TDT4265 - Computer Vision and Deep Learning

Vegard Iversen, Sebastian Skogen Raa

Mar 2022

1a

Q: Explain what the Intersection over Union is and how we can find it for two bounding boxes. Illustrate it with a drawing.
A:

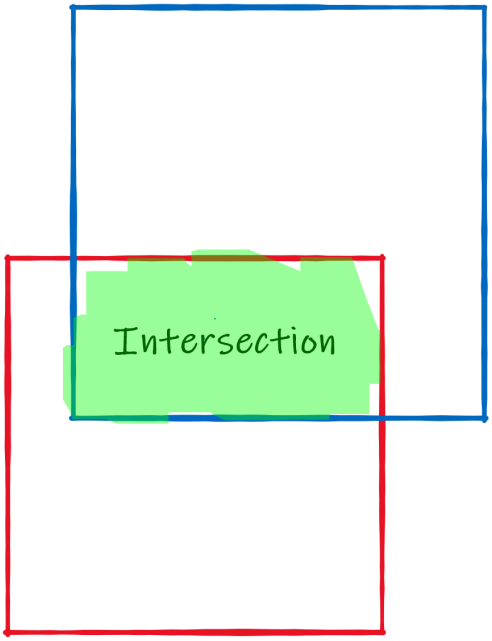

Figure 1: Intersection between two boxes, where the blue rectangle is the correct bounding box and the red is the predicted.

The IOU is given as the overlap area over the union area of the red and blue boxes.

1b

Q: Write down the equation of precision and recall, and shortly explain what a true positive and false positive is.
A: True positives are those examples which are classified as the correct class over a given threshold.

Oppositely, false positives are those examples which are classified under a given threshold as a different class than what they actually are.

$$\begin{aligned} \text{Precision} &= \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \\ &= \frac{\text{True Positives}}{\text{Total predicted positives}} \end{aligned} \tag{1}$$

Recall = TruePositives / (TruePositives + FalseNegatives)

$$\begin{aligned} \text{Recall} &= \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \\ &= \frac{\text{True Positives}}{\text{Total actual positives}} \end{aligned} \tag{2}$$

1c

Q: Given the following precision and recall curve for the two classes, what is the mean average precision?
A:

$$AP = \frac{1}{11} \sum_{recall_i} p_{interp}(r) \tag{3}$$

$$\begin{aligned} AP_1 &= \frac{1}{11} \sum_{recall_i} p_{interp}(r) \\ &= \frac{1}{11}(1 \cdot 5 + 0.5 \cdot 3 + 0.2 \cdot 3) \\ &= 0.65 \end{aligned} \tag{4}$$

$$\begin{aligned} AP_1 &= \frac{1}{11} \sum_{recall_i} p_{interp}(r) \\ &= \frac{1}{11}(1 \cdot 4 + 0.8 + 0.6 + 0.5 \cdot 2 + 0.2 \cdot 3) \\ &= 0.64 \end{aligned} \tag{5}$$

$$\begin{aligned} mAP &= \frac{0.65 + 0.64}{2} \\ &= 0.65 \end{aligned} \tag{6}$$

2f

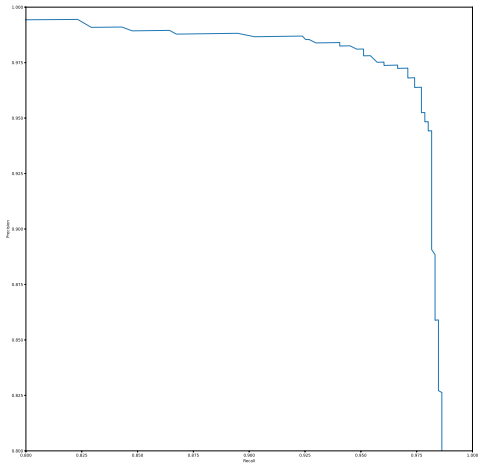

Figure 2: Precision recall curve.

3a

Q: The SSD architecture produces a fixed-size number of bounding boxes and a score for each bounding box. When performing inference with SSD, we need to filter out a set of overlapping boxes. What is this filtering operation called?
A: To filter out the useful overlapping boxes non-maximum suppression (nms) is used during inference. The operation removes duplicate predictions using a confidence threshold of 0.01 followed by discarding the corresponding previous bounding boxes with current predictions having IOU under 0.45. I effect this limits the number of predictions per image to 200.

3b

Q: The SSD architecture predicts bounding boxes at multiple scales to enable the network to detect objects of different sizes.
A: This is false as the feature maps of the high resolution is needed for detecting smaller objects and these are located at the front of the network.

3c

Q: SSD use k number of "anchors" 2 with different aspect ratios at each spatial location in a feature map to predict c class scores and 4 offsets relative to the original box shape. Why do they use different bounding box aspect ratios at the same spatial location?
A: This is done as the true bounding boxes does not have arbitrary shapes. Using a selection of different possible bounding boxes makes predictions more diverse, makes training easier and more stable.

3d

Q: What is the main difference between SSD and YOLOv1/v2 (The YOLO version they refer to in the SSD paper)?
A: Comparing the architectures, the SSD model have several added feature layers at the end of the base network. These predict the offsets to default boxes of different scales and aspect-rations and their corresponding confidences.

3e

Q: Given a SSD framework, where the first scale the network predicts at is at the last feature map with a resolution of 38 × 38 (H × W). For each anchor location, we place 6 different anchors with different aspect ratios. How many anchors boxes do we have in total for this feature map?
A: This can simply be computed as:

$$\begin{aligned} N_{\text{anchors boxes}} &= 38 \cdot 38 \cdot 6 \\ &= 8664 \end{aligned} \tag{7}$$

3f

Q: The network outlined in the previous subtask predicts at multiple resolutions, specifically 38 × 38, 19 × 19, 10 × 10, 5 × 5, 3 × 3 and 1 × 1. It uses 6 different aspect ratios at each location in every feature map as anchors. How many anchors boxes do we have in total for the entire network?
A: The total count can be found by adding them up:

$$\begin{aligned} N_{\text{total anchors boxes}} &= (38^2 + 19^2 + 10^2 + 5^2 + 3^2 + 1^2) \cdot 6 \\ &= 11640 \end{aligned} \tag{8}$$

4b

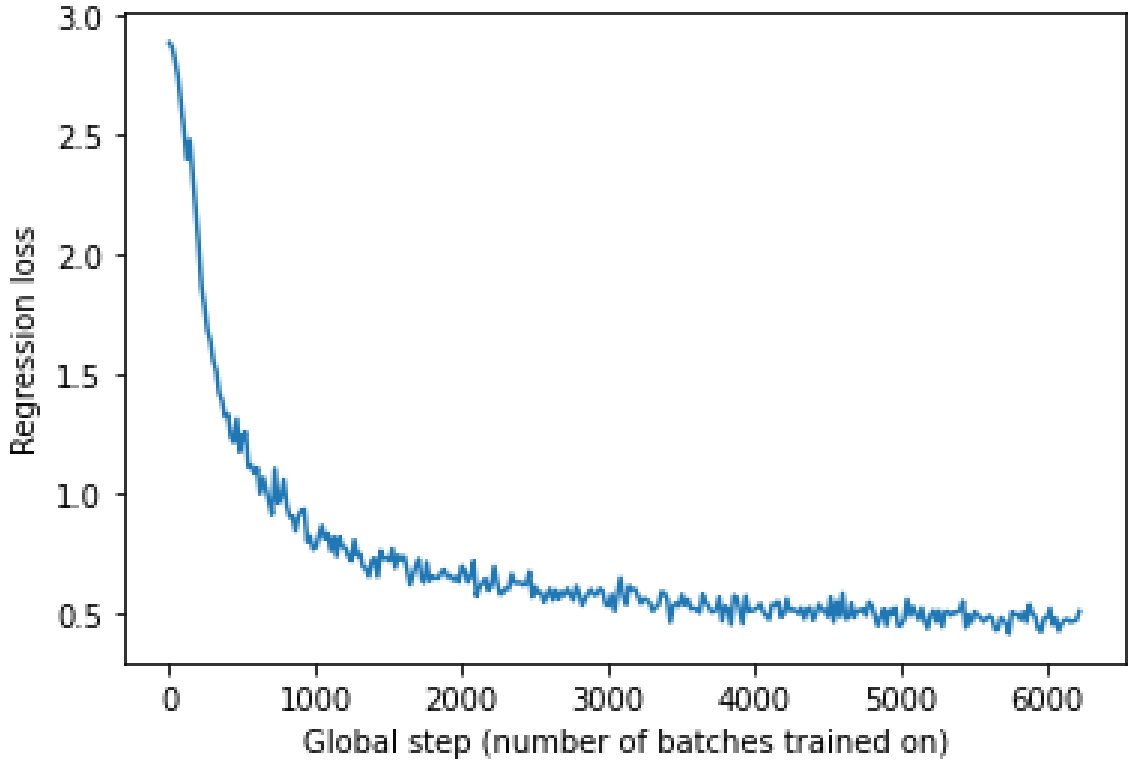

Figure 3: task4b 20 epoch

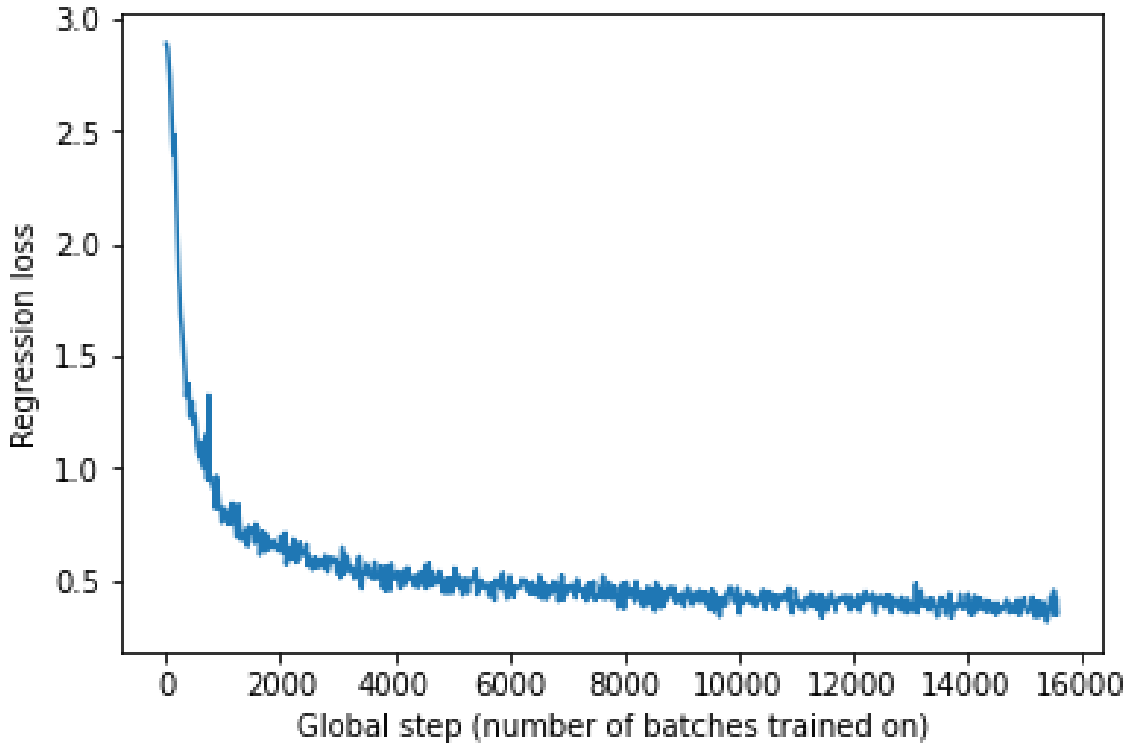

Figure 4: task4b 50 epoch

4c

Here we changed the activation function to Hardswish, added batchnorm, changed the first min sizes of anchors too [15,15] instead of [30,30], and changed the optimizer to adamW. Also tried a bigger network with skip layers, but couldnt get it to work.

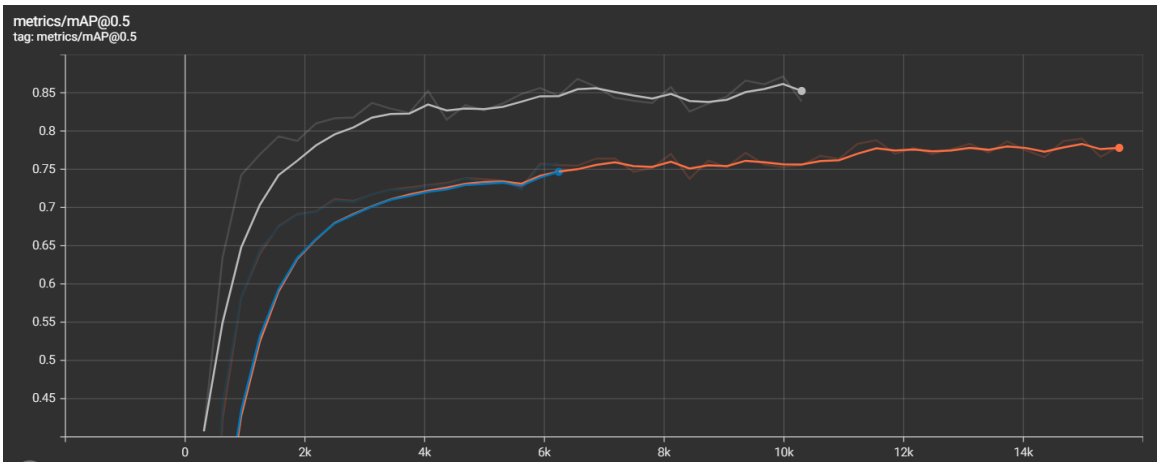

Figure 5: task4c, with the basicmodel with 6000 iterations in blue and with around 16000 in orange. The new model in grey.

4d

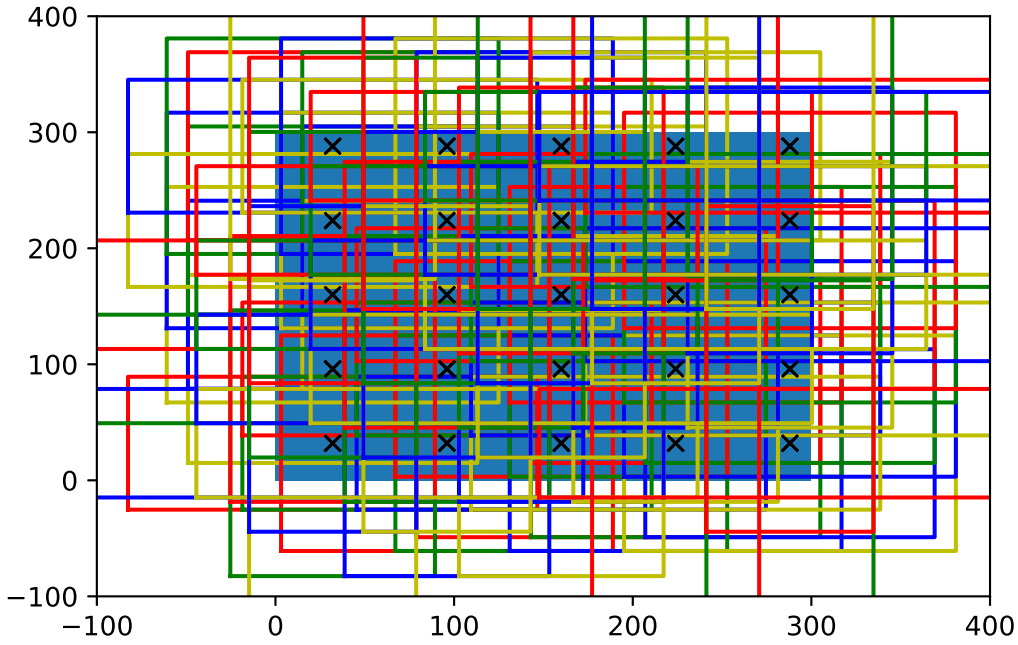

Figure 6: task4d boxes with centers

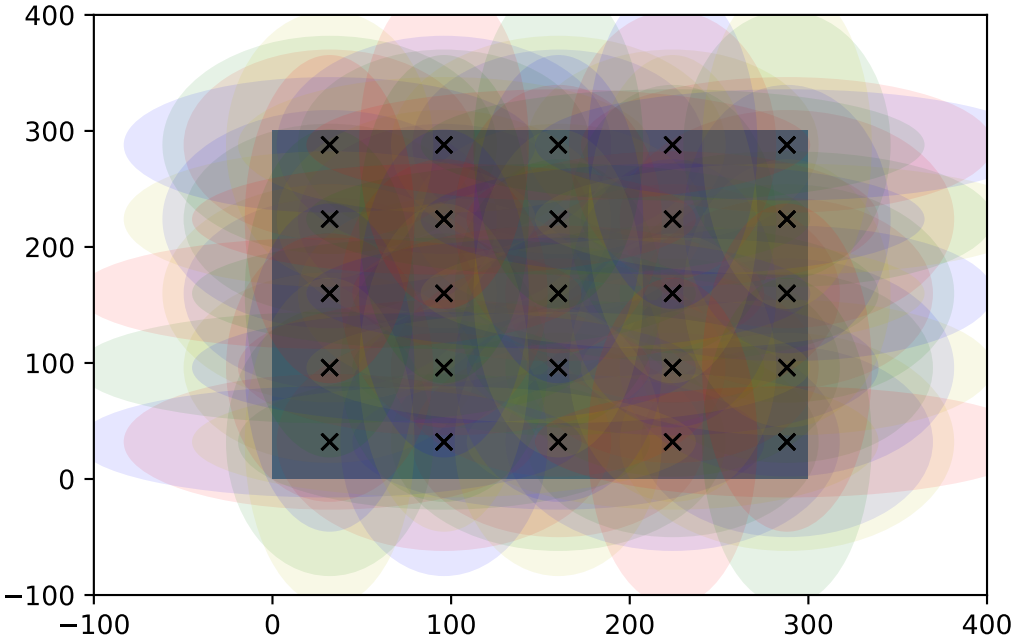

Figure 7: 4d circles

Comparing the calculations to the plots we observe that these seem to match:

Table 1: Center coordinates.

x	**y**
160.	32.
288.	32.
288.	32.
32.	96.
224.	32.
288.	96.
32.	160.
96.	160.
288.	160.
32.	224.
96.	224.
160.	224.
32.	288.
96.	288.
160.	288.
224.	288.
96.	32.
160.	32.
224.	32.
288.	32.
160.	96.
224.	96.
288.	96.
32.	160.
224.	160.

Table 2: Prob box sizes.

Width	**Height**
93.53075	280.59222
114.5513	229.1026
162.	162.
185.7579	185.7579
229.1026	114.5513
280.59222	93.53075

4f

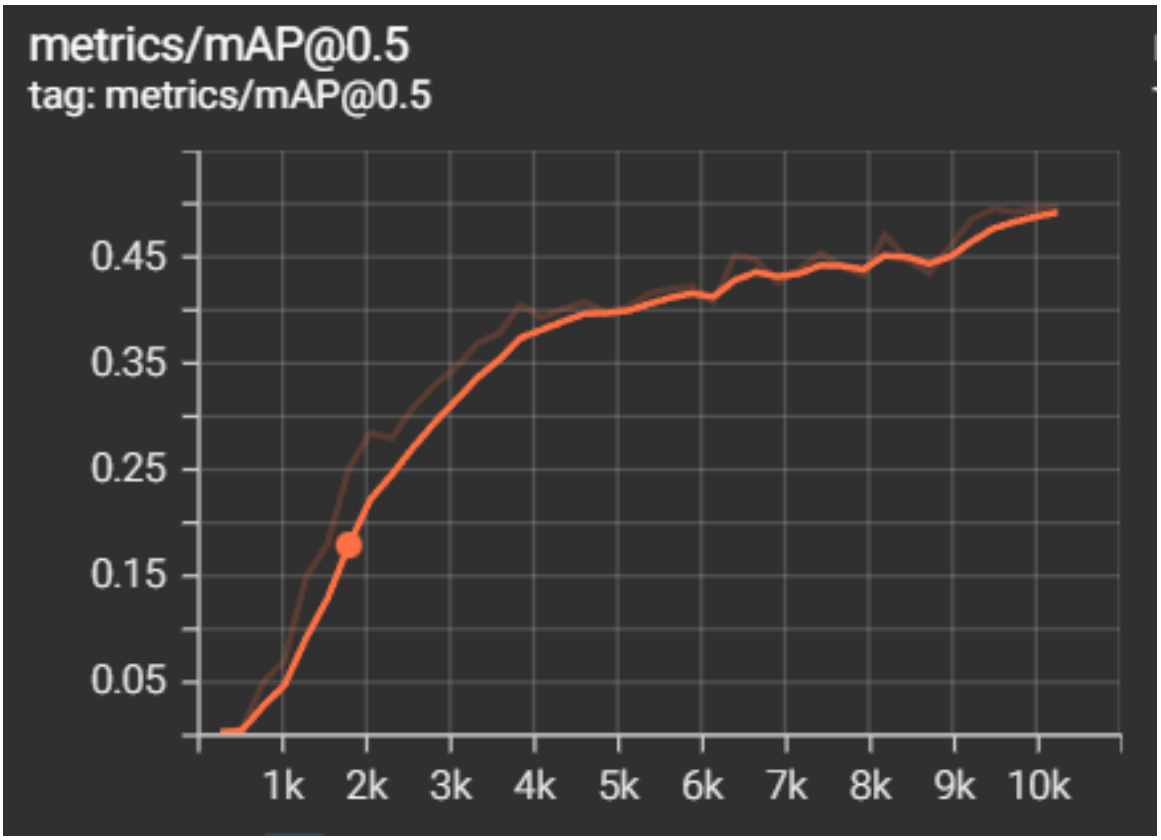

Figure 8: Task 4f

Figure 9: Task 4f model test

Figure 10: Task 4f model test

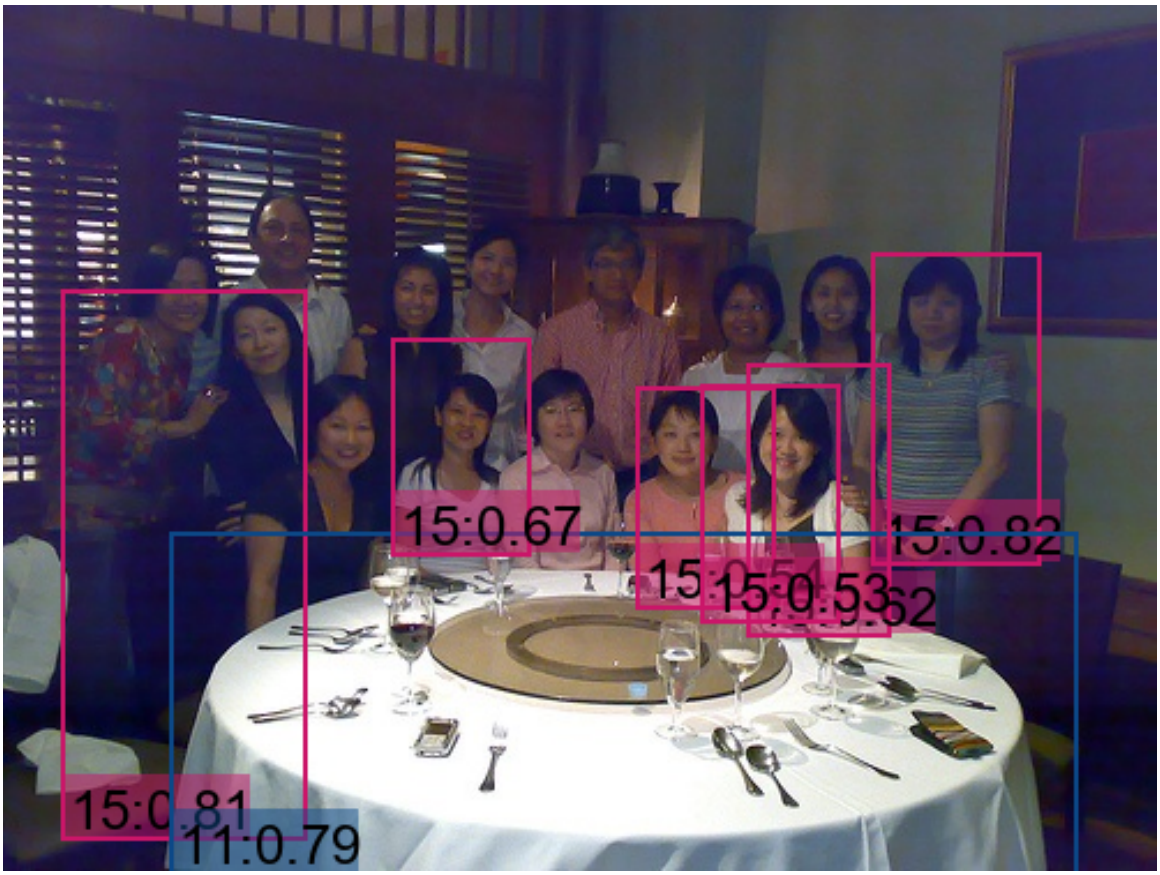

Figure 11: Task 4f model test

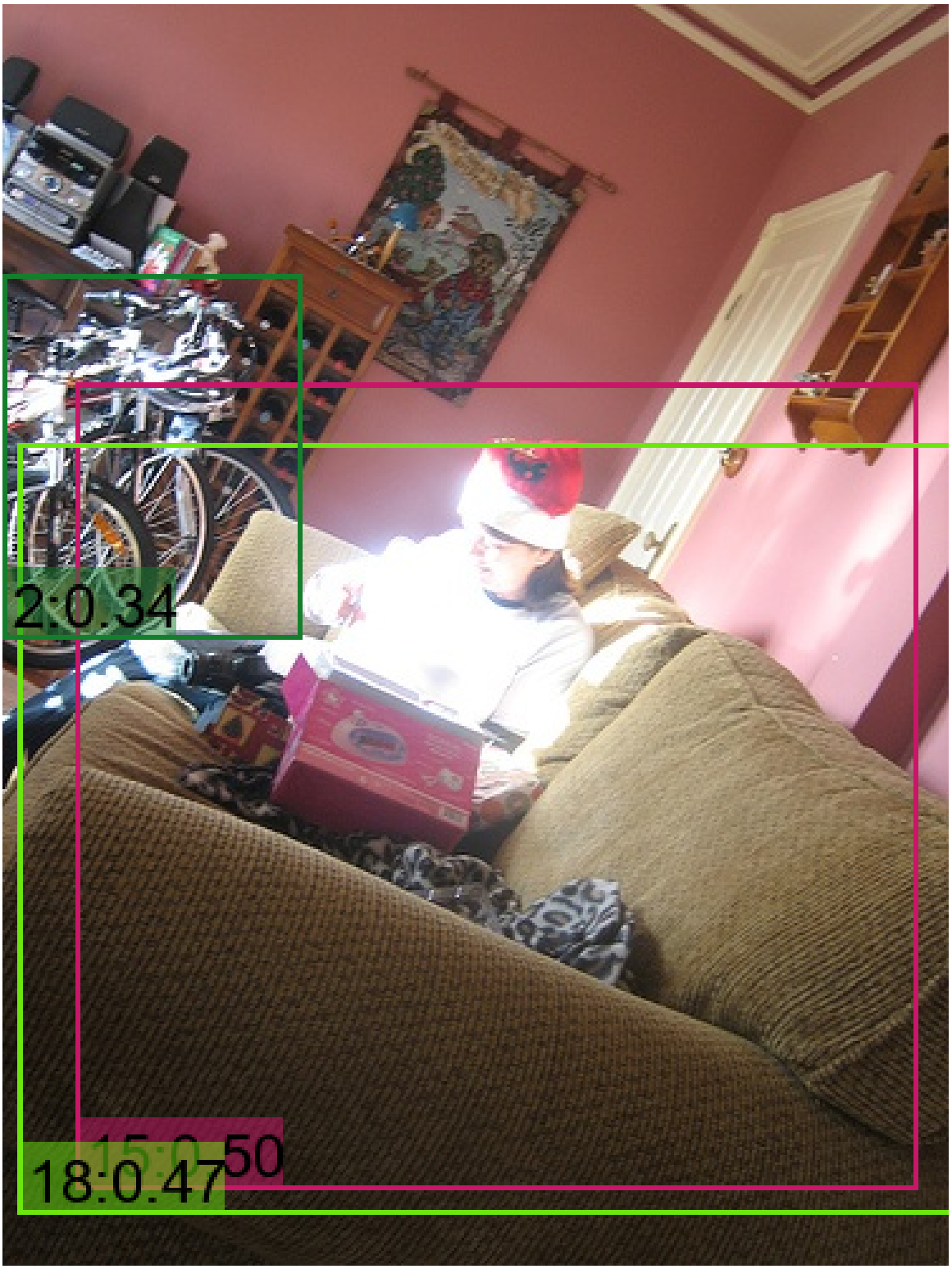

Figure 12: Task 4f model test

Figure 13: Task 4f model test