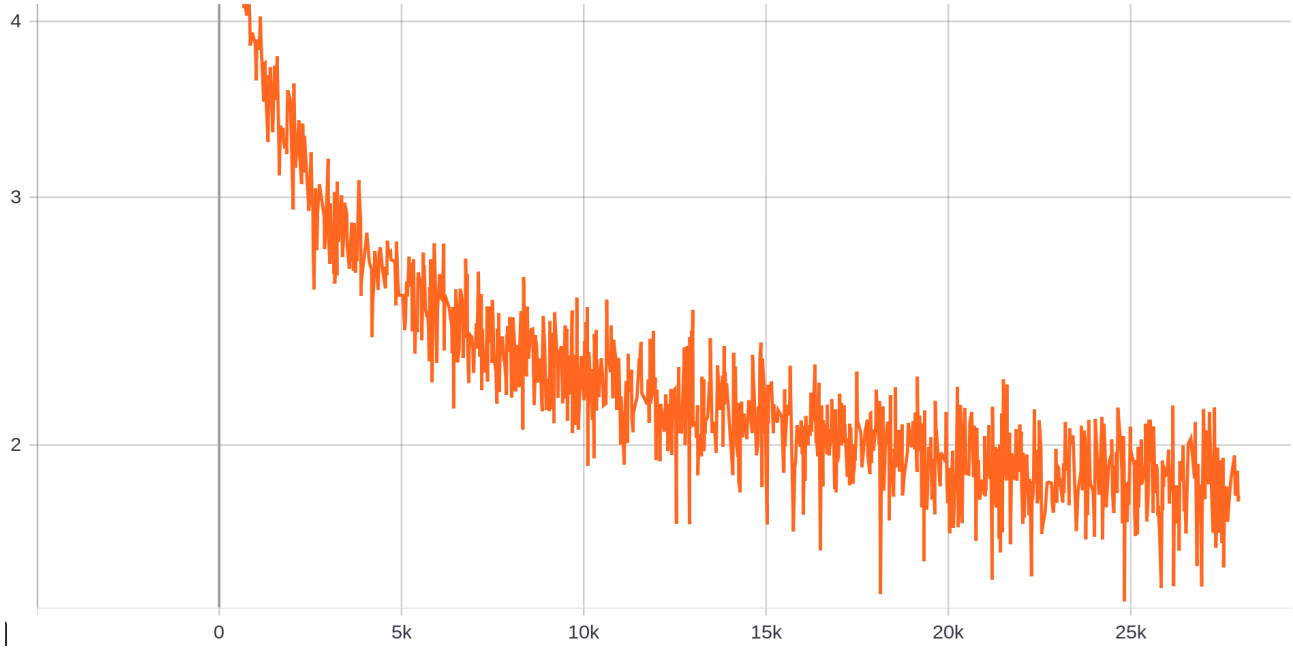


Abbildung 1: Total Loss

Final mAP after 6000 iterations is mAP = 0.7763

4c) and 4d)

I combined those tasks as I created a model which achieved a **final mAP of 90.6 %** after 15,000 gradient descent steps. The model also surpasses the 85% mAP after 10k gradient descent steps.

I implemented multiple changes to the given model:

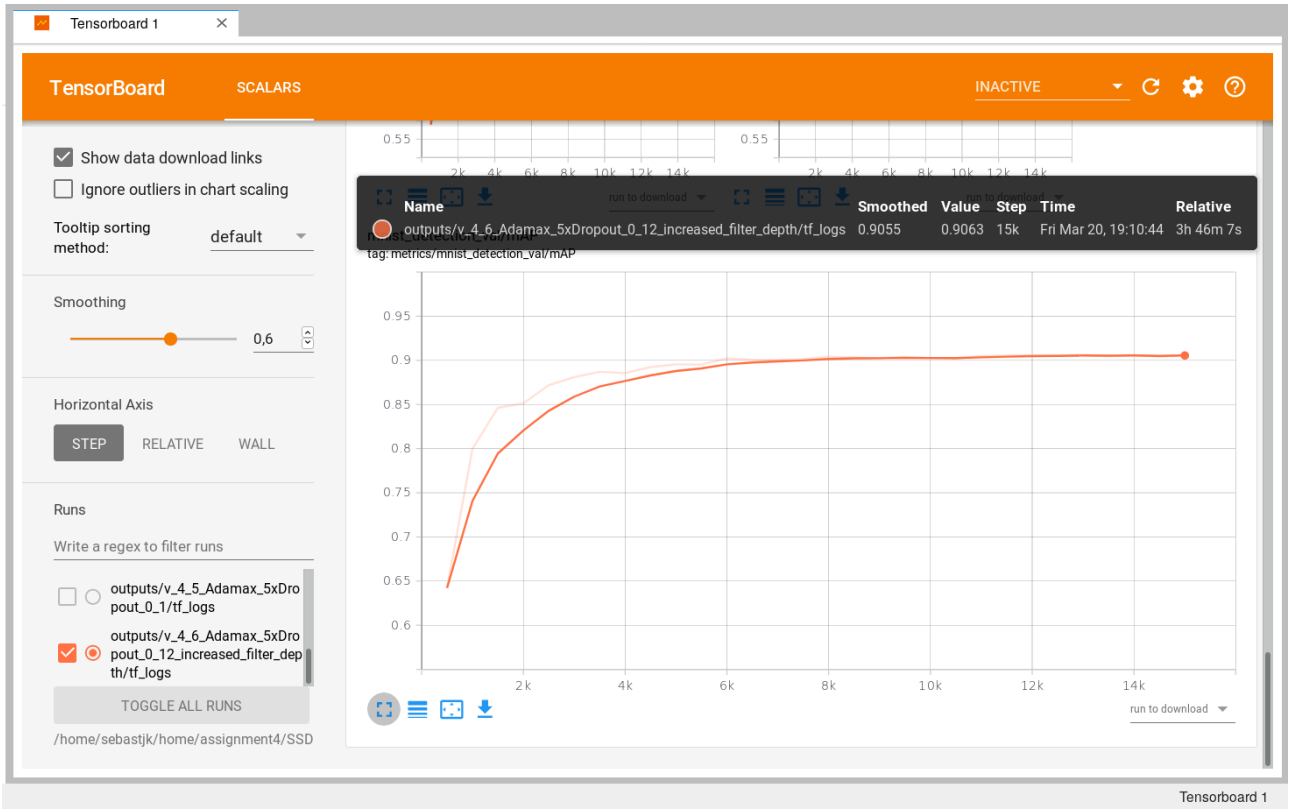
- four extra convolutional layers were added before the first output layer
- the number of filters in the first layers were increased
- after each output layer a dropout was added with a dropout probability of 12%
- the kernel size for the majority of the layers was increased to 5x5 from 3x3
- the Adamax optimizer was used
- the training data was augmented by applying image cropping using the provided function RandomSampleCropping()

Layer	Layer Type	Number of Hidden Units / Number of Filters / Dropout proabilitity	Activation Function	Kernel Size	Stride	Padding
0	Conv2D	64	ReLu	(5x5)	1	2
0	MaxPool2D	-	-			
1	Conv2D	128	ReLu	(5x5)	1	2
1	MaxPool2D	-	-			
2	Conv2D	128	ReLu	(5x5)	1	2
3	Conv2D	128	ReLu	(5x5)	1	2
4	Conv2D	128	ReLu	(5x5)	1	2
5	Conv2D	128	ReLu	(5x5)	1	2
6	Conv2D	128	ReLu	(5x5)	1	2
7	Conv2D (output)	128	ReLu	(5x5)	2	2
8	Dropout	P = 0.12	-			
8	Conv2D	64	ReLu	(5x5)	1	2
9	Conv2D (output)	256	ReLu	(3x3)	2	1
10	Dropout	P = 0.12				
10	Conv2D	128	ReLu	(5x5)	1	2
11	Conv2D (output)	128	ReLu	(5x5)	2	2
12	Dropout	P = 0.12				
12	Conv2D	128	ReLu	(5x5)	1	2
13	Conv2D (output)	128	ReLu	(5x5)	2	2
14	Dropout	P = 0.12				
14	Conv2D	128	ReLu	(5x5)	1	2
15	Conv2D (output)	64	ReLu	(5x5)	2	2
16	Dropout	P = 0.12				
16	Conv2D	128	ReLu	(5x5)	1	2
17	Conv2D	128	ReLu	(3x3)	1	0

Tabelle 1: Architecture of Model

Optimizer	Adamax
Augmentation of Data Set	• Randomly crop images

Tabelle 2: Parameters of Model 1



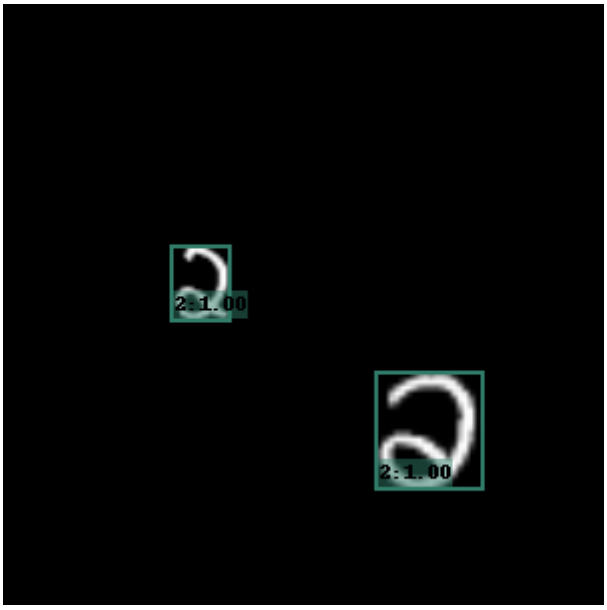
4e)

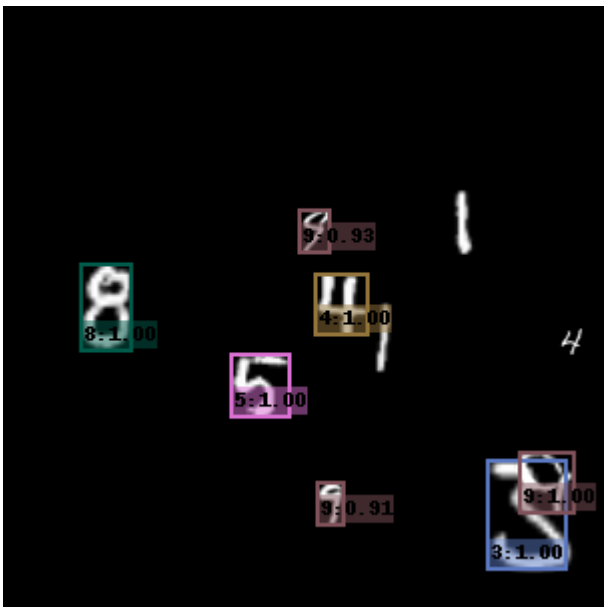
The model could detect the majority of the digits, but not all of them.

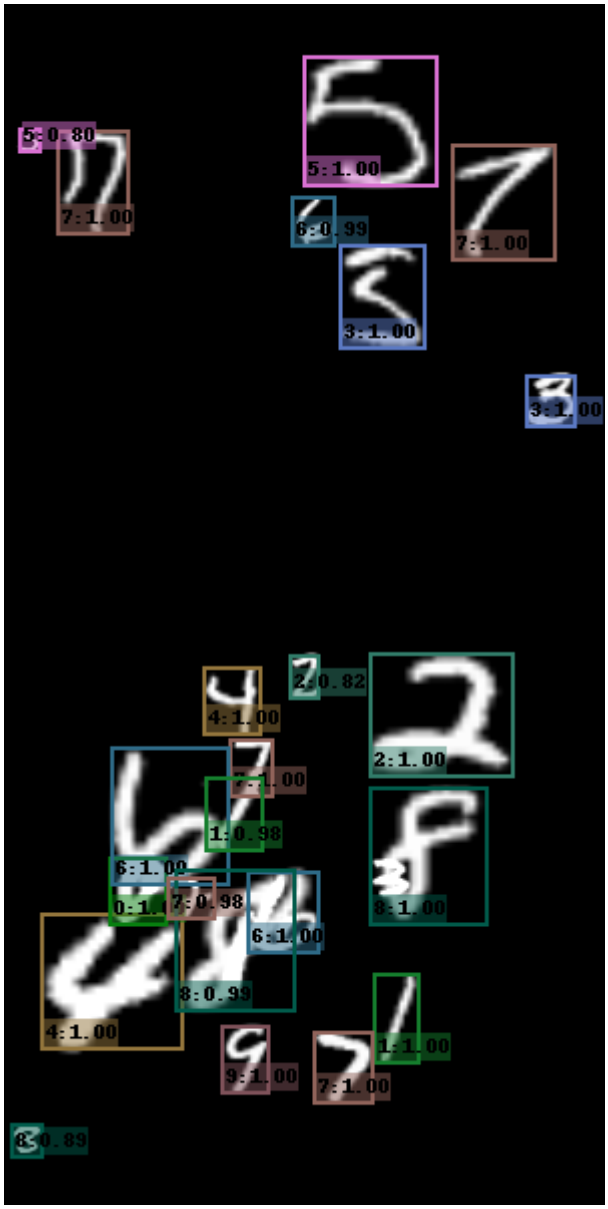
It struggled primarily with very small digits and when digits were very close or on top of each other.

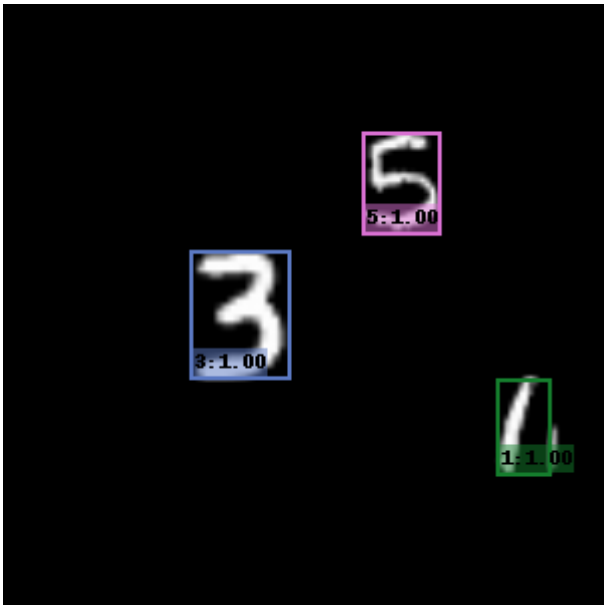
The model was not performing worse for certain digits.

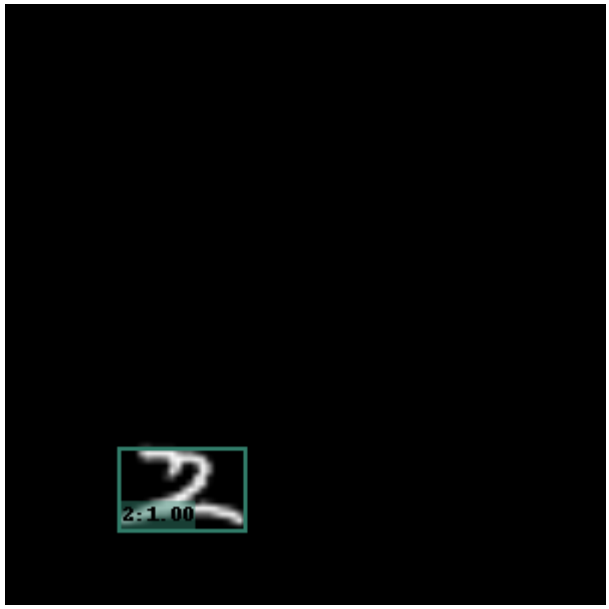
I appended the resulting pictures below.

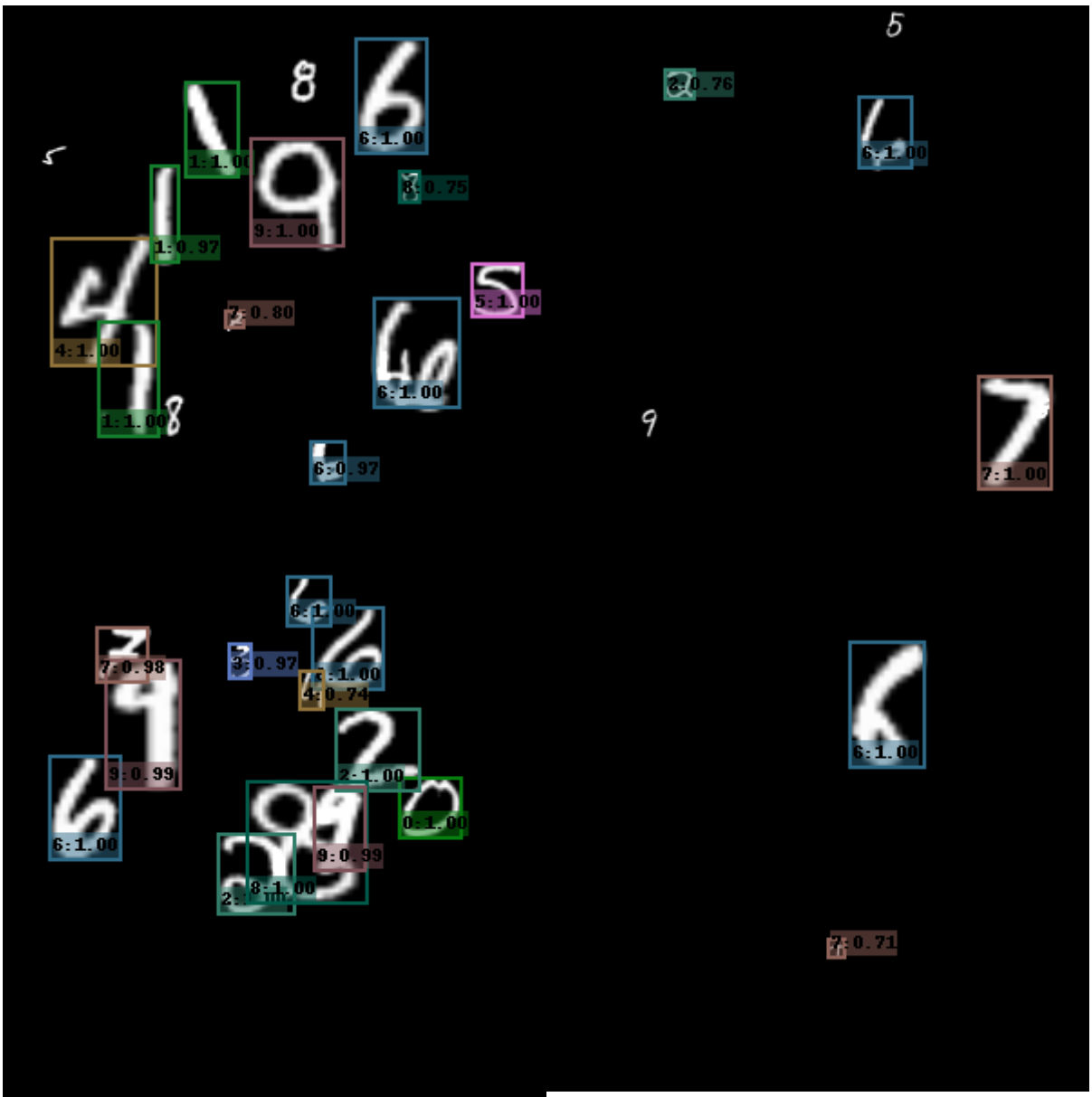

2:1.00
2:1.00


9:0.93
8:1.00
4:1.00
5:1.00
9:0.91
9:1.00
3:1.00


5:0.80
7:1.00
5:1.00
6:0.99
7:1.00
3:1.00
3:1.00
4:1.00
2:0.82
2:1.00
1:0.98
6:1.00
7:0.98
8:1.00
6:1.00
8:0.99
4:1.00
9:1.00
1:1.00
7:1.00
8:0.89

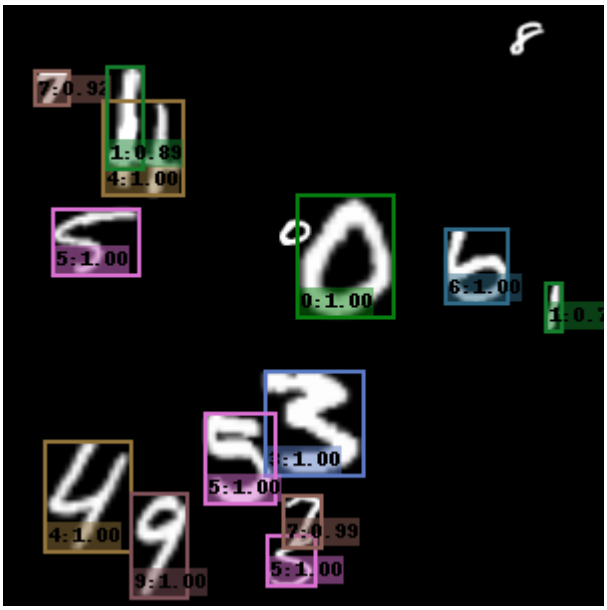

5:1.00
3:1.00
1:1.00

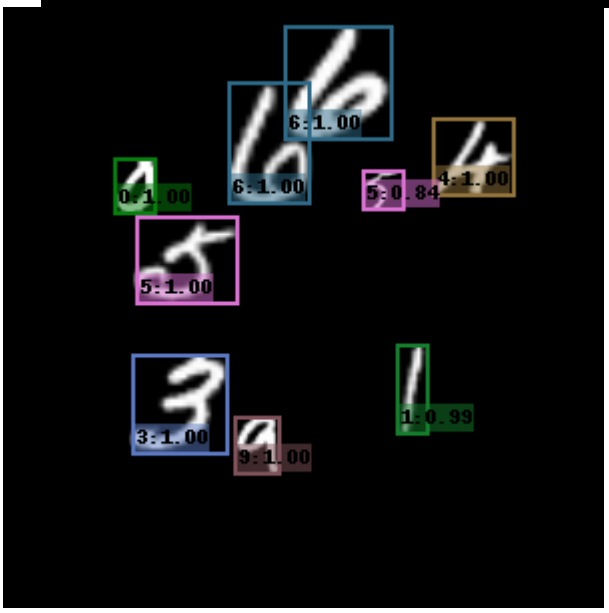

2:1.00


6:1.00
1:1.00
9:1.00
8:0.75
1:0.97
5:1.00
7:0.80
4:1.00
6:1.00
1:1.00
6:0.97
2:0.76
6:1.00
7:1.00
6:1.00
7:0.98
3:0.97
4:0.74
9:0.99
2:1.00
6:1.00
0:1.00
9:0.99
8:1.00
6:1.00
7:0.71


0:1.00
3:1.00
6:1.00
5:1.00
2:1.00
9:1.00
9:1.00
8:1.00
6:1.00
3:1.00
2:1.00
9:1.00
0:1.00
7:1.00
6:1.00
9:1.00
6:0.98
7:0.98
9:1.00
0:1.00
5:1.00
0:0.99
0:1.00

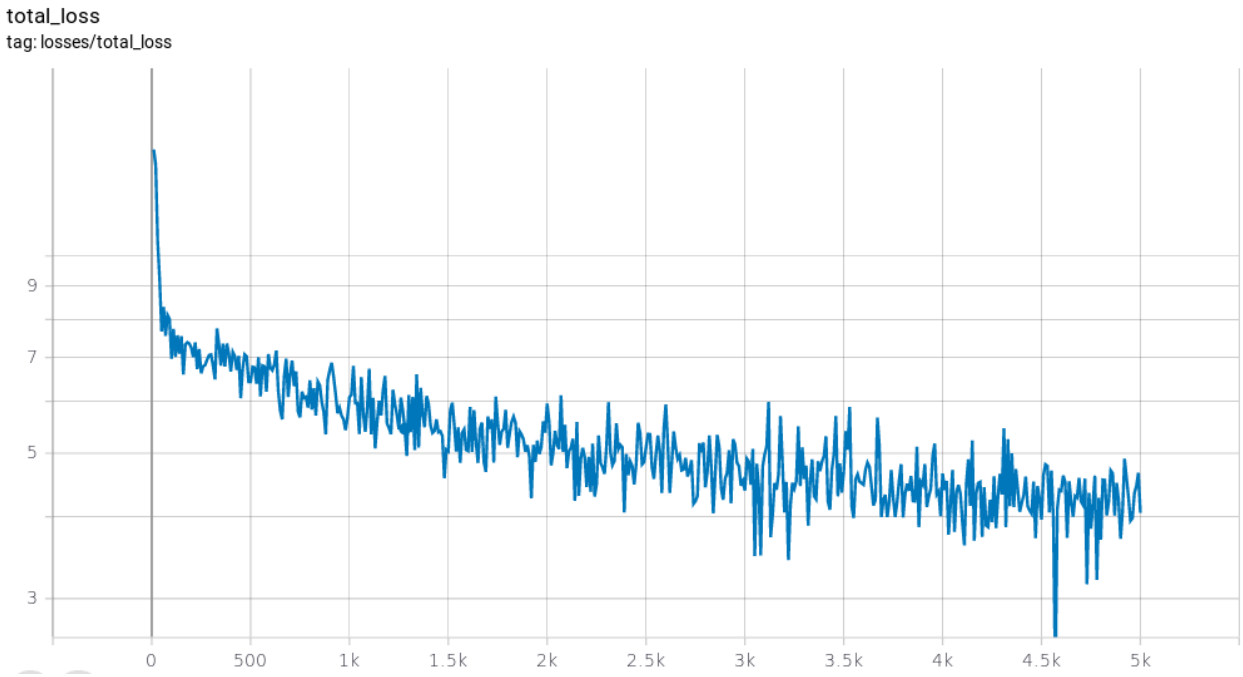

6:1.00
2:1.00
4:0.88
3:1.00
5
5:1.00
4:1.00
1:0.98
2:1.00


8
7:0.92
1:0.89
4:1.00
5:1.00
0:1.00
6:1.00
5:1.00
4:1.00
9:1.00
7:0.99
5:1.00


6:1.00
6:1.00
0:1.00
5:0.84
4:1.00
5:1.00
3:1.00
9:1.00
1:0.99

4f)

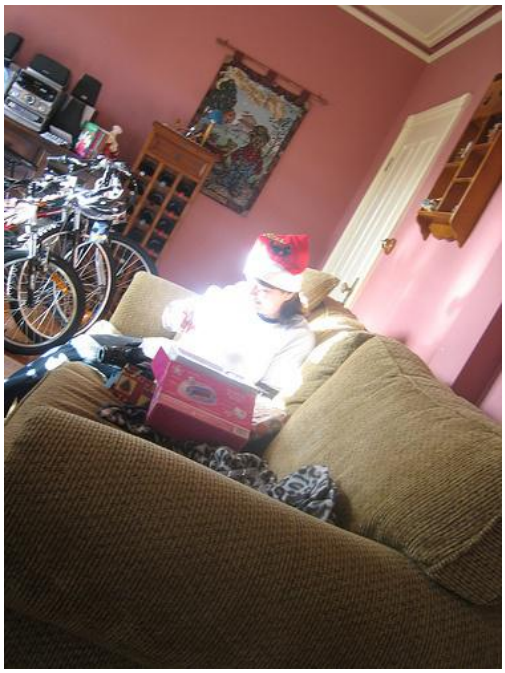
The final mAP is 0.4315



Below I appended the result of the tested image, the minimum confidence score for a bounding box to be displayed was at 0.3.


cat: 0.77


AGR
23
person: 0.48




bottle: 0.31
diningtable: 0.49


person: 0.36
person: 0.36
person: 0.31
person: 0.33