CS230

Neural Genre Transfer (Computer Vision)

Tyler Consigny
Department of Symbolic Systems
Stanford University
tconsign@stanford.edu
SUNet ID - tconsign

Ashwin Arasu
Department of Management Science and Engineering
Stanford University
ashwin04@stanford.edu
SUNet ID - ashwin04

1 Introduction

Existing examples of Neural Style Transfer applications effectively obtain results that represent a particular image or artists' distinctive physical style. While some current implementations do allow for multiple style images to be used, it seems these are often used with very stylistic images (painting styles). We wanted to develop a method for multiple style transfer that focuses on a more ambient, less narrow, effect on the content image. To do this, we thought of using sets of same genre movie images to apply a neural 'genre' transfer. We implemented a style transfer that combines multiple styles that are representative of the larger 'genre' that is to be emulated. By generalizing the source of the style transfer, we hoped to produce results that encapsulate the more nuanced, semantic features that pertain to an image's genre or ambience. In a sense, this is a neural "vibe" transfer, as the young folks would say.

2 Dataset

(a) A background picture for *The Good, the Bad, and the Ugly*

(b) A movie poster for *Aladdin*

We used The Movies Dataset[1] on Kaggle to obtain 45,000 movie id numbers from the TMDB database[2]. We then utilized the TMDB API to obtain background images and poster images associated with those movies. Both groups of images were grouped according to the genre of the movies they belong to. We obtained about 45,000 images from each. The background images were images from, or about, the movie that usually did not contain text. The poster images were usually text containing

,

posters of the movie. The images in the dataset, and within each genre, were quite varied. This was a large motivation for using a movie genre classifier to optimize the selection of images. We wanted to find a set of images that all optimally exemplified a given genre.

3 Methods

Our approach was two-step. We first needed to find an optimal set of images to use during neural style transfer. Second, we had to implement and optimize a version of neural style transfer using multiple style images. The first step centered around training a movie genre classifier. Using this model, we would create a 128-dimensional vector encoding of the images and find the images that were classified with highest probability into each genre. To do this, we decided to use the VGG-19 model with weights pre-trained on ImageNet. We dropped the last three pre-trained layers and added our own fully connected layers (1024,128,8), with a softmax activation on the last. We used a categorical cross entropy loss function with Adam optimization algorithm.

$$L_{CE} = -\sum_{i=1}^{8} y_i \log(\hat{y})$$

Training for 10 epochs with mini-batches of 64, we were able to achieve an accuracy of about 74% on the test set. After training, we obtained the predictions from the data and stored the activations after the 128-node Dense layer as encodings for the images. We then went through the predictions and grouped the five images that had the highest prediction probability for each genre. We also found the 4 images that were closest in Euclidean distance to the highest predicted image from each genre. We downloaded all of these images along with five random pictures from each genre. Using these sets of images, we optimized a neural style transfer implementation to best convey our idea of neural 'genre' transfer.

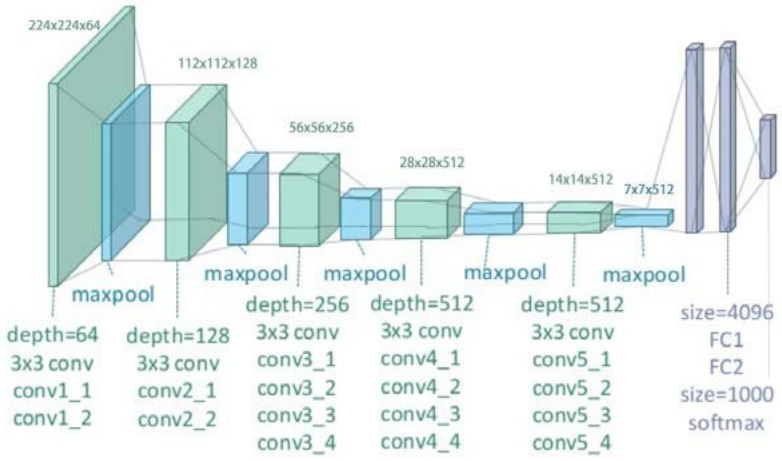
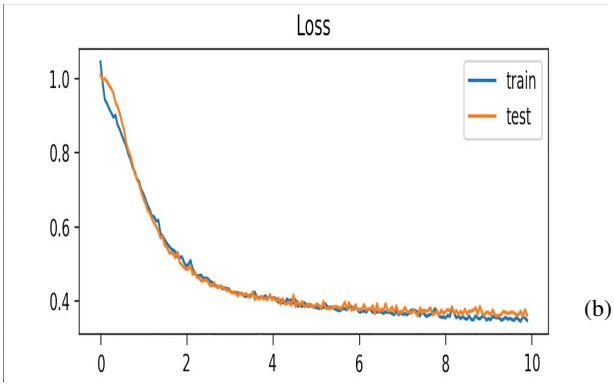


Figure 2: Our VGG-19 model uses randomly initialized layers of dimension:1024, 128, and 8 for the last three layers

The implementation we chose allowed for multiple style images[4]. This implementation utilizes a pre-trained VGG-16 model and provided features such as color preservation from the content image and style weighting. As a first attempt, we used the five random images from each genre as styles. After persistent hyperparameter searches, a satisfactory image could not be obtained while using all five random images. The same outcome occurred when we tried to use the top five images from each genre according to probability assigned by softmax. The likely reason for this is the vast differences between each image in these sets. This is explored in our report video. However, when we used the top image and those closest in Euclidean distance, we were able to approach the wanted result. Our best results came after iterating through various hyperparameters available in the implementation.



(a) Training and test loss for the genre classifier over 10 epochs

Train Accuracy	0.7507
Train Loss	0.3781
Test Accuracy	0.7389
Test Loss	0.3811

(b) Accuracy metrics for the genre classifier

The hyperparameters that we focused on were content weight, style weight, iterations, learning rate, pooling, and style blending. The content weight and style weight allowed us to put different coefficients in front of the style and content portions of the total loss function. We found 0.9 for style weight and 0.3 for content weight to be the optimal values. After doing a search for the learning rate, we found 20 to be most effective. When it came to pooling, average pooling achieved slightly better results than max pooling. Finally, the style blending parameter allowed us to weight the impact of each style image on the loss function. For this value (inputted as an array), we found that trying around 5 different combinations of values was sufficient to arrive at a satisfactory result image.

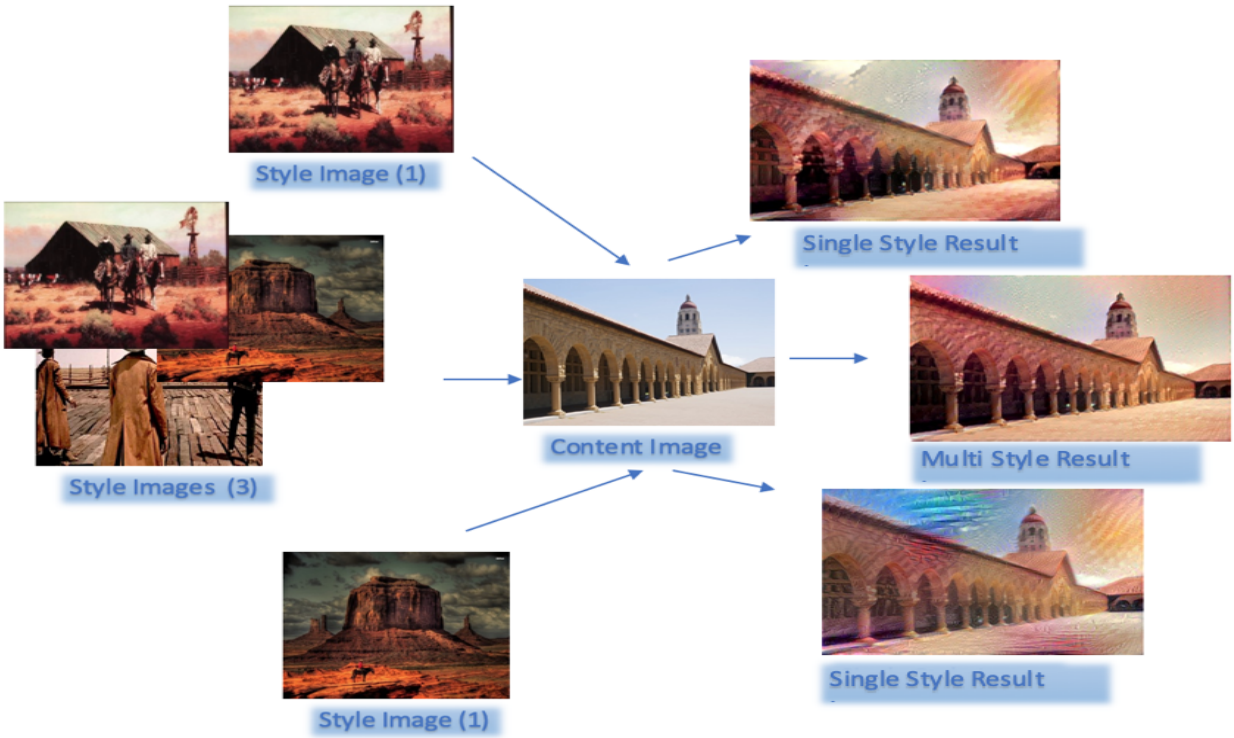


Figure 4: Example of the multi style transfer process versus the single style transfer process

4 Results

Evaluation was done by manually examining the results of the genre transfer. As the success of the transfer is largely subjective, we did not use a quantitative analysis for this part of our project. We did, however, qualitatively compare the results of single style transfer with those utilizing multiple style transfer.

Our results are comprised of 24 single-style images and 24 multi-style images, for a total of 48 new images. To create the 24 single-style images, we applied the top image (according to the genre classifier) from each of eight different genres to a set of three content images. As for the 24 multi-style images, we used the same sets of genres and content images, but we applied three style images rather than one. These three style images were: the top image from each genre (same as single-style) and the two closest images to that one by Euclidean distance between image encodings. We then rated the images in both groups on a scale from one to ten, with one being a gross misrepresentation of the target genre and ten being a representation that we can instantly and unquestionably link to the target genre. After rating each image, we averaged each group's results. Single-style transfer's average rating was 4.2, while the multi-style transfer images' was 5.9.

As a final exploratory test of our results, we inputted the 48 result images from above into the genre classifier. We wanted to see whether or not it would identify our result images as the genre of their respective style images. We also were curious to see whether or not the genre of the multi style transfer images were identified with a greater accuracy than single style transfer images. We found that the genre of the single style images were predicted with an accuracy of 23% and the multi style images with an accuracy of 31%. We are aware that there could be some artificial effect on the accuracy by using the images that performed best on the genre classifier. We feel this may optimistically bias the accuracy on this exploratory test. However, we thought it would be an interesting way to bring our project full circle.

(a) Neural Style Transfer using a single western style image

(b) Neural Style Transfer using three western style images

5 Limitations and Future Work

We were largely satisfied with our results but see many paths forward for improving the effect of the genre transfer. One of the major difficulties in our project was imparting the style of a genre without using heavily stylized images. Neural style transfer, understandably, seems to work best while using highly stylize images like *Starry Night*. Our goal was to impart the ambient nature of a genre using mostly photo-realistic/non-stylized images in the neural transfer algorithm. There are various alternative approaches that may help overcome this difficulty. In the future, it would be interesting to use methods such as deep photo transfer[7] and GANs[8].

6 Contributions

Tyler handled most of the data acquisition, data pre-processing, and the building/training of the genre classifier model. Ashwin handled the majority of the neural style transfer implementation and evaluation of the results of the transfer. However, both were active during all parts of the project.

References

[1] `https://www.kaggle.com/rounakbanik/the-movies-dataset`

[2] `https://www.themoviedb.org/documentation/api`

[3] Gatys, L.A., Ecker, A.S., & Bethge, M. A Neural Algorithm of Artistic Style. ArXiv, abs/1508.06576. 2015.

[4] athalye2015neuralstyle, Anish Athalye, Neural Style, `https://github.com/anishathalye/neural-style`. 2015

[5]Jing, Y., Neural Style Transfer: A Review. arXiv:1705.04058v1. 2017

[6] Cui, Qi, Wang, Multi-style Transfer: Generalizing Fast Style Transfer to Several Genres. 2017

[7] Luan, Fujun, et al., 2017, "Deep photo style transfer." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017

[8] Huang, Xun, and Serge Belongie. "Arbitrary style transfer in real-time with adaptive instance normalization." Proceedings of the IEEE International Conference on Computer Vision. 2017.