International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-1S, October 2019

# ROS Based Autonomous Robot with Open CV

**Danish Faraaz S, Desmond Savio Rozario, K. Arun Vasantha Geethan**

***Abstract— In this project, we have designed and developed an autonomous robot that is powered by Robot Operating System (ROS). The capabilities of the robot include autonomous navigation, image tracking and mapping. OpenCV has been implemented in the on-board microprocessor to process the images that are captured by the general purpose webcams on the robot. A microcontroller has also been used to control the motors. The ultimate aim of this project is to develop a mobile robot capable of making its own decisions based on the images received.***

***Keywords – Autonomous, Robot, ROS, Tracking***

## I. INTRODUCTION

Any robot that can be used to perform certain tasks or behaviors without human interference can be called as an autonomous robot. A fully autonomous robot can navigate its environment and act as required. With an ever increasing need for a much efficient and productive work-place, autonomous robots have been used in many industries. Still, there are a number of obstacles that must be overcome before autonomous robots can be used extensively. Communication between different robots is difficult because they robot must act independently on its own while communicating. Another problem is that of mapping unknown environments. While the mapping capabilities of robots have been evolving, robots must learn to work around un-mapped and poorly understood environments.

Robot Operating System (ROS) is a framework of a set of tools that provides functionality for communication between a number of machines, simulation, visualization and much more. The advantage of ROS lies in the way it works. ROS provides a way of connecting a number of nodes with a central master node. It had multi-platform support and can be coded in either C++ or Python. C++ is preferred due its better performance. ROS is an open source platform and hence can used for both commercial and personal purposes freely. The version of ROS that we are working of is ROS Kinetic which was released on May, 2016 and has long term support until April, 2021.

Open Source Computer Vision Library (OpenCV) is another open source platform that has C++, Python and JAVA compatibility.

**Manuscript published on 30 October 2019.**
***** Correspondence Author (s)
**Danish Faraaz S**, The Department of Mechanical Engineering, St. Joseph's Institute of Technology, Chennai, India.
**Desmond Savio Rozario**, The Department of Mechanical Engineering, St. Joseph's Institute of Technology, Chennai, India.
**Dr. K. Arun Vasantha Geethan**, The Department of Mechanical Engineering, St. Joseph's Institute of Technology, Chennai, India.



They have numerous algorithms for a number of tasks such as tracking images, tracking moving objects, detect faces and so on. The library is used widely for research and personal purposes.

## II. MECHANICAL DESIGN

The design of the robot was inspired by the Turtlebot 2 which is the world's most popular robot used for education and research. The robot is fabricated using acrylic which was selected for its ease of machining and low cost. It is designed to carry a maximum payload of 3Kg.

The 3d modeling of the robot was done using Siemens NX (see Fig. 1). The robot has three plates that are connected to each other using rods. Since the base plate carries the major portion of the weight, it is 12mm thick, whereas the other two plates are 8mm in thickness.

Each of the rods is 200mm in length and 20mm in diameter. The space between the plates is used to house various electronic components such as the motors, motor driver, microcontroller, the microprocessor and sensors.

**Fig. 1. Isometric view of the model of the robot**

Two 100 rpm Dc geared motors are used with a torque of 20Kgcm. Another two castor wheels are used to stabilize the motion of the vehicle. A differential drive model is used in the vehicle which uses the independent rotations of the wheels to turn the robot left, right or just go straight. The equation for a differential drive system is shown below [9].

$$\dot{x} = \frac{R}{2}(v_r + v_l)\cos\phi$$
$$\dot{y} = \frac{R}{2}(v_r + v_l)\sin\phi$$
$$\dot{\phi} = \frac{R}{L}(v_r - v_l)$$

**Fig. 2. Equations of differential drive robot**

## III. ELECTRONIC SETUP

### *A. Microprocessor*

The microprocessor used for this setup is the Raspberry Pi 3b+. The Robot Operating System (ROS) is run on this microprocessor since it has a Linux OS installed on its micro SD card. This is because ROS works only on Linux and doesn‘t work on Windows. It is necessary to have sufficient space in the SD card so that the program works without any lag. The components connected to the board are the cameras and a power supply. A monitor can be connected to the Raspberry pi to run the program.

But since, it is not convenient to connect a monitor with the board; we use our laptop to SSH into the Raspberry pi. SSH is a process wherein we can have temporary access to the Raspberry pi to run the programs installed in it (See Fig, 3) using the IP address of the board. The VNC software then acts as a monitor and shows the programs inside the Raspberry pi.

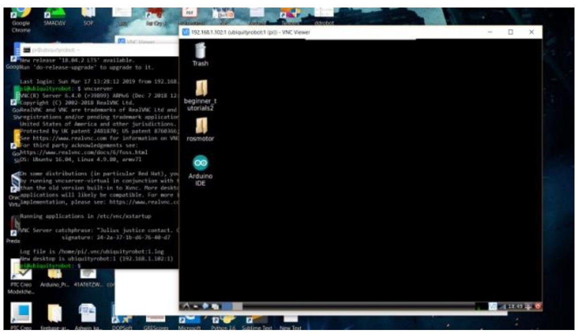

**Fig. 3. Using SSH to connect with the microprocessor**

### *B. Micro-controller*

The Arduino Uno board acts as the microcontroller in the vehicle. One of the key uses of the microcontroller is to reduce the load on the microprocessor. The Arduino controls the speed and direction of the motors through a motor driver which has been connected to it. In addition to these, the micro-controller can also be used to display messages inside the command prompt or the terminal depending on the direction of the robot.

### *C. Image Processing*

The image processing has been achieved using OpenCV. The images captured by the webcams are processed using the given code and the shape of the objects is detected. Depending on the use, the processing can be extended to pattern matching, edge detection and so on. In this project, we are using shape detection techniques to detect the shapes and to move the robot accordingly.

### *D. Robot Operating System*

ROS performs the core functions of the robot such as running the codes, publishing and receiving messages, displaying the output and manipulating the robot. ROS uses a number of nodes that publish and subscribe to certain topic that contain the necessary messages. The nodes and topic running during the program can be visualized using the Rqt graphs (See Fig. 4).

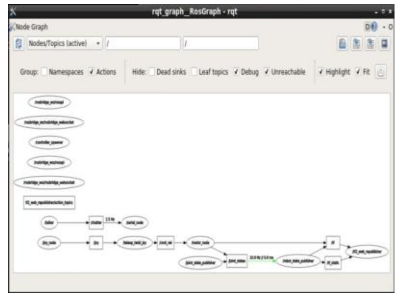

**Fig. 4.** Rqt Graph of the program

## IV. WORKING

The images detected by the two webcams are processed by OpenCV and then the information is sent to the publisher. ‘RECT’ is used to when a rectangle is detected and ‘CIR’ is used when a circle is detected (See Fig. 5).

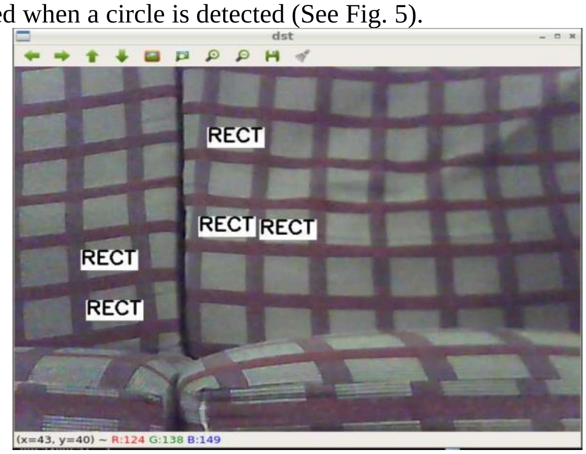


**Fig. 5. Shape detection by OpenCV**

According to the shape detected, the image is sent to the Publisher which displays the shape with the help of integers. The Publisher code works on the microprocessor and displays the messages in the Linux Terminal. A value of ‘0’ means no shape is detected whereas a value of ‘1’ and ‘2’ means a rectangle and a circle has been detected respectively (See Fig. 6).

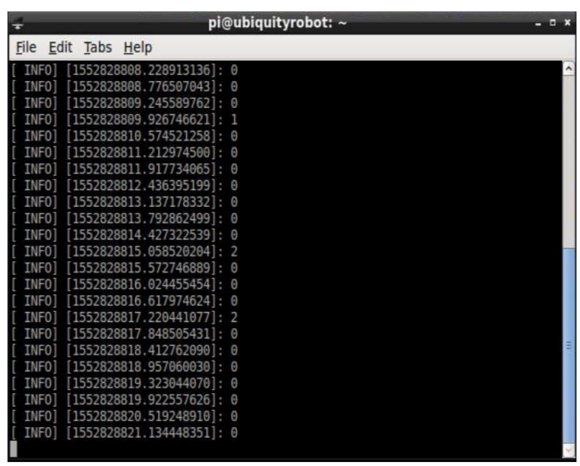


**Fig. 6. Working of the Publisher**


*Published By:*
*Blue Eyes Intelligence Engineering*
*and Sciences Publication (BEIESP)*

The next step is that of the Subscriber that receives the value from the Publisher and manipulates the robot accordingly. The Subscriber is run on the Arduino and is also used to the display the direction of the robot according to the value received (See Fig. 7).

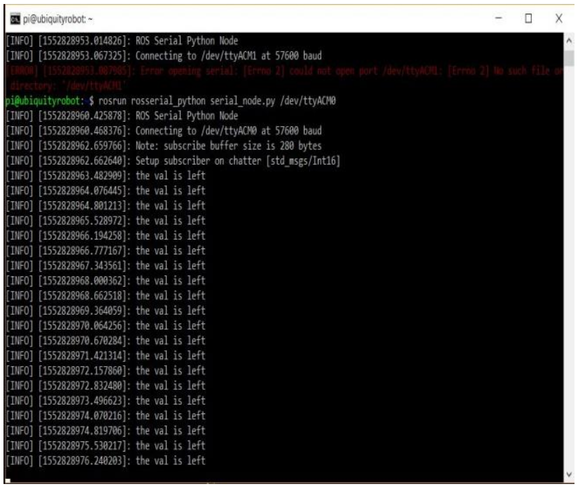


**Fig. 7. Subscriber displaying the direction**

## V. IMPLEMENTATION

Hence, the autonomous robot has been implemented using the techniques discussed above (See Fig. 8). Using ROS has enhanced the smooth working of the system and processing. The robot works as intended and we can develop it further according to our needs.

**Fig.8.Implementationo**

## REFERENCES

1. Haidi Jaber, Dr.Luc Jaulin, "Robot Path Planning using Interval Analysis", ENSTA Bretagne/OSM, 2012.
2. R. Negenborn, "Robot Localization and Kalman Filters". Master's Thesis, Utrecht University, 2003.
3. A. Kelly. "General solution for linearized systematic error propagation in vehicle odometry in Intelligent Robots and Systems", 2001. Proceedings. 2001 IEEE/RSJ International Conference on, volume 4, pages 1938 - 1945 vol.4, 2001.
4. Hugh Durrant-Whyte and Tim Bailey,
5. ―Simultaneous Localization and Mapping (SLAM), Part I The Essential Algorithms‖, IEEE Robotics & Automation Magazine, June 2006.
6. Dissanayake, M.W.M.G.; Newman, P.; Clark, S.; Durrant-Whyte, H. F.; & Csorba, M. (2001). ―A solution to the simultaneous localization and map building (SLAM) problem.‖ Robotics and Automation, IEEE Transactions on, 17(3), 229- 241.
7. André Gonçalves Araújo, ―ROSint - Integration of a mobile robot in ROS architecture‖, University of Coimbra, July 2012.
8. S.Roland, Illah R. Nourbakhsh, Introduction to Autonomous Mobile Robots, Text Book, pp. 12- 45, 2004.
9. Aaron Martinez, Enrique Fernandez, " Learning ROS for Robotics Programming", Text Book, pp. 7-9, September, 2013.
10. Video Tutorial by Dr. Magnus Egerstedt, Georgia Institute of Technology - Control of Mobile Robots
11. ROS Tutorial, ROS Documentation, Official Website: http://www.ros.org/
12. OpenCV Tutorial, OpenCV Documentation, Official Website: https://opencv.org/
13. B Abhishek; S Gautham; D Varun Rufus Raj Samuel; K Keshav; U.P. Vignesh; Shyam R Nair, ―ROS based Stereo vision system for autonomous vehicle,‖ Robotics and Automation (ICRA), 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPSI).

## AUTHORS PROFILE

**Danish Faraaz.S** is a final year student in the Department of Mechanical Engineering at St.Joseph's Institute of Technology for the batch of 2015 – 2019. His research interests include autonomous robots with SLAM, control theory and mobile robots.

**Desmond Savio** Rozario is a final year student in the Department of Mechanical Engineering at St.Joseph's Institute of Technology for the batch of 2015 – 2019. His research interests include 3D designing and optimization.

**Dr. K. Arun Vasantha** Geethan is a professor and Head of the Department in the Department of Mechanical Engineering at St.Joseph's Institute of Technology. He has more than 25 publications to his credit under various fields such as supply chain management and eco waste.