Carl Guo (carlguo2)

Emmanuel Gallegos (eg11)

CS 543 - Computer Vision

University of Illinois Urbana-Champaign

10 October 2022

Collision Detection and Avoidance With Multiple Cameras Views

Project description and goals:

We want to implement our project with a combination of Computer Vision and Robotics techniques. Using a mounted camera, we will provide live image feeds of the environment to mobile robots. These robots will use these camera feeds to perform various computer vision tasks such as detection, classification, segmentation, and localization. With these operations, our goal is to make the robot successfully predict possible future events (such as collision).

We will first attempt to implement the primary computer vision tasks successfully on our toy example, so our robot will be able to dynamically gather visual information of the environment while moving around. Upon completion, we will implement collision avoidance and test in an open environment with other immobile and mobile objects. If our robot is able to successfully detect and avoid collisions, we will discuss and come up with other important event prediction tasks.

As a minimum goal, we want to be able to perform live collision detection and avoidance on the RaspberryPi platform. As a maximum goal, we'd like to be able to perform these same tasks on the TurtleBot platform in such a way to interface with the Parasol Planning Library, and explore other tasks we can perform on the robots. This can include keypoint detection for localization.

Member roles:

Because our project involves robotics, we will need to work on both hardware and software levels. Carl will lead the high level software implementation, such as implementing deep learning frameworks relating to detection, classification, localization, and segmentation tasks. We will look into using fast deep learning techniques, such as YOLO (Redmon et al.), as well as classical feature detection depending on performance requirements. Carl will also be responsible for implementing Computer Vision tasks such as aligning and stitching multiple image feeds together.

Emmanuel will focus on the low level hardware tasks, such as connecting components on the robot to work properly, interfacing with any API's or SDK's needed to run our software on the robots, and generally making sure the robot is able to perform these high level tasks correctly and efficiently. There may also be some tasks that overlap in both areas, which we will work together on. These roles may not be strictly enforced and may be flexible as the project goes on.

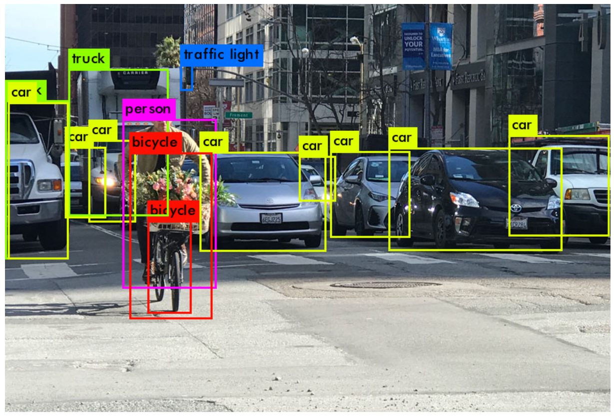

Figure 1: Segmentation + Classification example w/ Deep Learning

Resources:

The data we plan to use will be live image feeds captured by either live cameras mounted to robots or an environmental camera sensor. We will start by implementing the system in Python using Raspberry Pi's mounted on a small robot car. This will allow us to leverage the familiar Python environment for image processing with known image processing libraries such as OpenCV. For the detection portion of our project, we may employ CV models from TensorFlow or Pytorch and retrain them for our specific use cases, though we also wish to attempt to do some forms of classical detection and segmentation techniques that do not rely on CNN's time permitting.

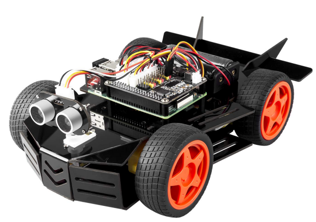

Figure 2: SunFounder Robot Car with Raspberry Pi

If this implementation is successful, we will attempt to move onto implementing the same system using TurtleBots in the Parasol Lab at UIUC. Ideally, this code will be able to interface with the Parasol Planning Library, which is written in C++, and help contribute to the research being done in the lab. We have access to TurtleBots, Raspberry Pi's, and sufficient computing resources in the lab to do all the work we have to in order to complete this project.

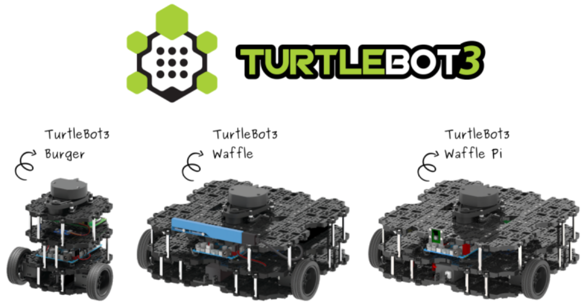

Figure 3: TurtleBot Robots

Reservations:

Some potential stumbling blocks will be working with new types of models we have not worked with. Neither of us are familiar with live event detection. Another issue will be attempting to interface with the Parasol Planning Library, which is a C++ based library. Emmanuel is aware of other students that use the robots to perform basic motion planning tasks, but so far nothing in the lab is using actual cameras for any kind of detection or localization techniques.

Relationship to your background:

Both team members have familiarity with the Raspberry Pi platform, and some limited experience with using pretrained CV models. We have knowledge of Python as well as major data science libraries from our computer vision class, AI/ML coursework, and other IoT classes (such as OpenCV, numpy, pandas, matplotlib, pillow, sklearn, PyTorch, and TensorFlow).

This work is vaguely related to Emmanuel's work in the Parasol Lab, in that the goal is to eventually write code that can be used to interface with the Parasol Planning Library. This does not relate to his thesis research, however, which is about parallel programming for multi-agent task and motion planning algorithms. As such, Emmanuel is familiar with the Parasol Planning Library, but not with any kind of hardware implementations used in the lab. This project is his opportunity to work with the image processing and sensing aspects of robotics, which only tangentially relates to his work.

Carl is not affiliated with the Parasol Lab at all, and this work does not relate to his thesis research either. Carl has some experience working with robots and autonomous vehicles in IoT classes. He also has taken courses on AI, as well as Prof. Lazebnik's course on Deep Learning.

Neither Carl nor Emmanuel have ever done any computer vision work in C++ (aside from using libraries like TensorFlow that use C++ under the hood).