MAIS 202 — Project Deliverable 1

Diego Lopez

February 21, 2019

1 Problem Statement

We wish to train a classification network on 90 classes. Then, for each image in a small handpicked set of famous paintings, we train an image transformation network which transfers the style of the famous painting to an arbitrary image.

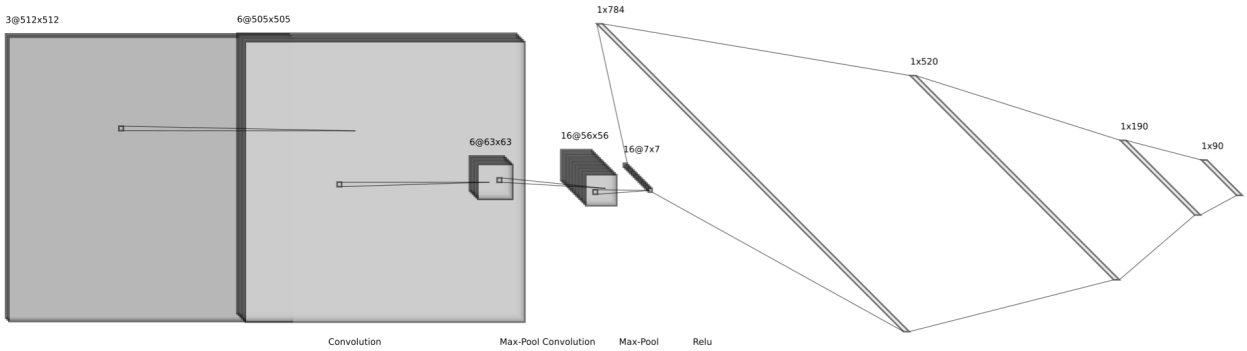
2 Data Preprocessing

We confirm that we are working with the Microsoft COCO dataset [1]. We will use the (labelled) 2014 train, validation and test images to train the classification network. Then, we will use the (unlabelled) images to train the image transformation networks. We proprocess the images by first rescaling it so that the smallest side of the image is 512 pixels wide, and then we center crop so as to keep the 512 by 512 pixels at the center of each image. We make sure the images are square so that the Gram matrices are easier to compute and so that training and inference is less computationally intensive. The 2014 train, validation and test datasets have 83 thousand, 41 thousand and 41 thousand images respectively. There are 90 categories. A single image may belong to more than one category (especially since 10 of the categories are supercategories of other categories).

3 Machine Learning Model

3.1 Tools

We use PyTorch thanks to its ease of use and its preprocessing capabilities. The architecture graph is as follows



3.2 Dataset Splits

We use the same splits that the Microsoft COCO dataset already has so that we can directly compare results with previously existing models.

3.2.1 Validation Methods

We used the validation set to test if the model was capable of overfitting on a single batch to ensure the model is working correctly.

4 Preliminary Results

Since the COCO dataset is a multi label dataset, it is not possible to consder the confusion matrix. We compute the accuracy of a model on a subset of the data by counting how many labels it predicted correctly (the predicted label is 0 if the network's output for that label is less than 0.5 and 1 otherwise). We were able to get an accuracy of 97% on a single batch, which means that the model is capable of memorizing. However, this measure of accuracy is misleading for multi label classification problems, since most images do not contain most labels, i.e. one can get a rather high accuracy by simply always predicting 0. For now, it seems like training on the whole dataset will take a long time, since the preprocessing already takes a while.

5 Next Steps

We wish to preprocess the whole labelled dataset and save the dataloaders as `.pt` files. Then, we train the neural network on the training set and stopping if the validation loss does not significantly change after some time. In addition, as a proof of concept we try the optimization method to do style transfer. Finally, we preprocess the unlabelled dataset and train the image transformation network to learn to transfer style.

References

[1] Microsoft. Coco dataset. http://cocodataset.org/#home.