RoboVision

3D Vision for picking Robot with ROS

Valentin Ezekiev, Lucas van der Laan, Bogdan Patraș,
Diana Rusu, Rareș Sucălescu

Project Description

This is a project, funded, supervised and assisted by GTL(GreenTech Labs) and Fontys University of Applied Sciences in Venlo. The purpose of the project is to find out if ROS (Robot Operating System) can be used to develop a system that uses conveyors, a robot arm and a depth camera to establish a working robot-object-sorting automated system.

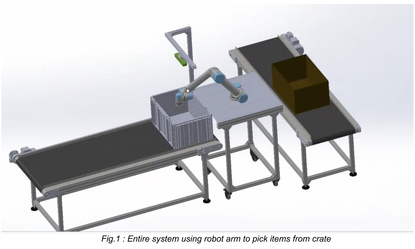

Fig.1 : Entire system using robot arm to pick items from crate

Problem description

At the moment, the picking solution involves an expensive camera, the Pickit3D camera. GTL wants to use a cheaper 3D IntelliSense camera for recognising parts in an auto store crate for order picking.

Furthermore, GTL wants to use the Robot Operating System (ROS) to speed up the development of a complete robot with the IntelliSense camera and gripper. A realsense2_description is available as a separate debian package of ROS distribution. The combination of this type of camera and the robotarm (Universal Robot, UR), however, is not tested in this combination.

Project goal

The Robovision project goal is to find out if Robot Operating System is suitable for building a cheaper version of the existing solution, which makes use of an expensive camera, the Pickit camera and is a black box system.

The point of the Robovision project, succinctly speaking, is to get a robot arm to pick up and put down objects effectively from box A into box B using a depth camera that is cheaper than it's industrial-standard counterpart. It's a sort of experiment for both cheaper solutions in depth cameras, as well as a testing grounds for the ROS(Robot Operating System) development environment in more practical fields. To put this to a list, it would go like this:

- Make the robot arm work with ROS code.
- Make the cheaper depth camera work with ROS code.
- Implement object recognition of the depth camera.
- Pick up and put down items from one box to another, using a robot arm.
- Document the entirety of this project(code, workflow diagrams, etc.) for later continuation.
- Learn and experience programming with ROS

Intel RealSense D435 Camera

The Intel® RealSense™ depth camera D435 is a stereo solution, offering quality depth for a variety of applications. It's wide field of view is perfect for applications such as robotics or augmented and virtual reality, where seeing as much of the scene as possible is vitally important. With a range up to 10m, this small form factor camera can be integrated into any solution with ease, and comes complete with the Intel RealSense SDK 2.0 and cross-platform support.

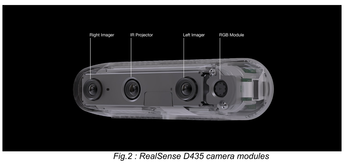

Fig.2 : RealSense D435 camera modules

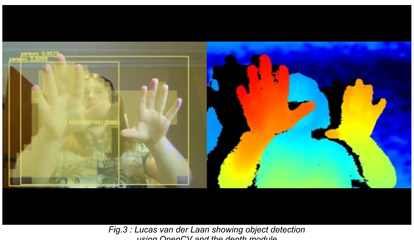

Fig.3 : Lucas van der Laan showing object detection using OpenCV and the depth module

Object recognition

The robot has the task to pick a defined item from the crate that comes along on the conveyer belt.The crate holds the items that the robot arm needs to pick and transfer in another place. In order for the system to know where to place the robot arm, it needs data on where the object is placed inside the crate.

Firstly, to get the data from the camera regarding the object and its location, the system has to recognize which object to inspect. For this, object recognition is used to understand what is in the crate and to retrieve the location of each item.

In order to implement object recognition, we looked at the different approaches that there are available:

- Object Analytics in ROS
- OpenCV library in python
- Convolutional neural network based machine learning algorithms

Mask RCNN

Mask RCNN is a deep neural network which has the goal to solve instance segmentation problem in machine learning. Instance Segmentation is the process of identifying each object inside an image at pixel level. This is the hardest task in computer vision compared to other sub types inside Image Segmentation.

Mask RCNN , which stands for regional convolutional neural network, is working in two stages. In the first stage it scans the input image and it generates numerous different proposals on where an object might be located. Basically, it generates a ton of boxes in different places inside the image and passes them on. In the second stage, the proposals are analyzed, classified. Bounding boxes are generated for each recognizes instance.

The decision is to use Mask RCNN for our project, a decision that was heavily influenced by our use case. The team needed technology that can recognize, with high accuracy, between different items inside a crate. This process should be fast and reliable. The feature which adds a mask to every object recognized made our decision clear.

The other major framework that was considered by our team was YOLO, which is another fast and accurate real-time object detection system. The framework outputs only bounding boxes around the recognized objects. Since the crate is filled with different items in different orders the bounding boxes can easily overlap and make coordination of the robot arm harder. Thus, having a mask on each object makes it easier to calculate what the dimensions of the object are and its location.

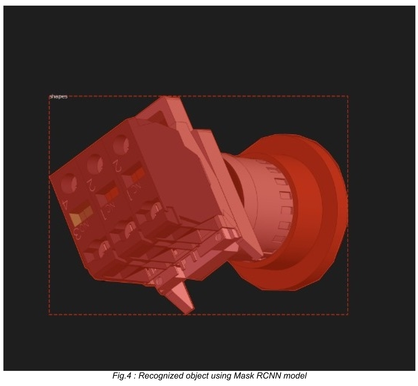

Fig.4 : Recognized object using Mask RCNN model

Results

At the end of the project, the overall results do not comply with the first expectations but the team is pleased with the progress and the entire project experience. Unfortunately, the challenges we faced during the SOFA project were not matching our skills yet. Every technology faced was new for most of us.

On the other hand, we were able to answer the client's question "Can a robotic system be built with an affordable depth camera and ROS?"

After our research and experiments, a robotic system using the RealSense camera and Mask RCNN as a object detection algorithm inside a ROS environment is attainable, but quite complex.

TOP RATED PROGRAMME universities KEUZEGIDS 2021

Informatics (semseter 7)

Software Factory project (2020-2021)