

MAIS202 Final Project Deliverable 1: Artistic Style Transfer

Rosie Zhao

1 Dataset

Since the release of Leon A. Gatys' paper, A Neural Algorithm of Artistic Style [1], many projects and applications of neural style transfer have been developed. The process of neural style transfer involves having two input images, a content image and a style image, and a single output image that is similar to both images—emulating the second image's 'style' while retaining the first image's content. As a big Studio Ghibli fan, I am aiming to take landscape photos from two of my favourite places (the Rocky Mountains and San Francisco) to generate a new image in the style of the scenery seen in these movies. See the folder inside the GitHub repository for the photos.

2 Methodology

2.1 Data Preprocessing

The input of this algorithm is two images, so I have some flexibility with which images I can use; I have some of my own scenery photos to use as content images, and I also chose style images with similar features to see if I can make a more seamless transfer (and maybe experiment with more abstract/unrelated style images in subsequent runs?). Neural style transfer takes quite a bit of computing power because particularly 'accurate' output images require several iterations (I've looked at implementations that use over a hundred iterations, and some that only use 10 to achieve decent results?). Thus I'm resizing both the style and content images to be **224 px x 224 px** (I believe the VGG-Network referenced below takes these fixed dimensions as input) to accommodate for this— unfortunately no high-res pictures!

2.2 Machine Learning Model

This project will attempt to implement the Convolutional Neural Network model explained in the well-known paper published by Gatys et al. [1] Although I do not have a gauge of the computing power required to implement this model, I am slightly concerned about implementing this on my local device, so I have looked into using Google Colaboratory to run the model on. The results presented by Gatys was generated on the basis of the 19-layer VGG-Network [2] that tracks both the squared-error loss between the feature (content) representations as well as building style representation on top of each layer of the network. I also looked into the more recent method of using adversarial networks [3] to implement style transfer, but it seems more difficult and I was advised to use CNNs instead.

2.3 Final Conceptualization

I'm aiming to present the images on a simple web application. I'm not very experienced with web development, so the implementation of something like deepart.io (allows users to input photos that are sent to a server to be processed) is ideal but perhaps not completely feasible. I would have to consider the possible dimensions of input images, and/or restrict the selection of Studio Ghibli style images.

References

1. L. Gatys, A. Ecker, and M. Bethge. A neural algorithm of artistic style. (2015). URL <http://arxiv.org/abs/1508.06576>. ArXiv: 1508.06576.
2. Simonyan, K. Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. (2014). URL <http://arxiv.org/abs/1409.1556>. ArXiv: 1409.1556.
3. J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. (2017). URL <http://arxiv.org/abs/1703.10593>. ArXiv: 1703.10593.