This report presents the progress on the proposed project for 15400 on the topic of object detection using machine learning models trained with image data collected from robots. There have not been major changes to either the focus of the topic or the general plan of implementation. Minor changes to the approach of labeling the image data will be discussed later in this report.

According to the proposed time-line of the project, the milestone goal for this semester is to get familiar with the platform for the computer vision part of the robot I will be working with, and get familiar with different object recognition approaches using and other libraries.

The first part of the milestone is fully met. After talking to Professor Manuela Veloso and some of her graduate students, I decided to work on Pepper, a human-shaped robot that will be participating in RoboCup@Home by accomplishing certain tasks involving computer vision, speech recognition and human interaction, etc. All dependencies for programming for the robot have been set up on an Ubuntu machine and I have been able to connect to Pepper and collect image data with the built-in camera using ROS, Naoqi APIs and the previously developed pipelines for commanding the robot.

The second part of the milestone is met as well. One of the existing and popular object detection implementations that presents fairly high accuracy is YOLO, which is a real-time object detection system. A major difference in this system than others is that it applies a single neural network to the full image, while dividing the image into regions and predicts bounding boxes and probabilities for each region; the bounding boxes are weighted by the predicted probabilities. YOLO has also been tested on Pepper to detect objects in the lab and in Gates and Hillman building. It produces acceptable results of recognizing chairs, mugs and humans, etc.

A discovery over the course of applying YOLO to Pepper is that we realized
that to faster label the image data collected from the robot for generating training dataset ourselves, we could use existing object detection systems to assign possible labels to objects in the frame, present the ones with the lowest probabilities (the one that Pepper is the most unsure about) on the touch screen which is part of Pepper’s built-in user interaction subsystem. By implementing certain human-robot interaction functionality, we could let Pepper ’ask’ people to tell it the correct label of objects in the frame, thus achieving a more efficient way to label the training dataset to be later used for our own context-constrained training. This could also be regarded as part of robot’s learning process.

Currently I am working with required dependencies on an Ubuntu Virtual Machine running on my MacBook Pro. However, with limited memory, the system runs significantly slowly. The resource still needed for me to accomplish this project more efficiently would be a PC that supports Ubuntu. One alternative would be spending more time in the lab to use the computers there. Meanwhile, I’m planning to looking for spare computers from other sources to be able to work remotely on the simulation of the robot.