Supplementary Document for MobileDeRainGAN: An Efficient Semi-Supervised Approach to Single Image Rain Removal for Task-Driven Applications

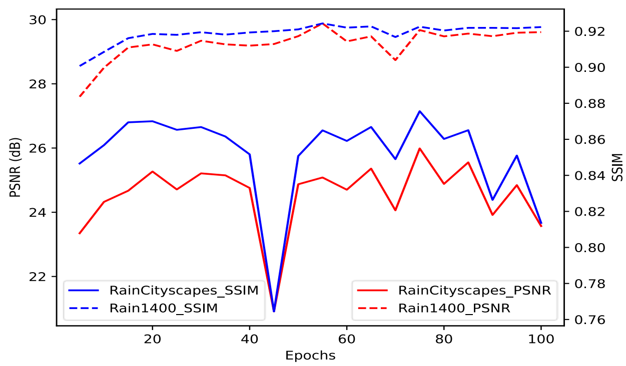

Figure 1. Performance in terms of PSNR and SSIM on the test set of Rain1400 and RainCityscapes dataset.

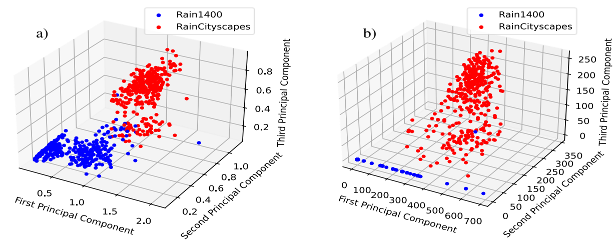

Figure 2. Principal component analysis in two-dimensional space for the latent spaces produced by the generator encoder for RainCityscapes and Rain1400 dataset during cross domain testing. a) Rainy images as inputs to the encoder. b) Clean images as inputs to the encoder.

A. Hyperparameters Used

As shown from the ablation via Fig. (8) in the paper, image resolution 256x256 was found to be the most optimal in terms of the achieving real-time performance while achieving relatively higher PSNR and mAP@0.5 scores for the inference time. Using an image resolution of 512x512 was found to significantly impact inference speed by reducing it by a factor of 3.74 (for expansion ratio = 6). The improvement in mAP@0.5 score was 0.36 and in PSNR was 0.43 dB. This slight performance improvement for a significant speed drop led us to choose 256x256 as the standard image resolution for the experiments. Similar results where observed for varying the expansion ratio, and expansion ratio = 6 was found to be an optimal choice. For the squeeze and excitation network the default reduction ratio $r = 16$ was used and we observed no significant change to model performance or speed for $r = 8$ and $r = 32$. For the latent bridge network, the hyperparameter for the weight associated with the loss λ_{bridge} (mentioned in Eq. (8)) was initialized with 0.1 and linearly increased by 0.1 every 50 epochs. The intuition behind this is that the initial predictions of rain-related latent spaces will be inaccurate and gradually improve over time as training progresses. The latent bridge network as proposed in the paper (Fig. (3)) only consists a single softmax layer to normalize the latent spaces to apply KL-Divergence Loss.

B. Experimental Details

To ensure uniformity in testing time we ignore the data loading time and just consider the forward pass time for a randomly initialized tensor of size 1x3x256x256. All the inference times were evaluated using PyTorch for uniformity with the exception of LPNet. The source code and models for LPNet made public by the authors were in TensorFlow. To train the semi-supervised model for S2VD, which is a video-based method, we use the RainCityscapes images as groups of 36. There are 295 such groups and each group has the same background with the only change being the type of rain. This is similar to how video based methods are trained on synthetic data. The training loss plots in terms of the PSNR/SSIM metrics are shown in Fig. 1. Although the epoch with the best performing model on Rain1400 (epoch 55) doesn't coincide with RainCityscapes (epoch 75), the performance on the labeled data roughly indicates the performance on the unlabeled data with few exceptions.

C. Latent space analysis

For the model to generalize between the two datasets, the encoder must be indifferent to them. However, the non-rainy background of the two datasets could be from very different distributions and must be treated differently. As established earlier, preserving the details in the non-rainy background is as important as it is to remove the rain. Thus, the network must be able to differentiate the backgrounds of the datasets while being indifferent to the rain. Fig. 2 shows the phenomenon in effect for the model trained on the Rain1400-RainCityscapes cross-domain setup. Parts a) and b) of Fig. 2 show that the latent space of both datasets condensed to two dimensions using PCA for rainy and clean images, respectively. We measure the Euclidean distance between the latent spaces for rainy and clean images as input. The mean distance between the rainy latent spaces for the labeled and unlabeled datasets is 0.85 while the same for clean images is 191.73. This demonstrates domain adaptation capability while also ensuring the preservation of details in the non-rainy layer.