Pedestrian Recognition

Team 06:

Alberto Di Maro

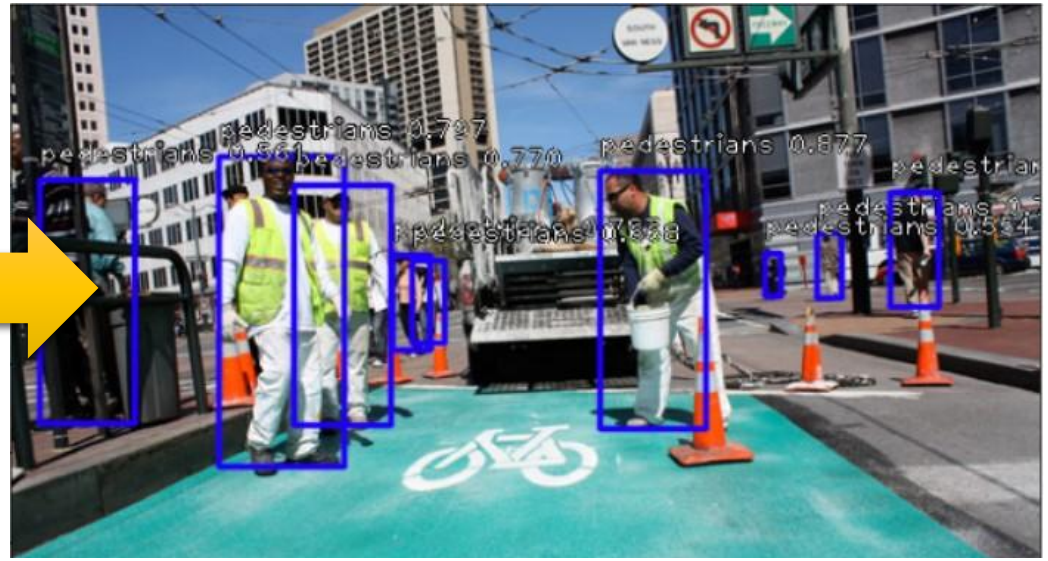
Abhishek Barandooru Janavejirao

Pietro Soglia

CONTENTS

SECTION 1:
INTRODUCTION

Case Study



Dataset

WiderPerson: A Diverse Dataset for Dense Pedestrian Detection in the Wild

(S. Zhang, Y. Xie, J. Wan, H. Xia, S. Li, and G. Guo)

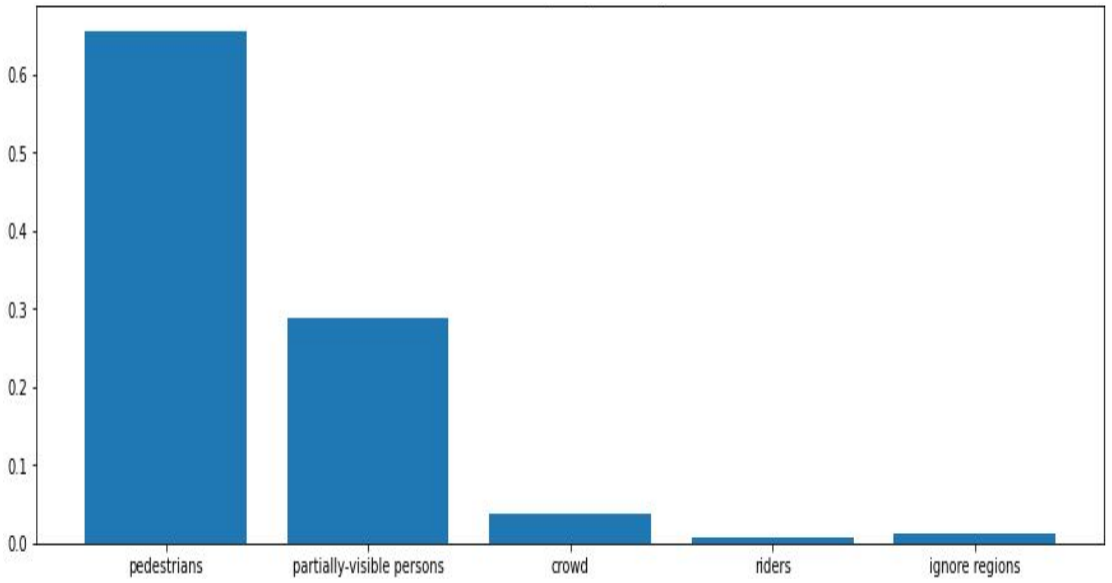
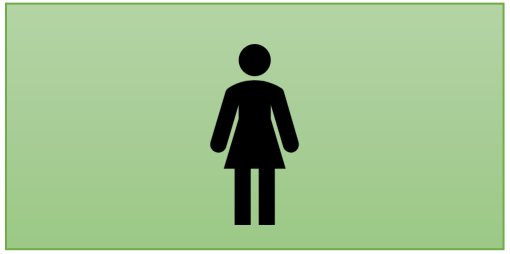


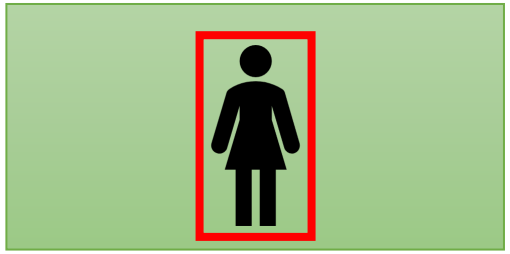
image.jpg



annotation.txt

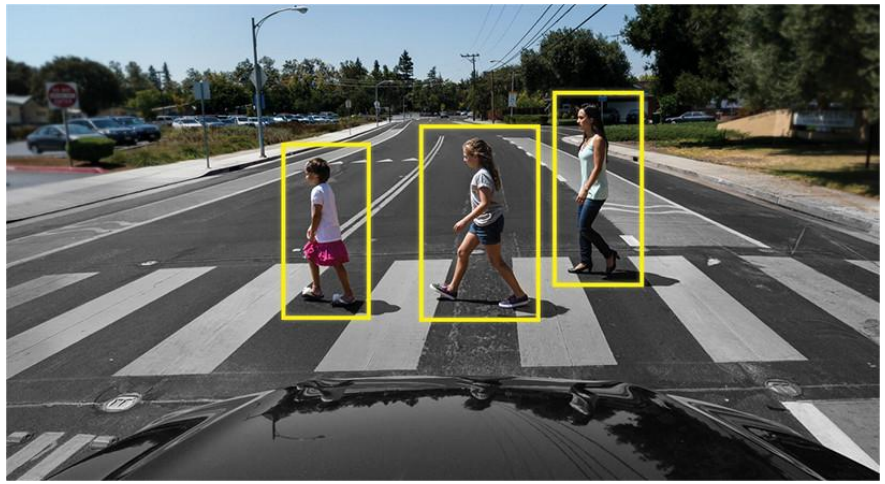
< # of annotations in this image = N >
< anno 1 >
< anno 2 >
.....
< anno N >

GROUND TRUTH

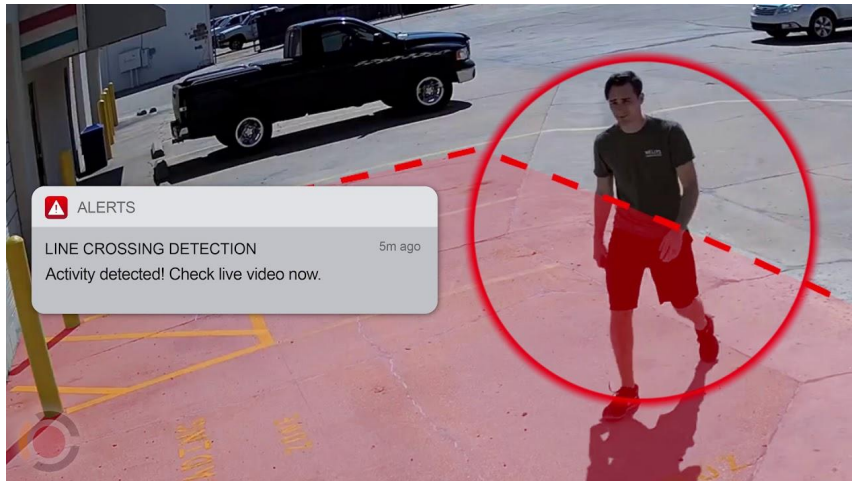


Some Applications

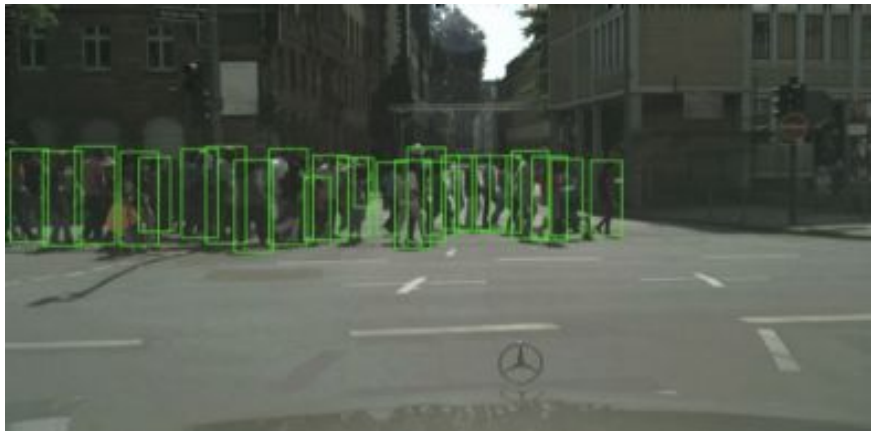
Autonomous Driving



Intrusion Detection



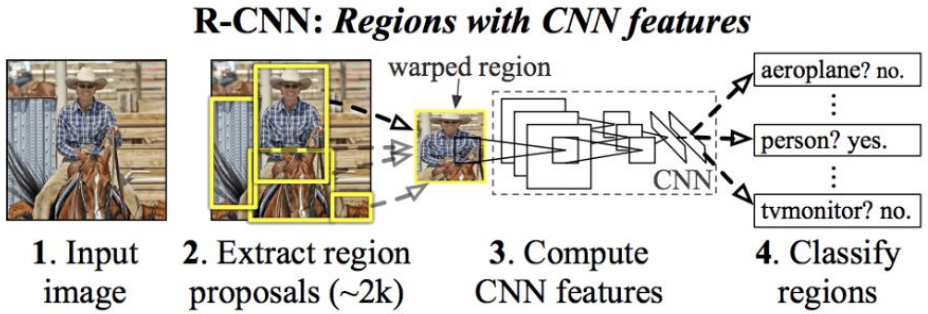
Crowd Counting



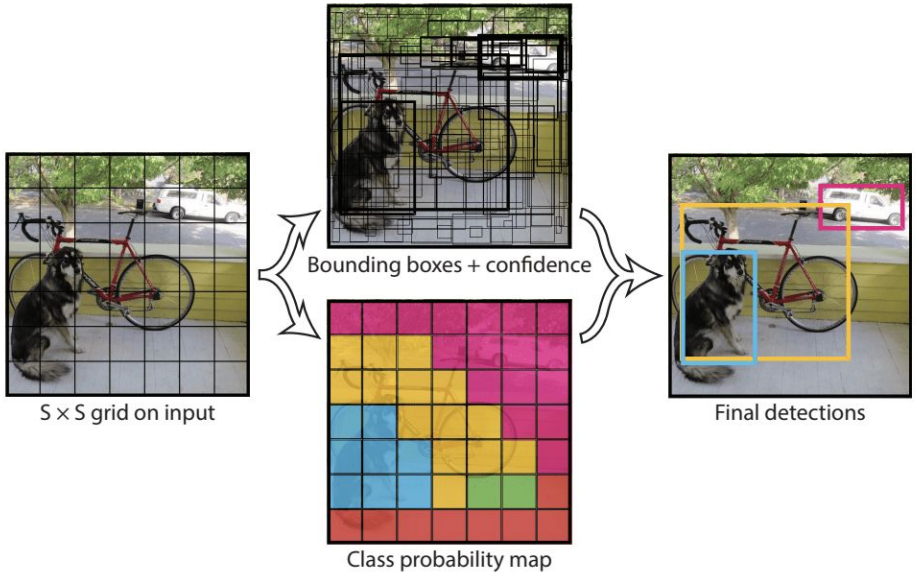
SECTION 2:
MODEL DESCRIPTION

Object Detection

Two-stages Detector

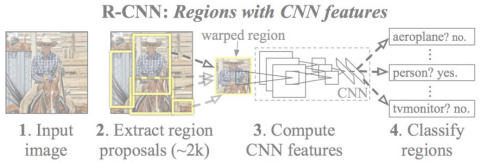


One-stage Detector

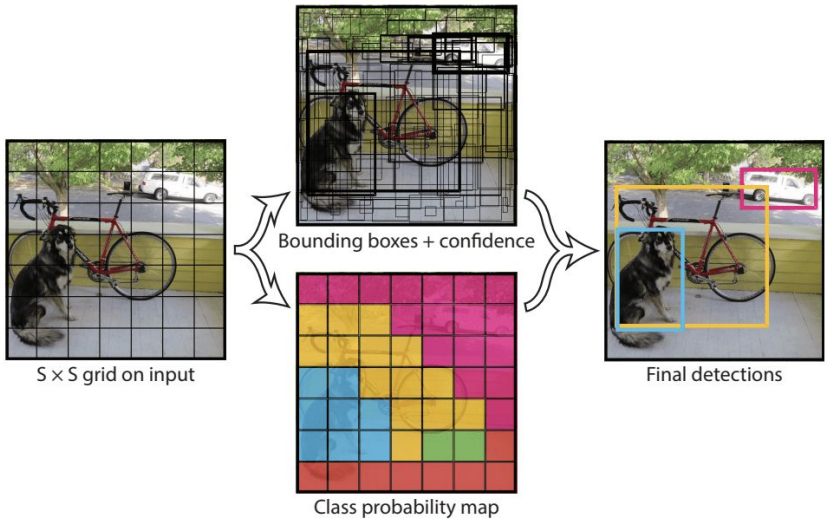


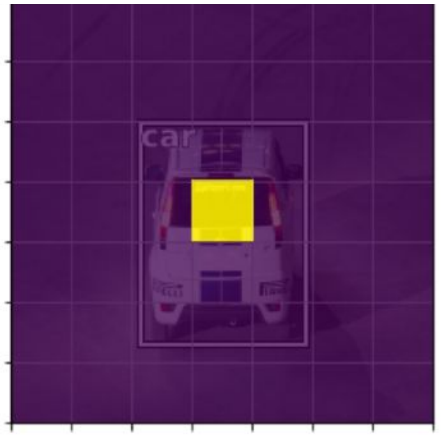
Object Detection

Two-stages Detector

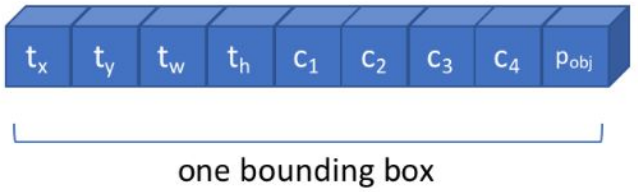


One-stage Detector



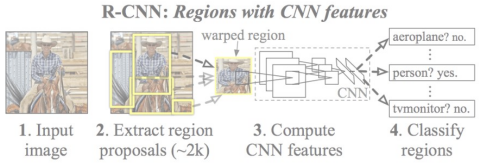


*supposing 4 classes

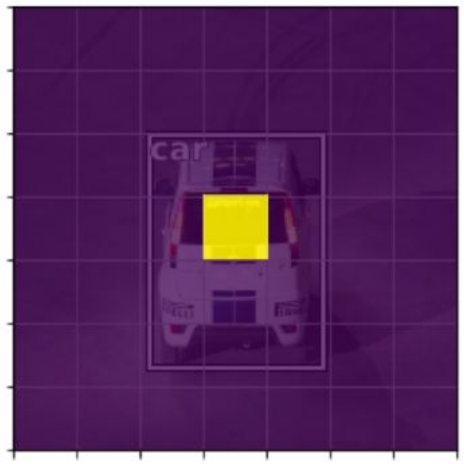


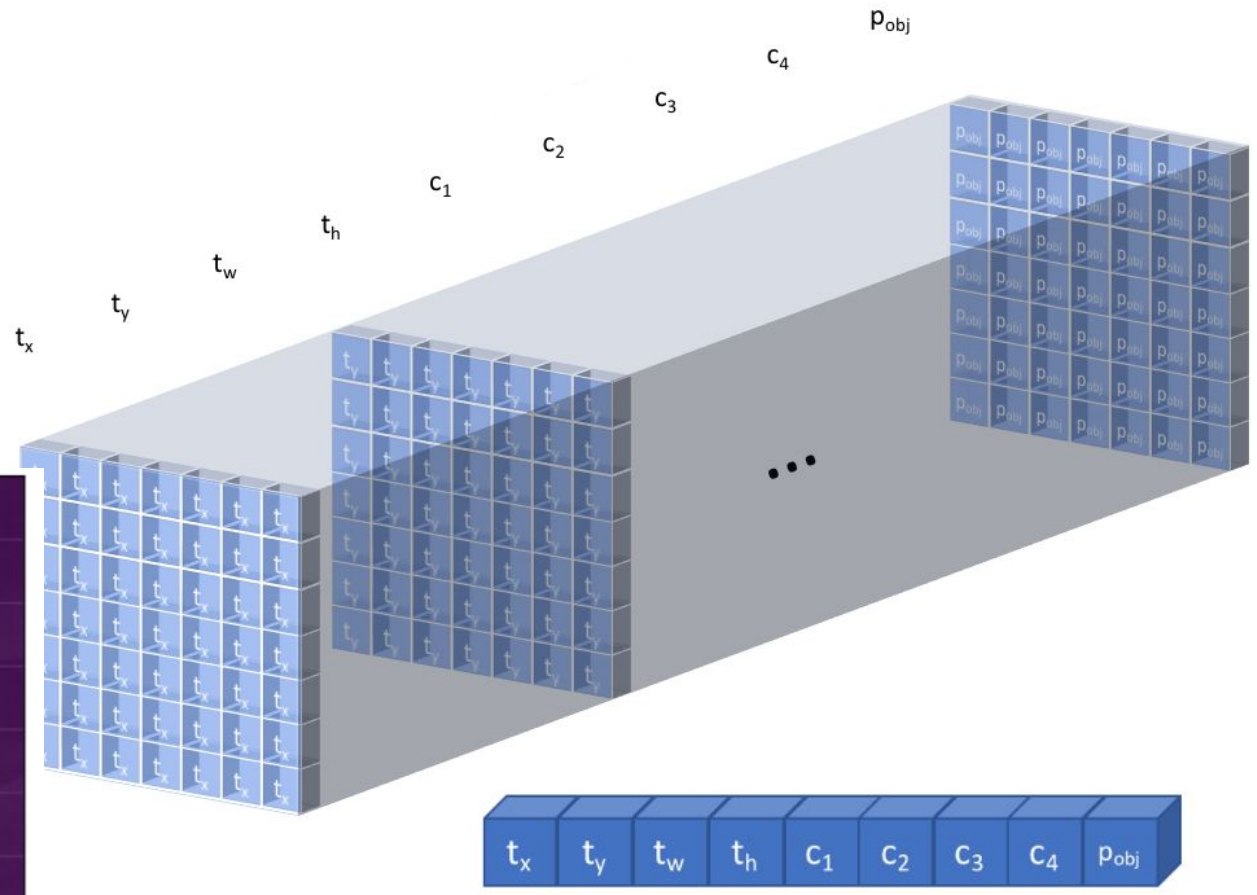
Object Detection

Two-stages Detector

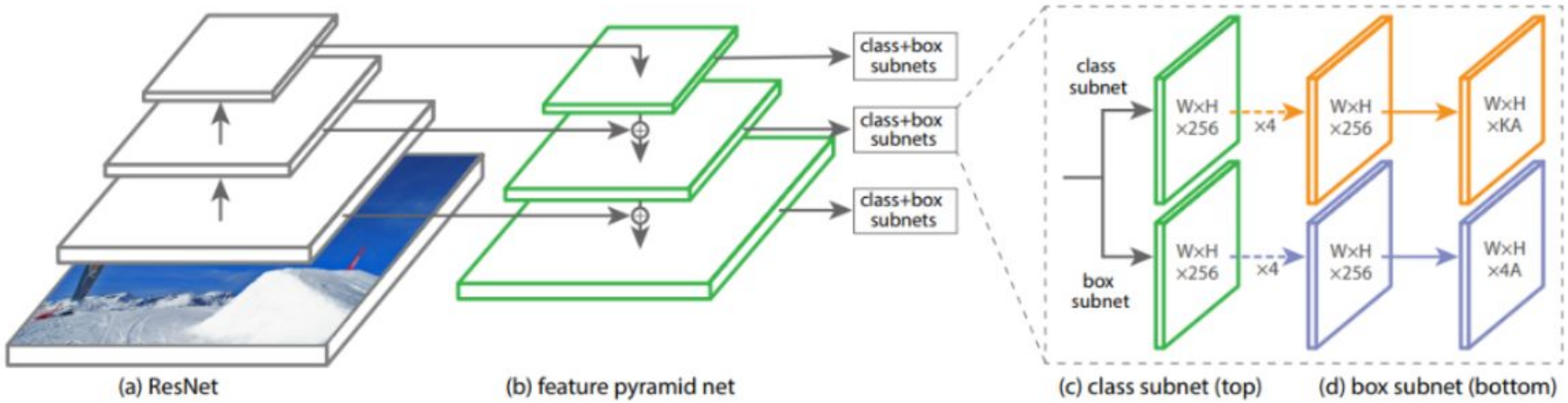


One-stage Detector

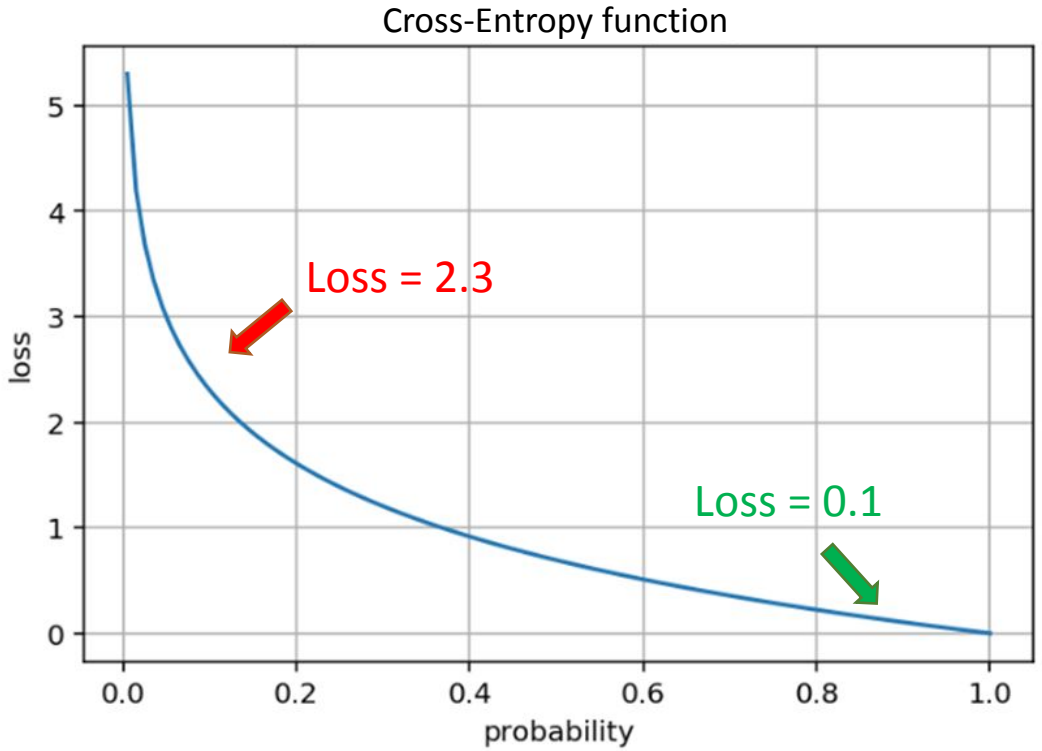




RetinaNet (1): Architecture

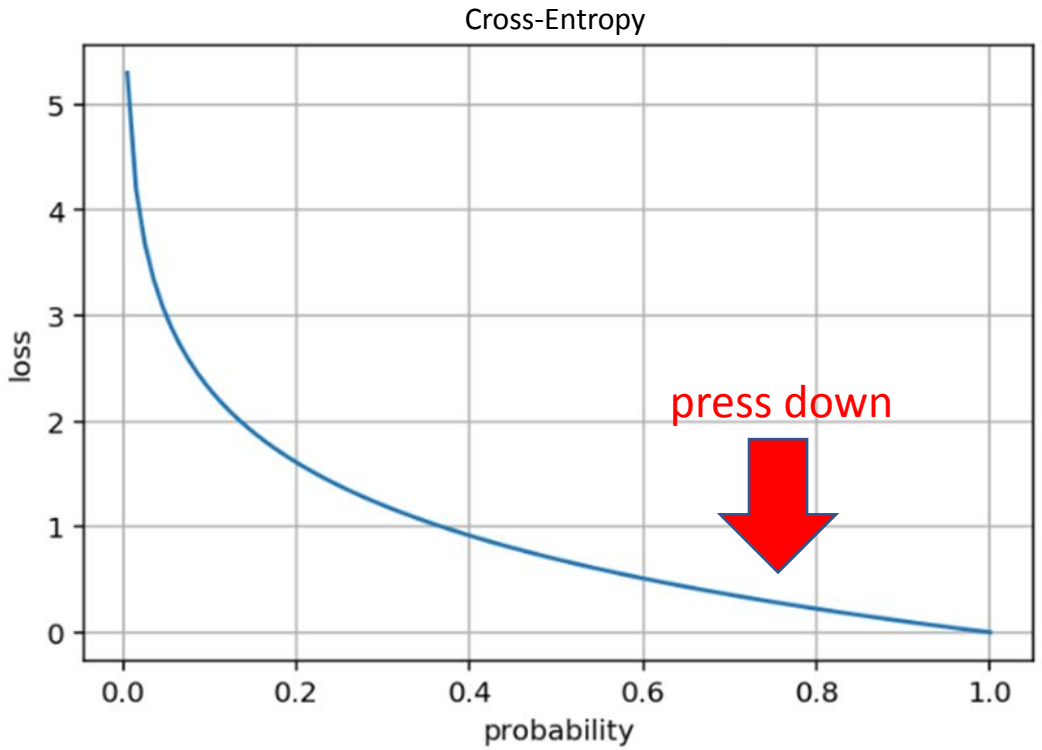


RetinaNet (2): Imbalance effects on the Loss



- 100.000 easy vs 100 hard examples
- The contribute is **40x** bigger from easy/non-significant/background examples

RetinaNet (2): Imbalance effects on the Loss

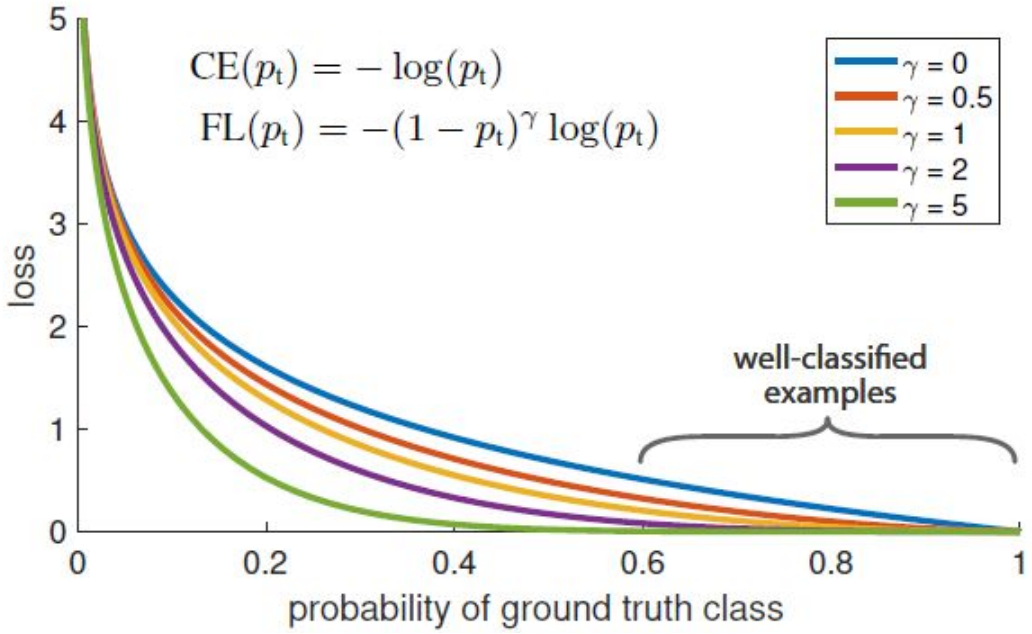


- 10000 easy vs 100 hard examples
- The contribute is **40x** bigger from easy/non-significant/background examples

Solution:

“press down” the Cross-Entropy function to reduce the loss value for easy examples

RetinaNet (2): Focal Loss

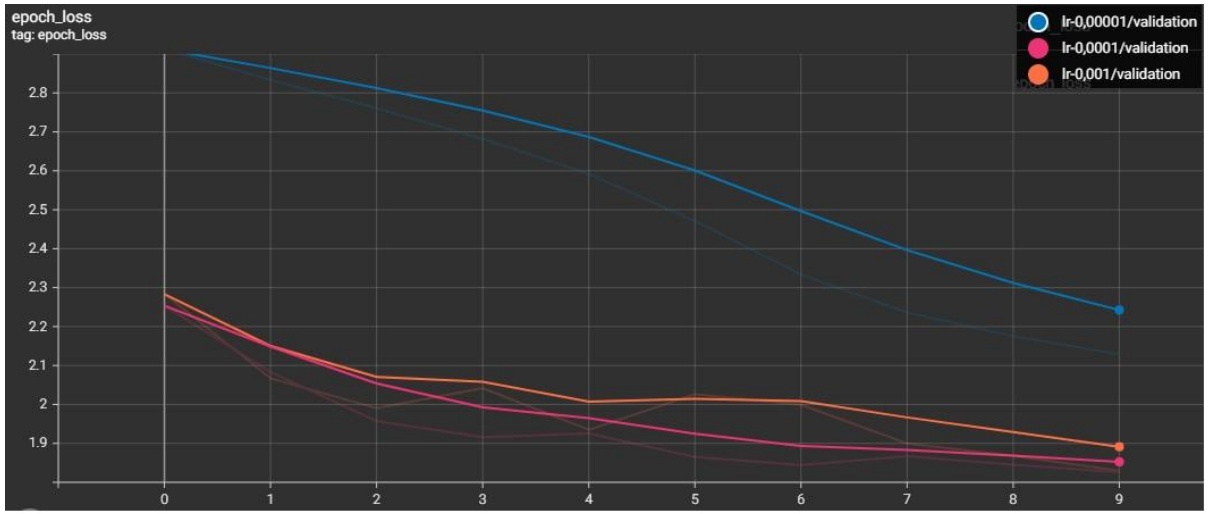


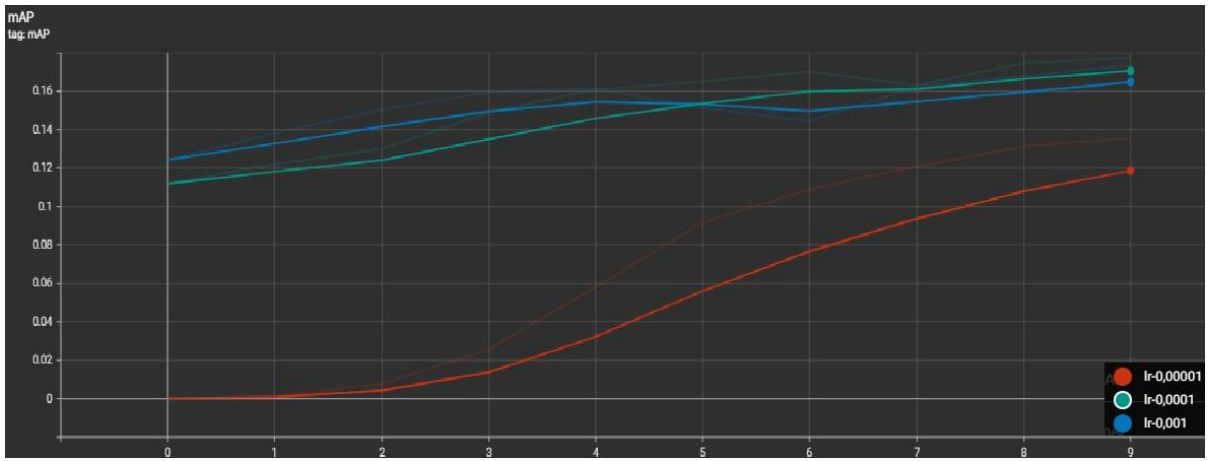
- Adds a factor $(1 - p_t)^{\gamma}$ to the standard cross entropy function
- Setting $\gamma > 0$ reduces the relative loss for well-classified examples (where $p_t > 0.5$), putting more focus on hard, misclassified examples
- Optimal results for $\gamma = 2$

SECTION 3:
APPLICATION

Setting:

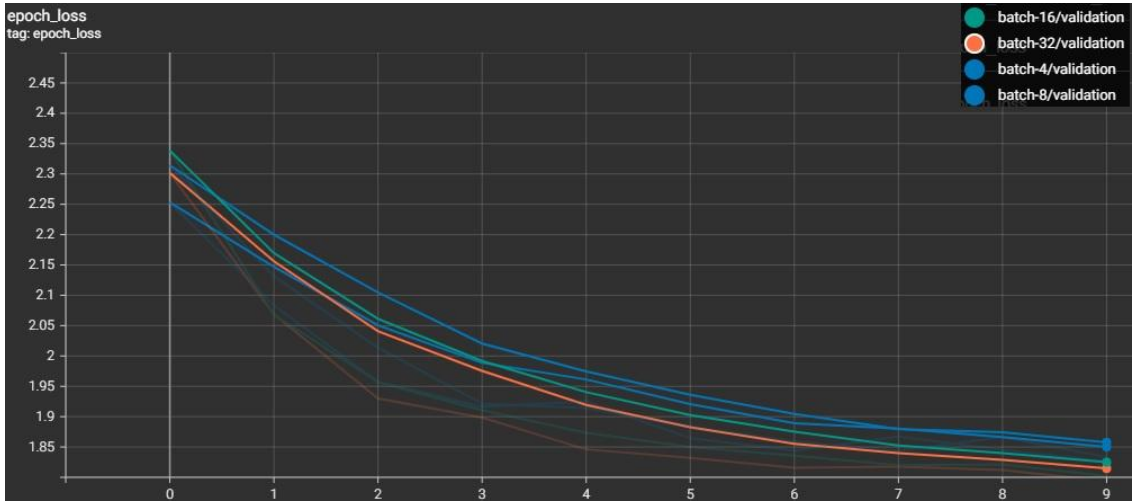
Learning Rate: **0.0001**

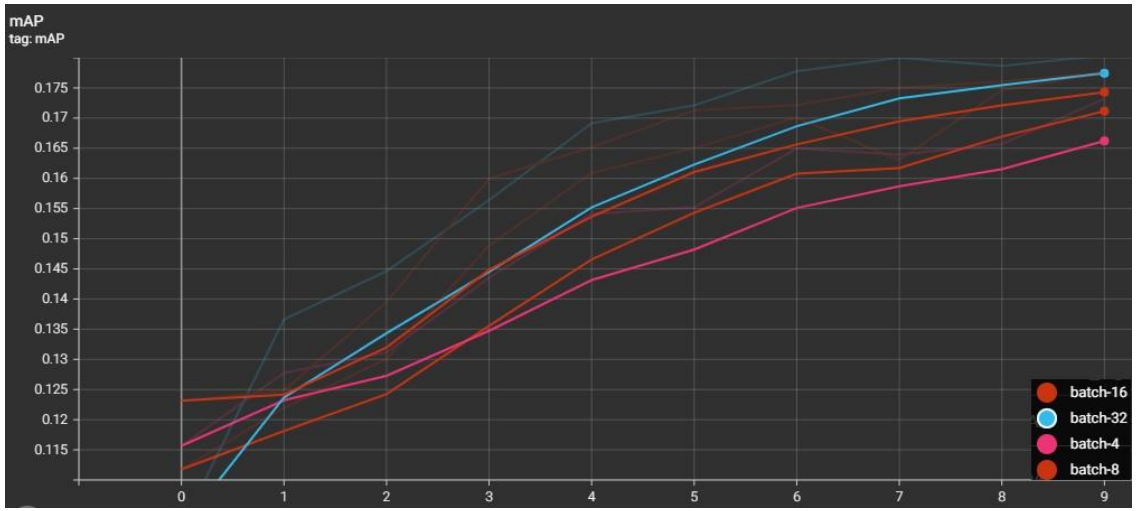




Setting:

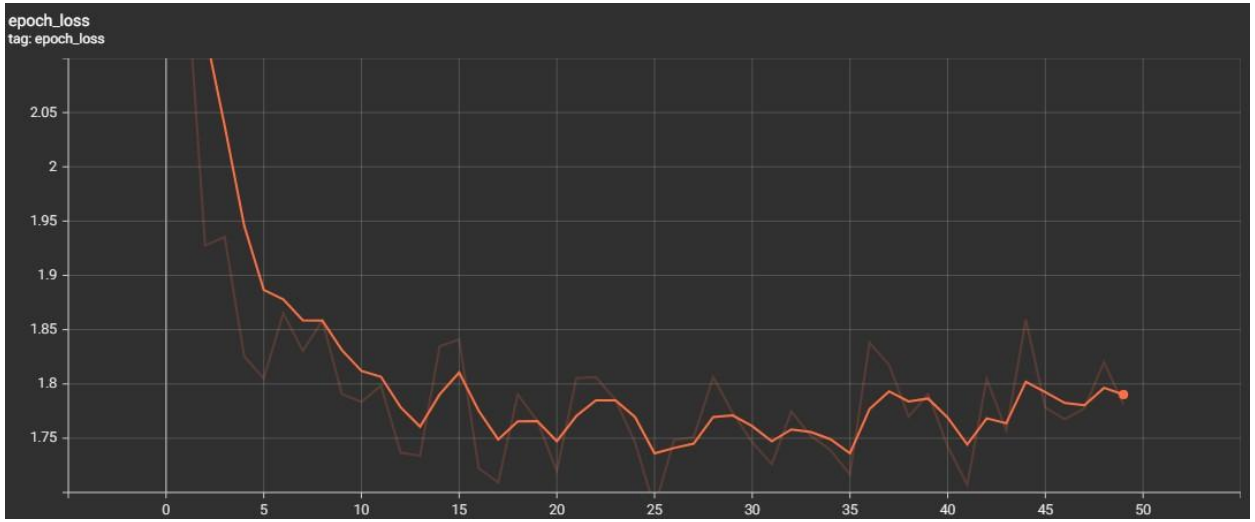
Learning Rate: **0.0001**
Batch Size: **32**





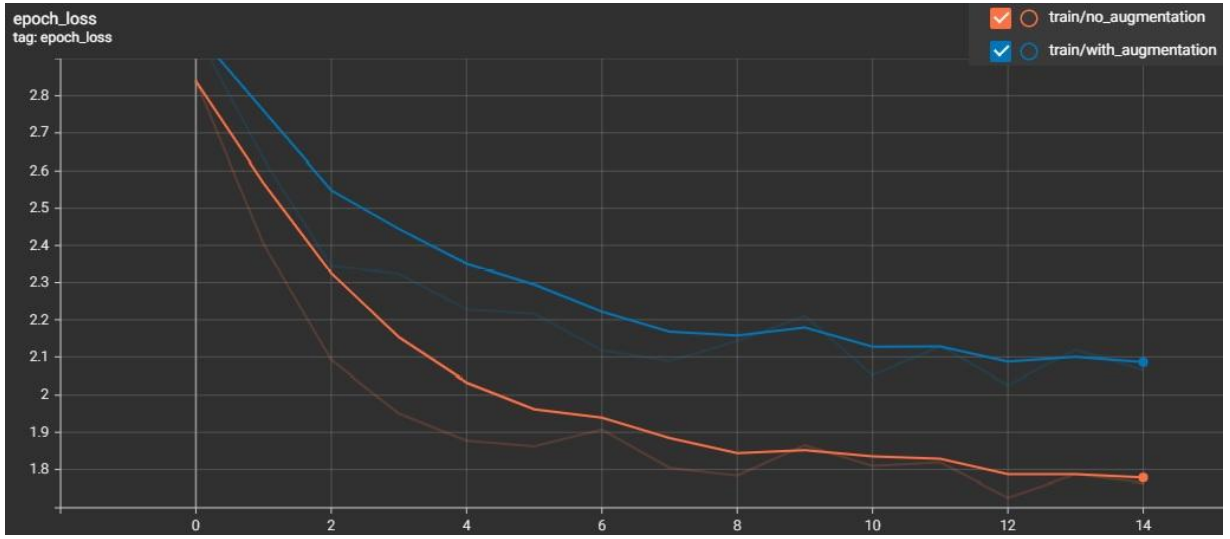
Setting:

Learning Rate: **0.0001**
Batch Size: **32**
Epochs: **15**



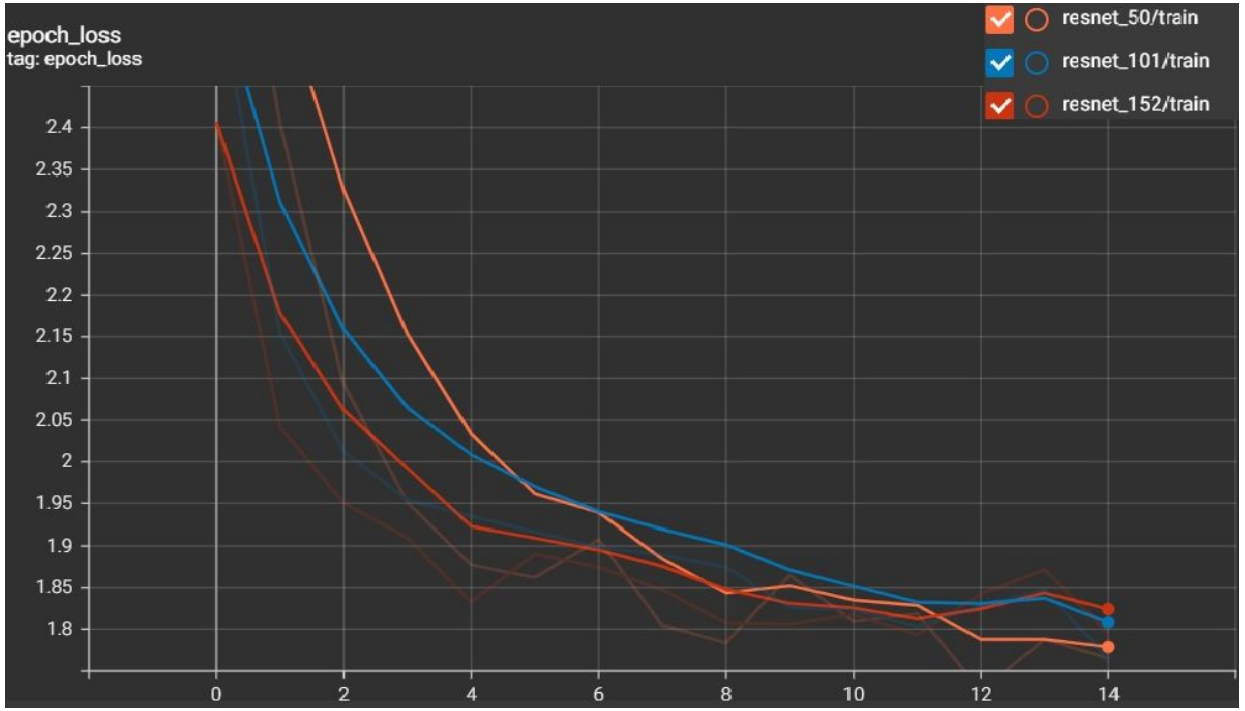
Setting:

Learning Rate: **0.0001**
Batch Size: **32**
Epochs: **15**
Augmentation: **No**



Setting:

Learning Rate: **0.0001**
Batch Size: **32**
Epochs: **15**
Augmentation: **No**
Backbone: **Resnet50**
(freezed)



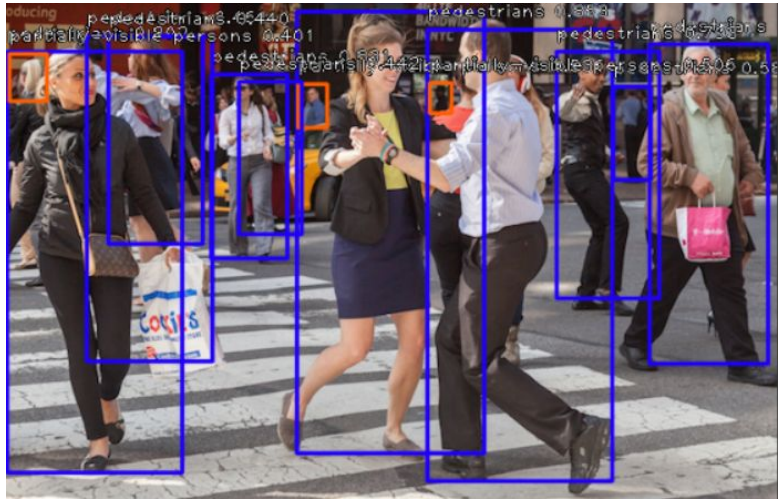
Results on Test Set

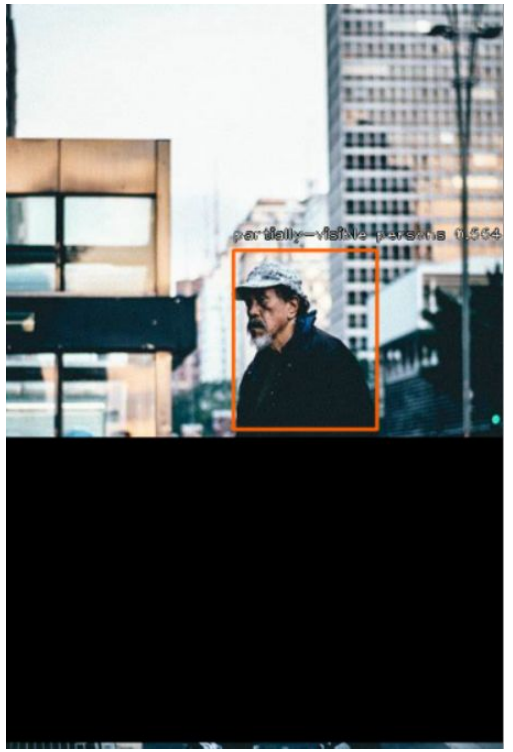
```
17833 instances of class pedestrians with average precision: 0.7904
185 instances of class riders with average precision: 0.0023
9335 instances of class partially-visible persons with average precision: 0.2263
409 instances of class ignore regions with average precision: 0.0000
661 instances of class crowd with average precision: 0.0001
Inference time for 1000 images: 0.2525
mAP using the weighted average of precisions among classes: 0.5702
mAP: 0.2038
```

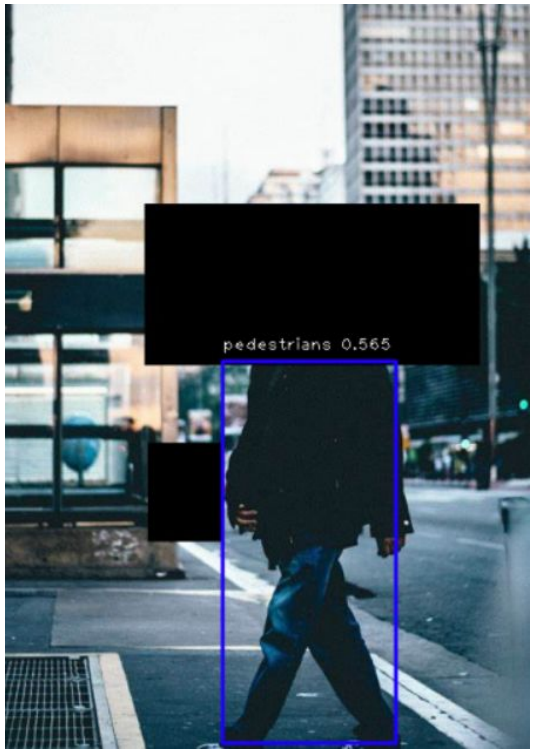
Ground truth vs Prediction

Inference


pedestrians 0.567


pedestrians 0.544
partially-visible persons 0.401


partially-visible persons 0.564


pedestrians 0.565

Thanks for your attention