

Supplementary Materials

Yuhui Quan¹, Mingqin Chen¹, Tongyao Pang² and Hui Ji²

¹School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China

²Department of Mathematics, National University of Singapore, 119076, Singapore

csyhquan@scut.edu.cn, csmingqinchen@mail.scut.edu.cn, matpt@nus.edu.sg and matjh@nus.edu.sg

1. Details on Partial Convolution

Partial convolution [1] is originally designed for inpainting image holes, which allows progressively filling the holes from the outside to the inside. Let $\mathbf{k} \in \mathbb{R}^Z$ be the weights of a convolution kernel and $b \in \mathbb{R}$ the corresponding bias. Let $\mathbf{f} \in \mathbb{R}^Z$ denote the feature values (pixels values) for the current convolution (sliding) window and $\mathbf{m} \in \mathbb{R}^Z$ is the corresponding binary mask. The partial convolution at every location is expressed as

$$\mathbf{f}' = 1[\|\mathbf{m}\|_1 > 0](\mathbf{k}^\top(\mathbf{f} \odot \mathbf{m}) \frac{Z}{\|\mathbf{m}\|_1} + b), \quad (1)$$

where \odot is the Hadamard's product. It can be seen that the output of the function only depends on the unmasked inputs. The scaling factor $\frac{Z}{\|\mathbf{m}\|_1}$ applies appropriate scaling to adjust for the varying amount of valid (unmasked) inputs. At the beginning, we initialize the mask \mathbf{m} such that it excludes the dropped pixels of the input Bernoulli sampled instance as well as those of the input images (*e.g.* in removing salt-and-pepper noise). After crossing the current PConv layer, we then update the mask for the next PConv layer as follows: if the convolution was able to condition its output on at least one valid input value, then we mark that location to be valid. This can be expressed as $\mathbf{m}' = 1[\|\mathbf{m}\|_1 > 0]$, which can be easily implemented as a part of forward pass. See [1] for more details.

2. Proof of Proposition 1

Proof. Rewrite the loss function as follows:

$$\begin{aligned} \sum_{m=1}^M \|\mathcal{F}_\theta(\hat{\mathbf{y}}_m) - \bar{\mathbf{y}}_m\|_{\mathbf{b}_m}^2 &= \sum_{m=1}^M \|\mathcal{F}_\theta(\hat{\mathbf{y}}_m) - \mathbf{y}\|_{\mathbf{b}_m}^2 = \sum_{m=1}^M \|\mathcal{F}_\theta(\hat{\mathbf{y}}_m) - (\mathbf{x} + \mathbf{n})\|_{\mathbf{b}_m}^2 \\ &= \sum_{m=1}^M \|\mathcal{F}_\theta(\hat{\mathbf{y}}) - \mathbf{x}\|_{\mathbf{b}_m}^2 + \sum_{m=1}^M \|\mathbf{n}\|_{\mathbf{b}_m}^2 - 2 \sum_{m=1}^M ((\mathbf{1} - \mathbf{b}_m) \odot \mathbf{n})^\top (\mathcal{F}_\theta(\hat{\mathbf{y}}_m) - \mathbf{x}) \\ &= \sum_{m=1}^M \|\mathcal{F}_\theta(\hat{\mathbf{y}}) - \mathbf{x}\|_{\mathbf{b}_m}^2 + \sum_{m=1}^M \|\mathbf{n}\|_{\mathbf{b}_m}^2 - 2\mathbf{n}^\top \left(\sum_{m=1}^M (\mathbf{1} - \mathbf{b}_m) \odot (\mathcal{F}_\theta(\hat{\mathbf{y}}_m) - \mathbf{x}) \right). \end{aligned} \quad (2)$$

Regarding the second term in (2), its expectation is

$$\mathbb{E}_{\mathbf{n}} \left[\sum_{m=1}^M \|\mathbf{n}\|_{\mathbf{b}_m}^2 \right] = \mathbb{E}_{\mathbf{n}} \left[\sum_{m=1}^M \|(\mathbf{1} - \mathbf{b}_m) \odot \mathbf{n}\|_2^2 \right] = \sum_{m=1}^M \|(\mathbf{1} - \mathbf{b}_m) \odot \boldsymbol{\sigma}\|_2^2 = \sum_{m=1}^M \|\boldsymbol{\sigma}\|_{\mathbf{b}_m}^2. \quad (3)$$

Regarding the last term in (2), for simplicity we define

$$\mathbf{r} = \sum_{m=1}^M (\mathbf{1} - \mathbf{b}_m) \odot (\mathcal{F}_\theta(\hat{\mathbf{y}}_m) - \mathbf{x}) = \sum_{m=1}^M (\mathbf{1} - \mathbf{b}_m) \odot (\mathcal{F}_\theta(\mathbf{b}_m \odot \mathbf{x} + \mathbf{b}_m \odot \mathbf{n})) - \mathbf{x}. \quad (4)$$

Note that $\mathcal{F}_\theta(\mathbf{b}_m \odot \mathbf{x} + \mathbf{b}_m \odot \mathbf{n})$ contributes to $\mathbf{r}(i)$ only if $\mathbf{b}_m(i) = 0$. But in this case, $\mathbf{n}(i)$ is erased by $\mathbf{b}_m(i)$. This means that $\mathbf{n}(i)$ has no contribution to $\mathbf{r}(i)$. Together with that $\mathbf{n}(i)$ is independent of $\mathbf{n}(j)$ for any $i \neq j$, we can conclude that $\mathbf{r}(i)$ is independent to $\mathbf{n}(i)$ for all i . Therefore, we have

$$\mathbb{E}_{\mathbf{n}}[\mathbf{n}^\top \mathbf{r}] = (\mathbb{E}_{\mathbf{n}}[\mathbf{n}])^\top (\mathbb{E}_{\mathbf{n}}[\mathbf{r}]) = 0. \quad (5)$$

Combining (2), (3) and (5) yields

$$\begin{aligned} \mathbb{E}_{\mathbf{n}} \left[\sum_{m=1}^M \|\mathcal{F}_\theta(\hat{\mathbf{y}}_m) - \bar{\mathbf{y}}_m\|_{\mathbf{b}_m}^2 \right] &= \mathbb{E}_{\mathbf{n}} \left[\sum_{m=1}^M \|\mathcal{F}_\theta(\hat{\mathbf{y}}) - \mathbf{x}\|_{\mathbf{b}_m}^2 \right] + \mathbb{E}_{\mathbf{n}} \left[\sum_{m=1}^M \|\mathbf{n}\|_{\mathbf{b}_m}^2 \right] - 2\mathbb{E}_{\mathbf{n}}[\mathbf{n}^\top \mathbf{r}] \\ &= \sum_{m=1}^M \|\mathcal{F}_\theta(\hat{\mathbf{y}}) - \mathbf{x}\|_{\mathbf{b}_m}^2 + \sum_{m=1}^M \|\sigma\|_{\mathbf{b}_m}^2. \end{aligned} \quad (6)$$

The proof is done. \square

3. More Results on Blind Gaussian Denoising

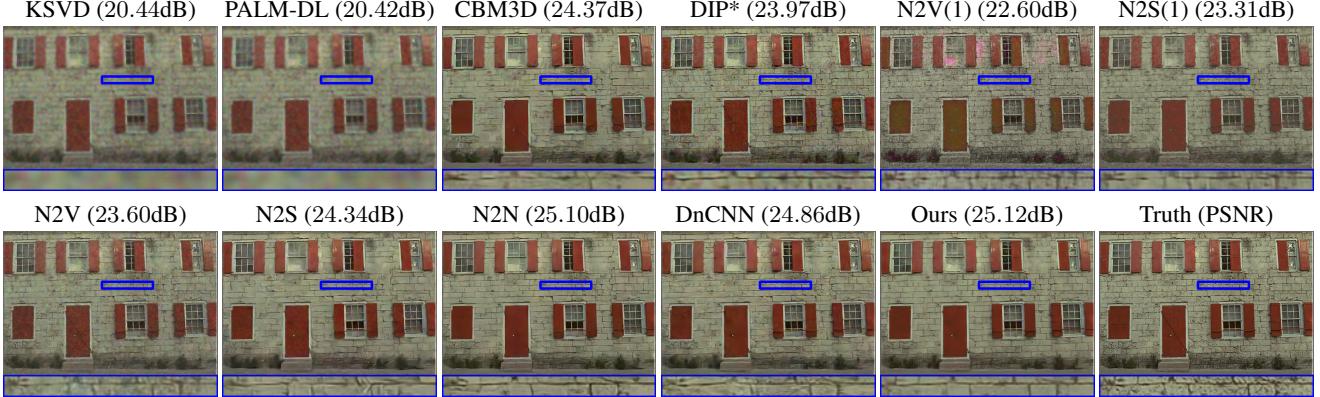


Figure 1: Visual results of blind AWGN denoising on image 'Kodim01' of Set9 with noise level $\sigma = 75$.

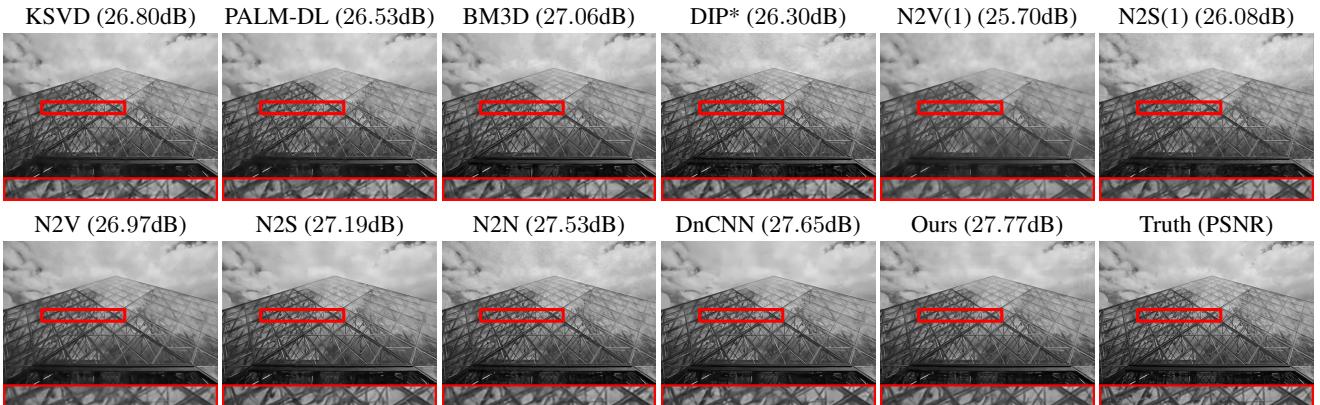


Figure 2: Visual results of blind AWGN denoising on image '223061' on BSD68 with noise level $\sigma = 25$.

4. More Results on Removal of Real-World Image Noise

Due to space limitation, the quantitative results of N2V(1) and N2S(1) are not listed in Table 2 in our main paper. The following are their results. (a) N2V(1): PSNR=34.14dB, SSIM=0.95; (b) N2S(1): PSNR=34.69dB, SSIM=0.97. Also, regarding the visual comparison in Fig. 3 in our main paper, the results of some methods are not presented. For completeness, we show the results of all compared methods in Fig. 3. See also Fig. 4 for one more example on visual comparison.

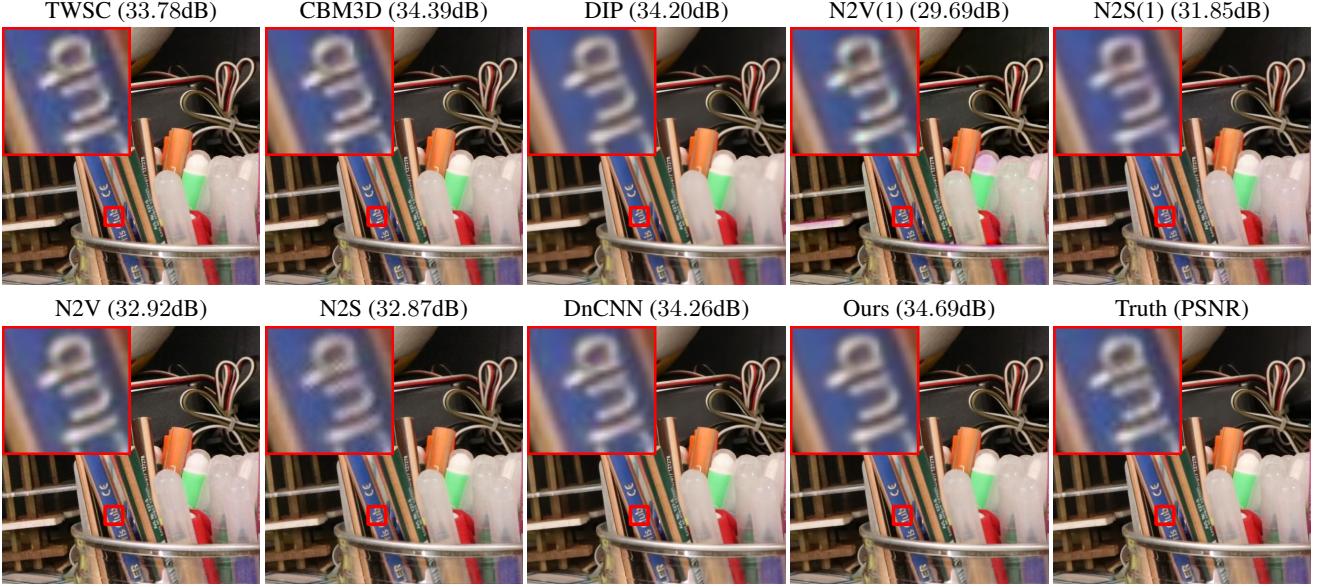


Figure 3: Denoising results on a real-world noisy image by different methods.



Figure 4: Denoising results on a real-world noisy image by different methods.

References

- [1] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proc. ECCV*, pages 85–100, 2018. 1