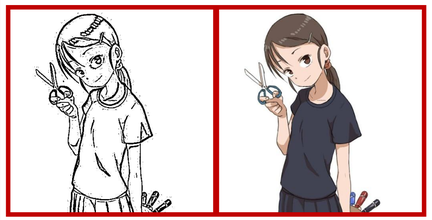
Automatic Colorization For Line Arts

Kaiyi Fu, Yang Wang, Baige Liu
Stanford University

Motivation

Normally, when animation artists try to create an animation frame, they first scratch the general skeleton of the content and fill in the color accordingly. The job of colorization is very onerous since a picture may have hundreds of partitions and each may require a different color. We are trying to create a tool that can automatically colorize the pictures, i.e., from left image to the right image in each pair.

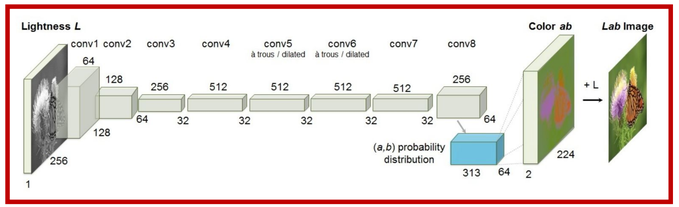


Related Work

Using deep learning strategies, researchers are trying to find methods to color an image from black and white images. An multinomial cross entropy loss is used here The Intensity may still provide some information about the color.

.

$$\mathrm{L}_{cl}(\widehat{\mathbf{Z}}, \mathbf{Z}) = -\sum_{h,w} v(\mathbf{Z}_{h,w}) \sum_{q} \mathbf{Z}_{h,w,q} \log(\widehat{\mathbf{Z}}_{h,w,q})$$

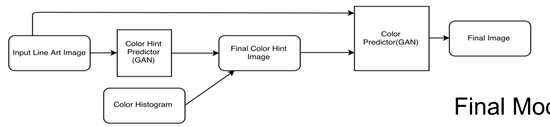


Our Algorithm

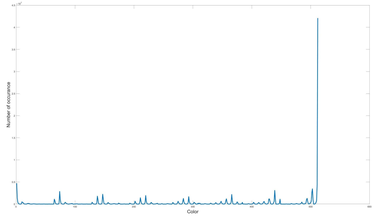
1. **Color histogram**
   In order to give some hint to the image, which acts like intensity, we sample the image and record the distribution of all possible colors. We distribute color space into 512 bins.
2. **Loss Function**
   We use L2 loss here to calculate the difference between the recovered and original images.

$$L = \sum_{i=1}^{m} \|\vec{c}_i - \vec{c}_i^0\|$$

3. **CNN**
   We trained a model using convolutional neural network with multiple convolution and RELU layers. We used L2 loss and Adam optimizer to train the neural network.
4. **c-Gan**
   We also trained a baseline model using convolutional GAN. The advantage of using GAN is that we can use adversarial loss, allowing the network to learn its own loss function.



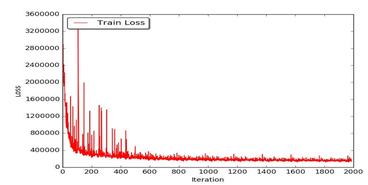
Final Model of our algorithm



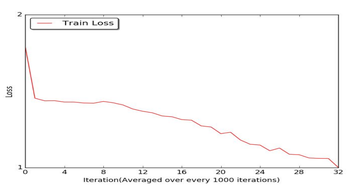
Data Acquisition

We downloaded 10,000 images from https://safebooru.org. All images are shaped to size 512x512. We used canny edge detector to detect edges. In such way, we create line-art images from original dataset.

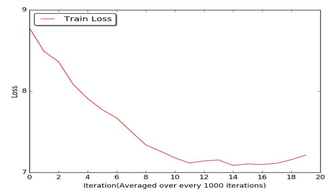
Experimental Results



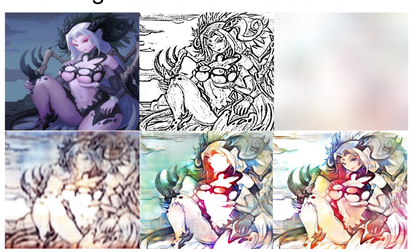
Training Loss for CNN baseline



Training Loss for discriminator in c-GAN



Training Loss for generator in c-GAN



In each figure, the first row from left to right: original image, line art image, color hint; the second row from left to right: CNN baseline colorization, c-GAN colorization, c-GAN with auto color hint colorization