Attentioned Deep Paint

Automatic Sketch Colorization Using Generative Adversarial Network (GAN)

Team D

Taebum Kim, Hyunseok Oh, Dongjae Lee, and Hyunwoo Kang of NCSOFT

Introduction

Generative Adversarial Network (GAN) is one of the most famous deep learning technique in Computer Vision. It generally has two model, generator and discriminator. Generator generates realistic fake output by deceiving discriminator, which make a decision of whether given image is real or fake.

Among various applications of GAN in Computer Vision area, we focus on Image Translation. In Image Translation, certain domain of images are translated into the other domain. One of that is sketch colorization. It maps sketch image into colorized image. In this project, we construct our novel model and conduct an experiment of automatic animation sketch image colorization using target reference image. Our contributions are

- More **practical** with 512 x 512 image size domain
- More **realistic** colorization
- **Totally automatic** colorization process (Absence of users' additional work)

Proposed Method

We propose our colorization method as follows. When feeding input into generator, sketch image and 4 color histogram images (extracted from reference image) are concatenated. Thus, generator gets 512 x 512 x 15 input image and generates 512 x 512 x 3 output image.

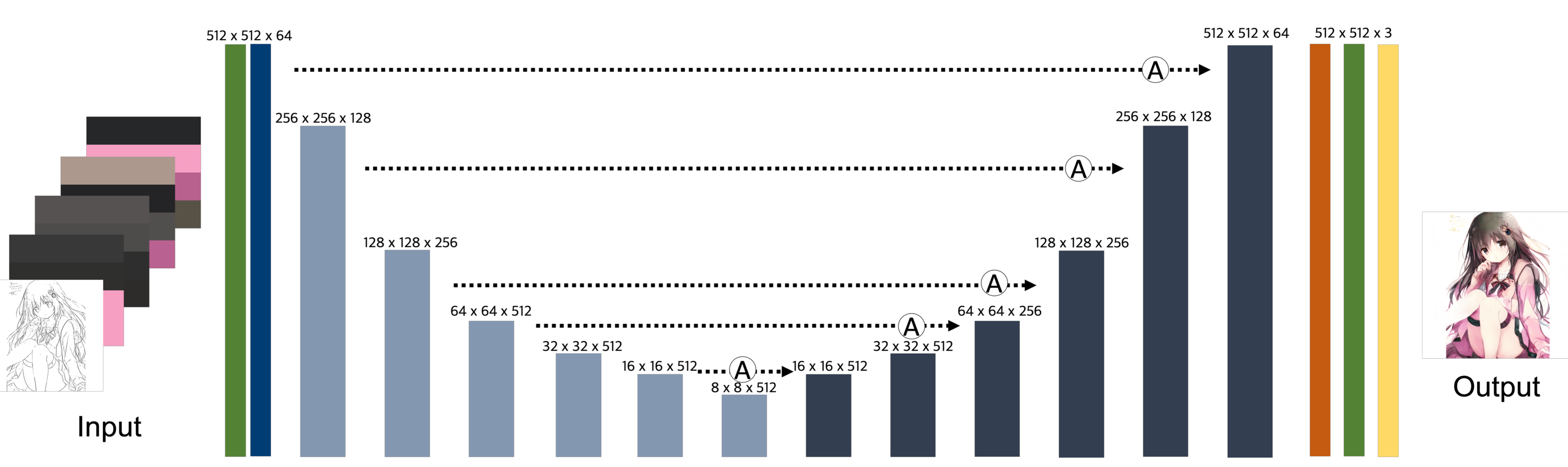
At the same time, discriminator gets both real image and generated fake image. After that, it determines which is real and which is not. For stable training, we use image pooling, which feeds fake image into discriminator among past 50 fake images, not latest one.

Let's denote generator as G, discriminator as D, color histogram extractor as C, sketch image as x, and original colorized image as y. Then training objective becomes

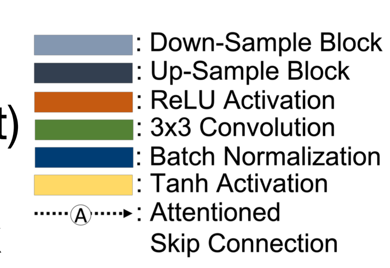
$$arg \min_G \max_D \mathcal{L}_{GAN}(G, D) + \lambda \mathcal{L}_1(G)$$

, where $\mathcal{L}_{GAN} = \mathbb{E}_{x,y}[\log(1 - D(x, G(x, C(y))))] + \mathbb{E}_{x,y}[\log(D(x, y))]$, λ=weight hyperparameter, and $\mathcal{L}_1 = \mathbb{E}_{x,y}[\|y - G(x, C(y))\|_1]$.

Generator Architecture



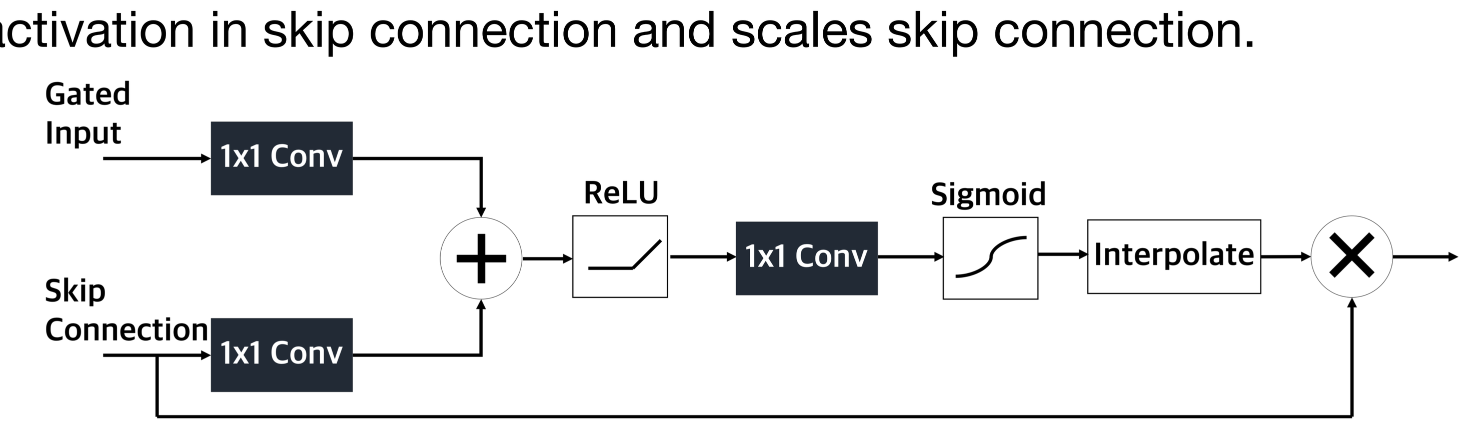
- Sketch is concatenated with color histogram of reference
- UNet with pooling & un-pooling, not deconvolution (SegUNet)
- Discriminator is 70x70 PatchGAN (Most clear output)
- The innermost feature maps are fed into attention gate block



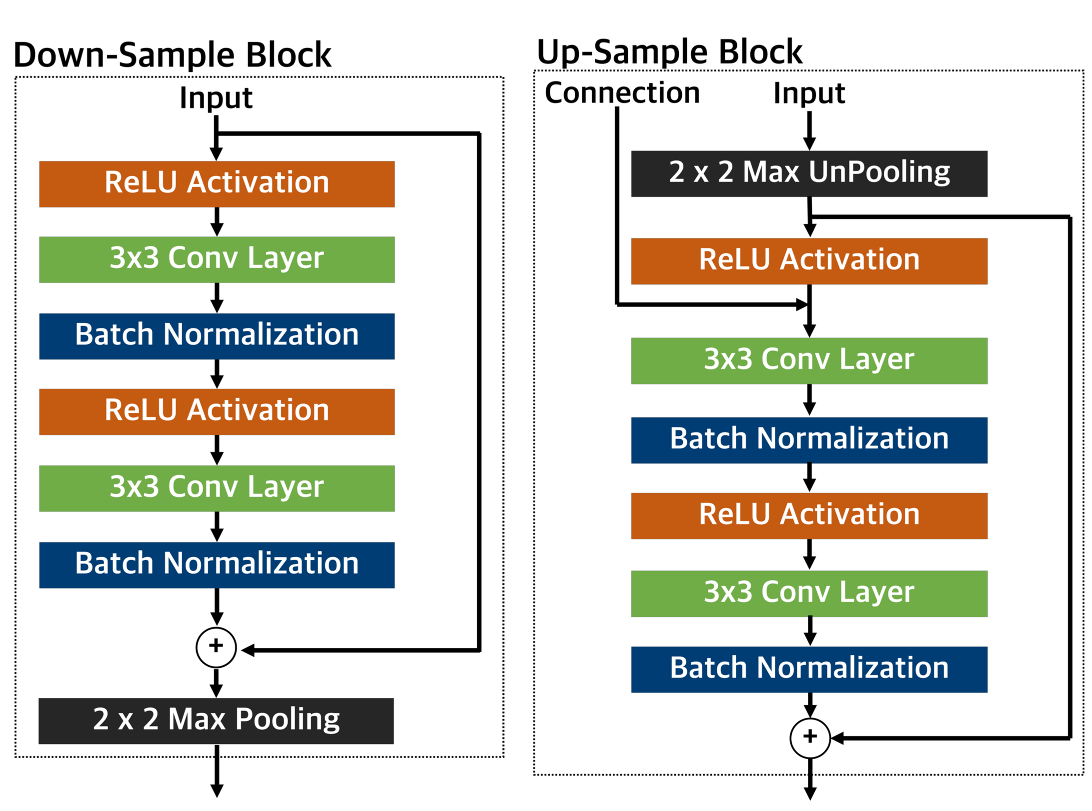
Attention Mechanism

Attention was proposed in natural language processing (NLP). It enhances Encoder-Decoder network's performance by focusing important context (not all context) of feature map at decoding stage.

In our experiment, we apply attention in skip connection. When concatenating skip connection in decoder, attention gate calculates weight of activation in skip connection and scales skip connection.



Generator Building Block

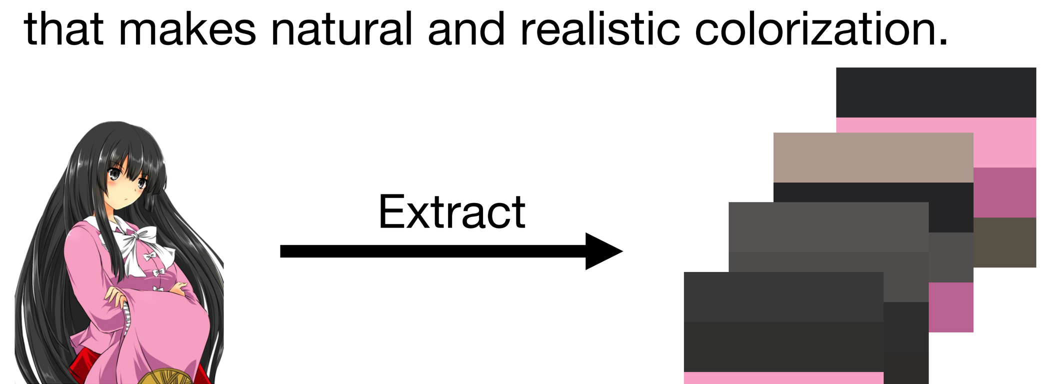


- Both blocks are residual block
- Skip-Connections are concatenated in Up-Sample Block

Color Histogram

For reference images, we crop image into 4 regions vertically. After that, we extract top-4 colors for each region. Using 16 extracted colors, we make 4 RGB images. The i-th image contains top-i color of each cropped region.

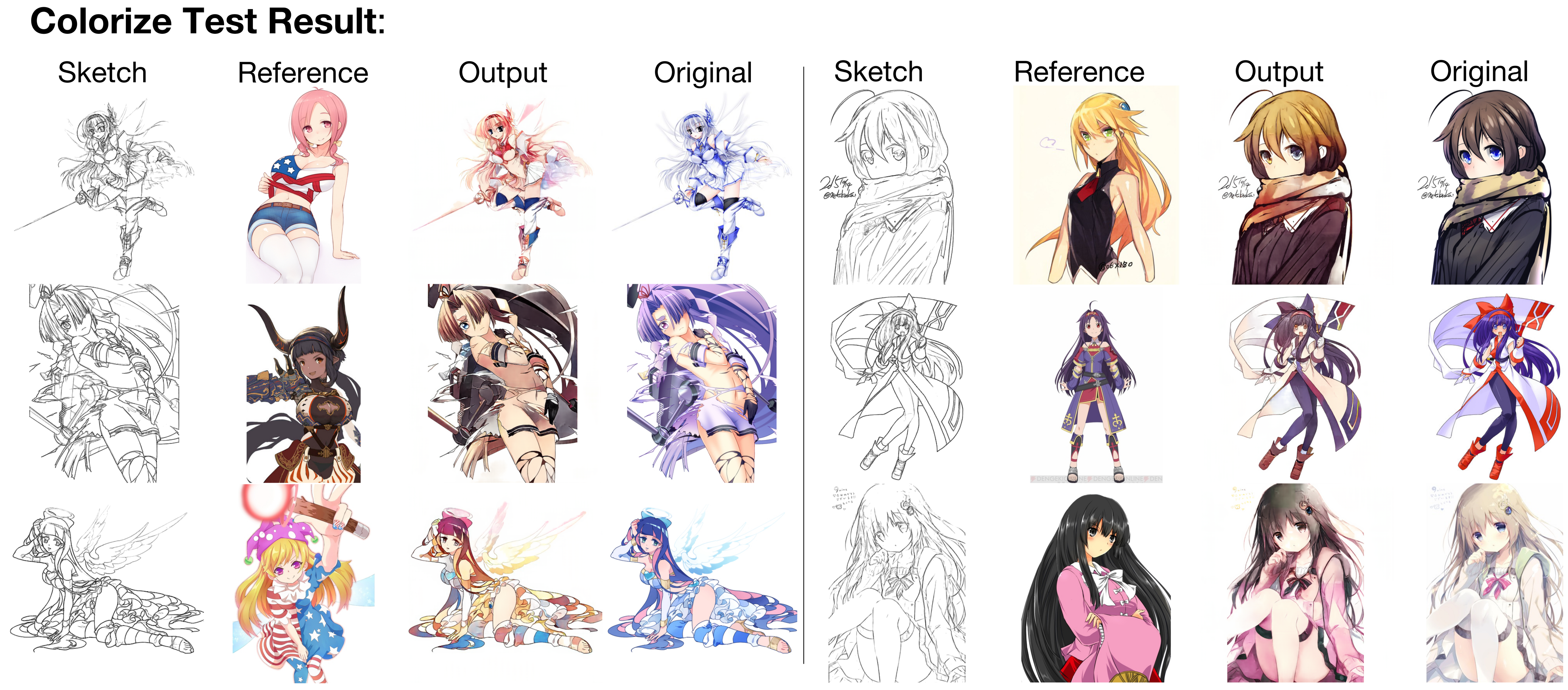
When the model is trained with color histogram information, we expect that it learns how to merge given regional color information properly that makes natural and realistic colorization.



Result

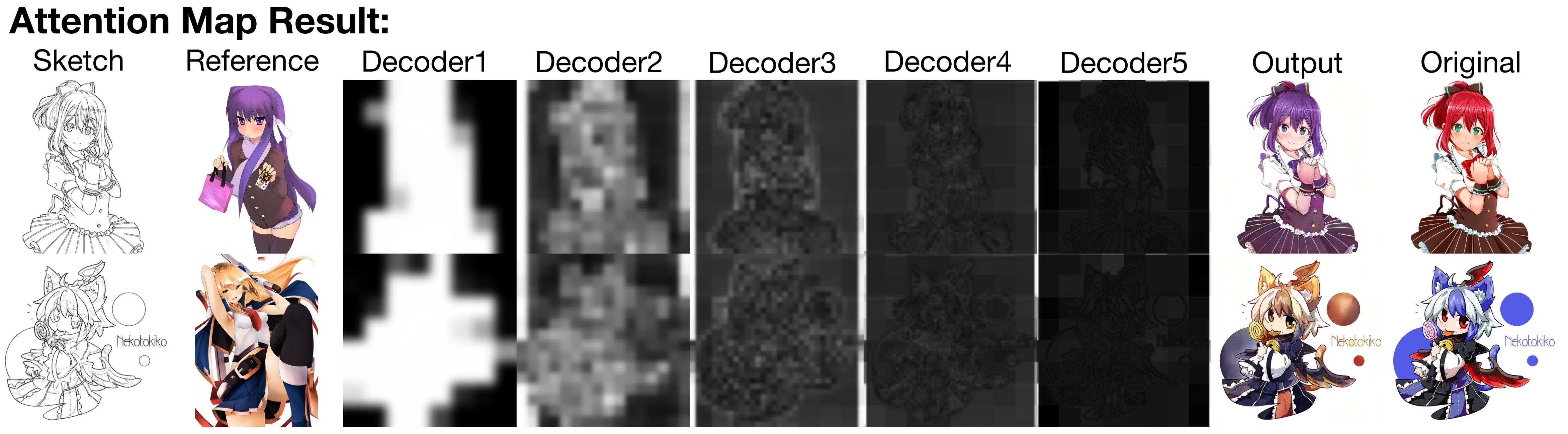
Dataset: 512 x 512 animation images in Danbooru 2017 dataset. Among them, we select image of single person whose background color is white. Based on these colorized image, we extract sketch using sketchKeras model. Finally, we make sketch – color image paired dataset and corresponding color histogram information in json format. Whole dataset are split into 14224 training, 3545 test images.

Colorize Test Result:



(Original images are just for comparison, not related to test)

Attention Map Result:



- Deeper decoder has much more high attention in overall colorize target region
- Outer decoders have weaker attention than inner decoders, but focused on edge of image