Assignment 1 Report

Authors:

- Alberto Robazza

- Ivan Garcia Alcaraz

Task 1

task 1a)

The intersection over Union(IoU) is the measurement of the ratio between the predicted and the ground truth bounding boxes. The measurement is considered true positive if the result is overlaped by 50%. This evaluation is used in object detection task to measure the similarity between the predicted bounding boxes and the ground true bounding boxes.

To calculate the IoU we have to calcualte the area of the intersection between the two bounding boxex, calculate the union between the two bounding boxes, and calculate the ratio between them.

IoU = $\frac{area(Bp \cap Bgt)}{area(Bp \cup Bgt)}$

where Bp is the predicted bounding box and Bgt is the ground truth bounding box.

task 1b)

Precision measures how accurate is your predictions. i.e. the percentage of your predictions are correct.

Recall measures how good you find all the positives. For example, we can find 80% of the possible positive cases in our top K predictions.

The precision and recall can be expressed mathematically as follows:

Precision = $\frac{TP}{TP+FP}$

Recall = $\frac{TP}{TP+FN}$

where TP (true positive) is the number of correct positive predictions, FP (false positive) is the number of incorrect positive predictions, and FN (false negative) is the number of incorrect negative predictions.

A true positive is a prediction that is correctly classified as positive, meaning that it belongs to the positive class and the model correctly identifies it as such. A false positive, on the other hand, is a prediction that is incorrectly classified as positive, meaning that it does not actually belong to the positive class but the model identifies it as such.

task 1c)

Precision values for class 1 at different recall levels:

Recall = 0.0: Precision = 1.0
Recall = 0.1: Precision = 1.0
Recall = 0.2: Precision = 1.0
Recall = 0.3: Precision = 1.0
Recall = 0.4: Precision = 1.0
Recall = 0.5: Precision = 1.0
Recall = 0.6: Precision = 1.0
Recall = 0.7: Precision = 0.5
Recall = 0.8 :Precision = 0.5
Recall = 0.9 :Precision = 0.5
Recall = 0.10 :Precision= 0.2

Area under the curve for class 1 AP = $\frac{(7*1+3*0.5+0.2)}{11}$ The solution is $AP = 0.7909$

Precision values for class 2 at different recall levels:

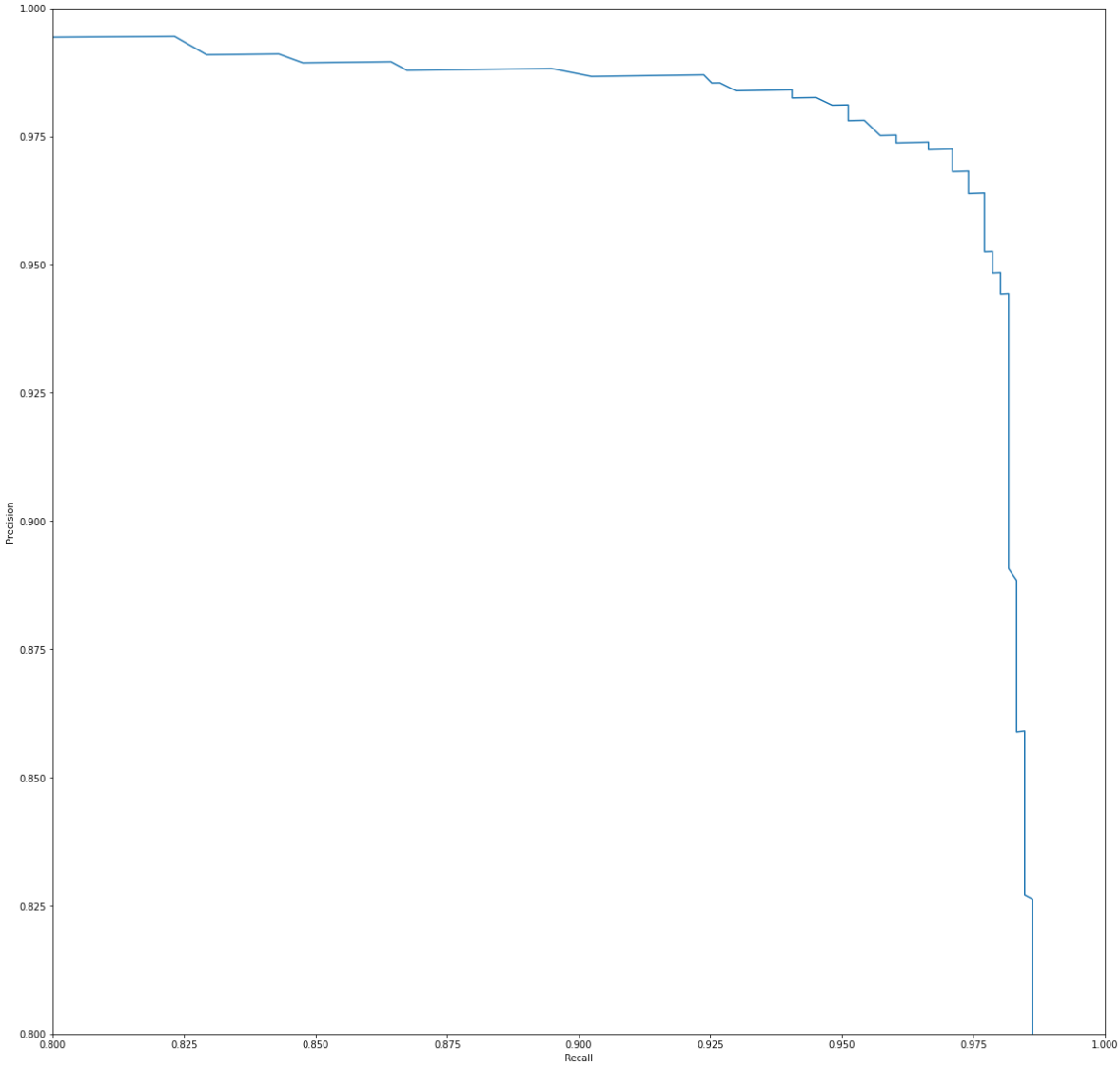
Recall = 0.0: Precision = 1.0
Recall = 0.1: Precision = 1.0
Recall = 0.2: Precision = 1.0
Recall = 0.3: Precision = 1.0
Recall = 0.4: Precision = 0.8
Recall = 0.5: Precision = 0.6

Recall = 0.6: Precision = 0.6
Recall = 0.7: Precision = 0.5
Recall = 0.8: Precision = 0.5
Recall = 0.9: Precision = 0.5
Recall = 1.0: Precision = 0.2

Area under the curve for class $AP = \frac{(4*1+0.8+2*0.6+3*0.5+0.2)}{11}$ The solution is $AP = 0.7$

Task 2

Task 2f)



Task 3

Task 3a)

The filtering operation to remove overlapping boxes in object detection with SSD (Single Shot Multibox Detector) is called Non-maximum Suppression (NMS). NMS is a post-processing technique commonly used in object detection to eliminate duplicate detections and improve the overall detection accuracy. It works by suppressing the boxes that have a high degree of overlap (measured using Intersection over Union) and keeping only the boxes with the highest confidence scores.

Task 3b)

The statement is false. Predictions from the shallower layers in SSD are responsible for detecting small objects. Shallow layers can be used to predict small objects because small objects don't need bigger receptive fields.

Task 3c)

This is used to be able to detect objects with different shapes and aspect ratios. For example, if we want to detect a car, we can use a square anchor box. However, if we want to detect a person, we can use a rectangular anchor box. This is because a person is not always in a square shape.

Task 3d)

One of the main differences between SSD and YOLOv1/v2 that SSD is more accurate than YOLOv1/v2. SSD is more accurate because it uses a multi-scale feature map, which allows it to detect objects at different scales. YOLOv1/v2, on the other hand, uses a single-scale feature map, which means that it can only detect objects at one scale.

Task 3e)

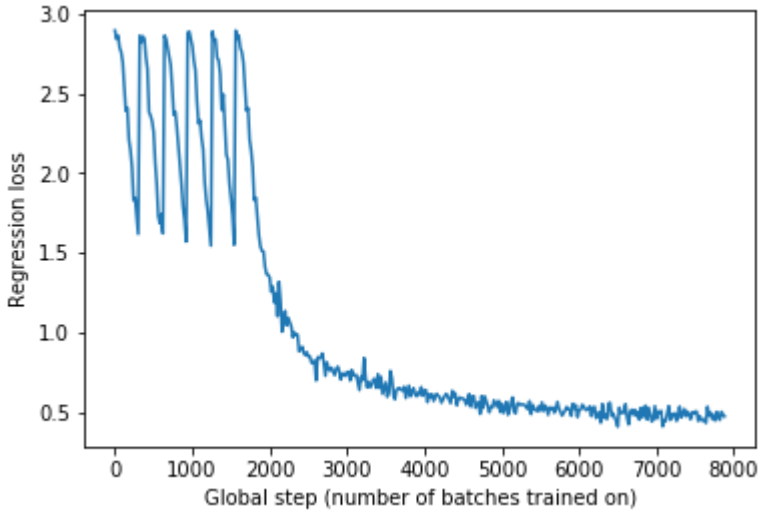
The total number of anchor boxes for this feature map, we can multiply the number of locations on the feature map (38x38) by the number of anchors per location (6). This gives us a total of 38 x 38 x 6 = 8664 anchor boxes for this feature map.

Task 3f)

The total number of anchor boxes for the entire network, we can sum up the number of anchor boxes for each feature map. This gives us (38 x 38 + 19 x 19 + 10 x 10 + 5 x 5 + 3 x 3 +1 x1) x6 = 34956 anchor boxes in total for the entire network.

Task 4

Task 4b)



mAP: 76.4% model trained on 20 epochs

Task 4c)

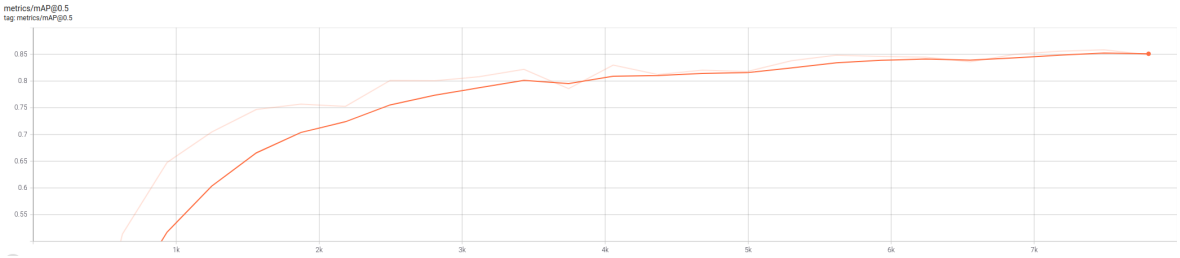
We have implemented some modifications to improve the performance of the model. We have introduced batch normalization, which helps to normalize the input data and reduce overfitting. Additionally, we have used the Adam optimizer, which is a variant of gradient descent that adapts the learning rate to the parameters. These modifications have resulted in a slight increase in the precision of the model.

Furthermore, we have experimented with different numbers of epochs and found that increasing the number of epochs by a small amount also improved the precision of the model. However, we have to be cautious in the choice of the number of epochs, as increasing the number too much can lead to overfitting, we increased the number of filters in some of the layers, which also contributed to the improvement in the model's precision.

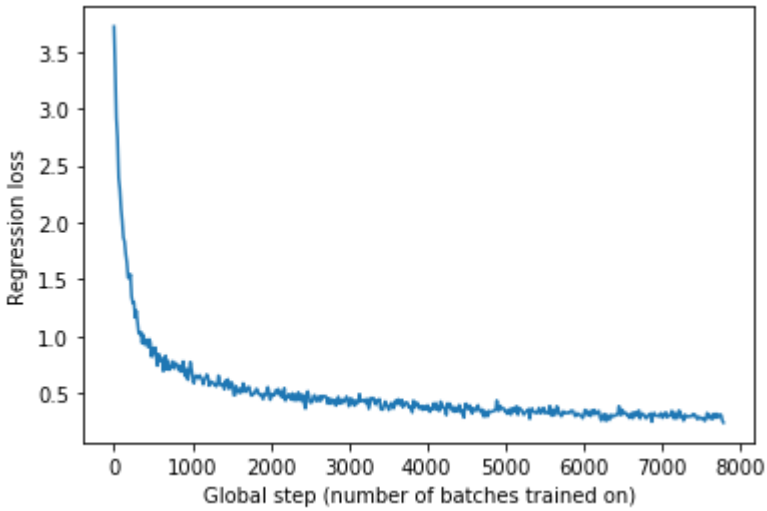
Finally, we have made a significant change by adjusting the minimum size of the first default box from [30, 30] to [15, 15] and [60, 60] to [30, 30]. This modification has resulted in a significant increase in the performance of the model. By decreasing the minimum size, we have increased the sensitivity of the model to smaller objects, which has led to a considerable improvement in the precision of the model for detecting small objects.

Overall, our modifications have led to a substantial improvement in the performance of the SSD model. The use of batch normalization and Adam optimizer, along with the right choice of minimum size of the default box, has resulted in a highly accurate model for object detection.

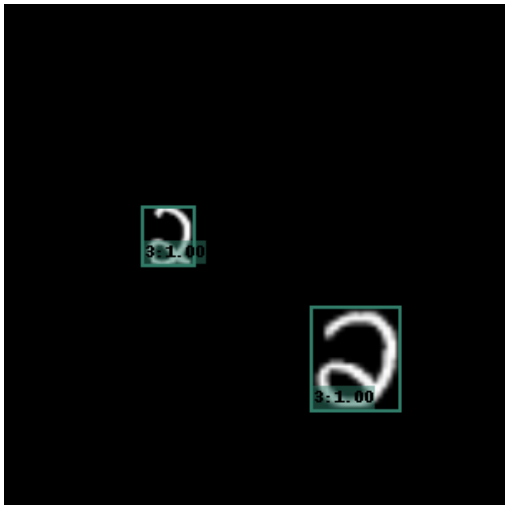
The precission:

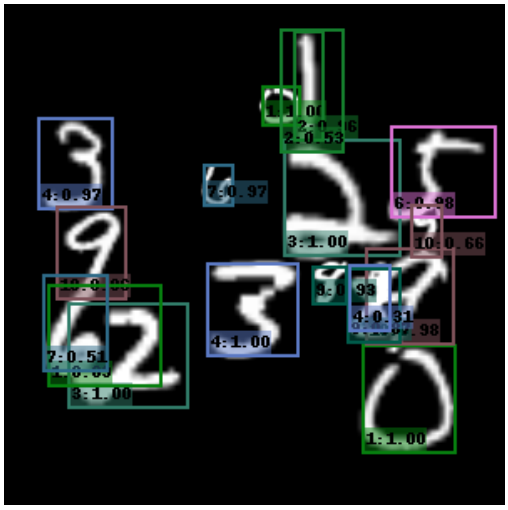


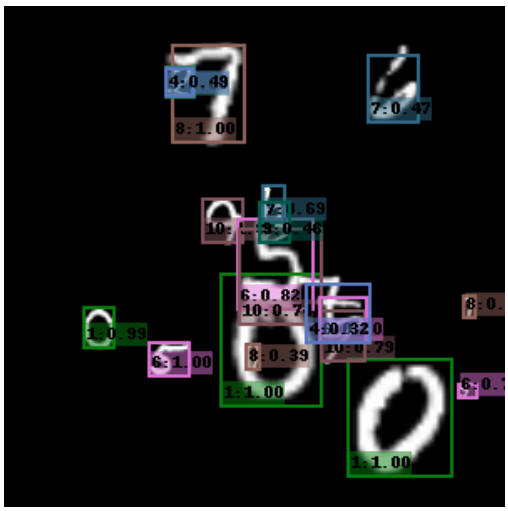
The loss:



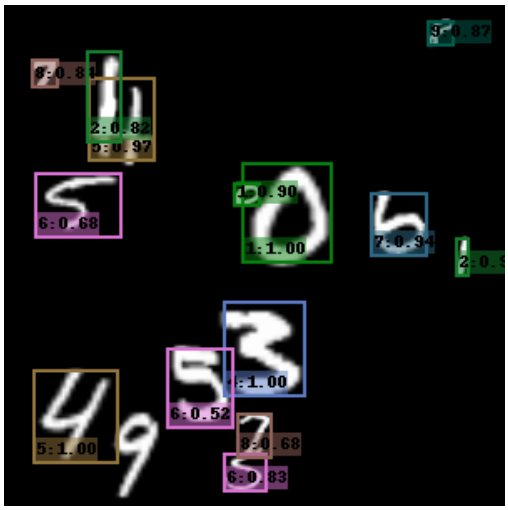
Task 4e)

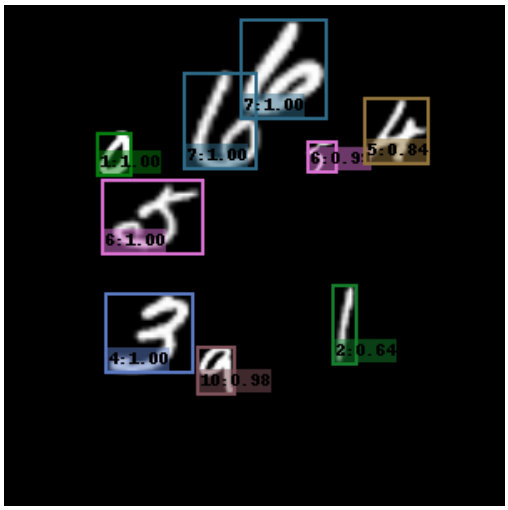

3: 1.00
3: 1.00


4: 0.97
7: 0.97
2: 0.53
3: 1.00
6: 0.98
10: 0.66
4: 1.00
7: 0.51
3: 1.00
1: 1.00

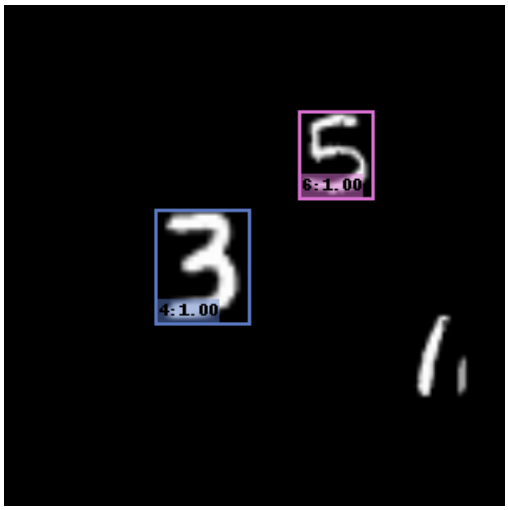

4: 0.48
8: 1.00
7: 0.47
6: 0.82
1: 0.99
6: 1.00
8: 0.39
1: 1.00
1: 1.00

10: 0.99
7: 0.96
2: 0.57
7: 0.99
4: 0.99
6: 0.97
2: 0.97
6: 1.00
3: 1.00
2: 0.85

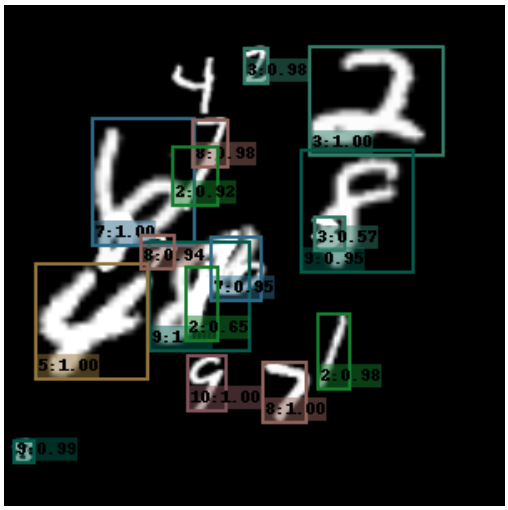

2: 0.82
6: 0.68
1: 1.00
6: 0.52
5: 1.00


7: 1.00
7: 1.00
5: 0.84
6: 1.00
4: 1.00
2: 0.64
10: 0.98

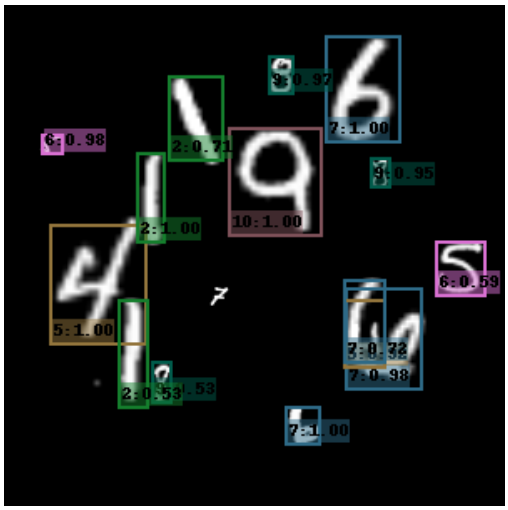
10:0.98
2:0.83
9:1.00
5:1.00
6:0.88
10:0.92
4:0.80


6:1.00
4:1.00

6:0.54
8:0.99
6:1.00
8:1.00
6:0.52
4:0.95


3:1.00
2:0.92
7:1.00
3:0.57
5:1.00
2:0.98
10:1.00
8:1.00
3:0.99

3:0.99


7:1.00
2:0.71
10:1.00
2:1.00
5:1.00
6:0.59
7:0.98
7:1.00

The model can't identify from the hole test some 4, 1 and 7. This number are quite obvious to identify but the model strugle to identify it.