ArtNet, Classification of Art Imamges

Giulia Silvestro GIULIA.SILVESTRO@STUDIUM.UNICT.IT

1. Model Description

We describe here **ArtNet**, a deep convolutional model that consists of four convolutional layers, to learn features mappings from 3-channel images, followed a fully connected layer and a classification layer which predicts the probability of each image to be an observation of each class.

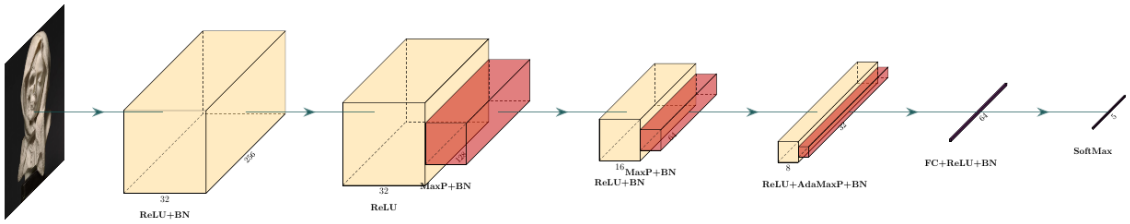

Figure 1: A simple representation of the proposed model.

The model is composed of 4 convolutional layers consisting of 3x3 kernels and a padding and stride of 1, each followed by max pooling (except for the first layer), and 2 fully connected layers. To each layer a batch normalization is applied and the chosen activation function is the ReLu, except for the last layer, to which a Softmax activation function is applied, as we are building a classifier.
In particular: The first convolutional layer takes as input the 3x32x32 images and has 256x32x32 out channels. The second convolutional layer has 128x32x32 out channels. A max pooling (128x16x16) is applied. The third convolutional layer has 64x16x16 out channels. A max pooling (64x8x8) is applied. The fourth convolutional layer has 32x8x8 out channels. An adaptive max pooling (32x2x2) is applied. All of the convolutional layers are made of 3x3 kernels with padding and stride of 1.
The output of the convolutional layers is then flattened to a tensor of dimension 128 and fed to the fully connected layers: The first fully connected layer has an input size of 128 and output of 64. The second fully connected layer is the output layer. It has an input size of 64 and output of 5, where each neuron represents one of the 5 classes to be predicted.
The network has a total of 404,263 trainable parameters.

2. Dataset

The dataset used in this work is a collection of five different types of artworks (drawings, engravings, iconographies, paintings, and sculptures) downloaded from google images, yandex images and this site.

The dataset is downloadable from Kaggle at this link The main folder contains 5 subfolders, one for each class, and each class contains a different number of images of different sizes. The total number of images is 8685, 108 of which were corrupted files that could not be read, and consequently were discarded; the cleaned version of the dataset can be downloaded from this link.
The final total number of images is 8577. The number of images per class is: Drawing: 1229, Engraving: 841, Iconography: 2308, Painting: 2270, Sculpture: 1929. As it can be observed, the dataset is fairly balanced, except for the category engraving that has a lower number of images. Particular attention will be given to the results of the models regarding this class in order to evaluate if some balancing is needed. The distribution of the aspect ratio of the images in the dataset is lightly bimodal, although most of the images are in the aspect ratio range of (0.5 - 1.6).
As a well performing classifier has to be invariant to a wide variety of transformations, before being fed to the network, the images go through a Data Augmentation process that consists in synthesizing plausible transformations. In particular the following transformations are made:
ColorJitter: Randomly changes the brightness, contrast, saturation and hue of an image uniformly in the range of the value passed as argument (0.2 for brightness, contrast and saturation, 0.01 for hue).
Resize: Resizes the input images to the same size of 32x32. This allows a consistent size of input and will speed the training process.
RandomHorizontalFlip: Horizontally flips the given image randomly with a given probability of 0.45.
RandomHorizontalFlip: Vertically flips the given image randomly with a given probability of 0.45.
Normalize: Normalizes a tensor image with mean 0.5 and standard deviation 0.5, which means that all pixels values will be in the range [-1,1]. This will make convergence faster while training the network.
The dataset was also divided in three subsets following a 60-20-20 splitting rule: Training Set: 60%, will be used to train the model; Validation Set: 20%, will be used to fine-tune the hyperparameters; Test Set: 20%, will be used once for each model and to compare the models in order to choose the best final one.

3. Training procedure

The model was trained for 44 epochs on Google Colab with a GPU accelerated runtime. The batch size was set to 128, the chosen optimizer was Adam with AMSGrad (which uses the maximum of past squared gradients rather than the exponential average to update the weights), and a starting learning rate of 0.0008. The chosen loss is the classical Cross Entropy Loss, which is the most used loss in classification problems:

$$J(\theta) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{M} {y_i}^{(n)} \log \widehat{y_i}^{(n)} \tag{1}$$

Where M is the number of classes, N is the number of observations, y_i the true class, and $\widehat{y_i}$ the predicted class of observation i.

This combination of hyperparameters was chosen after a large number of experiments that proved that this was the best performing setup.

4. Experimental Results

Several experiments were conducted to test the effectiveness of the proposed architecture.

An ablation study studies the performance of a neural network by removing certain components, to understand the contribution of the component to the overall model. As the chosen model holds 4 convolutional layers plus 1 fully connected and one classification layer, the performance of the following models is evaluated:
1st Conv layer + FC + Classifier.
1st + 2nd Conv layers + FC + Classifier.
1st + 2nd + 3rd Conv layers + FC + Classifier.
The ablation study shows that each decrease in the number of convolutional layers, leads to a noticeable decrease in performance.
In particular:
Removing the last convolutional layers shows a drop of around 1.5% in the test accuracy.
Removing the last two convolutional layers shows a drop of around 0.5% in the test accuracy
Removing the last three convolutional layers shows a drop of around 6% in the test accuracy.

Table 1 shows test results for the proposed architecture, as well as of the ablation study.

Model	**Validation**	**Test**
4 Conv Layers + 1 FC Layer	85.40%	**84.36**%
3 Conv Layers + 1 FC Layer	85.46%	82.90%
2 Conv Layers + 1 FC Layer	84.86%	83.83%
1 Conv Layers + 1 FC Layer	80.89%	78.47%

Table 1: Accuracy for validation and test datasets during the ablation studies. Bold font indicates the best accuracy and the chosen model.

It is also interesting to analyze the confusion matrix of the chosen model in Figure 2. As it is apparent from the matrix, the most correctly classified images were those of iconographies (92.6%), paintings (92.2%) and sculptures (85.7%). Drawings was the category that had the lowest percentage of correct predictions (63%), with misclassifications almost uniformally distributed across the other classes even though 12% of observations were incorrectly classified as engravings and 10% as sculptures. Engravings was the category that had the least observations (841 vs Drawing: 1229, Iconography: 2308, Painting: 2270, Sculpture: 1929). However, 70% of observations were correctly classified and almost all of the missclassifications (23.6%) were incorrectly classified as drawings. From these observations we can conclude that the category of drawings is the most challenging for the model.

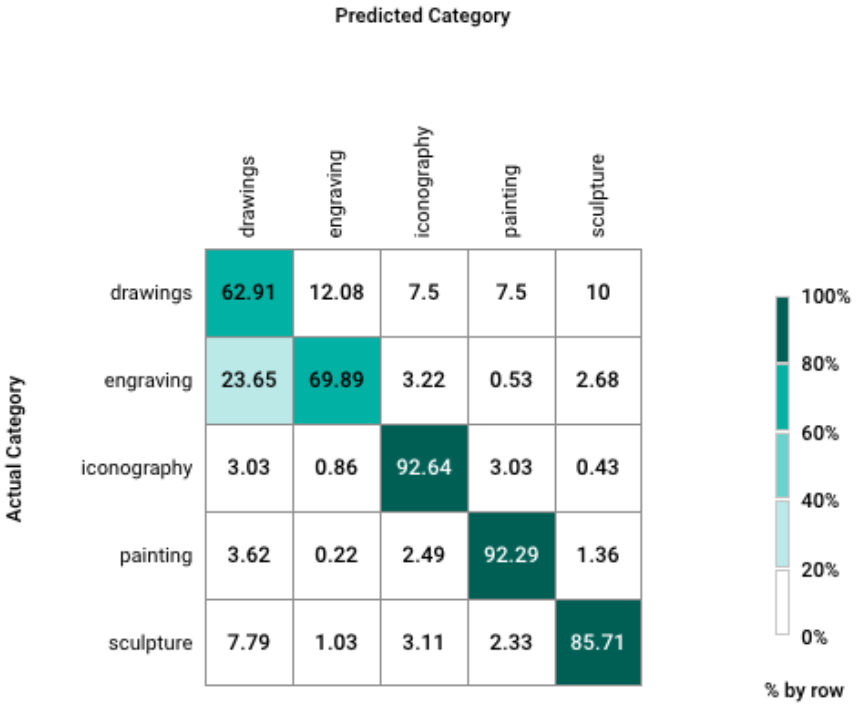

Figure 2: Confusion matrix (output from comet.ml)