YOLO Implementation on COCO Dataset

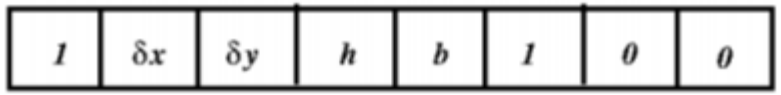
Tewbesta G. Alemayehu

Dataset Access

- Used the COCO API to access the images
- Resized to 128*128
- Checked that the correct bounding box was accessed by plotting image with its corresponding bounding box using line 106-109 of the code
- Returned the image’s tensor, label, bounding box tensor for each object within the image along with the number of objects within the object on the get_item function

YOLO Tensor

- The YOLO vector has 8 elements that contains the probability of the object being there, delta x and delta y (the center x and y coordinates of the image relative to the cell), the width and height of the image relative to the cell, and label is one hot encoded format.



- What is meant by cell is the grid of the image as the images is subdivided into an SxS grid. The smaller the grid, the more smaller objects that can be detected. In this case the coco images were subdivided in to a 6x6 cell using a yolo interval, width and height of each cell, of 20. This results in the 36 available cells per image.

Anchor Box

- 5 anchor boxes were used. Anchor box are meant to improve object detection by better detecting objects when there are overlapping bounding boxes. This is done by identifying the aspect ratio of the objects. Thus, if there are overlapping regions, the model can identify them based on the anchor box.
- The best anchor box is detected from the 1/5, 1/3, 1/1, 3/1, 5/1 aspect ratio and deciding which aspect ratio is best for that object. Hence, the YOLO Vector for that object is placed in the position index of that anchor box.
- This combined with the 8 element YOLO vector gives a 5*8 matrix. This means that each object is identified based on the resembling anchor box for each cell. Thus, for the total image, we will have a 36*5*8 yolo tensor.

IOU and Confidence Score

•It predicts boundary boxes and each box has one box confidence score.
•It detects one object only regardless of the number of boxes B.
•it predicts C conditional class probabilities (one per class for the likeliness of the object class).
•IOU is the intersection over union which I found by taking the maximum coordinates from the xmin and ymin of the ground truth and predicted bounding box and the minimum coordinates of the xmax, ymax coordinates from the width and height of the bounding added with the xmin and ymin of the bounding of the ground truth and predicted bounding box. Then by taking the difference, we can find the intersection. The union is calculated by summing the area of the ground truth and predicted bounding box and subtracting the intersection from it.

$$
\begin{aligned}
\text{box confidence score} &\equiv P_r(object) \cdot IoU \\
\text{conditional class probability} &\equiv P_r(class_i|object) \\
\text{class confidence score} &\equiv P_r(class_i) \cdot IoU \\
&= \text{box confidence score} \times \text{conditional class probability}
\end{aligned}
$$

where

$P_r(object)$ is the probability the box contains an object.
IoU is the IoU (intersection over union) between the predicted box and the ground truth.
$P_r(class_i|object)$ is the probability the object belongs to $class_i$ given an object is presence.
$P_r(class_i)$ is the probability the object belongs to $class_i$

•For this to happen the labels must match. Thus by taking the argmax of the predicted and ground truth label which are one hot encoded, I was able to find the labels. If the labels match, I compute the confidence score based on the formula given below: