Arbitrary Neural Style Transfer

Stanford University

Willow Wu; Ningyuan Zhang; Man Zhu {youw, ningyuan, manz68} @stanford.edu

PROJECT OVERVIEW

Style transfer models combine the content of one image with the style of another image[1]. Existing mobile apps only allow us to choose from existing style templates, because the fast neural style transfer method trains a feedforward neural network for each style[2]. We want to build an arbitrary style transfer network that can transfer an image to unseen styles. To resolve this problem, we add a layer after the first few layers of VGG to train the style. We tried three types of layers: (1) An AdaIN layer to transfer the channel-wise mean and variance[3], (2) a CORAL layer to align the covariance of the generated and style image, and (3) a histogram layer to match the cumulative distribution functions.

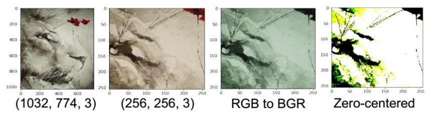
DATASET & FEATURES

Dataset

- Content: MSCOCO 2014, 330,000 images
- Style: WikiArt, 79,392 images

Preprocessing

- Resize the smallest dimension to 512 while keeping the aspect ratio, and then randomly crop a region of [256, 256, 3] from the images.
- Switch the image mode to BGR and then zero-center the image by subtracting BGR mean pixel, which is [103.939, 116.779, 123.68]



Features

- Raw inputs: RGB values of each pixel.
- Derived features: edges, colors, and textures.

MODELS

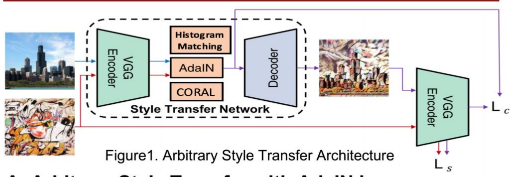


Figure1. Arbitrary Style Transfer Architecture

A. Arbitrary Style Transfer with AdaIN layer

The encoder is the first four layers of the pre-trained VGG-19 Network. Next, align the encoded style and content features by an AdaIN layer. Then go through a decoder to generate the new image, with loss function:

$$L = L_c + \lambda \cdot L_s \qquad L_c = \left\| f(g(t)) - t \right\|_2$$

$$L_s = \sum_{i=1}^{L} \left\| \mu\left(\phi_i(g(t))\right) - \mu(\phi_i(s)) \right\|_2 + \sum_{i=1}^{L} \| \sigma(\phi_i(g(t))) - \sigma(\phi_i(s)) \|_2$$

AdaIN layer aligns the channel-wise mean and variance of the content and style images:

$$AdaIN(x, y) = \sigma(y)\left(\frac{x - \mu(x)}{\sigma(x)}\right) + \mu(y)$$

B. Correlation Alignment Layer

CORAL layer aligns the input feature distribution of the content and style domains by recoloring the whitened content feature with the covariance of the style feature distribution.[4]

Table 1. CORAL Algorithm

```
Algorithm 1 CORAL for Unsupervised Domain Adaptation
  Input: Source Data $D_S$, Target Data $D_T$
  Output: Adjusted Source Data $D_s^*$
  $C_S = cov(D_S) + eye(size(D_S, 2))$
  $C_T = cov(D_T) + eye(size(D_T, 2))$
  $D_S = D_S * C_S^{\frac{-1}{2}}$                % whitening source
  $D_S^* = D_S * C_T^{\frac{1}{2}}$               % re-coloring with target covariance
```

C. Histogram Matching Layer

We replace the AdaIN layer with histogram matching, which maps the cumulative distribution function of the generated image to that of the encoded style image. Accordingly, we use the histogram loss for the style.

RESULTS

A. Qualitative Evaluation

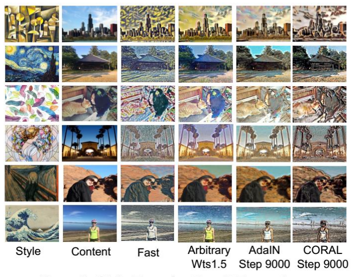


Figure2. Style Transfer Result Examples

Fast neural network is better at capturing the granularity of style while arbitrary style transfer can capture lower level style features well, such as general tints of an image, and can combine the content with distinct style textures in a more faithful way.

The CORAL model represents the style better than the AdaIN model, but the content loss of the CORAL model is large.

B. Quantitative Evaluation

Table 2. Loss Comparison of AdaIN Models with Different Hyperparameters after 10k Iterations

	Content loss	Style loss	Total loss
λ = **1.5**, learning rate = **10^{-4}**	5563	2010	**8579**
λ = **2.0**, learning rate = **10^{-4}**	9632	2752	15137
λ = **2.5**, learning rate = **10^{-4}**	12334	26129	77657
λ = **2.0**, learning rate = **10^{-3}**	14433	42157	98747
λ = **2.0**, learning rate = **10^{-5}**	15599	20058	55715

FUTURE WORKS

- **GANs**: Style transferred images have been criticized as lacking the “soul”. To address this problem, we can train a discriminator to act as art critics, and the generator can possibly learn the “soul” of art.
- **Architecture modification**: To promote style capturing, we can use a deeper neural network with residual architecture by adding additional skips from the encoder.

DISCUSSION & CONCLUSION

- **Arbitrary versus Fast Neural Style Transfer:** Arbitrary style transfer achieves the flexibility of style inputs at the cost of style representation quality. An intuitive explanation is that the fast method pre-trains one model for each style image, while the arbitrary method can be applied to any style. A more accurate explanation is two-folded. First, the arbitrary method only uses the first four layers of VGG, and this shallower architecture may not capture comprehensive features of the style. Second, AdaIN only matches the mean and variance, whereas gram matrix captures higher-order statistics.
- **AdaIN versus CORAL:** AdaIN only matches the mean and variance, which may not well capture the style features. CORAL captures second-order statistics and can capture more fine-grained style features.

REFERENCE

[1] Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge, "Image style transfer using convolutional neural networks," In IEEE Conference on Computer Vision and Pattern Recognition (CVPR) '06, 2016, pp. 2414-2423.

[2] Johnson, Justin, Alexandre Alahi, and Li Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," In European Conference on Computer Vision'10, 2016, pp. 694-711.

[3] Huang, Xun, and Serge Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," *ICCV*, 2017.

[4] Sun, Baochen, Jiashi Feng, and Kate Saenko, "Return of Frustratingly Easy Domain Adaptation," *AAAI*, vol. 6, no. 7, 2016.