BLACK AND WHITE IMAGE COLORIZATION USING GENERATIVE ADVERSARIAL NETWORKS

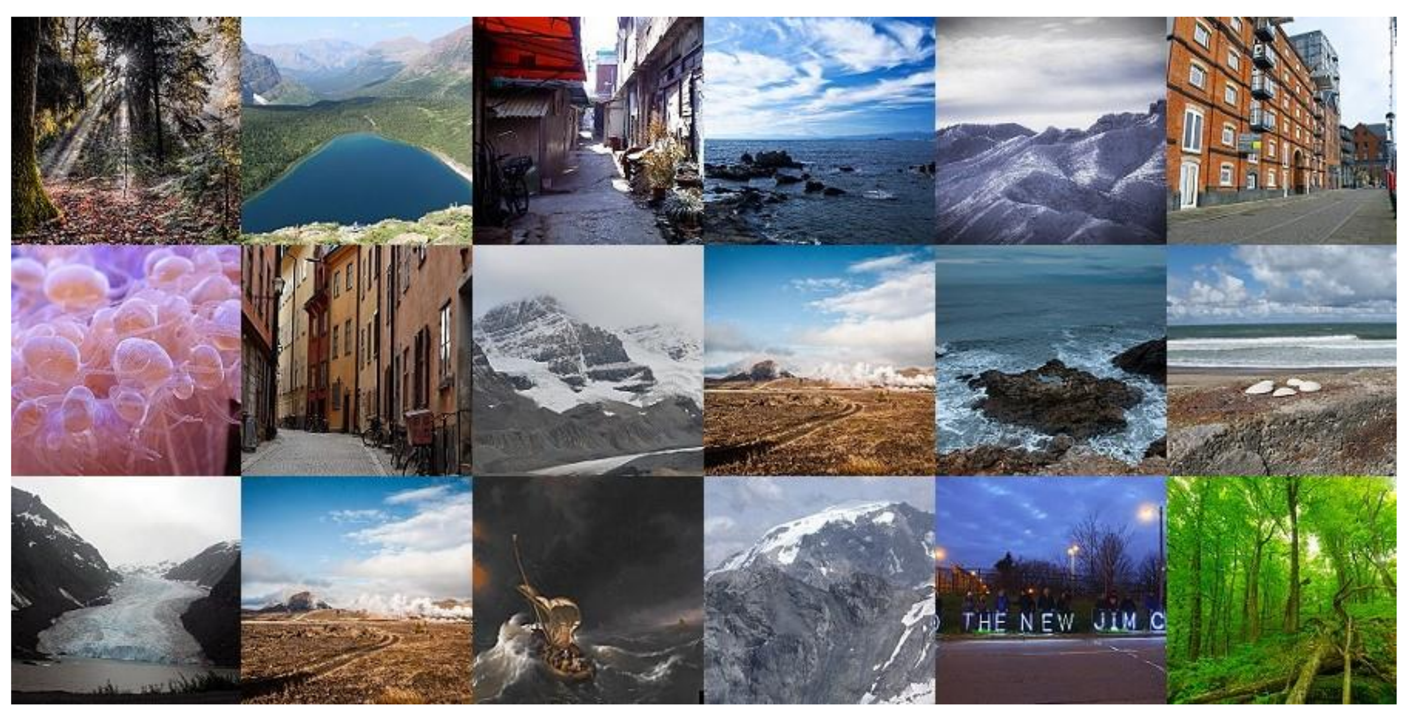
191180034 Umut DEMİREL, 191180090 Çağla YÜCE

Problem Statement

Colorizing black and white photographs is a complex, time-consuming and laborious process. Deep learning techniques greatly accelerate the colorization of these photographs. Nevertheless, various factors such as dataset availability, colorization accuracy, and computational efficiency must be considered when implementing these systems. To overcome these challenges in colorizing black-and-white photographs, we applied a deep learning approach using GANs.

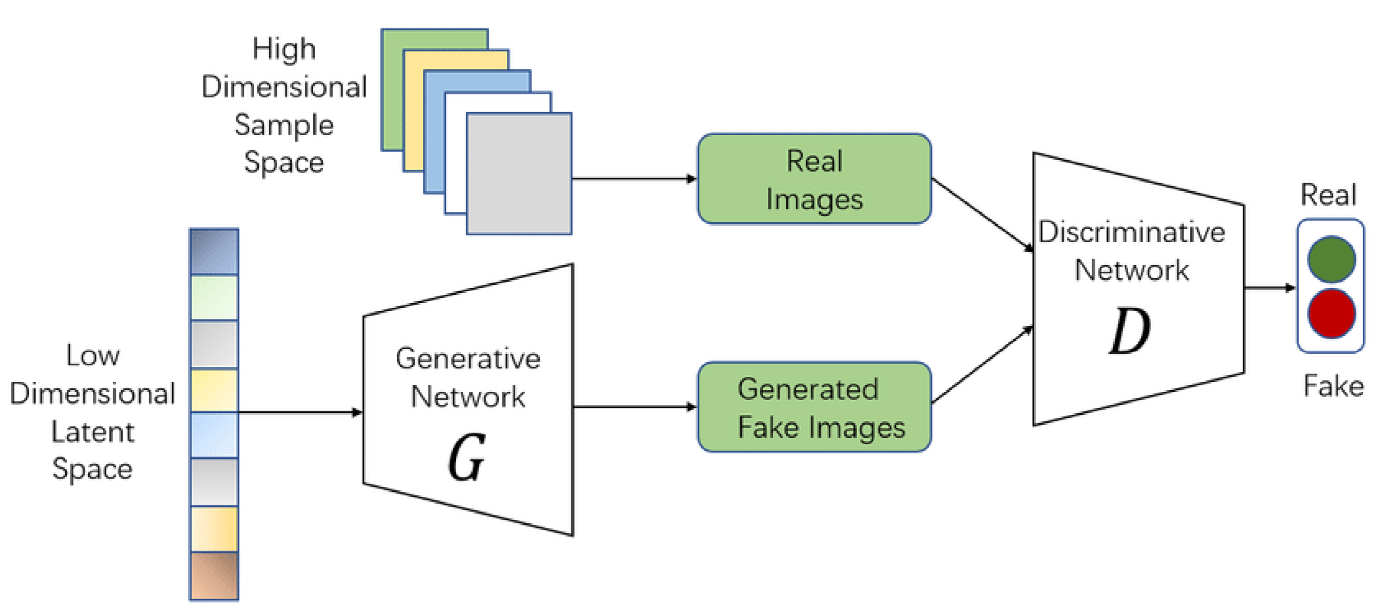
Dataset Generation

- The dataset was based on the MIRFLICKR [1] dataset.
- It comprises 3000 RGB images predominantly from nature and street photography.
- Images were resized to a uniform 120x120 resolution.
- Both RGB and grayscale images underwent normalization.
- A shuffle process was implemented.
- The dataset was divided into a training set (90%) and a test set (10%).
- A batch size of 64 images was chosen.



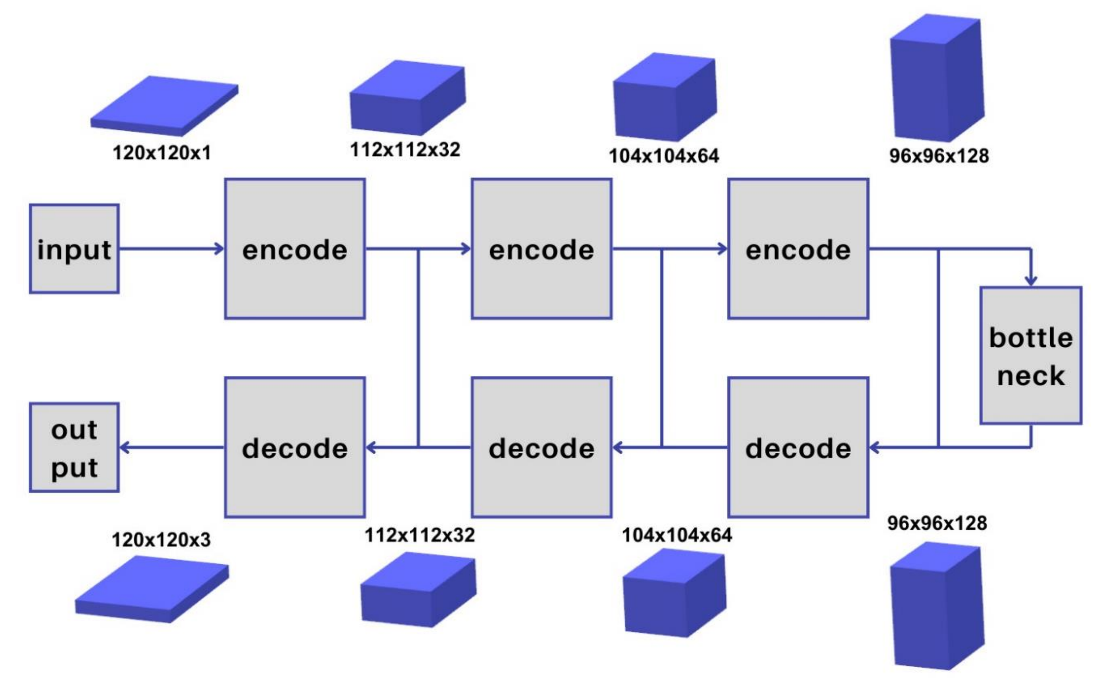
Example images from the dataset

GAN Architecture



GAN architecture

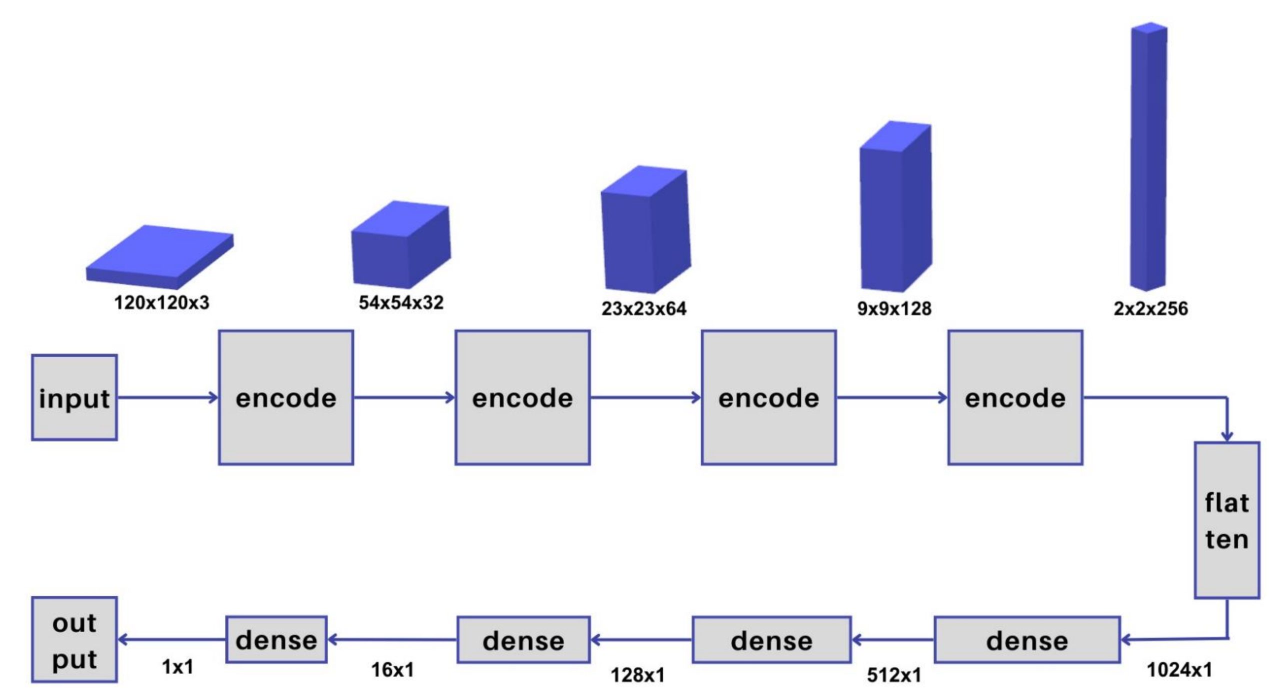
Generator Architecture



Generator architecture

The generator takes a grayscale image and produces a color image. Generator architecture was developed based on "U-Net" [2]. The generator consists of 3 encoder blocks and 3 decoder blocks. Encoder layers are connected to corresponding layers of the decoder. The bottleneck layer between the encoder and the decoder compresses the encoded information into a lower-dimensional representation. Mean squared error was chosen as the generator loss function, and Adam optimizer was chosen as the generator optimizer.

Discriminator Architecture

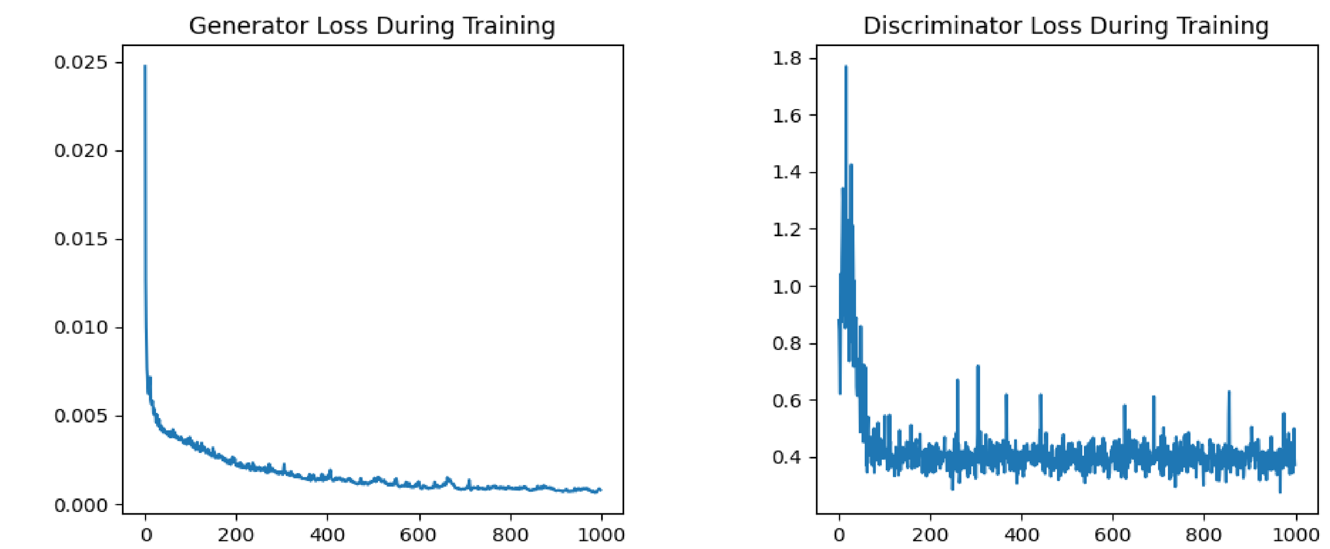


Discriminator architecture

Discriminator takes an RGB image and calculates the probability of this image being real. Discriminator should output 1 for real and 0 for fake images. Discriminator consists of 4 encoder blocks and 4 dense layers. A flatten layer is used to prepare the multidimensional output for the dense layer. The last dense layer has a point output and a sigmoid activation function. Binary cross entropy was chosen as the discriminator loss function and adam optimizer was chosen as the discriminator optimizer.

Training Results

The GAN model was trained for 1000 epochs on the training data set and the following training errors were obtained.



Generator and discriminator losses during training

Experimental Results

Example images from the test results

- Defects in the outputs are associated with the limitation of training time and insufficient complexity of the model.
- The blue color, which has a high frequency of occurrence in the images in the data set, has become dominant on the outputs.
- Objects with various colors in the images in the data set are colored according to the color these objects have most frequently.
- Objects such as animals and flowers, which are not common in the images in the data set, are insufficiently colored.

References

1. Michael S. Lew (2008). "MIRFLICKR-25000". In: https://press.liacs.nl/mirflickr/mirdownload.html
2. Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3431-3440. doi:10.1109/CVPR.2015.7298965