

# Supplementary Material

## Semi-MoreGAN: Semi-supervised Generative Adversarial Network for Mixture of Rain Removal

Yiyang Shen<sup>1</sup>, Yongzhen Wang<sup>1</sup>, Mingqiang Wei<sup>1†</sup>, Honghua Chen<sup>1</sup>, Haoran Xie<sup>2</sup>, Gary Cheng<sup>3†</sup>, Fu Lee Wang<sup>4</sup>

<sup>1</sup>School of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China

<sup>2</sup>School of Computing and Decision Sciences, Lingnan University, Hong Kong, China

<sup>3</sup>School of Mathematics and Information Technology, Education University of Hong Kong, Hong Kong, China

<sup>4</sup>School of Science and Technology, Hong Kong Metropolitan University, Hong Kong, China

This supplemental material contains five parts as follows:

1. Additional qualitative results on real-world rainy images.
2. Parameter settings analysis.
3. Data paradigm analysis.
4. Impact of precise depth prediction.
5. Application.

### 1. Additional qualitative results on real-world rainy images.

Fig. 1 and Fig. 2 show the mixture of rain scenes and those that only contain rain streaks, respectively. Our method restores the best results on these real-world rainy images, which demonstrate that Semi-MoreGAN can tackle various degradations in real-world rainy scenes and preserve image details effectively.

**Table 1:** Quantitative evaluation on the testing set of RainCityscapes. Note that deraining performance increases with different number settings of CFABs.

Number	1	2	3	4	5	6
PSNR	30.67	32.39	34.22	35.67	35.75	35.81
SSIM	0.912	0.932	0.941	0.948	0.949	0.951

**Table 2:** Quantitative evaluation and GPU time on the testing set of RainCityscapes, DGNet w PDNL indicates replacing DGNet module by our PDNL, and 56.3ms/52.8ms indicate the GPU time of replacing the whole PDNL and PDNL without attention module.

Datasets	Metrics	DGNet	DGNet w PDNL
RainCityscapes	PSNR	32.21	34.72/33.10
	Time	76.7ms	56.3ms/52.8ms

### 2. Parameter Settings Analysis

We also conduct experiments to adapt different number settings of CFABs as shown in Table 1. Deraining performance increases with the number of CFABs. While for a model with four CFABs, adding more layers obtains little improvement on deraining performance, thus we adopt four CFABs in the design of CFPN. Besides, we also compare DGNet [HZW\*21] with our PDNL. Concretely, we replace the DGNet [HZW\*21] by our PDNL with the pooling size  $n \in \{1, 2, 4, 8\}$ , and set the input size of depth map and feature map to (1,512,1024) and (64,256,512), respectively. Table 2 shows that the network with PDNL achieves better performance with a less computational time. Furthermore, the additional attention module costs a little more consuming time (3.5ms) with significant improvement (1.62dB PSNR).

**Table 3:** Deraining results (PSNR/SSIM) with different data paradigms. Note that &MOR-Rain200 denotes that the model is trained on RainCityscapes&MOR-Rain200, and all models are tested on the testing set of RainCityscapes.

Settings	&MOR-Rain200	&MOR-Rain600	&MOR-Rain1000
PSNR	35.67	34.39	32.36
SSIM	0.948	0.940	0.936

### 3. Data Paradigm Analysis

Since Semi-MoreGAN is trained in a semi-supervised manner, we also analyze the capacity of Semi-MoreGAN to use different data paradigms for training the model on RainCityscapes and real-world rainy images. Specifically, we add 600 and 1000 real rainy images to the unsupervised learning branch for training, which are denoted as MOR-Rain600 and MOR-Rain1000.

From Table 3 and Fig. 3, we make the following observations: Increasing the amount of real-world rainy images leads to significant deraining performance drop on the testing set of RainCityscapes,



**Figure 1:** Visual comparison results on heavy rainy images. Note that all models are trained on RainCityscapes&MOR-Rain200.

**Table 4:** Quantitative evaluation on the testing set of RainCityscapes. Note that w ADPN denotes networks incorporated with ADPN, r ADPN denotes replacing the depth prediction subnetwork of DGNet-Net [HZW\*21] by ADPN, and we use a common multiplication on obtained feature map and depth map.

Model	PReNet		Syn2Real		DGNet-Net	
	-	w ADPN	-	w ADPN	-	r ADPN
PSNR	26.83	28.17(↑ 1.34)	28.66	31.01(↑ 2.35)	32.21	34.14(↑ 1.93)
SSIM	0.910	0.917(↑ 0.07)	0.919	0.925(↑ 0.06)	0.936	0.940(↑ 0.004)

due to the domain shift between synthetic and real images, which makes the training process difficult to converge. However, the performance of real-world image deraining gradually becomes better when adding more real images. From these experiments, we can conclude that Semi-MoreGAN possesses a better generalization ability when leveraging more unpaired real-world rainy images.

#### 4. Impact of precise depth prediction

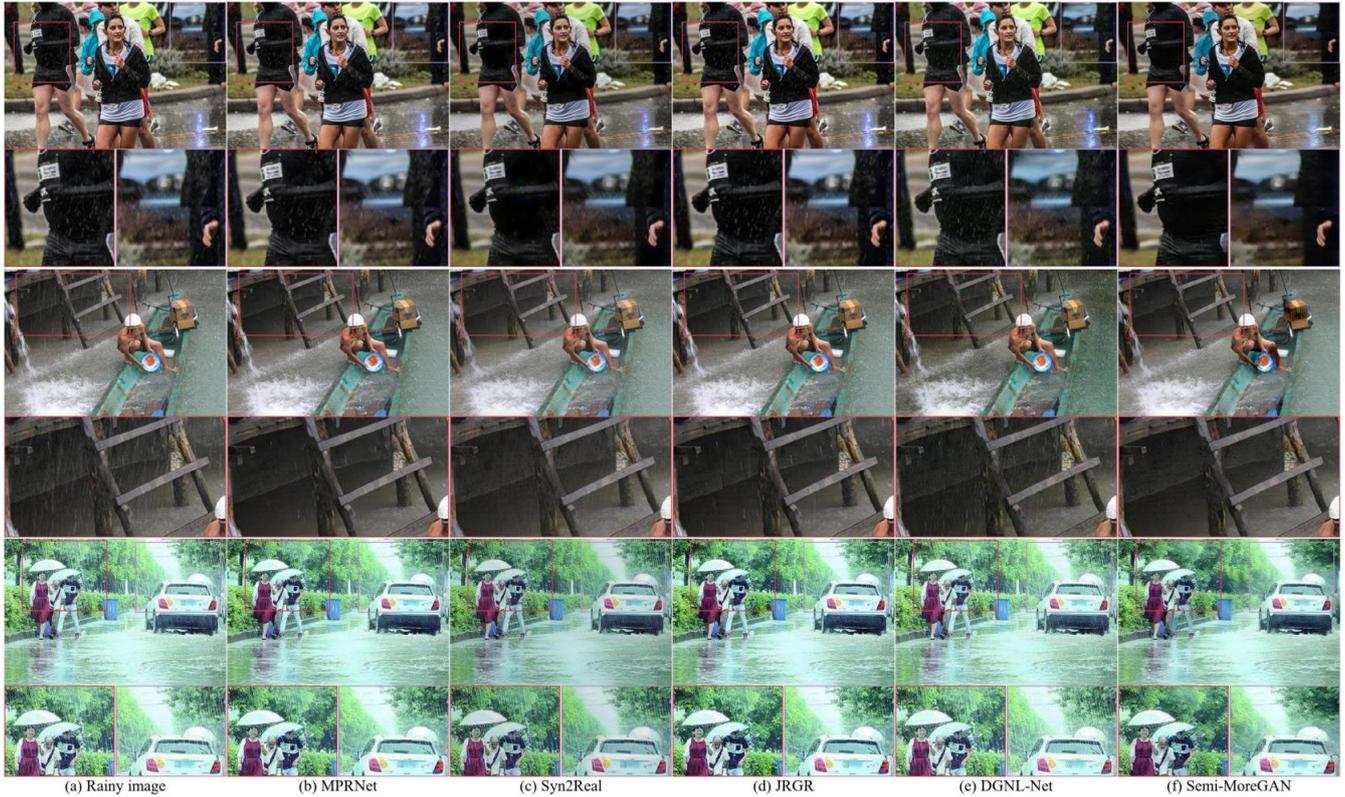
According to the model of Garg and Nayar [GN07], the visual intensity and presence of rain streaks and rainy haze depend on the scene depth, thus it is reasonable to develop a depth prediction network to predict depth maps, which guides the process of mixture of rain removal. To verify validity of our attentional depth prediction

**Table 5:** Comparison of depth prediction results. Note that monocular depth prediction methods DORN [FGW\*18], DVSO [YWSC18] and MDPGAN [CKBP18] are trained on RainCityscapes dataset.

Method	DGNet-Net	DORN	DVSO	MDPGAN	Semi-MoreGAN
RMSE	30.22	7.10	3.69	5.70	3.15
PSNR	32.84	34.21	35.29	34.96	35.67

network (ADPN), we incorporate ADPN into PReNet [RZH\*19], Syn2Real [YSP20] and DGNet-Net [HZW\*21]. The results are shown in Table 4, it is observed that PReNet [RZH\*19], Syn2Real [YSP20] and DGNet-Net [HZW\*21] improve the quantitative results significantly on the testing set of RainCityscapes. Especially, the PSNR of Syn2Real [YSP20] gains more than 2.35dB.

To further explore the impact of depth prediction accuracy on mixture of rain removal, we adopt a commonly accepted evaluation indicator RMSE [EPF14] to evaluate the results of DGNet-Net [HZW\*21], Semi-MoreGAN and three monocular depth prediction methods (i.e., DORN [FGW\*18], DVSO [YWSC18] and MDPGAN [CKBP18]) as shown in Table 5. It is observed that the RMSE of Semi-MoreGAN is lower than 27.07m compared with DGNet-Net [HZW\*21]. Fig. 4 shows more qualitative visualizations between Semi-MoreGAN and DGNet-Net [HZW\*21]. Furthermore,



**Figure 2:** Visual comparison results on dense rain accumulation images. Note that all models are trained on Rain200H&MOR-Rain200.



**Figure 3:** Deraining comparison of Semi-MoreGAN on real-world images. From (a) to (d) are Input, model trained on RainCityscapes&MOR-Rain200, model trained on RainCityscapes&MOR-Rain600, and model trained on RainCityscapes&MOR-Rain1000, respectively.

Our method even produces better and more accurate depth maps than explicit monocular depth prediction methods. Then, we input these depth maps into the saved generator  $G_s$  model to obtain derained images and take quantitative evaluations on the testing set of

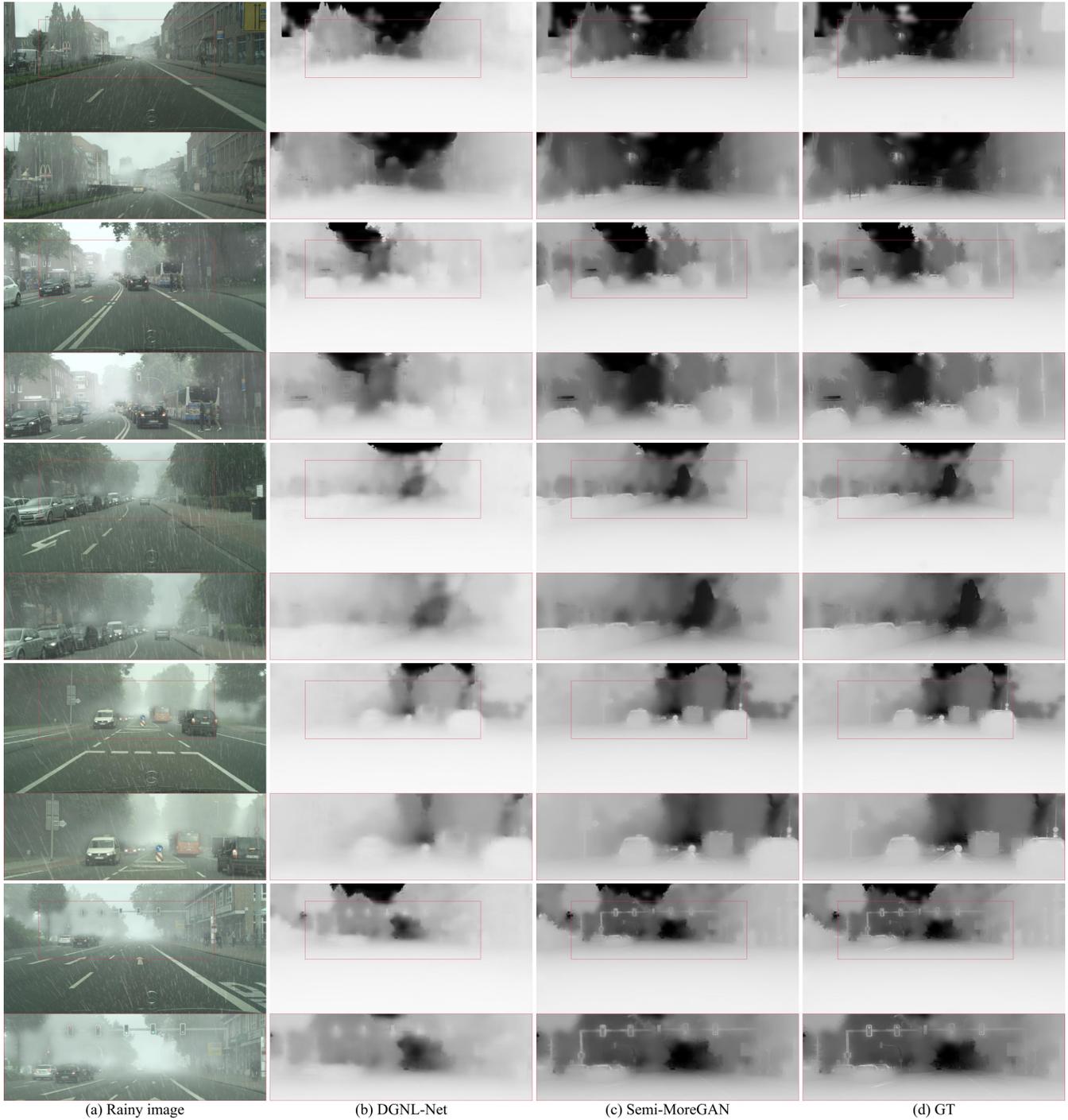
RainCityscapes. As demonstrated in Table 5, the depth map with lower RMSE will obtain higher PSNR. These experiments verify that the robust and accurate depth map from ADPN can improve performance and generalization capability in mixture of rain images.

### 5. Application

To provide further evidence that the visibility enhancement of Semi-MoreGAN could be helpful for computer vision applications, we employ Google Vision API to evaluate our deraining results. As can be seen in Fig. 5 (a-b), the Google API can recognize most objects in the derained image rather than the original rainy image. Especially, the scores of the hat and T-shirt are improved by 13% and 15% after deraining by our Semi-MoreGAN, respectively. Moreover, we also employ Google API to test 100 real rainy and corresponding derained images by our Semi-MoreGAN. As demonstrated in Fig. 5 (c), our approach has greatly improved the number of recognized object labels and the accuracy of detection.

### References

[CKBP18] CS KUMAR A., BHANDARKAR S. M., PRASAD M.: Monocular depth prediction using generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (2018), pp. 300–308. 2



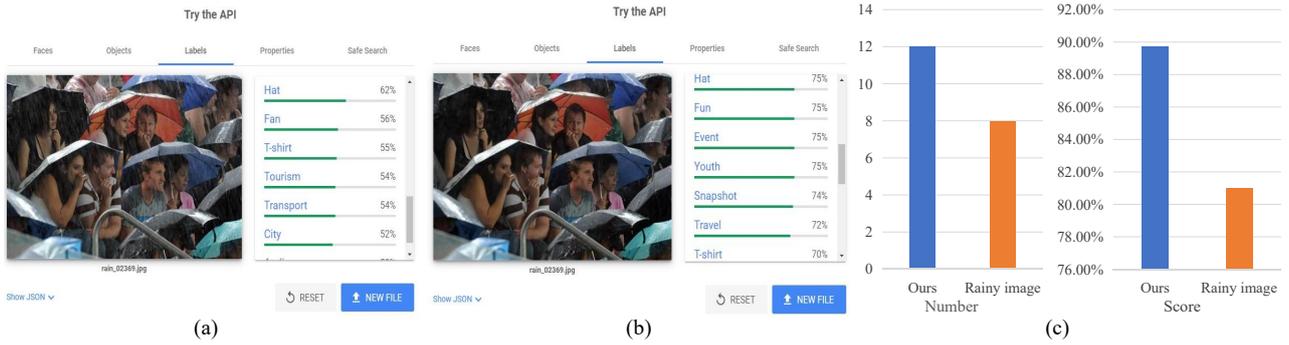
**Figure 4:** Visualizing the predicted depth maps by DGNL-Net [HZW\*21] and our Semi-MoreGAN. Note that DGNL-Net only extracts coarse depth maps and our method can obtain more accurate depth maps.

[EPF14] EIGEN D., PUHRSCH C., FERGUS R.: Depth map prediction from a single image using a multi-scale deep network. *arXiv preprint arXiv:1406.2283* (2014). 2

[FGW\*18] FU H., GONG M., WANG C., BATMANGHELICH K., TAO D.: Deep ordinal regression network for monocular depth estimation.

In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), pp. 2002–2011. 2

[GN07] GARG K., NAYAR S. K.: Vision and rain. *International Journal of Computer Vision* 75, 1 (2007), 3–27. 2



**Figure 5:** The deraining results tested on the Google Vision API. From (a)-(c): (a) object recognition result in the real-world rainy image, (b) object recognition result after deraining by our Semi-MoreGAN, and (c) column chart of improvement based on Google Vision API contains the number of object labels recognized in the rainy image and derained image, and the average score of identifying the main objects in the rainy image and derained image.

[HZW\*21] HU X., ZHU L., WANG T., FU C.-W., HENG P.-A.: Single-image real-time rain removal based on depth-guided non-local features. *IEEE Transactions on Image Processing* 30 (2021), 1759–1770. [1](#), [2](#), [4](#)

[RZH\*19] REN D., ZUO W., HU Q., ZHU P., MENG D.: Progressive image deraining networks: A better and simpler baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 3937–3946. [2](#)

[YSP20] YASARLA R., SINDAGI V. A., PATEL V. M.: Syn2real transfer learning for image deraining using gaussian processes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 2726–2736. [2](#)

[YWSC18] YANG N., WANG R., STUCKLER J., CREMERS D.: Deep virtual stereo odometry: Leveraging deep depth prediction for monocular direct sparse odometry. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2018), pp. 817–833. [2](#)