Deep Learning Major Exam Report

Self-augmented Unpaired Image Dehazing and Denoising via Density and Depth Decomposition

- **Soumik Roy, B20AI042**

Research Paper : 🔗 Self-augmented Unpaired Image Dehazing via Density and Depth Decomposition

Overview

About the Research Paper

Recently several methods have tried to fix the problem of overfitting that happens when dehazing models are trained on pairs of fake hazy and clean images. They do this by training the models on unmatched data. Most of them just come up with dehazing and rehazing cycles and don't take into account the physical features of hazy environments in the real world.

In this paper, the Authors suggest a framework called D4 (Dehazing via Decomposing transmission map into Density and Depth) for making and getting rid of haze. Instead of just estimating transmission maps or clean content, the suggested framework looks at the scattering coefficient and depth information in both hazy and clean images. With an estimate of the scene depth, their method can re-render hazy pictures with different thicknesses, which helps train the dehazing network even more. It's important to note that the whole training process only needs unpaired hazy and clean pictures, but the scattering coefficient, depth map, and clean content were all reconstructed from a single hazy image. Extensive tests show that our method is better than current unpaired dehazing methods, even though it uses a lot fewer factors and FLOPs.

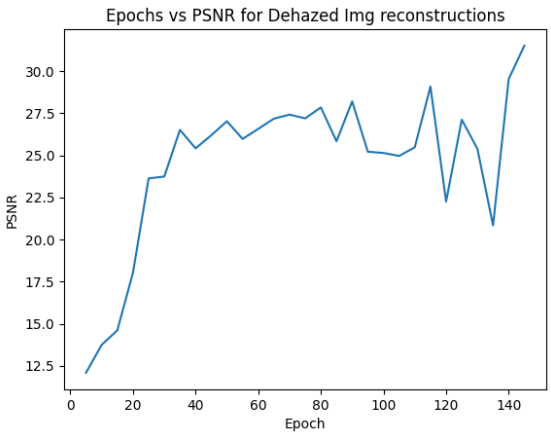
Part A

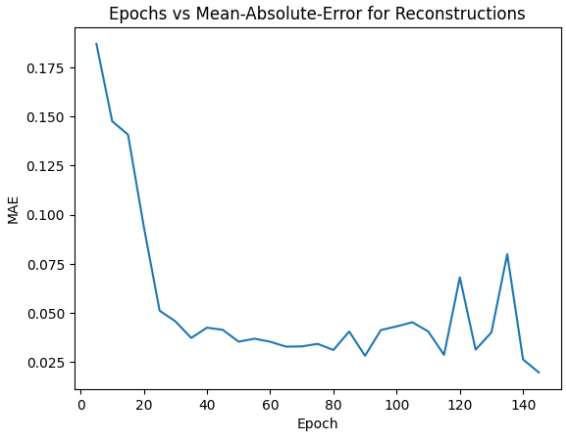
Reproducing the Results

The authors suggest a GANs based framework they call as D4 to do the tasks at hand. I have referred to their official code for the model related implementations. Further I imported one of the datasets they suggest for training, that is the Reside Dehazing Dataset (indoor for train, outdoor for test)

I train the model for 150 Epochs, and train it on the dataset. The authors have used 2 metrics, TSNR and Mean-Absolute-Error as the evaluation criteria of the results.

Metric curves while training came out as follows :





Testing

Finally on testing the Dehazing performance on Test Dataset (outdoor Reside Hazy dataset), we get the following results

- ***Average PSNR = 23.17***

Some Examples:

Part B

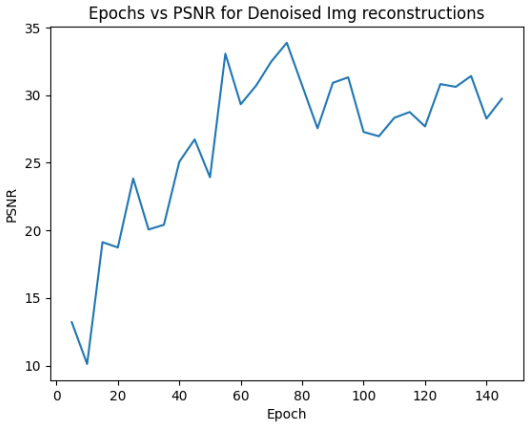
Another Application : applying the Model for DeNoising

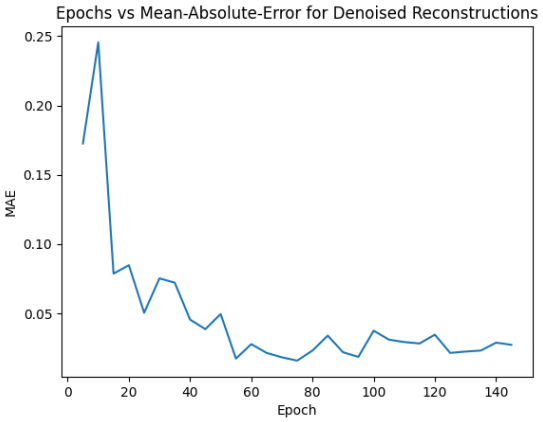
As we can see the model works on the images to decompose the Density and Depth related information. Using this, the D4 model learns how to remove the haze that was above the actual content of the model. (Haze was not actually over the image, but it looks like that). So I decided to try the same implementation but inorder to do DeNoising instead of Dehazing. We can think of the noise as some content that lies over the actual content of the image. Hence we can take the pretrained model, and train in for a few more epochs on a Noisy dataset inorder to fine tune the model for DeNoising.

To create the dataset, I imported the CIFAR10 Dataset, and applied 25% Salt and Pepper Noise to all the images. Finally I provided these Noisy images instead of the Hazy dataset to the previously trained model, and trained it for 150 more epochs.

Results

Metric curves while training :



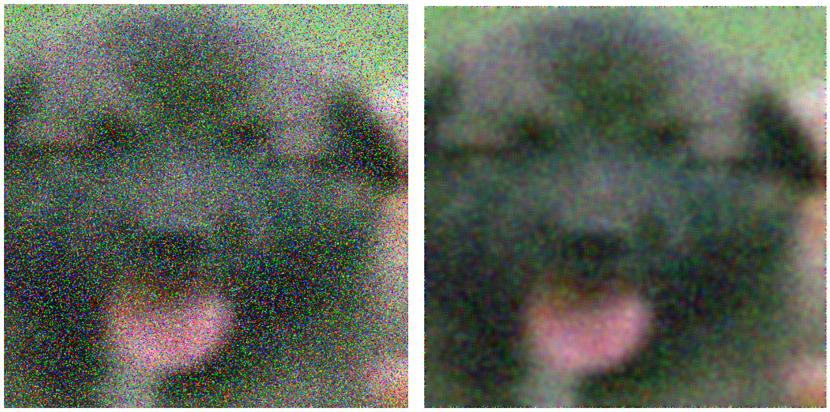


Testing Results :

- **Average PSNR = 18.036**

Some Examples on Test Data :

Noisy Denoised



Part C

Asking ChatGPT for a new application / Views on my Idea

I asked ChatGPT for a new application for the research paper. It suggested the following :

1. **Fog Removal:** You can train the same model to remove fog from images, by using a dataset of images that contain foggy scenes. This would be a similar problem to dehazing, as fog can also reduce the visibility of objects in an image.
2. **Image Super-Resolution:** You could modify the model to perform super-resolution of images, by training it on a dataset of low-resolution images and their corresponding high-resolution versions.
3. **Image Colorization**: Another possible application would be to train the model to colorize grayscale images, by using a dataset of grayscale images and their corresponding color versions.
4. **Image Inpainting**: You can modify the model to perform image inpainting, by training it on a dataset of incomplete images and their corresponding complete versions.
5. **Style Transfer**: You could train the model to perform style transfer, by using a dataset of images with different styles and their corresponding content images.
6. **Image Restoration:** You can use the model to restore images that have been degraded due to various factors such as compression, noise, or blur. You can train the model on a dataset of degraded images and their corresponding clean versions.

All the above implementations seem quite promising ideas. Fog Removal is an application that is almost the same as DeHazing.

Image Super Resolution can also be a good implementation, however would require some model changes to upscale the images.

Image Colorization also seems like a good idea. However I tried to do this Implementation on the proposed framework by training for 150 epochs. The training and metrics showed good performance, however in the end, when actually we compare the results, the model performs very bad. **It is almost completely ineffective in recolorization**. For doing this, I coveted the CIFAR10 images to grayscale, and provided it in place of the hazy data set to the D4 model for training.

Recolorization results after 150 epochs :

Image Inpainting, restoration are also tasks similar to denoising so they would also be good implementations.
Style Transfer however seems like something that will be **difficult** as an application for D4. **Style transfer has no relations with Depth or density**, it involves combination of 2 images, which is not something our model is trained to do.