

Deep Learning: Basic & Advanced

Clint Sebastian, Herman Groot, Egor Bondarev

For this assignment there are two possible advanced exercises. If you select this topic for your advanced assignment you only need to complete the basic exercise and ONE advanced exercise. You may choose to do either person re-identification or to build your own CNN architecture.

1 Basic - Object Detection

In this assignment you will experiment with TensorFlow's object detection API. Install TensorFlow and TensorFlow object detection API from the link below.

https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/installation.md

1.1 Pretrained models

Use the pretrained models (trained on COCO dataset) provided in the TensorFlow model zoo. Fine-tune the pretrained models (SSD MobileNet v2, Faster RCNN ResNet - 50 and RFCN ResNet - 101) on the COCO dataset itself. However, select 5 of the classes of your choice for fine-tuning. For all the questions below use only the 5 classes you have selected.

https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md

- Report the time taken of training and inference (per image) for 3 models listed above.
- Plot the precision-recall curve and report the results in Mean Average Precision (mAP).
- Why is SSD MobileNet v2 the fastest model? Explain your reason(s) why it is faster than other models?
- Which model offers the better performance? Faster RCNN ResNet-50 or RFCN ResNet-101? Explain why you think one is better over the other.

1.2 Train from scratch

Train your model from scratch and report the performance for SSD MobileNet v2.

- What happens if you don't use a pretrained model? How long does it take to train your model until you observe reasonable performance?

- Plot the precision-recall curve and report the results in Mean Average Precision (mAP). Compare the performance against when using a pretrained model.
- What are the disadvantages of a pretrained model?
- What would you recommend if you don't have a lot of computational resources nor a pretrained model for your problem? (you can be creative here)

2 Advanced - Object Detection

Implement your own Convolutional Neural Network architecture on top of one of the meta-architectures (SSD, Faster RCNN or RFCN). You can find more details on the implementation below.

Hint: Try to implement cheap and efficient mobile models so that you train and test fast.

https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/defining_your_own_model.md

- Explain the design of your network architecture and describe the details of your design choice? (Bad results from the network are also fine)
- If you were to apply your network for an embedded platform, how would you modify your network.
- Report the performance (precision-recall curves and mAP) and the time taken for a single image for inference using your network.

3 Advanced - Person Re-identification

As part of this exercise, you are going to experiment with person re-identification. To do so, perform the following steps and report on each of them:

1. Download the following person re-identification datasets: blue Market-1501 blue DukeMTMC
2. Perform the actual person re-ID
 - (a) Download the TensorFlow code from the paper 'In Defense of the Triplet Loss for Person Re-Identification', apply it on your own dataset and report your performance score that you achieved on your own dataset. You can find the paper [bluehere](#), and the code [bluehere](#).
 - (b) To help you get started, there are two files attached to this assignment.
 - i. The first one, named `train_embed_eval_Market1501.sh`, shows which steps are involved from training a model to determining the performance scores. The example code uses the public dataset Market1501. NB: the steps in this file may need to be customized based on the input dataset.
 - ii. The second one is a MATLAB script that contains a tool that you can use to create the required csv files.

4 Evaluation

For this project you must write a short report (6 pages single column maximum) preferably in L^AT_EX or in other word processing software such as Microsoft Word addressing at least the following points:

4.1 Advanced: Person Re-Identification

- How did you implement each step?
- What are the exact processes involved in training the network?
- Comment on your results with each dataset. Can you think of any possible improvements?
- Provide the source code (this can be delivered electronically to h.g.j.groot@tue.nl).

4.2 Basic & Advanced: Object Detection

- Your code must be readable and should include comments. Instructions on running the code must be provided.
- Zip (or rar) your scripts and send it. Note: Do not zip the whole TensorFlow object detection API, just the relevant parts.
- The final zip file must contain your report and scripts. For questions and submission, email Clint Sebastian (c.sebastian@tue.nl)

Additionally, a brief demonstration of your code is necessary for evaluation. During this demonstration you will run your code live while showing some intermediate results and explaining them.