Supplementary for Syn2Real Transfer Learning for Image Deraining using Gaussian Processes

Rajeev Yasarla*  Vishwanath A. Sindagi*  Vishal M. Patel
Johns Hopkins University
Department of Electrical and Computer Engineering, Baltimore, MD 21218, USA
{ryasar11, vishwanathsindagi, vpatel36}@jhu.edu

1. Introduction

Here we provide the supplementary material for the paper Syn2Real Transfer Learning for Image Deraining using Gaussian Processes. We provide more details such as network configuration, ablation experiments for different hyperparameters along with additional qualitative results.

2. Network configuration

The network consists of U-Net [5] style encoder decoder with dense connections [2], constructed using Denseblock. Denseblock contains a sequence of three $3 \times 3$ convolutional layers as shown in the Fig. 1. The proposed network consists of Encoder($h(\cdot, \theta_{enc})$) and Decoder($g(\cdot, \theta_{dec})$). Encoder($h(\cdot, \theta_{enc})$) with the following sequence: Conv2d $3 \times 3$($3,16$)-Denseblock($16,32$)-Downsample-Denseblock($32,32$)-Downsample-Denseblock($32,32$)-Downsample-Denseblock($32,64$)-Denseblock($64,64$)-Denseblock($64,64$)

Decoder($g(\cdot, \theta_{dec})$) consists of the following sequence: Denseblock($64,32$)-Upsample-Denseblock($32,32$)-Upsample-Denseblock($32,16$)-Upsample-Conv2d $3 \times 3$($16,3$),

where Conv2d $3 \times 3(m, n)$ is a $3 \times 3$ convolutional layer with $m$ input channels and $n$ output channels, Denseblock($m, n$) means Denseblock with $m$ input channels and $n$ output channels.

3. Ablation study: Hyperparameters

Here, we provide the results of the ablation experiments conducted to analyze the effect of different hyperparameters: $\lambda_{\text{unsup}}$ (see Eq. 18), number of nearest neighbours: $N_n$ and $N_f$ (see Eq. 16).

Table 1 illustrates the performance of the proposed method for different values of $\lambda_{\text{unsup}}$. For this experiment, we used 40% of data as labeled and rest was used as unlabeled. It can be observed that the proposed method is fairly robust to different values of $\lambda_{\text{unsup}}$.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>$D_c$</th>
<th>$D_u$</th>
<th>Metrics</th>
<th>$\lambda_{\text{unsup}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rain200H</td>
<td>40%</td>
<td>60%</td>
<td>PSNR</td>
<td>25.98, 26.28, 26.34, 26.43</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SSIM</td>
<td>0.828, 0.833, 0.833, 0.835</td>
</tr>
</tbody>
</table>

Further, we also analyze the performance of the network for different values of nearest neighbours used for obtaining the pseudo-GT via the Gaussian Processes. The results for this experiment are provided in Table 2. It can be observed that the results are approximately consistent for different values of $N_n$ and $N_f$.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>$D_c$</th>
<th>$D_u$</th>
<th>Metrics</th>
<th>No. of nearest neighbors $N_n$</th>
<th>No. of farthest neighbors $N_f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rain200H</td>
<td>40%</td>
<td>60%</td>
<td>PSNR</td>
<td>$N_n=16$, $N_f=16$</td>
<td>$N_n=32$, $N_f=32$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SSIM</td>
<td>0.829, 0.827, 0.833</td>
<td>0.829, 0.827, 0.833</td>
</tr>
</tbody>
</table>

4. Additional Qualitative results

Figures 1-4 illustrate additional qualitative results on real-world and synthetic datasets for different experiments.

5. Perceptual loss

Inspired by the importance of the perceptual loss in many image restoration tasks [3, 10], we use it to further improve the visual quality of the de-rained images. The perceptual loss is feature based loss, and in our case, extracted features from layer relu1.2 of pretrained network VGG-16[6], and computed perceptual loss similar to method proposed.
Let $\Phi_{VGG}(\cdot)$ denote the features obtained using the VGG16 model [6], then the perceptual loss is defined as follows:

$$\mathcal{L}_p = \|\Phi_{VGG}(y^{pred}_{i}) - \Phi_{VGG}(y_{i})\|_2^2,$$  \hspace{1cm} (1)

where $y^{pred}_{i} = g(z, \theta_{dec})$ is the predicted output, $y_{i}$ is the ground-truth, $z = h(x, \theta_{enc})$, and $x$ is rainy image.

References

Figure 2: Qualitative results on DDN-SIRR synthetic test set.
Figure 3: Qualitative results on DDN-SIRR real-world test set
Figure 4: Results of experiments with 10% labeled data on Rain200H (a) Input rainy image (b) Using only labeled data (c) Using labeled and unlabeled data (d) Ground-Truth image
Figure 5: Results of experiments with 40% labeled data on Rain200H (a) Input rainy image (b) Using only labeled data (c) Using labeled and unlabeled data. (d) Ground-Truth image