An Exploration in Object Detection and Application in Autonomous Vehicle

Duc Dao | Mathematics & Computer Science | Advised by Rob Kelvey & Max Taylor

THE COLLEGE OF WOOSTER

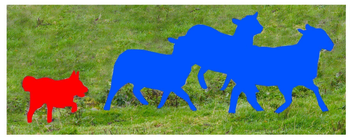
Introduction

This study aims to assess the effectiveness of object detection models in autonomous vehicles. This study evaluates how well the autonomous vehicle can identify and classify obstacles. Two commonly used object detection models, Faster Region-based Convolutional Neural Network (Faster R-CNN) and You Only Look Once version 5 (YOLOv5), are examined. The study begins with a discussion of various computer vision tasks, followed by an overview of the structure of Mask R-CNN (a model that adds instance segmentation capabilities to Faster R-CNN) and YOLOv5. Finally, a comparison of Mask R-CNN and YOLOv5's effectiveness in detecting objects in traffic scenes from the nuImage dataset is conducted.

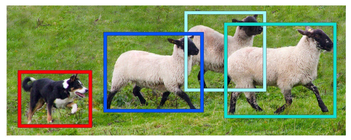
Different Computer Vision Tasks

[1]

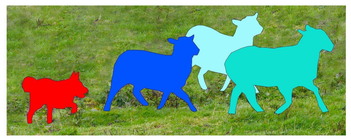
a.Image Classification



b.Semantic Segmentation



c.Object Detection

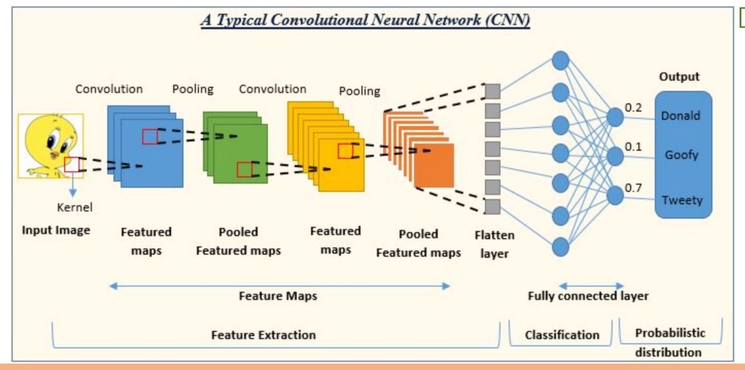


d.Instance Segmentation

Convolutional Neural Network (CNN)

- Both Mask R-CNN and YOLOv5 are based on a CNN architecture

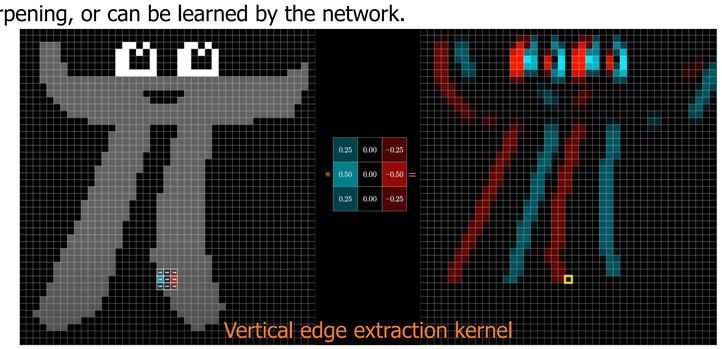
[2]



Convolutional & Pooling Layer

- The convolutional layer: slide kernel(s) through an image to extract features.
- The kernel can be predefined to extract out vertical edge, horizontal edge, burying, sharpening, or can be learned by the network.

[3]



- The output of the convolutional layer is a feature map.
- Extracting multiple features results in stacked feature maps along the depth dimension.
- Pooling layer is used to reduce the number of pixels inside the feature map.
- Each pixel in extracted feature maps is an entry in the input vector (flatten operation) for FCs.

Fully Connected Layer (Forward Pass – Make a Guess)

- Neuron's value in a forward pass (making a guess):

$$a_j = \max(0, a_j\ netin) = \max\left(0, \sum_{i=1}^{n} x_i w_{i,j} + 1(w_{0,j})\right)$$

- Convert the output neuron's value to probability:

$$\hat{y}_i = \sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_i}}$$

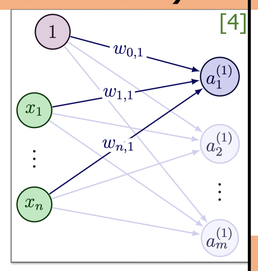
where k is number of classes, z_i is neuron's value of class i.

- Network's error in predicting class i:

$$CrossEntropy\ (CE)\ of\ \hat{y}_i = -\sum_{i=1}^{k} (t_i) \times \log(\hat{y}_i)$$

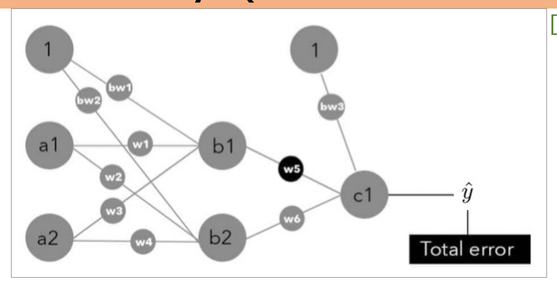
- Network's Total Error (or Total Cross-Entropy) is the sum of all classes' errors.
- Goal: Reduce total error to minimum.

[4]



Fully Connected Layer (Backward Pass – Learning)

[4]



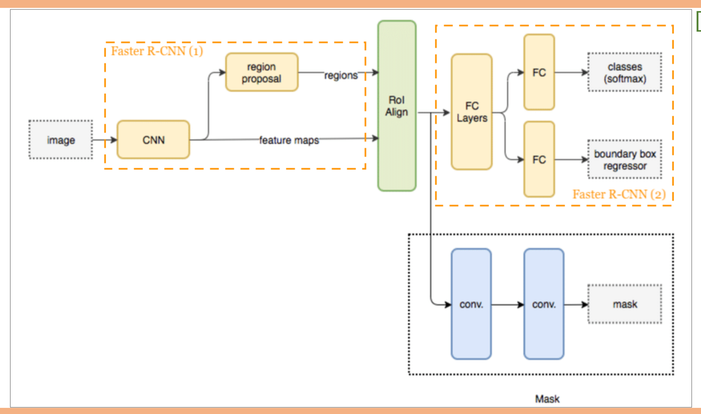
- The gradient value reflects how a change in weight affects the total error.
  - Eg: gradient of weight w_1 (perform chained partial derivative) is:

$$\frac{\Delta \text{Tot. CE}}{\Delta w_1} = \frac{\Delta \text{CE of } \hat{y}}{\Delta \hat{y}} \times \frac{\Delta \hat{y}}{\Delta c_1} \times \frac{\Delta c_1}{\Delta c_1\ \text{netin}} \times \frac{\Delta c_1\ \text{netin}}{\Delta b_1} \times \frac{\Delta b_1}{\Delta b_1\ \text{netin}} \times \frac{\Delta b_1\ \text{netin}}{\Delta w_1}$$

- Then the new weight w_j value that reduces the total error is: $current\ w_j - \frac{\Delta\ Tot.\ CE}{\Delta w_j}$

Mask R-CNN

[5]



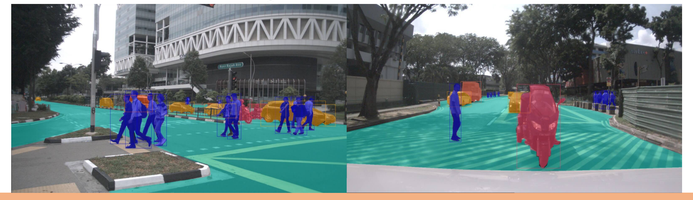
YOLOv5

- YOLOv5 architecture is heavily based on the YOLOv3 architecture.

[6]

nuImages Dataset

- A dataset that contain 93,000 images on traffic scenes



Metrics

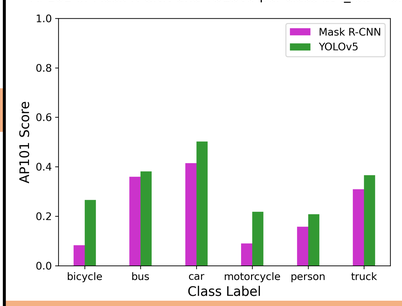
- Object's location correctness: $\boldsymbol{IoU} = \frac{Area(BB_{predict} \cap BB_{truth})}{Area(BB_{predict} \cup BB_{truth})}$ (higher $IoU \Rightarrow$ better overlap)
- **Confidence score**: the level of confidence that the model is in a prediction
- At each IoU and confidence threshold, a **confusion matrix** is created based on label correctness (matching between predicted and ground-truth labels)
- With the confusion matrix, precision and recall are computed
  - **Precision**: model's correctness in classifying samples as belonging to a class
  - **Recall**: model's ability to identify all object instances of a class
- With precision and recall, **N-point interpolation Average Precision (AP_N)**, an approximation of model's performance in predicting a class at an IoU threshold, is:

$$AP_N = \frac{1}{N} \sum_{R \in R_N} P_{interp}(R) \quad with \quad P_{interp}(R) = \max_{R':R' \geq R} P(R')$$

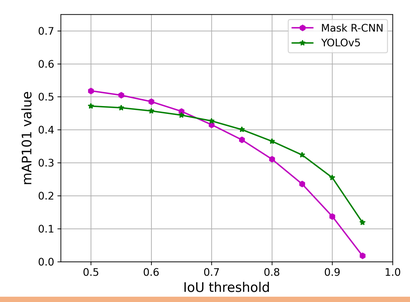
- Then **mean N-point interpolation Average Precision ($\mathbf{mAP_N}$)** denotes model's performance in predicting objects of any class across all confidence scores at a given IoU threshold is $mAP = \frac{1}{K} \sum_{i=1}^{i=K} AP_i$ where K is the number of supported class

Result

AP101 of Mask R-CNN and YOLOv5 per class at t_iou = 0.85



mAP101 of Mask R-CNN and YOLOv5 at different IoU thresholds



Conclusion

- The result shows that YOLOv5 outperforms Mask R-CNN as an object detection model for autonomous vehicles as it can accurately detect objects at their correct locations
- However, the results also reveal that neither model is fully prepared for deployment in real-world scenarios, as their performance rating falls below 60%
- The performance of these models is evaluated using their pretrained weights, which may not fully reflect their potential in traffic detection applications. Thus, additional domain training is necessary to gain a better understanding of their performance

Acknowledgment

I would like to thank my advisors, Rob Kelvey and Max Taylor, for their guidance, and the College of Wooster's Mathematical and Computational Science Department for their support during this project.

References

- [1] Tedrake, R. (n.d.). Robotic manipulation. Ch. 9 - Object Detection and Segmentation. Retrieved April 6, 2023, from https://manipulation.csail.mit.edu/segmentation.html
- [2] Shah, S. (2022, March 15). Convolutional Neural Network: An overview. Analytics Vidhya. Retrieved April 6, 2023, from https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-network-an-overview/
- [3] 3Blue1Brown. (2022, November 18). But what is a convolution? YouTube. Retrieved April 6, 2023, from https://www.youtube.com/watch?v=KuXjwB4LzSA&t=1184s&ab_channel=3Blue1Brown
- [4] Taylor, M. (2017). Neural Networks: A visual introduction for beginners. Blue Windmill Media.
- [5] Gonzalez, S., Arellano, C., & Tapia, J. E. (2019). Deepblueberry: Quantification of blueberries in the wild using instance segmentation. Ieee Access, 7, 105776-105788.
- [6] Datahacker.rs. (2020, March 17). #011 TF Yolo V3 object detection in tensorflow 2.0. Master Data Science. Retrieved April 6, 2023, from https://datahacker.rs/tensorflow2-0-yolov3/