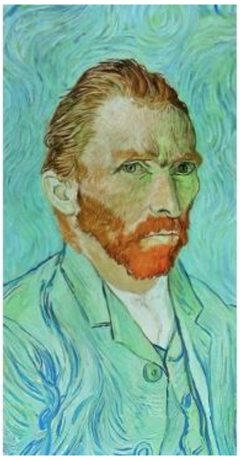




Does anyone actually “get” art? My computer does! Understanding Artistic Style Using Machine Learning

Assessing if two paintings are similar is as easy for humans as it is difficult for computers. Looking at the paintings above one can spot many similarities amongst all pairs of paintings: The left paintings are portraits, the left, and the right both black and blue. Nonetheless it is clear us humans that the centre and right are the most similar. In this Blogpost on the exciting intersection of art and machine learning I will elaborate how computers can be taught to understand artistic style and to say: ‘the two on the right are the most similar!”

Style in itself basically is the *way* an artist applies paint to canvas. It is determined by characteristics such as composition, colour form, stroke and many more. To help a computer understand what defines an artworks style we will find a mathematical expression for it by using machine learning.

How To Make A Computer “Understand” A Painting’s Style

Convolutional Neural Networks (CNNs) are machine learning systems that are trained to perform a specific task on an image. During the training process the network looks at thousands of (usually hand labelled) images. By looking at the images and their labels repeatedly the network learns to understand the visual patterns in the image that consistently correlate with the image’s label. After the initial training we can use it to extract the learned patterns from the painting.

A popular and freely available network is the object detection network “VGG-19”. It was trained on images from different objects such as cars, horses, flowers, and many others. In 2016, Gatys et al. proposed a method for style transfer, where the style of a painting is transferred to a photograph input. [1] To do so the correlations of the patterns detected by the object detection network are calculated. Not using the patterns relevant for object detection adds a level of abstraction. It allows us to adopt the patterns for the task of style extraction. The pattern correlation values are stored in a vector, which we will call a paintings *style vector*.

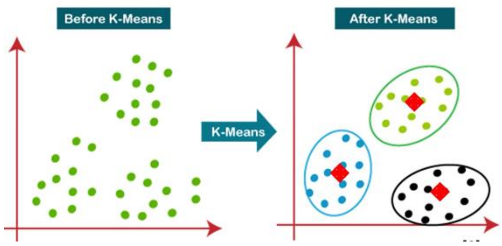
[5]

After recreating S. Gatys way of finding a style vector we tried to improve on the representation's quality. First, the network's "architecture" (the way it is built) was changed to the "ResNet-34" architecture, for it achieves the best results on detecting patterns in art. [2]
Second, the networks training was changed. Instead of objects we used ~300.000 paintings from the years 1300 – today to teach the network to predict both the paintings artist and stylistic epoch. The network now directly learns to recognize patterns correlating with a paintings style! To see if calculating the filter's correlations is still beneficial, we compared it to directly using the patterns activations. In this case each entry of the style vector represents the network's reaction to one of the patterns.

Finding Style Groups In A Collection Of Paintings

For we do not know what the vectors entries specifically correspond to we cannot make direct statements about the paintings style. However, we can compare the style-vectors and find similar paintings within the Dataset!

First, let us have a look at how well we can group a dataset of paintings by similarity. For doing so we use the K-Means algorithm. It works by dividing the set of style vectors into K (128) partitions, so that the average distance of a style vector to the cluster's centre is minimal.
Looking at some sample clusters we can see that all algorithms manage to cluster the paintings to a certain degree of similarity. However, there are some clusters where clustering did not work. [add example?]



Gram Object-VGG Gram Art-ResNet Embedding Art-ResNet

Measuring How Similar The Groups Are

Unfortunately, there currently is no accepted measurement of stylistic similarity (really!). To find a way to quantify our results we use the paintings epoch labels such as renaissance, post impressionism etc. as a proxy for style. Even though paintings from the same epoch can look vastly different paintings within an epoch do look stylistically similar most of the time. By looking at the number of paintings from an epoch that are in a cluster that is mainly filled with paintings from the same epoch we can gauge the algorithms quality.
Looking at the quality scores we notice that the art detection network performed better that the object detection network. Apparently, our improvements have worked!

Algorithm	**Quality Score**
Object Detection Network – Gram	0.21
Art Detection Network – Gram	0.40
Art Detection Network – Direct	0.38

Finding A Paintings Most Similar "Neighbour"

Let us have a look at how well the algorithms are suited to find a painting's most similar "style neighbour"! For doing so we compare the paintings style vectors using the cosine distance. It essentially measures how similar two vectors (and therefore the two painting's styles) are.

We can see that using the art detection network yields much better results and that directly using the patterns detected by the network gives results of similar quality! This can be explained for the gram matrix was applied to abstract the patterns relevant for object detection to use them for style representation. Now that we learned the patterns relevant for style detection directly, this step is no longer necessary!

To me personally the opportunity to bring together my passion for both art and machine learning was as unexpected as it was delightful for it brings together the quantitative and the creative. And this young interdisciplinary world has still much more to explore than what I could learn so far! For example, there is already machine learning involved at comparing paintings to detect fakes paintings [3]. Also, machine learning is used to discover connections overarching the historically defined art epochs! [4]. After working in this field for six months I am still as excited as on day one. I am looking forward to see the developments this exciting field will bring in the future!

[1] [easy: https://medium.com/tensorflow/neural-style-transfer-creating-art-with-deep-learning-using-tf-keras-and-eager-execution-7d541ac31398 Paper: https://openaccess.thecvf.com/content_cvpr_2016/html/Gatys_Image_Style_Transfer_CVPR_2016_paper.html]

[2] http://proceedings.mlr.press/v77/lecoutre17a.html

[3] https://www.technologyreview.com/2017/11/21/105077/this-ai-can-spot-art-forgeries-by-looking-at-one-brushstroke/

[4] https://medium.com/@ahmed_elgammal/the-shape-of-art-history-in-the-eyes-of-the-machine-6c9090257263

[5] https://medium.com/@build_it_for_fun/neural-style-transfer-with-swift-for-tensorflow-b8544105b854