Painter by Numbers: Generalization for Unseen Artist using Siamese Networks

Nels Numan (4615107) Jose Soengas (4235762) Navin Raj Prabhu (4764722) Jakub Pietrak (4347293)

Delft University of Technology

Problem Statement

In this project we have attempted to tackle a challenge proposed on Kaggle, namely *Painter by Numbers*[3]. The objective of this challenge is to distinguish, in **pairwise comparison, whether two paintings were created by the same artist**. A challenge here is **the ability of our model to efficiently extrapolate to paintings of unseen artists**. To tackle this, we used a special type of neural network called the *siamese Network*.

Siamese Network

A *siamese network* uses the same weights while working **in tandem** on two different input vectors to compute comparable output vectors. These output vectors are used in comparing the inputs in terms of a distance function between them.

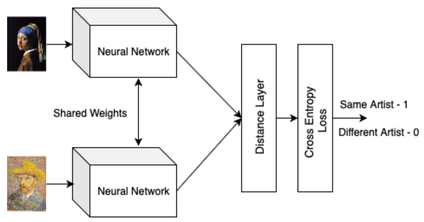

Figure 1. Siamese Neural Network Architecture

- ***Hypothesis 1***: As a baseline model, how well can the CNN-based siamese network trained on the paintings dataset perform well in terms of generalisation error?
- ***Hypothesis 2***: Will a siamese network trained on pre-trained features outperform the baseline CNN model?
- ***Hypothesis 3***: Can further fine-tuning of the pre-trained features and feature representations give a better generalisation for unseen artist?

Dataset

Contains **103,250 unique paintings** by about **2300 different artists**, of which **79,432** are in the *training set* and **23,818** in the *test set*.

Preprocessing

- All images were resized to 224×224 pixels, and were consequently represented by a feature vector of size 2048.
- To ensure balanced pairwise comparisons, triples were generated by taking a *reference image*, and respectively pairing this up with a *dissimilar image* and *similar image*.

Baseline CNN

The baseline **CNN based Siamese model**[2] in total had *13,351,872* trainable parameters. Being an end-to-end complex learning model, each epoch ran for approximately 12 hours.

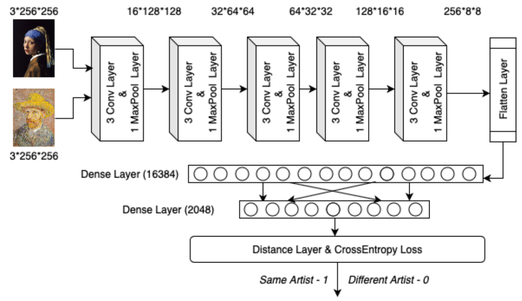

Figure 2. An illustration of our CNN based Siamese Network

Transfer Learning

- **InceptionV3** and **Xception** pre-trained networks were used initially used without fine-tuning. Both have *24,107,096* parameters.
- As an attempt to further improve our results, we **fine-tuned Xception** network by adding a dense layer for our 1584 training classes. Firstly the top layer and subsequently the top block, namely block 14 (separable convolution, batch normalization and activation). In both cases, there were respectively, *3,245,616* and *7,994,416* trainable parameters.

Concatenated feature vectors based on corners

To more closely capture the characteristics of each painting, we constructed a complex feature representation which **concatenates** the features of the original image of size 224×224 pixels with crops of each corner. This resulted in a final **feature vector size of 10240**, which dramatically **increased the complexity** of the model.

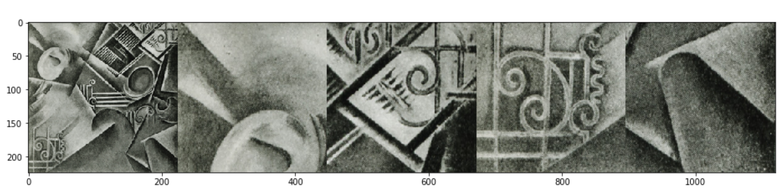

Figure 3. Concatenated images, each of size 224×224

Results and Discussion

The ***Xception*-based siamese network is the best performing model** across the three datasets. The performance gains of *Xception* over the baseline CNN and *InceptionV3* are not due to increased capacity but rather to a **more efficient use of model parameters** [1]. The results for the fine-tuned models are worse than simply using the pre-trained model. The accuracy of the corner feature approach was significantly lower than the baseline CNN.

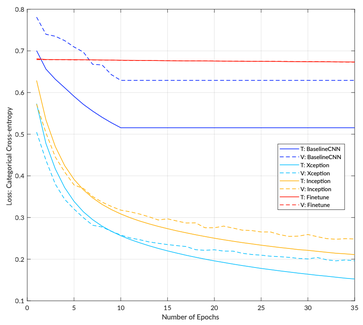

Figure 4. Learning curve for each approach

	Seen artists	Unseen artists	Mix of seen and unseen artists
Baseline CNN**	0.64	0.56	0.59
InceptionV3	0.821	0.608	0.612
Xception	**0.888**	**0.647**	**0.672**
Fine-tuned	0.73	0.53	0.56
Corner features	0.512	0.501	0.502

Table 1. Results - Accuracy across the three test datasets for each of the model explained earlier. All models were run with one Tesla P100 GPU (3584 NVIDIA CUDA Cores with 16 GB of RAM). ***10** epoch runs, while others used **35** epochs.*

Conclusion

- ***Hypothesis 1***: The baseline CNN model with ~ 13 million trainable parameters takes a huge amount of time for loss convergence. We also note the **overfitting of the model** on the training dataset (Figure 4). Though the loss of the model converges moderately in the 10 epochs, it fetches a **sub-optimal generalization accuracy**.
- ***Hypothesis 2***: Siamese network trained on pre-trained features is less complex and produces better results (Table 1) than the baseline model. We also see there is **no overfitting** in the learning process (Figure 4). While the model performs **considerably well in the test set of seen artists**, it **fails to extrapolate** efficiently to unseen artists.
- ***Hypothesis 3***: Fine-tuning the Xception model **does not improve the performance** of the model. Reducing the feature vector to 1584 may have contributed to this. The concatenation of **corner features made the model too complex**, and dimensionality reduction or a smaller representation should be considered.

References

[1] Francois Chollet. Xception: Deep learning with depthwise separable convolutions. *correarXiv: 1610.02357v3 [cs.CV] 4 Apr 2017*, 2017.

[2] Nejc Ilenic. `https://github.com/inejc/painters`.

[3] small yellow duck. https://www.kaggle.com/c/painter-by-numbers.