A2I2 – Convolutional Neural Networks

Summer 2020
Jochen Schmidt

Technische
Hochschule
Rosenheim

Convolutional Neural Networks (CNN) – Parameters

Given: A small CNN having the following architecture:

Input: RGB images of size 28x28
Convolutional layer with 32 filters of size 3x3, stride 3, zero-padding with 1
Convolutional layer with 64 filters of size 3x3, stride 1, zero-padding with 1
Max-pooling of size 2x2, stride 2
Flattening
Fully connected layer with 128 neurons
Fully connected layer with 10 neurons (output layer)

What size are the single layers? How many parameters have to be trained in each layer? How many in total?

Programming Exercise – Python / Tensorflow – MNIST

We will use the MNIST database (http://yann.lecun.com/exdb/mnist/) in this exercise. It contains (normalized) hand-written digits in images of size 28x28 as well as corresponding labels. The images are separated in training set (60,000) and test set (10,000). Tensorflow/Keras contains a function for loading the data set.

A Python-program training a simple CNN is provided for download in the Learning Campus. You can either run it using a local installation of Python/Tensorflow, or copy/paste the code to an Azure-Python-Notebook. A local installation is recommended (Azure contains an old version of Tensorflow 2.0.0; the code was tested using 2.2.0). GPU support is not required, CPU will be sufficient.

Modify the program in order to achieve better classification results, e.g. by changing the number of layers, filter sizes, pooling, activation functions, Dropouts, number of epochs, batch size.

The program saves log-files during training for visualization using Tensorboard (in the project directory, folder "logs"). Tensorboard can visualize the network-structure as well as training progress. On a local installation:

- Open a terminal window
- Change to your virtual Python environment if applicable:
  - Linux/MacOS (bash): `source ./venv/bin/activate`
  - Windows: `.\venv\Scripts\activate`
- cd to the CNN-project directory
- Start Tensorboard with

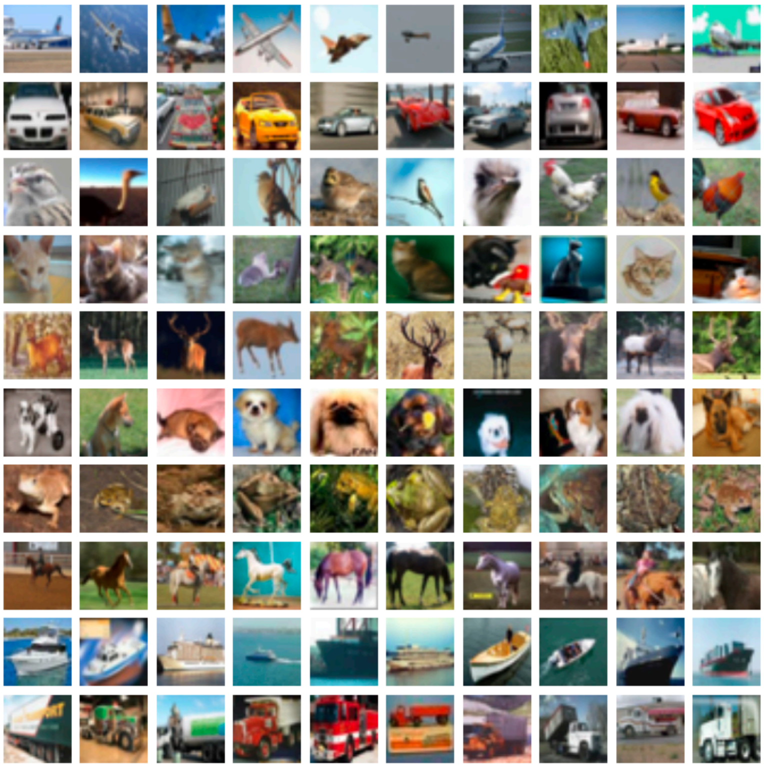
  ```
  tensorboard --logdir=logs/ --host localhost
  ```
- Copy the displayed link to a browser
- Start training the CNN

For using Tensorboard in Azure (this is not implemented in the sample code, a local installation is recommended) see: https://notebooks.azure.com/azureml/projects/azureml-getting-started/html/how-to-use-azureml/training-with-deep-learning/export-run-history-to-tensorboard/export-run-history-to-tensorboard.ipynb

CNN – CIFAR-10

This is an additional exercise, with longer training time (in particular if no GPU is used).

We use the CIFAR-10 database (http://www.cs.toronto.edu/~kriz/cifar.html). It contains 60.000 color images of size 32x32 in 10 classes (50.000 training and 10.000 test):



Create a CNN with good performance (on the test images). This is how you load the data set: https://www.tensorflow.org/api_docs/python/tf/keras/datasets/cifar10/load_data