Automatic Colorization of Line Sketch Images

Lilly Kumari (lkumari@uw.edu), Narendra Shivaraman (narensh@uw.edu)

Introduction to Deep Learning - CSE599G1 - Fall 2018

UNIVERSITY of WASHINGTON

Introduction

Automatic colorization of line sketch images is a challenging image-to-image translation task that has many applications in image compression, information mining, and digital art. The aim is to generate visually appealing and realistic colorized images from line/outline sketch images.

Dataset

We use the UT Zappos50K shoe 256x256x3 image dataset [1], along with their line sketch images. For this project, we have resized the images to have the dimensions 128x128x3. We use 48,025 images for training the networks. We use 200 images to observe the colorizations.

Figure 1: Eg. 1

Figure 2: Eg. 2

Figure 3: Eg. 3

Approach

We train convolutional neural network (CNN) models and conditional generative adversarial network (cGAN) models for this task. We observe the effects of using different loss functions and compare the colorizations obtained. The loss functions that were used in this project:

1. Absolute Error (L1) Loss:

$$\mathcal{L}_{\text{L1}} = |(y - \hat{y})| \tag{1}$$

2. Square Error (L2) Loss:

$$\mathcal{L}_{\text{L2}} = (y - \hat{y})^2 \tag{2}$$

3. Smooth Mean Square Error (Huber) Loss:

$$\mathcal{L}_{\text{huber}} = \begin{cases} \frac{1}{2}(y - \hat{y})^2, |y - \hat{y}| \leq \delta \\ \delta(|y - \hat{y}| - \frac{1}{2}\delta^2), |y - \hat{y}| > \delta \end{cases} \tag{3}$$

This is approximately equal to L2 loss for small prediction errors and approximately equal to L1 loss for large prediction errors.

CNN Model

The task is treated as a regression problem and a CNN based encoder decoder architecture is used to predict the color values for the pixels.

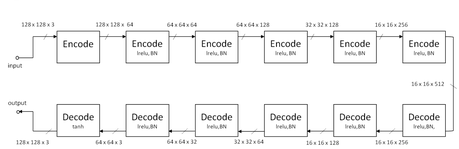

Figure 4: Encoder-Decoder Structure for the CNN Model

Conditional GAN Model

We use the conditional GAN model as proposed in [2] on our edges2shoes dataset. The generator and discriminator networks are shown in Figure 6 and 7. The generator model is based on an encoder-decoder architecture. The discriminator network functions as a classifier. In both the networks, a kernel size of 4 and a convolutional stride of 2 are used unless specified. The training model is shown in Figure 5

$$\mathcal{L}_{cGAN}(G,\ D) = \mathbb{E}_{x,y\sim p(data)}[logD(x,y)] + \mathbb{E}_{x\sim p(data)}[log(1 - D(x,\ G(x))]$$

In order to make the output images look similar to the target images, the models adds another loss to optimize the Generator model. In our experiments, we compare the model's performance using L1-loss, L2-loss, and Huber loss as explained in equations 1, 2, and 3.

$$G^* = arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda\mathcal{L}_L(G)$$

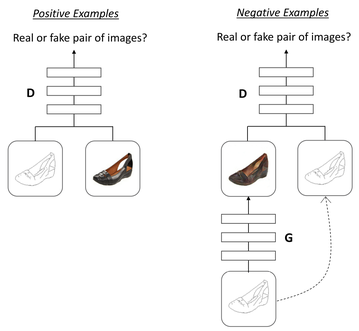

Figure 5: cGAN Training Model

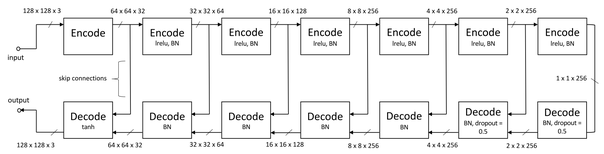

Figure 6: Generator Architecture

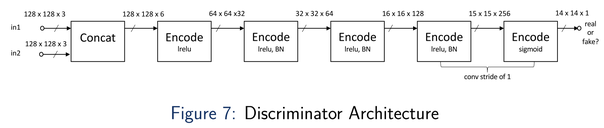

Figure 7: Discriminator Architecture

Qualitative Results

The following table displays some of the generated colorizations using the CNN and Conditional GAN based models :

Image source	Image 1	Image 2	Image 3	Image 4	Image 5	Image 6	Image 7	Image 8	Image 9	Image 10
Ground Truth										
Line Sketch										
GAN - L1										
GAN - L2										
GAN - Huber										
CNN - L1										
CNN - L2										
CNN - Huber										

Table 1: Generated colorizations

Quantitative Results

We use structural similarity (SSIM) as a measure of performance as it takes into account the contrast, luminance and structure of the images [3]. The SSIM of the generated colorizations are evaluated with respect to the original images. The relative SSIM is calculated by evaluating the increase in the SSIM values when it is evaluated using the line sketch images.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu^2{}_x + \mu^2{}_y + C_1) + (\sigma^2{}_x + \sigma^2{}_y + C_2)}$$

Model	**SSIM value (base: 0.5416)**	**Relative SSIM value**
CNN - L1	0.5814	0.1598
CNN - L2	0.5798	0.1566
CNN - Huber	0.5774	0.1519
GAN - L1	0.706	0.3035
GAN - L2	0.7037	0.2994
GAN - Huber	0.6855	0.2657

Table 2: Evaluation of SSIM values

It can be observed that the **GAN-L1 model** has the highest relative SSIM value.

Conclusions

- Images generated by the GAN model are quantitatively and qualitatively better than the images generated by the CNN models. This suggests that the GAN models are robust, even though training for these models is tricky.
- The outputs generated using L1 loss are more sharp while those generated using L2 loss are comparatively blurry.
- It can be observed that the coloring in both the CNN and GAN generated images are generally within the outline of the input images. This can be attributed to the uniformity in the orientation of the images in the dataset.
- In the case of sports shoes, the GAN gets confused by the presence of too many edges near the laces region. Hence, it fails to recognize them sharply and mixes it up with the background color patches.
- Since the *edges2shoes* dataset has classes and subclasses metadata available, we plan to learn a classification based colorization model to compare improvement across classes.

References

[1] A. Yu and K. Grauman, "Fine-grained visual comparisons with local learning," in *CVPR*, 2014.

[2] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *CVPR*, 2017.

[3] Z. Wang and E. P. Simoncelli, "Translation insensitive image similarity in complex wavelet domain," in *ICASSP*, 2005.