Optical Media
Interface Laboratory

Integrating AECR-Net and Principled S2R Dehazing for Image Defogging

Final Report
NAPI 2022

Presented By
John Benedict Du

Date
January 21, 2022

Optical Media
Interface Laboratory

Introduction

- Outdoor images often suffer from low contrast and limited visibility because of haze.
- Affects the performance of tasks, such as object detection and recognition.

https://www.kaggle.com/zekunn/merge-label-unlabel/data

https://github.com/GlassyWu/AECR-Net

Image Defogging

- A process to visually improve degraded visibility by reconstructing the latent clear image.
- Enhance visibility in bad weather due to light being scattered and absorbed by atmospheric particles.

Objectives

Goal: Improvement of Defogging

RESIDE DATASET

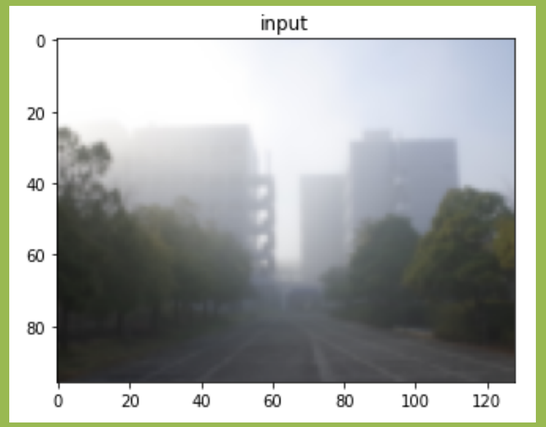
Training & Testing

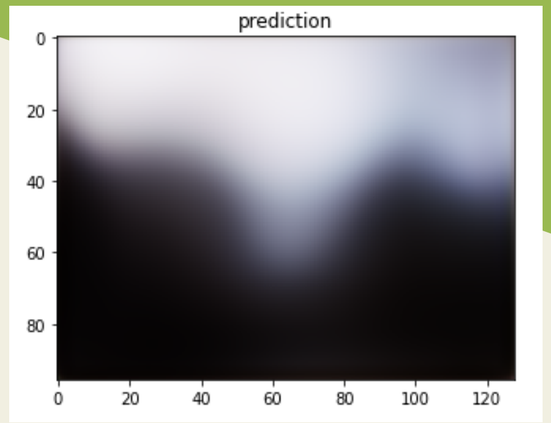
PRINCIPLED S2R DEHAZING

Framework to use

AECR-NET

Model as the backbone





Intended Outputs

- Combine Principled S2R Dehazing and AECR-Net using Python
- Train it using the RESIDE dataset
- Compare its results with other models that were also merged with PSD

https://sites.google.com/view/reside-dehaze-datasets/reside-%CE%B2?authuser=0

RESIDE Dataset

Outdoor Training Set (OTS)

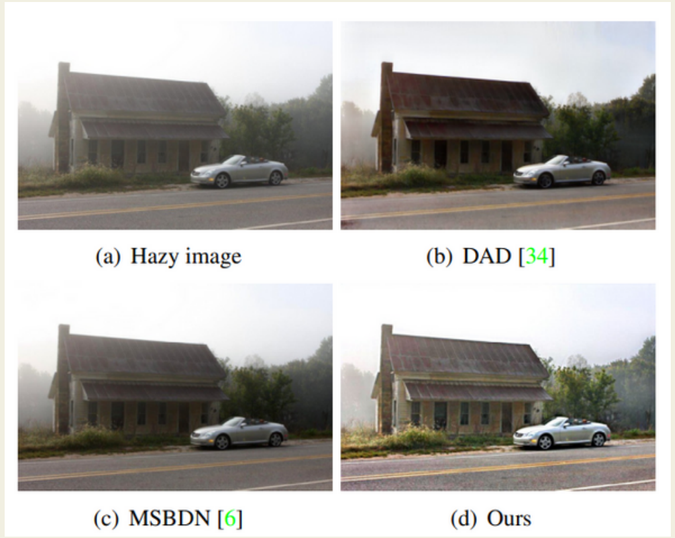
- Used for Training
- 72,135 Sets of Hazy & Clear Images

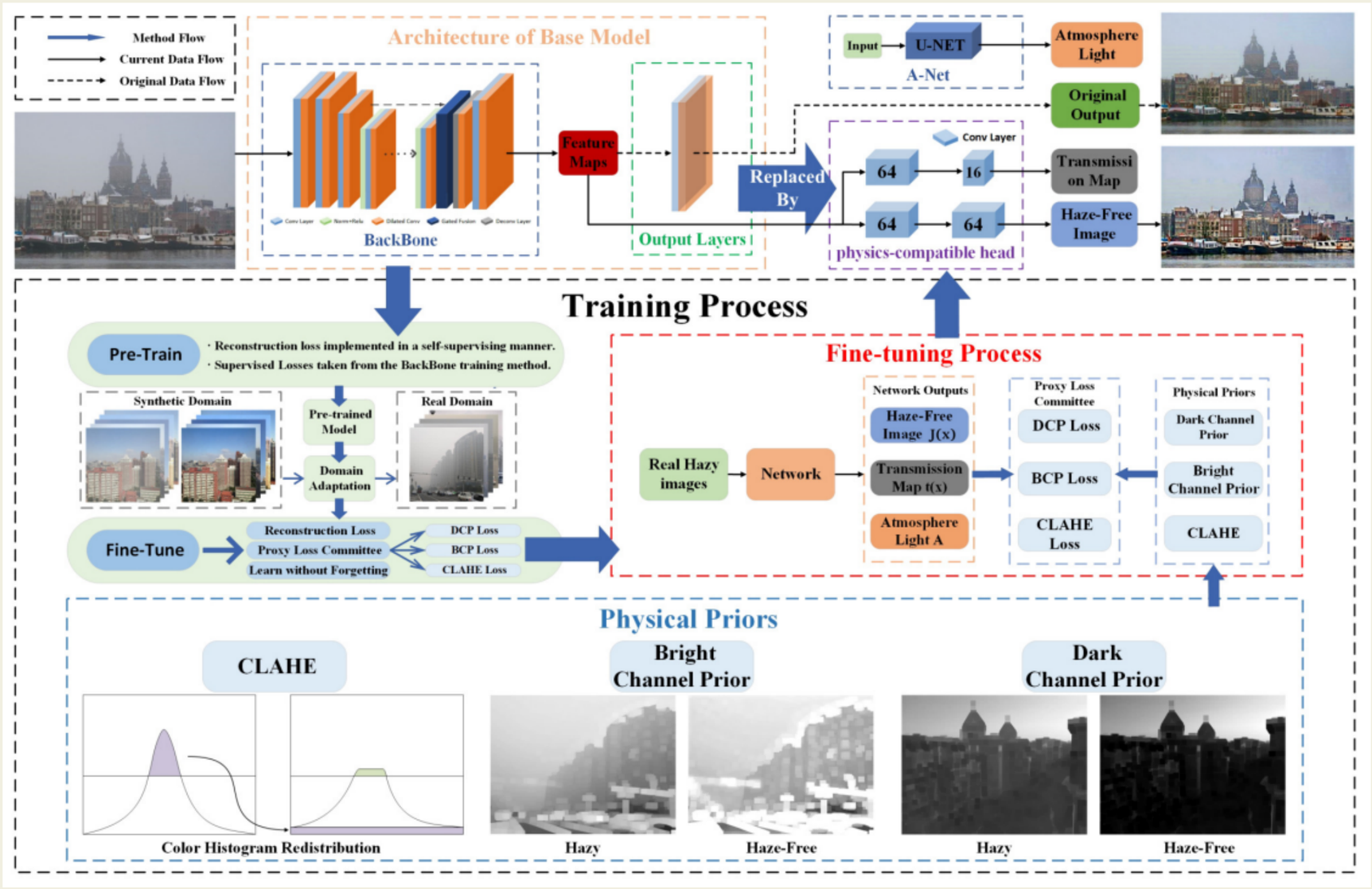
Synthetic Objective Testing Set (SOTS)

- Used for Validation and Testing
- 500 Sets of Hazy & Clear Images

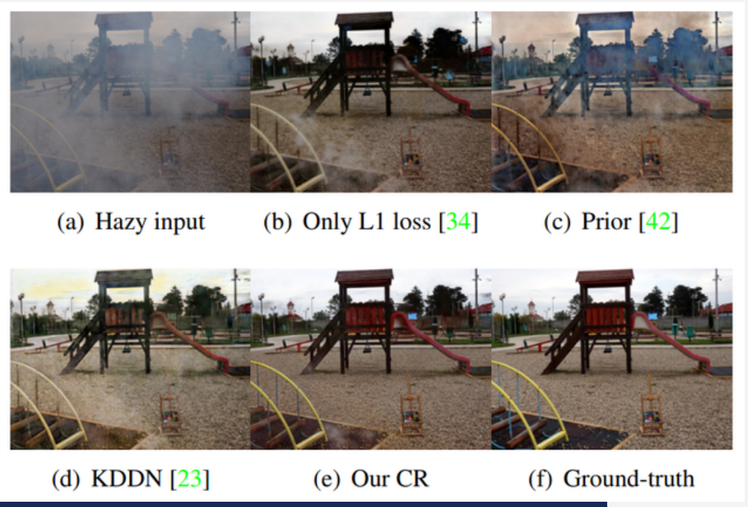
Principled S2R Dehazing

Framework applicable to generalize most of the existing dehazing models to the real domain.



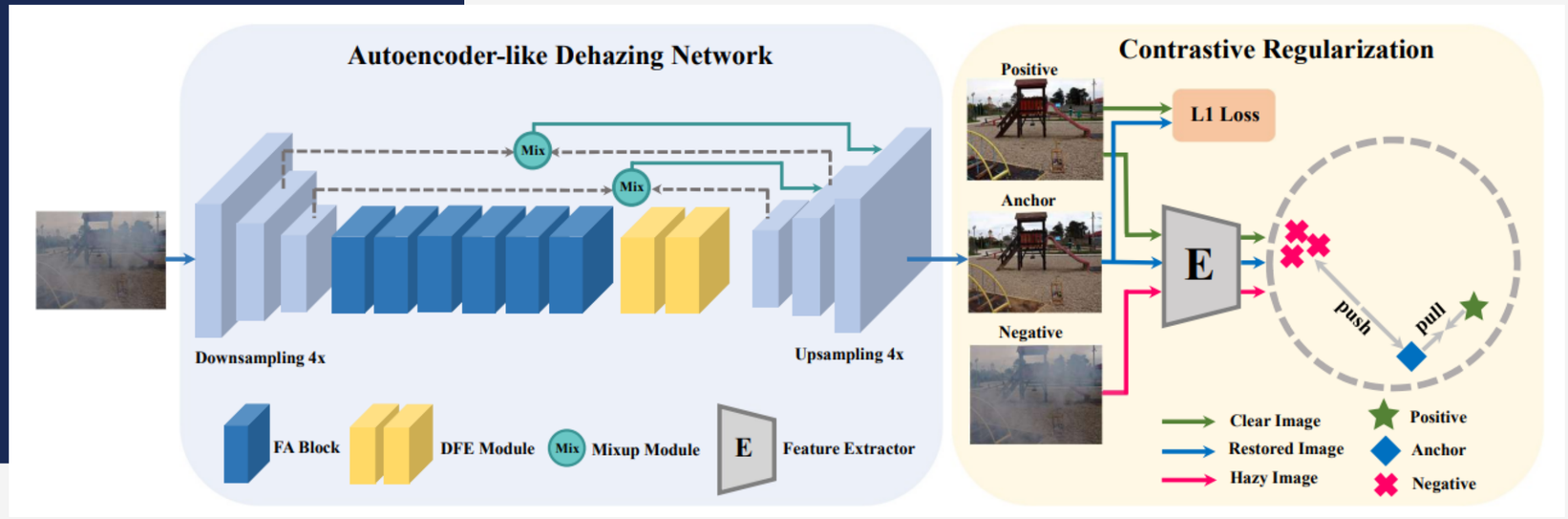


https://github.com/zychen-ustc/PSD-Principled-Synthetic-to-Real-Dehazing-Guided-by-Physical-Priors



AECR-Net

Single image dehazing, which consists of contrastive regularization (CR) and autoencoder-like (AE) network.



https://github.com/GlassyWu/AECR-Net

Methodology

JAN 8 - 11	JAN 12 - 14	JAN 15 - 17	JAN 18 - 19	JAN 20 - 21
Setup Local Python Environment	Study & Get AECR-Net to Work		Testing & Debugging	
Download RESIDE Dataset	Study & Get Principled S2R Dehazing to Work		Data Gathering	
	Modify AECR-Net to work as PSD backbone		Code Clean Up & Documentation	
		Train with Model RESIDE Dataset		Final Report

Setup

RESIDE DATASET

Download of OTS & SOTS ~ 2 days

AECR-NET MODULES

deconv

- GitHub Link

DCNv2

- GitHub Link
- NVIDIA CUDA Toolkit is required
- Run terminal w/ admin in DCNv2 folder and execute:

```
python setup.py build develop --user
```

PSD TESTING

Pre-trained Models

- MSBDNNET
- FFANET
- GCANET

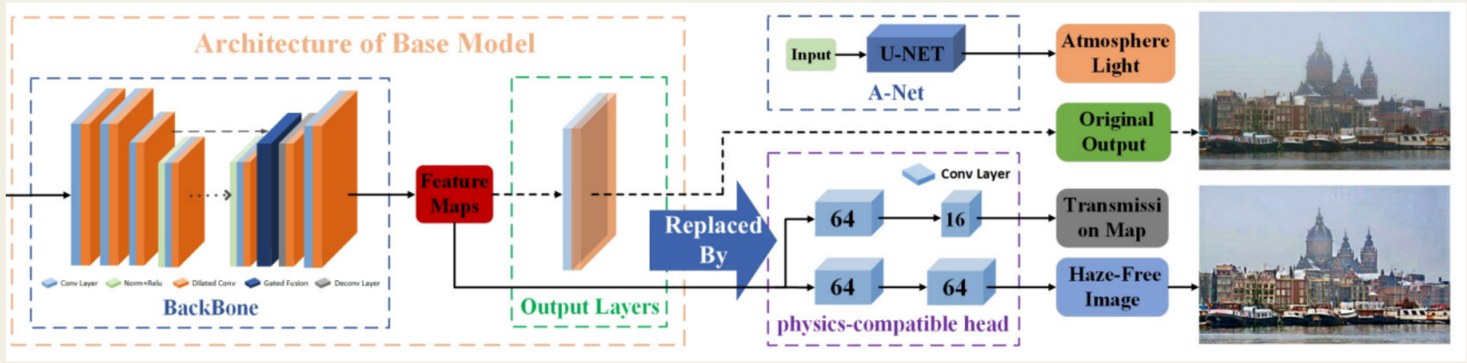
Testing

- The test.py file is hard coded, and the default code is for the testing FFANET model.

RESIDE DATASET TESTING

Trained U-Net & AECR-Net with OTS dataset

AECR-Net Modification



1 ADDED MODULES FOR A-NET

- BlockUNet1
- G2

```
self.ANet = G2(3, 3)
```

2 ADDED 4 CONV LAYERS

```
self.conv_J_1 = nn.Conv2d(64, 64, 3, 1, 1, bias=False)
self.conv_J_2 = nn.Conv2d(64, 3, 3, 1, 1, bias=False)
self.conv_T_1 = nn.Conv2d(64, 16, 3, 1, 1, bias=False)
self.conv_T_2 = nn.Conv2d(16, 1, 3, 1, 1, bias=False)
```

3 UPDATED AECR-NET'S FORWARD FUNCTION

```
out_J = self.conv_J_1(x_up2)
out_J = self.conv_J_2(out_J)
out_T = self.conv_T_1(x_up2)
out_T = self.conv_T_2(out_T)

 if Val == False:
   out_A = self.ANet(input)
else:
   out_A = self.ANet(input2)
out_I = out_T * out_J + (1 - out_T) * out_A

return out, out_J, out_T, out_A, out_I
```

Used the modified versions of FFANet, GCANet & MSBDNNet as reference

Training & Testing

TRAINING SETUP

- The main.py file is hard coded, and the default code is for the training FFANET model.
- RESIDE OTS dataset

TRAINING SETTINGS

num_epoch = 20
batch_size = 8
num_workers=1

VALIDATION METHOD

- PSNR & SSIM
- RESIDE SOTS outdoor dataset
- Used upsampling to match dimensions

TESTBENCH FOR ALL MODELS

- Python Notebook
- All Models
- Output model results as figures
- SOTS & CVPR Datasets

Training Results

```
Date: 2022-01-16 16:59:45s, Time_Cost: 4457s, Epoch: [1/20], Train_PSNR: 21.98
Date: 2022-01-16 18:05:56s, Time_Cost: 3971s, Epoch: [2/20], Train_PSNR: 24.57
Date: 2022-01-16 19:14:19s, Time_Cost: 4103s, Epoch: [3/20], Train_PSNR: 25.54
Date: 2022-01-16 20:22:16s, Time_Cost: 4077s, Epoch: [4/20], Train_PSNR: 26.11
Date: 2022-01-16 21:27:21s, Time_Cost: 3904s, Epoch: [5/20], Train_PSNR: 26.66
Date: 2022-01-16 22:31:06s, Time_Cost: 3826s, Epoch: [6/20], Train_PSNR: 27.06
Date: 2022-01-16 23:34:17s, Time_Cost: 3790s, Epoch: [7/20], Train_PSNR: 27.42
Date: 2022-01-17 00:37:28s, Time_Cost: 3791s, Epoch: [8/20], Train_PSNR: 27.74
Date: 2022-01-17 01:40:23s, Time_Cost: 3775s, Epoch: [9/20], Train_PSNR: 28.00
Date: 2022-01-17 02:43:37s, Time_Cost: 3794s, Epoch: [10/20], Train_PSNR: 28.27
Date: 2022-01-17 03:46:36s, Time_Cost: 3779s, Epoch: [11/20], Train_PSNR: 28.52
Date: 2022-01-17 04:49:57s, Time_Cost: 3801s, Epoch: [12/20], Train_PSNR: 28.73
Date: 2022-01-17 05:52:53s, Time_Cost: 3776s, Epoch: [13/20], Train_PSNR: 28.94
Date: 2022-01-17 06:55:51s, Time_Cost: 3778s, Epoch: [14/20], Train_PSNR: 29.09
Date: 2022-01-17 07:58:44s, Time_Cost: 3773s, Epoch: [15/20], Train_PSNR: 29.25
Date: 2022-01-17 09:01:52s, Time_Cost: 3788s, Epoch: [16/20], Train_PSNR: 29.39
Date: 2022-01-17 10:11:51s, Time_Cost: 4199s, Epoch: [17/20], Train_PSNR: 29.48
Date: 2022-01-17 11:21:59s, Time_Cost: 4208s, Epoch: [18/20], Train_PSNR: 29.54
Date: 2022-01-17 12:34:05s, Time_Cost: 4326s, Epoch: [19/20], Train_PSNR: 29.58
Date: 2022-01-17 13:47:06s, Time_Cost: 4380s, Epoch: [20/20], Train_PSNR: 29.60
```

PSD AECR-Net

Epoch	1	2	3	4	5	6	7	8	9	10
PSNR	24.18	25.11	26.68	26.34	25.08	26.65	26.97	26.32	26.59	26.40
SSIM	0.9246	0.9370	0.9499	0.9498	0.9407	0.9531	0.9563	0.9468	0.9517	0.9421

Epoch	11	12	13	14	15	16	17	18	19	20
PSNR	27.03	26.29	26.46	**27.31**	26.42	27.14	26.92	26.57	26.78	26.71
SSIM	**0.9568**	0.9491	0.9493	0.9562	0.9458	0.9519	0.9504	0.9485	0.9503	0.9500

Validation Results

Model	U-Net	AECR-Net	PSD FFA-Net	PSD GCA-Net	PSD MSBDN-Net
PSNR	19.02	25.28	15.00	14.72	15.18
SSIM	0.8696	0.9350	0.7498	0.7773	0.7656

Model Comparisons

- AECR-Net's CR into SOTA methods reduced the effect of black spots and color distortion

Input | Ground Truth | AECR-Net | PSD AECR-Net | U-Net | PSD FFA-Net | PSD GCA-Net | PSD MSBDN-Net

AECR-Net & PSD AECR-Net Comparison

AECR-Net

PSD AECR-Net

- PSD's A-Net Improves atmosphere color & consistency
- Image's hue look warmer
- Black smudge around objects are reduced

AECR-Net

PSD AECR-Net

Thank You!

REFERENCES

- **RESIDE**
  - https://sites.google.com/view/reside-dehaze-datasets/
- **AECR-Net**
  - https://github.com/GlassyWu/AECR-Net/
  - https://github.com/yechengxi/deconvolution
  - https://github.com/jinfagang/DCNv2_latest
- **PSD**
  - https://github.com/zychen-ustc/PSD-Principled-Synthetic-to-Real-Dehazing-Guided-by-Physical-Priors
- **CVPR**
  - https://www.kaggle.com/zekunn/merge-label-unlabel/data