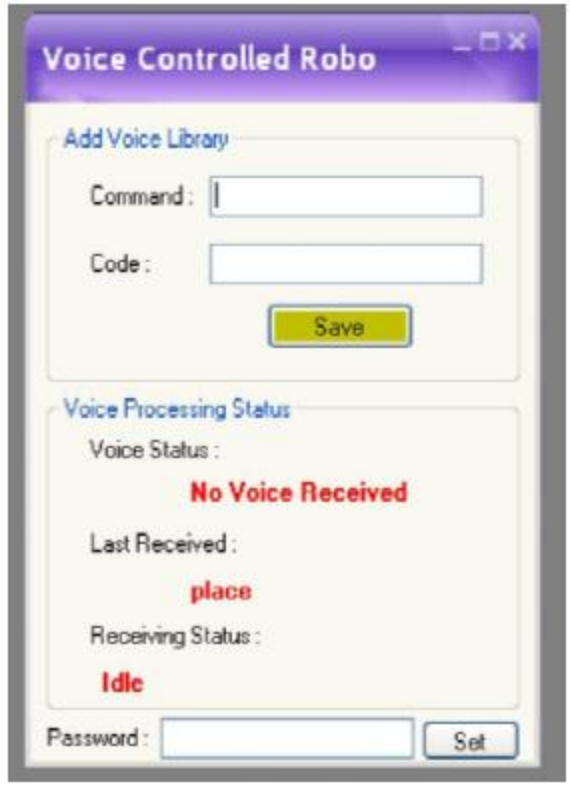


Figure 7. Voice Control Panel

VI. SAMPLE WORKING

The required action is commanded by the human master as a specific word which is picked by the microphone and converted to a code alphabet by the client PC using Roboclient software. This alphabet is transmitted to the server laptop at the robot platform through high speed internet. At the reception end the alphabet is transmitted to the master controller from the server laptop through RS-232 cable. Now the master controller controls the slaves according to the program for the particular alphabet.

For example, when a command 'move' is said by the human master, a code alphabet 'm' is transmitted and the microcontroller on receiving this 'm', executes the internal loop program and waits for the next instruction.

VII. CONCLUSION

The 32 DOF andro humanoid robot was designed successfully and was found to have advantages of unbounded control with the use of high speed internet. The humanoid also has the advantage of self securing ability.

In future the humanoid can be made to walk by designing the legs for the robot. Additional sensors can be fused to improve efficiency of the humanoid.

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

[1] Kenji KANEKO, Fumio KANCHIRO, Shuuji KAJITA, Hirohisa HIRUKAWA, Toshikazu KAWASAKI, Masaro HIRATA, Kazuhako AKACHI and Takakatsu ISOZUMI “Humanoid robot HRP-2”, proceedings of the 2004 IEEE International Conference on Robotics and Automation.

[2] Kenji KANEKO, Fumio KANCHIRO, Shuuji KAJITA, Hirohisa HIRUKAWA, Toshikazu KAWASAKI, Kensuke HARADA, Kazuhako AKACHI and Takakatsu ISOZUMI “First Humanoid Robot that has the Same Size as a Human and that can Lie Down and Get Up”, proceedings of the 2003 IEEE International Conference on Robotics and Automation.

[3] T.Asfour and R.Dillmann “Human-Like Motion of a Humanoid Robot Arm based on a Closed-Form Solution of the Inverse Kinematics Problem”, IEEE International Conference on Intelligent Robots and Systems.

[4] Amon Tunwannarux and Supanunt Tunwannarux “The CEO Mission II, Rescue Robot with Multi-Joint Mechanical Arm”, Proceedings of World Academy of Science, Engineering and Technology 2007.

[5] Mbaitiga Zacharie “Intelligent OkiKOSenPBX Security Patrol Robot via Network and Map-Based Route Planning”, Journal of Computer Science 5(1);79-85,2009.

[6] Fen Xu, Zheng Xi Li, Kui Yuan “The design and implementation of an autonomous campus patrol robot” Proceedings of the 2008 IEEE International Conference on Robotics and Biomimetics.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[7] S. Ali A. Moosavian, Hesam Semsarilar, Arash Kalantari “Design and manufacturing of a mobile rescue robot” international conference on intelligent robots and system oct 9-15-2006 IEEE

[8] Robert W. Hogg, Arturo L. Rankin, Stergios I. Roumeliotis “Algorithms and Sensors for Small Robot Path Following”

[9] Sebastian Thrun1, Maren Bennewitz2, Wolfram Burgard2”MINERVA: A Second Generation Museum Tour-Guide Robot” Carnegie Mellon University.

[10] Stephen J. Tobias, A. Antonio Arroyo “Autonomous Path finding” 2000 Florida Conference on Recent Advances in Robotics May 4-5, 2000, Florida Atlantic University.

[11] Ali Sekmen, Sheldon Greene “Vision-based Mobile Robot Learning and Navigation”2005 IEEE International Workshop on Robots and Human Interactive Communication

[12] Yongxing Hao, Benjamin Laxton ,Sunil K. Agrawal, Edward Lee , Eric Benson “Planning and Control of UGV Formations in a Dynamic Environment: A Practical Framework with Experiments” Proceedings of the 2003 IEEE International Conference on Robotics & Automation Taipei, Taiwan, September 14-19, 2003

[13] Johann Borenstein+ and Yoram Koren “MOTION CONTROL ANALYSIS OF A MOBILE ROBOT” Transactions of ASME, Journal of Dynamics, Measurement and Control, Vol. 109, No. 2, pp. 73-79. [20]

[14] Brandi House, Jonathan Malkin, Jeff Bilmes “The VoiceBot: A Voice Controlled Robot Arm” CHI 2009, April 4-9, 2009, Boston, Massachusetts, USA.

[15] Shyam.R.Nair, “Cascaded Computational Intelligence for Robotics”, ICCVR 2012.

[16] Shyam.R.Nair, “Design of a Robotic Arm for Picking and Placing an Object Controlled using LABView”, IJSRP, May 2012.

AUTHORS

First Author –

Shyam.R.Nair, PG Scholar, Hindustan University, Chennai, shyamieee@gmail.com

Second Author - Prof. Manjula Pramod, MS, HOD, Dept. of EIE, Hindustan University, Chennai.