Identifying Chinese Calligraphers from their Characters: A manifold-learning approach

Evan Gerritz
Yale University
evan.gerritz@yale.edu

Abstract

Many humans have the remarkable ability to correctly guess the artist of a painting they have never seen before, so long as they have seen enough other paintings by that artist. This fact implies there may exist an invariant between multiple works by one author, which we will refer to as "style." A few questions then naturally arise: 1. Can we teach a computer to learn such an invariant? 2. If so, by analyzing what the computer does, can we arrive at an understanding of how humans perform this task? 3. How can we be sure that the computer is truly learning the invariant and not memorizing image statistics?

We looked at Chinese calligraphy as an example of a particularly challenging instance of this problem. We hypothesized that the output of an intermediate layer of a sufficiently advanced deep neural network can be used as a kernel to learn a manifold of calligraphic style. We trained two network architectures from scratch and used their activations as kernels for the embeddings. Assuming the manifold of style is nonlinear, we used the diffusion maps nonlinear dimensionality reduction algorithm to assess a given kernel's ability to cluster images of characters never seen by the neural network. To assess the validity of our nonlinearity assumption, we compared the results of this nonlinear embedding to a linear embedding using Principal Component Analysis (PCA). Additionally, we performed a qualitative analysis of several embedding kernels to understand what features they were trying to detect. Ultimately, we created a kernel that produced compelling clusters of calligraphers on the learned manifold of calligraphic style.

1 Introduction

1.1 Chinese calligraphy

Limitation breeds creativity. Perhaps no more does this aphorism hold than in the discipline of Chinese calligraphy (in Chinese, 书法 (*shu'fa*)—literally "rules/methods of writing"), which is one of the oldest and deepest art forms in the world. In contrast to other art forms, Chinese calligraphy is characterized by the use of only four materials: ink brush, ink, paper, and inkstone. The ease with which experts can differentiate a character drawn by the "four great masters of regular script" (楷书四大家), in Figure 1, then, is quite surprising and speaks to their advancement of the craft. The fact that these calligraphers have distinctive styles is not incidental but essential both to the art form, as well as the creation and preservation of these works. To become a proficient calligrapher, one begins by reproducing as exactly as possible the characters written by famous calligraphers (a process known as 临帖 or *lin'tie*) before attempting to create a work of their own.

1.2 Artist recognition

This emphasis on distinctive style is what led us to explore Chinese calligraphy in particular. The authors of a previous, related work trained a deep neural network to recognize the artists of paintings using a network that has seen examples of those artists' paintings (van Noord et al., 2015). This approach is a supervised classification algorithm, which comes with the limitation that it can only be used to classify inputs into the class labels present in the training set.

For example, consider someone trying to train a network to recognize handwritten digits, however, their dataset only has examples of the digits 1 through 9. Even if the network is trained with 10 output classes, when a 0 is used as input to the network in testing, the network will never correctly classify it as the 10th class, since the backpropagation algorithm will give 0 weight to any connection leading to a class for which there were no examples.

This fact is interesting as the authors of the paint-

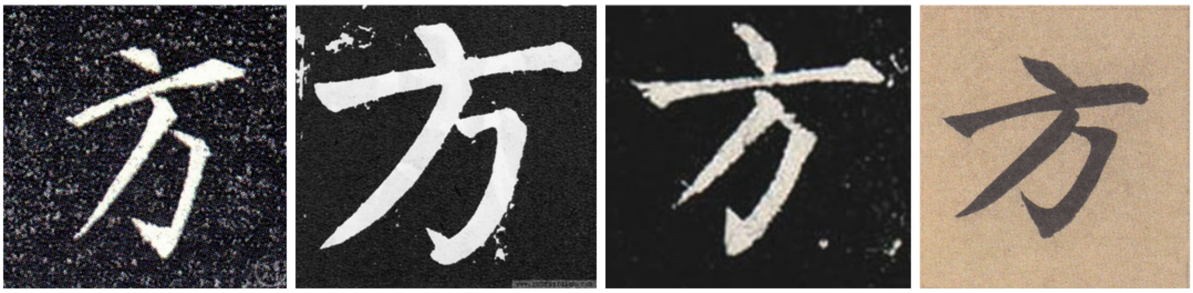

Figure 1: Four instances of the character 方 (place) written by (from left to right): Ouyang Xun (欧阳询, 557-641 CE) Yan Zhenqing (颜真卿, 709-785 CE), Liu Gongquan (柳公权, 778-865 CE), Zhao Mengfu (赵孟f ǔ (obscure character), 1254-1322 CE) (Gao, 2007). The original paper on which these calligraphers wrote has long disintegrated, so these images come from copies of their work in the form of steles一large stones in which master carvers painstakingly transferred the ink characters on paper to engravings in a rock, a significantly more durable material.

ing recognition paper claim that the network is learning something about each artist's style, but how can it be shown that their network has learned an invariant and not memorized image statistics for each of the artists. If every painting in the data set for some artist has a certain frame, or was photographed under specific conditions, the network may just memorize these particulars, and potentially fail to generalize to other datasets. Our project attempted to find a way to consider the generalizability of a classification network outside its training classes. In other words, if we train a network on French impressionists, can we get it to tell us anything interesting about Renaissance paintings?

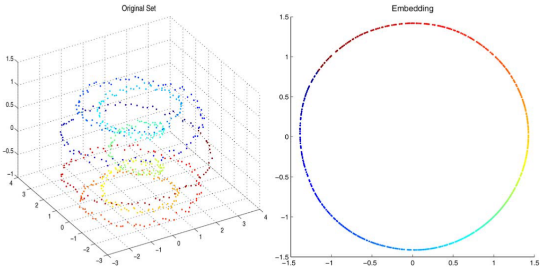

Figure 2: (from Coifman et. al., 2006): On the left, notice how the colors on the helix change according to one's location on the helix and not in Euclidean space itself. Only a nonlinear dimensionality reduction can express this relationship; the result of a diffusion map embedding is on the right.

1.3 Diffusion Maps

Our approach, then, must rely on an unsupervised learning algorithm. We used the nonlinear dimensionality reduction technique called diffusion maps, which we will give some background on now.

Dimensionality reduction techniques are useful for finding patterns in data that are high dimensional (such as a 128x128, 8-bit image). Linear algorithms, such as Principal Component Analysis, are useful as they find the "best" lines onto which to project the data, according to some criterion (maximum variance while maintaining orthogonality, in the case of PCA); however, the linearity restriction will fail to capture a potentially rich manifold on which the data truly lie. Figure 2 demonstrates the necessity for nonlinear techniques.

The diffusion maps algorithm solves this linearity constraint by considering each datapoint as a node in a graph, with connections to each other node weighted according to some similarity measure. If one now considers taking random walks on this graph using the similarity measure between two nodes as the transition probability (after rescaling such that the values are between 0 and 1), one can create a notion of how "close" two nodes are—not in the Euclidean sense, but instead how close they are on the manifold on which we believe the data live. To actually compute these manifold embeddings, one computes the singular value decomposition of the similarity matrix, after rescaling by the row sums, to get the coordinate functions of the learned manifolds.

Using these coordinate functions, one can compute the coordinates on which each artist lives on a calligraphic style manifold, in order to obtain insights into the latent, intrinsic structure of calligraphic style.

2 Methodology

2.1 Data

We used a dataset containing over 100,000 black-and-white, 64x64 images of individual characters drawn by 20 calligraphers. Some examples of these images are shown in Figure 3. This dataset was augmented to become larger and thus more generalizable by performing random rotations, horizontal flips, random crops, and pixel-value normalization.

2.2 Kernels

We experimented with several different kernels for the diffusion embedding, becoming increasingly sophisticated. The first was simply a cosine similarity between the pixel values. The other two consisted of intermediate activations of deep convolutional neural networks trained to classify the calligrapher based on the characters. The intuition for this choice is that a network trained to classify characters' artists will be forced to learn some efficient representation of the characters that is useful for distinguishing their artists on the basis of style. For example, some form of alignment or transformations of the data is likely necessary in order for a comparison between two images to be more complicated than, say, which one has a black pixel at a certain location. Instead of manually creating a kernel, we can simply create a neural network that is highly effective at distinguishing the calligraphers and then try various intermediate layer activations to see which one results in the best clusters in the embedding.

2.3 Deep CNN Models

For the deep CNN models, we started by training the simpler VGG16, which is composed of 13 convolutional layers and three fully-connected layers. It consists of convolutional blocks, each comprising multiple convolutional layers followed by max-pooling operations. The final layers are dense layers for classification.

We also trained the more advanced ResNet-18, named for its 18 layers. These layers consist of residual blocks each with two convolutional layers, which use skip connections to solve the vanishing-gradient problem. Initialization is performed using He initialization for more efficient convergence.

2.4 Training Procedure

To determine good hyperparameters for the deep networks, we performed a grid-search of various choices within the hyperparameter space. Specifically, we tested all combinations of the following: Adam and SGD optimizers, three learning rates spanning a logarithmic range from 1e-5 to 1e-1, and four weight decay parameters.

The final hyperparameters were chosen by using those which empirically maximized the test-set accuracy after 10 epochs of training. The final training was performed over 150 epochs using the Adam optimizer without AMSGrad and a batch size of 256 for both models. For VGG16, a learning rate of 0.001 and a weight decay of 0.0001 were used; for ResNet-18, a smaller learning rate of 0.0001 and a larger weight decay of 0.001 were found to lead to faster convergence. The models were trained using training and validation datasets, and performance metrics were recorded after each epoch.

3 Results

To establish a baseline for later results, we first created a diffusion maps embedding by calculating the cosine similarity of each pair of untransformed images. As can be seen in Figure 6, this resulted in no clustering by calligrapher. We are looking for clustering, as this feature would imply that we have found a coordinate system in which different characters drawn by one person are close to each other. In other words, we have located that calligrapher on our hypothesized manifold of calligraphic style, and moreover there must be some invariant latent in all of this person's characters (style, perhaps) which allows a human to distinguish that person's characters from others'.

Quantitatively, we measured the degree of clustering of an embedding by performing K-means to estimate clusters based on the embedding and then comparing those clusters to the ground truth of clusters based on each calligrapher. This comparison was done by calculating the *normalized mutual information* (NMI) between each of the points. An NMI of 1 means there is a perfect correspondence between the class labels and the predicted labels, i.e., if a point x has the label y_1 in the first labelling, there is an equivalent label y_2 in the second labelling containing x along with every other point labelled as y_1 in the first labelling. An NMI of 0 means there is no correspondence between the two labellings, or, put more intuitively, knowing a point has label y_1 in one clustering gives you no information about its label in the second labelling

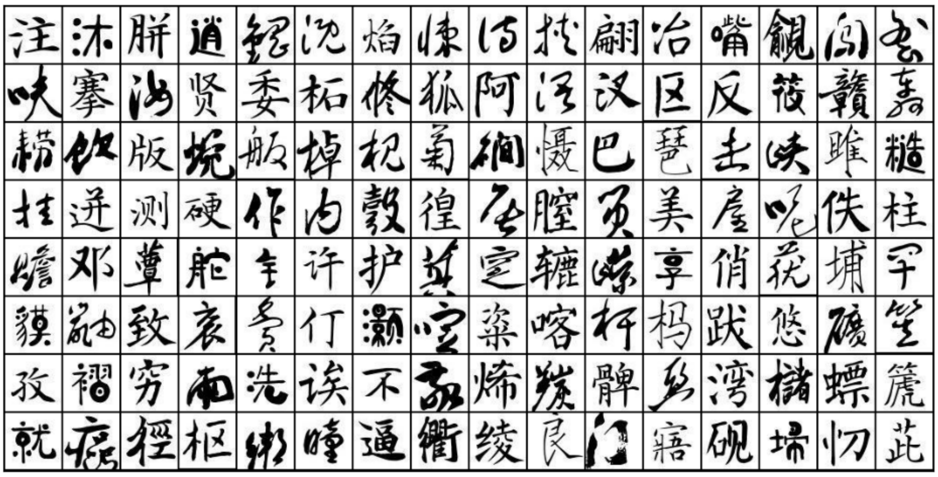

Figure 3: Subset of dataset of individual characters drawn by various calligraphers. See if you can find a few characters that appear to look like they were drawn by the same person. This task is essentially what we are trying to train the network to do.

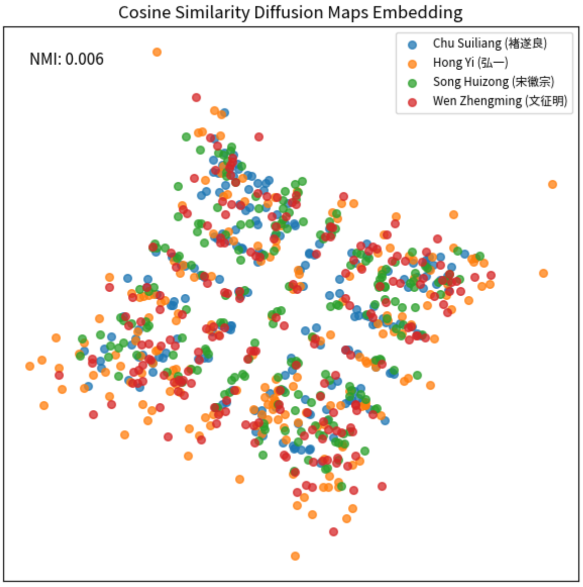

Figure 4: Applying diffusion maps using a trivial kernel results in no clustering by calligrapher, meaning the embedding has failed to learn the coordinates of style.

y_2. For the purposes of this paper, a high NMI means that the K-means cluster labels correspond well to the ground-truth calligrapher labels. We want to learn an embedding that leads to clustering based on calligraphers, so we want to find a kernel that maximizes the NMI. The embedding in Figure 6 was 0.006, so achieving anything higher would mean our project was a success.

As described earlier, we chose to use a deep CNN to provide us with a kernel more capable of identifying the most important features of an image of a character. The VGG16 model we trained had an accuracy of 95% on the test set, meaning it was quite successful at distinguishing the various calligraphers, even from images the network had not seen before. To use this network as a kernel, we fed the network an image of a character through each layer of the network until we hit the final convolutional layer. At this layer, we took the output vector consisting of the activations of each neuron in that layer and used that vector instead of the input image as a feature vector to compare via cosine similarity all the data points for an embedding.

The results of this approach for two different sets of calligraphers are displayed in Figure 5. One embedding exhibits a high degree of clustering, and furthermore these clusters are highly correlated with the ground-truth calligraphers, while the other exhibits significantly less clustering. One detail intentionally omitted earlier is that while there

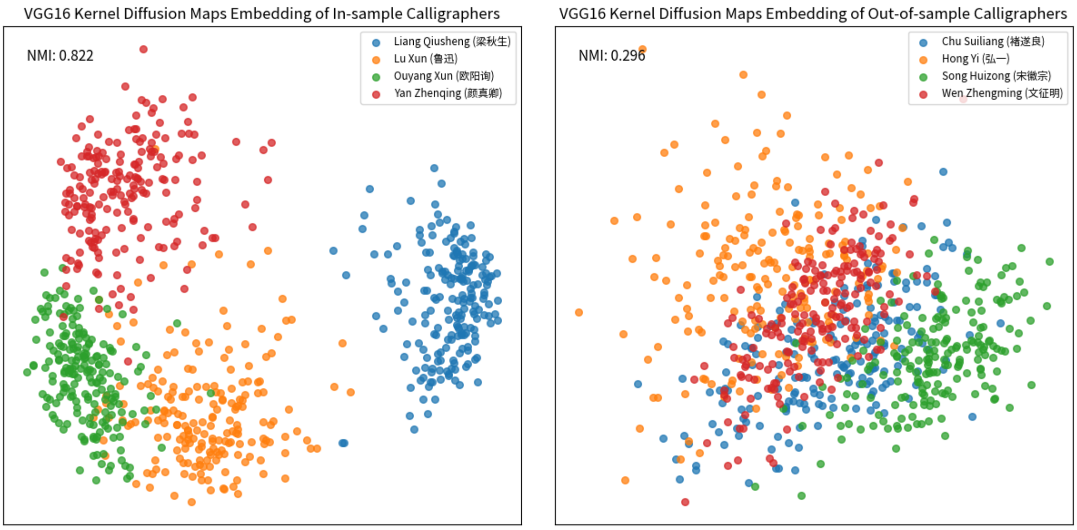

Figure 5: On the left: using the VGG16 intermediate activation layer as a kernel resulted in spectacular embeddings and our hypothesis was confirmed and the project was completed. On the right: ...so long as the embedding only included artists the kernel's network had already been trained to recognize. Clearly, VGG16 was mainly memorizing the artists, and it is unclear whether it was learning anything invariant regarding style.

were images for characters from 20 calligraphers in our dataset, the model was only trained using 15. The 'good' (higher NMI) embedding consists exclusively of artists the CNN was trained on, a network that was very capable of distinguishing artists based on their characters, so of course the activation of an intermediate layer close to the output layer resulted in largely separable clusters. The 'bad' embedding consists entirely of artists not seen by the neural network, and the NMI has dropped significantly.

Our hypothesized manifold of style should generalize to artists not seen by the network, or else it cannot be called a manifold of style but rather a manifold of image features the network learned to distinguish artists it had many examples of. This result reveals the thorniness in (van Noord et al., 2015)'s claim that their network learned something about the style of an artist, since they had no way of testing the generalizability of their network to artists not in the training set. By looking at embeddings of the intermediate layers, however, we were able to test the network's generalizability without a new dataset. This approach does require one to retrain an entire network for each subset of classes to which would like to test the ability of the network to generalize.

So, we found an initially promising result that suggested it was possible for a computer to perform the visual differentiation of artists based on style that humans can do. But along with it, we discovered a method of testing whether the computer was performing this differentiation by learning something about style, or whether it was simply memorizing the artists in the training set. It turned out the VGG16 kernel was not learning as much about style as it first seemed. The natural next question is can we find a more advanced kernel that can learn the manifold of style?

For an answer to this question, we turned to the more recent and more advanced ResNet-18 model. We trained this model on 15 calligraphers, however, even with significant experimentation with hyperparameters, this model converged much more slowly than VGG16, so the resulting model actually had a lower accuracy of 93.7% on the test set. Despite this lower accuracy, however, the model was still able to produce embeddings with a significantly higher level of clustering corresponding to the ground-truth labels than VGG16 on out-of-sample calligraphers. This result means that the network seems to have truly learned something about calligraphic style.

To assess our nonlinearity assumption of the calligraphic style manifold, let's compare diffusion maps to a linear dimensionality reduction algorithm, using the same kernel. The embedding is shown in Figure 7 Using the same kernel as the diffusion maps embedding, the principal component embedding has an NMI that is around 20% lower than the diffusion maps embedding, indicating that the manifold of style is likely somewhat nonlinear.

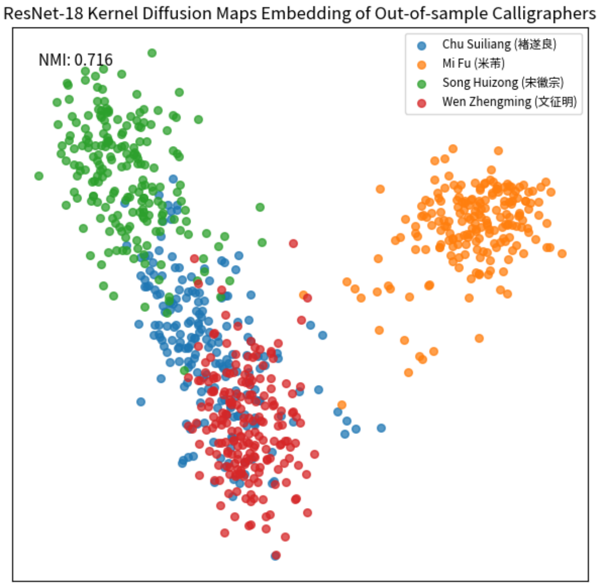

Figure 6: The ResNet-18 Kernel embedding contains quite prominent clusters, even on calligraphers the network had never seen before! This fact implies it has actually learned a coordinate system of calligraphic style, as desired.

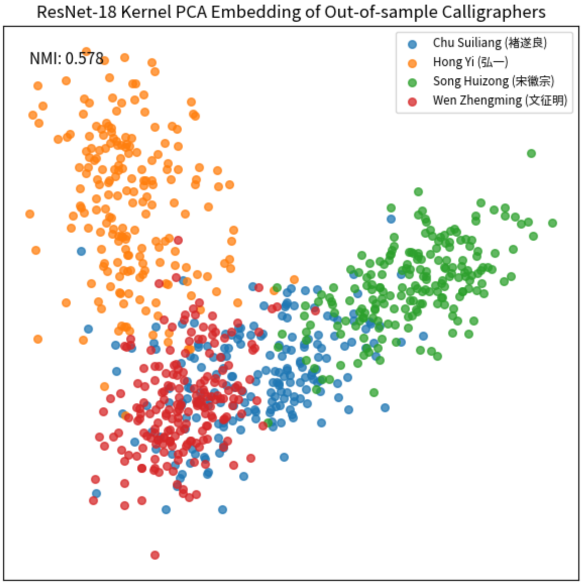

Figure 7: Projection of the same points as the ResNet-18 out-of-sample diffusion maps embedding but onto the first two principle components, a linear dimensionality reduction.

So we have found our manifold, and it appears to be somewhat nonlinear, but what did it learn? For insight into this, we need to look at an embedding with more datapoints and examples of their characters. Figure 8 contains an embedding of 10 different calligraphers, and notably the clustering's correspondence to ground-truth remains quite high. The separability of the clusters would likely get even better if we used more diffusion coordinates and making this a 3D plot. The reason the NMI is higher here than with only four artists plotted before is that many of the artists in this plot are in-sample.

Looking at the representative characters for each of the clusters, it is not obvious to discern what these coordinates are quantifying. My best guess is that the x-coordinate is measuring line thickness, or perhaps line-width variation, and the y-coordinate reflects some aspect of the speed at which the character was written (with larger y being written faster). These coordinates make sense because they are not character-dependent, but rather dependent on other factors unlikely to change between different characters by the same artist.

The speed of the strokes, which has great influence on both the neatness of the character and definition of lines, is likely similar between multiple characters by one artist. Furthermore, the thickness of the lines is dependent on technique, such as how much ink is applied to the brush and how much pressure one applies, as well as physical aspects of the materials used, such as the size of a brush one uses, the absorbency of the paper, and the viscosity of the ink. These are the kinds of invariants that are likely used by humans to differentiate characters drawn by different calligraphers, and the manifold we learned provided support for this theory.

4 Conclusion

While our initial angle was quite specific and practical (can one teach a computer to recognize Chinese calligraphers by learning a manifold of calligraphic style?), it led us to explore some broader, theoretical questions. What does it mean for a computer to implement a similar computation as a human? How can we be sure a system is even learning anything semantically meaningful in learning to classify things? Our research direction led us to a method of evaluating the generalizability of a model beyond its training set, without using any external datasets by looking at clustering on em-

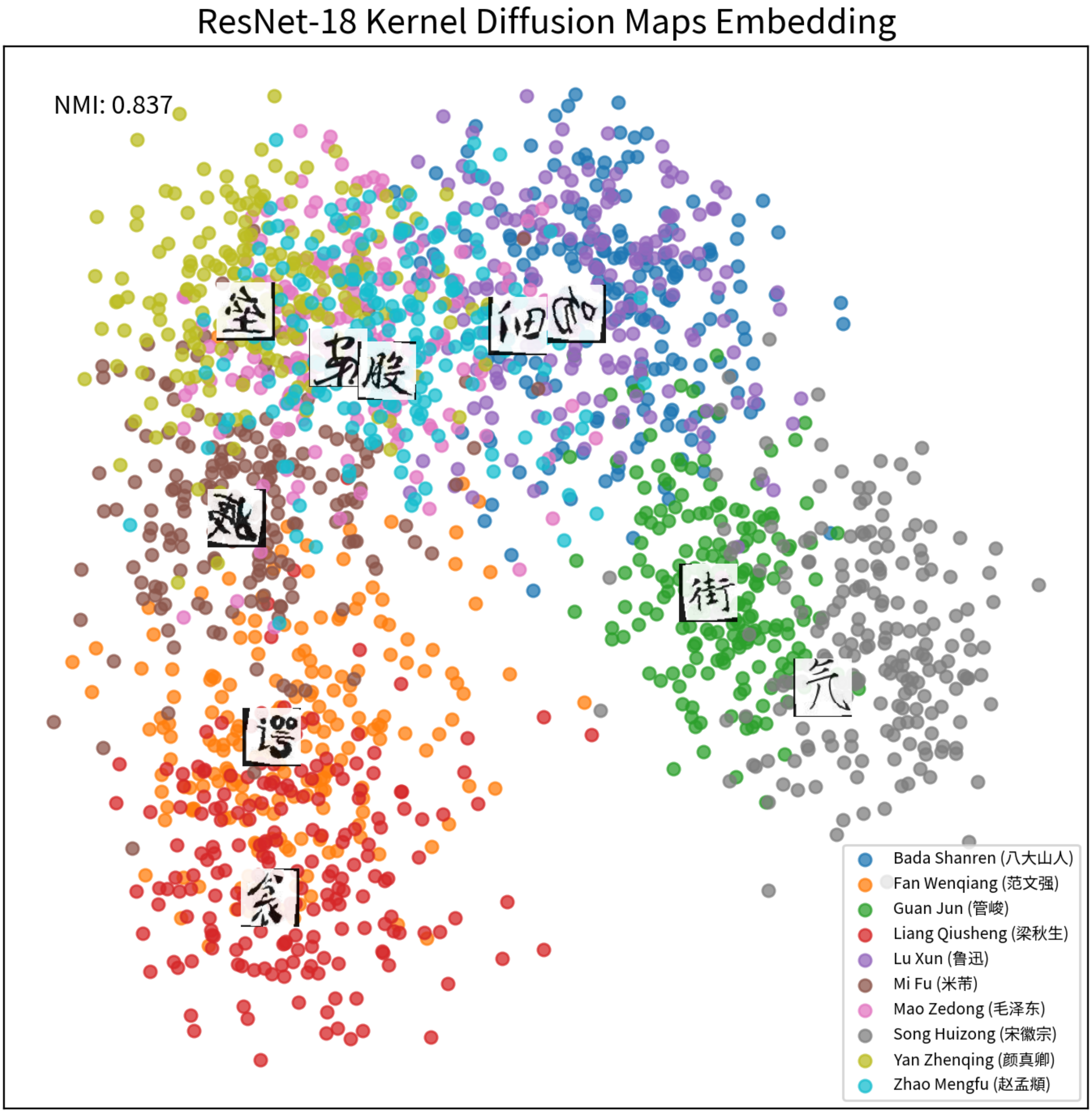

Figure 8: Even with 10 different artists, there are quite visible clusters. The images of characters are the images of the character closest to the centroid of each cluster. Thus, they represent what a typical character in each cluster looks like. Using these representative characters, one can attempt to analyze what element of style each diffusion coordinate is looking at. From left-to-right, notice that the average line thickness seems to increase. From top-to-bottom, the speed of the strokes seems to be increasing.

beddings.

Additionally, we found evidence supporting that the most important stylistic differentiations are related to physical characteristics of the process of character creation. Future research in this area could look at significantly larger datasets with more calligraphers, as well as how much the learned manifold changes when leaving out different sets of calligraphers in the train sets. This manifold also becomes a powerful tool for examining other aspects of calligraphic style: Are certain locations on the manifold physically impossible to create? How have positions of calligraphers on the manifold changed over time?

Acknowledgments

I would like to thank Professor Steven W. Zucker and Dr. Luciano Dyballa for their guidance and support throughout the project. They have been wonderful mentors and friends for the past few years, and I have cherished our time working together.

References

R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker. 2005. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proceedings of the National Academy of Sciences*, 102(21):7426–7431.

Ronald R. Coifman and Stephane Lafon. 2006. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30. Special Issue: Diffusion Maps and Wavelets.

Changsan Gao. 2007. China's calligraphy art through the ages.

Kauvin Lucas in Kaggle. 2021. Calligraphy style classification.

Nanne van Noord, Ella Hendriks, and Eric Postma. 2015. Toward discovery of the artist's style: Learning to recognize artists by their artworks. *IEEE Signal Processing Magazine*, 32(4):46–54.