CSL 603: Machine Learning
Course Project
Painter by Numbers

Pratham Gupta
2015CSB1024

Rajat Sharma
2015CSB1026

Abstract

Nowadays, the artworks of masters are sold for millions of dollars, such as an original Picasso carrying a 106 million dollar price tag. Accurately distinguishing the artwork of a master from a forgery can mean a difference in millions of dollars as a small mistake may lead to a huge loss, hence this field requires very high accuracy. Through this project we aim to develop a model which can identify the authenticity of a painting. Our model predicts the painter of a given painting from our database of famous painters. This can help in the identification of original and forged paintings. This can also help us predict the painters of some unknown masterpieces and helping to judge their true values.

Introduction

In our model we develop a convolutional neural network which is trained on our training images, which consist of the paintings of 5-10 painters and predicts whether an unseen painting belongs to them. Many different approaches have been tried and described for the purpose, so as to determine the most accurate model for the process. The training has been done on random patches extracted from different paintings of some painters, and a new patch from these paintings has been used for validation and testing purposes.

Related Works

Not much work has been done in this field in the past and most of the methods for detecting painters and identifying forgeries are manual till today. This project was presented as a competition on kaggle wherein the participants were asked to develop a program to examine pairs of paintings and determine if they are by the same artist.
The most significant contribution was made by the 1st Place Winner of the competition Nejc Ileni, whose model, which was a combination of unsupervised and supervised learning methods achieved a final accuracy 0.9289. His model generally predicts greater similarity among authentic works of art by Johannes Vermeer compared to imitations by the fraudulent artist, Han van Meegeren, whose works were provided by the competition.

Methodology

Data Pre-processing

Initially around ten famous painters have been chosen and around 400 paintings by each painter are taken. The images are of different sizes so the first pre-processing step was to resize each images smallest dimension to 256 pixels (retaining the aspect ratio) and then cropping it at the center of the larger dimension, obtaining 256x256 images. This gives all the images of equal dimensions. After this the resulting images are resized to 64x64 size. From these images 10 random patches are generated, each of size 20x20. 9 out of these 10 patches are used for training and 1 is used for validation. This gives a total of 35,757 training images and 3973 validation images for the 10 painters. More random patches have been generated for testing purposes.

Predictive Models

Convolutional Neural Networks were primarily used for modeling this problem. Convolutional Neural Networks are a special class of neural networks designed specifically for images. They primarily exploit the localized nature of the image wherein most of the relevant information is present in local neighboring regions. So, convolutional neural networks use convolutional filters for learning much lesser number of weights (as compared to a fully connected layer) and so, allow the training to be done on as large amounts of data as contained in the images.

Convolutional neural networks consist of convolutional layers where convolutional filters are applied onto the image and these weights are learnt. Relu activation function is used for these layers since the image pixels are positive and this does not diminish the gradients. This is generally followed by maxpool layers which shrinks the information down by resizing the image onto a smaller size. After the convolutional and the maxpool layers, the output is passed onto fully connected layers which do the final information extraction and produce the final output.

For the problem at hand, different convolutional neural network models were tried. Some of them include
-

Normal CNN The primary model used is a CNN with 2 convolutional layers which takes in a colored 20 x 20 image as an input. The first convolutional layer has 16 filters of size 3 x 3, while the second convolutional layer has 32 filters of size 3 x 3. The padding remains the same, i.e. the image is padded with black pixels so that the size of the convolved image remains the same as the input image. Relu was used as an activation function for the convolution layers. Since, the output of these convolutional layers is very large to process, a max-pooling layer with stride (2, 2) was used to reduce the information size by 4.

After the maxpool layer, the information is passed

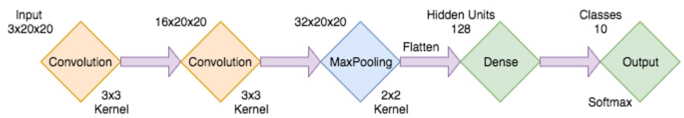

Figure 1: Network Architecture

onto a fully connected layer with 128 sigmoid neurons followed by a 10 class softmax neuron output.

Hierarchical CNN The main idea with this approach was that the model might find learning differentiating criteria among the 10 classes too hard to learn in one go and so, may only be able to learn how to differentiate between extreme classes but, might get confused on some very similar classes.

So, the 10 classes were grouped in groups of 2 to

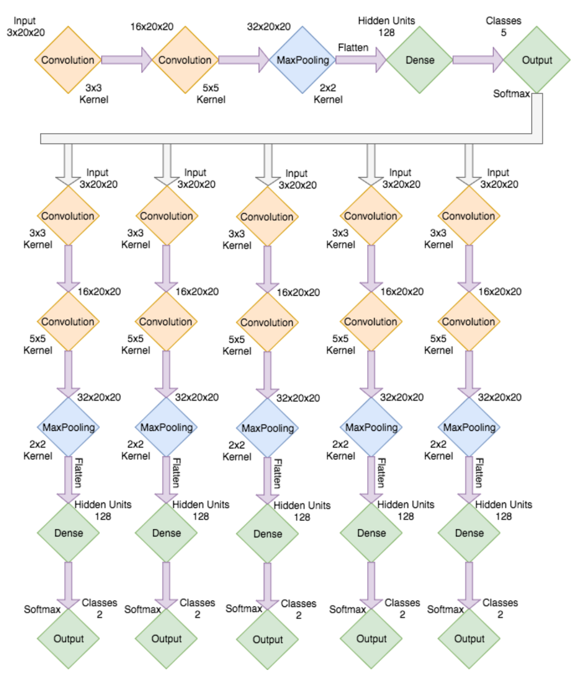

Figure 2: Network Architecture

form 5 such groups randomly. A similar model (as described above) was used to learn to classify between the 2 groups $\mathbf{S_i}$ (so, these models were trained only on images belonging to these 2 classes). The filter size was increased to 5 x 5 so as to learn slightly higher level features by considering larger neighborhood size.

For differentiating which group does the input image belongs to, another similar model $\mathbf{MM}$ was trained with a 5 class split so as to detect which classifier should have the highest weightage for the final classification. So, for the test image was classified by taking the argmax of the scores as -

$$(I,\ J) = \mathbf{argmax}_{i\ \in\{1,\ 2,..,5\},j\ \in\{1,\ 2\}}(mm(i)\ *\ s_i(j)) \tag{1}$$

$$Final\ Class = 2*(I\ -\ 1)\ +\ J \tag{2}$$

where mm(i) is the softmax score by model $\mathbf{MM}$ for i^{th} class and
s_i(j) is the softmax score of the S_i model for its local class j.

Results and Discussion

The accuracy obtained for the normal CNN model for 10 classes was **46.11125%**
The accuracy obtained for the hierarchical CNN model for 10 classes was **46.91669%**.

So, the hierarchical model performed just as good as the normal CNN. The normal CNN was also able to learn the differentiating features accurately. The hierarchical model performed very well on 2 class classifiers, but the 5 class classifier didn't perform as well. This probably is because the grouping of the artists was done randomly. This might've resulted in artists having very different paintings and painting styles being grouped together thus confusing the model. Due to the lack of a similarity measure amongst artists, the grouping had to be done randomly. The S_i models individually, were able to learn to classify between 2 classes very efficiently with testing accuracies of around 85% to 95%.

Summary

In short, we can summarize that our idea still has a lot of shortcomings but it can become an important part of the art industry if appropriate work and resources are invested in it. Although our model does not provide a very high accuracy but it can work as a base for more complex models which require high computational powers and give better results. From all the approaches tried for prediction some of them have much scope for improvements provided better machines for fast computations.

Future Work

Due to lack of computational power, the amount of data that could be used for training was limited by a lot and we aim to train the model on the complete data with GPU. This will help us perform a more in depth testing of our model with larger number of painters. We will also be able to incorporate many more features by increasing the complexity of the model so as to improve the overall accuracy obtained in matching the painters. We also aim to predict the confidence value of a painting being an original or a forgery. We would also like to try other approaches such as siamese neural networks which examine several images simultaneously. Also we can further augment the training data to improve accuracy by making small adjustments to the colour balance in the images, converting the images to grayscale, flipping the images left-to-right,etc.

References

[1] Painter by Numbers Competition on Kaggle, 1st Place Winner *Nejc Ileni*
http://blog.kaggle.com/2016/11/17/painter-by-numbers-competition-1st-place-winners-interview-nejc-ilenic/

[2] Dataset from Kaggle
https://www.kaggle.com/c/painter-by-numbers