Object Detection In Differing Training Environments

ECE 383/ ME 555 Intro to Robotics - Fall 2023

Professor Oca

Marcus Ortiz, Jasmine King, Jess Tiu, Breanna Sandoval

December 14th, 2023

Abstract

This study investigates the accuracy of object detection models trained on photo sets from different sources, including a simulated Gazebo environment, real world robotics lab, and online sources. For our project, we use a YOLOv5 object detection algorithm to recognize bottles of different colors and a cup to feed into a broader robot bartender system. Three distinct datasets were used to independently train the YOLO objection detection algorithm; then, each of these trained algorithms were validated and tested on the same validation and testing sets that included only lab pictures. Based on the testing results from each of the models, the lab based model resulted in the highest mAP50-95 scores across all object classes against the training set.

Introduction

Computer vision has become an important component of many of today's robotic systems. For example, Boston Dynamics' Spot uses cameras to conduct inspections of industrial sites (Automation that Works with You. Boston Dynamics.). Object detection, a sub-task of computer vision, is used to check whether an instance of a specific class (e.g. humans, cars) is in an image, and where that instance is in the image.

One of our bartending robot system's goals is to determine the correct bottle to pick up in order to create a specific drink. An Intel RealSense d435i camera was used to capture images of the robot and its task space (table with bottles and a cup on it). The images were then fed into a YOLOv5 object detection algorithm, which would output the x-y coordinates of the bottle that the UR5e arm would have to pick up.

The YOLOv5 algorithm, however, does not work out-of-the-box and needs to be trained on a photo set of bottles to execute its task accurately. Machine learning theory strongly suggests that training on different photo sets would yield different levels of performance when YOLOv5 is implemented after training. This study aims to compare the performance of versions of YOLOv5 trained on three different photo sets, each containing pictures of blue and green bottles in a specified "world": (1) Gazebo simulation, (2) cage in Wilkinson Garage Lab, and (3) the real world, using photos found online.

The results of this study can be used to guide future efforts to add object detection to robotic systems, specifically by showing whether it is useful to train algorithms on photo sets outside the environment that the robot is expected to encounter.

Methods

We collected photo data from three different sources: (1) Gazebo simulation, (2) the cage in Wilkinson Garage Lab, and (3) the real world, using photos found online and then RoboFlow was used to label and classify each image. For each source, we created a training set with 51 images; we also created a combination training set with 17 images from each source. Datasets are equal in size to have an even distribution and equally weighed. Eleven (11) lab images were collected to create a validation set. A separate set of data from the lab environment with 11 images was collected for testing each of the algorithms, with each algorithm trained on one of the 3 photo sets to measure the performance of the algorithms for this specific application. Regardless of what training set an algorithm was trained on, it was validated on the “lab” validation set and tested on the “lab” test set. Within Roboflow, augmentation on the images is done to generate variability in brightness and noise resulting in three times the number of images giving us 153 images in each training set, 33 images in the validation set, and 33 images in the testing set. Furthermore, pre-processing steps are taken to create consistency in sizing and orientation of images across all sets. We trained each of these algorithms on CoLab GPU for time efficiency.

Results and Discussion

Training Set	Class	Precision	Recall	mAP50	mAP50-95
Simulated	All	0.816	0.948	0.966	0.8
	Blue Bottle	0.965	0.844	0.964	0.797
	Cup	0.913	1	0.994	0.815
	Green Bottle	0.57	1	0.939	0.788
Lab	All	0.998	1	0.995	0.95
	Blue Bottle	0.998	1	0.995	0.954
	Cup	0.997	1	0.995	0.927
	Green Bottle	0.998	1	0.995	0.969
Online	All	0.721	0.475	0.439	0.246
	Blue Bottle	0.544	0.606	0.67	0.405
	Cup	1	0	0.0147	0.00628
	Green Bottle	0.618	0.818	0.632	0.328

Combination	All	0.993	0.99	0.994	0.937
	Blue Bottle	0.995	1	0.995	0.942
	Cup	0.986	0.969	0.993	0.933
	Green Bottle	0.999	1	0.995	0.937

Table 1: Performance Scores of each Algorithm on Training Set

Table 1 displays the key performance metrics of each model on the test dataset: a focus is placed on the mAP50-95 metric which provides an overall performance picture with a stricter scoring system than mAP50. The lab trained model topped this metric, scoring the highest with all object classes averaged and individually with both bottle classes. However, the combination trained model had the second highest mAP50-95 score across all object classes and the highest score for the cup class. Regardless, both scored quite close and are high enough to be used in our system, indicating that although the combination of different environments may not produce the best for our static testing it would be a wise option with any increase of variability such as different bottle shapes. The online based model had the lowest and least consistent mAP50-95 scores on each of the object classes, with the worst performance detecting the cup. This model would be unusable in the bartending system and would take significantly more data to perform well without much benefit, unless the environment and objects are very dynamic. Lastly, the simulated model performed well enough that to be useful in development, but not for a finalized solution.

Figures 1-4 display the F1-confidence curves for each training set across all object classes. F1 scores balance the scores of precision and recall of each model and is calculated with the following equation: $F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$. The lab based model showed the most consistency across all object classes, while the online based model was the least consistent and lowest scores overall.

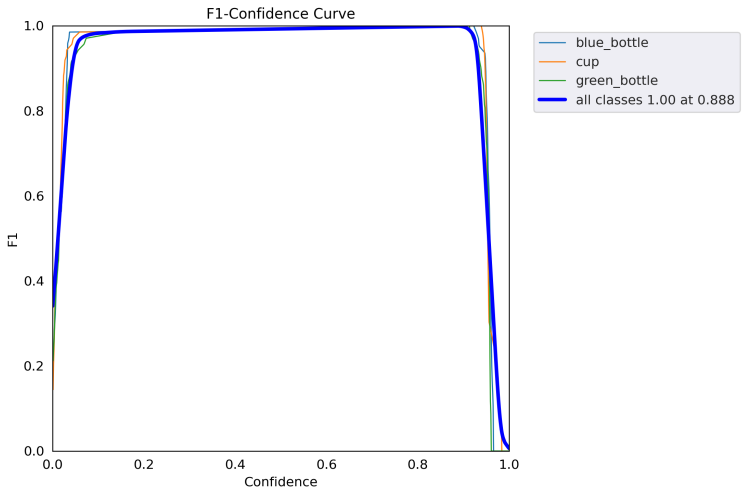

Figure 1: F1 Confidence curve from Lab Based Model against Testing Set

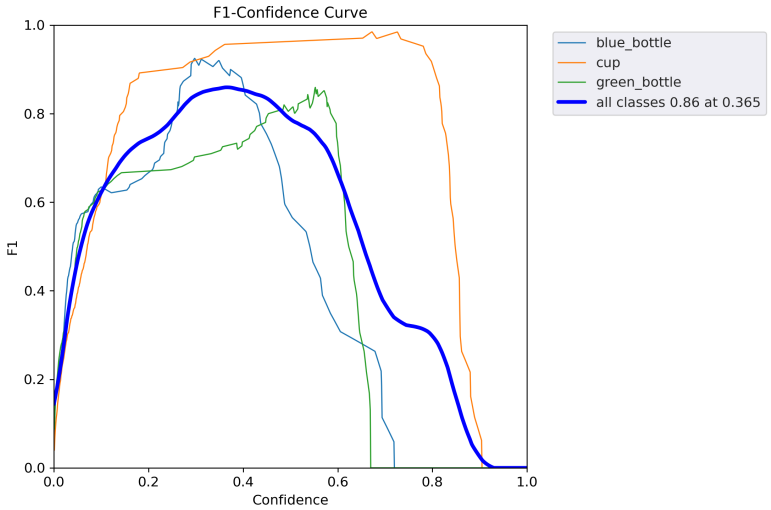

Figure 2: F1 Confidence curve from Simulated Environment Based Model against Testing Set

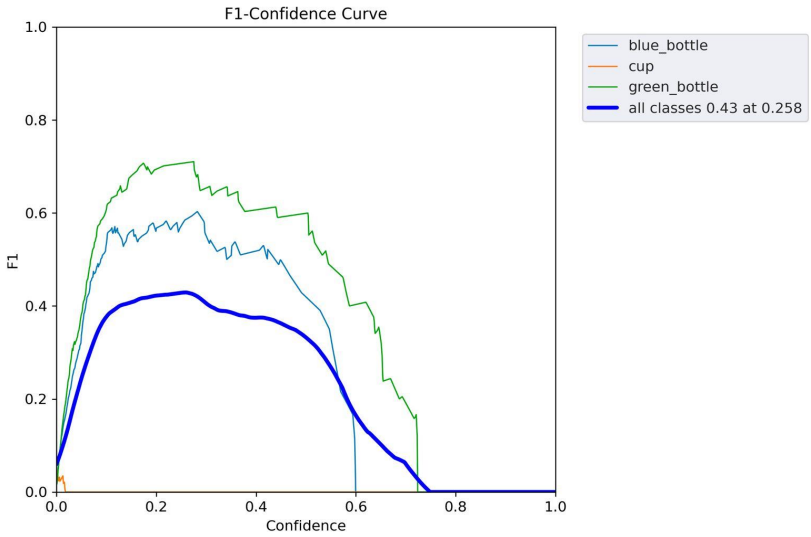

Figure 3: F1 Confidence curve from Online Based Model against Testing Set

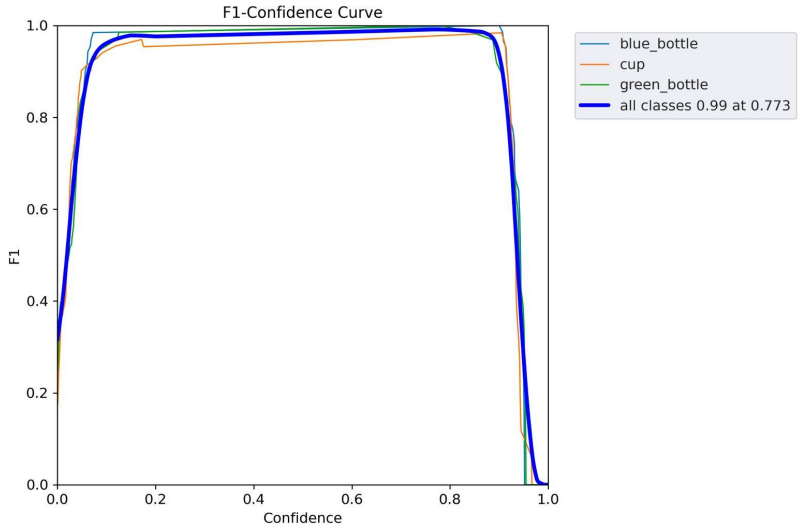

Figure 4: F1 Confidence curve from Combination Based Model against Testing Set

Limitations and Future Work

With only 153 images per photo set, we cannot expect to train YOLOv5 on each photo set from scratch and expect accurate results. Instead, we first trained YOLOv5 on the COCO dataset (the subset containing photos of bottles and cups) and fine-tuned it with our three photo sets. This method's results may misrepresent how well a photo set can influence the performance of YOLOv5, especially with the "cup" class, since this is a standard class in YOLO COCO. A more robust method with generalizable results would

have needed around 1500 photos per class (The practical guide for Object Detection with YOLOv5 algorithm | by Lihi Gur Arie, PhD | Towards Data Science).

Another limitation of our method had to do with the photo set containing images found online. This dataset was meant to depict bottles and cups in the “real world”. The bottles and cups shown in these images, however, did not reflect the angles that the bottles and the cup would take in the robot’s task space. Because the Gazebo simulation and Garage Lab photo sets did reflect the bottles and cup’s expected angles, it was hard to judge whether the YOLOv5 algorithm trained on the “real world” photo set performed worse because of the bottle and cup angles, or because of some other factor.

It would be interesting to expand this study by training and testing YOLOv5 on photo sets of bottles of different sizes and colors, to reflect the variation of bottles that a bartending robot would encounter in a more commercial environment.

Conclusion

After running each of these algorithms on the testing set with 33 images, the most consistent mAP50-95 scores across all classes of data came from the lab photos trained algorithm. Furthermore, the lab based model showed the most consistency balancing precision and recall as shown in the F1 curve in figure 1 with the lowest score of 0.888.

Our results show that it is best to train on images from the environment that the robot system is expected to interact with. Images from other environments are not likely to be helpful.

Appendix: Code

Bottle Yolo Notebook
https://colab.research.google.com/drive/1kjC3my32g0MU6VHbZS_vTTCts-XpPrf4?usp=sharing

Original YOLOv5 Provided Notebook
https://colab.research.google.com/github/ultralytics/yolov5/blob/master/tutorial.ipynb