Classifying Artwork of the Top 50 Influential Artists of All Time

Clarissa Bernardo
Loyola Marymount University
1 Loyola Marymount University Dr.
Los Angeles, CA 90045
`cbernar9@lion.lmu.edu`

Michael Langmyr
Loyola Marymount University
1 Loyola Marymount University Dr.
Los Angeles, CA 90045
`mlangmay@lion.lmu.edu`

Abstract

We classified artwork of the top 50 influential artists of all time. We ultimately achieved successful image classification through exploring various convolutional neural networks (CNN) along with its layers and fully connected networks. In addition, we performed numerous trials of image augmentation which increased our data set and improved our metrics. Lastly, we examined optimization tuning and its iterative effectiveness. Our best classification performance is 80% accuracy, which was achieved with experimenting with image augmentation and the deep learning network, ResNet34.

1 Introduction

Image classification is ubiquitous in our everyday lives. Its applications are used in many areas. Examples of this are medical imaging in which machine learning can predict if a patient has a medical condition by recognizing the shape of cancer cells. Another example of an application is facial and object recognition. Technology in privacy and security are elevated when a computer requires facial or fingerprint recognition to access their account. Last example, is visual search. When a user takes a picture with their smart phone of an object they are interested in, machine learning can provide more information about that object and even suggest similar objects that a user might be interested in. Image classification is powerful and can be used for good in society however, the technology can also be misused, abused, and unfortunately, can deceive society. An example of this are deep fake videos or audios. An instance of this is taking existing video content of a subject and replacing it with another subject's likeness. This unethical and fraudulent act could inflict harm to the targeted subject and spread misinformation.

This paper is organized as follows: Section 2 is about the data set. Section 3 is a deep dive into the networks we explored along with analysis and results. Section 4 provides more detail about image augmentation. Section 5 is Conclusion and Future Work.

2 Data

We obtained our data set through Kaggle which is a website that provides data sets of various topics. As users of Kaggle, we are able to explore and find different data sets that are formatted specifically for the data science environment to build machine learning models around them. Our data set included over 8,000+ normal and resized images that belonged to one of the fifty iconic artists. The artists represent our classes. The data set did not provide a validation set so we performed a data split of 90% and 10% to test our data. The number of images that belonged to each artist were not evenly distributed amongst all the artists. In addition, we were provided a .csv file that contained columns of information such as: art genre, artist nationality, artist timeline, brief biography of the artist, and a link to their Wikipedia page. We did not find the columns to be useful features for our data but instead provide supplemental material after classifying the artist. Resources can be found in Electronically-available resources.

2.1 Electronically-available resources

The data set was provided by Kaggle (`http://kaggle.com`). The data set 'Best Artworks of All Time - Collection of Paintings of the 50 most Influential Artist of All Time' is available at `https://www.kaggle.com/ikarus777/best-artworks-of-all-time`.

Table 1: Accuracy Scores of Custom Neural Networks

Batch Size	Num. Of Epochs	Learning Rate	Momentum	Layers - Conv. / Pool / FC	Accuracy
64	2	.001	.9	2/1/1	16.00%
64	5	.001	.9	2/1/1	22.50%
64	5	.01	.9	2/1/1	29.30%
64	5	.1	.9	2/1/1	8.66%
64	5	.01	.5	2/1/1	23.70%
64	5	.01	.9	2/1/2	27.33%
64	5	.01	.9	2/1/3	22.78%

3 Networks and Analysis

3.1 Custom Neural Network

Before deep-diving into sophisticated model architectures like AlexNet, VGG, or ResNet we wanted to recognize the functionality of convolutional layers and fully-connected layers and their important role through each iteration. Through numerous trial runs we developed a deeper understanding of the optimization parameters such as the learning rate and momentum. We also found that batch and epoch size contribute to the time complexity of our models. All these variables contribute to the increase or decrease of our scores. We achieved a low 16% accuracy score and a high 29.30% accuracy score after tuning the parameters; We had a flawed model. We also learned that simply duplicating a model that performed well on a similar goal does not necessarily mean the same model would have great performance on our goal. We quickly learned that some architectures perform better than others. Failing at this step was a necessity in that we have learned a significant amount during this process in order to take the next pivotal steps toward deep learning to improve our metrics.

3.2 VGG16 and AlexNet

VGG16 : State of the art in image classifier that is currently used today. We set up our parameters using a batch size of 65 and an epoch size of 5. We experienced extremely long run times. In this instance, VGG16 took 7 and a half hours to run. Upon finishing, we ran into issues on our compiler, Google Colab. Our two main issues are the compiler timing out and running out of memory. We realized that Google Colab had limitations. We needed to work around them without compromising a high score. Our next strategy was to lower the epoch size from 5 to 3. The compiler took more than 4 hours to complete and calculated an accuracy score of 41%. We found that to be lower than expected. VGG16 is a popular and established convolutional neural network. Having that knowledge prompted us to re-evaluate the validity of our data set. We discovered that the unevenly distributed data among the artists were affecting the accuracy score. The problem with an undistributed data set is that some artists with a higher number of art pieces in the data set could be trained on more frequently than other artists which lead to high accuracy scores on those artists versus those that had fewer art pieces in the data set.

AlexNet : Another deep learning model we tried. This model is not as current as ResNet or VGG, but during its time in 2012, it was a top performing image classifier. We also knew that AlexNet has less layers than VGG and ResNet. Given our time constraint, we decided to test if we could obtain similar scores of a VGG model but with shorter compile times. AlexNet took about 40 minutes to compile and had a 37% accuracy score.

3.3 ResNet18 and ResNet34

The ResNet models calculated the best results for classifying the artist. We first experimented with ResNet18. We used a batch size of 64 and an epoch size of 5. We also re-sized the images to 255x255 and randomly cropped them. The accuracy score of this trial was 67%. Observing an increased score prompted us to augment the images. We continued with the ResNet models because it did not exceed Google Colab's limitations. For the second trial, we used a batch size of 32 and an epoch size of 5. We resize the images to 750x750 and randomly cropped them. The accuracy score for the second trial was 75%. Lastly, we wanted to test if ResNet34, which has more layers, could compile without straining Google Colab. We pre-

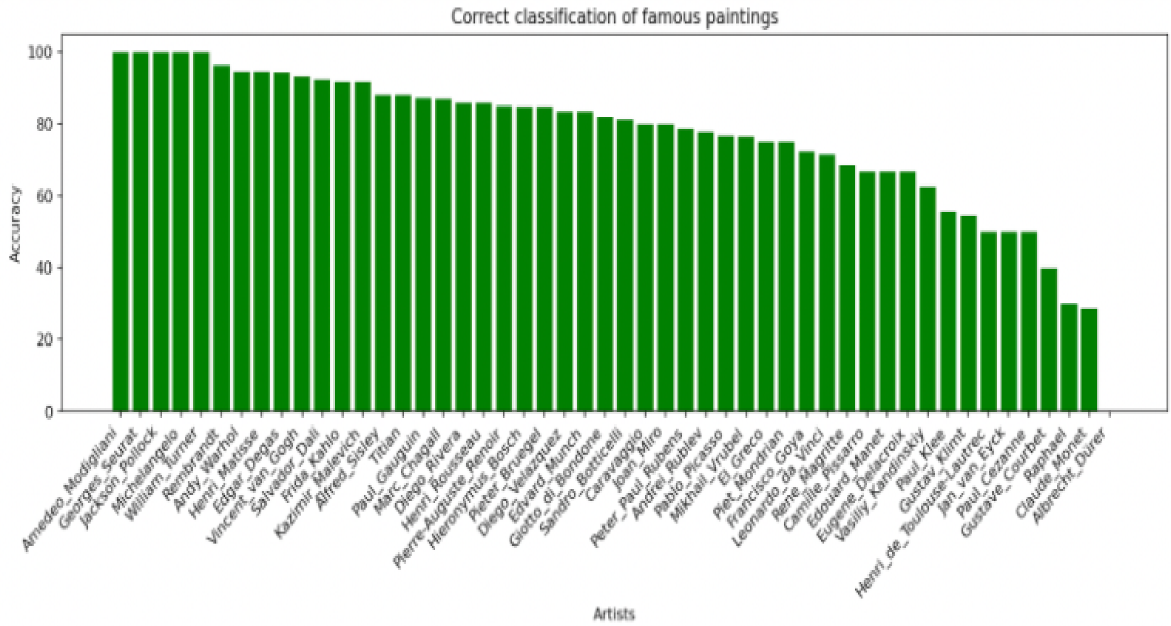

Figure 1: Accuracy Scores Across All Artists

served the same parameters as the second trial but added color jittering to the images. This model produced our highest score: 80% accuracy.

4 Image Augmentation

The performance of neural networks often improves with an increasing amount of data. Data augmentation is a method to artificially increase the amount of training data by making slight changes to the existing data. Those changes can include rotations, changes of the color saturation, cropping and others. The idea is to create new plausible examples of images to train the neural network. We had over 8,000+ images which seems sufficient to train a network to a satisfactory accuracy but we also had 50 classes which gave us an average of 160 training images per class. Furthermore, the number of paintings belonging to each class were unevenly distributed. For example, one class contained 877 images while another class contained 24 images. Therefore, increasing the amount of training data seemed vital to increase the accuracy. The image augmentation methods we used were rotations, color jitter, gaussian noise and random crops. While rotations and random crops are self explanatory, color jitter is a function that changes the brightness, saturation and hue of each image. Gaussian noise is a function that adds noise to the image which is distributed according to a normal distribution. By optimizing and augmenting our image data, we managed to increase the accuracy from around 50% to 80%.

5 Conclusion and Future Work

After all the optimization, we reached an accuracy of overall 80%. We narrowed our scope to analyze the individual artist score. While for some of the artists we obtained a 100% accuracy, we had one artist, Albrecht Durer, with 0% accuracy. Please see Figure 1: Accuracy Scores Across All Artists. If we would have had more time and resources we could have increased the number of layers in the CNN. Limited internal memory was a bottleneck and dictated the type of CNN and the number of layers we could run. As mentioned above, ResNet34 was our best performing model within our hardware constraints. Furthermore, larger image sizes could have been used. Out of performance concerns, we did resize the images to 750x750 pixels. With large size images, the neural network would have in all likelihood produced more accurate classifications. The last possible improvement is obtaining additional training data. We are convinced that if our data set was twice the size with the same amount of classes we could have achieved better results.

Acknowledgments

Dr. Mandy Korpusik and the Keck Computer Lab
at Loyola Marymount University