Project Report

TDT4265 Computer Vision and Deep Learning

Sivert Utne

May 7, 2022

Contents

General Information

For convince i have created a couple scripts that can be used to reproduce each of the tasks results in the project. These are all located in the **tasks** folder. To run a task all that is needed is to (from the root of the project) do:

`./tasks/<task>-<subtask>.sh`

Due to time constraints all training was limited to 1000 iterations, with the exception of task *2.5* when training on the updated dataset with my best model.

2 Model Creation

2.1 Creating the first Baseline

The complete model is shown in Table 1 and the hyperparameters used are listed in Table 2.

Table 1: The Improved Model. Using output_channels = [128, 256, 128, 128, 64, 64]

Is Output	Layer Type	Number of Filters	Kernel Size	Stride	Padding
	Conv2d	32	3	1	1
	ReLU	-	-	-	-
	MaxPool2d	-	2	2	-
	Conv2d	64	3	1	1
	ReLU	-	-	-	-
	Conv2d	64	3	1	1
	ReLU	-	-	-	-
	Conv2d	output_channels[0]	3	2	1
Yes - Resolution 32×256	ReLU	-	-	-	-
	Conv2d	128	3	1	1
	ReLU	-	-	-	-
	Conv2d	output_channels[1]	3	2	1
Yes - Resolution 16×128	ReLU	-	-	-	-
	Conv2d	256	3	1	1
	ReLU	-	-	-	-
	Conv2d	output_channels[2]	3	2	1
Yes - Resolution 8×64	ReLU	-	-	-	-
	Conv2d	128	3	1	1
	ReLU	-	-	-	-
	Conv2d	output_channels[3]	3	2	1
Yes - Resolution 4×32	ReLU	-	-	-	-
	Conv2d	128	3	1	1
	ReLU	-	-	-	-
	Conv2d	output_channels[4]	3	2	1
Yes - Resolution 2×16	ReLU	-	-	-	-
	Conv2d	128	3	1	1
	ReLU	-	-	-	-
	Conv2d	output_channels[5]	2	2	0
Yes - Resolution 1×8	ReLU	-	-	-	-

Table 2: Hyperparameters for the improved model.

Hyperparameter	value
Optimizer	*SGD*
Batch Size	32
Learning Rate	0.005

2.2 Augmenting the Data

2.3 Implementing RetinaNet

2.3.1 Feature Pyramid Network

This new model is implemented across several files. Firstly i wrapped a pretrained RetinaNet model in the file **ssd/modeling/backbones/resnet_model.py**. This model is then used as the backbone of the FPN, which is implemented in the file **ssd/modeling/backbones/fpn_model.py**. The use of this model without any further modifications are done in the config file **task_2_3_1.py**.

2.3.2 Focal Loss

This change is implemented in the file *ssd/modeling/ssd_multibox_loss.py*. See config file **task_2_3_2.py** for use of these changes.

2.3.3 Deep Regression and Classification Heads

This change is implemented in the file *ssd/modeling/ssd.py*. See config file **task_2_3_3.py** for use of these changes.

2.3.4 Classification Head Bias

This is also implemented in the file *ssd/modeling/ssd.py*. See config file **task_2_3_4.py** for use of these changes.

2.4 Using knowledge from the Exploration

2.5 Extending the dataset

mAP when using the model from task 2.3.4 on the extended dataset for 2500 iterations (50 epochs).

We see that my model achieves a mAP of 0.898.

3 Discussion and Evaluation

3.2 Discussion and Qualitative Analysis

3.2.1 What are the strengths of the model?

3.2.2 What are the limitations of the model?

3.2.3 What is the reason for each modeling decisions impact?

3.2.4 Alternative methods to the modeling decision

3.3 Final Discussion

4 Going Beyond

4.2 Explaining the Model with CAM