Research Article

Open Access

CrossMark

Simulated robotic arm control using sound recognition system commands

Abstract

This manuscript deals with two specific issues for simulation a sound recognition system which has ability for controlling the movement of the manipulator robot arm (JACO). The robotic arm should be modulated then implemented using simulation program (MORSE), the simulated arm must have the same real design with all the details which involved the joints, links length, rotational joints angels. The sound system will build using robot operating sysytem (ROS) then connect the system to the simulated arm for creating one block system that can control the arm with the specific words.

Keywords: JACO arm, morse, ROS

Volume 3 Issue 1 - 2017

Salih Rashid Majeed, Klause Dieter Kuhnert

Institute for real time learning, Siegen University, Germany

Correspondence: Salih Rashid Majeed, Institute for real time learning, Siegen University, Germany,
Email salih.majeed@uni-siegen.de

Received: July 28, 2017 | **Published:** September 20, 2017

Abbreviations: ROS, robot operating system; SR, speech recognition; NL, natural language; HCI, human computer interaction; HRI, human robot interaction.

Introduction

Robotics systems are becoming highly complex and sophisticated, with an increasing number of hardware and software components there is also an increasing variety of tasks involved in performing robotics experiments, which induces much time and resources for validation the use of a simulator can ease the development, allowing to verify the component integration and to evaluate their behavior under different controlled circumstances.[1,2] The simulation today becomes one of the important issues which involved in the robotic fields due to many factors such as low cost , easily redesign, the safety especially the applications which involve the human applications and health such as assistive robot. They are many methods for controlling the robot like joystick, sound commands, programming, keyboard etc. Speech is the most used way of communication for people. We born with the skills of speaking learn it easily during our early childhood and mostly communicate with each other with speech throughout our lives. By the developments of communication technologies in the last era, speech starts to be an important interface for many systems. Instead of using complex different interfaces, speech is easier to communicate with computers.[3] Human-Robot interaction is an important, attractive and challenging area in HRI. The Service Robot popularity gives the researcher more interest to work with user interface for robots to make it more user friendly to the social context. Speech Recognition (SR) technology gives the researcher the opportunity to add Natural language (NL) communication with robot in natural and even way. The working domain of the Service Robot is in the society to help the people in every day's life and so it should be controlled by the human. Our future work will focus on introducing more complex activities and sentence to the system and also introducing the non-speech sound recognition, like footsteps (close), footsteps (distant) etc. Humans normally use gestures such as pointing to an object or a direction with the spoken language, i.e., when the human speaks with another human about a close object or location, they normally point at the object/location by using their fingers. This interface called multi-model communication interface,[4] Speech Recognitions a prominent technology for Human-Computer Interaction (HCI) and Human-Robot Interaction (HRI). The increase in the use of robots and automation has significantly attracted the attention of both academic research and industrial applications. In addition to facilitating the daily work the use of robots and automation has helped the productivity and reduces wastage. Although there are many ways that were developed to communicate with the robot, but the ability to communicate verbally can provide a new communication approach. The main objective of this paper is to develop a system that is able to recognize the voice of a consumer and to control the robot movements with verbal instructions.[5]

Background

Brandi House, Jonathan Malkin, Jeff Bilmes

Present a system whereby the human voice may specify continuous control signals to manipulate a simulated 2D robotic arm and a real 3D robotic arm. Our goal is to move towards making accessible the manipulation of everyday objects to individuals with motor impairments. Using our system we performed several studies using control style variants for both the 2D and 3D arms. Results show that it is indeed possible for a user to learn to effectively manipulate real-world objects with a robotic arm using only non-verbal voice as a control mechanism. Our results provide strong evidence that the further development of non-verbal voice controlled robotics and prosthetic limbs will be successful.[6]

David Be, Cinhtia González, Manuel Escalante, Michel García, Carlos Miranda, Sergio Gonzalez

Presents a wireless interface to control a LEGO NXT robot using voice commands through a computer. To perform speech recognition is used CSLU TOOLKIT with a corpus of Mexican Spanish voice, recognized commands are sent via Bluetooth from computer to robot, programming and motion routines to control the motors are done using Java and LeJOS NXJ. The interface consists of two main modules interconnected through the implementation of sockets: the voice recognition module and the wireless control module. The results indicate that the wireless control system of the LEGO NXT robot through voice commands successfully meets its objective.[7]

Praveen Blessington, Madhav BTP, Sagar Babu M, Rajiv Reddy R, Mani Kumar DIP, Naga Raju, Anil Babu N

Presents a robotic vehicle that can be operated by the voice commands given from the user. Here, we use the speech recognition system for giving &processing voice commands. The speech recognition system use an I.C called HM2007, which can store and recognize up to 20 voice commands. The R.F transmitter and receiver are used here, for the wireless transmission purpose. The micro controller used is AT89S52, to give the instructions to the robot for its operation. This robotic car can be able to avoid vehicle collision, obstacle collision and it is very secure and more accurate. Physically disabled persons can use these robotic cars and they can be used in many industries and for many applications.[8]

Jayesh Chopade, Dattatray Barmade, Swapnil Tonde

Proposed a method for controlling a spy robot either through voice commands or computer commands and also composed with camera. Study of human robot communication is one of the most important research areas. The voice communication is significant in human robot interaction among various communication media. The voice commands are used to control the robot and visual feedback is used to provide the precision control to the robot. This robot is also build up with obstacle detection module that generates the signal as the obstacle detects. The proposed system is capable of positioning the robot at tedious work space as instructed through command to get the actual visual feedback. This proposed system is controlled either by voice commands or by computer commands as per user convenience.[9]

Ekapol Chuangsuwanich, Scott Cyphers, James Glass, Seth Teller

Describe a speech system for commanding robots in human-occupied outdoor military supply depots. To operate in such environments, the robots must be as easy to interact with as are humans, i.e. they must reliably understand ordinary spoken instructions, such as orders to move supplies, as well as commands and warnings, spoken or shouted from distances of tens of meters. These design goals preclude close-talking microphones and "push-to-talk" buttons that are typically used to isolate commands from the sounds of vehicles, machinery and non-relevant speech.[10]

Method software

In this section we will present the software and the programmes that has been used to accomplish this research before beginning into the sound system, the first step is implanting of the JACO manipulator arm that using a simulation program (Morse) here we will present the required software:

MORSE: a new open-source robotics simulator. MORSE provides several features of interest to robotics projects: It relies on a component-based architecture to simulate sensors, actuators and robots; it is flexible, able to specify simulations at variable levels of abstraction according to the systems being tested; it is capable of representing a large variety of heterogeneous robots and full 3D environments (aerial, ground, maritime); and it is designed to allow simulations of multiple robots systems MORSE uses a "Software in- the- Loop" philosophy.[11]. It focuses on realistic 3D simulation of small to large environments, indoor or outdoor, with one to tenths of autonomous robots MORSE can be entirely controlled from the command line simulation scenes are generated from simple Python scripts.[12] As described before MORSE is based on Game engine BLENDER and python programming language.[13]

Blender: Is the free and open source 3D creation suite, it supports the entirety of the 3D pipeline modeling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing and game creation, modern users employ Blender's API for Python scripting to customize application and write the specific tools; often these are included in Blender's future releases.[12,13] The sound system will be designed using Ros, robot opening source.

Robot Operating System (ROS): Is a robotics middle ware (i.e. collection of software frameworks for robot software development). Even though ROS is not an operating system, it provides services designed for heterogeneous computer cluster such as hardware abstraction, low-level device control, implementation of commonly used functionality, message-passing between processes, and package management. Running sets of ROS-based processes are represented in a graph architecture where processing takes place in nodes that may receive, post and multiplex sensor, control, state, planning, actuator and other messages. The main ROS client libraries (C++, Python, LISP) are geared toward a Unix-like system.[13]

The experimental work and results

The first step is implementation of the JACO manipulator robotic arm, this arm has been drawn and implemented with the same specifications (links length, rotational angels limit for each joints) this specifications should follow the real arm design as shown in the Figure 1 the simulated JACO arm. After simulated the JACO arm in MORSE, some actuators and sensors should be added to the arm body. The bones are using for moving the arm as actuators; this bones have rotation about itself and about specific axis as shown in Figure 2. The chain of bones called armature, the armature plays very important objective in the simulation of the arm in Morse, this will be the main core for the movement of the skeleton:

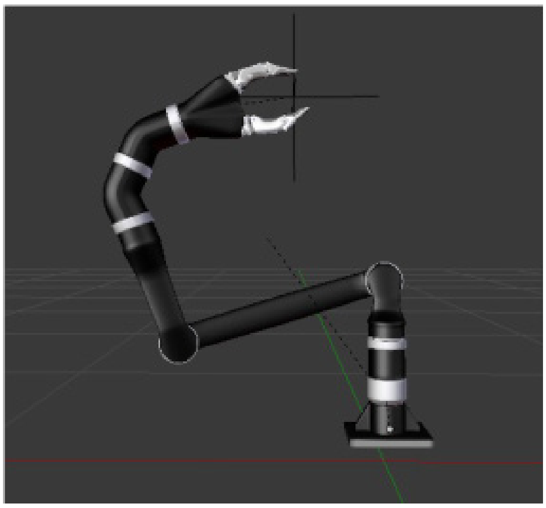


Figure 1 The simulated JACO robotic arm.

Create the armature: Armatures are the MORSE way to simulate kinematic chains made of a combination of re volute joints (hinge) and prismatic joints (slider). Kinematic chains are what Blender calls armatures. Armatures are made of bones. The pose of a bone is stored

Citation: Majeed SR, Kuhnert KD. Simulated robotic arm control using sound recognition system commands. *Int Rob Auto J.* 2017;3(1):250–256.
DOI: 10.15406/iratj.2017.03.00044

as the bone's pose channel. A bone is both a joint and the rigid segment attached to it. We often use the term joint (in the documentation and in MORSE code) either for a bone or its channel.[12]

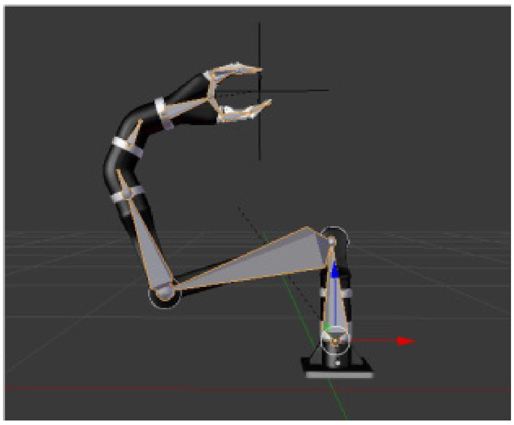


Figure 2 The bones added to the JACO robotic arm.

Actuator type of the armature: An actuator to manipulate Blender armatures in Morse, This actuator offers two main ways to control a kinematic chain: either by setting the values of each joint individually via a continuous data stream or via dedicated services, or by placing the end-effector and relying on a inverse kinematic solver. To use inverse kinematics, you must first define IK targets to control your kinematic chain According to this requirements the armature will be controlled using trajectory action controller using ROS.[12]

Armature pose sensor: The sensor streams the joint state (i.e, the rotation or translation value of each joint belonging to the armature) of its parent armature. This sensor must be added as a child of the armature, the data structure on data stream exported by the armature sensor depends on the armature. It is a dictionary of pair (joint name, joint value). Joint values are either radians (for re volute joints) or meters (for prismatic joints)[12] after simulated and add the required actuators and sensors, there are two ways to control the arm motion is by using IK or trajectory controller, in this research we used IK. The user can give the target position then the end -effector drives there. The controller node has been built in Ros as shown in the Figure 3. The Ik technique has been programmed in Ros node (JACO) a shown in Figure 3 this node responsible for two main functions:

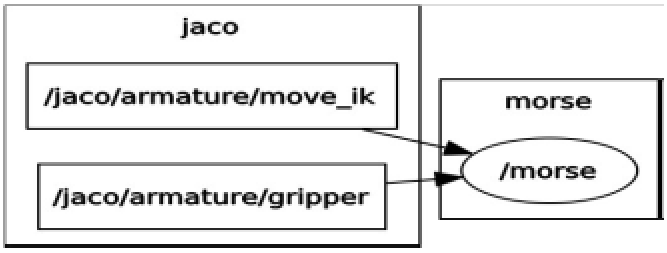


Figure 3 The Ros schematic for the IK control node.

The gripper function: The Ros topic on the branch/jaco/armature/gripper, this topic will receive two messages TRUE for grip the object and FALSE for release the object. The IK function: the Ros topic on the branch/jaco/armature/move_IK the received message will be classified into options:

a. Name of the IK target (this will be as string refer the name of the target).

b. The translation vector (x,y,z) which represented the translate position of the target.

c. Rotation vector (rx, ry, rz) represented rotational position for the target, the angels in radian.

d. Relative: specified the translation and the rotation relative to the actual target movement (the default option is true).

e. Linear speed (m/s) the linear speed for the inverse kinematic target.

f. Rotational speed (rad/s) the rotational speed for the inverse kinematic target (Figure 3).

The sound recognizer system

The sound recognition system is the core of this research and how to implement it with more accurate and effective way and as understood with more details and examples from the[14] we used the same algorithms with some editing according the research demand. This will explained and applied as follow.

The sound system will consist of three main subdivided scripts:

a. The main script: will run the node to publish the messages on the specific topic also will be like the administer which arranges the function. Also responsible to launch the node of the recognition system.

b. The dictionary script (.doc): in this script contains the words as real pronounced and as follow in our research:

c. MOVE M UW V

d. STOP S T AA P

e. The static language model script (.lm) : this is most important part of the our sound system, it is a type of the language models As shown below:

-1.2923 MOVE -0.2543

-1.9912 STOP -0.2341

The language model

The main part of the sound recognition system, it contains the decoder of the words that could be recognized, there are different types of the model language like keywords list, grammars and static language mode which used in our system.

Keywords list: is a type of the language model, the principle of this model is specified threshold for every word, this words can be detected easily through the speech. The threshold is different between the long key-phrase and the short key-phrase. Keywords list also has supported pocket sphinx. There is a way to specify the threshold, however for more information about this could be searched.

Grammars: using this model would be easy in the control and the commands, also there is no many options for the word sequences I.e the word has just one or two possible option to be detectable . This method needs to specify the input data precisely and carefully, if the user unintentionally did some mistakes and leave some words without the correct grammars, the recognition process will be failed, and the grammars have usually extension. Gram, jsgf. The user should avoid the using of complex grammars and the phrases with complex rules, this will take much time through the recognition process, because this the process may subject to fail.

Static language model: In this model the design consist of many words or combination of words and it is allowable to edit these words very easily so that it very recommended using this model in the design of the sound recognition system. The user can simply treats with this model through saying everything in a normal language then these words could be defined, stored and programmed with simple engineering effort. From the above properties the static language model is using in the new generation of the sound recognition interfaces because this interfaces when dependents on the natural language becomes more effective, using of this model to escape from the language of the commands and controlling which represented the old versions or generations. There are many methods for creating the static language model and it is depended on the data set size or the application:

a. Small data: using on-line quick web service.

b. Large data: using CMU language modeling toolkit.

c. Sometimes need to build a favorite toolkit: using ARPA.

d. The static language model could be saved into three formats:

Text ARPA: It will take high storage space but in the other hand it's very easy and possible to edit it, the extension will be .lm.

Binary BIN: Will take less space and faster loaded but it is hard to edit; the extension will be .lm .bin.

Binary DMP: It is not recommended to use because it is very hard.

It is allowable to change between the formats and change form one format to another. To build the static language , need first to prepare the text or the words that will be delectable then training the ARPA which can be done using many toolkits like SRILM , CMUCLMTK , IRSLM and MITLM the using of SRILM is very recommended because it represented very advanced toolkit , the training would be as follow: ngram-count-kndiscount-interpolate-text train-text.txt-lm arm move .lm the user can reduced the size of the model : ngram -lm armmove.lm -prune 1e-8 -write-lm reduced armmove.lm. The last step is after finishing the training, the model should be tested: ngram -lm armmove.lm -ppl test-text.txt.

The easiest way to build the static language model using web service. This method is recommended if the user want a simple and effective system without the complex control commands and in case the language is English in this case a file .txt should be created contains the required words in our project (move, stop) then it would be loaded into website LM Tool page. This page has the capability to create the file .dic and .lm then download it. After that a launch script has been programmed to be the guide for the information cycle between the two scripts. Then a program inside ROS has been written and programmed to create the sound recognition node, this node as wrote consists of the main three parts. There is a pocket in Ros for defining everything regards to sound for example the microphone and how to connect to the program, the data line how to received the sound and how to deals with it. This pocket sphinx is responsible for that, this pocket usually needs the model and dictionary file to make the access to the sound recognizer which has been programmed to recognized the specific words. As shown in the Figure 4 the Ros node recognizer consists of two parts one for the processing and the other for the output which will connect between the Ros node output and the output environment. The whole system in the ros schematic will be drawn as shown in the Figure 5, the combination of this two system will give us the flexibility to control the robotic arm with the sound commands.

The instructions cycle would be explained as follow and as shown in the Figure 6:

a. The word comes from the microphone and processes in the sound recognition system to recognize it.

b. After recognition, for every word there are specific target coordinates. The word move means that the target would have specific random coordinates that described in the position and orientation, stop means the coordinates are the same for the first case so the arm won't move because it is already in the correct position.

The data cycle from the beginning to the output in the environment has been shown in this figure, the Figure 7 & 8 represented the output of the recognizer Ros node when word move or stop come from the microphone. In this Figure 9 the simulated JACO arm is moving after the word (move) comes from the microphone in the number (1) then stop the moving because the word (stop) comes in the number (2) and still doesn't move in the number (3) at the end in the number (4) the word (move) comes again so that the arm is moving.

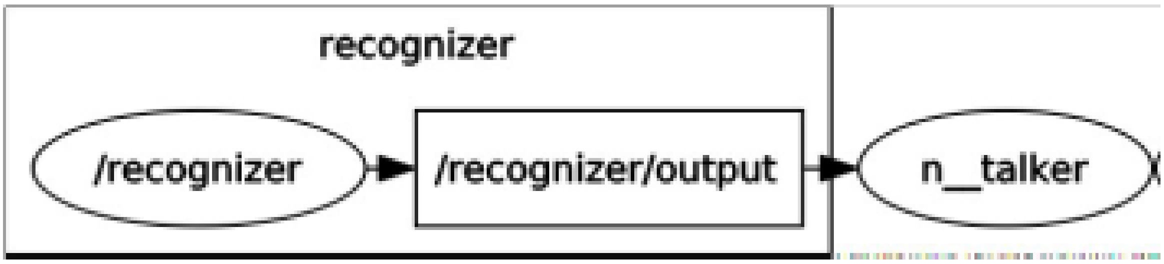


Figure 4 The Ros schematic for the sound recognizer.

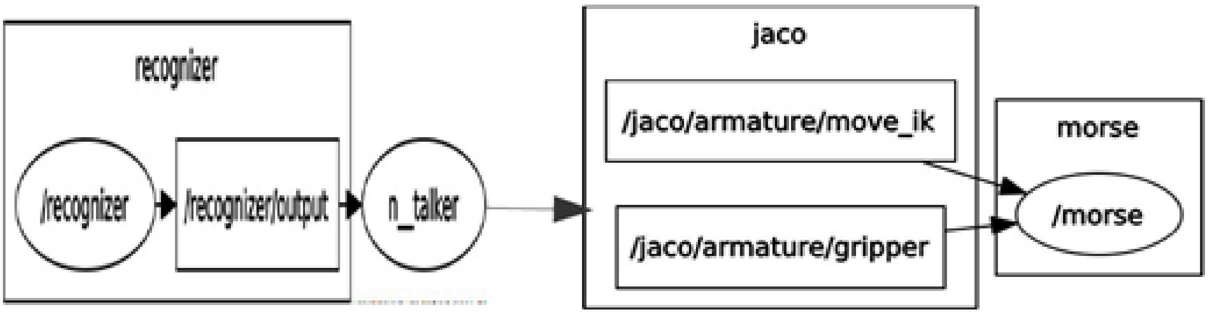


Figure 5 The Ros schematic for the whole system.

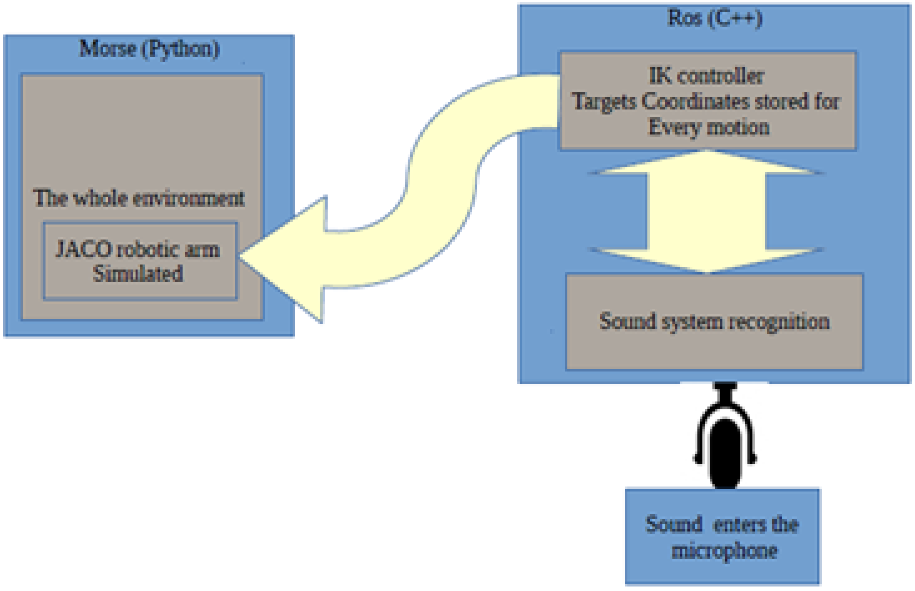
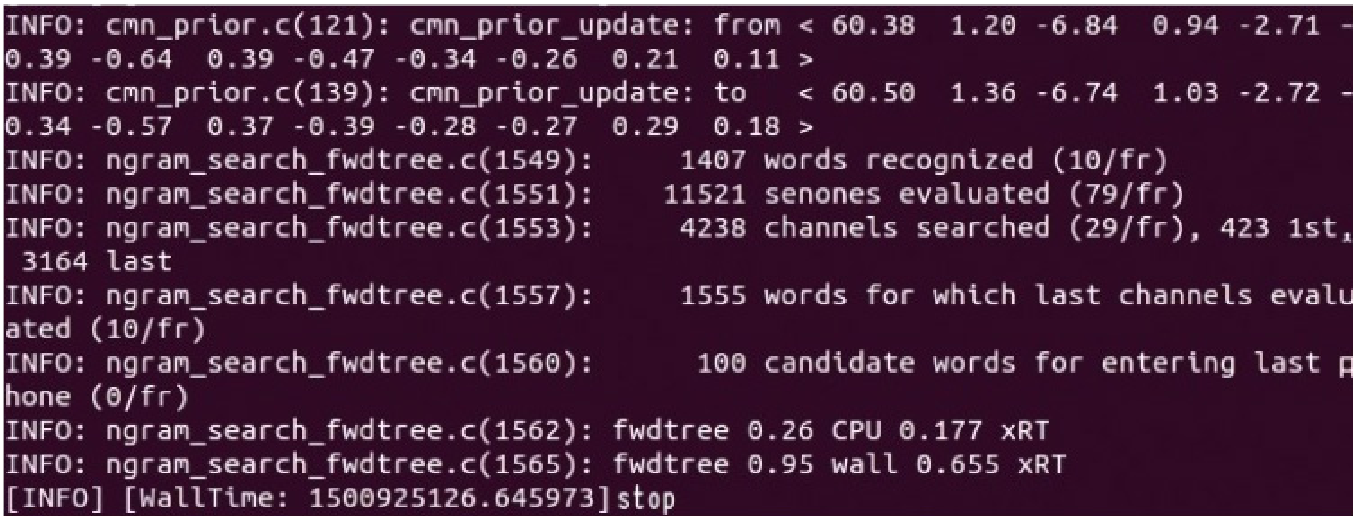
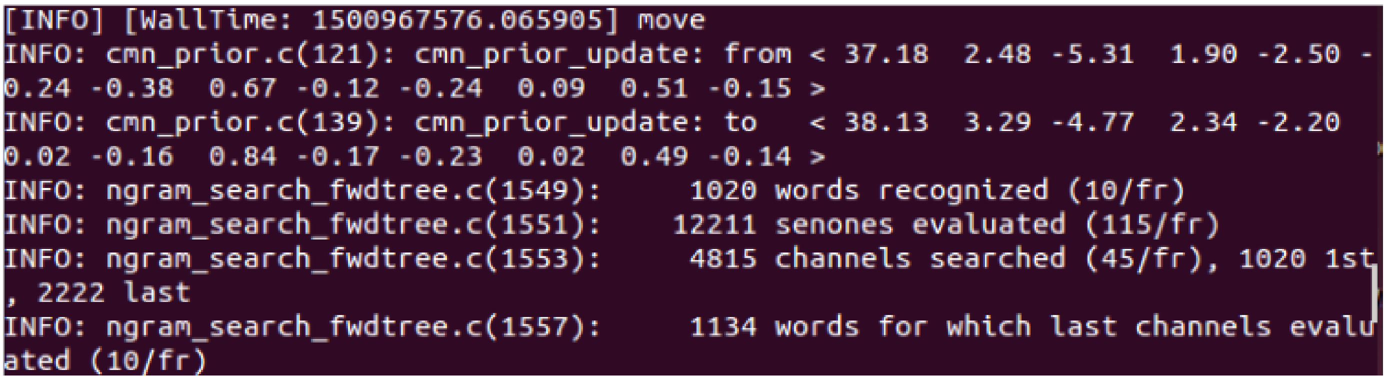


Figure 6 The data cycle for the whole system.



```
INFO: cmn_prior.c(121): cmn_prior_update: from < 60.38  1.20 -6.84  0.94 -2.71 -
0.39 -0.64  0.39 -0.47 -0.34 -0.26  0.21  0.11 >
INFO: cmn_prior.c(139): cmn_prior_update: to   < 60.50  1.36 -6.74  1.03 -2.72 -
0.34 -0.57  0.37 -0.39 -0.28 -0.27  0.29  0.18 >
INFO: ngram_search_fwdtree.c(1549):     1407 words recognized (10/fr)
INFO: ngram_search_fwdtree.c(1551):    11521 senones evaluated (79/fr)
INFO: ngram_search_fwdtree.c(1553):     4238 channels searched (29/fr), 423 1st,
 3164 last
INFO: ngram_search_fwdtree.c(1557):     1555 words for which last channels evalu
ated (10/fr)
INFO: ngram_search_fwdtree.c(1560):      100 candidate words for entering last p
hone (0/fr)
INFO: ngram_search_fwdtree.c(1562): fwdtree 0.26 CPU 0.177 xRT
INFO: ngram_search_fwdtree.c(1565): fwdtree 0.95 wall 0.655 xRT
[INFO] [WallTime: 1500925126.645973]stop
```

Figure 7 The Ros recognizer output (stop).



```
[INFO] [WallTime: 1500967576.065905] move
INFO: cmn_prior.c(121): cmn_prior_update: from < 37.18  2.48 -5.31  1.90 -2.50 -
0.24 -0.38  0.67 -0.12 -0.24  0.09  0.51 -0.15 >
INFO: cmn_prior.c(139): cmn_prior_update: to   < 38.13  3.29 -4.77  2.34 -2.20
0.02 -0.16  0.84 -0.17 -0.23  0.02  0.49 -0.14 >
INFO: ngram_search_fwdtree.c(1549):     1020 words recognized (10/fr)
INFO: ngram_search_fwdtree.c(1551):    12211 senones evaluated (115/fr)
INFO: ngram_search_fwdtree.c(1553):     4815 channels searched (45/fr), 1020 1st
, 2222 last
INFO: ngram_search_fwdtree.c(1557):     1134 words for which last channels evalu
ated (10/fr)
```

Figure 8 The Ros recognizer output (move).

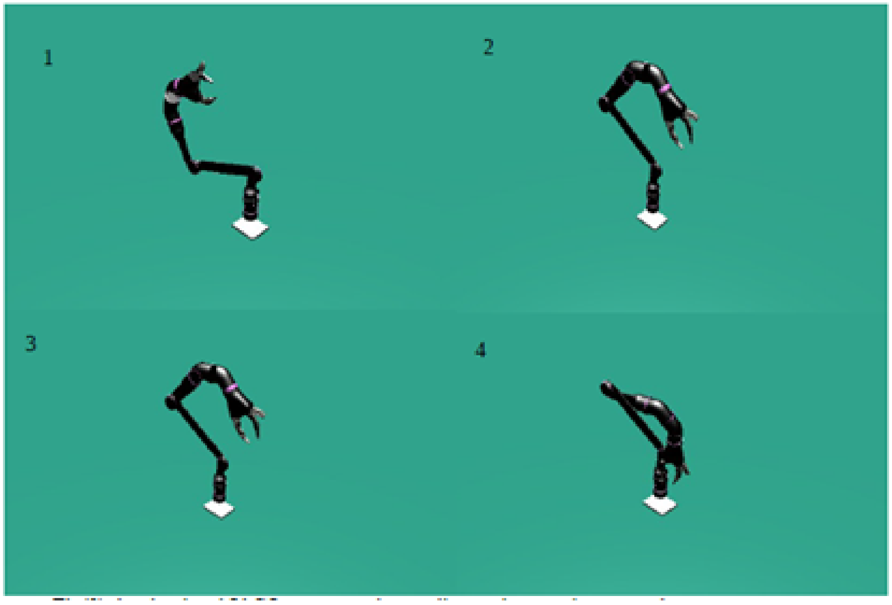


Figure 9 The simulated JACO arm moved according to the sound commands.

Conclusion

In this paper we supposed a new sound recognition system using (ROS). The robot arm has been simulated and defined physically using MORSE program, also this arm has been programmed with its functions and moving strategy by Python. The middle ware for controlling the sound system was ROS, this controlled the arm functions topics and messages. This sound system connected in ROS with the arm moving topics and programmed using C++. The sound system has been tested experimentally; it proved that this system is very accurate and effective system.

Acknowledgements

This research is a part of whole project involved smart eating table for handicapped people, the sound recognition system has been used with another words then tested.

Conflict of interest

Author declares that there is none of the conflicts.

References

1. Salih Rashid Majeed, Klaus D Kuhnert. Object Detection based on Semantic Camera for Indoor Environment. *International Journal of Advance Robotics & Expert Systems (JARES)*. 2016;1(1):1–10.
2. Gilberto Echeverria, Nicolas Lassabe, Arnaud Degroote, et al. Modular open robots simulation engine: Morse. *Robotics and Automation (ICRA), IEEE International Conference*. 2011.
3. Muhammed Zahit Özdemircan. *Robot Control with Voice Command*. Yildiz Technical University. 2008. p. 1–48.
4. Aswinbalaji R, Arunraja A. Wireless Voice Controlled Robotics Arm. *International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)*. 2015;12(4):1–6.

Citation: Majeed SR, Kuhnert KD. Simulated robotic arm control using sound recognition system commands. *Int Rob Auto J*. 2017;3(1):250–256. DOI: 10.15406/iratj.2017.03.00044

5. Zakariyya Hassan Abdullahi, Nuhu Alhaji Muhammad,Jazuli Sanusi Kazaure, et al. Mobile Robot Voice Recognition in Control Movements. *International Journal of Computer Science and Electronics Engineering (IJCSEE)*. 2015;3(1):1–6.

6. Brandi House, Jonathan Malkin, Jeff Bilmes. The VoiceBot: A Voice Controlled Robot Arm. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2009. p. 183–192.

7. David Be, Cinhtia González, Manuel Escalante, et al. Wireless Control LEGO NXT robot using voice commands. *International Journal on Computer Science and Engineering (IJCSE)*. 2011. p. 1–9.

8. Praveen Blessington, Madhav BTP, Sagar Babu M, et al. Acoustic Controlled Robotic Vehicle. *International Journal of Emerging Technology and Advanced Engineering*. 2012;2(4):1–6.

9. Jayesh Chopade, Dattatray Barmade, Swapnil Tonde. Control of Spy Robot by Voice and Computer Commands. *International Journal of Advanced Research in Computer and Communication Engineering*. 2013;2(4):1–3.

10. Ekapol Chuangsuwanich, Scott Cyphers, James Glass, et al. Spoken Command of Large Mobile Robots in Outdoor Environments. *MIT Computer Science and Artificial Intelligence Laboratory*. 2010. p. 1–6.

11. Salih Rashid Majeed, Klaus Kuhnert D. Simulation of the Inverse–Kinematics for JACO Manipulator Robot Arm. *International Journal of Robotics and Automation*. 2016.

12. Salih Rashid Majeed, Klaus Kuhnert D. Trajectory Controller Building for the (KUKA, JACO) simulated manipulator robot arm Using ROS. *International Journal of Robotics and Automation*. 2017.

13. Salih Rashid Majeed, Klaus Kuhnert D. Automatic Skinning of the Simulated Manipulator Robot Arm. *International Journal of Computer Graphics & Animation (IJCGA)*. 2016.

14. Training an acoustic model for CMUSphinx.

Citation: Majeed SR, Kuhnert KD. Simulated robotic arm control using sound recognition system commands. *Int Rob Auto J.* 2017;3(1):250–256.
DOI: 10.15406/iratj.2017.03.00044