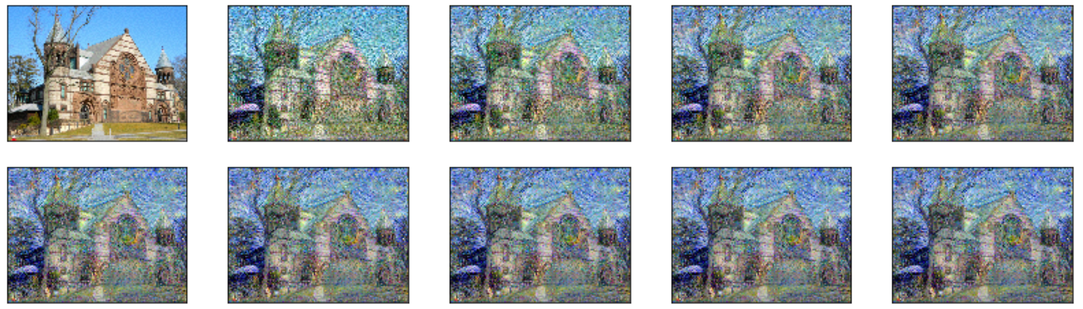
The Elements of Neural Artistic Style Transfer

Alexander Chen
alexanderKchen@princeton.edu

Princeton University, Department of Mathematics, Princeton, NJ



Motivation

- The fundamental problem of style transfer is to extract and apply the `style' of artpiece S, say a painting, music recording, or written work, to another artpiece X to generate a new artpiece Y that matches the content of X but with the style of S.
- The initial difficulty is how to define the `style' of a picture or piece of literature, but in recent years deep neural networks have provided a way.
- Neural artistic style transfer (or NST for short) found its first success in using deep convolutional nets.
- A number of NST unsupervised learning methods are surveyed here including GANs and Gatys' original method.

Deep Methods for Artistic Transfer

Gatys' method

Let C be the content image and S the style image. We want to create x with the content of C and style of S.
We use a neural network F, the pre-trained VGG-19 CNN.
Suppose $F(x)$ represents the neuron weights of the network when x is fed in.
The content distance is

$$L_c(x, C) = |F(x) - F(C)|$$

The style distance is

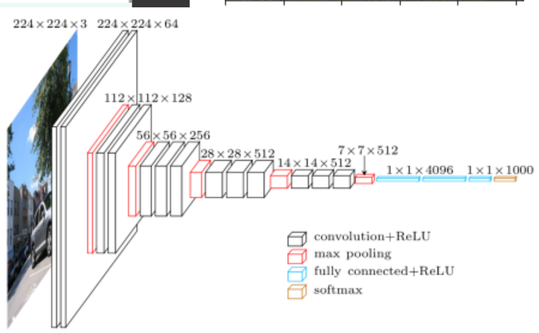
$$L_s(x, S) = |G(F(x)) - G(F(C))|$$

where G is the Gram matrix (of correlations). The goal is to make the given x as close as possible to the style of S and content of C

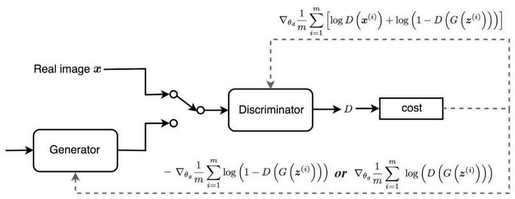
$$\min L = L_c(x, C) + kL_s(x, S)$$

k can be tuned up if you want more style. Gradient descent is

$$\Delta x = -\eta \partial_x L$$



Generative Adversarial Models



General gradient descent algorithm for GAN

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**
  for k steps **do**
    • Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
    • Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
    • Update the discriminator by ascending its stochastic gradient:

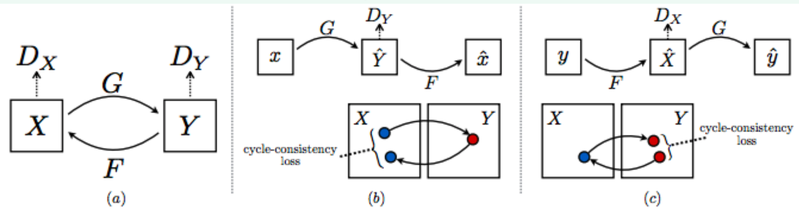
$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

  end for
  • Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  • Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for
The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

Cycle-GAN



$$\mathcal{L}_{\text{cyc}}(G, F) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1]$$

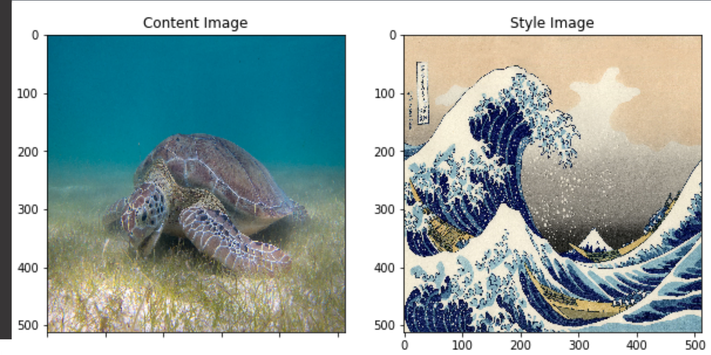
$$\mathcal{L}_{\text{GAN}}(G, D, X, Y)$$
For G, minimize $\mathbb{E}_{x \sim p_{\text{data}}(x)}[(D(G(x)) - 1)^2]$
For D, minimize $\mathbb{E}_{y \sim p_{\text{data}}(y)}[(D(y) - 1)^2] + \mathbb{E}_{x \sim p_{\text{data}}(x)}[D(G(x))^2]$

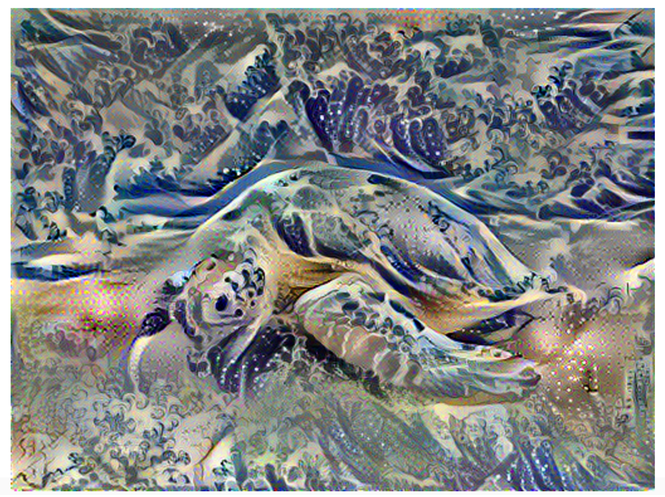
Minimize total loss function: $\mathcal{L}(G, F, D_X, D_Y) = \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) + \mathcal{L}_{\text{GAN}}(F, D_X, Y, X) + \lambda \mathcal{L}_{\text{cyc}}(G, F),$

Style-GAN

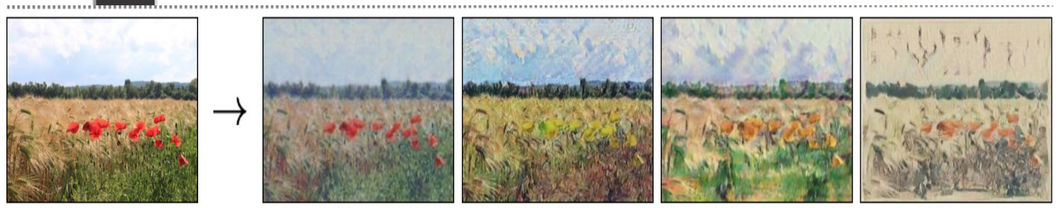
As the training progresses Style-GAN adds more layers to the generator network G.
Initially it starts with one shallow layer to produce low resolution images 4 by 4.
After a few iterations of learning `coarse' features of an image, G adds another layer to produce slightly higher resolution images 8 by 8.
It continues doing this, learning finer more detailed features as training progresses until producing HQ 1024 by 1024 images.

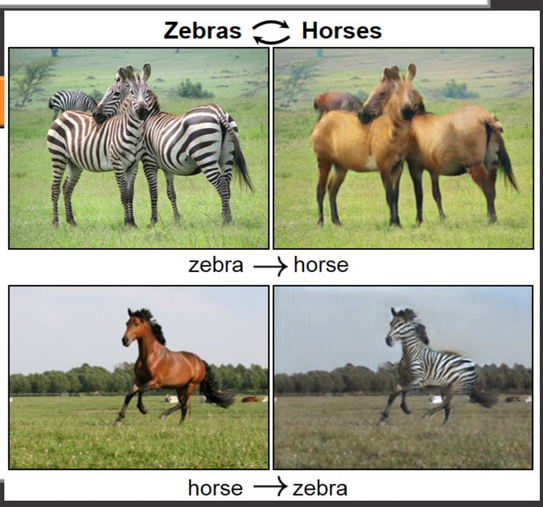
Gatys Artistic Transfer (GAT)





Cycle-GAN





Style-GAN
These people (probably) don't exist.

For similar stuff, see
https://thispersondoesnotexist.com/

References

Leon A. Gatys, Alexander S. Ecker, Matthias Bethge. *A Neural Algorithm of Artistic Style.*
Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, et al. *Generative Adversarial Networks*
Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros. UC Berkeley. *Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks*
Tero Karras, Samuli Laine, Timo Aila. *A Style-Based Generator Architecture for Generative Adversarial Networks*