Intelligent Turtlebot Gripper

Desiree dos Santos
LSA/FACIN
PUCRS/Brazil
desiree.santos.001@acad.pucrs.br

ABSTRACT

Nowadays technology in most cases and robotics system in particular search for explore [2] and manipulate their environment have many potential applications such as search and rescue, remote inspection, or planetary exploration (e.g. [Mehling et al., 2007, Deegan et al., 2006, Guizzo and Deyle, 2012, Stilman et al., 2009, Borst et al., 2009, Schenker et al., 2003]). This paper presents how a pick and place robot can be designed with a low cost hardware and robust embedded software. Various problems and obstructions for the loading process has been analyzed and been taken into consideration while designing the pick and place robot.

INTRODUCTION

Demand on autonomous robots is currently rising. Robotics is the branch of engineering science and Technology related to robots, and their design, manufacture, application, and structural disposition. Robotics [9] is related to electronics, mechanics, and software [7, 6] and the current research today is focused on developing systems that exhibit modularity, flexibility, redundancy, fault-tolerance, a general and extensible software environment and seamless connectivity to other machines, some researchers focus on completely automating a manufacturing process or a task, by providing sensor based intelligence to the robot arm, while others try to solidify the analytical foundations on which many of the basic concepts in robotics are built. One type of robot commonly used in industry is a robotic manipulator or robotic arm. The pick and place robot is a microcontroller [5] based mechatronic system that detects the object, picks that object from source location and places at desired location. This environment includes robotics applications in a modular and flexible to answer the demands of a wide range of problems in an efficient manner to increase productivity and to deliver uniform quality. The pick and place robot is a microcontroller [5] based mechatronic system that detects the object, picks that object from source location and places at desired location. For detection of object, infrared sensors are used which detect presence of object as the transmitter to receiver path for infrared sensor is interrupted by placed object.

COMPONENTS OF PICK AND PLACE ROBOT:

The use of robots for placing products in cartons and transfer of cartons and products between different stations in the packaging lines is very common in all industries. High speed pick-and-place robots (figure 1) for placing small items like candy and cookies in packages are often combined with a visual observation system for identifying products.

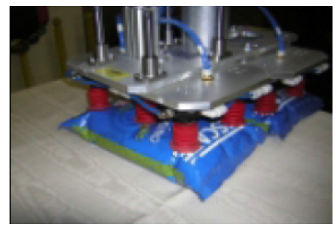

Figure 1. Pick and place robot

- **Structure:** The structure of a robot is usually mostly mechanical and can be called a kinematic chain. The chain is formed of links, actuators, and joints which can allow one or more degrees of freedom. Most contemporary robots use open serial chains in which each link connects the one before to the one after it. These robots are called serial robots and often resemble the human arm. Robots used as manipulators have an end effector mounted on the last link. This end effector can be anything from a welding device to a mechanical hand used to manipulate the environment.
- **Power Source:** At present mostly (lead-acid) batteries are used, but potential power sources could be: Pneumatic (compressed gases). Hydraulics (compressed liquids), Flywheel energy storage, Organic garbage (through anaerobic digestion), Still untested energy sources (e.g. Nuclear Fusion reactors)
- **Vision:** Computer vision is the science and technology of machines that see. As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences and views from cameras. In most practical computer vision applications, the computers are pre-programmed to solve a particular task, but methods based on learning are now becoming increasingly common. Computer vision systems rely on image sensors which detect electromagnetic radiation which is typically in the form of either visible light or infra-red light. The sensors are designed using solid-state physics. The process by which light propagates and reflects off surfaces is explained using optics. Sophisticated image sensors even require quantum mechanics to provide a complete understanding of the image formation process.
- **Manipulation:** Robots which must work in the real world require some way to manipulate objects; pick up, modify,

destroy, or otherwise have an effect. Thus the 'hands' of a robot are often referred to as end effectors, while the arm is referred to as a manipulator. Most robot arms have replaceable effectors, each allowing them to perform some small range of tasks. Some have a fixed manipulator which cannot be replaced, while a few have one very general purpose manipulator, for example a humanoid hand.

1. Mechanical Grippers: One of the most common effectors is the gripper. In its simplest manifestation it consists of just two fingers which can open and close to pick up and let go of a range of small objects. Fingers can for example be made of a chain with a metal wire run trough it.
2. Vacuum Grippers: Pick and place robots for electronic components and for large objects like car windscreens, will often use very simple vacuum grippers. These are very simple astrictive devices, but can hold very large loads provided the pretension surface is smooth enough to ensure suction.

EMBEDDED HARDWARE

This section presents all hardware elements used in pick and place robot built.

- **TurtleBot**
  It is a low-cost, personal robot kit with open-source software. It consists of a mobile base, 2D/3D distance sensor, laptop computer or SBC(Single Board Computer), and the TurtleBot mounting hardware kit. The TurtleBot's core technology is SLAM and Navigation,can run SLAM(simultaneous localization and mapping) algorithms to build a map and can drive around your room. Also, it can be controlled remotely from a laptop, joypad or Android-based smart phone. [10]. We're using the version TurtleBot2 (fi) is the second generation of the TurtleBot following within the REP 119 specification

Figure 2. TurtleBot2 is the second generationt

- **Raspberry Pi**
  The RPi(Raspberry Pi) is a credit card size computer that can be used as a common personal computer and/or in ubiquitous systems projects [11]. The main motivation was the use of the Python language. It considered a very high level language because of its simple syntax and its dynamic typing. Moreover,it is an interpreted language which makes it a great option for scripting and has a wide range of features for embedded systems.[3]

  Regarding the raspberry PI another important items are is the low-price to prototype projects and microprocessor easy-to-use with a good documentation and tech community supported. Specification:

  – Full size HDMI
  – 40-pin extended GPIO
  – 1GB RAM, 4 USB 2 ports
  – Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
  – CSI camera port for connecting to plug raspicam
  – Upgraded switched Micro USB power source up to 2.5A
  – Micro SD port for loading operating system and storing data
  – BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board

Figure 3. Raspberry version 3

- **Sensor and Actuator**
  Sensor is a device that responds to a specific physical / chemical stimulus and can be transformed into another physical quantity for measurement and / or monitoring purposes. In this way, the sensor associated with a stimulus transformation module in a quantity for measurement and / or monitoring purposes can be defined as a transducer or meter, which converts one type of energy to another. In this project there are the sensors :

  – Ultrasonic HC-SR04: It is to get the distance between the robot and the object target;
  – Force Sensitive FSR 406: It is an analog sensor, responsible to define how much force the robot can do to hold an object;
  – Current ACS712: It is responsible to define how much force the robot can do to hold an object;

  Actuator is an element that produces movement, attending to commands that can be manual, electrical or mechanical. These mechanisms, in general, transform the incoming energy (several natures) into movements that can be considered kinetic energy. In this project there are the sensors:

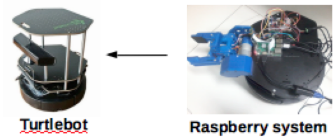

Figure 4. System elements

- 2 Servos motores Tower PRO 996R: They are to open/close the claw, so it holds an object;
- 1 Motor DC Pololu 50:1 37Dx54L: This claw motor will go up and down;

EMBEDDED SOFTWARE DESIGN

This section presents the software elements used in pick and place robot built.

- **ROS - Robot Operating System**
  Control software running in the standard desktop PC is developed on ROS where modules communicate using the publish-subscribe mechanism [8]. In parallel, several packages from ROS stacks are used for communicating with robot via TCP/IP communication protocol [1].

- **OpenCV - Open Source Computer Vision Library**
  OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous compute platform [4].

EXPERIMENTAL

The intelligent gripper robot is composed by two elements. First Turtlebot v2 and second raspberry with sensors, actuators and claw (figure 4). **Turtlebot** is the master side which is responsible for receving the instruction to move(left,right, forward and backward) the base(chassi) from place to other. The second part is composed by several elements: the arm printed in a 3D, raspberry pi version 3, raspicam, sensors and actuators listed before (figure 5). Computer vision is a strong element in this project, so from the raspicam V2(figura 6) integrated with a raspberry integrated we can process the images.

WORKFLOW: Raspicam package is responsible to get all image from raspicam integrated with raspiberry, those image are sending to topic and republished into the topic from *raspicam node* with type Image from sensor msgs. It's the start point to proccess, which the class Camera is responsible. To process image we're using the framework opencv version 2.4 to identify all yellow circle which the class Enum Color and

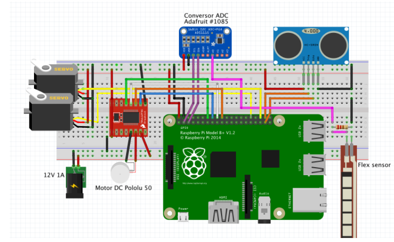

Figure 5. Raspberry system

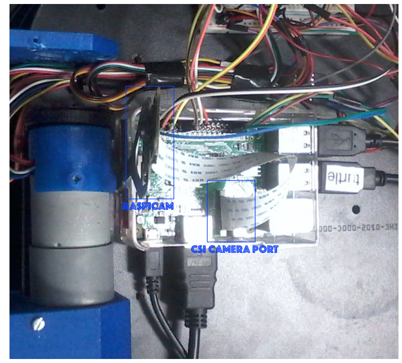

Figure 6. Raspicam

Image is responsible to do. The class robot has all logic decisions/sends commands to turtlebot to move (left/right) and the class raspberry to physical integrated with raspberry.

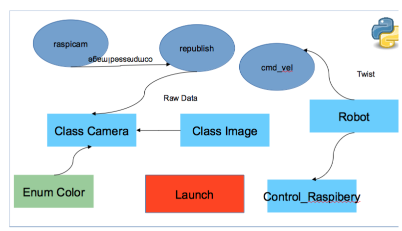

Figure 7. workflow

EXPERIMENTAL RESULT:

Raspicam node works very well, with a good results, as you can see the figure 8 in the left size box. The right size is a result of getting a mask of yellow objects.

REFERÊNCIAS BIBLIOGRÁFICAS

1. C., E. S. L. ROS-Industrial-Applying the Robot Operating System(ROS)to Industrial Applications. *IEEE International Conference on Robotics and Automation* (2012).
2. Jorge, S. NimbRo Explorer: Semi-Autonomous Exploration and Mobile Manipulation in Rough Terrain. *IEEE International Conference on Robotics and Automation* (2015).
3. Manoel Carvalho Marques Neto, E. M. J. J. LibsensorPy: A Library to Improve the Development of Ubiquitous Applications on Raspbery Pi. *International Conference on Computer Communication and Informatics* (2015).

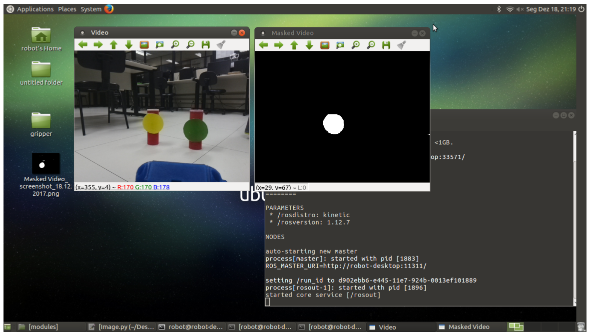

Figure 8. workflow

4. opencv. `https://opencv.org/`.

5. P., R. P. C. S. C. Project Report on Pick and Place robot.

6. P. Mandal, R. K. Barai, M. M. S. R., and Ghosh, S. Autonomous robot coverage in rescue operation. *International Conference on Computer Communication and Informatics* (2013), 1–5.

7. P. Mandal, R. K. Barai, M. M. S. R., and Ghosh, S. Basic research on the driving performance of an autonomous rescue robot with obstacles. *IEEE International Conference on Robotics and Biomimetics (ROBIO)* (2015).

8. Quigley M. Gerkey B Conley K. Faust J. Foote T. Leibs, J. B. E. W. R. A. Y. N., Ed. *ROS: an open-source Robot Operating System*, vol. 3 (2009).

9. Romero, R. A. F. *Robótica Móvel*. 1. 2014.

10. turtlebot. `http://www.turtlebot.com/about/`.

11. Upton, E., and Halfacree, G. . J. W. . S. *Raspberry Pi user guide*. 2014.