Practical work 09 – 15/11/2018
Conv Neural Networks with Keras - part 2

Objectives

The objective of this PW is to understand some more advanced methods to train Convolutional Neural Networks (CNN). Another objective is to experiments with the **functional API** of Keras that allows to build more complex network structures.

As for the last practical work, we ask you to submit the solution for next week.

Submission

- **Deadline** : Wednesday 21 November, 12am
- **Format** : Zip with report and iPython notebook.

Exercice 1 Data Augmentation

Use the notebook `CIFAR10CNN_from_raw_augmented_data_stud.ipynb` available on Moodle as starting point.

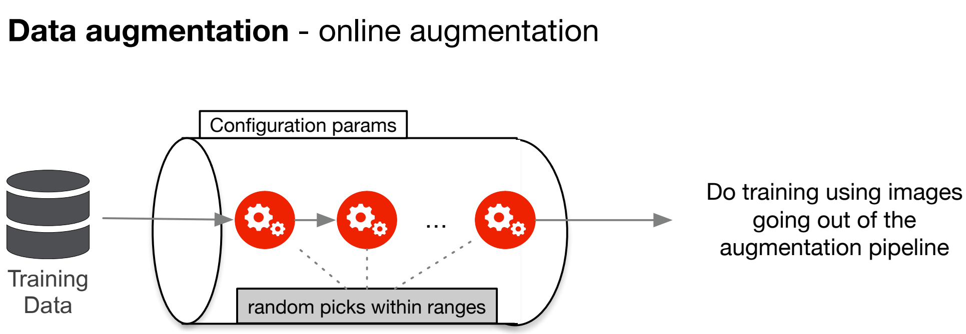

FIGURE 1 – Online data augmentation pipeline

a) Train a CNN

Define a CNN with the following structure : `CONV(32F,same)-RELU-CONV(32F,same)-RELU-MAXP(2)-CONV(32F,same)-RELU-MAXP(2)-DENSE`. Train the network using 10 epochs and batches of 128 images. Use a `categorical_crossentropy` loss and the `adam` optimizer.

b) Train the CNN with data augmentation

Re-read the section of the slides explaining the principles of data augmentation for images. Keras allows you to use an *online* data augmentation strategy as illustrated on Figure 1. Using the example given for the FashionMNIST dataset (cf. slides), implement a similar data augmentation strategy for CIFAR10.

You may try with different strategies and hyperparameter values of the data augmentation tool of Keras.

a) Report the accuracy on the train set and on the test set for your different experiments. Do you observe an improvement using data augmentation ?
b) Compare the evolution of the loss through the training epochs, with and without using data augmentation. Comment your observations.
c) If you tried with different data augmentation strategies, which one seems to give the best results ?

Exercice 2 Visualisation of activations

The objective is here to visualise the different activation maps in the network previously trained. The Figure 2 illustrates the principle for the first CONV layer on the first 6 filters of a given network.

Using the best of your network previously trained on CIFAR10 in exercise 1, implement a visulaizer for the activations at different layer outputs.

a) Read again the example of the visualisation presented on slides 25-26 of the class.
b) Implement a code to visualise all the filters at a given layer. Hints : use subplots to have a grid of images, use `for` loops to avoid code repetition.
c) Visualise the different activations maps of your network : outputs of CONV, RELU, MAXP. Comment on what you see.

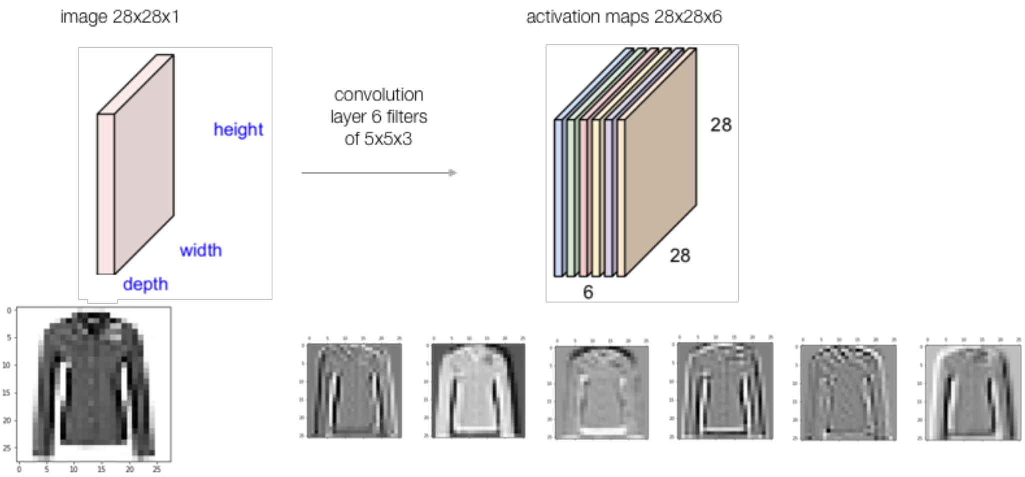

FIGURE 2 – Visualisation of activations of a CONV layer

Exercice 3 Deep architectures

In 2016, the **Inception-v4** architecture have been declared as outperforming ResNet and GoogleNet architecture on the ImageNet competition (see slide 64 of the class).

a) Download the paper presenting the architecture at https://arxiv.org/abs/1602.07261.

b) Read the paper up to the point you have an understanding of their strategy [1].

c) Re-explain in few phrases what you understood from the architecture doing comparison with the architectures presented in the class.

Exercice 4 Optional : Review Questions

a) Explain the notion of hierarchical features with CNNs.

b) Explain 2 strategies to visualise what is going on in CNNs.

c) What do we try to fight when using data augmentation ?

d) What are the implementation strategies for data augmentation ?

e) Explain the main differences for the deep architectures seen in class : AlexNet, VGGNet, GoogLeNet, ResNet. What were their intuitions when putting together such architectures ?

1. No need to understand all the details