Project 1 Image Enhancement: Single Image Haze Removal

Cole Crescas, Bronte Li, Karan Shah

2023-09-21

CS 7180 Advanced Perception

Northeastern University

Papers Referenced

Haze removal for single image: A comprehensive review

Vision Transformers for Single Image Dehazing

Physical-priors-guided DehazeFormer

Efficient Image Dehazing with Boundary Constraint and Contextual Regularization

Dataset

A Multi-Purpose Realistic Haze Benchmark With Quantifiable Haze Levels and Ground Truth (P. Narayanan et al., "A Multi-Purpose Realistic Haze Benchmark With Quantifiable Haze Levels and Ground Truth," in IEEE Transactions on Image Processing, vol. 32, pp. 3481-3492, 2023, doi: 10.1109/TIP.2023.3245994.)

https://a2i2-archangel.vision/haze

Github

DehazeFormer: main

Project repo

Abstract

In this project, we read multiple papers in the area of Haze removal, documented previous work and experimented with different algorithms on a real world dataset. After reviewing prior work, research can be grouped into prior based, deep-learning based and constraint based. Experiments and other work generally applied some combination of solutions in an aim to incrementally improve results. The two models used, took different approaches to removing haze from input images. The first approach, DehazeFormer [2], employs supervised learning to pre-train the dehazing transformer-based network, followed by fine-tuning through semi-supervised learning, guided by physical principles. The second method aims to improve efficiency by adding geometric constraints and contextual regularization to a single image [1]. Starting with variations of the model DehazeFormer, we implemented fine-tuning through multiple model variants and unfrozen training layers. Measured by PSNR on real world images, we achieved similar results on real world images but identified a best transformer-based model on this dataset. We found that the DehazeFormer-s model with the fourth and fifth layers unfrozen for fine-tuning achieved the best human observable results but not the best PSNR. Next, we tested different parameters of the constraint-based algorithm, and applied a more qualitative analysis of the results. We found that while a lot of haze was removed, the image was significantly darker as a byproduct. The task of dehazing images proved to be difficult and prone to unwanted artifacts or darkening images overall, continuing to be a difficult task in image enhancement.

Introduction

One of the main issues at hand is the difficulties with image dehazing, specifically applied to single images and real world data. Images collected on hazy days suffer from reduced contrast and color degradation, which seriously affect the functionality of outdoor machine vision systems, such as object detection and video surveillance. Single image dehazing, is a more challenging problem, since fewer information about the scene structure is available. We used algorithms from two papers, with two different approaches on the A212-Haze dataset.

The A212-Haze dataset, is built around the goal of semantic scene understanding in vision systems, particularly applied to intelligence, surveillance, and reconnaissance (ISR). Scene understanding for ISR and autonomous vehicles becomes extremely challenging in adverse weather conditions such as haze, fog and mist.

The model in Song et all [2] employs supervised learning to pre-train the dehazing network, followed by fine-tuning through semi-supervised learning, guided by physical principles. This two-step process enhances dehazing performance on synthetic and real-world hazy images.

The work in Meng et all [1] introduces an efficient technique for single-image haze removal. Their approach leverages the inherent boundary constraint within the transmission function. They incorporate this constraint, along with a weighted L1-norm contextual regularization, into an optimization framework for estimating the unknown transmission. Additionally, they present an efficient algorithm using variable splitting to solve this optimization problem. Compared to existing methods, their approach produces visually pleasing results with accurate color representation and finer image details and structures.

Prior Work

Prior work on this problem space can be grouped into various different solutions and often combines methods to achieve incremental progress.

Prior-based:

- Overall prior based methods are very effective on image enhancement but can add unwanted artifacts, such as Halos, when the image is hazy
- He *et al*. [7] present an interesting image prior - *dark channel prior* for single image dehazing. This prior comes from an observation that most local patches in haze-free images often contain some low intensity pixels. The prior, combined with a soft-mating operation, can achieve a quite compelling haze-free result of very high quality.

Deep-learning-based approaches:

- Deep-learning-based methods restore haze-free images by training neural networks on large-scale datasets. Works well on synthetic images but since neural nets are prone to overfitting, they do not perform well on real world images. For the context of our work, this does not work on well with the proposed dataset.

Constraint based:

- Kratz *et al*. [8] model an image as a factorial Markov random field, in which the scene albedo and depth are two statistically independent latent layers.
- Meng et. all [1] summarized that single image dehazing is essentially an under-constrained problem. The general principle of solving such problems is therefore to explore additional priors or constraints.

Combining deep learning with physical priors:

There exist studies that aim to combine physical priors with deep learning for dehazing tasks, such as RefineDNet [5], and PSD [6].

- RefineDNet: effective at dehazing but prone to noise artifacts
- PSD: The images recovered by PSD are prone to color distortion and remaining haze

Methods

Dataset

A2I2-Haze is the first real haze dataset with in-situ smoke measurement aligned to aerial imagery. A2I2-Haze has paired haze and haze-free imagery that will allow fine-grained evaluation of computer vision algorithms [3].

Subsets	Training	Testing
UAV	224 pairs of hazy and haze free images, 240 additional haze free images	119 hazy images
UGV	50 pairs of hazy and haze free images, 200 additional haze free images	200 hazy images

Table 1: Dataset samples

Figure 1: Dataset image samples

DeHazeFormer Model Fine-tuning

The DehazeFormer model has a main network structure of SwinIR blocks which consist of a residual model and Swin Transformer layers. There was a slight adjustment made by simplifying the structure of SwinIR and reducing the number of residual Swin Transformer blocks. This was found to significantly improve the inference speed of the model.

This model also uses a slightly varied non-linear activation function called SoftReLU. Different normalization functions were tested like GELU, ReLU and LeakyLU. Eventually, this work settled on using SoftReLU, which is a simple smooth approximation to the ReLU as an excess between GELU and ReLU.

Shifted Window partitioning with Reflection Padding

Swin Transformer uses cyclic shift with masked MHSA for efficient batch computation but has smaller windows at image edges. For tasks like image dehazing where edge information matters, this work proposes using reflection padding instead, avoiding inter-patch interactions. This slightly increases computational costs, but for larger images, it's less significant as edge regions become a smaller portion. This adjustment aims to improve performance. The work proposed also created multiple invariants for testing which we will explore in our results.

TABLE I
DETAILED ARCHITECTURE SPECIFICATIONS.

	Num. of Blocks	Embedding Dims	MLP Ratio	Attention Ratio	Num. of Heads	Conv Type
DehazeFormer-T	[4, 4, 4, 2, 2]	[24, 48, 96, 48, 24]	[2, 4, 4, 2, 2]	[1/4, 1/2, 3/4, 0, 0]	[2, 4, 6, 1, 1]	DWConv
DehazeFormer-S	[8, 8, 8, 4, 4]	[24, 48, 96, 48, 24]	[2, 4, 4, 2, 2]	[1/4, 1/2, 3/4, 0, 0]	[2, 4, 6, 1, 1]	DWConv
DehazeFormer-B	[16, 16, 16, 8, 8]	[24, 48, 96, 48, 24]	[2, 4, 4, 2, 2]	[1/4, 1/2, 3/4, 0, 0]	[2, 4, 6, 1, 1]	DWConv
DehazeFormer-M	[12, 12, 12, 6, 6]	[24, 48, 96, 48, 24]	[2, 4, 4, 2, 2]	[1/4, 1/2, 3/4, 0, 0]	[2, 4, 6, 1, 1]	ConvBlock
DehazeFormer-L	[16, 16, 16, 12, 12]	[48, 96, 192, 96, 48]	[2, 4, 4, 2, 2]	[1/4, 1/2, 3/4, 0, 0]	[2, 4, 6, 1, 1]	ConvBlock

Figure 2: We provide five DehazeFormer's variants (-T, -S, -B, -M, and -L for tiny, small, basic, middle, and large, respectively)

For our testing we configured the environment and processed images from our dataset into training and test sets using an 80-20 split. We then loaded the trained variations of outdoor de-hazing models (-S, -B, -M) based on our dataset size and unfroze the last two layers for fine-tuning. We saw the best results using Dehazeformer-s fine-tuned with 2 layers unfrozen. In addition, we measured the training loss and PSNR through increasing epoch to achieve best results.

Efficient image dehazing

Images taken in foggy conditions suffer from poor visibility due to blurring, color fading, and low contrasts. Meng et al [1] proposes a boundary constraint on the scene transmission along with a weighted L1 - norm based contextual regularization between neighboring pixels which is modeled into an optimization problem to estimate the unknown scene transmission. This method consists of three main contributions:

1. A constraint on the scene transmission with a geometric interpretation
2. New contextual regularization that allows for incorporating a filter bank into image dehazing
3. Efficient optimization scheme enabling dehazing images of large sizes quickly

$$\mathbf{I}(x) = t(x)\mathbf{J}(x) + (1 - t(x))\mathbf{A}$$

The hazing imaging model (equation above)presents the widely used linear interpolation model to explain creation of haze images. I(x) is the observed image, J(x) is the scene radiance, A is global atmospheric light, and t(x) is the scene transmission. Goal of image dehazing is to obtain J(x) from I(x).

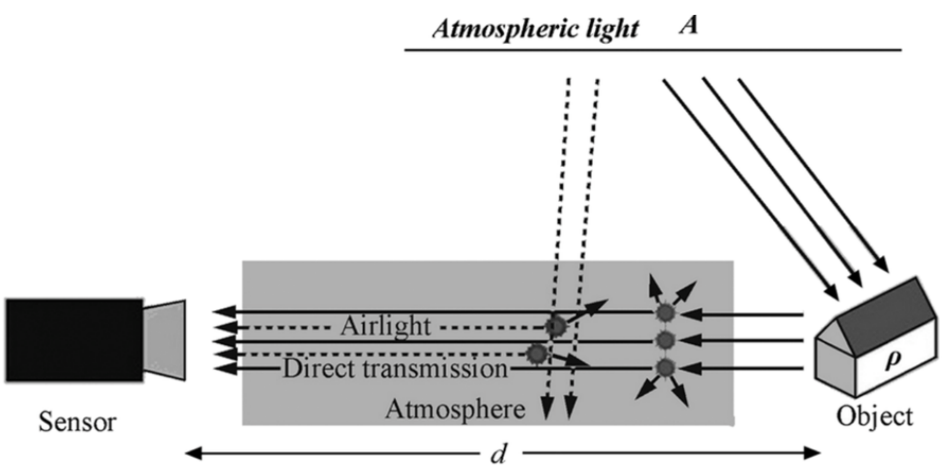


Figure 3. Depiction of hazing model

Figure 3 shows the depiction of the hazing model. In the figure, atmospheric light bounces off the object and if it makes it to the sensor without any hindrance than that is a direct transmission otherwise air light makes it to the sensor.

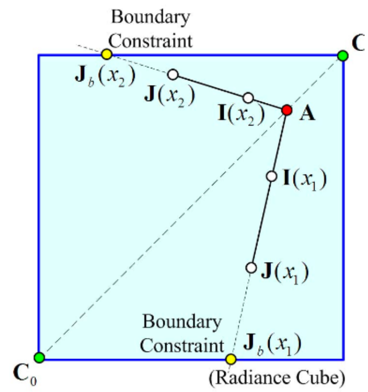


Figure 4. Radiance cube and
boundary constraint

Geometrically, value of hazy pixel (I(x)) is the linear interpolation between the clear pixel (J(x)) and the atmospheric light (A). Pixel I(x) moves toward A when contaminated by fog. To dehaze I(x), an intuitive idea is to push it back towards J(x) through linear extrapolation. This process has to stop when I(x) is pushed to the boundary of the radiance cube. The geometry can be observed by viewing the radiance cube in Figure 4.

$$t_b(x) = \min\left\{ \max_{c\in\{r,g,b\}} \left(\frac{A^c - I^c(x)}{A^c - C_0^c}, \frac{A^c - I^c(x)}{A^c - C_1^c} \right), 1 \right\},$$

The boundary of the scene transmission is defined above. Next, rather than pushing a single pixel, an image patch is pushed back. The process stops when any pixel within the image patch touches the boundary.

$$\tilde{t}(x) = \max_{y\in\omega_x} t_b(y)$$

This produces an estimate of t(x) defined in the equation above which has a direct relationship to the amount of linear extrapolation. W_x is a local patch centered at x.

$$\hat{t}(x) = \min_{y\in\omega_x} \max_{z\in\omega_y} t_b(z).$$

However, for pushing an image patch, the stopping condition stated earlier is strict. Therfore, some pixels are allowed to be pushed outside resulting in the new estimate of t(x) show in the equation above.

$$W(x,y)\left(t(y) - t(x)\right) \approx 0$$

Next, within a local patch, pixels share similar transmission values. However, this assumption fails when there are random depth jumps within the patches. To resolve this, weighting function, W(x,y), is introduced. X and y represent neighboring pixels. The above equation shows the weighted contextual constraint of t(x) between x and y.

$$W(x,y) = e^{-\|\mathbf{I}(x) - \mathbf{I}(y)\|^2 / 2\sigma^2}$$

Now, for the contextual regularization, an appropriate W(x,y) needs to be chosen and in the paper, there are two options, but I only present the one above because I believe it is used in the library implementation.

$$\int_{x\in\Omega}\int_{y\in\omega_x} W(x,y)\,|t(x)-t(y)|\,dxdy$$

Thereafter, the weighted contextual constraints on the entire image and not just the patch is defined by the integral above.

$$\sum_{i\in I}\sum_{j\in\omega_i} w_{ij}\,|t_i - t_j|$$

$$\sum_{j\in\omega}\sum_{i\in I} w_{ij}\,\left|(D_j \otimes t)_i\right|$$

$$\sum_{j\in\omega} \|W_j \circ (D_j \otimes t)\|_1$$

Next, to make computation easier, the discrete form is given in the first step above. Dj represents the first order differential operator in the intermediate step. An L1 norm is used on the penultimate step because more robust to outliers.

Finally, dehazing an image requires estimating an appropriate t(x) and A. To estimate A, filter each color channel of an input image by a minimum filter with a moving window. The maximum value of each color channel is taken as the estimate for each component of A.

$$\frac{\lambda}{2}\|t-\hat{t}\|_2^2 + \sum_{j\in\omega}\|W_j \circ (D_j \otimes t)\|_1,$$

The optimal transmission function t(x) is obtained by minimizing the objective function above. The first part measures the fidelity of t(x) to the patch-wise transmission t_hat(x) and the second part represents the contextual constraints of t(x). Variable splitting is used to optimize this objective function.

The above explanation represents how Efficient Image Dehazing with Boundary Constraint and Contextual Regularization method works. For our experiment, we were able to find the implementation in the following repository: Utkarsh-Deshmukh/Single-Image-Dehazing-Python . Therefore, we used the implementation and loaded the library into simple_dehaze.py and applied it to the three hazy images within our a2i2 UAV-train dataset where we built a new train and val folder which contains hazy and GT folders. This folder hierarchy can be created by running the dehaze.ipynb or train_test_dehazeformer.py Specifically, we applied it to the following hazy images: 060.png, 074.png, 080.png. The default parameters we worked with include the following:

`airlightEstimation_windowSze=15`

`boundaryConstraint_windowSze=3`

`C0=20`

`C1=300`

`regularize_lambda=0.1`

`sigma=0.5`

`delta=0.85`

`showHazeTrasmissionMap=True`

We also used run haze removal to modify some of the parameters and produce different results but ultimately, we used the default parameters to show the results below because these seemed to be the most appropriate for our used case. run haze removal is an app that can be used to adjust three of the parameters: sigma, delta, and the regularization of lambda. Based on various manipulations, delta and the regularization term constitute the major contributions to the darkening and bluing effects seen in the results.

Results

Fine-tuning Dehazeformer

Below are results from fine-tuning different Dehazeformer models, specifically ones pre-trained on outdoor synthetic images. Dehazeformer-s showed best human observed results even though Dehazeformer-m was able to achieve the highest average PSNR. All models are trained with default parameters from the original Dehazeformer configurations, for 30 epochs with a learning rate of `2e-4`.

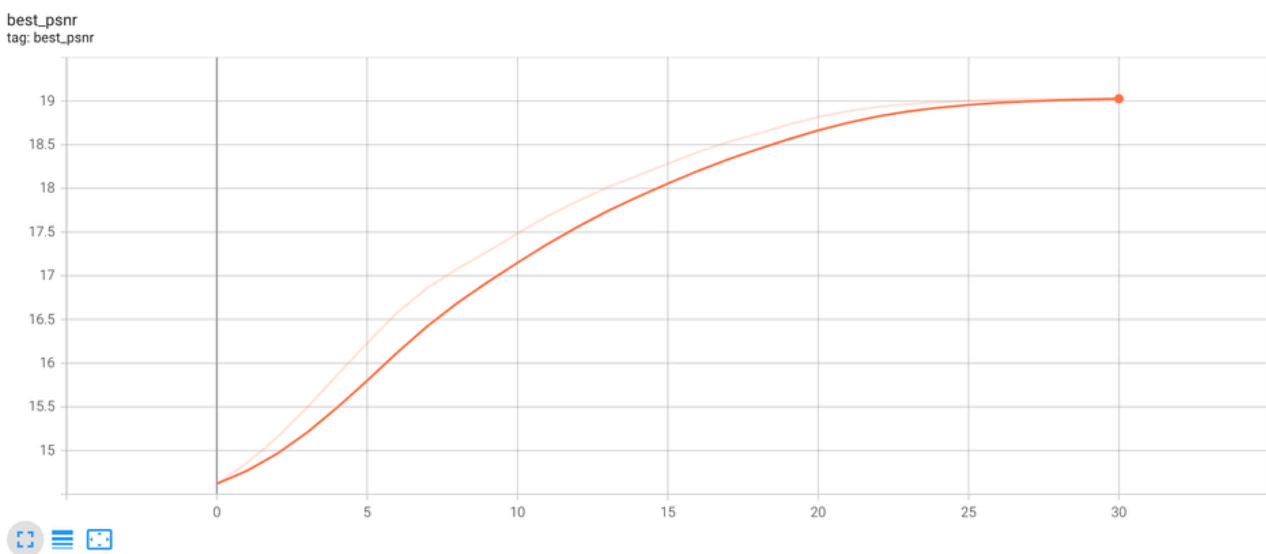
All training results are uploaded to Tensorboard here.

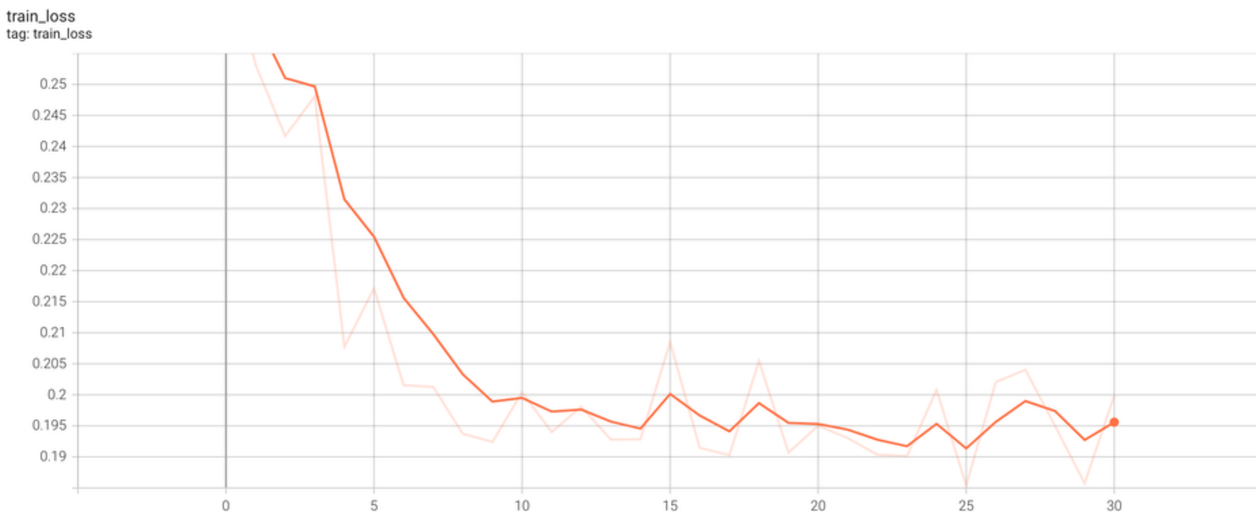
Model: Dehazeformer-s

Finetuning method: Fifth layer unfrozen

Average PSNR: 18.8

Average SSIM: 0.66





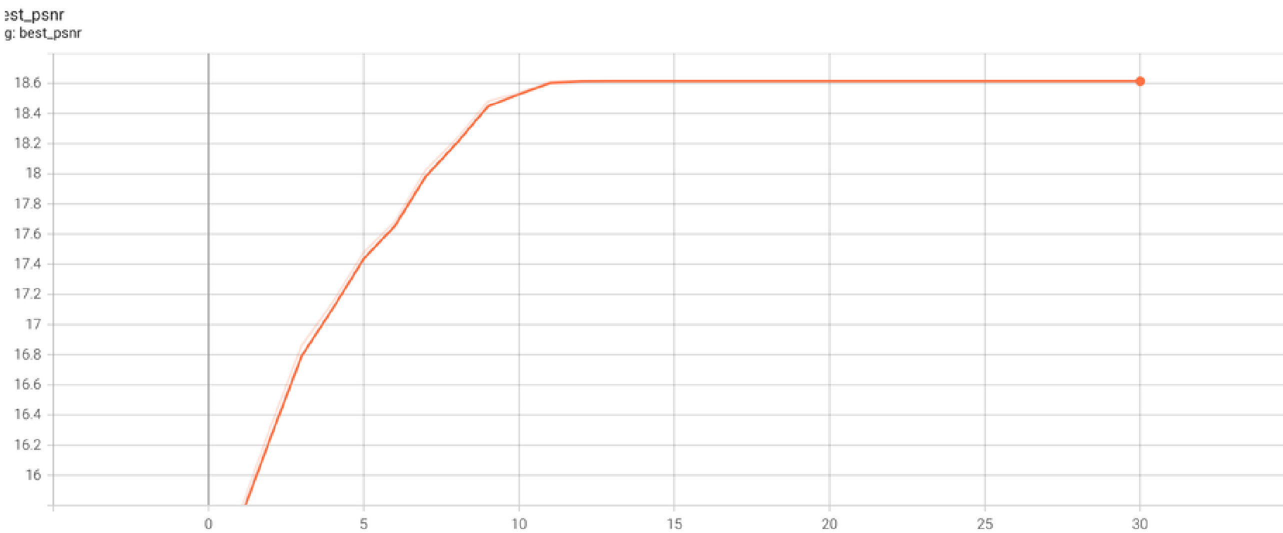
Example Results (Input, output, ground truth)

Model: Dehazeformer-s

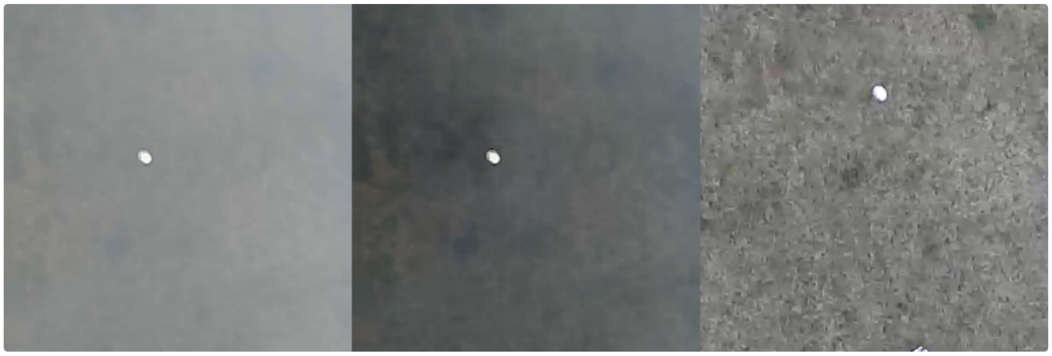
Finetuning method: Fourth and Fifth layer unfrozen

Average PSNR: 15.2

Average SSIM: 0.56



Example Results (Input, output, ground truth)



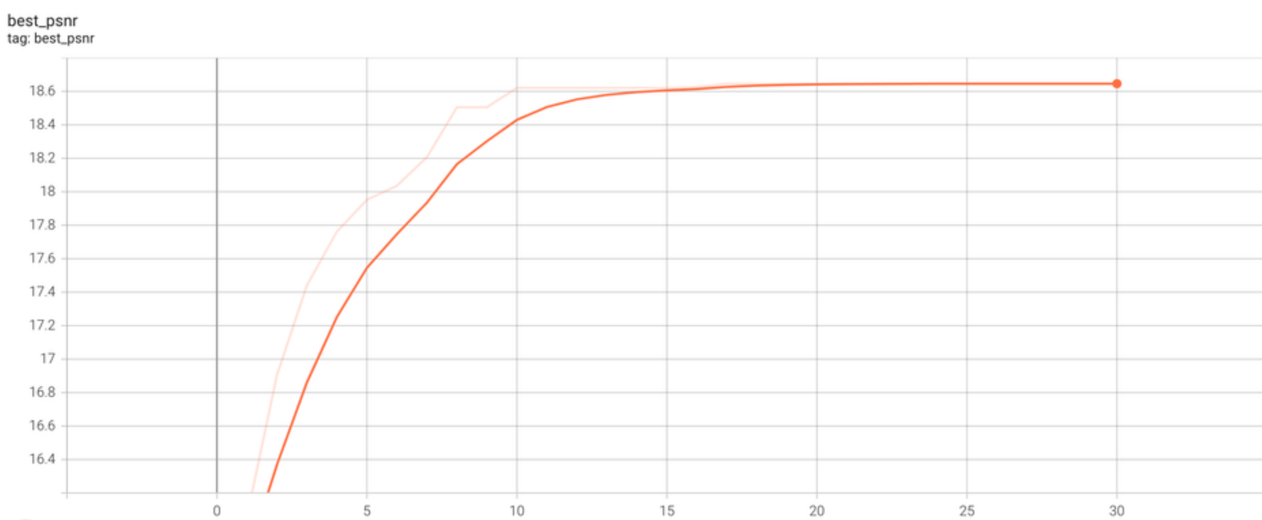
Model: Dehazeformer-b

Finetuning method: Fourth and Fifth layer unfrozen

Average PSNR: 18.65

Average SSIM: 0.64

Training Curve



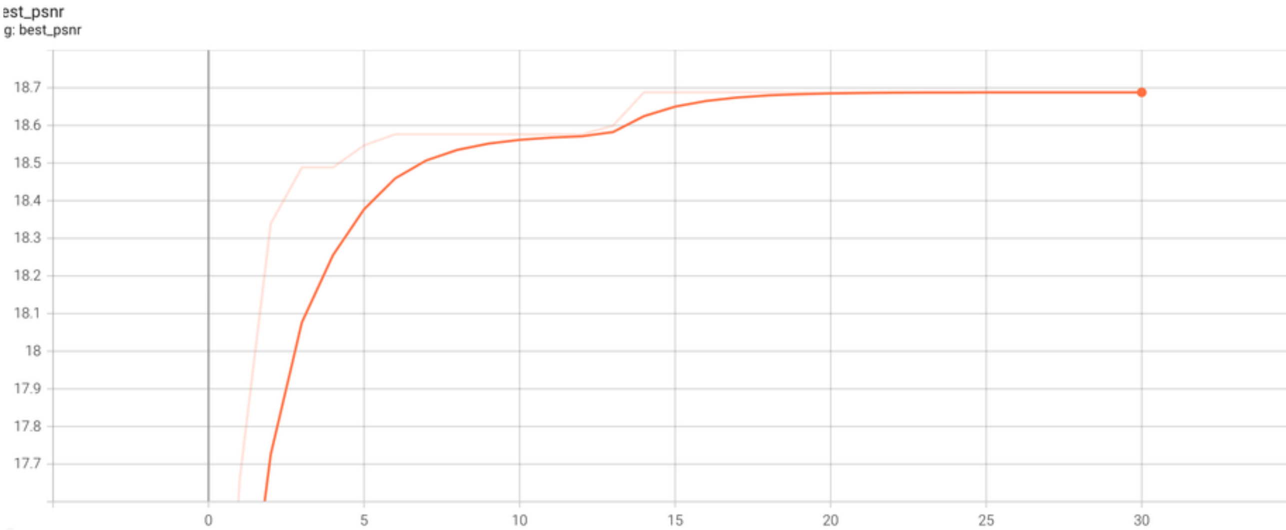
Model: Dehazeformer-m

Finetuning method: Fourth and Fifth layer unfrozen

Average PSNR: 18.69

Average SSIM: 0.63

Training Curve



Results from Image Dehazing Algorithm with Boundary Constraint and Contextual Regularization

Dehaze results using algorithm from paper *Efficient Image Dehazing with Boundary Constraint and Contextual Regularization* are shown below.

Figure 3: Multiple images from results with efficient de-hazing

After reviewing the results of the dehazing algorithm with the boundary constraint and contextual regularization, it seems the algorithm is unable to remove or reduce the haze. In fact, based on the default settings, the results seem to be darker and possibly bluer pixelations in some areas of the different results. Visually, compared to the dehazeformer-s with layer 4 and 5 unfrozen, this algorithm performs worse. On the other hand, this algorithm does not require training like the deep network.

Reflection

- PSNR is not necessarily a good indicator of image haziness, or rather, un-haziness. We saw this as we compare images from outputs from the Dehazeformer models - the largest model `dehazeformer-m` produced higher PSNR, however, the de-hazing effect was empirically not as good.
- Simple de-hazing algorithms tend to make image very dark, the haze itself often simply becomes darker patches instead of being removed.
- Dehazeformer-s fine-tuned with 2 layers unfrozen worked best, without making the image very dark.
- Single image dehazing often suffers from the problem of ambiguity between image color and depth which can lead to inadequate enhancements on the scene objects.
- The A2I2 dataset itself also has limitations such as small sample size, the hazy images and clear images not always matching exactly, etc. This could have contributed to the generally low performance of the methods.

References

[1] G. Meng, Y. Wang, J. Duan, S. Xiang and C. Pan, "Efficient Image Dehazing with Boundary Constraint and Contextual Regularization," *2013 IEEE International Conference on Computer Vision*, Sydney, NSW, Australia, 2013, pp. 617-624, doi: 10.1109/ICCV.2013.82.

[2] Y. Song, Z. He, H. Qian and X. Du, "Vision Transformers for Single Image Dehazing," in *IEEE Transactions on Image Processing*, vol. 32, pp. 1927-1941, 2023, doi: 10.1109/TIP.2023.3256763.

[3] https://a2i2-archangel.vision/haze

[4] https://medium.com/@andrew_johnson_4/deep-dive-into-transformers-understanding-fine-tuning-by-freezing-gradients-528208925990#:~:text=In%20the%20context%20of%20transformer,fine%2Dtune%20the%20final%20layers.

[5] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows", *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pp. 10012-10022, Oct. 2021.

[6] Z. Huang, Y. Ben, G. Luo, P. Cheng, G. Yu and B. Fu, "Shuffle transformer: Rethinking spatial shuffle for vision transformer", *arXiv:2106.03650*, 2021.

[7] K. He, J. Sun, and X. Tang. Single image haze removal using dark channel prior. In CVPR'09, pages 1956-1963, 2009.

[8] L. Kratz and K. Nishino. Factorizing scene albedo and depth from a single foggy image. In ICCV'09, pages 1701-1708, Oct. 2009.