

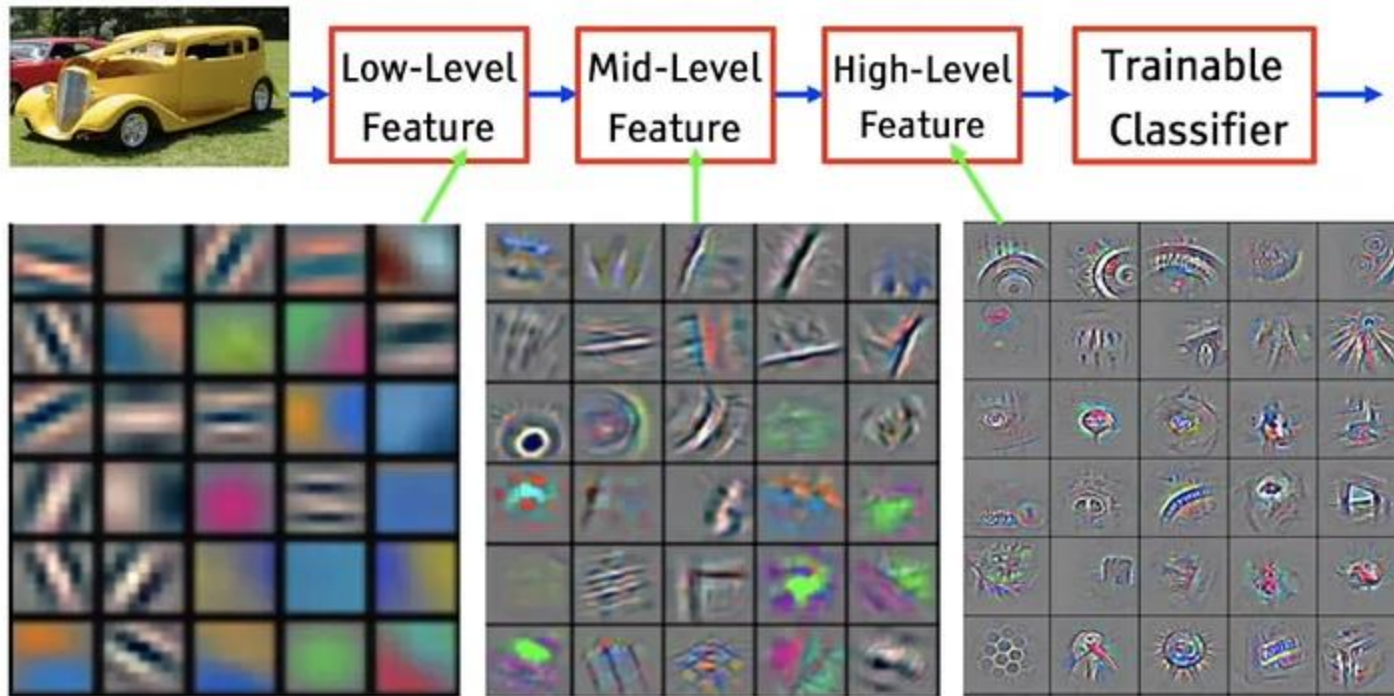
Exercise 8

Practical Data Science (PDS)



1. Feature Visualization
2. Transfer Learning
3. Assignment 4 – Car Classifications

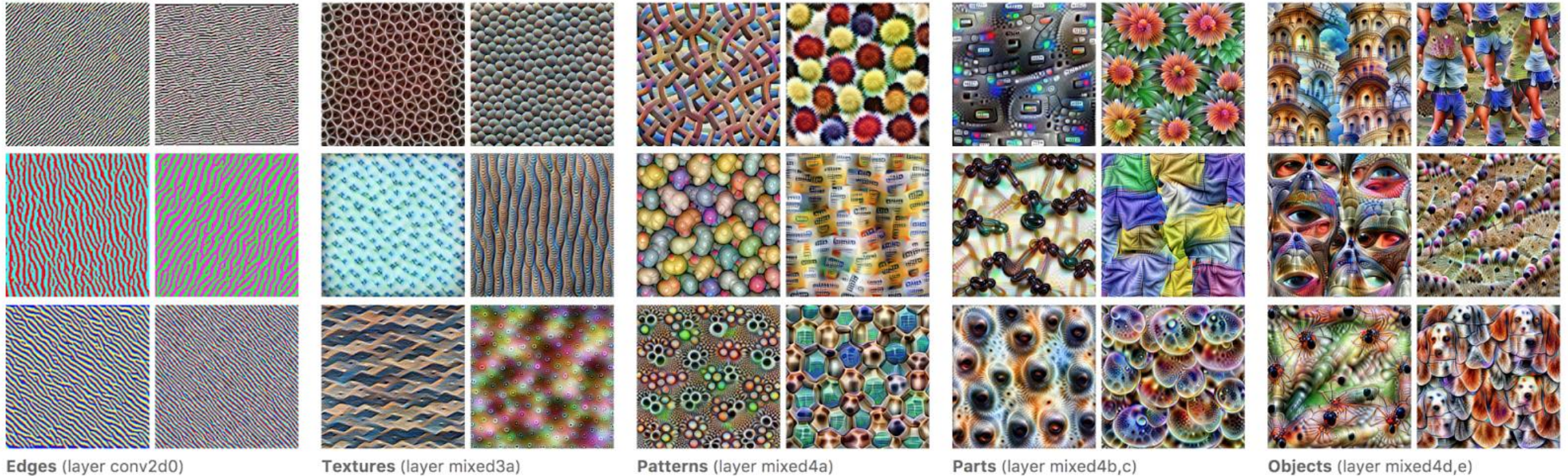
Feature Maps Learned by a CNN model



<https://cvg.cit.tum.de/media/teaching/ws2024/i2dl/9.convnets.pdf>

Examples: edges, corners textures, patterns concepts, objects

Another Example: Visualization with DeepDream



<https://research.google/blog/feature-visualization/>

Recap: Transfer Learning

- Transfer learning reuses knowledge from a pretrained model to solve a new but related task

1. Pre-training
(Large, general)



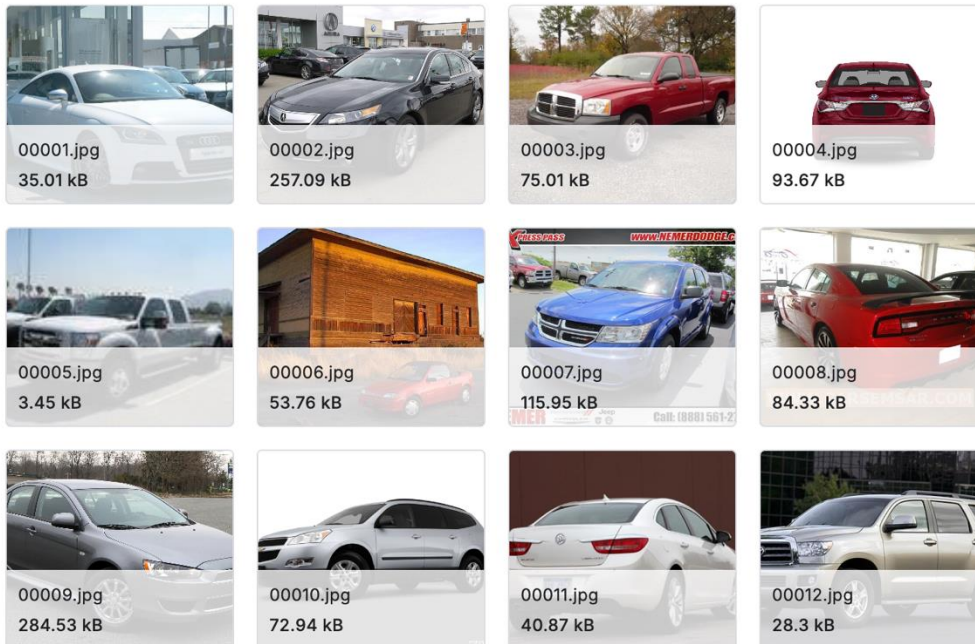
<https://serokell.io/blog/guide-to-transfer-learning>

- By default, vision_learner uses a pre-trained models from **Torchvision**:

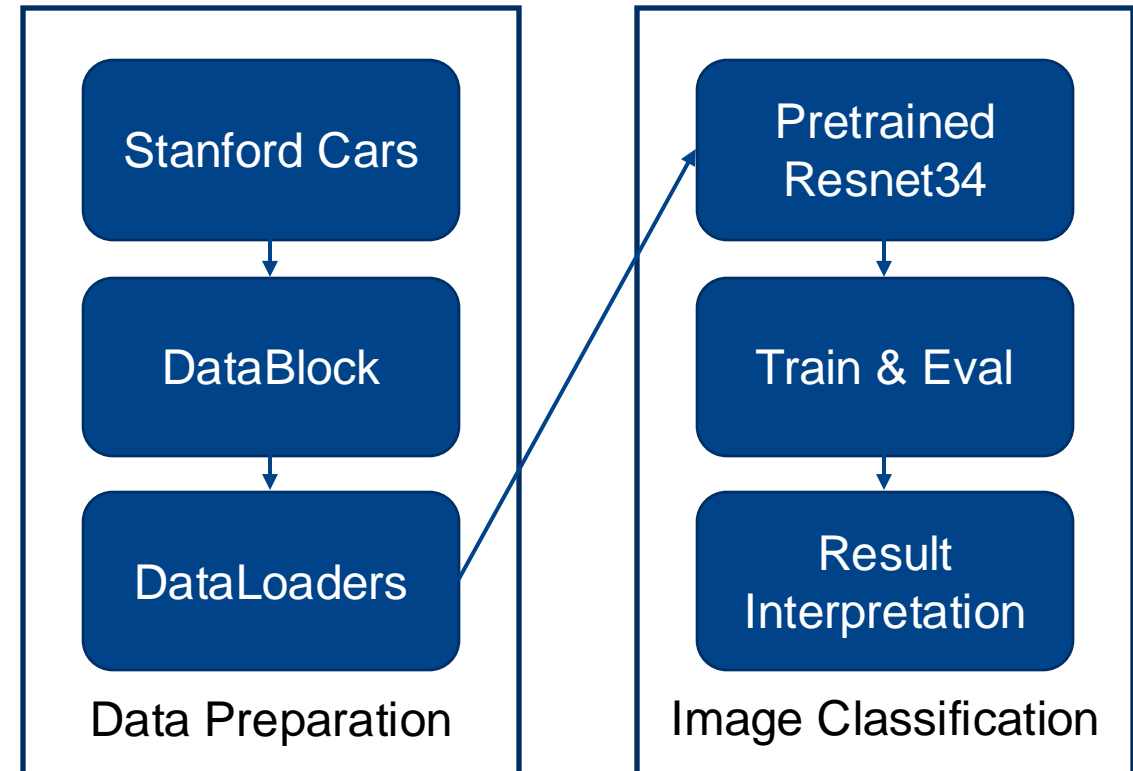
```
learn = vision_learner(dls=dls, arch=models.resnet34, metrics=accuracy)
```

- Here, ResNet34 was trained on ImageNet dataset: <https://www.image-net.org/>
- Then, we fine-tune it on Imagewoof dataset (not training from scratch)
- Benefits of transfer learning:
 - Reuse knowledge from the larger dataset
 - Reuse edges, corners, patterns, textures of simple objects (low/mid-level features)
 - Adapt to domain-specific concepts like dogs, wolf (high-level features)
 - Time-saving + overcome data scarcity

Assignment 6 – Car Classifications



<https://www.kaggle.com/datasets/jessicali9530/stanford-cars-dataset>



Passing Criteria

- Complete *ALL* tasks, including **3 text-based questions**
- Code is runnable
- All cells with **assert()** functions must not return errors

Your accuracy now should reach **72%** in the `assert()` function to pass the assignment.

```
In [ ]: # Load the best model and validate it
        learn = ...
        # call validate() and store it in the "results" variable
        results = ...

        # Don't modify this and run the code. The cell should not output any error.
        accuracy_v2 = results[1]
        assert accuracy_v2 > 0.72
```

Your trained model should achieve overall > 72% accuracy!

- Feature Visualization using DeepDream by Google Research: <https://distill.pub/2017/feature-visualization/>
- Stanford Lecture on Convolutional Neural Networks: <https://cs231n.github.io/convolutional-networks/>
- TUM Lecture on Convolutional Neural Networks:
 - Slides: <https://cvg.cit.tum.de/media/teaching/ws2024/i2dl/9.convnets.pdf>
 - Video: <https://www.youtube.com/watch?v=eWGIlkQq0Hzg&feature=youtu.be>
- Survey of Convolutional Neural Networks: <https://arxiv.org/pdf/2004.02806>
- Transfer Learning tutorial: <https://serokell.io/blog/guide-to-transfer-learning>