Predicting the artists of paintings using neural networks

Finn Tobin, Jas Dhaul, Puwentao Yan
University of Victoria
3800 Finnerty Road

Abstract

This report outlines how neural networks could be used to predict the artist and style of paintings. The dataset for this project was obtained from Kaggle and consisted of over 100,000 records [1]. The data was trimmed to roughly 5000 entries for the artist dataset and 3900 entries for the styles dataset. The images from this dataset were augmented and downsized for runtime optimization. After the image pre-processing was completed, the data was trained on a convolutional neural network (CNN) and a more sophisticated residual neural network (ResNet). The results revealed that neural networks are capable of predicting such features. However, to produce highly accurate predictions, the models must be fine-tuned, and the dataset needs to be large, diverse and optimized for the job.

1 Introduction

1.1 Artwork

Identifying the features of artwork through machine learning can be highly beneficial for many reasons. Firstly, identifying artwork is already a difficult job, even for professionals. There are meticulous details experts need to investigate when assessing the legitimacy and value of any artwork. These details can include brushstroke thickness, texture, impasto, and signatures [2]. By utilizing modern-day deep learning techniques, historians could have another tool to help them make discoveries in their field, determine the authenticity of artwork and assess the valuation of artwork.

1.2 Problem Definition

The goal of the paper is to show that it is possible to utilize neural networks to predict the artist and the style of artwork. This is something that has been done before using deep learning models such as CNNs as well as newer neural network architectures such as residual neural networks (ResNet) [3]. This paper aims to utilize similar strategies to produce accurate predictions for the two features.

In his paper, Viswanathan reports a 0.437 accuracy for artist predictions using a baseline CNN model and a 0.511 accuracy using a ResNet model [3]. The dataset used to train this model contained over 13,000 paintings from over 57 artists. Our goal for the project was to expand upon this work and see if it would be possible to make accurate predictions for the style of painting as well.

1.3 Dataset

The dataset used for this project can be retrieved from Kaggle [1]. The original dataset contains over 124,000 rows with features including the style of art, the artist name, the approximate date of origin and a URL to the artwork on WikiArt, an online repository containing millions of images of artwork [4]. Initially, the plan was to clean up the data since some artist names were unavailable and were replaced with a location description instead. After these rows were removed from the dataset, there were still ample amounts of data to process.

Due to computational constraints, the data had to be resized even further. By sorting the dataset by the number of paintings for each artist, we analyzed the top 10 artists, as shown in Table 1.

Table 1: Top 10 artists sorted by the number of paintings

Artist Name	Number of Paintings
Giovanni Battista Piranesi	1126
Vincent van Gogh	917
Pablo Picasso	812
Albrecht Durer	797
Marc Chagall	713
Salvador Dali	678
Claude Monet	670
Rembrandt	579
Alfred Freddy Krupa	528
Henri Matisse	498

We see that numbers are quite distributed. To maintain consistency among all artists, 450 paintings were used in the training, validation, and testing sets with an 80-10-10 split respectively. This distribution entails approximately 4500 images in total.

For predicting the style of painting, we filtered the 55 different style categories covered by these artists to only the styles with at least 50 paintings. Otherwise, there could be too few paintings per category for our 80:10:10 split. In total, we had 3972 paintings for this dataset, with 16 unique categories.

Going into our study, we anticipated higher accuracy for learning styles over artists. This is because the most prolific artists have some overlap in time periods, likely influencing each other's works. The styles should have a bit more distinction, given their different classifications. That said, when cleaning the data, we did select the first style label where there were multiple entries for the painting. Therefore, paintings falling under multiple styles may impact accuracy.

2 Approach

Neural networks are powerful tools for image classification problems. However, their accuracy depends on several variables ranging from the complexity of the input data to the minute details in the architecture of the network. To gauge the effectiveness of neural networks, we train a CNN and a ResNet to solve the problem of identifying artist and style features.

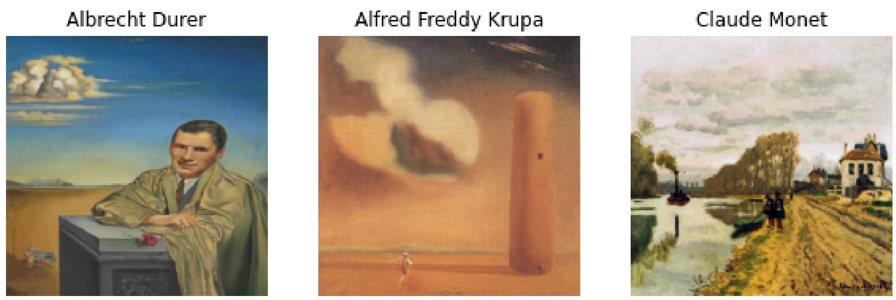
2.1 Data Processing and Augmentation

The images retrieved from WikiArt come in various sizes. This presents a problem for neural networks, as they expect a fixed-size input. Choosing a proper image size for this problem is a challenge in and of itself. If images are downsized too much, unique details in brushstrokes or textures that help the network make the ultimate prediction could be lost. On the other hand, if the image resolution is too high, it will create a bottleneck when training the data. When solving such problems, compromises are made. In his paper, Viswanathan took a center crop of images of size

224x224 pixels [3]. In our case, we resized the images down to 100x100 pixels and trained the network on the entire image, hypothesizing that the network would perform better when given access to the entirety of the image.

Preliminary testing revealed two issues with the models. Firstly, the images took a very long time to process since they were directly loaded into memory by sending web requests to the associated URL for each image. We improved our speed by downloading the images directly to disk and using TensorFlow to process the images [5]. The second problem was that our models were overfitting the image data. We observed very high training accuracy but abysmal validation and testing accuracy. To alleviate this issue in the early stages, we introduced Data Augmentation, a technique that introduces random mutations to the data, further diversifying the training set [6]. In particular, we introduced a random horizontal flip to our data with a 50% probability.

Figure 1: Sample images from the dataset



2.2 Convolution Neural Network

CNNs are one of the most popular tools for image classification problems. The model we trained our data on consisted of various layers and techniques to further combat the effects of overfitting the data. Table 2 displays the layers of the model in detail.

Table 2: CNN model layers

Layer Type	Output Shape
Conv2D	(100, 100, 3)
Max Pooling 2D	(50, 50, 64)
Conv2D	(50, 50, 32)
Max Pooling 2D	(25, 25, 32)
Flatten	(20000)
Dense	(128)
Dropout	(128)
Dense	(10)

Observe the addition of the Dropout layer and the inclusion of L2 regularization in the Dense layer. These techniques are helpful strategies to reduce overfitting the image data [7]. In addition to this, we activate the padding option for the two convolution layers, as they help fill the perimeter of the image [8]. We observed improved performance with this option enabled. Moreover, the first convolution layer and the second convolution layer include filter sizes of 64 and 32, respectively and a kernel size of 3. The numbers for these layers were derived from experimentation with the model and ended up yielding the best results. Finally, we include two max pooling layers after the two convolution layers so that the model focuses on features in the artwork that are relevant to making the predictions.

2.3 ResNet-50 Neural Network

ResNet-50 is a pre-trained model available through Keras that was specifically designed for image recognition [9]. It is a 50-layer CNN (48 convolutional layers, one MaxPool, and one average pool layer) that is based on the VGG neural networks but has fewer filters and is less complex [9]. When exploring different algorithms for image and painting recognition, we attempted a VGG16 and an Inception model, both of which performed poorly (0.3 and 0.1 accuracies respectively) and were very slow. This led us to try ResNet-50, which was much faster and performed better.

3 Results

Early prediction attempts on our models did not yield desirable accuracy results. However, after some tuning, the models started to perform better after 10 epochs. Here we reveal artist and style prediction results for the CNN and ResNet models.

3.1 Artist Prediction Results

The regular CNN model achieved a training accuracy of 59%, and a validation and test accuracy of 53%. For the most part, this model was stable and did not have any overfitting issues after the regularization and dropout layers were added. Moreover, during training, we observed a steady decrease in training and validation loss, which was also a good indication that the model was not overfitting the data. On the other hand, the ResNet model seemed to struggle in the early epochs. During these epochs, we observed the model had high training accuracy and low validation accuracy. By the final iteration of training, we observed a training accuracy of 96%, a validation accuracy of 71%, and a test accuracy of 74%. Clearly, the model was still overfitting. Figure 2 and Figure 3 show the training and validation scores during the training period for the two models.

Figure 2: Training graph for CNN artist prediction

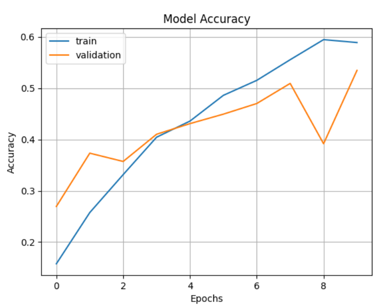
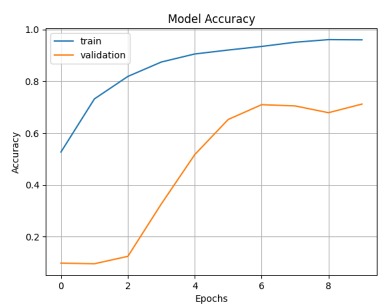


Figure 3: Training graph for ResNet artist prediction



3.2 Style Prediction Results

The models performed slightly worse for the style predictions in comparison to the artist prediction. The CNN model recorded a training accuracy of 57%, a validation accuracy of 45%, and a testing accuracy of 46%. Similar to the artist predictions, this model seemed to do a good job of not overfitting the data. The ResNet neural network had the same overfitting of issues with this style's dataset as it did with the artist's dataset. Here we observed a training accuracy of 96%, a validation accuracy of 62%, and a testing accuracy of 64%. Figures for these training models are shown below.

Figure 4: Training graph for CNN style prediction

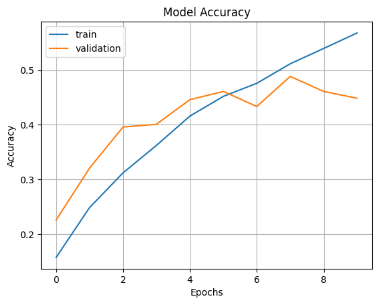
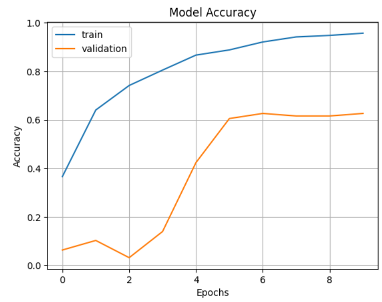


Figure 5: Training graph for ResNet syle prediction



3.3 Analysis

Some limitations of our study were runtime and image complexity. These are works of art, and as such are complex blends of colours and shapes. The file size for each was so significant that we simply had to downsize the images for runtime purposes. When compared to other standard tests, like MNIST or fashion-MNIST, the detail lost is more significant when downsizing paintings. This undoubtedly impacted how well our models could learn and predict the labels for said paintings. Likewise, many of the images used for the paintings could have been photos of the paintings, as opposed to scans. With our dataset, we did not have a way to filter out those paintings. As such, other details irrelevant to the paintings (i.e. frames, surroundings, etc.) could have been learned and obstructed the prediction accuracy. With these limitations in mind, our models still performed well. Certainly, some overfitting still occurs, but through all our hyperparameter tuning, the accuracies in the above section were the best we found.

Interestingly, our prediction that accuracy for styles would be higher than artists turned out to be false. Upon further analysis, our leading theory on why is the representation of data. Whereas for artists, we had a roughly equal amount of paintings, styles were disproportionate. Some had 450 training samples, while others had 60. So, although it is theoretically easier for the models to tell the difference between styles of paintings than artists working in the same time period, using the styles from the top 10 artists could have impacted our accuracy here.

4 Conclusion

In summary, we attempted to train a CNN and a ResNet to make artwork predictions. More precisely, we wanted to know how well neural networks could predict the style and the author of the paintings. We trained our models on images retrieved from a popular online artwork repository. After implementing image augmentation techniques and fine-tuning our models, our data showed that neural networks are capable of producing accurate predictions if tuned correctly. Problems such as overfitting the data were persistent throughout the project, especially with the more sophisticated ResNet model. In the future, we would like to see neural networks trained on a larger dataset with high-resolution images, resources permitting. Additionally, it would be interesting to work out the data so that there is a similar proportion of styles and artists. Although it may be difficult to do this from the same artists and styles, given that artists may favour one style disproportionately.

References

[1] Antoine Gruson. WikiArt: All images (120k+) kaggle [online]. https://www.kaggle.com/datasets/antoinegruson/-wikiart-all-images-120k-link?resource=download, Nov 2021.

[2] “How to research an artwork: Identification” berkeley library [online]. https://guides.lib.berkeley.edu/c.php?g=387200&p=2626559.

[3] Nitin Viswanathan. "Artist Identification with Convolutional Neural Networks" stanford vision and learning lab, [online]. http://vision.stanford.edu/teaching/cs231n/reports/2017/pdfs/406.pdf, 2017.

[4] Wikiart [online]. https://www.wikiart.org/.

[5] Tensorflow [online]. https://www.tensorflow.org/.

[6] "Data augmentation: Tensorflow Core" tensorflow, [online]. https://www.tensorflow.org/tutorials/images/data_augmentation.

[7] "Overfit and underfit: Tensorflow Core" tensorflow, [online]. https://www.tensorflow.org/tutorials/keras/overfit_and_underfit.

[8] "Convolutional neural networks" 7. convolutional neural networks - dive into deep learning, [online]. https://d2l.ai/chapter_convolutional-neural-networks/index.html.

[9] "ResNet-50: The Basics and a Quick Tutorial" datagen, [online]. https://datagen.tech/guides/computer-vision/resnet-50/.