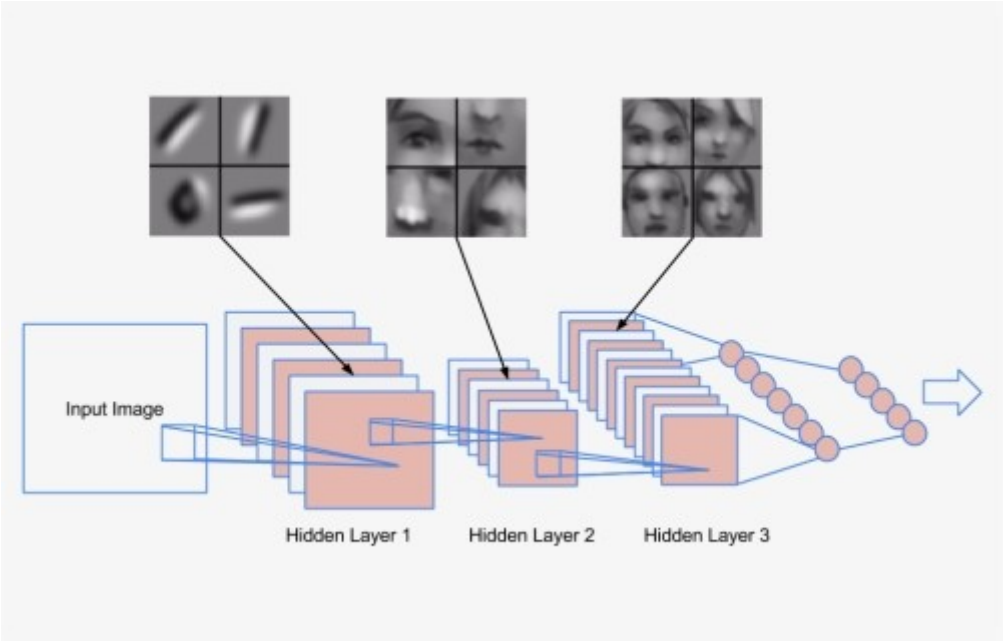
Kratos – Autonomous Subsystem QSTP-2021

Week 4

Transfer Learning

Here at Kratos, we are not endowed with supercomputers or state of the art GPUs that can train huge models in a reasonable amount of time. When we are stuck at home, this problem is magnified almost a billion times. It’s not reasonable to train models from scratch, so we apply the idea of transfer learning, and take pretrained models, and fit our dataset over it, which saves an extraordinary amount of computation time.

It is based on the idea that the features that a particular CNN architecture learns to identify on one dataset can be used on a different but similar dataset. The idea stems from the fact that when training a CNN network, the shallower layers of the architecture learn to identify the finer details like lines, strokes, edges etc while the deeper layers learn to identify more complicated features like a pattern, face etc.



Hence we can take a pre-trained model, remove the last few layers, freeze the shallower layers that have been trained to identify the finer details, attach a few layers in the end according to our needs, and then train the model on our customized dataset. This speeds up the training process and allows us to obtain decent accuracy over our test sets in spite of a smaller training dataset.

Resources:

All the resources of the previous week are applicable.

TASK

Your next task is to apply transfer learning and train any standard pretrained CNN-model (you can choose vgg-16 or any other if you want) and apply it on the given custom fruit dataset.
You have to download fruit-360 dataset from the Kaggle site: https://www.kaggle.com/moltean/fruits and train it on any 4 fruits of your choice. Use data augmentation on the images to improve the training score.
As mentioned in the previous assignment: your final submission should print out the model's accuracy after each training loop and in the end contain a predict function which takes as input an image and outputs its label, along with a demonstration of the function on 5-6 randomly chosen images from the test-dataset.

For your benefit, I will be providing helper files that can help you directly import the datasets for training. You are also free to extract the dataset on your own. For using google colab for this task, you can import the final dataset to your google drive and import it from there into colab. Check out https://colab.research.google.com/notebooks/io.ipynb