ISPL
Intelligent *Signal* Processing Lab

Deep Universal Blind Image Denoising

Jae Woong Soh & Nam Ik Cho

Seoul National University

ICPR 2020

Image Denoising

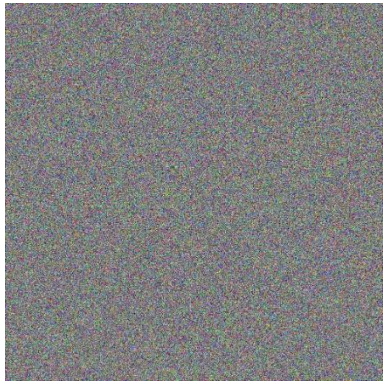
- Aims to recover the latent clean image from an observed noisy image.
- Has been a longstanding problem due to inevitable noise corruption during image acquisition.

- Degradation model
  - Clean image $\mathbf{x}$, Noisy image $\mathbf{y}$, Noise $\mathbf{n}$

$$\mathbf{y} = \mathbf{x} + \mathbf{n}$$

$=$

$+$



Traditional Methods

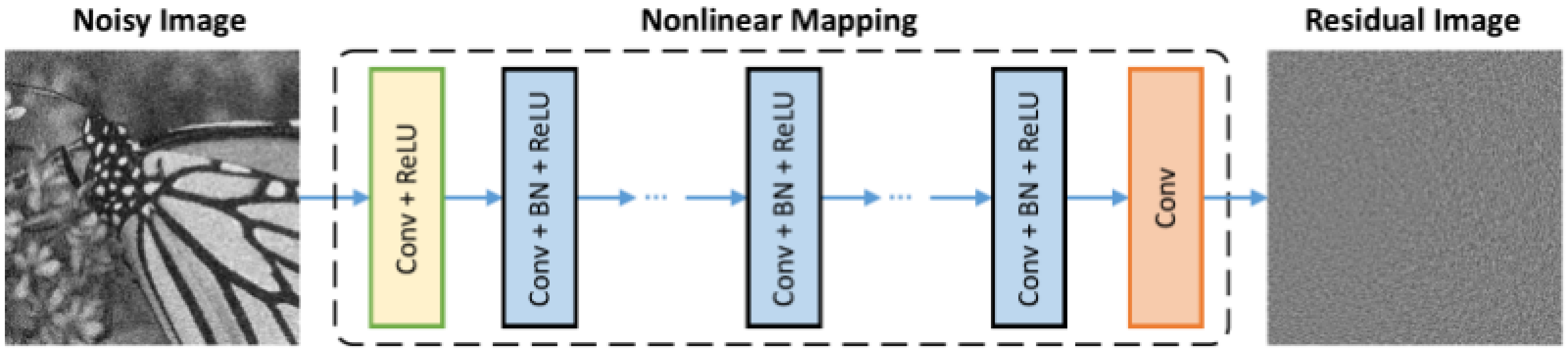
- Maximum a posteriori (MAP) inference

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} \log p(\mathbf{x}|\mathbf{y}),$$
$$= \arg\max_{\mathbf{x}} \underbrace{\log p(\mathbf{y}|\mathbf{x})}_{\text{Likelihood (Data fidelity)}} + \underbrace{\log p(\mathbf{x})}_{\text{Prior (Regularization)}}$$

- Explicitly define the data fidelity term and the image prior term.
- Superior in terms of interpretability and flexibility.
- Limited to i.i.d. Gaussian noise, requires a long computation time.

CNN-based Methods

- Convolutional Neural Networks
  - Implicitly learns the mapping based on the supervised learning.
  - Surpass the conventional non-learning methods by large margin.
  - Lack in flexibility, especially for blind denoising.
  - Handcrafted prior knowledge cannot be easily injected.



DnCNN (TIP ’17)

Proposed Method

Deep Universal
Blind Image Denoiser

Leverages the advantages of MAP inference
& power of deep learning.

Proposed Method

- Deep Universal Blind Image Denoiser (DUBD)
    - CNN-based universal blind denoiser that can handle a wide range of noise-level including spatially & spectrally varying noise.
    - Incorporates prior knowledge, further lifts the performance.
    - Outperforms other denoisers with a comparable number of params.
    - Can also be applied to real-world noisy images.

Probabilistic View

- Reformulate of the log-posterior by introducing a new random variable **c**.

$$\begin{aligned}\log p(\mathbf{x}|\mathbf{y}) &= \log \int_{\mathbf{c}} p(\mathbf{x}|\mathbf{y},\mathbf{c})p(\mathbf{c}|\mathbf{y})d\mathbf{c}, \\ &\approx \log p(\mathbf{x}|\mathbf{y},\hat{\mathbf{c}})p(\hat{\mathbf{c}}|\mathbf{y}), \\ &= \log p(\mathbf{x}|\mathbf{y},\hat{\mathbf{c}}) + \log p(\hat{\mathbf{c}}|\mathbf{y})\end{aligned}$$

- The MAP inference into two sub-problems.

$$\hat{\mathbf{c}} = \arg\max_{\mathbf{c}} \log p(\mathbf{c}|\mathbf{y}),$$

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} \log p(\mathbf{x}|\mathbf{y},\hat{\mathbf{c}})$$

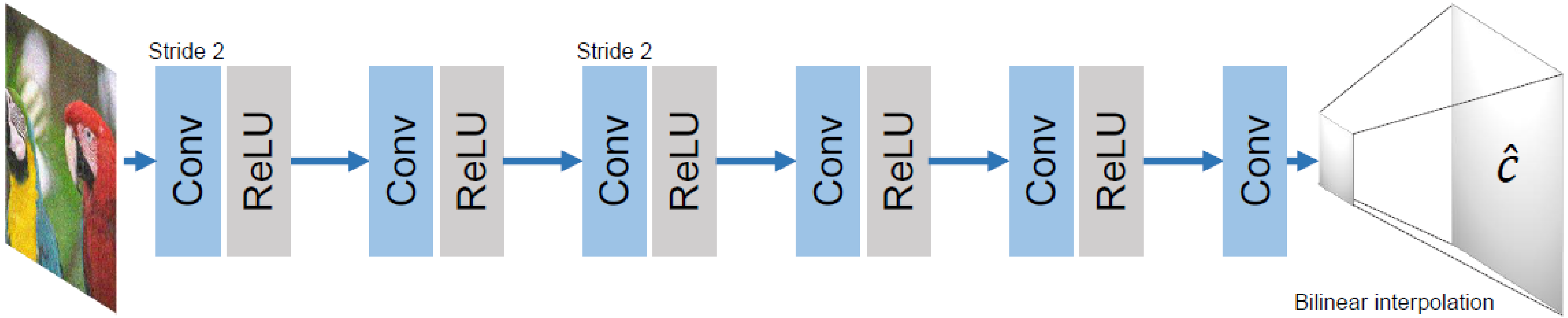
- Two neural networks for inference.

$$\hat{\mathbf{c}} \approx g_\theta(\mathbf{y})$$

$$\hat{\mathbf{x}} \approx f_\phi(\mathbf{y};\hat{\mathbf{c}})$$

Conditional Estimation Network: $g_\theta(\mathbf{y})$

- We choose the noise-level as **c**.
  - Unimodal with a sharp peak $p(\mathbf{c}|\mathbf{y})$
- Network Architecture
  - 3×3 kernels with 64 channels.

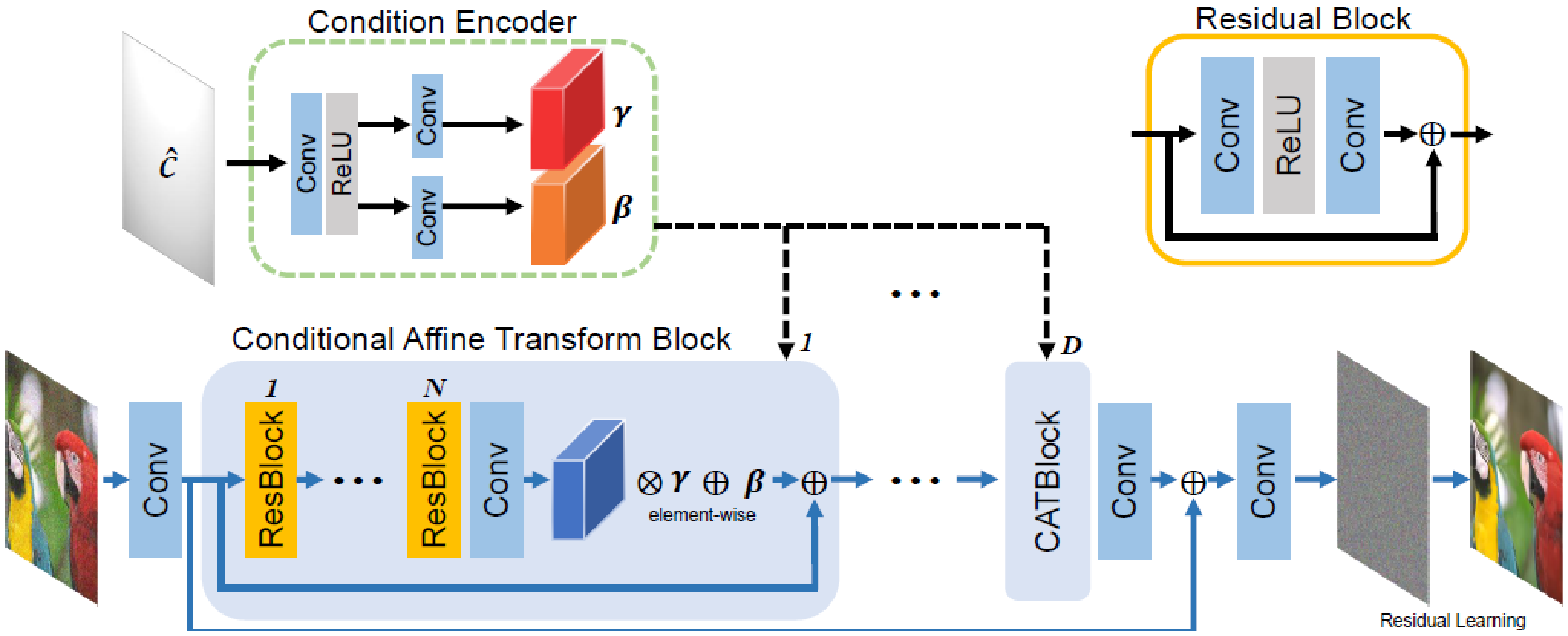


- Loss function

$$\mathcal{L}_c(\theta) = \mathbb{E}[||\sigma - g_\theta(\mathbf{y})||_2^2]$$

Tunable Denoising Network: $f_\phi(\mathbf{y};\hat{\mathbf{c}})$

- Network Architecture



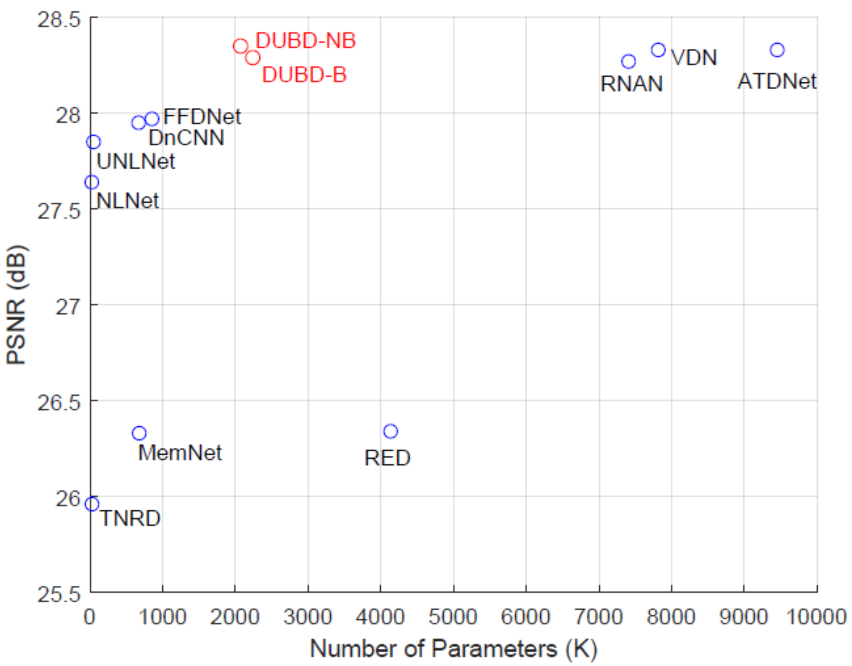
- Loss function

$$\mathcal{L}_{dn}(\phi) = \mathbb{E}[||\mathbf{x} - f_\phi(\mathbf{y};\mathbf{c})||_2^2]$$

Analysis

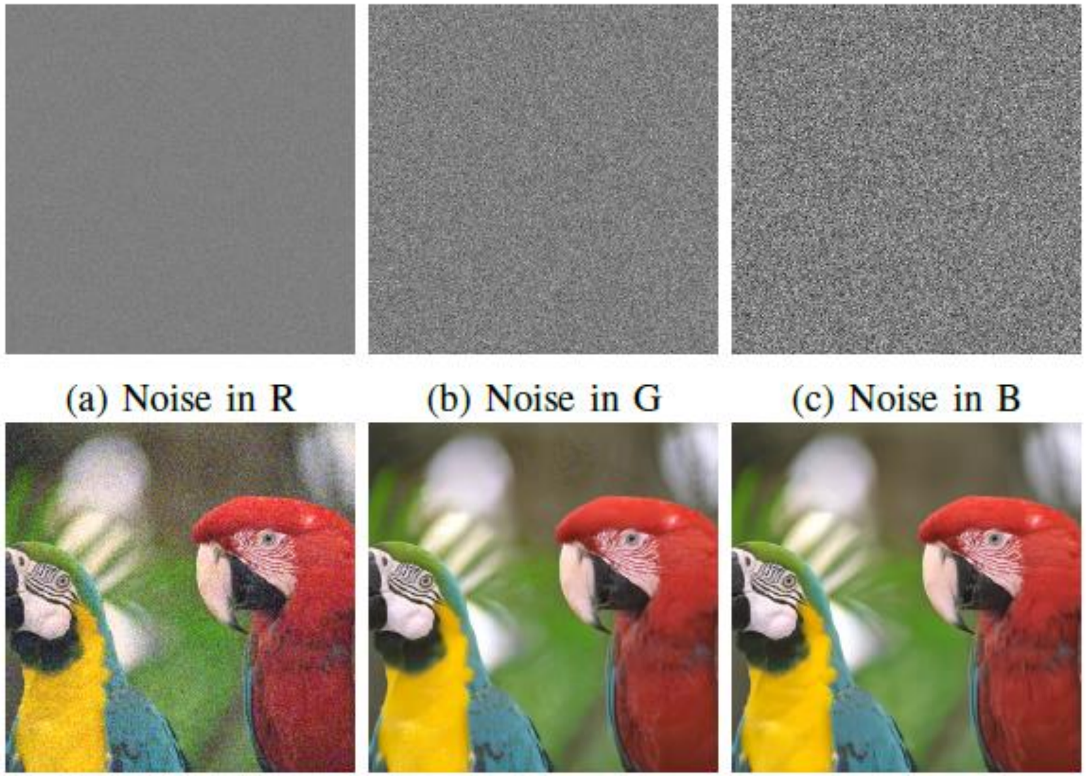
- The number of parameters

Methods	RED [11]	CDnCNN* [9]	FFDNet [10]	ATDNet* [14]	RNAN [16]	VDN* [17]	DUBD-NB (Ours)	DUBD-B* (Ours)
Parameters	4,135 K	668 K	825 K	9,453 K	7,409 K	7,817 K	2,088 K	2,239 K
PSNR (dB)	26.25	27.59	28.05	29.20	29.08	28.86	29.16	29.14



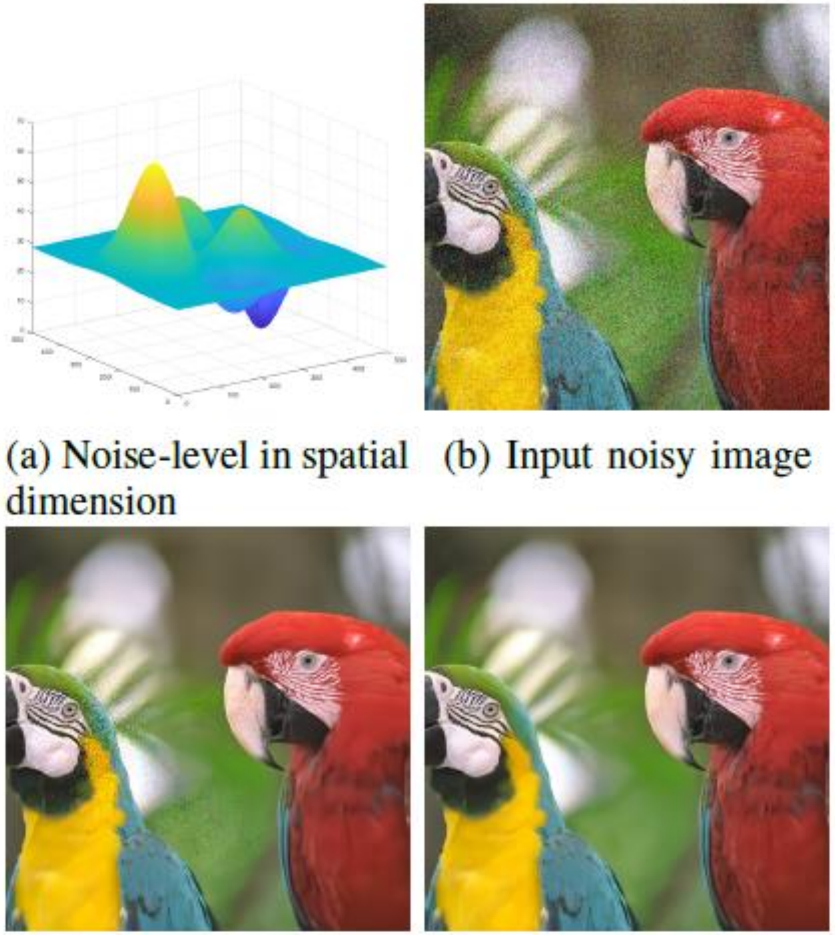
Analysis

- Dealing with spectrally-spatially variant noise



(a) Noise in R (b) Noise in G (c) Noise in B

(d) Input noisy image (e) Output with average noise-level (f) Our result

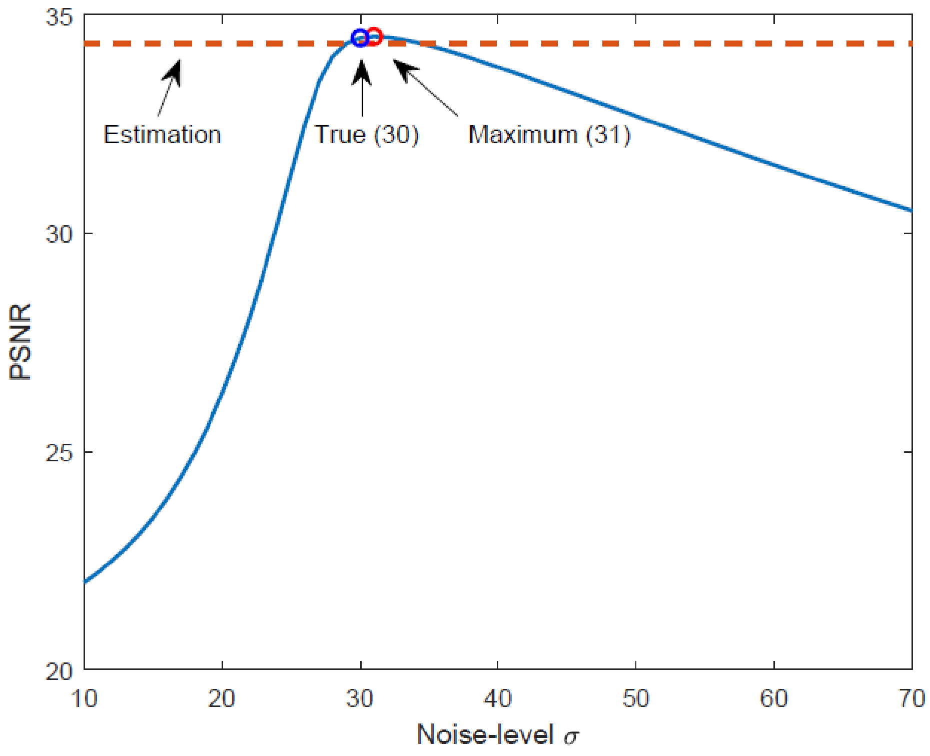


(a) Noise-level in spatial dimension (b) Input noisy image

(c) Output with average noise-level (d) Our result

Analysis

- Traversing conditional variable

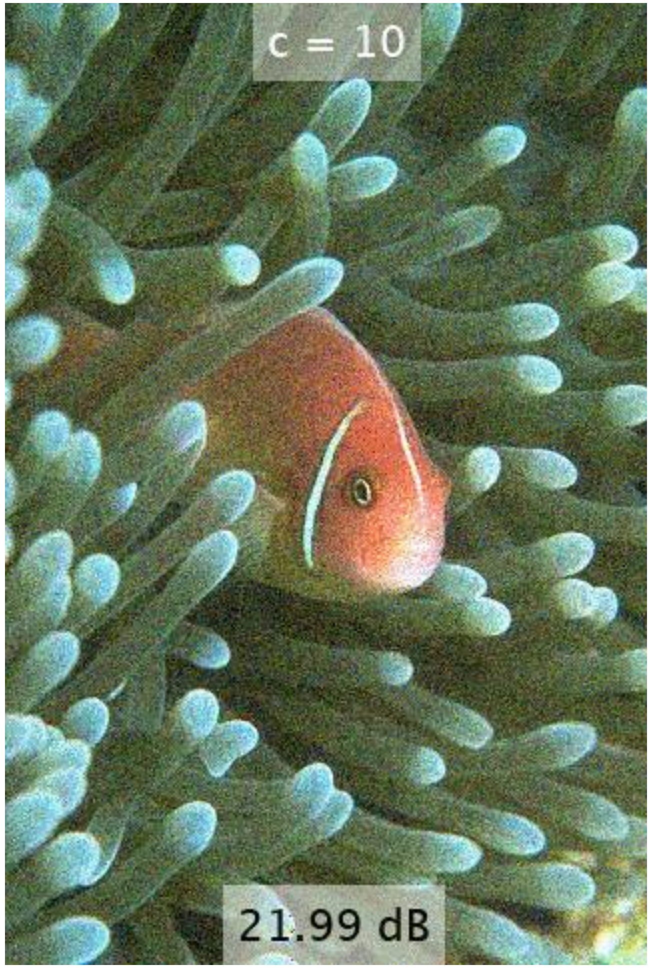


Analysis

- Traversing conditional variable

Analysis

- Traversing conditional variable

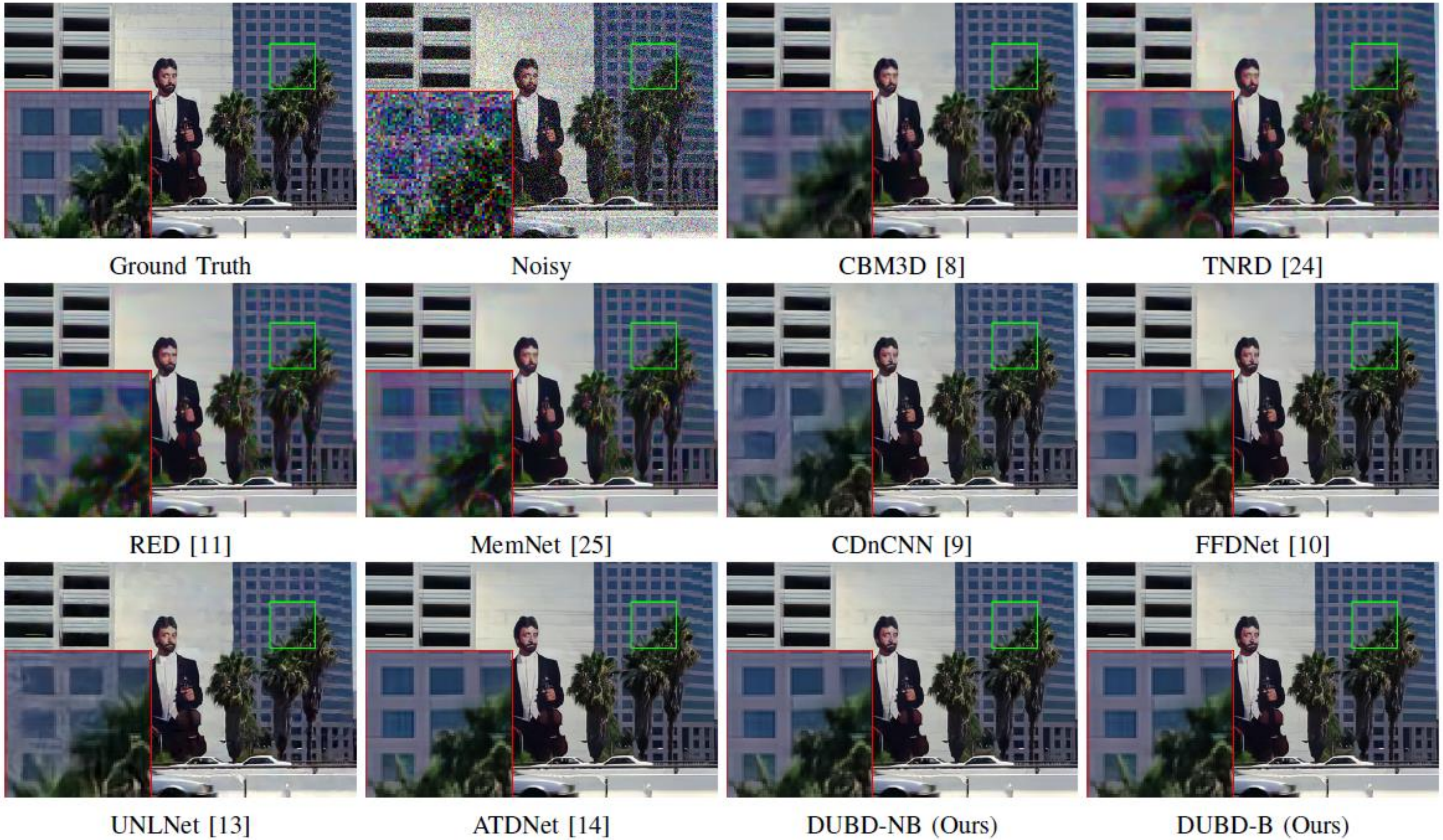


Experimental Results (PSNR)

- Results on AWGN synthetic noise.

Noise-level	Dataset	CBM3D [8]	TNRD [24]	RED [11]	MemNet* [25]	CDnCNN* [9]	FFDNet [10]	UNLNet* [13]	ATDNet* [14]	VDN* [17]	DUBD-NB (Ours)	DUBD-B* (Ours)
$\sigma = 10$	CBSD68	35.91	-	33.89	28.52	36.13	36.14	36.20	36.29	36.29	36.35	36.33
	Kodak24	36.43	-	34.73	29.70	36.46	36.69	-	36.98	36.85	37.03	37.02
	Urban100	36.00	-	34.42	29.44	34.61	35.78	-	36.31	35.97	36.32	36.23
$\sigma = 30$	CBSD68	29.73	-	28.45	28.39	30.34	30.32	30.21	30.61	30.64	30.65	30.62
	Kodak24	30.75	-	29.53	29.55	31.17	31.27	31.18	31.72	31.67	31.75	31.75
	Urban100	30.36	-	28.84	28.93	30.00	30.53	30.41	31.48	31.14	31.46	31.43
$\sigma = 50$	CBSD68	27.38	25.96	26.34	26.33	27.95	27.97	27.85	28.33	28.33	28.35	28.31
	Kodak24	28.46	27.04	27.42	27.51	28.83	28.98	28.86	29.48	29.44	29.51	29.50
	Urban100	27.94	25.52	26.25	26.53	27.59	28.05	27.95	29.20	28.86	29.16	29.14
$\sigma = 70$	CBSD68	26.00	-	25.09	25.09	25.66	26.55	-	-	26.93	26.96	26.89
	Kodak24	27.09	-	26.16	26.24	26.36	27.56	-	-	28.05	28.12	28.11
	Urban100	26.31	-	24.58	24.93	25.24	26.40	-	-	27.31	27.59	27.58

Experimental Results (Visualization)



Application to Real-World Noisy Image

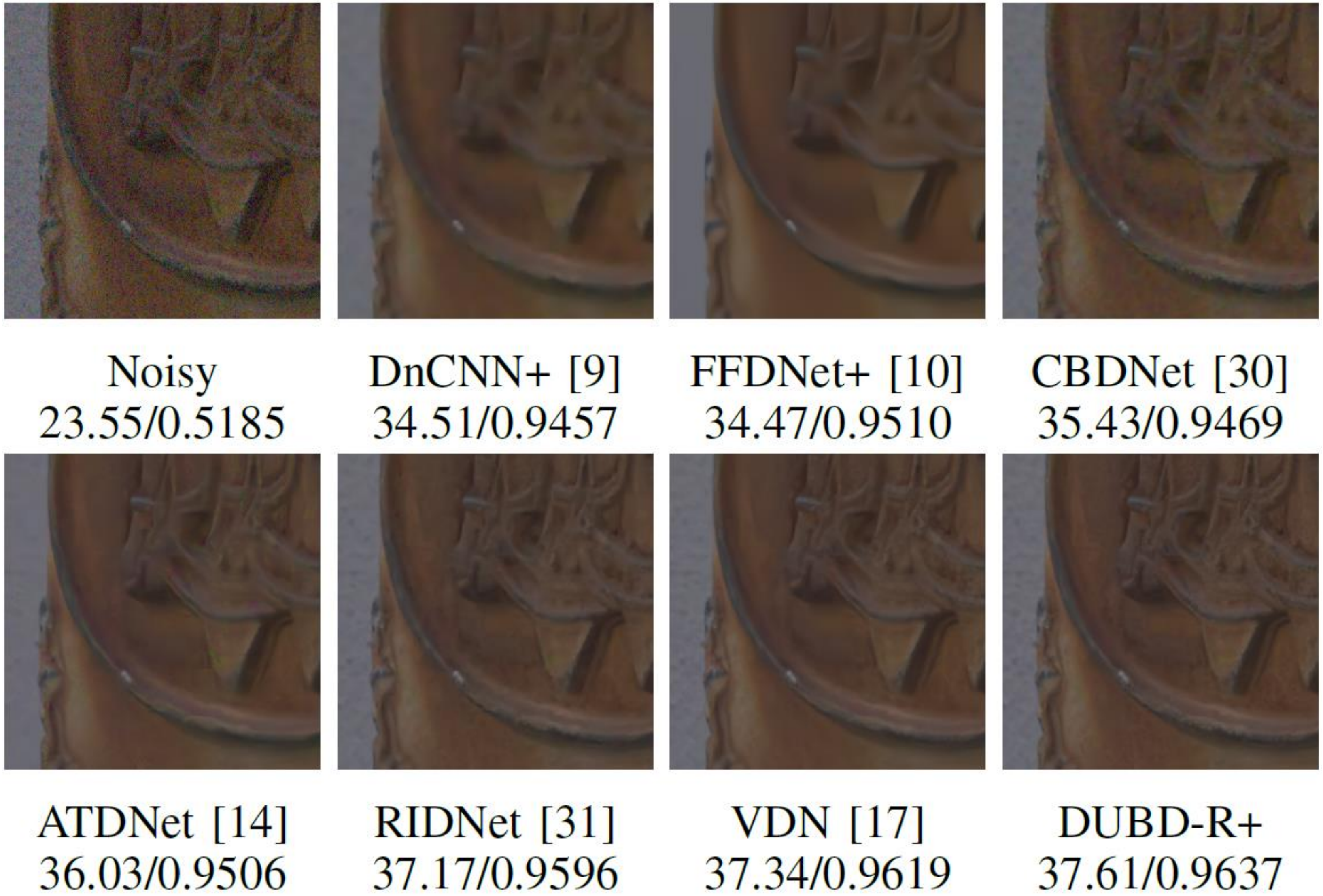
- Choice for the conditional variable

$$\mathbf{c} = Avgpool_{4\times4}(\mathbf{y})$$

- Results (DND)

Method	Parameters	PSNR	SSIM
DnCNN+ [9]	668 K	37.90	0.9430
FFDNet+ [10]	825 K	37.61	0.9415
CBDNet [30]	4,365 K	38.06	0.9421
ATDNet [14]	9,453 K	39.19	0.9526
RIDNet [31]	1,499 K	39.26	0.9528
VDN [17]	7,817 K	39.38	0.9518
DUBD-R (Ours)	2,088 K	39.38	0.9526
DUBD-R+ (Ours)	2,088 K	39.44	0.9530

Visualized Results on Real-World Noise



Conclusion

- We have proposed a CNN-based universal blind denoiser.
  - Based on splitting the original MAP problem into two sub-problems.
  - Can reduce noise from various environments.
  - Can also be manually tuned in accordance with user preference.
  - Outperforms other methods.

ISPL
Intelligent *Signal* Processing Lab

Thank You!

ICPR 2020