Bryant Pong
Micah Corah
4/20/15
CSCI-4962

Computer Vision Proposal

Description

We would like to write a package for detecting objects of interest for the NASA Sample Return Robot Challenge. This competition requires teams to design, build, and program a fully-autonomous robot to search for sample objects in a large outdoor field. Samples vary in difficulty from tightly-specified and easily recognizable objects such as a pink tennis ball to harder and more loosely specified objects such as a painted rock or a metal object with an identifying mark. Our software will be used to identify regions of interest given a static image. For instance, consider the case in which the robot is given an image containing a water bottle and a ball on the ground. We would like our software to be able to isolate two regions of interest: 1) the water bottle; and 2) the ball. The output from our software will be bounding boxes around the regions of interest. This addresses the first step to identifying and retrieving a sample. Afterward the robot will be able to approach the region of interest to inspect the potential sample more closely using algorithms specific to each kind of sample. If a sample is identified the robot will then attempt to collect the sample and return it to the starting platform.

We want to tackle this project because there currently is no software package that can perform this task for us and because we need to tailor this package for our specific needs.

Approach

We will be using the sliding windows concept to look for regions of interest (ROIs). The sliding window dimensions will initially be small (i.e. a 10x10 pixel region) and will grow gradually with each image (i.e. 15x15, then 30x30 and so forth). The maximum sliding window dimension will be bounded by the largest object in the database. Note that the examples in the database will focus on realistic cases at a reasonable distance.

Our code will train a neural network via the Scikit-Learn package. At each iteration of the sliding window algorithm the neural network will be used to determine if there is a ROI lurking within the window as well as the location of the ROIs.

As stated above the return value of our code will be the number and locations of the ROIs.

Data Sources

The Sample Return team has a collection of sample objects similar to what will be used in the challenge. We will create database of images by taking pictures of the samples in various locations and lighting conditions outdoors at RPI. An attempt will be made to create realistic examples. Pictures will be taken ranging from approximately one to ten meters from the samples. This will ensure that the samples are still visible while excluding cases where the sample is near enough to be recognized by other methods. The database will be annotated with bounding boxes around the samples to identify the regions of interest. We will also be adding images of the competition field and images found online to the database to serve as hard negatives.

Software

We plan to use the following technologies on our project:

Language: Python
Supporting Libraries: OpenCV, Numpy, Scikit (Neural Network Module), Matplotlib, pickle
Additional Software: ROS (Robot Operating System) - Provides a bridge from OpenCV to the robot code.