Supplementary for Self-Supervised Denoising Transformer with Gaussian Process

Rajeev Yasarla Jeya Maria Jose Valanarasu Vishal M. Patel*
Johns Hopkins University
Department of Electrical and Computer Engineering, Baltimore, MD 21218, USA
{ryasarl1, jvalana1, vpatel36}@jhu.edu

1. Ablation study

1.1. λ_{GP}

We conduct different experiments for various values of k and N on BSD dataset [7]. Table 1 shows the performance of SST-GP when we vary the value of λ_{GP} used in training SST-GP.

Table 1. Ablation experiments for λ_{GP} using BSD dataset.

Noise type	Noise level	$\lambda_{GP} = 0.3$	$\lambda_{GP} = 0.03$	$\lambda_{GP} = 0.003$
Gaussian	$\sigma = 25$	31.12/0.878	31.18/0.880	31.01/0.877
	$\sigma = [5, 50]$	31.08/0.868	31.12/0.869	30.94/0.862

1.2. Kernel Function

We conduct different experiments for different kernel function (Linear kernel LIN[.], Square Exponential SE[.], and Rational Quadratic RQ[.]) on BSD dataset [7]. Table 2 shows the performance of SST-GP is best when we rational quadratic kernel function for Gaussian process in SST-GP.

Table 2. Ablation experiments for kernel function used in SST-GP using BSD dataset.

Noise type	Noise level	LIN[.]	SE[.]	RQ[.]
Gaussian	$\sigma = 25$	30.88/0.872	31.02/0.877	31.18/0.880
	$\sigma = [5, 50]$	30.91/0.863	30.99/0.865	31.12/0.869

1.3. Down-sampled images

In our SST-GP given a noisy image y we use down-sampling technique proposed in [3] with cell size $k = 2$, and obtain down-sampled images. Additionally, we randomly cyclic-shift each down-sampled image four times to obtain $\{y_1^d\, y_2^d, y_3^d, \dots, y_N^d\}$ (implies $N = 8$). An example down-sampled images are shown in Fig. 1. In Table 3, we conduct ablation study for different values of N.

*This work was supported by NSF CAREER award 2045489.

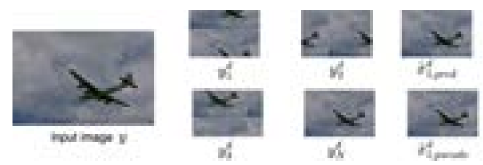

Figure 1. Examle downsampled and cyclically shifted images.

Table 3. Ablation experiments for different values of N using BSD dataset.

Noise type	Noise level	$N = 2$	$N = 4$	$N = 8$	$N = 12$
Gaussian	$\sigma = 25$	30.80/0.871	30.99/0.875	31.18/0.880	31.2/0.880
	$\sigma = [5, 50]$	30.77/0.857	30.90/0.861	31.12/0.869	31.15/0.869

1.4. Random sampling

In our SST-GP given a noisy image y we use down-sampling technique proposed in [3] with cell size $k = 2$, and obtain down-sampled images. Additionally, we randomly cyclic-shift each down-sampled image four times to obtain $\{y_1^d\, y_2^d, y_3^d, \dots, y_N^d\}$ (implies $N = 8$). In Table 4, we conduct ablation study to know hoe random shifthing in obtaing down-sampled images effects SST-GP's performance.

Table 4. Ablation experiments for random shifting using BSD dataset.

Noise type	Noise level	w/o random shifting	w/ random shifting
Gaussian	$\sigma = 25$	30.83/0.872	31.18/0.880
	$\sigma = [5, 50]$	30.75/0.857	31.12/0.869

2. Algorithm

Algorithm 1 shows the pseudo algorithm for the proposed SST-GP.

Algorithm 1 Pseudo code for training 3SD

Input: Set of noisy images $\mathcal{D} = \{y^i\}_{i=1}^{M}$
Output: $\hat{\theta}$, optimized network parameters of SST-GP. optimized GP parameters $\alpha, \beta, \sigma_\epsilon$.

1: **for** every epoch **do**
2: **for** $\{y^i\} \in \mathcal{D}$ **do**
3: Generate down-sampled images [3] and clyical shit the images to obtain $\{y_1^{d,i}\ y_2^{d,i}, y_3^{d,i}, \ldots, y_N^{d,i}\}$ from y^i
4: forward them through SST-GP to obtain $\{\hat{x}_{1,pred}^{d,i}\ \hat{x}_{2,pred}^{d,i}, \hat{x}_{3,pred}^{d,i}, \ldots, \hat{x}_{N,pred}^{d,i}\}$
5: using GP obtain corresponding pseudo-GTs $\{\hat{x}_{1,pseudo}^{d,i}\ \hat{x}_{2,pseudo}^{d,i}, \hat{x}_{3,pseudo}^{d,i}, \ldots, \hat{x}_{N,pseudo}^{d,i}\}$
6: Compute loss $\mathcal{L}_{total}$
7: update SST-GP parameters
8: compute gradients for GP parameters(α, β, and σ_ϵ) using loss $\mathcal{L}_{total}$
9: update GP parameters (α, β, and σ_ϵ)
10: **end for**
11: **end for**

3. Comparisons

3.1. Quantitative Comparisons

Table 6 comparisons of SST-GP additional SOTA methods [8, 12] on BSD dataset for Guassian, and Poisson noise images test sets.

3.2. Training and Inference time

We both networks U-Net and Den-T for 60 epochs with training set images (please refer sections 4.3 and 5.1 in the main paper). Table 5 shows the training time for U-Net and Den-T. Additionally we compare inference time of U-Net and Den-T when the input images is 256 pixels.

Table 5. Training and tesing time comparisons for U0Net and Den-T.

Method	U-Net	U-Net w/ GP	Den-T	SST-GP (Den-T w/ GP)
Training time (hrs)	65	90	50	76
Inference time(ms)	98	98	84	84

4. Real Test Comparisons

The SIDD [1] dataset is used to compare the performance of SST-GP against the other methods. We train all the networks using the SIDD Medium training dataset images, and follow the steps mentioned in the respective SOTA methods. As BM3D [2] requires prior information to denoise, we use Anscombe for Poisson to estimate the priors. Results corresponding to this experiment are shown in Table 7 and Figure 2. In contrast to other methods [3,5,6,9], we used down-sampled images and modelled joint distribution using GP, that helped the proposed SST-GP outperform the other methods by a significant margin and it is able to produce sharper images than the other methods.

5. Synthetic Test Qualitative Analysis

Figure 3 and Figure 4 illustrates sample denoising results compared with recent approaches. As it can be observed, the results of the proposed method are more clearer and sharper in contrast to the outputs of the other methods [3, 5, 6, 9].

6. Comparison with Liu *etal.*

we compared our SST-GP with Liu *etal.*[2] using Confocal Mice dataset [3], where our SST-GP method outperformed Liu *etal.*[2] (refer to Table 8) by 0.35dB.

7. Sigma values $\mathbf{\Sigma}_j^d$

Figure 5 shows $\mathbf{\Sigma}_j^d$ values at different epochs of training SST-GP. Here we can clearly observe that initial $\mathbf{\Sigma}_j^d$ values are high as the corresponding output prediction images are noisy and eventually the variance values reduce over the training process as the output predictions get clearer and sharper. This shows that minimizing the variance helps GP model to learn the joint distribution more accurately, and obtain accurate pseudo-GT labels.

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S Brown. A high-quality denoising dataset for smartphone cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1692–1700, 2018. 2, 3

[2] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007. 2, 3

[3] Tao Huang, Songjiang Li, Xu Jia, Huchuan Lu, and Jianzhuang Liu. Neighbor2neighbor: Self-supervised denoising from single noisy images. *arXiv preprint arXiv:2101.02824*, 2021. 1, 2, 3

[4] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2129–2137, 2019. 3

[5] Samuli Laine, Tero Karras, Jaakko Lehtinen, and Timo Aila. High-quality self-supervised deep image denoising. *arXiv preprint arXiv:1901.10277*, 2019. 2, 3

[6] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila.

[0] https://github.com/AbdoKamel/simple-camera-pipeline

Table 6. PSNR/SSIM comparisons on synthetic test sets created using Gaussian noise and Poison noise. Higher number represents better performance.

Type of Noise	Dataset	N2C [9]	N2N [6]	CBM3D [2]	DIP [10]	N2V [4]	Laine19-mu [5]	Laine19-pme [5]	DBSN [11]	Self2self [8]	Noise2Same [12]	Huang *et al.* [3]	SST-GP (ours)	Den-T w/ GP oracle (ours)
Gaussian $\sigma = 25$	BSD	31.05/0.879	31.04/0.878	30.48/0.861	26.38/0.708	29.34/0.824	28.62/0.803	30.99/0.877	29.80/0.839	28.70/0.807	27.95/0.782	30.79/0.873	**31.18/0.880**	31.44/0.900
Poisson $\sigma = 30$	BSD	30.36/0.868	30.35/0.868	29.18/0.842	26.07/0.698	28.46/0.798	28.25/0.794	30.25/0.866	28.19/0.790	28.16/0.791	27.41/0.764	30.10/0.863	**30.84/0.897**	31.04/0.910

Table 7. PSNR/SSIM comparisons onreal-world noise dataset SIDD [1]. PSNR/SSIM higher the better performance.

Methods	N2C [9]	N2N [6]	BM3D [2]	N2V [4]	Laine19-mu [5] (Gaussian)	Laine19-mu [5] (Poisson)	DBSN [11]	Huang *et al.* [3]	Huang *et al.* [3]	SST-GP (ours
Network used	U-Net	U-Net	–	U-Net	U-Net	U-Net	DBSN	U-Net	RRGs	Den-T w/ GP
SIDD [1] Benchmark	50.60/0.991	50.62/0.991	48.60/0.986	48.01/0.983	49.82/0.989	50.28/0.989	49.56/0.987	50.47/0.990	50.76/0.991	**50.87/0.992**
SIDD [1] Vaidation	51.19/0.991	51.21/0.991	48.92/0.986	48.55/0.984	50.44/0.990	50.89/0.990	50.13/0.988	51.06/0.991	51.39/0.991	**51.57/0.992**

Noisy Image | N2C | N2N | Huang*et al.* | Laine19-pme | SS-GMM | SST-GP(ours)

Figure 2. Comparisons on real-world noisy images from the SIDD Benchmark in RAW formats. For display purpose we use the code provided by the authors of SIDD[1]to convert images from raw format to srgb

Table 8. Ablation experiments for random shifting.

Dataset	Confocal Mice[3]	Confocal ZebraFish[3]	Two-Photon Mice[3]
SST-GP (ours)	38.28	32.70	34.11
Liu *etal.* [1]	37.97	32.26	33.83

Noise2noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*, 2018. 2, 3

[7] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423. IEEE, 2001. 1

[8] Yuhui Quan, Mingqin Chen, Tongyao Pang, and Hui Ji. Self2self with dropout: Learning self-supervised denoising from single image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 3

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2, 3

[10] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9446–9454, 2018. 3

[11] Xiaohe Wu, Ming Liu, Yue Cao, Dongwei Ren, and Wangmeng Zuo. Unpaired learning of deep image denoising. In *European Conference on Computer Vision*, pages 352–368. Springer, 2020. 3

[12] Yaochen Xie, Zhengyang Wang, and Shuiwang Ji. Noise2Same: Optimizing a self-supervised bound for image

Figure 3. Comparisons on noisy images with Gaussian noise $\sigma = 25$

denoising. In *Advances in Neural Information Processing Systems*, volume 33, pages 20320–20330, 2020. 2, 3

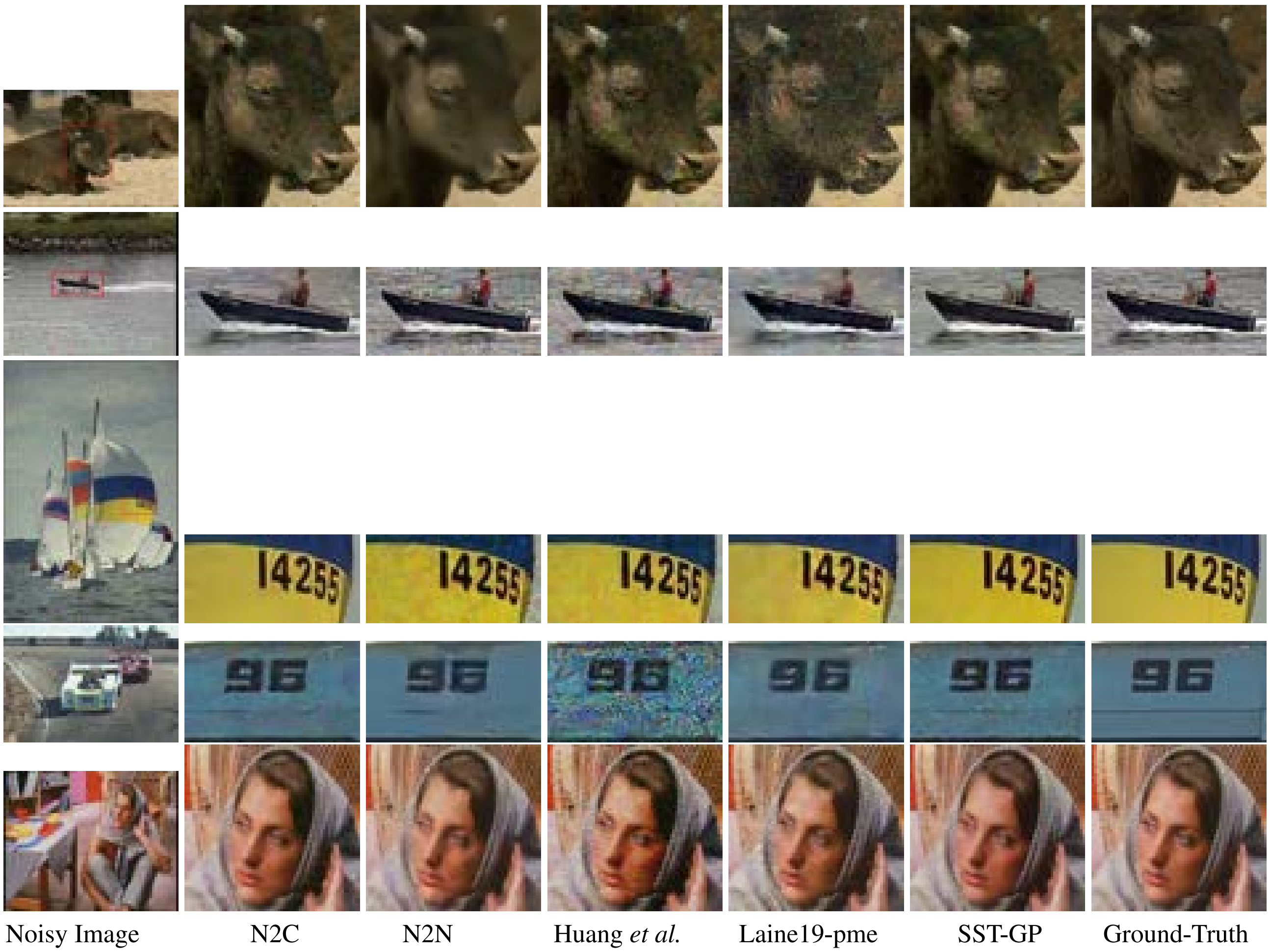

Figure 4. Comparions on noisy images with Poisson noise $\sigma = 30$

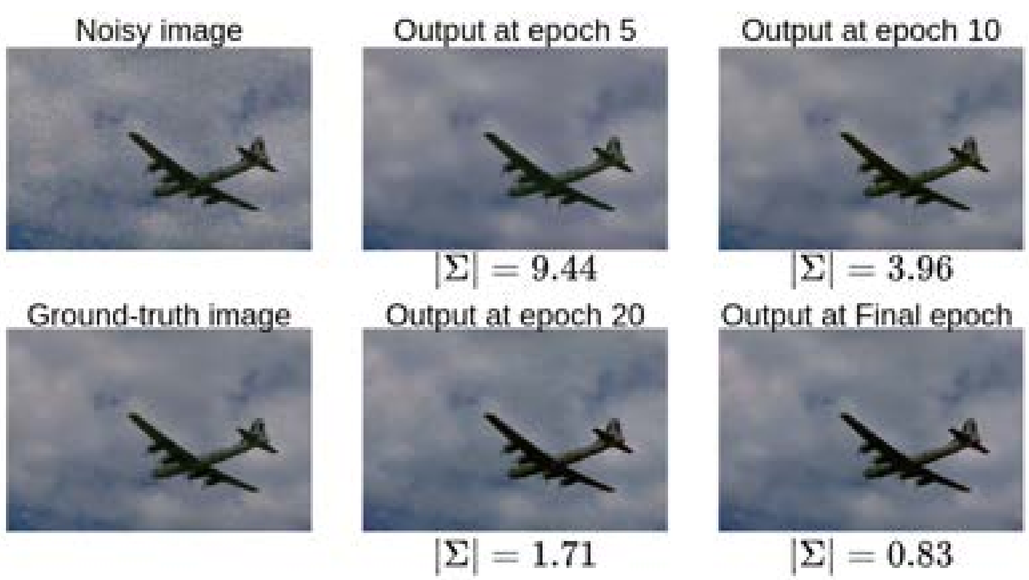

Figure 5. Denoised images on a sample down-sampled image at different epochs with corresponding variances computed using GP.