CS 194-129 Final Project Proposal: Cloud MoNet

Team

Our team has four members, all enrolled in the 194-129 (undergraduate) version of the course.

Akhil Jalan - Junior, Math & CS Major
Phat Pham - Senior, CS Major
Albert Chen - Senior, CS Major
Noah Wu - Sophomore, CS Major

Problem Statement and Background

Our project has two main parts:

1. **Implementation of state-of-the-art style transfer network.** We will implement the unobserved neural style transfer network described in [1]. This is a group of deep convolutional networks which perform *neural style transfer* on images. That is, given a content image C and a style image S, the network creates a "stylized" version of C in the manner of S.

Unobserved style transfer [1] is its generalization to previously unseen styles. Previous versions of neural style transfer network would learn some set of styles, and then transfer a learned style to a new content image. However, in [1] the net could also transfer the style of a previously unseen style image S. The image below is from [1].

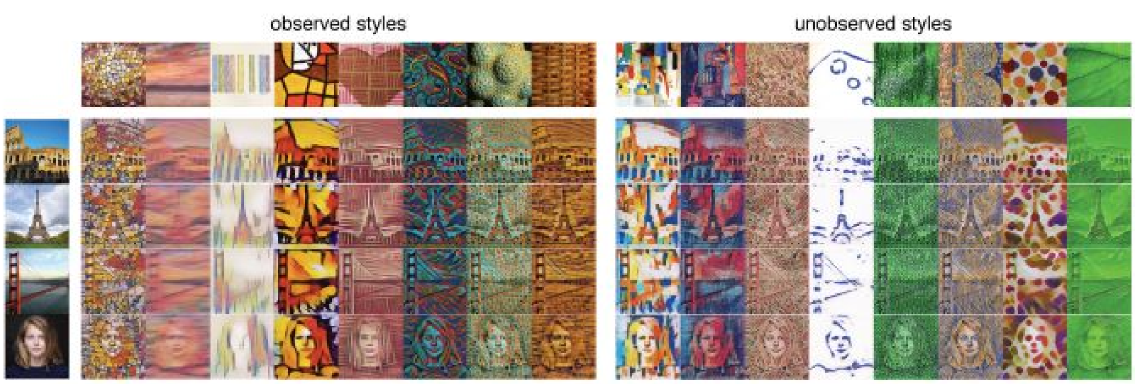

Figure 1: Stylizations produced by our network trained on a large corpus of paintings and textures. The left-most column shows four content images. Left: Stylizations from paintings in training set on paintings (4 left columns) and textures (4 right columns). Right: Stylizations from paintings never previously observed by our network.

2. **Prediction on the style embedding space.** In [1], the authors describe that "the style prediction network has learned a representation for artistic styles that is largely organized based on our perception of visual and semantic similarity without any explicit supervision." This

indicates that the *embedding space* of artistic styles has learned representations of styles in the process of network training. We intend to explore this embedding space in the following ways:

- **Classification**: After the model has been trained to perform style transfer, we can use *transfer learning* to take the already-trained initial layers of the network and use them for a new problem - classification of style. We hope that the learned representation of styles, which crucially can generalize well to unseen styles, leads to better representation for classification into a set of artistic categories ("abstract", "impressionist," etc).
- **Dimensionality Reduction**: In [1], the authors use t-SNE to visualize the embedding space in 2 dimensions, and they suggest that "the model might capture a local manifold from an individual artist painting style." We intend to use other dimensionality reduction methods such as principal components analysis (PCA) and multidimensional scaling (MDS) to test this hypothesis.
- **Clustering**: Within the artistic community, there is lively debate over the validity of categories commonly used to group artistic styles and periods together. By comparing the effectiveness of different clusterings through metrics such as the silhouette score, we may find novel groupings of art that suggest new categories. Of course, we should caution that the embedding space learns only visual textures, and is unable to interpret the "meaning" of a painting.

Data Source

For style images, we will use the *Kaggle Painters by Numbers* dataset. For content images, we will use the ImageNet dataset [3]. Depending on computational constraints, we will pick some subset of both that are small enough to be computationally tractable.

Both of these datasets are publicly available for download.

Descriptions of the Tools You Plan to Use

We will write code in Python, using Tensorflow and Keras as the main neural network libraries.

Evaluation

Because our problem statement came in two parts, we'll evaluate our model according to those two parts:

1. To evaluate the implementation of the style transfer network, we'll use the content loss and style loss metrics defined in [1]. Briefly, loss is a function of the content image C and style image S. If X is the stylized image returned by the style transfer network, then L_{c}(X, C) measures content loss and L_{s}(X, S) measures style loss. The objective function is a weighted linear combination of L_{c}(X, C) and L_{s}(X, S). See "Methods" in [1] for details.

2. To evaluate the effectiveness of our model of utilizing the embedded space to learn the artistic style, there'll be a set of arts generated from the first half of our problem statement. From there, we'll have our style transfer engine label each of those generated arts. Thus, we'll use a generated dataset from our first half of our problem statement to measure the accuracy of the prediction in the latter half.

References

1. Ghiasi, Golnaz, et al. "Exploring the structure of a real-time, arbitrary neural artistic stylization network." arXiv preprint arXiv:1705.06830 (2017).

2. https://www.kaggle.com/c/painter-by-numbers/data

3. http://image-net.org/download