Project # 5: Final Project

CSC 391: Computer Vision

Caroline Li

May 6, 2019

For my final project, I wanted to integrate my interest in art with computer vision. Inspired by Memo Akten, I wanted to see if I could create art from input from my laptop webcam. I was curious to see what a picture or video of myself would look like if a famous artist had painted my portrait. For this project, I utilized Cycle GANs to map webcam input to paintings by Monet, Cezanne, Van Gogh, and Ukiyo-e.

Cycle GANs Overview

Generative Adversarial Networks (GANs) consist of a generator and a discriminator. While discriminative algorithms predict labels from features, generative algorithms predict features from labels. The generator generates fake images, and the discriminator classifies the fake image. The generator tries to generate authentic-looking images that are, for the discriminator, indistinguishable from the ground truth. The generator essentially maps unpaired images from domains X to Y; for example, this could be unpaired images of zebras and horses, if we were interested in generating images of horses from images of zebras. The generator improves during training due to the use of adversarial loss, which encourages outputs similar to the original data set via negative log likelihood.

Cycle GANs utilize two mapping functions $G: X \rightarrow Y$ and $F: Y \rightarrow X$ and their respective discriminators. Cycle GANs introduce the use of a cycle consistency loss to improve upon adversarial loss. Cycle consistency essentially means that if we map onto another domain, and then map to our original domain, the result should be the same as the original input. So, $x \rightarrow G(x) \rightarrow F(G(x)) \approx x$ and $y \rightarrow F(y) \rightarrow G(F(y)) \approx y$, creating consistency between the two functions.

Thoughts on the project

While Cycle GANs provides two directions for mapping, I was really only interested in mapping photos to paintings of a certain style, rather than the other way around. The implementation I had found already included pretrained networks, I simply needed to have input be from the webcam. Going into this project, I knew computation power would be an issue. For example, there is quite a bit of delay in output for processing each image, even when I lower input image resolution.

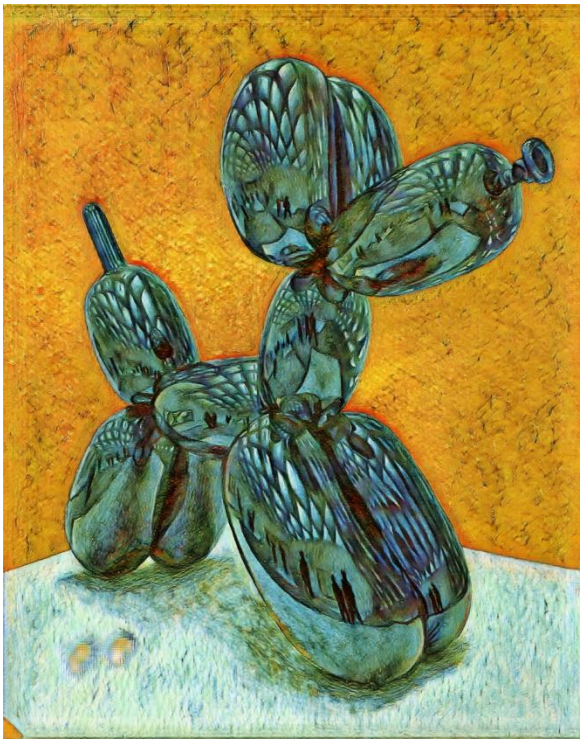
Below are two examples of what results can look like. These images were generated using the pretrained Cycle GAN using paintings from Cezanne.

Me, seated in my living room

My kitchen

Ideally, I could have input my webcam images at the original resolution. I did try running the pretrained networks on some images I had on my computer, to see what results would look like if the time to process each image did not matter. Below is the result of running the pre-trained Van Gogh model on a photo of one of my drawings. The results can be quite convincing, in both choice of color and the (artificial) application of paint. For example, we can see that the generated image has the yellow-ish hues and visible brush strokes that can be seen in Van Gogh's paintings.

Original image



Using pre-trained Van Gogh model

In addition to being able to input larger images real-time, I also believe it would be very interesting to have Cycle GANs map photographs to something completely unrelated. Paintings are often an artists' interpretation of what is around them, and served as photographs long before the creation of the camera. It would be interesting to see the results after training on images of food, oceans, microscopic images, cartoons, etc.