

Image Colourization

Saurabh Bazari
160101061

Namit Kumar
160101039

Abhinav Mishra
160101005,

Ritik Agrawal
160101055

G Sharath Kumar
150101023

Image Colorization using Generative Adversarial Network:

Implementation:

Generative Adversarial Network (GAN) is a generative model used to get colorized images from the grayscale images. It mainly consists of 2 parts: Generator and Discriminator. The paper proposes models for both parts. The generator tries to create colorized images that are similar to the ground truth image whereas the discriminator tries to differentiate between the ground truth and the image generated by the generator. This way both the parts are trained simultaneously. Finally, the output of the generator is the required colorized image.

Both the generator and the discriminator are Convolutional Neural Networks and follow the multilayer perceptron model. The architecture of the models are discussed below -

Implemented Improvement :

- We added ResNet block in the generator which helps us to identify the high-level features of the image. This will help in better isolation of objects which finally leads to more accurate colorization.

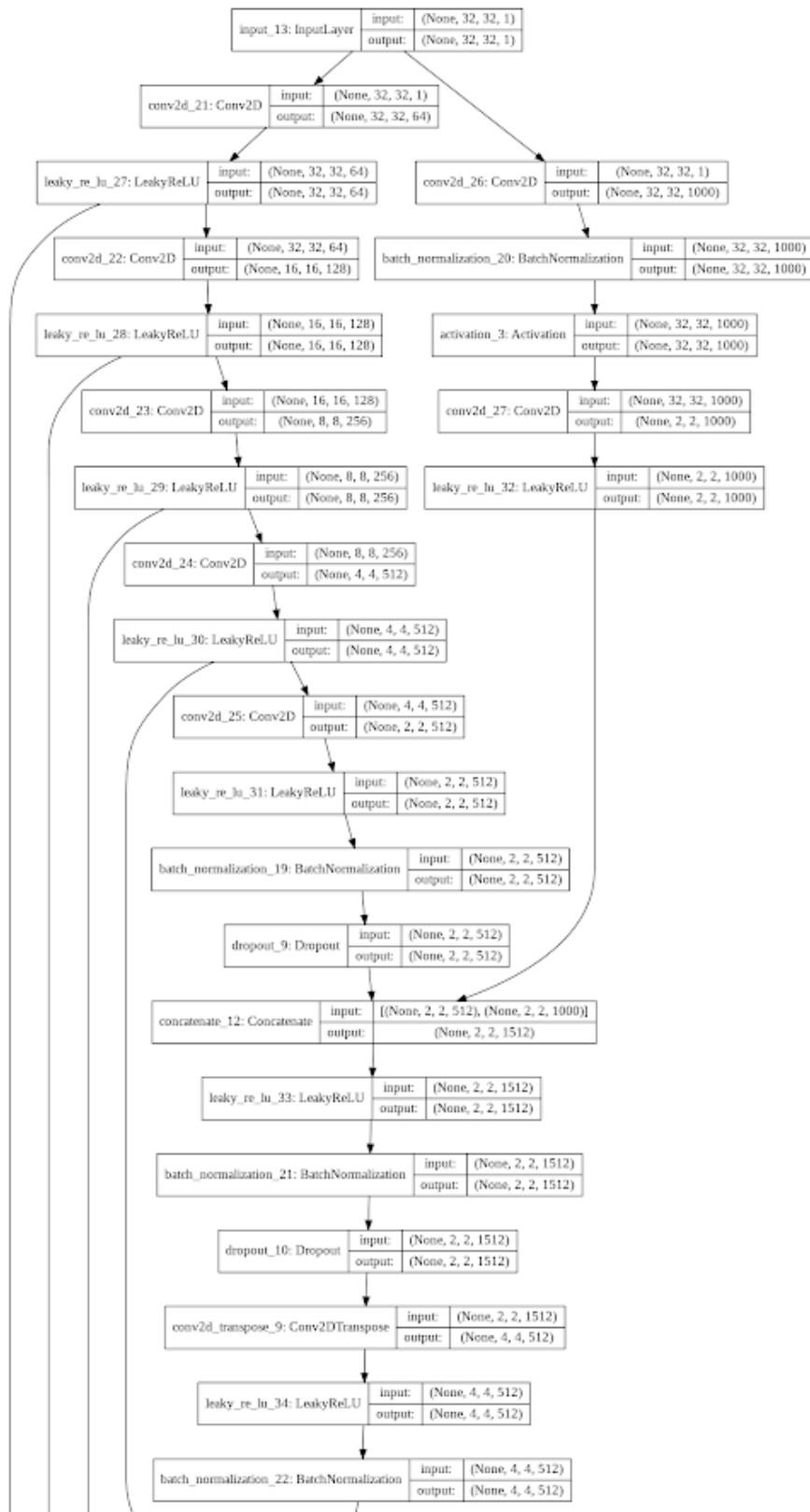
Generator :

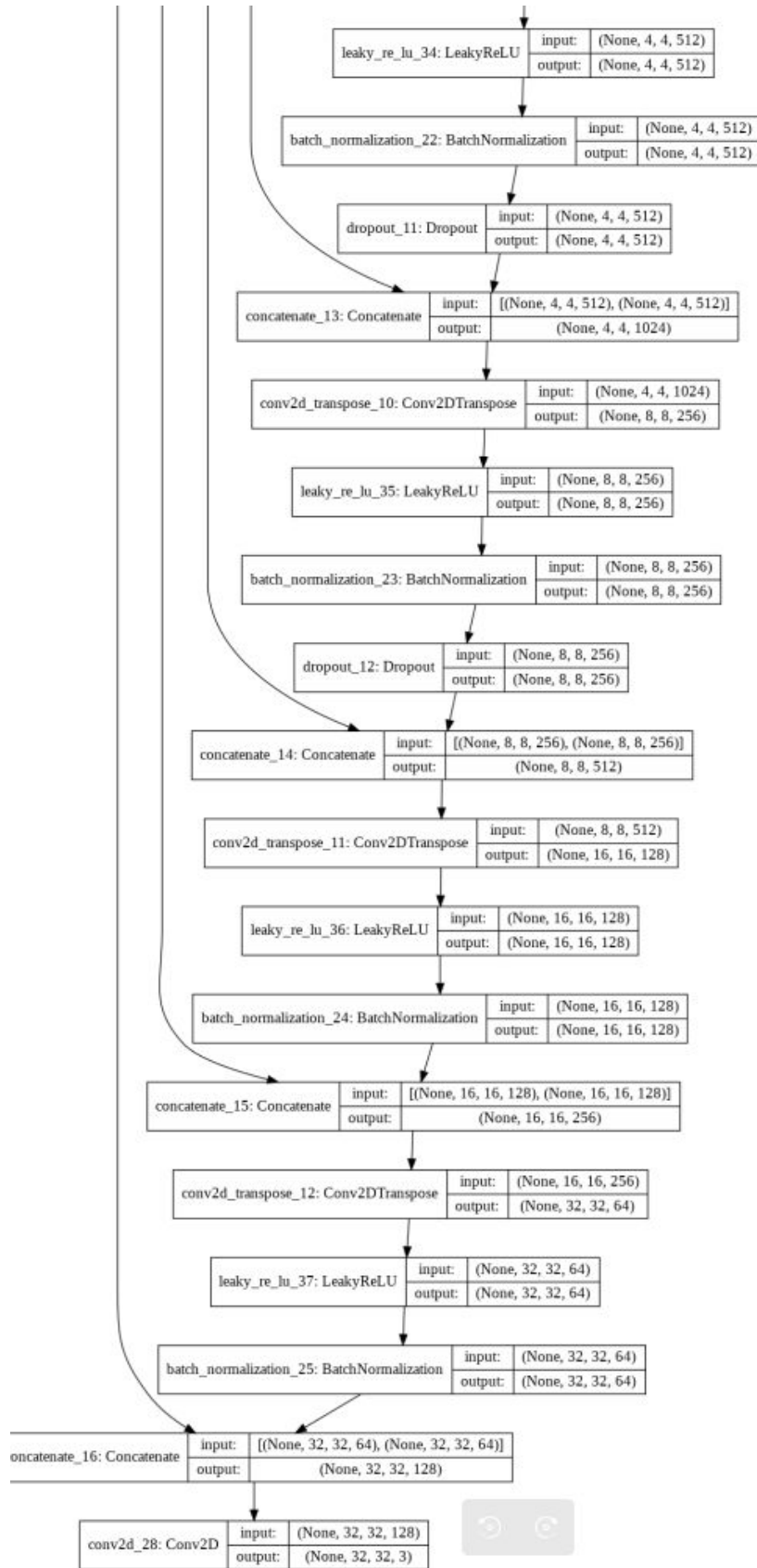
- Input to the generator was required to be in 32 X 32 size. The baseline model is UNet.
- Downsampling is applied at second, third, fourth and fifth layers with stride 2 to reduce the number of computations required. The 32 X 32 size input is reduced to 512 X 2 X 2. It uses the ReLU activation function in the process.
- Resnet block of 1000 X 2 X 2 is added
- The input to the decoder segment becomes of the shape 1512 X 2 X 2.
- Upsampling with stride 2 at each layer is used to get the final colorized image. Except for the last layer, which uses the Tanh activation function, other layers use the ReLU activation function.
- Batch Normalization is used to prevent the generator from generating similar outputs for all inputs which are able to fool the discriminator.

Discriminator:

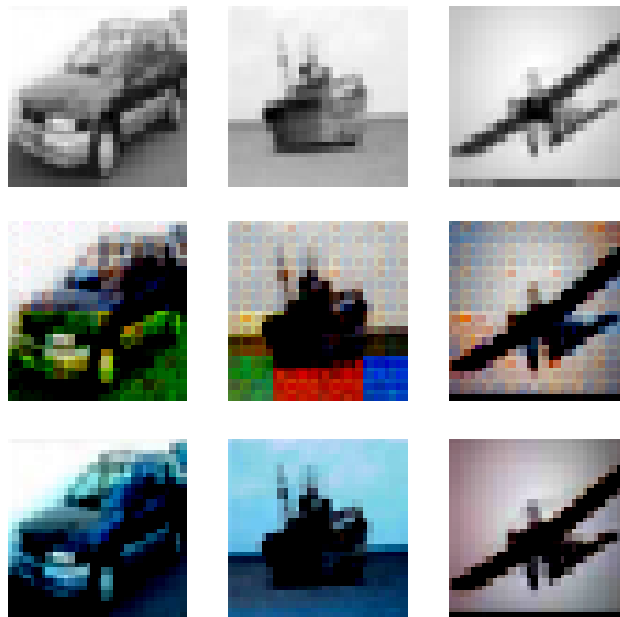
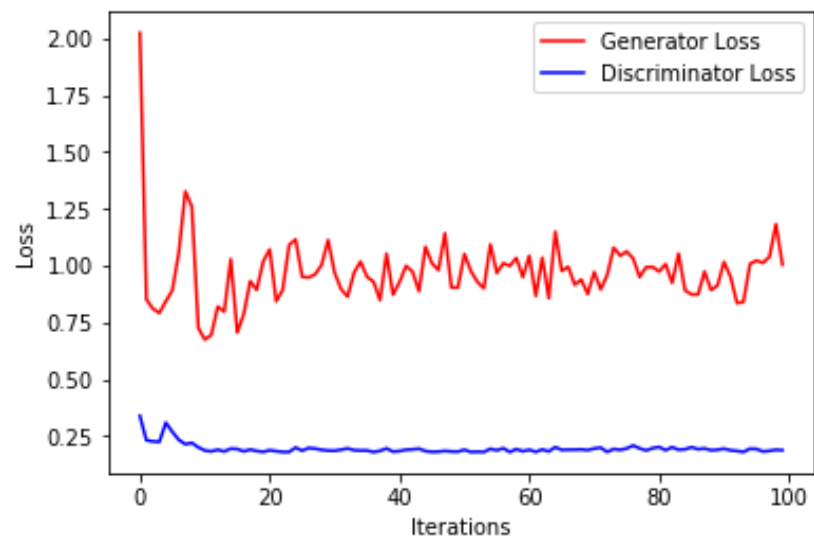
- The input of the discriminator is 2 colorized images of 32 X 32 X 3 size.
- Downsampling is applied on each layer with stride 2 which finally gives the probability value of the input being real or fake. Except for the last layer, which uses the Sigmoid activation function, other layers use the ReLU activation function.

New Model Architecture:





Result before improvement:



Results after improvement:

