Line Art Colorization with Concatenated Spatial Attention

Mingcheng Yuan  
Waseda University  
t-yuan@toki.waseda.jp

Edgar Simo-Serra  
Waseda University  
ess@waseda.jp

Abstract

Line art plays a fundamental role in illustration and design, and allows for iteratively polishing designs. However, as they lack color, they can have issues in conveying final designs. In this work, we propose an interactive colorization approach based on a conditional generative adversarial network that takes both the line art and color hints as inputs to produce a high-quality colorized image. Our approach is based on a U-net architecture with a multi-discriminator framework. We propose a Concatenation and Spatial Attention module that is able to generate more consistent and higher quality of line art colorization from user given hints. We evaluate on a large-scale illustration dataset and comparison with existing approaches corroborate the effectiveness of our approach.

1. Introduction

Line art colorization is a time-consuming process in 2D illustration, design, and animation. Although progress has been done in automating this process, currently it is still dominantly done manually given the inconsistency and low quality of automatic approaches. Most of these approaches are based on Generative Adversarial Networks (GAN) [2, 4, 5, 15, 16, 25], and there are still open issues on color consistency with user hints, color harmony of the final image, and low quality results on small and complicated areas such as eyes of characters.

In this paper, we design a new colorization model based on the conditional GAN framework and our proposed Concatenation and Spatial Attention module. Inspired by [13, 28], our proposed module is able to emphasize important features and focus on high-level consistency across the image by using complex features extracted from the user hints. We employ a U-Net [21] inspired architecture that is able to preserve low-level information and multiple discriminators at different scales to increase the robustness of our model to varied inputs. Our model is trained with adversarial losses, a feature matching loss, and perceptual loss to obtain higher quality results.

2. Related Work

Traditional colorization approaches have been optimization-based [20, 24], using features taken from the image and enforcing smoothness terms and consistency with the user inputs. These approaches require user input and are inflexible when dealing with complex line drawings, which has lead to an increase of research in data-driven approaches.

In recent years, Generative Adversarial Networks [6] have gained popularity in conditional image generation tasks, including line art colorization. Frans [4] and Auto-painter [16] showed that GAN can achieve better results for line art colorization task when compared to traditional methods. Ci et al. [2] achieved high-quality user-guided line art colorization and overcame the problem of overfitting to synthetic line art. Furusawa et al. [5] and Lee et al. [15] proposed a simpler and more user-friendly way of user-guided line art colorization where reference images are used as user hints. Tag2Pix [13] utilizes their SECat module to generate illustrations with quality details using text tags as user hints. Our approach focuses on using user hints in the form of color patches and dots, which gives more flexibility and control to the user.

In line art colorization, the color hints are usually concatenated with the line art and encoded as the input of the networks [2, 25]. Kim et al. [13] proposed the SECat module which is inspired by the SENet [9] and StyleGAN [12], and encodes the color hint information for the mid-level blocks, achieving detailed colorization and high color consistency. However, the SECat[13] module is made for encoding color hint information from text tags instead of actual color hints such as color patches and scribbles. Our approach builds upon SECat and incorporates concepts of spatial attention [28] to be applicable to color hints and obtain higher quality and more consistent results.
3. Proposed Approach

3.1. Concatenation and Spatial Attention

Our Concatenation and Spatial Attention block builds upon the SECat block [13], but allows preserving spatial features of the color hints through spatial attention. The hints are first processed by a pre-trained VGG19 model [23], and the features from the conv4_4 layer without the ReLU activation function are processed by three additional convolutional layers. Afterwards, they are concatenated with the processed input from the Resnet [7] block and convolved with two convolutional layers with a ReLU and Sigmoid activation function, respectively, to obtain the attention map. This attention map is multiplied element-wise with the processed input and added to the input. The entire block is shown in Figure 1 and is used throughout the model.

3.2. Network Structure

An overview of our approach is shown in Figure 2. The structure of the generator is based on the U-Net [21] model, aiming to preserve the low-level information such as the location of important edges in the line art, and to generate colors with quality details. At the input of the generator network, the line art image and color hint image are concatenated and then transformed to feature maps via a convolutional layer. The feature maps are downsampled 4 times before reaching the mid-level of the network. There are 8 Concatenation and Spatial Attention blocks in the mid-level where the feature maps from the previous block are processed and concatenated with the extracted features of the color hint, before computing the spatial attention. Then the spatial attention maps are multiplied with the processed input feature maps and added to the input feature maps. The output feature maps from the mid-level are then upsampled with transposed convolutional layers to produce the final colorized illustration.

The discriminator is adopted from the multi-discriminator framework by Wang et al. [26], where 3 discriminators have identical structure, but take input images of different resolutions to encourage the generator to generate results with more details. Each discriminator is a PatchGAN [10] model that classifies whether a 70 × 70 pixel patch is real or fake.

3.3. Loss Function

We train our model with a combination of adversarial, feature matching, and perceptual losses to obtain high-quality results.

The adversarial loss for a conditional GAN given a generator $G$, a discriminator $D$, and dataset $\rho_{\text{data}}$ is as follows:

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

(1)

where $x$ is a line art image, $y$ is a color image, and $h$ is the user hints. In particular, we use 3 different discriminators $D_i$ computed at the original resolution, half the original resolution, and on quarter of the original resolution.

We also incorporate the discriminator-based feature matching loss proposed by Wang et al. [26] that is defined as:

$$
\mathcal{L}_{FM}(G, D_i) = \mathbb{E}_{(x,y)} \sum_{j=1}^{T} \frac{1}{N_j} \|D_i^{(j)}(x, y) - D_i^{(j)}(x, G(x, h))\|_1
$$

(2)

where $T$ is the number of layers in the discriminator $D_i$, $N_j$ are the number of elements in $j$-th layer of the discriminator, and $D_i^{(j)}$ is the extracted feature maps by the $j$-th layer of discriminator $D_i$. The $D_i$ here is only used for extracting the feature maps and does not try to maximize the feature matching loss $\mathcal{L}_{FM}(G, D_i)$.

Utilizing perceptual losses [11] from pre-trained networks can slightly increase the performance of the results [26] and has also proven to be beneficial for the training of line art colorization models [2, 13]. We use a perceptual loss computed with a VGG19 network [23] pre-trained on ImageNet [3] as the content loss for the generator as:

$$
\mathcal{L}_{perc}(G) = \sum_{k=1}^{N} \frac{1}{M_k} \|F^{(k)}(y) - F^{(k)}(G(x, h))\|_1
$$

(3)

where the $M_k$ is the number of elements in the $k$-th layer of VGG19, $F^{(k)}$ is the feature maps by the $k$-th layer.

The final objective function for our model becomes:

$$
G^* = \arg \min_G \left( \max_D \sum_{i=1}^{3} \mathcal{L}_{cGAN}(G, D_i) + \lambda \sum_{i=1}^{3} (\mathcal{L}_{FM}(G, D_i) + \mathcal{L}_{perc}(G)) \right)
$$

(4)
4. Experiments

4.1. Dataset

For our experiments we use a dataset consisting of 1,299,232 training images, 12,422 validation images, and 12,946 testing images, which are taken from the large-scale illustration dataset Danbooru2019[1]. We filter out the greyscale images within the dataset using tags such as “greyscale” and “monochrome”.

We use the illustration as the ground truth and automatically extract line art using 3 different extraction methods to minimize the amount of overfitting. In particular, we use XDoG [27], sketchKeras [17], and Sketch Simplification [22]. An example of the generated data is shown in Figure 3, where we can see that the line art generated by XDoG have sharp and clear edges, with large amounts of noise and artifacts. Line art extracted using sketchKeras and Sketch Simplification are very close to digital line art with a consistent line thickness. As the line thickness generated by Sketch Simplification depends on the input resolution of the image, we first pre-process the image by enlarging it to 3 times the original size using waifu2x [19], generate the line art, then resize it back to the original resolution. The type of sketch is randomly chosen in each iteration during the training, and the probability is \( p_1 = 0.1, p_2 = 0.5, p_3 = 0.4 \) for the XDoG, sketchKeras, or Sketch Simplification to be chosen.

We follow the approach of Zhang et al. [29] for simulating user generated color hints. The locations of the points are determined by a 2D gaussian which \( \Sigma = \text{diag}([(H/4)^2, (W/4)^2]) \) and \( \mu = 1/2[H, W]^T \), where the \( H \) and \( W \) are the height and width of the input image. The number of points/patches being sampled is determined by a geometric distribution where \( p = 0.125 \), and the size of the points/patches is chosen randomly from \( 1 \times 1 \) to \( 8 \times 8 \) with equal probability. The color of each sampled patch is the average color within the area of the patch.

4.2. Training Details

We use Adam [14] to train the model with momentum hyper-parameters \( \beta_1 = 0.5, \beta_2 = 0.999 \), and the learning rate set to 0.0002. The model is trained on 8 NVIDIA 1080Ti
GPUs with a batch size of 32. Both the generator and the discriminators are updated once in every iteration.

All the input images are randomly cropped to $512 \times 512$ and randomly flipped horizontally.

### 4.3. Comparison with Existing Approaches

We compare our proposed approach with that of Wang et al. [26], Ci et al. [2], and evaluate using the Fréchet Inception Distance (FID) [8]. The FID is adopted due to its robustness to noise and the sensitivity to mode dropping [18], which can test whether a model can generate diverse and quality results. FID measures the similarity between two sets of images, a low FID score means the images in the two sets are very similar. In particular, we compute the FID score between the real illustration images in the test dataset, and the fake images generated with fixed color hints. All models are trained using the same training data for a fair comparison.

As shown in Table 1, our model achieves the lowest FID score compared to Wang et al. [26] and Ci et al. [2]. Removing the proposed module from the mid-level ResNet blocks results in significantly worse performance.

### 4.4. Qualitative Evaluation

We show results in Figure 4. We can see that as seen by the difference in FID score, our results show higher quality details and more consistency with the user hints compared to existing methods. Furthermore, we can see less color bleeding which is a common issue of deep learning based colorization approaches.

### 5. Conclusions

In this paper, we proposed a conditional GAN model for line art colorization that can produce high quality colorized illustrations from line drawings and user hints. Our model is based on our proposed Concatenation and Spatial Attention block which allows the model to focus on important features and harmonizes the output illustration with the color hints. Evaluation on a large-scale dataset shows that our approach is able to significantly outperform existing approaches both quantitatively and qualitatively. Future work includes adapting the model to more types of user input and improving the model to work on higher resolution images.
References


