GAN2GAN: Generative Noise Learning for Blind Denoising with Single Noisy Images

Sungmin Cha[1], Taeeon Park [1], Byeongjoon Kim [2], Jongduk Baek [2] and Taesup Moon [1]
[1]Department of Electrical and Computer Engineering, Sungkyunkwan University
[2]School of Integrated Technology and Yonsei Institute of Convergence Technology, Yonsei University
{csm9493, pte1236, tsmoon}@skku.edu, bjkim2006@naver.com, jongdukbaek@Yonsei.ac.kr

Blind Image Denoising

- **Image denoising without *clean images***
  - **Classical denoising methods** (Ex. BM3D, WNNM)
    - **Time** and **computation** consuming
  - Recently, several **neural network based methods** are proposed
    - Consider **different settings**, such as

Alg.\ Requirements	Clean image	Noisy "pairs"	Noise model
N2N [12]	✗	✓	✗
HQ SSL [11]	✗	✗	✓
SURE [18]	✗	✗	✓
Ext. SURE [26]	✗	✓	✓
GCBD [6]	✓	✗	✗
N2V [10]	✗	✗	✗
GAN2GAN (Ours)	✗	✗	✗

We consider more challenging settings!

- **Contributions of our work**
  - Propose **"Noisy N2N"** as a core motivation
  - Devise **three components of GAN2GAN**
  - Achieve **state-of-the-art performance** in various datasets

Motivation

- **The core motivation: "Noisy N2N"**
  - consider **a single-letter Gaussian noise** setting
  - Let $Z = X + N$, in which $X \sim \mathcal{N}(0, \sigma_X^2)$ and $N \sim \mathcal{N}(0, \sigma_N^2)$
  - **The noisy observation** of **a noisy version** of X
    - $Z_1' = X' + N_1$ and $Z_2' = X' + N_2$, in which $X' = X + N_0$ and $N_0 \sim \mathcal{N}(0, \sigma_0^2)$
  - **"Noisy" N2N estimator** of Z_1' given Z_2'

$$f_{\text{Noisy N2N}}(Z_1', y) \triangleq \arg\min_f \mathbb{E}(Z_2' - f(Z_1'))^2$$
$$= \mathbb{E}(X'|Z_1') = \frac{\sigma_X^2(1+y)}{\sigma_X^2(1+y) + \sigma_N^2} Z_1'$$

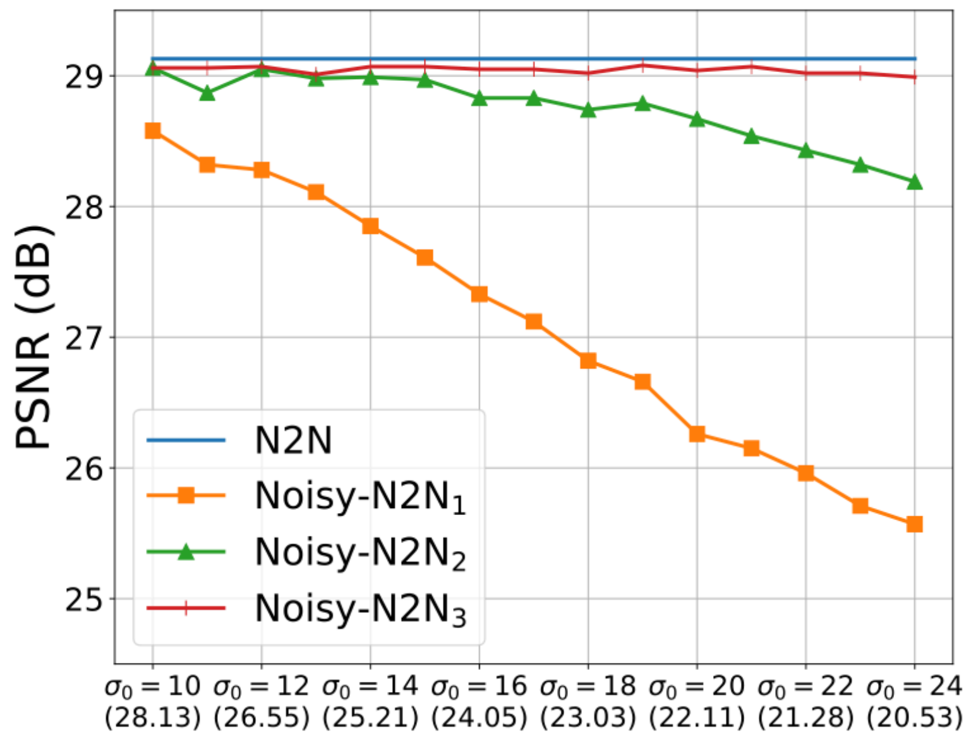
  - In which, $y \triangleq \sigma_0^2/\sigma_X^2$ and assume that $0 \leq y < 1$

Q) What happens when we use the mapping $f_{\text{Noisy N2N}}(Z_1', y)$ for estimating X given $Z = X + N$?

Theorem 1 *Consider the single-letter Gaussian setting and $f_{\text{Noisy N2N}}(Z, y)$ obtained in (2). Also, assume $0 < y < 1$. Then, there exists some y_0 s.t. $\forall y \in (y_0, 1)$, $\mathbb{E}(X - f_{\text{Noisy N2N}}(Z, y))^2 < \sigma_0^2$.*

- For a sufficiently large σ_0^2, $f_{\text{Noisy N2N}}(Z)$ gives a better estimate X than X'

- **Iterative "Noisy N2N"**
  1. Start from **a noisy estimate of** X
  2. **Simulate the noise** in image
  3. Carry out **the N2N training**
  4. **Do iteratively**



Three Components of GAN2GAN

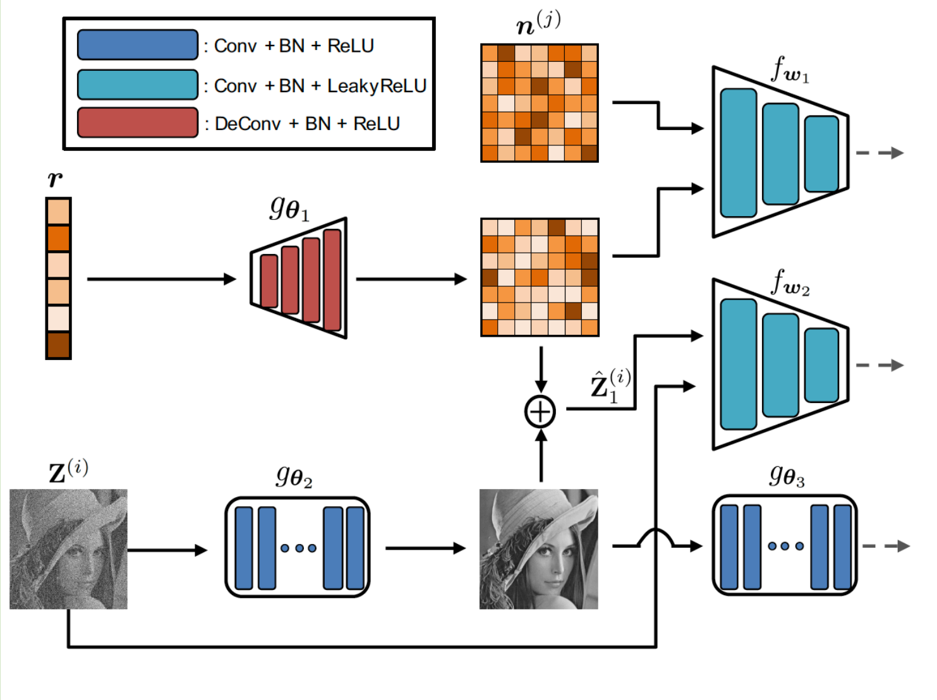
- **Notations and settings**
  - Consider the noisy image $\mathbf{Z} = \mathbf{x} + \mathbf{N}$
    - $\mathbf{x}$: clean image
    - $\mathbf{N}$: the **zero mean**, **additive** and **source-independent** noise

- **Component 1: Noise patch extraction**
  - Propose to use **the 2D discrete wavelet transform (DWT)**

$$\frac{1}{4}\sum_{k=1}^{4}\left|\underbrace{\hat{\boldsymbol{\sigma}}}_{\text{the empirical standard deviation}}(\underline{W_k(\boldsymbol{p})}) - \mathbb{E}[\hat{\boldsymbol{\sigma}}_W(\boldsymbol{p})]\right| \leq \lambda\mathbb{E}[\hat{\boldsymbol{\sigma}}_W(\boldsymbol{p})],$$

  - $\mathbb{E}[\hat{\boldsymbol{\sigma}}_W(\boldsymbol{p})] \triangleq \frac{1}{4}\sum_{k=1}^{4}\hat{\boldsymbol{\sigma}}(W_k(\boldsymbol{p}))$ and $\lambda \in (0,1)$ is hyperparameter
  - Determine $\boldsymbol{p}$ is smooth if it satisfy above rule
  - Once N^p patches are extracted from $\mathcal{D} = \{\mathbf{Z}^{(i)}\}_{i=1}^n$
    - Subtract each patch with its **mean pixel value**
    - Obtain **a set of 'noise' patches**, $\mathcal{N} = \{\boldsymbol{n}^{(j)}\}_{j=1}^N$

- **Component 2: Training a generative model**
  - Overall structure and loss functions



$$\mathcal{L}_{\boldsymbol{n}}(\boldsymbol{\theta}_1, \boldsymbol{w}_1) \triangleq \mathbb{E}_{\boldsymbol{n}}[f_{\boldsymbol{w}_1}(\boldsymbol{n})] - \mathbb{E}_{\boldsymbol{r}}[f_{\boldsymbol{w}_1}(g_{\boldsymbol{\theta}_1}(\boldsymbol{r}))]$$

$$\mathcal{L}_{\mathbf{Z}}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{w}_2) \triangleq \mathbb{E}_{\mathbf{Z}}[f_{\boldsymbol{w}_2}(\mathbf{Z})] - \mathbb{E}_{\mathbf{Z},\boldsymbol{r}}[f_{\boldsymbol{w}_2}(g_{\boldsymbol{\theta}_2}(\mathbf{Z}) + g_{\boldsymbol{\theta}_1}(\boldsymbol{r}))]$$

$$\mathcal{L}_{\text{cyc}}(\boldsymbol{\theta}_2, \boldsymbol{\theta}_3) \triangleq \mathbb{E}_{\mathbf{Z}}[\|\boldsymbol{z} - g_{\boldsymbol{\theta}_3}(g_{\boldsymbol{\theta}_2}(\mathbf{Z}))\|_1]$$

  - To approximately solve,

$$\min_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3}\max_{\boldsymbol{w}_1,\boldsymbol{w}_2}\left[\alpha\mathcal{L}_{\boldsymbol{n}}(\boldsymbol{\theta}_1,\boldsymbol{w}_1) + \beta\mathcal{L}_{\mathbf{Z}}(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{w}_2) + \gamma\mathcal{L}_{\text{cyc}}(\boldsymbol{\theta}_2,\boldsymbol{\theta}_3)\right],$$

    - (α, β, γ) is the hyperparameters
  - Obtain **a noise generator** g_{θ_1} and **rough denoiser** g_{θ_2}

- **Component 3: Iterative GAN2GAN**
  - Carry out the **iterative Noisy N2N** with **generators** $g_{\boldsymbol{\theta}_1}, g_{\boldsymbol{\theta}_2}$
  - **1st iterative GAN2GAN**
    - Generate the pairs for collecting given $\mathbf{Z}^{(i)} \in \mathcal{D}$

$$(\hat{\mathbf{Z}}_{11}^{(i)}, \hat{\mathbf{Z}}_{12}^{(i)}) \triangleq (g_{\boldsymbol{\theta}_2}(\mathbf{Z}^{(i)}) + g_{\boldsymbol{\theta}_1}(\boldsymbol{r}_{11}^{(i)}), g_{\boldsymbol{\theta}_2}(\mathbf{Z}^{(i)}) + g_{\boldsymbol{\theta}_1}(\boldsymbol{r}_{12}^{(i)})),$$

    - A denoiser $\hat{X}_{\boldsymbol{\phi}}(\mathbf{Z})$ is trained by

$$\mathcal{L}_{\text{G2G}}(\boldsymbol{\phi}, \hat{\mathcal{D}}_1) \triangleq \frac{1}{n}\sum_{i=1}^{n}(\hat{\mathbf{Z}}_{11}^{(i)} - \hat{X}_{\boldsymbol{\phi}}(\hat{\mathbf{Z}}_{12}^{(i)}))^2$$

  - ***j*-th iterative GAN2GAN** (with $j \geq 2$)
    - Generate $\hat{\mathcal{D}}_j$ using $g_{\boldsymbol{\theta}_1}$ and $\hat{X}_{\boldsymbol{\phi}_{j-1}}$
    - A new denoiser G2G_j is obtained by

$$\phi_j \triangleq \arg\min_{\phi}\mathcal{L}_{\text{G2G}}(\phi, \hat{\mathcal{D}}_j)$$

Replace and perform iteratively

Warm-starting from ϕ_{j-1}

Experimental Results

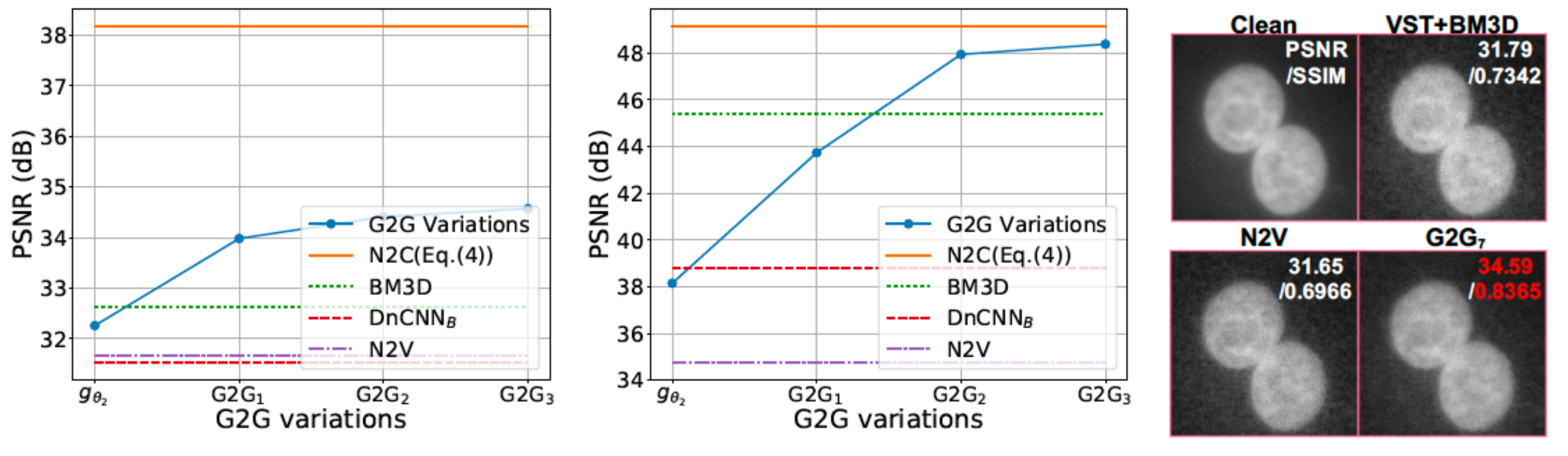
- Experimental result on **a synthetic noise**
  - Training data: BSD400, Test data: BSD68, Model: DnCNN
  - **Gaussian noise**

PSNR/SSIM	Baselines				G2G variation				Upper Bound
	BM3D	DnCNN-B	N2N	N2V	g_{θ_2}	G2G$_1$	G2G$_2$	G2G$_3$	N2C(Eq.(4))
$\sigma = 15$	**31.07/0.8717**	31.44/0.8836	31.20/0.8745	**29.48/0.8199**	**25.94/0.7519**	**30.98/0.8552**	**32.51/0.8827**	**31.45/0.8825**	31.64/0.8870
$\sigma = 25$	**28.56/0.8013**	28.92/0.8137	28.74/0.8041	**26.97/0.7083**	**24.16/0.6630**	**28.23/0.7669**	**28.82/0.8056**	**28.96/0.8080**	29.11/0.8189
$\sigma = 30$	**27.78/0.7727**	28.06/0.7812	27.91/0.7720	**26.38/0.6657**	**23.43/0.5967**	**27.58/0.7413**	**27.99/0.7783**	**28.03/0.7759**	28.28/0.7890
$\sigma = 50$	**25.60/0.6866**	25.78/0.6721	25.71/0.6712	**24.30/0.5765**	**20.58/0.4482**	**25.08/0.6215**	**25.55/0.6639**	**25.78/0.6749**	26.03/0.6951

  - **Mixture/Correlated noise**

PSNR/SSIM			Baselines				G2G variation				Upper bound
			BM3D	DnCNN-B	N2N	N2V	g_{θ_2}	G2G$_1$	G2G$_2$	G2G$_3$	N2C(Eq.(4))
Mixture noise	Case A	$s = 15$	**41.44/0.9822**	39.62/0.9749	40.59/0.9860	**33.53/0.9368**	**31.85/0.9522**	**42.35/0.9876**	**42.56/0.9888**	**42.49/0.9885**	42.92/0.9843
		$s = 25$	**37.97/0.9647**	37.23/0.9616	37.39/0.9737	**31.62/0.9057**	**32.73/0.9478**	**39.13/0.9761**	**39.64/0.9800**	**39.72/0.9807**	40.42/0.9843
	Case B	$s = 30$	**30.12/0.8549**	30.58/0.8655	30.58/0.8655	**28.10/0.7543**	**27.55/0.7728**	**29.05/0.8199**	**30.32/0.8456**	**30.49/0.8538**	30.78/0.8685
		$s = 50$	**29.27/0.8190**	30.20/0.8547	30.20/0.8547	**28.22/0.7755**	**27.36/0.7712**	**29.78/0.8345**	**30.04/0.8392**	**30.00/0.8417**	30.39/0.8574
Correlated noise		$\sigma = 15$	**29.84/0.8504**	30.84/0.9011	30.69/0.9223	**28.80/0.8367**	**28.13/0.8370**	**30.73/0.8889**	**31.09/0.8949**	**31.26/0.8954**	31.60/0.9075
		$\sigma = 25$	**26.69/0.7544**	27.39/0.8257	27.32/0.8594	**26.11/0.7348**	**25.68/0.7607**	**27.80/0.8130**	**28.01/0.8271**	**28.00/0.8447**	28.42/0.8376

- Experimental result on **a real noise**
  - The **source-independent** and **pixel-wide correlated** real noise



(a) Wide-Focal (WF) (b) Reconstructed CT (c) Visualization