Practical Work 1: Deep Convolutional Neural Networks

The goal of this practical work is to study the performance of different Deep CNN configurations for image classification. The data set proposed is the MAMe dataset, which contains images gathered from three different museums: The Metropolitan Museum of Art of New York, The Los Angeles County Museum of Art and The Cleveland Museum of Art. Data from all museums is aggregated by art experts into 29 classes of mediums (i.e., materials and techniques). The number of images available is 37,407. The training set contains 20,300 images (700 for every class), the validation set 1,450 (50 per class) and the test set 15,657 (unbalanced with respect to the training and validation set). The goal is to classify every image in one of the 29 predefined classes. Although the original data contains high-resolution images of variable shape, there is a 256x256 images version of the dataset, where all the images are downsampled to this size. The 256x256 images can be downloaded from https://www.kaggle.com/datasets/ferranpares/mame-dataset/data, and is the starting point of this practical work. More details can be found at https://hpai.bsc.es/MAMe-dataset and https://arxiv.org/pdf/2007.13693.

The idea is to test different models with different complexities (all models referred in this document are supposed to be Deep Convolutional Neural Networks). For every model we will use the following methodology:

1) Select a subset of (hyper) parameters so that your model presents underfitting
2) Select a subset of (hyper) parameters so that your model presents overfitting
3) Refine the (hyper) parameters so as to improve your results. You can modify the architecture, use data augmentation, dropout, batch normalization or any other regularization technique

Regarding the models, you **must** test **at least** (see the slides of the course for the difference between standard and non-standard architectures):

1) A standard architecture (preferably implemented by you) trained from scratch
2) A non-standard architecture trained from scratch

The idea behind "(preferably implemented by you)" is that you design of the whole network following the criteria of a standard architecture. There is no point in simply implementing an already existing standard architecture (you can do that in the optional part, see below), although you can of course take them as a source of inspiration.

Beyond that, you can explore on your own. For example, you can use a predefined (and trained) architecture and perform transfer learning, or use (modifying it or not) any other model you can find in the literature.

Write a brief document (**9 pages maximum**, including references and appendices) explaining your work:

1) Justify all your decisions.
2) Describe the architectures tested.
3) Describe all the details of the training and test procedures: optimization scheme (including batch size, learning rate, etc), weights initialization, data processing and/or rescaling, output averaging, etc.
4) Explain how the (hyper) parameters have been selected for every model/architecture.
5) Include those tables and figures that you consider important to describe the results obtained. Explain and reason the results presented in the tables and figures.
6) Draw your own conclusions with respect to the results obtained.
7) **If you DO NOT use ChatGPT (or another similar tool) in the document, declare it. Otherwise, explain in which parts of your work has been used. We want to evaluate your work, not someone else's.**

IMPORTANT: It is always a good idea to test the hardware+software environment so to obtain an estimation of the execution times. Waiting until the last week for starting the experiments is not a good idea.