Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

computer graphics laboratory
Prof. Dr. Markus Gross

Semester Thesis

New Adventures in Noise Modeling

3rd October 2022

Supervised by:
Dr. Yang Zhang

Author:
Zhaowei Gao

Contents

Abstract

In recent years, deep neural network-based image and video denoising technologies have achieved unprecedented progress. Existing techniques rely on large-scale noisy-clean image pairs for training the denoising models. However, capturing a real camera dataset is an unacceptable expensive, and laborious procedure. This Semester Project is dedicated to investigating a novel method to generate realistic noisy images. In this report, firstly, we will compare Noise Modeling approaches, which can be divided into two categories: DNN-based and Physical-based noise modeling. Subsequently, we will introduce and evaluate our Noise Modeling approach, which combines physics-based statistical methods with GAN-based training techniques.

Chapter 1

Introduction

Image denoising has lately received great attention. It aims to estimate the original image by removing unintended noise from a noise-contaminated image. The task plays an important role in image restoration, visual tracking, image segmentation, and image classification, where obtaining the original image content is crucial for strong performance.

Recent works have achieved promising denoising performance thanks to a powerful deep learning tool.However, the denoising results in real-world scenarios are less satisfying. This phenomenon is mainly due to the discrepancy between the synthetic noise and real sensor noise distribution. The quality of the paired dataset is critical to the performance of the denoising neural network. Despite the unprecedented progress, the learning-based image denoising method made, its development remains hindered by obtaining large-scale real paired data. Collecting a large-scale real raw image dataset so far is laborious and time-consuming. Hence, to date, considerable researches have been conducted on real noise modeling.

Existing methods for synthesizing image noise usually carry out the following two steps. Firstly building a noise model and optimizing the parameters by fitting the real noise distribution, then generating synthetic noise randomly from the noise model. The different types of noise models can be divided into two categories: physics-based statistical methods and deep neural network (DNN)-based methods.

Physical statistical noise models, including Additive White Gaussian Noise (AWGN)[1, 2], Poisson-Gaussian(P-G)[3] model, Poisson Mixture model[4], etc., are commonly used in the early exploration of noise models. They provide insights into the design of more refined physics-based noise models. Recently, more sophisticated physical-based noise models[5, 6] were proposed, including a Tukey lambda(TL) distribution, etc. They follow the physical process of the specific camera sensor and model various noise step by step from photos to digital numbers. However, the physical-based method also has limitations, because first of all, most of the natural noise distribution is unknown, so it is not possible to use an accurate assuming distribution to describe it, and secondly, the real noise distributions vary dramatically in different camera settings and lighting conditions.

As an alternative, some deep neural network-based methods like GAN[7, 8, 9] and Normalizing Flow[10] have emerged, however, it remains challenging to acquire large-scale noisy/clean pairs of certain cameras. Moreover, in comparison with a more fine-grained calibrated statistical noise model, DNN-based methods have inferior performance.

In this work, inspired by Kristina et al.[11], we introduced a new perspective for synthesizing realistic image noise. We propose to use a combination of physical-based and DNN-based (GAN)methods. The noise model is learned by using a physics-inspired noise generator and easy-to-obtain paired images. Figure1.1 shows the training approach for our noise generator.

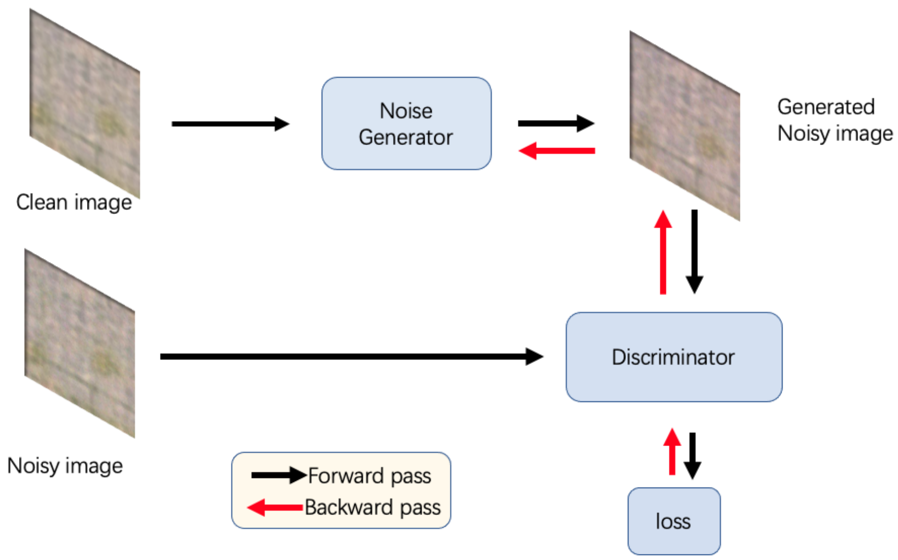

Figure 1.1: We train noise generator along with a discriminator, which aims to distinguish between real and synthetic noise. After training, the noise generator can synthesize realistic noise.

Chapter 2

Related Work

In this section, we review the related works for image noise modeling. First, we introduce widely used physics-based statistical methods, then we introduce existing deep learning-based noise modeling approaches.

2.1 Physics-based statistical noise modeling

The conventional assumption for noise synthesis is normally using Gaussian noise model[1, 12, 2], which is commonly used when we evaluate denoising methods. It assumes that the term is signal-independent while real-world noises are not, hence its performance in real-world denoising is relatively poor.

Signal-dependent models, such as Poisson-Gaussian[3] or a heteroscedastic Gaussian model[13], consider the unstable photon count on the sensor plane, which is the so-called 'shot noise'. Other statistical models including the Poisson Mixture model[4], mixed AWGN with random Value Impulse Noise(RVIN)[14], and Gaussian Mixture Model[15] are also proposed to model real noise.

However, there are many more effects that these models don't account for. Additional works have focused on delineating the full picture of sensor noise and precise statistical models.[5] They build the noise model for the low-light condition by analyzing the sensor processing pipeline and modeling the noise sources with the assumed distributions. Later, Zhang et al.[6], directly sample readout signal-independent noise from real bias patches. However, as studied in the electronic imaging community[16, 17], the camera sensor processing pipeline is complicated and it is almost impossible to accurately extract and model all noise sources since the noise can be sensor-dependent.

2.2 DNN-based noise modeling

Deep learning-based methods are also presented to implicitly model real sensor noise. Early methods use GAN[7, 8, 9] to generate noise distribution. Nevertheless, these methods show promising results for signal-independent and synthetic noise only. They oversimplify the modern sensor imaging pipeline and ignore the noise sources corrupted by sensor electronics.

More recent works exhibit greater performance improvement. Noise Flow[10] following the sensor processing pipeline adopts a flow-based generative model to generate raw image noise. PNGAN[18] establishes pixel-level adversarial training to conduct noise domain alignment. Chen et al.[19] use a ConvNet to estimate the entire ISP process, taking raw data as input and an sRGB patch as output, later CycleISP[20] models the camera ISP both in forward and reverse directions. Nevertheless, these models are unstable to train and have special needs for camera-specific data, e.g., calibration frames or clean/noisy pairs for each target camera. Collecting a multitude of paired datasets remains challenging due to several practical issues.

To alleviate the dependence on paired datasets, some works have been conducted, Cao et al.[21], propose an unpaired learning scheme, which iterates with four steps(pre-Denoising, Learning noise model, Noisy image synthesis, and Denoiser adaption), C2N[22] presents a noise generator architecture, through that, the initial noise maps are sampled in the signal-independent part and signal-dependent part of the generator respectively. Deflow[23] is also able to learn complex image degradation processes from unpaired training data.

Chapter 3

Method and Experiments

3.1 Method

In this section, we describe the specific method of our noise modeling and the dataset and settings required for the experiments. Since our approach combines physics-based noise statistical distributions and GAN-based learning, we will first introduce the required physical statistics distribution parameters, and then clarify the generator and discriminator of our GAN respectively.

3.1.1 Physics-inspired Noise Formation Model

An accurate noise model is central to noisy synthesis. Hence, we consider the electronic imaging pipeline of how incident light is converted from photons to electrons, from electrons to voltage reading, and then converted to bits by an analog to digital converter(ADC).

Digital images are corrupted during these steps of the electronic imaging pipeline. Among all noise sources, the three most significant components in real-world images are shot noise, readout noise, and quantization noise.

From Photon to Electrons: As is known, due to the quantum nature of light, the number of photos collected by sensors is unstable. As a result, inevitable shot noise is added to the original photon signal. Such uncertainty imposes a Poisson distribution over the number of electrons, which follows

$$(I + N_s) \sim P(\lambda) \tag{3.1}$$

where P is the Poisson distribution. I denotes the number of real incident photon, N_s is termed as the photon shot noise.

In our work, we combine the shot and read noise distributions together as a single heteroscedastic Gaussian random variable, details are given below.

From Electrons to Voltage: After electrons are collected, they are typically integrated, amplified, and read out as measurable charge or voltage. Readout noise is generated when the circuit reads electronic signals and transforms them into a voltage level. Considering the combination of different noise sources readout noise can be approximated as a zero-mean Gaussian random variable. It can be presented as

$$N_r \sim N(\mu, \lambda^2) \tag{3.2}$$

We note that a Poisson noise is more accurate for shot noise, but a Gaussian model is differentiable concerning its mean and variance, allowing these parameters to be learned. Thus, they are approximated together using a single heteroscedastic Gaussian random variable, where the mean is equal to the true signal x and the variance is parameterized by the readoutλ_{read} and shot noiseλ_{shot}.

$$N_s + N_r \sim N(\mu = x, \sigma^2 = \lambda_{read} + \lambda_{shot}) \tag{3.3}$$

From Voltage to Digital Numbers: During this stage, the analog voltage signal readout is quantized into discrete codes using an analog-to-digital converter(ADC). As a result, the quantization noise N_q is created.Quantization noise N_q is a rounding error between the analog input voltage to the ADC and the output digitized value, which can be assumed to follow a uniform distribution

$$N_q \sim U(\lambda_{quant}) \tag{3.4}$$

Table 3.1 summarize the necessary parameters in our noise model, which include λ_{read} and λ_{shot} for shot noise N_s and read noise N_r, λ_{quant} for quantization noise N_q. These three parameters are optimized during training to produce a realistic synthetic image from a clean image.

Thus, our physics-inspired noise model consists of the following components:

$$N = N_s + N_r + N_q \tag{3.5}$$

where N_s, N_r, N_p approximate the contributions of shot noise, read noise and quantization noise.

Table 3.1: Summary of noise model

Noise type	Formulation	Parameters
Shot noise N_s	Poisson distribution $(I + N_s) \sim P(\lambda)$	System gain K
Readout noiseN_r	Gaussian distribution $N_r \sim N(\mu, \lambda^2)$	λ_{read}
Combined noise N_r+N_s	heteroscedastic Gaussian distribution $N_s + N_r \sim N(\mu = x, \sigma^2 = \lambda_{read} + \lambda_{shot})$	λ_{read} and λ_{shot}
Quantization noise N_q	Uniform distribution $N_q \sim U(\lambda_{quant})$	λ_{quant}

3.1.2 Noise Synthesis Pipeline

Using the above physics-inspired noise model framework, our noise generator is trained to synthesize realistic noise.

Noise Synthesis Pipeline: The clean image is added by the initial noise using physics-inspired parameters, the intermediate noisy images are passed to a CNN to improve the initial noise estimation and capture any other factors not captured by the physics-inspired noise model. We use a residual 2D U-Net to achieve this goal. Then the final output of our noise generator is clipped to [0, 1].

In total, we have three physics-inspired parameters $(\lambda_{read}, \lambda_{shot}, \lambda_{quant})$, as well as the parameters of the U-Net. During the training process, all parameters are optimized to produce a realistic synthetic noise image from a noiseless image. Figure3.1 shows a sample of our physics-inspired noise generator with each noise component.

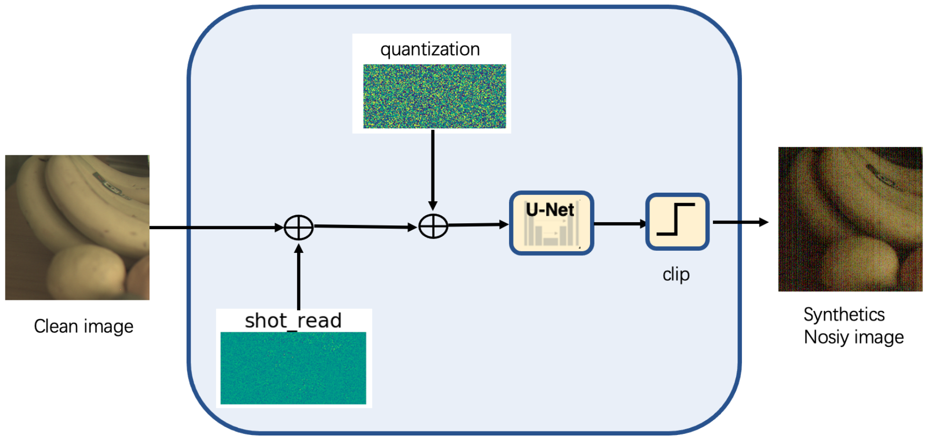

Figure 3.1: Our noise generator Pipeline contains several physics-inspired parts as well as a convolutional neural network to capture any additional effects. The final output is clipped to [0,1].

3.2 Experiments

3.2.1 Experimental Setting

Metrics. For noisy image synthesis, we use KL divergence to evaluate the distance between synthetic noise and noisy data captured by real camera sensor. We follow previous work[24] to perform discrete KL divergence between the histogram of noise patches, which can be formulated as $\sum p(x_i) \log(p(x_i)/q(x_i))$, where $p(x_i)$ and $q(x_i)$ are the normalized histogram bins of real and estimated samples.

Dataset. Our approach is trained and evaluated on a widely used real image denoising dataset SIDD[25]. SIDD is collected by five smartphone cameras, including Samsung Galaxy S6 Edge (S6), iPhone 7 (IP), Google Pixel (GP), Motorola Nexus 6 (N6) and LG G4 (G4). In addition, used as a comparison, we also synthesize noise on DUS[11].

3.2.2 GAN Training

We use an adversarial setting to train our noise generator. The generative adversarial network consists of a generator and a discriminator, where the discriminator can be used to evaluate the realism of the synthesized noisy images.

Generator: For the CNN in our noise generator, we use a standard 2D residual U-Net architecture, with 3/4 input and output channels, 4 upsampling and downsampling layers, stride- 2 convolutional downsampling layers, stride-2 transpose convolutional upsampling layers, and SeLU activations. The number of channels in our 4 downsampling and upsampling layers are 32, 64, 128, and 256. We initialize our shot, read noise parameters to 2e-1, 2e-2 respectively. We initialize our uniform noise parameter to 1e-1. In addition, we add LPIPS[26] and L2 loss to our generator loss. We re-train the generator after every 5 gradient steps on the discriminator.

Discriminator: Our discriminator is a standard Wasserstain GAN with a gradient penalty framework[27]. Its training objective function is:

$$\mathbb{E}_{\widetilde{x}\sim p_{g(x)}}[D(\widetilde{x})] - \mathbb{E}_{x\sim p_r}[D(x)] + \lambda\mathbb{E}_{\hat{x}\sim p_{\hat{x}}}[(\|\nabla_{\hat{x}}D(\hat{x})\|_2 - 1)^2]$$

where P_r is the real noisy data distribution, P_g is the model distribution defined by the generator, $\widetilde{x}$ = G(z), z is a noiseless image patch, and D is our discriminator. $\lambda\mathbb{E}_{\hat{x}\sim p_{\hat{x}}}[(\|\nabla_{\hat{x}}D(\hat{x})\|_2 - 1)^2]$ is a gradient penalty.

Our discriminator's architecture is outlined in Figure3.2

We use an Adam optimizer with a learning rate of 0.0002, with the exponential decay rate for the first and second moment estimates $\beta_1 = 0.5$ and $\beta_2 = 0.999$ for both the generator and discriminator.

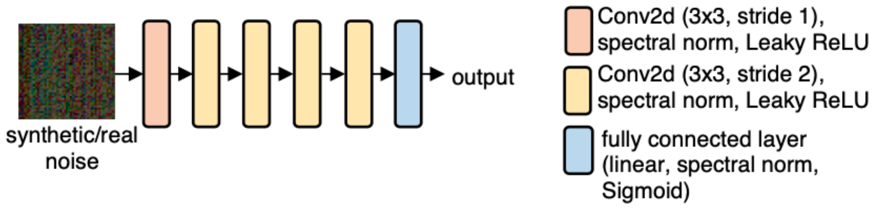

Figure 3.2: Discriminator architecture.

Chapter 4

Result and Discussion

In this section, we conduct experiments with our noise synthesis pipeline on different datasets (SIDD[25] and DUS[11]). Then, we evaluate our noise generator performance against another existing noise model DeFlow[28]. Finally, we compare the quantitative results for different datasets and evaluated them in terms of KL divergence.

4.1 Result

4.1.1 Noise Synthesis on DUS and SIDD

Figure4.1and Figure4.2 show the synthesized noisy images on DUS and SIDD datasets. Figure4.3 shows the visualization of synthetic noisy images for compared noise model(DeFlow and AWGN) and our method. The noise synthesis accuracy of all compared methods and ours are listed in Table 4.1.

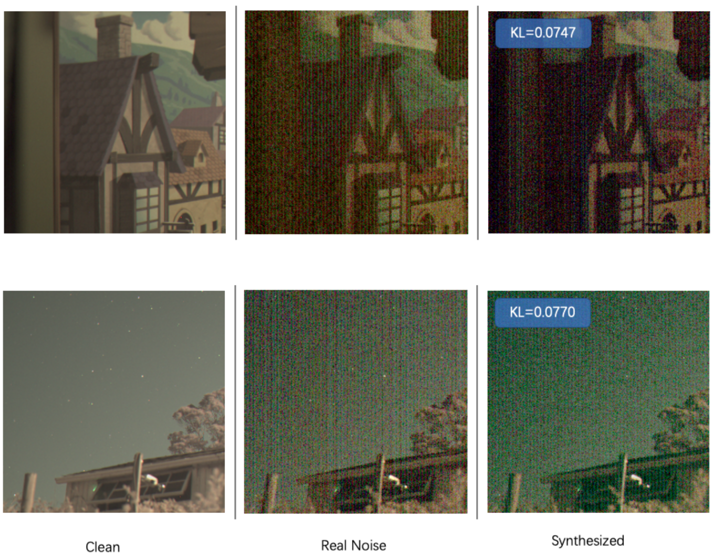

Figure 4.1: The synthesized noisy images on DUS dataset

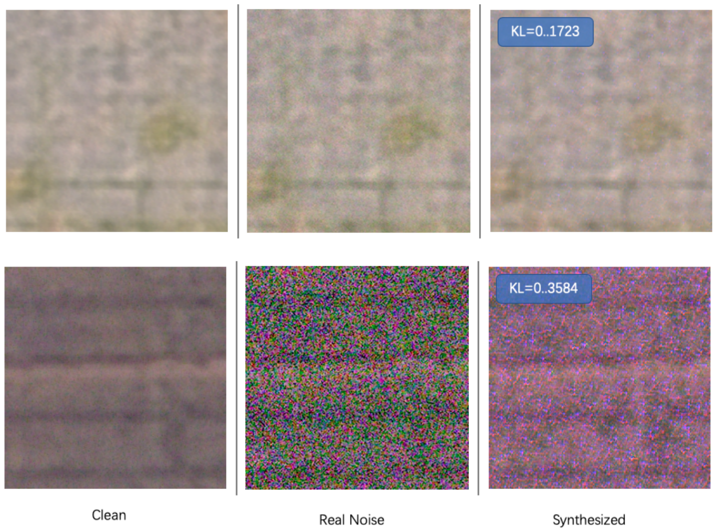

Figure 4.2: The noise synthesis results on SIDD

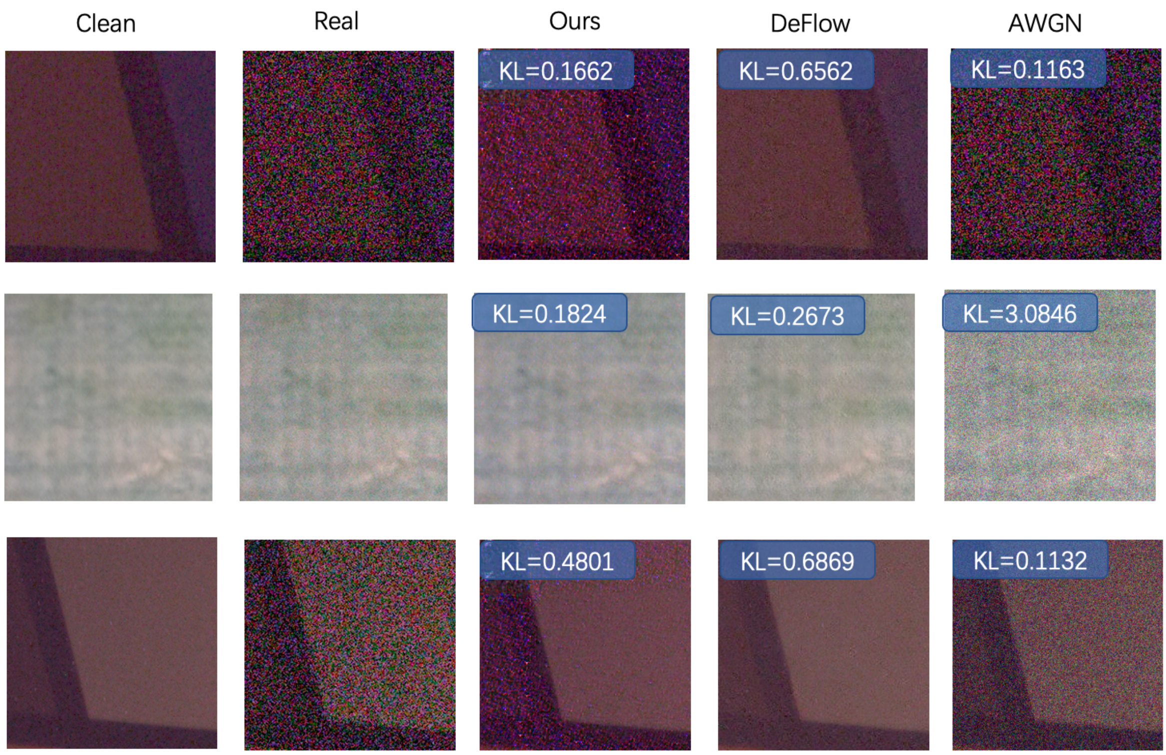

Figure 4.3: Clean input/ Real noisy image/The results of Ours/ DeFlow/ AWGN are shown from left to right.

	Ours	DeFlow	AWGN
Average KL	0.415	0.420	2.478

Table 4.1: We compare our generator to prior work Deflow and AWGN noise model. Their performance are evaluated on SIDD dataset

4.2 Discussion and future work

4.2.1 Discussion

From the Figure4.1 and Figure 4.2, we can see that our noise modeling approach outperforms on the DUS dataset than on the SIDD dataset.The synthesis results on the SIDD either produce unexpected artifacts or are not sensitive to strong real noise. It does not synthesize real strong noise very well. This is mainly due to the following two reasons, firstly, the DUS dataset contains fixed pattern noise, and all the images in the dataset are taken under same low light conditions. This suggests that our method may be more suitable for noise generation in low-light conditions, and secondly, the images in the SIDD dataset were taken by different phone types with different ISO, shutter speed and illuminant temperature. We did not differentiate according to the different settings during training, which may lead to the GANs poor performance.

4.2.2 Future Work

Due to the time constraints of the semester project, the work completed so far is not perfect. In the future, we can try to add a channel about camera settings(like ISO, shutter speed and illuminant condition) in addition to the rgb channel of sidd images. Moreover, we can try to optimize the training parameters of GAN and the physics-based noise statistical distribution parameters.

Bibliography

[1] 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*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[2] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a Fast and Flexible Solution for CNN-Based Image Denoising," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018.

[3] A. Foi, S. Alenius, V. Katkovnik, and K. Egiazarian, "Noise measurement for raw-data of digital imaging sensors by automatic segmentation of nonuniform targets," *IEEE Sensors Journal*, vol. 7, no. 10, pp. 1456–1461, 2007.

[4] J. Zhang and K. Hirakawa, "Improved denoising via poisson mixture modeling of image sensor noise," *IEEE Transactions on Image Processing*, vol. PP, pp. 1–1, 01 2017.

[5] K. Wei, Y. Fu, Y. Zheng, and J. Yang, "Physics-based Noise Modeling for Extreme Low-light Photography," *arXiv e-prints*, p. arXiv:2108.02158, Aug. 2021.

[6] Y. Zhang, H. Qin, X. Wang, and H. Li, "Rethinking Noise Synthesis and Modeling in Raw Denoising," *arXiv e-prints*, p. arXiv:2110.04756, Oct. 2021.

[7] J. Chen, J. Chen, H. Chao, and M. Yang, "Image blind denoising with generative adversarial network based noise modeling," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3155–3164.

[8] B. Henz, E. S. L. Gastal, and M. M. Oliveira, "Synthesizing camera noise using generative adversarial networks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 3, pp. 2123–2135, 2021.

[9] Z. Yue, Q. Zhao, L. Zhang, and D. Meng, "Dual Adversarial Network: Toward Real-world Noise Removal and Noise Generation," *arXiv e-prints*, p. arXiv:2007.05946, Jul. 2020.

[10] A. Abdelhamed, M. A. Brubaker, and M. S. Brown, "Noise Flow: Noise Modeling with Conditional Normalizing Flows," *arXiv e-prints*, p. arXiv:1908.08453, Aug. 2019.

[11] K. Monakhova, S. R. Richter, L. Waller, and V. Koltun, "Dancing under the stars: video denoising in starlight," *arXiv e-prints*, p. arXiv:2204.04210, Apr. 2022.

[12] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2Noise: Learning Image Restoration without Clean Data," *arXiv e-prints*, p. arXiv:1803.04189, Mar. 2018.

[13] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical poissonian-gaussian noise modeling and fitting for single-image raw-data," *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1737–1754, 2008.

[14] Y. Zhou, J. Jiao, H. Huang, Y. Wang, J. Wang, H. Shi, and T. Huang, "When AWGN-based Denoiser Meets Real Noises," *arXiv e-prints*, p. arXiv:1904.03485, Apr. 2019.

[15] F. Zhu, G. Chen, and P. A. Heng, "From noise modeling to blind image denoising," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 420–429.

[16] M. Konnik and J. Welsh, "High-level numerical simulations of noise in CCD and CMOS photosensors: review and tutorial," *arXiv e-prints*, p. arXiv:1412.4031, Dec. 2014.

[17] G. Healey and R. Kondepudy, "Radiometric ccd camera calibration and noise estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 3, pp. 267–276, 1994.

[18] Y. Cai, X. Hu, H. Wang, Y. Zhang, H. Pfister, and D. Wei, "Learning to Generate Realistic Noisy Images via Pixel-level Noise-aware Adversarial Training," *arXiv e-prints*, p. arXiv:2204.02844, Apr. 2022.

[19] C. Chen, Q. Chen, J. Xu, and V. Koltun, "Learning to See in the Dark," *arXiv e-prints*, p. arXiv:1805.01934, May 2018.

[20] S. Waqas Zamir, A. Arora, S. Khan, M. Hayat, F. Shahbaz Khan, M.-H. Yang, and L. Shao, "CycleISP: Real Image Restoration via Improved Data Synthesis," *arXiv e-prints*, p. arXiv:2003.07761, Mar. 2020.

[21] Y. Cao, X. Wu, S. Qi, X. Liu, Z. Wu, and W. Zuo, "Pseudo-ISP: Learning Pseudo In-camera Signal Processing Pipeline from A Color Image Denoiser," *arXiv e-prints*, p. arXiv:2103.10234, Mar. 2021.

[22] G. Jang, W. Lee, S. Son, and K. M. Lee, "C2n: Practical generative noise modeling for real-world denoising," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 2350–2359.

[23] V. Wolf, A. Lugmayr, M. Danelljan, L. Van Gool, and R. Timofte, "DeFlow: Learning Complex Image Degradations from Unpaired Data with Conditional Flows," *arXiv e-prints*, p. arXiv:2101.05796, Jan. 2021.

[24] Z. Yue, Q. Zhao, L. Zhang, and D. Meng, "Dual adversarial network: Toward real-world noise removal and noise generation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, August 2020.

[25] A. Abdelhamed, S. Lin, and M. S. Brown, "A high-quality denoising dataset for smartphone cameras," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1692–1700.

[26] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The Unreasonable Effectiveness of Deep Features as a Perceptual Metric," *arXiv e-prints*, p. arXiv:1801.03924, Jan. 2018.

[27] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved Training of Wasserstein GANs," *arXiv e-prints*, p. arXiv:1704.00028, Mar. 2017.

[28] V. Wolf, A. Lugmayr, M. Danelljan, L. V. Gool, and R. Timofte, "Deflow: Learning complex image degradations from unpaired data with conditional flows," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2021.

Appendix A

Acknowledgments

I am very grateful to my advisor Dr. Yang Zhang and my colleague Ming Yang Song for their help during my semester-project. They were always patient in answering my questions and giving me positive feedback. It made me feel warm and fuzzy.

They made me cheerful when I was once a bit confused and inferior, and now i gradually find the rhythm and fun of studying and researching. Not only did they guide me in the project but also brightened up my life. I would love to have the opportunity to work with them again

Appendix B

Declaration of originality

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Authored by (in block letters):
For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**

With my signature I confirm that
- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date **Signature(s)**

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.