FoggyStereo: Stereo Matching with Fog Volume Representation Supplementary Material

Chengtang Yao[1,2], Lidong Yu[2]
[1]Beijing Laboratory of Intelligent Information Technology, Beijing Institute of Technology,
[2]Autonomous Driving Algorithm, NIO
yao.c.t@bit.edu.cn, yvlidong@gmail.com

Contents

1. Network Architecture

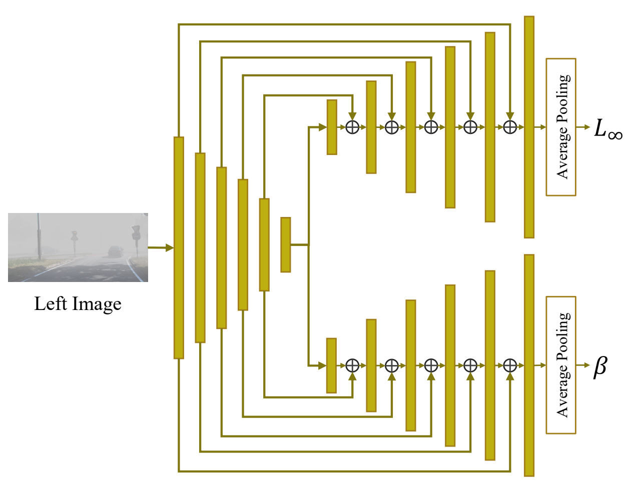


Figure 1. The network architecture of scattering parameters estimation.

1.1. Scattering Parameters Estimation

We present the network architecture of scattering parameters estimation in Fig. 1. We first use six convolutions to encode the input image. We then use the decoding in different branches for two scattering parameters, including the atmospheric light L_∞ and attenuation coefficient β. An average pooling is applied to the decoded results to obtain the final scattering parameters.

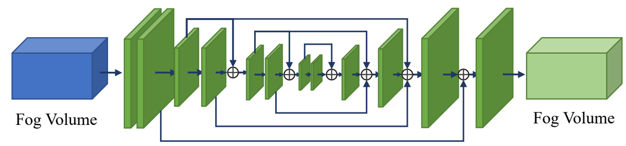


Figure 2. The network architecture of aggregation for fog volume.

1.2. Aggregation for Fog Volume

After constructing the fog volume, we use a series of 3D convolutions to aggregate the information. We present the network architecture of aggregation for fog volume in Fig. 2. We use two 3D convolutions to aggregate the information at first. We then downsample the fog volume three times and then upsample it with skip connection.

1.3. Fusion

As aforementioned in the main body, we compute the variance of cost volume and fog volume and fuse the two volumes via concatenation. We then use three 3D convolutions to improve the fusion further.

2. Experiments

2.1. Visualization of Synthesized Foggy Images

We present the foggy images in Fig. 3 and Fig. 4 to visualize the synthesis results in different datasets. The foggy images are synthesized according to the atmospheric scattering process:

$$I(x) = J(x)T(Z_x) + L_\infty(1 - T(Z_x)), \tag{1}$$

where $T(Z_x) = e^{-\beta Z_x}$, $J(x) = L_\infty \rho(x)$ represents the data in clear scene and $I(x)$ represents the data in foggy scene.

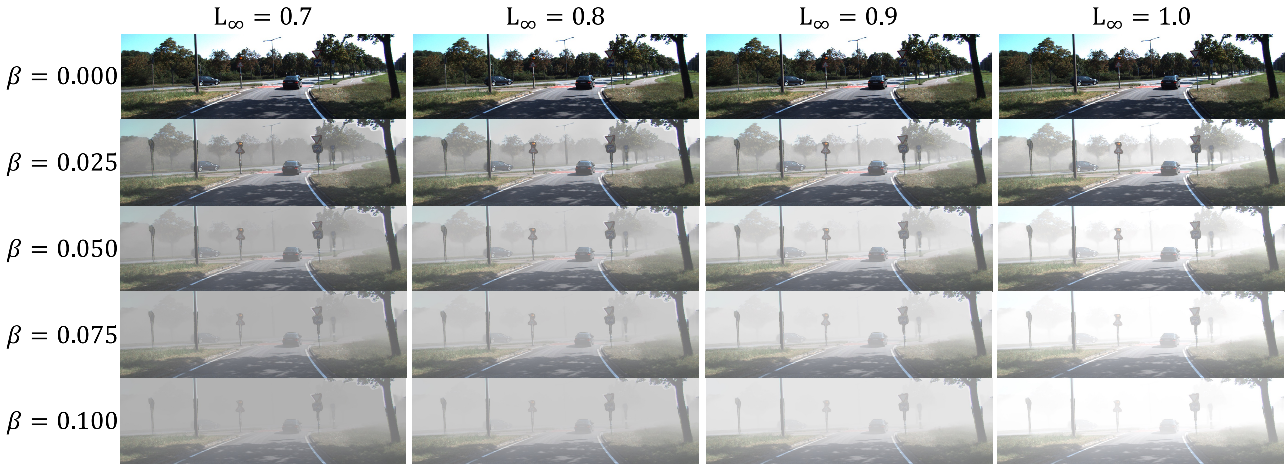


Figure 3. The visualization of synthesized foggy images in the KITTI 2015 dataset.

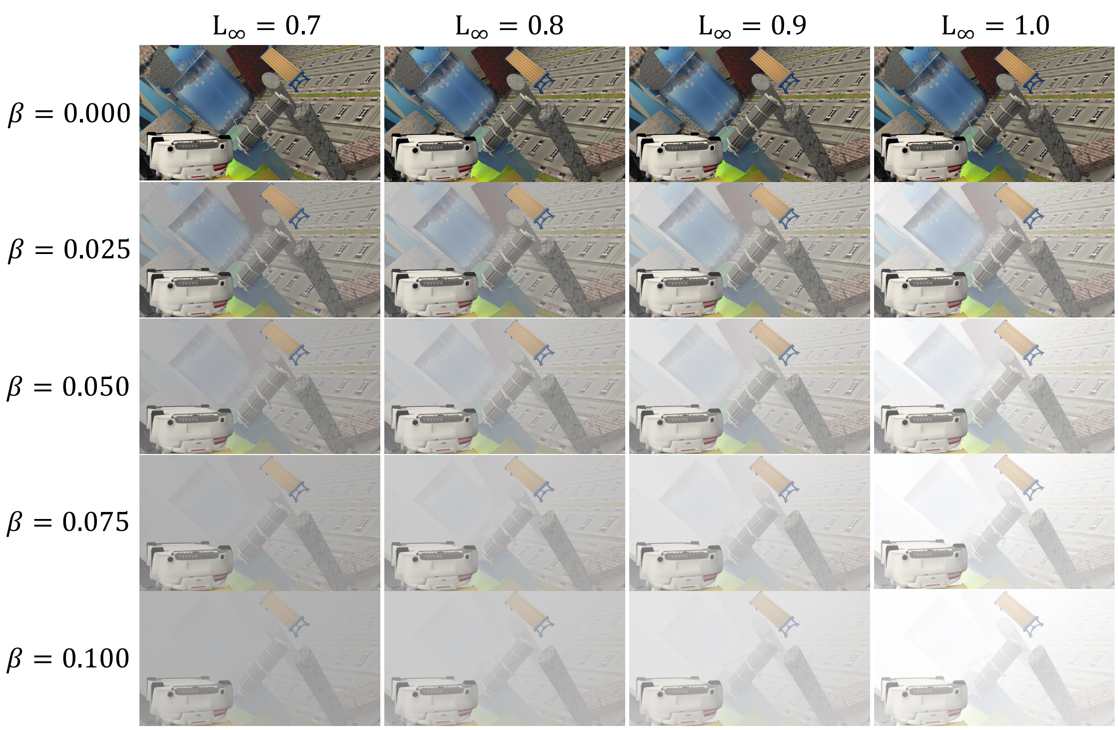


Figure 4. The visualization of synthesized foggy images in the Scene Flow dataset.

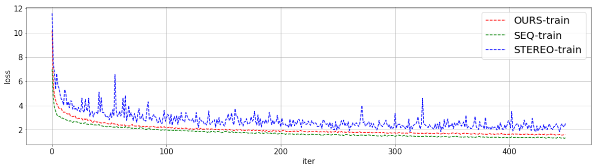
2.2. Learning Stability

In order to show the learning stability of our method, we present the loss distribution in the training process and the testing loss at every 30 epochs. As shown in Fig. 5a, the training loss of DeepPruner [1] is unstable and much higher than that of other methods. Similar results are observed in the testing loss in Fig. 5b. Compared to the 4Kdehazing [2] + DeepPruner [1], our method achieves a stable decrease of loss in testing and is less overfitting, as shown in Fig. 5c.

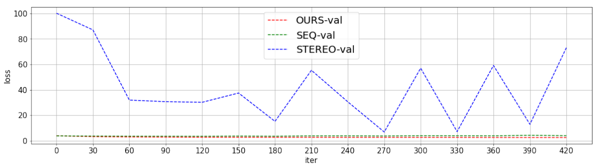
References

[1] Shivam Duggal, Shenlong Wang, Wei-Chiu Ma, Rui Hu, and Raquel Urtasun. Deeppruner: Learning efficient stereo matching via differentiable patchmatch. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4384–4393, 2019. 2, 3

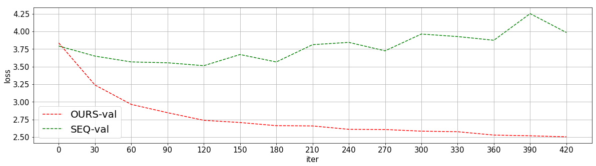
[2] Zhuoran Zheng, Wenqi Ren, Xiaochun Cao, Xiaobin Hu, Tao Wang, Fenglong Song, and Xiuyi Jia. Ultra-high-definition image dehazing via multi-guided bilateral learning. In *Proceedings of the IEEE Conference on Computer Vision and*



(a) The training loss distribution.



(b) The testing of of each epoch.



(c) The detailed testing of of each epoch.

Figure 5. The visualization of loss in training and testing process on KITTI 2015. 'OURS' represents the distribution result of our method. 'SEQ' represents 4Kdehazing [2] + DeepPruner [1]. 'STEREO' represents the distribution result of DeepPruner [1].

Pattern Recognition (CVPR), pages 16185–16194, 2021. 2, 3