Design and Implementation of Self Learning Autonomous Robot using Neural Networks under ROS (Robot Operating System) Platform

Salam Al-Khammasi, Adnan Qayyum, Prof. Tarek Sobh
Computer Science and Engineering Department, University of Bridgeport, Bridgeport, CT 06604, USA, (salkhamm, aqayumm)@mybridgeport.edu, sobh@bridgeport.edu

Abstract
In this poster, we are designed and implemented an avoidance obstacle robot using ROS (Robot operating system) as main platform. Neural Network algorithm has been used to program the robot. The algorithm has been written in Python, C++, and Arduino. The sensor of the robot is given by Xbee from Maxstream, and the robot can detect where objects are placed. The robot can give the control of the program for the user. It's the main function of the robot. The robot can give a faster and slower speed for moving around obstacles. The robot can detect and avoid obstacles automatically. The robot can be used in many situations such as surveillance and traffic. The robot can also be used in the medical field for biomedical and robotic research. The robot can also be used in the future for industrial and household uses.

Hardware Implementation
For the hardware part, as shown in the figure the robot has been built with Arduino Uno board with ultra-sonic sensor. The ROS will minimize the coding and gives us relative results. The communication between the robot and the base station will be wireless.

Software Architecture
Nodes, Messages, & Topics
- Nodes: processes performing computation in the ROS system
- Topics: ROS communication graph
- Messages: packets of data sent between ROS nodes

ROS Packages

Goals of Using ROS
- Flexible:
  - Hardware:ROS friendly
  - Software:
    - Operating system: Linux
    - Programming language: C++, Python

Proposed Work and Algorithm
We have implemented the algorithm that the robot learns internal model of the environment by recurrent neural network, it predicts succession of sensors inputs and on the base of the model it generates navigation steps as a motor commands. Robot will collect the data from its environment and saved all the instances of collected data for the situation if it ever comes again. Therefore we use the sensor data from the environment and the classical first object problem in our strategy was transform to the procedure ‘learning your environment’. The robot has in any position will give environment information about its distances to the all objects in this workspace. We use this information in neural network that learns these situations and in any position gives the free segment of space for safe and fastest path as output.

Conclusion
In conclusion, our team came up with a new modern robot to avoid the obstacles avoidance by using the new robotic platform (ROS) and our team enhances the time consumption to program the robot itself. In addition, our contribution will be how the robot will learn the environment with the neural network algorithm. Also, in this project, we reduced the time consumption for the programmer.

Future work
As future work for this project. Any programmers can apply his own code or algorithm on this robot by just adding it to the platform like, mapping, navigation, surveillance and made small modification on the hardware part.

Acknowledgment
Some slide content is adapted from the ROS.org Wiki under the Creative Commons Attribution 3.0 license. Also, some of the figure is adapted from the Lincoln laboratory at MIT.

Figure 1: Robot application sing ROS

Figure 2: The increment in the number of packages and the number of nodes supporting by ROS

Figure 3: Self-Learning autonomous robot with Ultra-Sonic Sensor

Figure 4: ROS Graph, Topic, and Plot Plugins

Figure 5: Global Planning

Figure 6: 2D Planning

Figure 7: Motion Planning

Figure 8: Move, Base Recovery procedure

Figure 9: ScreenShot