Task 1 - Network Dissection

The goal of this task is to figure out which specific units of a model correspond with specific concepts that were learned by the model.

This analysis, called a Network Dissection, is done by running a large amount of images that show a wide variety of concepts (objects, scenes, parts, materials, textures, colors, ...) through the model and monitor the activation of each unit.
An activation threshold is determined for each unit that is only reach by a small percentage of all inputs. It is assumed that this small subset of inputs depict the same concept and that the unit learned to recognize this concept.
The activation map gets scaled to the size of the input and a binary separator is applied that divides the map into the areas that did and did not meet the threshold. This map now shows in which area of the input picture the concept was detected by the model.
To tell what exactly the concept is that, was learned by the specific unit, the model is run on a dataset like the Broaden dataset. This set contains images that are labeld pixel wise. By getting the activation-separation map of such an image one can look up what the shared label of the pixels in the activated area is, which is the label for the concept that was learned by the unit.

For this assignment the entire process of network dissection was done automatically by the tool CLIP-Dissect. Network dissection was performed on the last three layers of two different models:

- ResNet18 trained on ImagNet
- ResNet18 trained on places 365

The following commands were used to run CLIP-Dissect on the two models:

```
python describe_neurons.py --target_model resnet18_places --target_layers layer2,layer3,layer4 --device cuda --activation_dir tml/saved_activations --result_dir tml/results
```

```
python describe_neurons.py --target_model resnet18 --target_layers layer2,layer3,layer4 --device cuda --activation_dir tml/saved_activations --result_dir tml/results
```

CLIP-Dissect outputs a cvs-file that contains the concepts that were matched to each unit of the model. These files can be found in *task1/tml/results* and were used for further analysis. The analysis was performed using the code in *task1/anlyze_models.ipynb*.

It was compared how many concepts each of the models learned overall and in each layer. For a ResNet18 Model the number of units in each layer corresponds with the number of channels.

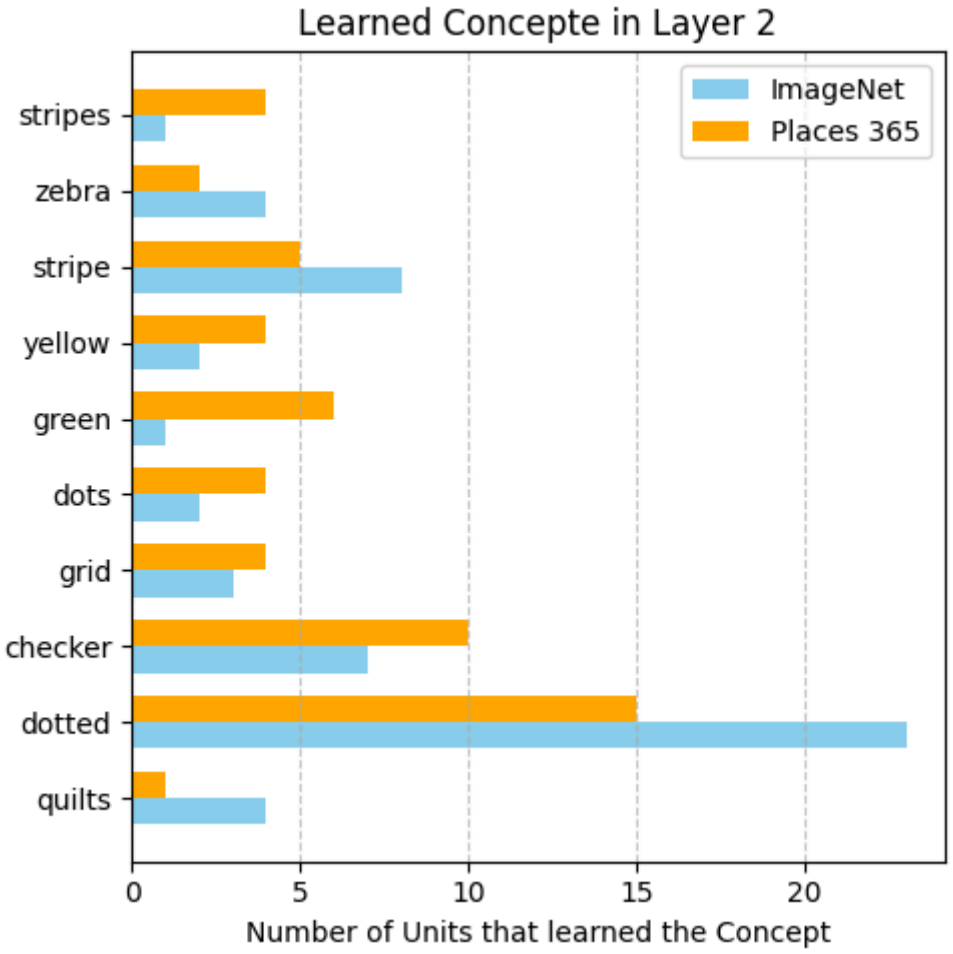
For the last three layer that is 128 units in layer 2, 256 units in layer 3 and 512 units in layer 4. This brings the maximum of concepts the model could learn to 896, if all the units would lean different concepts. Achieving this number is very unlikely since multiple units tend to learn the same concepts.

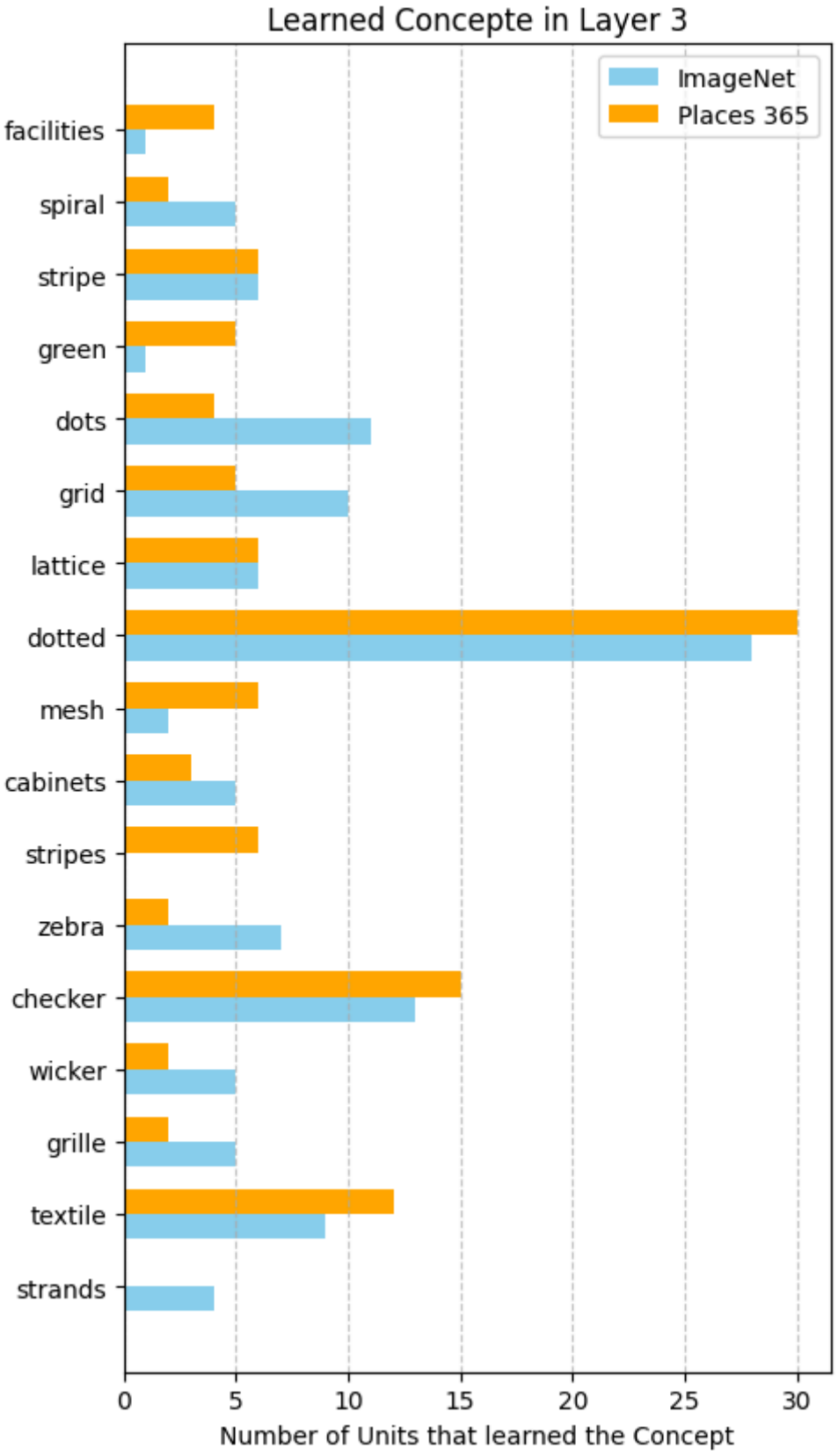
The following shows the number of unique concepts that were learned by each of the models. It can be seen that the model trained on the Places 365 dataset can distinguish a wider variety of concepts:

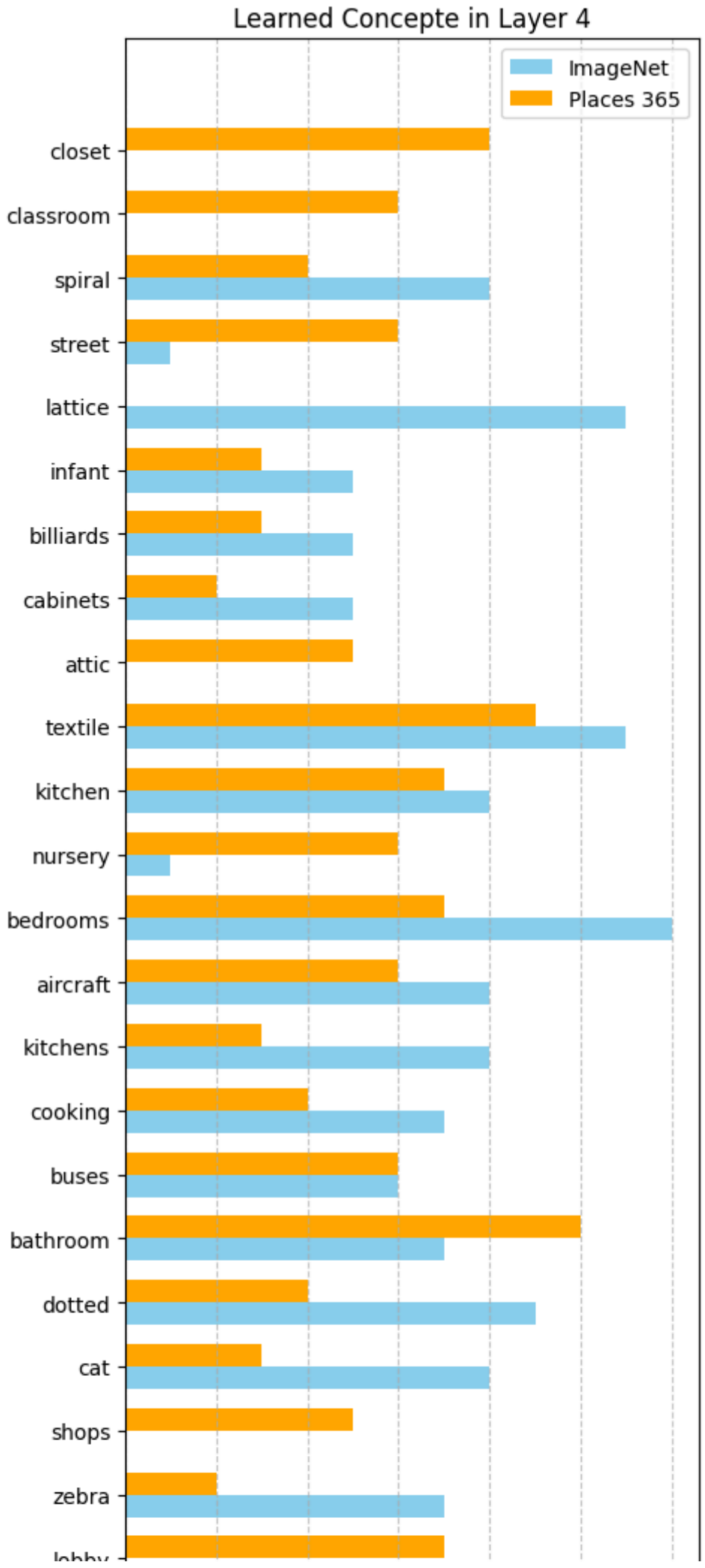
Concepts learned on ImageNet:

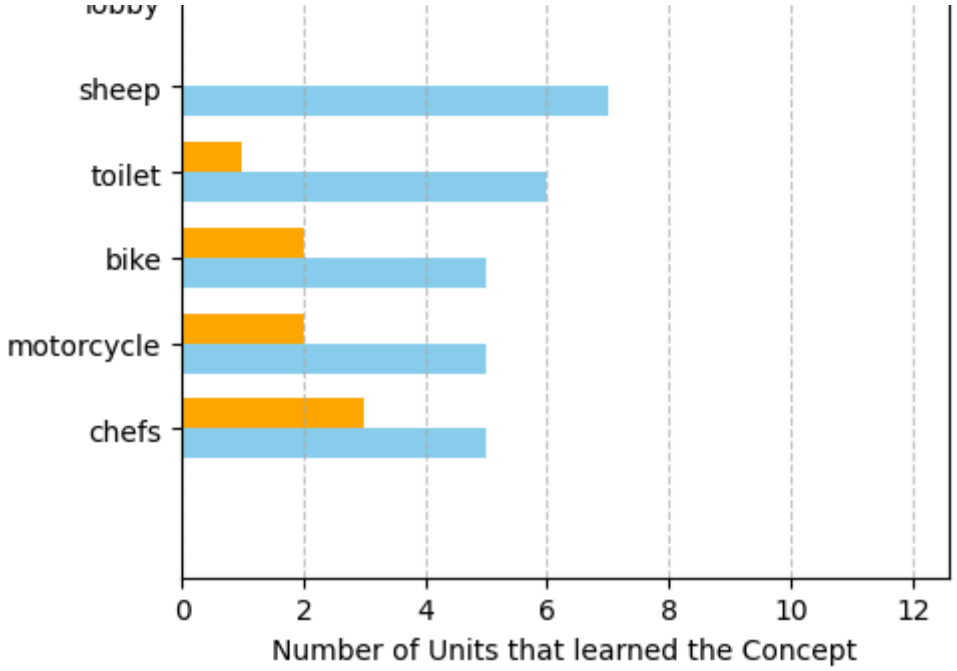
- Overall: 374
- Layer 2: 68
- Layer 3: 124
- Layer 4: 278
  Concepts learned on Places 365:
- Overall: 428
- Layer 2: 66
- Layer 3: 135
- Layer 4: 314

It was further analysed what what concepts the models learned in each layer and what what concepts were learned by the multiple units. The following graphs compare the number of units that leaned each concept on each of the layers. Concepts that were learned by 3 or less units are not included in the graphs to increase readability.


Learned Concepte in Layer 2
ImageNet
Places 365
stripes
zebra
stripe
yellow
green
dots
grid
checker
dotted
quilts
0
5
10
15
20
Number of Units that learned the Concept


Learned Concepte in Layer 3
ImageNet
Places 365
facilities
spiral
stripe
green
dots
grid
lattice
dotted
mesh
cabinets
stripes
zebra
checker
wicker
grille
textile
strands
0
5
10
15
20
25
30
Number of Units that learned the Concept


Learned Concepte in Layer 4
ImageNet
Places 365
closet
classroom
spiral
street
lattice
infant
billiards
cabinets
attic
textile
kitchen
nursery
bedrooms
aircraft
kitchens
cooking
buses
bathroom
dotted
cat
shops
zebra
lobby



Conclusion

One can see that the complexity of the learned concepts increases with the depth of the layer. Most of the concepts leaned by units on layer 2 are very basic like patterns (dotted, checker, ...) or colors (green, yellow, ...) while units on the deeper layer 4 learned way more complex concepts like specific scenes (bedrooms, bathrooms, ...) or objects (aircraft, sheep, ...).

Comparing the two models one will notice that the model trained on the Places 356 dataset was able to learn specific scenes like classroom, attic or lobby that were not learned by the model trained on ImageNet. Vice versa the model trained on ImageNet learned to detect specific objects like lattice or sheep that are not detected by the other model. This difference is based on the different kind of images found in each of the dataset: Places 356 includes mostly pictures of scenes and environments while ImageNet has a wider variety of pictures and is more object focused.