Image Dehazing and Color Reconstruction Using Pix2Pix GAN

DHYEY ITALIYA - 2021A7PS1463P
ABIR ABHYANKAR - 2021A7PS0523P

April 2024

1 Choice of Model

Pix2Pix GANs[2] have proven to be particularly useful in image dehazing tasks due to their ability to learn complex transformations from paired image data. Let's delve into why Pix2Pix GANs are well-suited for image dehazing:

1.1 Complex Non-linear Transformations

Image dehazing involves removing or reducing the effects of atmospheric haze from images, which is a complex non-linear transformation. Pix2Pix GANs excel at learning such complex transformations because they can capture both global and local features of the images through their encoder-decoder architecture based on U-Net. This enables the model to understand and remove haze while preserving important details and structures in the image.

1.2 Context-Aware Transformations

Haze often varies spatially across an image, and Pix2Pix's conditional GAN architecture allows the model to be context-aware. By conditioning the generator on the input hazy image, the Pix2Pix model can generate a dehazed image that takes into account the spatial context and structure of the original image. This helps in producing more realistic and visually pleasing dehazed images.

1.3 Generator

The generator we used in Pix2Pix is based on a U-Net architecture. U-Net is known for its ability to capture both global and local features of images through its encoder-decoder structure.

1.4 Discriminator

We have used the PatchGAN discriminator as focuses on classifying local image patches as real or fake rather than classifying the entire image. This approach allows the discriminator to provide fine-grained feedback on the realism of local image regions, making it well-suited for image-to-image translation tasks.

2 Loss functions

2.1 Adversarial Loss

We use binary cross-entropy loss to train the generator to produce images that are indistinguishable from real images according to the discriminator. We tried out Wasserstein Loss but we received unsatisfactory results.The model's loss was explosive. Hence, we continued with Binary-Cross-Entropy.

2.2 L1 Loss

L1 loss is used to measure the pixel-wise difference between the generated and target images. This encourages the generator to produce images that closely resemble the target images.

2.3 Color Loss

Hazy images usually lack brightness and contrast, to improve these lacking features we define a Color Loss, inspired by Enlightengan[4] to measure color difference between the haze-free images and the reconstructed images.

This loss function forces the generator to generate images with the same color distribution as the haze free images.

2.4 Perceptual Loss

In order to handle construction of high level feature in areas of dense haze, inspired by Cycle-dehaze[1]. Perceptual loss aims to minimize the perceptual difference between the generated and target images by comparing their feature representations in a pre-trained deep neural network. The idea is to encourage the generator to produce images that not only fool the discriminator but also resemble the target images in terms of their high-level features.

3 Training

For training we need two sets of training datasets: trainA includes hazy images and trainB includes ground truth images. The names of corresponding images must be same. We are providing a python file for copying the images in the desired directory structure. The same structure is required for validation phase as well. In total, we require a file in the given structure:

```
local_dataset
├─ trainA
├─ testA
├─ trainB
└─ testB
```

We are providing a python file for copying the dataset to the required format.

We opted for Adam optimizer (momentum = 0.5) with batch size of 1. Our initial learning rate was 0.0002 for the first 20 epochs, with linear decay to zero over the next 20 epochs. We implemented our model in PyTorch using two NVIDIA T4 GPUs available on Google Colab and trained our network for 40 epochs. After 40 epochs, the model starts perform worse on validation set.

4 Quality Measures

4.1 PSNR

It measures the ratio between the maximum possible value of a signal and the power of distorting noise that affects the quality of its representation. The higher the PSNR, the more effective the reconstruction method is.

4.2 SSIM

It is a Structural Similarity Index which is a perceptual metric that quantifies image quality degradation caused by processing. In this measurement, image degradation is considered as the change of perception in structural information.

5 Metrics

Training Time	8408 seconds
SSIM	0.8470
PSNR	24.4623

6 Trials

We tried the use of simple conditional GANs as well as Cycle GAN for dehazing but they did not give satisfactory results and we moved to the use of Pix2Pix GAN. The dehazed colors and lack of structure made us to use the color loss and Perceptual loss respectively.

We thought of using patches of images for calculating the loss (Local Loss) in case of spatially varying Hazy images similar to Enhanced Cycle GAN[3], but it gave unsatisfactory results due to patches of noise in the image.

References

[1] D. Engin, A. Genç, and H. K. Ekenel, *Cycle-dehaze: Enhanced cyclegan for single image dehazing*, 2018. arXiv: `1805.05308 [cs.CV]`.

[2] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, *Image-to-image translation with conditional adversarial networks*, 2018. arXiv: `1611.07004 [cs.CV]`.

[3] Z. Anvari and V. Athitsos, "Enhanced cyclegan dehazing network," in *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, vol. 4, Feb. 2021, pp. 193–202.

[4] Y. Jiang, X. Gong, D. Liu, *et al.*, *Enlightengan: Deep light enhancement without paired supervision*, 2021. arXiv: `1906.06972 [cs.CV]`.