Training CNN to Denoise Images Corrupted by Mixed Poisson-Gaussian Noise Without Ground Truth Data

Ruangrawee Kitichotkul
Department of Electrical Engineering
Stanford University
Stanford, CA 94305
rk22@stanford.edu

Patin Inkaew
Department of Physics
Stanford University
Stanford, CA 94305
pinkaew@stanford.edu

Abstract

In this project, we explore the use of unbiased risk estimators as an objective function to train CNN-based denoiser for images corrupted by Poisson and mixed Poisson-Gaussian noise. Our result shows the unbiased risk estimator can be used for the dataset without ground truth. The difference between unbiased risk estimator and standard mean square error (MSE) loss during training procedure is only little. Furthermore, the model trained on unbiased risk estimator shows only small decrease in the performance on test set, compared to model trained with MSE with ground truth. Lastly, we demonstrate training model with unbiased risk estimator can be used in transfer learning to improve performance on domain-specific application, such as astrophotography.

Keywords: Image Denoising, CNN, Unbiased Risk Estimator

1 Introduction

Denoising is a standard task in computer vision. The goal of denoising is to remove noise from images according to noise models, such as Gaussian noise and Poisson noise. Because the noise is unknown, classical methods usually impose priors, such as non-local similarity and minimum total variation, to denoise images. For example, a benchmark classical method BM3D imposes non-local self similarity prior [1].

With recent development in deep learning, especially the convolutional neural networks (CNN), current state-of-the-art denoisers are CNN-based algorithms. Usually, these denoisers take noisy images as the input, and output the denoised images. CNN denoisers, such as DnCNN [2], trained on domain-specific datasets usually offer better performance than classical denoisers. CNN denoisers can also exploit information in addition to the raw pixel values to improve performance. For example, Remez et al. proposed a Poisson-Gaussian denoiser which uses the class of the image, e.g. "dog" or "face", as a label [3]. Moreover, CNN denoisers tends to offer better runtime than classical methods, because once the model is trained, the inference requires only one forward pass through the CNN.

A drawback of CNN denoisers is that large amount of data are required for training. This limitation is a problem in some domains in which acquiring clean images is difficult or impossible, such as astrophotography and microscopy. In this work, we proposed an approach to overcome this limitation by using unbiased estimators of the mean squared error (MSE) loss, i.e. risk, which do not require access to the ground truth images. We applied our approach to the Poisson noise model and the mixed Poisson-Gaussian noise model. We also showed that our approach can be used in transfer learning to improve the performance of CNN denoisers on domain-specific images with mixed Poisson Gaussian noise model — including astronomy, medicine, and biology — where low-intensity signal is captured by CCD camera [4].

2 Related work

Researchers have proposed methods to CNN-based denoising when ground truth images are not available. An approach is to estimate the MSE explicitly. We used this approach in this work. Risk estimators have been used to tune parameters of classical denoisers in Gaussian noise model [5], Poisson noise model, and mixed Poisson-Gaussian noise model [4, 6]. For the Gaussian noise model, some works have been done to train [7] and fine-tune [8] CNN denoisers.

Preprint. Under review.

There are also approaches which do not estimate MSE explicitly. For example, in the Noise2Noise approach, the models are trained with two noisy images sharing the same underlying ground truth [9]. The Noise2Void work extended this idea by using a CNN with a "blind spot" which only requires a single noisy image as a training example [10].

Finally, Deep Image Prior exploits the structure of the CNN which captures low-level details of images as a prior in image reconstruction tasks [11]. This method is very different from those previously mentioned in a sense that there is no training at all.

3 Methods

3.1 Image model and objective function

First, we introduce our image model. Li et al. discussed a image model for mixed Gaussian Poisson model [4] and Lanteri and Theys discussed the application of this model to the astrophysical images [12]. The mixed Poisson-Gaussian noise model is given by

$$\mathbf{y} = \alpha \mathcal{P}\left(\frac{\mathbf{H}\mathbf{x}}{\alpha}\right) + \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}), \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$ is a true image, $\mathbf{H} \in \mathbb{R}^{n \times n}$ is a matrix implementing the convolution with point spread function, $\alpha \in \mathbb{R}$ is Poison noise strength, and σ^2 the variance of the zero-mean additive Gaussian noise. This model can be simplified for purely Poisson noise ($\sigma = 0$) and purely Gaussian noise ($\alpha \to 0$).

Let f be a denoiser. The denoised image is then $\hat{\mathbf{x}} = f(\mathbf{y})$. Many denoising methods are applicable. In this work, We used CNN-based algorithms, including DnCNN [2] and the denoiser proposed by Remez et al. [3]. We need to define the objective function and in this case, we use the mean squared error (MSE) objective and the goal is to minimize the MSE.

$$\text{MSE} = \frac{1}{n}\mathbb{E}\{||\hat{\mathbf{x}} - \mathbf{x}||^2\} = \frac{1}{n}\mathbb{E}\{||f(\mathbf{y}) - \mathbf{x}||^2\} \tag{2}$$

However, in practice, we cannot compute MSE directly since the true image $\mathbf{x}$ is unknown. To solve this problem, Li et al. proposed a random variable solely depending on the observed image $\mathbf{y}$ whose expectation equals to MSE, thus is a unbiased estimate for MSE. For purely Poisson noise case, a random variable Poisson Unbiased Risk Estimate (PURE) can be defined as

$$\text{PURE}\{f\} = \frac{1}{n}||f(\mathbf{y})||^2 - \frac{2}{n}\mathbf{y}^T\mathbf{H}^{-T}f^-(\mathbf{y}) + \mathcal{E}_{\mathcal{P}}, \tag{3}$$

where $f^-(\mathbf{y}) = [f_i(\mathbf{y} - \alpha\mathbf{e}_i)]$, $\mathbf{e}_i$ is standard basis of $\mathbb{R}^n$ whose ith component is one and other components are all zero, and $\mathcal{E}_{\mathcal{P}} = \frac{1}{n}\left(\mathbf{y}^T\mathbf{H}^{-T}\mathbf{H}^{-1}\mathbf{y} - \alpha\mathbf{1}^T\mathbf{H}^{-T}\mathbf{y}\right)$, independent of f.
For mixed Poisson-Gaussian model defined in (1), Li et al. proposed a Stein-Poisson Unbiased Estimator (SPURE) defined as

$$\text{SPURE}\{f\} = \frac{1}{n}||f(\mathbf{y})||^2 - \frac{2}{n}\mathbf{y}^T\mathbf{H}^{-T}f^-(\mathbf{y}) + \frac{2\sigma^2}{n}\text{div}\{\mathbf{H}^{-T}f^-(\mathbf{y})\} + \mathcal{E}_{\mathcal{P}+\mathcal{G}}, \tag{4}$$

where $\mathcal{E}_{\mathcal{P}+\mathcal{G}} = \mathcal{E}_{\mathcal{P}} - \frac{\sigma^2}{n}\text{Tr}\{\mathbf{H}^{-T}\mathbf{H}^{-1}\}$, independent of the denoiser f. The proof that SPURE is an unbiased estimator of the expected MSE can be found in the appendix of [4].

Now, we can use PURE and SPURE as the objective function for training models with no ground truth, instead of MSE which requires the ground truth images.

3.2 Stochastic evaluation of objective function

For simplicity, we consider a only denoising problem, instead of deconvolution problem. In this case, we have the point spread function convolution matrix $\mathbf{H} = \mathbf{I}$. We assume that the denoiser f as a blackbox algorithm. We does not have access to internal structure of f and we can only feed noisy image $\mathbf{y}$ and receive the denoised image $f(\mathbf{y})$. This is reasonable assumption since we need versatility in the choice of the neural network structure, and the calculation of objective function should be model-free. However, treating the denoiser as a blackbox algorithm poses several technical problems when computing PURE and SPURE. As discussed in [4], evaluating the value of $f^-(\mathbf{y}) = [f_i(\mathbf{y} - \alpha\mathbf{e}_i)]$ will be computational unrealistic. We approximate $f^-(\mathbf{y}) = f(\mathbf{y}) - \alpha\partial f(\mathbf{y})$ where $\partial f(\mathbf{y}) = \left[\frac{df_i(\mathbf{y})}{dy_i}\right]$ or the vector of diagonal of the Jacobian. Then, the approximation of PURE is

$$\begin{aligned}\text{PURE}\{f\} &= \frac{1}{n}||f(\mathbf{y})||^2 - \frac{2}{n}\mathbf{y}^T(f(\mathbf{y}) - \alpha\partial f(\mathbf{y})) + \mathcal{E}_{\mathcal{P}} \\ &= \frac{1}{n}||f(\mathbf{y})||^2 - \frac{2}{n}\langle\mathbf{y}, f(\mathbf{y})\rangle + \frac{2}{n}\langle\alpha\mathbf{y}, \partial f(\mathbf{y})\rangle + \mathcal{E}_{\mathcal{P}}\end{aligned} \tag{5}$$

And the approximation of SPURE is

$$\begin{aligned}\text{SPURE}\{f\} &= \frac{1}{n}||f(\mathbf{y})||^2 - \frac{2}{n}\mathbf{y}^T(f(\mathbf{y}) - \alpha\partial f(\mathbf{y})) + \frac{2\sigma^2}{n}\text{div}\{(f(\mathbf{y}) - \alpha\partial f(\mathbf{y}))\} + \mathcal{E}_{\mathcal{P}+\mathcal{G}} \\ &= \frac{1}{n}||f(\mathbf{y})||^2 - \frac{2}{n}\langle\mathbf{y}, f(\mathbf{y})\rangle + \frac{2}{n}\left\langle\alpha\mathbf{y} + \sigma^2\mathbf{1}, \partial f(\mathbf{y})\right\rangle - \frac{2}{n}\alpha\sigma^2\left\langle\mathbf{1}, \partial^2 f(\mathbf{y})\right\rangle + \mathcal{E}_{\mathcal{P}+\mathcal{G}}\end{aligned} \tag{6}$$

However, estimation of $\partial f(\mathbf{y})$ with finite difference is also computational unrealistic. Le Montagner et al. proposed a stochastic evaluation of $\partial f(\mathbf{y})$. Le Montagner et al. shows that for vector $\mathbf{u}, \mathbf{v}$, the inner product $\langle\mathbf{u}, \partial f(\mathbf{y})\rangle$ and $\left\langle\mathbf{v}, \partial^2 f(\mathbf{y})\right\rangle$ can be estimated with empirical formulas [6]:

$$\langle\mathbf{u}, \partial f(\mathbf{y})\rangle = \frac{1}{\epsilon}\langle\boldsymbol{\delta}\odot\mathbf{u}, f(\mathbf{y}+\epsilon\boldsymbol{\delta}) - f(\mathbf{y})\rangle \tag{7}$$

$$\left\langle\mathbf{v}, \partial^2 f(\mathbf{y})\right\rangle = \frac{1}{\epsilon^2\kappa}\langle\boldsymbol{\delta}\odot\mathbf{v}, f(\mathrm{y}+\epsilon\boldsymbol{\delta}) - 2f(\mathbf{y}) + f(\mathrm{y}-\epsilon\boldsymbol{\delta})\rangle \tag{8}$$

where ϵ is small, $\boldsymbol{\delta}$ is a random perturbation with each component δ_k is independent and has expectation value 0 and variance 1, κ is the third moment of δ: $\kappa = \mathbb{E}_{\boldsymbol{\delta}}\{\delta_i^3\}$, and $\odot$ is element-wise multiplication between two vectors. Le Montagner et al. proves that the expectation value over $\boldsymbol{\delta}$ with limit $\epsilon \to 0$ of the right hand side yield the value of the left hand side.
Thus, the empirical formula for approximate PURE is given by

$$\text{PURE}\{f\} = \frac{1}{n}||f(\mathbf{y})||^2 - \frac{2}{n}\langle\mathbf{y}, f(\mathbf{y})\rangle + \frac{2}{n\epsilon_1}\langle\boldsymbol{\delta}_1\odot\alpha\mathbf{y}, f(\mathbf{y}+\epsilon_1\boldsymbol{\delta}_1) - f(\mathbf{y})\rangle + \mathcal{E}_{\mathcal{P}} \tag{9}$$

And the empirical formula for approximate SPURE is given by

$$\begin{aligned}\text{SPURE}\{f\} =& \frac{1}{n}||f(\mathbf{y})||^2 - \frac{2}{n}\langle\mathbf{y}, f(\mathbf{y})\rangle + \frac{2}{n\epsilon_1}\left\langle\boldsymbol{\delta}_1\odot(\alpha\mathbf{y}+\sigma^2\mathbf{1}), f(\mathbf{y}+\epsilon_1\boldsymbol{\delta}_1) - f(\mathbf{y})\right\rangle \\ &- \frac{2}{n\epsilon_2^2\kappa}\alpha\sigma^2\langle\boldsymbol{\delta}_2, f(\mathrm{y}+\epsilon_2\boldsymbol{\delta}_2) - 2f(\mathbf{y}) + f(\mathrm{y}-\epsilon_2\boldsymbol{\delta}_2)\rangle + \mathcal{E}_{\mathcal{P}+\mathcal{G}}\end{aligned} \tag{10}$$

In our experiment, we choose values for $\epsilon_1, \epsilon_2, \boldsymbol{\delta}_1$, and $\boldsymbol{\delta}_2$ according to optimal values in [6].

4 Experiment, Result, and Discussion

The code for experiment can be found at `https://github.com/rkitichotkul/pgblindtrain`. First, we verified whether PURE and SPURE can be used to train CNN denoisers effectively. We used the peak signal-to-noise ratio (PSNR), defined as $\text{PSNR} = 10\log_{10}\left(\frac{(\text{max pixel value})^2}{\text{MSE}}\right)$, as the evaluation metric. We chose two CNN denoisers for the experiment – DnCNN [2] and the denoiser proposed by Remez et al. [3], referred to as Remez for short in this section. We used the same number of layers for each CNN as in the respective original works – 17 convolution layers for DnCNN and 20 convolution layers for Remez. Specific details about the architectures can be found in the original works. We used the same dataset of natural images as in the DnCNN work and preprocessed the training images in the same way. The training set contains 400 images of size 180×180 from the Berkeley segmentation dataset [13]. We cropped patches of size 40×40 with stride of 10 pixels between patches and used these patches as the training data. When training, each patch is randomly flipped, vertically and horizontally, and rotated by a multiple of 90° angle in order to append the training data. The noisy training patches are generated according to the respective noise model. For the Poisson noise model, we used a constant noise strength $\alpha = 0.01$. For the mixed Poisson-Gaussian noise model, we set $\alpha = 0.01$ and $\sigma = 25/255$. We trained each model by 15 epochs. We used Adam optimizer with initial learning rate 0.001 and the exponential decay rate for the first and the second moments 0.9 and 0.999 respectively. We used 12 standard test images as the validation set. These images are a test set in the original work. After each epoch, we evaluated the PSNR. If the validation PSNR does not improve for 5 epochs, we scale down the learning rate by 10. However, no validation PSNRs during training satisfied this criteria during the 15 epochs, so there was no change in the learning rate. Our test set contains 68 images from the Berkeley segmentation dataset (BSD68) [13]. All natural

model	PSNR on Test Set with Poisson noise	
	MSE	PURE
DnCNN	30.56	28.88
Remez	29.00	29.36

model	PSNR on Test Set with mixed Poisson-Gaussian noise	
	MSE	SPURE
DnCNN	27.43	25.49
Remez	27.54	25.89

Table 1: PSNR on Test Set with Poisson noise (left) and with mixed Poisson-Gaussin noise (right)

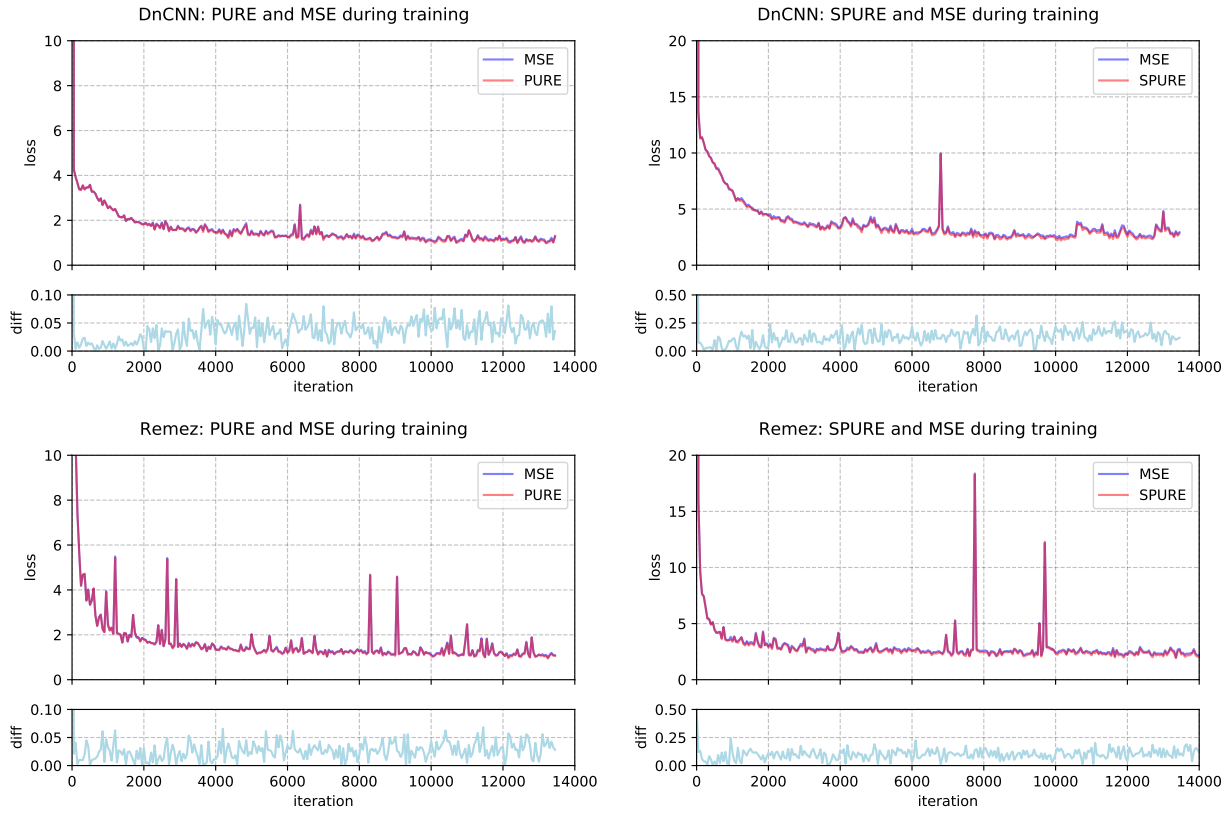

Figure 1: Training loss using PURE (left) and SPURE (right), compared to MSE. The plots in the top row belong to training of DnCNN, and the plots in the bottom row belong to training of Remez.

image training data were taken from `https://github.com/SaoYan/DnCNN-PyTorch`, a git repository containing an independent implementation of DnCNN. For each noise model, we trained the CNNs using the MSE loss and the respective risk estimator – PURE for the Poisson case and SPURE for the mixed Poisson-Gaussian case. While the terms $\mathcal{E}_{\mathcal{P}}$ in equation (9) and $\mathcal{E}_{\mathcal{P}+\mathcal{G}}$ in equation (10) do not contain the denoising function f, so they are not required for backpropagation, we included these terms during the training so that we could compare PURE and SPURE to the corresponding MSE. The training was done on the n1-highmem-2 machine with NVDIA Tesla P100 GPU on Google Cloud Platform.

For training with the risk estimators, the training loss and the difference between the estimated loss and the true MSE loss are shown in Figure 1. The estimated losses, both PURE and SPURE, track MSE closely for both DnCNN and Remez denoisers. The difference between MSE and the estimated loss appears to become stationary after 2000-4000 iterations. This difference shows the inherent error of the risk estimators. Although the model parameters had not converged yet, we have shown that PURE and SPURE can be used to train CNN denoisers. The test PSNRs of all trained models are shown in Table 1. The PSNRs of the models trained on risk estimators are less than their MSE counter parts by 1-2dB with an exception of Remez in the Poisson noise case. We believe this result is due to spikes in the training loss later in the training. With more epochs, the PSNR of the Remez denoiser trained with MSE should be higher than the PURE counterpart as well. We also experimented with modifying the number of layers of the model. We trained a DnCNN model with 20 convolution layers with the same procedure described earlier. The test PSNR is 25.55, which is a slight improvement from the DnCNN model with 17 convolution layers.

In some settings, we might have ground truth images for one dataset, such as natural image data, but not the specific dataset we want to test our model on, such as astrophotography data. An approach to overcome this limitation is to train the denoiser on the available data, and then use the estimated loss to perform transfer learning or fine-tuning on the specific dataset. This approach is particularly useful when the specific dataset is small. To simulate the training in this setting, we first train Remez on MSE with the same setting as described above but for 40 epochs. The test PSNR is 27.95. Then, we perform transfer learning on an astrophotography dataset. We used the images of planetary nebulae taken by Professor Jerome Yasavage in this experiment. The size of most of the taken photos is 500×500 pixels. We chose the patch size for the training images is 100×100 due to the sparse nature of astrophotography images. The noisy images are generated with the similar noise parameters as in experiment on natural images. Due to the limited size of astrophotography data, we used Remez model on MSE on natural image and then use SPURE to perform transfer learning on 20 images. For the validation and test, we use the similar procedure as in experiments with natural images. We benchmark the performance of model after transfer learning with the model trained on MSE before transfer as well

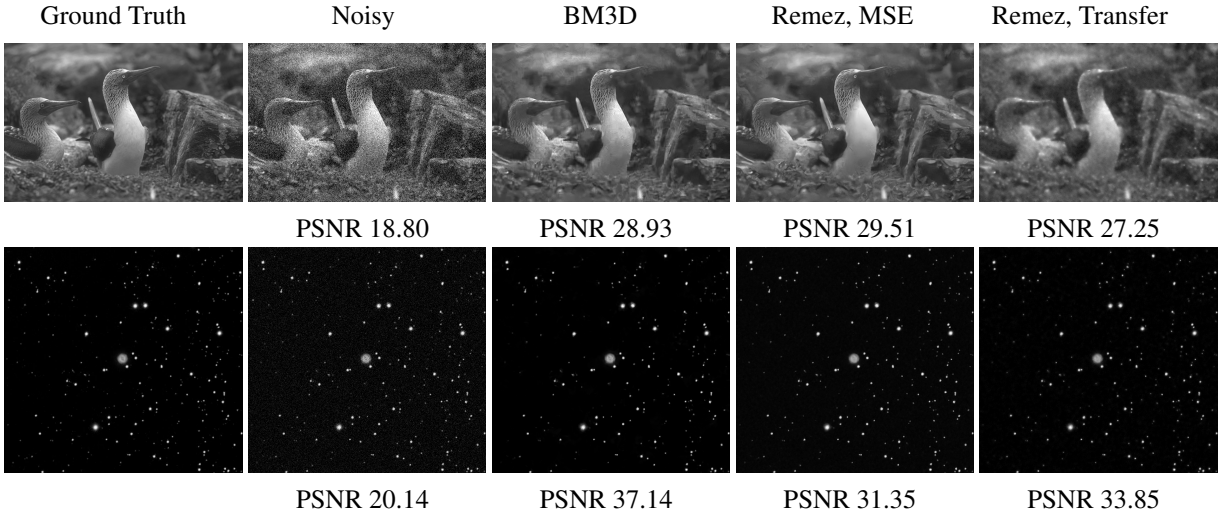

Figure 2: Sample denoising results of a test natural image and a test astrophotography image. The denoisers used are BM3D, Remez trained with MSE loss for 40 epochs, and Remez after transfer learning on the astrophotography dataset. The PSNRs are shown below each image.

as BM3D denoiser. The sample denoised images together with PSNR are shown in Figure 2. The average PSNR on astrophotography test set is 32.48 for transfer learned model and 32.55 for model without transfer learning.

From experiment on astrophotography, we observed that transfer learned model does not always perform better than the model before transfer learning. Moreover, on most astrophotography images, BM3D outperforms our transfer learning model. However, on natural images, our model always outperforms BM3D. We suspect that limited training astrophotography data contribute to the performance of transfer learned model can be worse than model before transfer in some cases. This reason also explains the better performance of BM3D since BM3D performs particularly well on sparse images, like astrophotography. Furthermore, with extensive dataset like natural images, our model consistently perform better than BM3D. From this experiment, we conclude that transfer-learned model can perform better with more training data. Our experiment demonstrates the plausibility of using SPURE on transfer learning for domain-specific imaging tasks.

5 Conclusion and Future Work

In this project, we experimented on training deep-learning-based denoiser with unbiased risk estimator, specifically PURE and SPURE, on dataset without ground truth. During the training procedure, we observed only small deviation between estimator and MSE loss. Moreover, our result shows comparable performance on test set, measured by PSNR, of model trained on MSE loss with ground truth and model trained on PURE and SPURE loss in both Poisson case and mixed Poisson-Gaussian cases. This project suggests the possible use of unbiased risk estimator in training deep-learning-based denoiser in dataset without ground truth. Furthermore, we demonstrated the use of unbiased risk estimator in transfer learning on domain-specific applications with relevant noise model - in particular, astrophography. The result shows better performance of transfer-learned model with estimator, compared to the model before transfer.

This project is only a preliminary starting point for the possible use of unbiased risk estimator with deep-learning models. Our work can be improved and further explored in many aspects. To extend this project, ones can experiment on using estimator loss to train denoiser model for images corrupted with a range of values of noise parameters, as in multi-task learning. Ones can also experiment on transfer learning with more dataset and also in other applications with mixed Poisson-Gaussian noise, such as medical and microscopy imaging. Additionally, ones can explore the use of unbiased risk estimator with other architectures, such as Image Transformer [14].

6 Acknowledgement

We would like to thank Professor Jerome Yesavage, Stanford professor in Psychiatry and Behavioral Sciences, and Neurology and Neurological Sciences (By courtesy), for allowing us to use his collection of planetary nebulae for the transfer learning demonstration. Professor Yesavage owns a astrophotography setup at his house near Stanford campus and examples of his work can be found at at `www.astrobin.com/users/jerryyyyy`.

7 Contribution

Kitichotkul implemented the pipeline for training and testing as well as the CNN models, and conducted the training on Google Cloud Platform. Inkaew implemented both PURE and SPURE objective functions and acquired the datasets. Both authors took part in writing the report.

References

[1] 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] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.

[3] Tal Remez, Or Litany, Raja Giryes, and Alex M. Bronstein. Class-aware fully convolutional gaussian and poisson denoising. *IEEE Transactions on Image Processing*, 27(11):5707–5722, Nov 2018.

[4] J. Li, F. Luisier, and T. Blu. Pure-let image deconvolution. *IEEE Transactions on Image Processing*, 27(1):92–105, 2018.

[5] T. Blu and F. Luisier. The sure-let approach to image denoising. *IEEE Transactions on Image Processing*, 16(11):2778–2786, 2007.

[6] Y. Le Montagner, E. D. Angelini, and J. Olivo-Marin. An unbiased risk estimator for image denoising in the presence of mixed poisson–gaussian noise. *IEEE Transactions on Image Processing*, 23(3):1255–1268, 2014.

[7] Shakarim Soltanayev and Se Young Chun. Training deep learning based denoisers without ground truth data. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 3257–3267. Curran Associates, Inc., 2018.

[8] Sungmin Cha and Taesup Moon. Fully convolutional pixel adaptive image denoiser, 2019.

[9] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data, 2018.

[10] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void - learning denoising from single noisy images, 2019.

[11] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. *International Journal of Computer Vision*, 128(7):1867–1888, Mar 2020.

[12] H. Lanteri and C. Theys. Restoration of astrophysical images - the case of poisson data with additive gaussian noise. *EURASIP Journal on Advances in Signal Processing*, 15, 08 2005.

[13] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.

[14] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Łukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer, 2018.