# Deep Priors inside an Unrolled and Adaptive Deconvolution Model: Supplementary Materials

Hung-Chih Ko, Je-Yuan Chang, and Jian-Jiun Ding

National Taiwan University
{r06942148, r07942158, jjding}@ntu.edu.tw

## S1 Implementation Detail of ADM

### S1.1 Determination Rules of Hyperparameters

In Eq.10, a deterministic hyperparameter $\beta_{\{edge,\sim edge\}}$ is adopted in proposed ADM layer and we let $\beta_{edge} = \alpha\beta$ and $\beta_{\sim edge} = \beta$ for handling edge and non-edge pixels, respectively. As mentioned in Section 3.2, $\beta$ is determined implicitly by a relationship $\sigma = \sqrt{\lambda/\beta}$. To be clear, $i$-th iteration deconvolves with $\beta_i = \lambda/{\sigma_i}^2$, where $\lambda$ *i.e.* the square of perturbed image's noise level is fixed and $\sigma_i$ *i.e.* the denoising level decreases over iterations. In this work, we set $\sigma_i$ as an exponentially decreased sequence like

$$\sigma_i = \sigma_{init}(\frac{\sigma_{end}}{\sigma_{init}})^{i/n} \tag{1}$$

where $n$ is the number of iterations (for example, $n = 30$ in Fig.2; $n = 8$ in UDN8; $n = 10$ in UDN10) and $\sigma_{\{init,end\}} = \{49, 5\}$ which is set empirically. Therefore, $\beta_i$ increases at latter ADM layers to encourage the tendency to prior regularization.

On the other hand, while $\beta_i$ is implicitly determined at each iteration, $\alpha$ needs to be adjusted accordingly since the discrepancy of restored pixels between edge and non-edge region increases during optimization process. Therefore, we set $\alpha_i$ as an exponentially decreased sequence that follows a relationship

$$\alpha_i = \alpha_{init}(\frac{\alpha_{end}}{\alpha_{init}})^{i/n} \tag{2}$$

$$\alpha_{\{init,end\}} = 0.9 - c_{\{init,end\}}\frac{\sigma_{noise} - \sigma_{min}}{\sigma_{max} - \sigma_{min}} \tag{3}$$

where $\sigma_{noise}$ is the noise level of perturbed image; $\sigma_{\{min,max\}} = \{1\%, 10\%\}$ is set for the tested noise range in our experiment. The subtracting value $c_{\{init,end\}} = \{0.3, 0.6\}$ are set by heuristics. Intuitively, when $\sigma_{noise}$ becomes high, the interval between $\sigma_{init}$ and $\sigma_{end}$ increases as well. Hence, for the solution on edge and non-edge region, the discrepancy derived from ADM are emphasized, which is designed under an intention to encourage edge preservation when noise are dominant.

### S1.2 The Edge-Awareness Criterion

In this paper, an edge-awareness criterion is adopted to determine the edge region from image $I$. It depends on the value of a multi-scaled Gaussian-smoothed edge detector which takes the maximum of edge response over a set of Gaussian-smoothed images $S_i = G_{\sigma_i} \otimes I$ to detect edges on different scales, where $G_{\sigma_i}$ is the Gaussian filter with scale $\sigma_i$ and we set $\sigma_i^2 = \{0.5, 2.5, 10, 50\}$. In short, the final edge response is defined as

$$E = \max_i \left\{ \sqrt{(\nabla_x S_i)^2 + (\nabla_y S_i)^2} \right\} \tag{4}$$

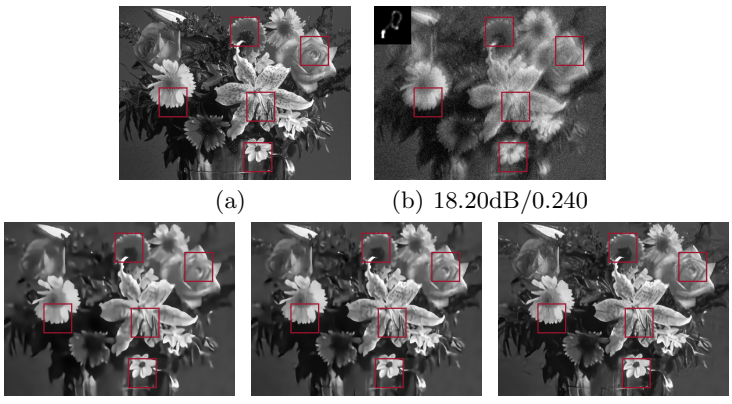
where $\nabla_{\cdot}$ is a gradient operator along $x$ or $y$ direction. Those pixels with $E$ greater than a threshold (0.05) are be classified as edge, while others not.

## S2 Combination of ADM and UDN results

In this section, we introduce a linear combination method to merge the restoration results from the two proposed methods of the ADM and the UDN into a single output.

### S2.1 Motivated by Visual Outcomes

In our simulation like an example displayed in Fig. 1, we have already found the introduction of ADMs into the iterative deconvolution process can successfully alleviate the over-smoothness that generally observed in IRCNN [1]; thus, a significant improvement is observed in our simulation results. On the other hand, our second proposed framework, UDNs, rather restore more detail information compared to ADM. However, the performance is not always superior than ADM since the emergence of artifacts in smoothed regions.



(a) (b) 18.20dB/0.240

(c) 25.95dB/0.726 (d) 26.52dB/0.759 (e) 26.43dB,0.765

**Fig. 1.** A deblurring case in Set14[2]/Levin[3] dataset. (a) ground truth (b) noisy and blurry observation ($\sigma = 5\%$) and restoration outputs solved by (c) IRCNN [1] (d) ADM and (e) UDN.

By an empirical observation on patches shown in Fig. 2 and Fig. 3, we regard the ADMs can preserve accurate structural information; for example, the edges on flower structures and the borders between foreground and background. However, as a trade-off, the restoration outputs have also lost a good portion of details originally observed in ground truth. In contrast, UDNs especially excel on restoring high frequency components like the fine textures on petals, but some aliasing structures are also appeared in flat regions. Therefore, we merge two outputs into one single image by weighted combination in frequency domain.

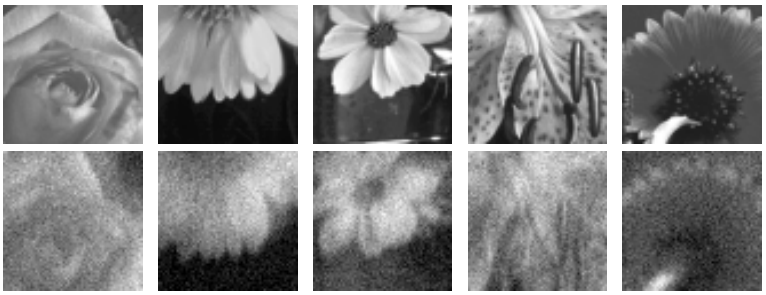


**Fig. 2.** Zoomed-in patches in (top) Fig. 1(a) and (bottom) Fig. 1(b).

**Fig. 3.** Zoomed-in patches in (top) Fig. 1(c), (mid) Fig. 1(d) and (bottom) Fig. 1(e)

### S2.2 Method

With the aim to retrieve textural information from UDN and emphasize the structural components which restored by ADM, we design a weighting mask that places its minimum at the center and increase its values when closing to the high frequency bands. The mask can be describes as (5) and a corresponding

visualization is plotted in Fig. 4(a).

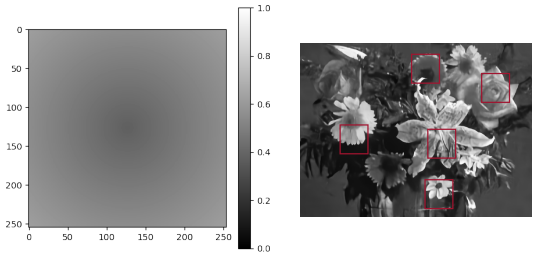
$$M(x,y) = \frac{1}{1 + exp\left(-\left(\frac{\sqrt{x^2+y^2)}}{\sqrt{(\frac{h}{2})^2+(\frac{w}{2})^2}} - \frac{1}{2}\right)\right)} \tag{5}$$

where $x$ and $y$ denote the coordinates which range from $[\frac{-h}{2}, \frac{h}{2}]$ and $[\frac{-w}{2}, \frac{w}{2}]$. The mask is designed in essence like a sigmoid function such that it gives $1/2$ when input equals to 0, so that it tends to place equal weights on both outputs in intermediate bands. One may also notes that the maximum and minimum are fixed for any image size since the coordinates are normalized in advance. The rationale is to prevent overly inclination to a specific output, which usually leads to inferior restoration quality in our empirical studies. After a weighting mask is determined, two outputs are combined as follows:

$$X(x,y) = \mathcal{F}^{-1}\left\{M(x,y)\mathcal{F}\{X_{UDN}(x,y)\} + (1 - M(x,y))\mathcal{F}\{X_{ADM}(x,y)\}\right\} \tag{6}$$

where $X$, $X_{ADM}$ and $X_{UDN}$ indicate the combined result, restoration outputs from ADM and UDN, respectively. $\mathcal{F}(.)$ denotes the Fourier transform operator. It is worth noting that the additional computational complexity is required by merely 3 FFT operations, which implies

The corresponding combined results displayed in Fig. 4(b) and Fig. 5 have achieved better performance than that of ADM or UDN on both PSNR and SSIM. Especially, the textures on petals in Fig. 5 (a-b) are well preserved; the artifacts in Fig. 5 (e) are mitigated as well.




(a) (b) 26.72dB/0.772

**Fig. 4.** (a) The mask $M(x,y)$ used in combination and (b) the combined restoration result.

(a) (b) (c) (d) (e)

**Fig. 5.** Zoomed-in patches in Fig. 4(b).

### S2.3 Quantitative Comparison

We have designed a simple and fast combination method that can not only merge two restoration outputs effectively but also take advantages of both structural and textural information. Besides, the preliminary result has shown a significant progress on restoration quality. Moreover, such an improvement is not only observed in this single case, but also works in other simulation cases as well; for example, the averaged PSNR and SSIM for Set14[2]/Levin[3] and Sun *et al.*[4]/Pan *et al.*[5] (identical to the simulation dataset described in paper) displayed in Table. 1 and Table. 2 have shown the superiority of ADM-UDN combination. In general, ADM-UDN has significantly enhanced the restoration quality at a wide range of noise levels. Even some of the performance is not better than either that of ADM or UDN in specific circumstances, ADM-UDN has a good ability to ensure a reasonable quality and only leads to neglectable loss of performance in the end.

**Table 1.** Comparison on the Set14[2]/Levin[3] dataset.

| $\sigma$ | IQA | IDD-BM3D | IRCNN | ADM | UDN | ADM-UDN |
|---|---|---|---|---|---|---|
| 1% | PSNR | 31.76 | 31.57 | 31.62 | 31.20 | **31.87** |
| | SSIM | 0.879 | 0.876 | 0.879 | 0.869 | **0.884** |
| 3% | PSNR | 27.79 | 27.63 | 27.84 | 28.12 | **28.32** |
| | SSIM | 0.765 | 0.760 | 0.774 | 0.785 | **0.787** |
| 5% | PSNR | 26.00 | 25.97 | 26.52 | 26.51 | **26.83** |
| | SSIM | 0.705 | 0.697 | 0.718 | 0.729 | **0.733** |
| 10% | PSNR | 23.80 | 23.99 | **24.49** | 23.84 | 24.48 |
| | SSIM | 0.625 | 0.624 | **0.657** | 0.625 | 0.649 |

**Table 2.** Comparison on the Sun *et al.*[4]/Pan *et al.*[5] dataset.

| $\sigma$ | IQA | IDD-BM3D | IRCNN | ADM | UDN | ADM-UDN |
|---|---|---|---|---|---|---|
| 1% | PSNR | 32.65 | 32.45 | 32.51 | 32.85 | **33.00** |
| | SSIM | 0.887 | 0.880 | 0.884 | 0.900 | **0.901** |
| 3% | PSNR | 28.73 | 28.59 | 28.84 | 29.36 | **29.43** |
| | SSIM | 0.775 | 0.759 | 0.773 | **0.812** | 0.806 |
| 5% | PSNR | 27.08 | 27.11 | 27.48 | 27.83 | **27.97** |
| | SSIM | 0.714 | 0.704 | 0.724 | **0.757** | 0.753 |
| 10% | PSNR | 25.22 | 25.34 | 25.91 | 25.92 | **26.18** |
| | SSIM | 0.642 | 0.640 | 0.677 | 0.677 | **0.684** |

# S3 Additional Visual Results

## S3.1 Additional Visual Comparison for Different Deconvolution Methods

We provide additional visual comparison for several examples in Set14[2] and Sun *et al.*[4] dataset. Both datasets have included a variety of scenes like photography of portraits, landscapes and buildings or illustrations to cover widely scoped cases in practical situations. The results have shown latent clear images, synthetic blurry and noisy images, outputs restored by representative methods like a patch-based method (IDD-DM3D) [6] and a learning based plug-and-play method (IRCNN) [1], and finally, our proposed ADM, UDN and ADM-UDN. For all of the examples, our methods demonstrate evident improvement for detail preservation as well as precise edge restoration.

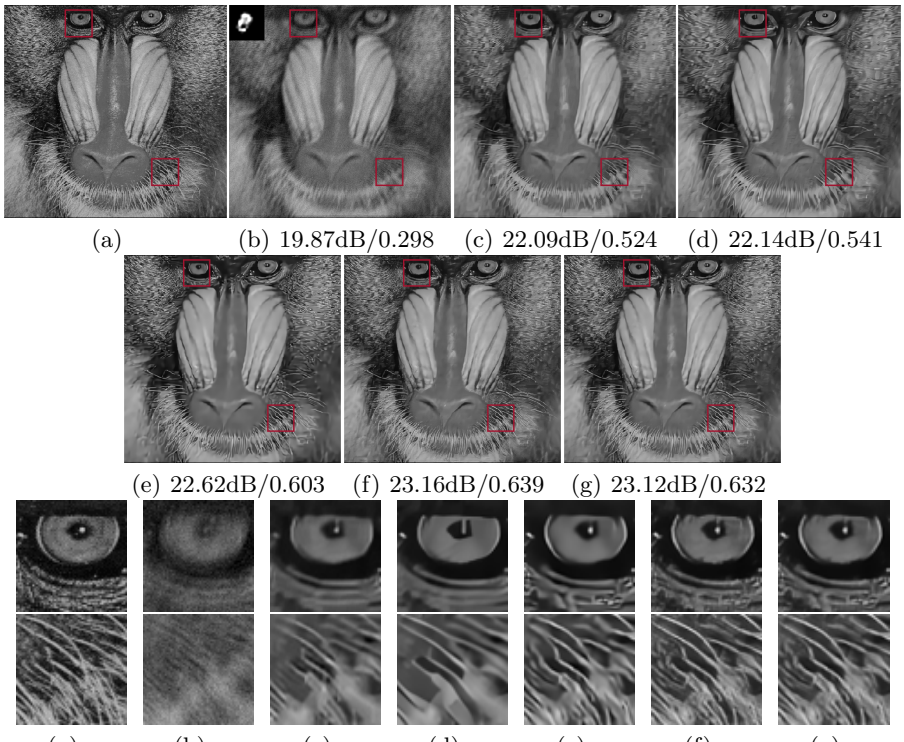



**Fig. 6.** A deblurring case in Set14 dataset [2]. (a) ground truth (b) noisy and blurry observation ($\sigma = 3\%$)(c) IDD-BM3D (d) IRCNN (e) ADM (f) UDN (g) ADM-UDN.

## References

1. Zhang, K., Zuo, W., Gu, S., Zhang, L.: Learning deep cnn denoiser prior for image restoration. In: IEEE Conference on Computer Vision and Pattern Recognition. (2017) 3929–3938
2. Zeyde, R., Elad, M., Protter, M.: On single image scale up using sparse representations. In: International Conference on Curves and Surfaces. (2010) 711–730
3. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Understanding and evaluating blind deconvolution algorithms. In: IEEE Conference on Computer Vision and Pattern Recognition. (2009) 1964–1971
4. Sun, L., Cho, S., Wang, J., Hays, J.: Edge-based blur kernel estimation using patch priors. In: IEEE International Conference on Computational Photography. (2013) 1–8
5. Pan, J., Lin, Z., Su, Z., Yang, M.H.: Robust kernel estimation with outliers handling for image deblurring. In: IEEE Conference on Computer Vision and Pattern Recognition. (2016) 2800–2808
6. Danielyan, A., Katkovnik, V., Egiazarian, K.: Bm3d frames and variational image deblurring. IEEE Transactions on Image Processing 21 (2011) 1715–1728