



---

## Graduate Theses, Dissertations, and Problem Reports

---

2020

# Blind Image Denoising using Supervised and Unsupervised Learning

Surekha Pachipulusu

West Virginia University, sp0080@mix.wvu.edu

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

---

### Recommended Citation

Pachipulusu, Surekha, "Blind Image Denoising using Supervised and Unsupervised Learning" (2020).  
*Graduate Theses, Dissertations, and Problem Reports*. 7577.  
<https://researchrepository.wvu.edu/etd/7577>

This Problem/Project Report is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Problem/Project Report in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Problem/Project Report has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact [researchrepository@mail.wvu.edu](mailto:researchrepository@mail.wvu.edu).

2020

## Blind Image Denoising using Supervised and Unsupervised Learning

Surekha Pachipulusu

# **Blind Image Denoising using Supervised and Unsupervised Learning**

**Surekha Pachipulusu**

**Problem Report Submitted**

**To the Benjamin M. Statler College of Engineering and Mineral Resources**

**at West Virginia University**

**in partial fulfillment of the requirements for the degree of**

**Master of Science in  
Electrical Engineering**

**Xin Li, Ph.D., Chair  
Roy S. Nutter, Ph.D.  
Brian D. Woerner, Ph.D.**

**Lane Department of Computer Science and Electrical Engineering**

**Morgantown, West Virginia  
2020**

**Keywords: Convolutional Neural Networks, Blind image denoising, Microscopic  
Fluorescence, Tensor flow  
Copyright 2020 Surekha Pachipulusu**

## Abstract

### Blind Image Denoising using Supervised and Unsupervised Learning

Surekha Pachipulusu

Image denoising is an important problem in image processing and computer vision. In real world applications, denoising is often a pre-processing step (so-called low-level vision task) before image segmentation, object detection and recognition at higher levels. Traditional image denoising algorithms often make idealistic assumption with the noise (e.g., additive white Gaussian or Poisson). However, the noise in the real-world images such as high-ISO photos and microscopic fluorescence images are more complex. Accordingly, the performance of those traditional approaches degrades rapidly on real world data. Such blind image denoising has remained an open problem in the literature.

In this project, we report two competing approaches toward blind image denoising: *supervised* and *unsupervised* learning. We report the principles, performance, differences, merits and technical potential of few blind denoising algorithms.

Supervised learning is a regression model like a CNN with a large number of pairs of corrupted images and clean images. This feed-forward convolution neural network separates noise from the image. The reason for using CNN is its deep architecture for exploiting image characteristics, possible parallel computation with modern powerful GPU's and advances in regularization and learning methods to train. The integration of residual learning and batch normalization is effective in speeding up the training and improving the denoising performance. Here we apply basic statistical reasoning to signaling reconstruction to map corrupted observations to clean targets

Recently, few deep learning algorithms have been investigated that do not require ground truth training images. Noise2Noise is an unsupervised training method created for various applications including denoising with Gaussian, Poisson noise. In N2N model, *we observe that we can often learn to turn bad images to good images just by looking at bad images*. An experimental study is conducted on practical properties of noisy-target training at performance levels close to using the clean target data. Further, Noise2Void(N2V) is a self-supervised method that takes one step further. This is method does not require clean image data nor noisy image data for training. It is directly trained on the current image that is to be denoised where other methods cannot do it. This is useful for datasets where we cannot find either a noisy dataset or a pair of clean images for training i.e., biomedical image data.

## Acknowledgements

Foremost, I would like to express my sincere gratitude to my Committee Chair and advisor, Dr. Xin Li for his invaluable assistance and constant support throughout my time at WVU. He has been an incredible source of support and motivation. His guidance helped me in all the time work and in writing this document.

Apart from my advisor, I would like to acknowledge and thank my Problem Report committee members Dr. Roy Nutter and Dr. Brian Woerner for their time, encouragement and valuable advice.

I am also thankful to Dr. Brian Powell for providing me an opportunity to work as a Graduate Teaching Assistant for CS101. I would also like to thank the Lane Department of Computer Science and Electrical Engineering for giving me the opportunity to pursue my master's degree in such an academically vibrant environment.

Last but not the least, I owe my sincere thanks to my family and friends both back home and here, for their continued love and support whose love and sacrifice for me is beyond anything I will ever understand.

## Table of Contents

<b>Chapter 1 .....</b>	<b>1</b>
<b>Introduction.....</b>	<b>1</b>
<b>1.1 Motivation .....</b>	<b>1</b>
<b>1.2 Digital Images .....</b>	<b>2</b>
1.2.1 Noise Modelling for Real-world noisy images.....	2
1.2.2 Microscopic Fluorescence Images.....	4
<b>1.3 Development of research in image denoising .....</b>	<b>7</b>
1.3.1 Discriminative Learning based approach.....	7
1.3.2 CNN Model based approach .....	9
<b>1.4 Contribution .....</b>	<b>10</b>
<b>1.5 Overview .....</b>	<b>10</b>
<b>Chapter 2 .....</b>	<b>11</b>
<b>Supervised learning .....</b>	<b>11</b>
<b>2.1 Introduction .....</b>	<b>11</b>
<b>2.2 Technical Background.....</b>	<b>13</b>
2.2.1 Residual Learning.....	13
2.2.2 Batch Normalization .....	14
<b>2.3 DnCNN training.....</b>	<b>14</b>
<b>2.4 Network Architecture of DnCNN .....</b>	<b>15</b>
2.4.1 Network Architecture .....	15
2.4.2 Model Learning.....	17
2.4.3 DnCNN with unknown noise levels .....	18
<b>2.5 DnCNN with Microscopy Fluorescence Images .....</b>	<b>19</b>
<b>Chapter 3 .....</b>	<b>21</b>
<b>Unsupervised Learning.....</b>	<b>21</b>
<b>3.1 Introduction .....</b>	<b>21</b>
<b>3.2 Technical background .....</b>	<b>22</b>
<b>3.3 Noise2Noise Training.....</b>	<b>24</b>
<b>3.4 Deep Architecture .....</b>	<b>25</b>
3.4.1 RED30 .....	25
3.4.2 U-Network .....	25
3.4.3 MRI Images.....	28
<b>3.5 Noise2Noise on Microscopic Fluorescence images.....</b>	<b>28</b>

<b>Chapter 4 .....</b>	<b>29</b>
<b><i>Self-Supervised learning</i>.....</b>	<b>29</b>
<b>4.1 Introduction .....</b>	<b>29</b>
<b>4.2 Technical Background.....</b>	<b>30</b>
4.2.1 Internal statistics method in Noise2Void .....	32
4.2.2 Image formation model.....	33
<b>4.3 Noise2Void Training .....</b>	<b>33</b>
<b>4.4 Network Architecture .....</b>	<b>35</b>
<b>4.5 Noise2Void on Microscopic Fluorescence Images .....</b>	<b>37</b>
<b>Chapter 5 .....</b>	<b>38</b>
<b><i>Comparison of Results and Summary</i> .....</b>	<b>38</b>
<b>Chapter 6 .....</b>	<b>46</b>
<b><i>Future Work</i> .....</b>	<b>46</b>

## List of Figures

Figure 1: Timeline with a selection of representative denoising approaches .....	8
Figure 2: Categories of Supervised Learning.....	11
Figure 3: Proposed architecture of DnCNN.....	16
Figure 4: The Gaussian denoising results of four specific models under two gradient-based optimization algorithms. ....	18
Figure 5: Conventional neural network along with the blind-spot network. ....	34
Figure 6: Blind-spot masking scheme used in Noise2Void Training.. ....	34
Figure 7: U-net architecture with 32x32 as the lowest resolution.. ....	36
Figure 8: DnCNN at Noise Level = 25.....	38
Figure 9: DnCNN on CBSD68. ....	39
Figure 10: DnCNN on noisy image corrupted by AWGN. ....	39
Figure 11: Noise2Noise on BSD Dataset.....	40
Figure 12: Noise2Noise on IXI-T1 .....	40
Figure 13: Noise2Noise on BSD Dataset.....	41
Figure 14: DnCNN on Microscopic Fluorescence Images .....	42
Figure 15: PSNR and MSE on the mixed test set with raw images during training.....	42
Figure 16: Noise2Noise on Microscopic Fluorescence images.....	43
Figure 17: PSNR and MSE on the mixed test set with raw images during training.....	43
Figure 18: Noise2Void on Microscopic Fluorescence images.....	43
Figure 19: Results and average PSNR values obtained by DnCNN, Noise2Noise and Noise2Void.....	44
Figure 20: Illustration for CBDNet for blind denoising real-world images.....	47
Figure 21: Denoising results of a real-world noisy image using CBDNet .....	48
Figure 22: Denoising results of microscopic fluorescence images using CBDNet .....	48



# Chapter 1

## Introduction

Image processing has a number of applications including image segmentation, object detection, image classification, video tracking and image restoration. Especially, denoising is an important branch in image processing and can be used as an example of growth in image processing tools in recent years. In real world applications, denoising is often a pre-processing step (so-called low-level vision task) before image segmentation, object detection and recognition at higher levels. A passive approach to improve image quality is one that lags behind improvements in imaging hardware, awaiting better sensor technology of acquisition devices. Recent improvement in hardware and imaging systems made the digital cameras appear everywhere. Though the hardware development improved the quality of image, image degradation is unavoidable due to several factors. Those factors exist during image acquisition process and in its post processing. The process of image denoising to reconstruct the high-quality image from noisy observation is still an active topic in the area of computer vision. It still holds its importance in the real time applications like medical image analysis, digital photography, High ISO images, MRI, remote sensing, surveillance and digital entertainment field.

In this report, we consider a typical blind image denoising problem, which is to remove unknown noise from noisy images. Our extensive experiments demonstrate that irrespective of noisy data training, there are methods that not only exhibit high effectiveness in image denoising tasks but also benefited by efficiently implementing GPU Computing.

### 1.1 Motivation

Researchers expose that deep learning technologies have obtained enormous success in the field of image denoising. Several deep learning algorithms refer properties of image denoising to propose wise solution methods that are embedded with multiple hidden layers and connection to deal better. Recently, discriminative learning-based algorithms have received attention and are studied to a large extent because of their high denoising performance.

Most of the previous image denoising algorithms focus on additive white gaussian noise (AWGN). However, the real-world image denoising problem with advancing of the computer vision techniques. In this report, we consider a blind image denoising problem which removes unknown noise from a given real-world noisy images with a single model and produce visually pleasant images. Most of the blind denoising algorithms have shown the limited quality of restored images because they lose the fine image details due to over-smoothing which results in visually unpleasant images. Several deep learning algorithms are implemented to overcome this problem to restore noise free images by improving the visual quality. We review the principles, performance, differences, merits, shortcomings and technical potential of few blind denoising algorithms (DnCNN, Noise2Noise, Noise2Void). The potential challenges and directions of deep learning are also discussed in this report.

## 1.2 Digital Images

### 1.2.1 Noise Modelling for Real-world noisy images

Image analysis is defined as inspecting images for the purpose of recognizing objects and judging their importance. Various mathematical procedures are applied to this data in a technique called Image processing. Through this technique we can enhance image which in turn can be used to perform some of the analysis and detection tasks. Image processing is uses computers to execute image processing algorithms on digital images to fulfil tasks like Image enhancement, acquisition and pre-processing. As there is an increase in the availability of fast computers and signal processors, digital image processing is commonly used.

With increase in massive production of digital images and videos often taken in poor conditions with noise, the need for image restoration methods has grown extensively. Regardless of how great the cameras are, improvement in image quality is always desirable to broaden the range of action.

An image is often corrupted during storage, image acquisition and transmission. Image denoising usually removes the noise and try to retain the original image as much as possible. For the most part, images datasets have images contaminated with noise. Flowed instruments,

problems in the process of data acquisition and interference of natural phenomenon causes corruption in data. Consequently, noise removal has a significant role in image analysis and the initial step to be taken before image analysis. Therefore, image denoising algorithms play a necessary role to prevent noise from digital images.

Noise modelling in images are usually caused due to the instruments for capturing data, data transmission, image quantization and sources of radiation. There are different algorithms available depending on the noise model and also algorithms that handle different noise models at the same time. In general, natural images are expected to have additive random gaussian noise. Speckle noise is expected in ultrasound images whereas MRI images are affected by Rician noise. The statistical property of real-world noise has been studied for CCD and CMOS image sensors. There are five sources for real-world noise like photon shot noise, fixed pattern noise, dark current, readout noise, and quantization noise.

1. Shot noise is the one inevitable noise due to stochastic arrival process of photons. This arrival of photons is modelled as the process of number of photons arriving at the sensor following a Poisson distribution. This noise is proportional to the mean intensity of the pixel and is not constant across the whole image.
2. Fixed pattern noise includes dark current nonuniformity noise and pixel response non-uniformity. The pixel response non-uniformity noise has slightly different output level or different response for a fixed light level. Main reason for this kind of noise is the loss of light and color mixture in the surrounding pixels.
3. The dark current noise is from the electronics with sensor chip and is due to thermal agitation when there is no light reaching the camera sensor.
4. Read out noise is from the discretization of measured signals. This is generated in the process of charge-voltage conversion which is not accurate.
5. Quantization noise is when the readout values are quantized to integers. The final pixel values are a discrete value of the original raw pixel values.

Based on the above reasons, image denoising is the first step taken in data analysis. Several denoising techniques are created to compensate the data corruption. Few spatial filters like mean and median filters are used in the process of noise removal from noisy images. In the process of removing noise in digital images and smoothing the data, these spatial filters adds blur edges in image. This is the drawback of spatial filters. To overcome this drawback and to preserve

edges of the images, Wavelet Transforms are used. Wavelet transform is a powerful tool in image processing for its multi-resolution possibilities. Due to its properties like sparsity and multi-resolution structure and multi-scale nature, Wavelets have a superior performance in image denoising. With gaining popularity in algorithms for denoising in wavelet domain were introduced. Focus was shifted to Wavelet transform from Spatial and Fourier transform.

The goal of Image Denoising is inspecting a noisy image  $x = s + n$  and separate them into two components - signal image  $s$  and signal degrading noise  $n$ . Its purpose is to remove unwanted noise while preserving the important features as much as possible. A typical assumption in image denoising is that the pixel values in signal  $s$  are not statistically independent. By observing the image context of an unobserved pixel will allow us to make sensible predictions on the intensity of the pixel.

## 1.2.2 Microscopic Fluorescence Images

The ability of fluorescence microscopy to identify and distinguish cells as cellular particles made it as an essential tool in the field of biomedical science. This is due to the development of synthetic protein called fluorophores. These fluorophores are used to target specific cellular objects and are characterized by individual fluorescent profile like color, emission and excitation wavelength.

The process of creating a fluorescence microscopy images are followed by these fluorophores added to the biological samples. These fluorophores tags to each specific cellular object. The samples are then photographed using conventional light microscopy when the fluorophores present in the sample start emitting fluorescence. The emitted light is captured by the detector when filtering it out from the sample excitation light. With this excitation a high contrast fluorescent image is generated against a black background to highlight the specific object visible. This recorded samples in the form of photographs are generated at specific intervals and specific duration as required. These photographs are used to interpret about the cellular object captured.

The images captured are unable to study the cellular object due to the presence of noise in it. The noise in these images are due to two reasons (i) All photons excited by the fluorophores are not captured by the detectors. (ii) The measurement noise is observed due to imperfections in the imaging system. Also due to photo-toxicity and photo-bleaching, the excitation time has to

be limited which results in limited photon emission. All these results in a downgraded low signal-to-noise-ratio image. The noise resulted due to loss of photons is termed as Poisson noise while the measurement of noise is modelled as a Gaussian process. Therefore, noise in fluorescence microscopy images are due to mixture of Poisson and Gaussian statistics. These photos are weak when compared to our photography ( $\sim 10^5$  per pixel). Due to this, the optical signal in fluorescence microscopy is approximated to restrict a set of discrete photos and are dominated by Poisson instead of Gaussian that is dominated in photography.

One can achieve a clean image by increasing the excitation power of laser or lamp. It is not limited by the part of light an object can receive but limited by the rate of saturation in fluorescence and this fluorescence will stop once it reaches high excitation power.

And also, one can achieve the clean image by increasing the exposure time, imaging time, or number of frames but this may cause photodamage. Increasing imaging time may not be possible due to time taken to capture the image (tens of milliseconds).

All these above scenarios make it difficult to improve the fluorescence images i.e., converting it to clean image and is very important in the field of biomedical research. A proper dataset is also required in order to prove that an effective algorithm can remove this noise in images. Most of the existing datasets that are created using gaussian noise domination with a real noisy image from smart phones and digital cameras. But our algorithms have to be tested on the dataset with Poisson noise of microscopic fluorescence images as well. For testing the below algorithms, we are using a fluorescence image dataset with Poisson noise. Name of the dataset is Fluorescence Microscopic Dataset (FMD) for testing our blind denoising algorithms. This dataset consists of 12,000 real noisy microscopic images which covers confocal, two-photon, wide-field that represents biological samples including cells, zebrafish, mouse brain tissues. This image set is prepared with five different noise levels and ground truth. This FMD dataset is the first dataset designed from real noisy fluorescence microscopy images and designed for Poisson-Gaussian denoising purposes.

The FMD dataset that we used to test our blind image denoising algorithm are acquired by keeping the power of excitation lamp as low as possible for all imaging modalities like confocal, two-photon and wide-field. This excitation power of lamp is enough to produce a noisy image but enough to save the image features. To avoid pixel clipping, we manually set the camera gain to proper value. Although it is inevitable because of distinct biological structures with various optical properties to generate bright fluorescence images. To increase the difficulty of denoising task, the

images are taken in low excitation power with high noise level. The images with high noise level will allow us to generate images with low noise level by averaging them.

## High ISO Images

Capturing the images under High ISO (International Standard Organization) which is a standard for sensitive rating for camera sensors enables capturing fast moving objects, record in dark scenes and avoid blur artifacts. Several cameras were developed in order to capture high ISO images. However, at high ISO conditions, shot noise and read noise will increase. The noise in this kind of images are due to demosaicing, white balance, JPEG compression etc., which is quite different from synthetic noise like Gaussian or Poisson. Using a regular denoising method which removes the synthetic noise cannot handle this high ISO images.

## MRI

Capturing an MRI (Magnetic Resonance Image) can often be diminished in regions and tissues that has low signal to noise ratio. This is especially in the case of quantitative MRI where the quantitative parameters are suffered from low PSNR values. Denoising an MRI can be quite different that removing noise from the traditional images due to inefficiency of applying denoising methods to MRI examinations that can be typically 20 to 200 slices to be denoised. Quantitative MRI often times uses images with various contrasts and signals. This can be overcome by down sampling the high-resolution MRI data and discard the high-resolution information by low pass filtering. In this report we mention methods to address some of the concerns of using denoising for MRI in clinical research.

## Signal and Noise ratios

The quality of an image can be defined by the 256 gray-level values where 255 is defined by white and 0 is defined by black. Signal Noise ratio is a measure of noise by its standard deviation  $\sigma(n)$

$$SNR = \frac{\sigma(s)}{\sigma(n)}$$

where  $\sigma(s)$  is the empirical standard deviation of the signal image  $s$ .

$$\sigma(u) = \left( \frac{1}{|I|} \sum_{i \in I} (u(i) - \bar{u})^2 \right)^{\frac{1}{2}},$$

where  $\bar{u}$  is the average grey-level value. The standard deviation can be computed by the noise model and parameters known or by using an empirical formula.

Peak-signal-to-noise ratio (PSNR) is the ratio between the maximum possible power of signal and the maximum power of corrupting noise that impacts the constancy of illustration. It is usually expressed in terms of logarithmic decibels. PSNR is used to measure the quality of reconstruction of lossy compression formats like image denoising and image compression. The signal  $s$ , in this case is the original image and  $n$  is the noise added to the image. This defines the human perception of the resultant denoised image quality.

General denoising algorithms do not find a difference between minute details and noisy signals and remove them. Few denoising artifacts are created like blur, staircase effect, wavelet outliers etc., Denoising algorithms are generally based on noise models and the image model. Basic assumption for all the models is that noise is oscillatory whereas the image is smooth. But the weak point in algorithms is that image model is as oscillatory as the noise model.

## 1.3 Development of research in image denoising

### 1.3.1 Discriminative Learning based approach

Image denoising is the fundamental problem in the field of image processing. Due to its properties like sparsity and multi-resolution structure and multi-scale nature, Wavelet transform has become an attractive tool for image denoising. There were several algorithms designed based on the Wavelet Domain in the past two decades. The focus was shifted from Spatial and Fourier domain to Wavelet transform. To reduce the image artifacts Multiwavelets were introduced to get similar results. Probabilistic models, Bayesian denoising in wavelet domain, Independent Component Analysis (ICA) have been explored for sparse reduction. Then there is an increase in high-resolution cameras, electron microscopes, and DNA sequencers are capable of producing several feature dimensions.

But by pushing the limits of these devices to take videos of ultra-fast frame rates at low illuminations, sequencing tens of thousands of cells simultaneously, each individual feature can generate noise. Image denoising has taken a leap forward due to machine learning. However, they are mostly tested on synthetic noises rather than real-life images. One common assumption is that noise is always an Additive White Gaussian Noise (AWGN) with standard deviation. This is particularly true for learning based methods which require training data to improve the performance. Image prior algorithms play an important role in image processing when the likelihood is known.

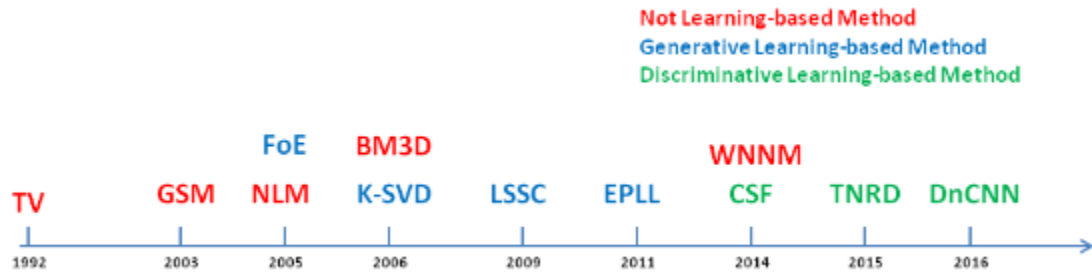


Figure 1: Timeline with a selection of representative denoising approaches

Several algorithms have been developed in the past few decades such as sparse models, gradient models, Markov models and especially nonlocal self-similarity models like BM3D, LSSC, NCSR, WNNM. Most of the image prior methods typically suffers from a few drawbacks. Firstly, they involve a complex optimization problem in testing stage which makes time consuming. Thus, they need to sacrifice computational efficiency in order to achieve high performance. Next, these methods involve several manual parameters to provide boosting for denoising tasks. To overcome these limitations of prior based models, discriminative learning methods are developed. Schmidt and Roth proposed Cascade Shrinkage Fields (CSF) that combines random field-based models and optimization algorithm into single learning framework. Chen et al. proposed Trainable Nonlinear Reaction Diffusion (TNRD) by unfolding a fixed number of gradient descent inference steps. Though these methods brought down the difference between computational complexity and denoising quality with a promising result, they are limited in capturing the full characteristics of image structures. Along with that several handcrafted parameters have to be designed and parameters are learned by stage wise greedy algorithms. Mostly, they are all trained for a specific model and specific noise level.



Later Convolution neural networks (CNN) came into the picture. It became possible to learn the structure, denoise the measurements and recover the signal without any prior training of the signal or the noise.

### 1.3.2 CNN Model based approach

In the last several years, deep neural networks achieved great success on various computer vision tasks. These networks also suggested to solve image denoising problem. Researchers showed that deep learning can perform better to automatically learn and find the features more efficiently than the manual settings. This way of estimating the parameters is different from a traditional approach mentioned above. Even the increase in availability of GPU and Big data are also essential for the development of deep neural networks. These deep learning models includes many layers such as convolutional layer, pooling layer, batch normalization and fully connected layers. Convolution neural network are said to be the most successful deep learning network for image processing. There are several attempts to handle Image denoising problem using deep neural networks. [1]Jain et al. and Seung et al. proposed an approach to use convolution neural networks as an image processing architecture to deal with image denoising problem. Their network has only 4 hidden layers and each layer uses 24 feature maps. They applied this approach on a set of natural images and found that these CNN's provide comparable results and, in some cases, superior results than state-of-the-art wavelet and Markov random Field (MRF) methods. It is observed that CNN's avoid the computational difficulties in MRF approaches and makes it possible to learn image processing architectures that have a high degree of representational power. [2]Burger et al. and Schuler et al. implemented Multi-Layer Perceptron (MLP) successfully for image denoising. In [3] stacked sparse autoencoder was adopted for handling Gaussian Noise Removal compared to [4]. In [5], a feed forward neural network Trainable Nonlinear Reaction Diffusion (TNRD) method was proposed by unfolding a fixed number of gradient descent inference steps. Based on the neural networks developed only TNRD and MLF achieved a comparable result with state-of-art method like BM3D. These specific models are trained for certain noise levels.

Driven by the advances of using deep neural networks and working on large datasets, convolution neural networks show great success in handling image denoising tasks. The achievements of training a neural network include gradient based optimization algorithms [6] - [7] [8], usage of Rectified Linear Unit (ReLU) [9], batch normalization [10] and residual learning [11].

## 1.4 Contribution

For the scope of this study, few blind image denoising algorithms are chosen to study and evaluate based on the training speed and performance

- We found that residual learning and batch normalization can benefit the CNN learning like increasing the training speed and boosting up the denoising performance.
- By using a single model to train for different noise levels, general image denoising tasks like Gaussian denoising, JPEG deblocking, denoising microscopic fluorescence can be achieved.
- Based on the available training dataset and the type of image denoising task, different algorithms are to be performed to achieve better performance.
- High resolution performance of deep neural networks can be achieved entirely without clean data based on the general-purpose deep convolution model. Noise2Noise algorithm removes the need for strenuous collection of clean data.
- Comparison of results from different denoising with existing CNN training schemes and non-trained methods.

## 1.5 Overview

The remainder of this report is organized as follows. Section 2 overviews introduction of image denoising. Section 3 provides a supervised deep learning algorithm DnCNN. Section 4 points out an Unsupervised algorithm Noise2Noise. Section 4 presents a self-supervised algorithm Noise2Void. Section 5 shows the comparative experimental results along with the summary and Section 6 is about future work in image denoising.

# Chapter 2

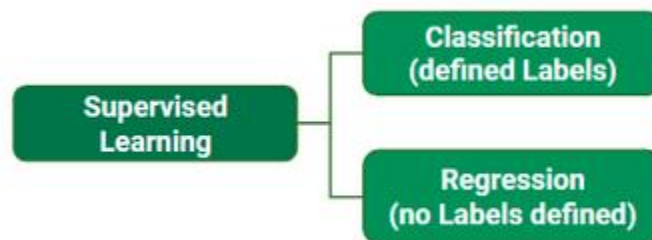
## Supervised learning

### 2.1 Introduction

Supervised learning is training a model on a labelled dataset. Labelled dataset is something which have both input and output parameters. Supervised learning always has both training and validation dataset as labelled data. Supervised learning, which requires labeled training data, provides promising results in different computer vision tasks. However, it may be expensive, time consuming, or impossible to collect and label enough training data for certain specific tasks.

While training any model, the dataset is divided into parts of training and testing data. The model is built from the training data. By learning it means that a model is created based on a logic of its own. Once a model is generated, then testing can be done by feeding the new data that the model has never seen before. Supervised Learning uses classification algorithm and regression algorithm to develop predictive models. Based on the given model, a prediction is generated and is then compared to the actual ground truth value to calculate the accuracy.

There are two categories in Supervised Learning: Classification and Regression.



*Figure 2: Categories of Supervised Learning*

## Types of Supervised Learning

Classification tasks generally predicts discrete responses. If the data has to be categorized, separated, tagged into specific groups or classes, then classification algorithm is used. These classification models classify input data into different categories. Few applications of classification include credit scoring apps, medical imaging and speech recognition. Also application that needs classification of letters and numbers or whether a tumor is benign or cancer causing one. These techniques mainly focus on predicting a qualitative response by analyzing data and recognizing patterns.

Regression task is to predict continuous responses. This algorithm attempts to model a relationship between variables by generating a linear equation. Based on regression analysis, one can start making predictions. This task is typically used in predicting, forecasting and finding relationships. This technique is still widely used for learning. This technique is used when the output is real or computed values.

For this report, we are considering one of the state-of-art methods in Supervised learning - DnCNN (Denoising Convolutional Neural networks)

DnCNN is a feed-forward neural network to embrace the progress in very deep architecture, learning algorithm, regularization method into image denoising. Instead of learning a discriminative learning algorithm with image prior, we consider a simple discriminative learning network that separates noise from noisy image using feed forward neural network. There are three main reasons for using CNN

CNN's very deep architecture is effective in increasing flexibility and capacity of image characteristics.

Advanced learning and regularization methods for training CNN, usage of Rectifier Linear Unit (ReLU), Batch Normalization and Residual Learning.

Parallel computation using powerful GPU.

These methods are adopted to fasten the speeding process, improve denoising performance and run time performance.

The proposed denoising network is a Supervised method which requires training with pairs of corrupted images and clean images. Along with outputting just the denoised images, the proposed method is designed to predict the difference between the latent clean image and noisy image i.e., the residual image. It removes the clean image from the noise image with the operations in hidden layers. Batch normalization and residual learning is benefited from each other and helps in stabilizing the training performance along with speeding up the training process and boosting the overall performance.

Though the aim of this network is to be a more effective Gaussian Denoiser, this image degradation model can be converted to Single Image Super-Resolution (SISR) problem by taking  $v$  as the difference between the original image and compressed image. Along with this JPEG image deblocking problem can also be handled by the same image degradation model.

But the basic difference between these models is the noise  $v$  is much different than AWGN. With analyzing the connection between TNRD and DnCNN, this network is extended to handle the general image denoising tasks like Gaussian denoising, SISR and JPEG deblocking.

## 2.2 Technical Background

DnCNN method mainly focuses on design and learning of CNN for image denoising. The achievements in training a CNN includes Residual Learning, batch normalization and usage of modern powerful GPUs for efficient training implementation. Below is the discussion about two methods that are related to DnCNN.

### 2.2.1 Residual Learning

Deep learning is mostly based on the basic idea of stacking multiple layers together. With the growing availability of high-performance GPUs and training data, there is an increase in depth of network layers the training accuracy begins to decrease. To solve this performance degradation problem residual networks [11] are introduced. These residual networks explicitly learn residual mapping for few stacked layers with the assumption that residual mapping is easy to learn when

compared to unreferenced mapping. By utilizing such residual learning strategy, training is made easy and increase in accuracy has been achieved for image classification.

DnCNN proposes a model that employs a single residual unit [11] in order to predict the image. This strategy of implementing the residual image is adopted in few low-level vision problems such as color demosaicing [12] and SISR [13]. For more info on residual learning, please refer to [11].

### 2.2.2 Batch Normalization

Batch Normalization is to alleviate the internal covariate shift using a normalization task and a scale and shift before the non-linearity in each step. It is like doing a preprocessing at every layer of the network. This allows each layer of network to learn more independently than other layers. For batch normalization only two parameters are added such as standard deviation and mean. Only these two weights are changed for each activation instead of changing all the weights and losing the stability. Using this normalization in our network enjoys fast training, fast and stable performance and low sensitivity to initialization.

Residual learning and batch normalization together improve the denoising performance and fast training.

## 2.3 DnCNN training

Let us consider a convolutional neural network with one image as input and other as the target. Each pixel in the output is influenced by certain set of pixels that influences the pixel prediction called receptive field  $x_{RF(i)}$  of input pixel. Receptive field is usually the square patch around the given pixel.

A general CNN takes the path  $x_{RF(i)}$  as an input and predicts the signal  $s_i$  for each pixel  $i$ . Similarly, the denoising of an entire image can be obtained by feeding the network with overlapping patches. So, the CNN can be defined as

$$f(x_{RF(i)}, \theta) = \hat{s}_i \text{ where } \theta \text{ is the vector parameters to train}$$

Any supervised network is provided with a training pairs of clean image and corrupted image

$(x^j, s^j)$  where  $x^j$  is the noisy image and  $s^j$  is the clean ground truth. By applying the patch based receptive fields around each pixel in training data pairs as  $(x_{RF(i)}^j, s_i^j)$  where  $x_{RF(i)}^j$  is the patch around pixel  $i$  and  $s_i^j$  is the ground truth at same position. The loss in this method can be calculated as  $arg \min_{\theta} \sum_j \sum_i L(f(x_{RF(i)}^j, \theta) = \hat{s}_i, s_i^j)$ . A standard MSE loss is considered.

## 2.4 Network Architecture of DnCNN

DnCNN model handles general image denoising tasks. Generally, any CNN model requires specific steps for training.

1. Network Architecture - For DnCNN, VGG network architecture [14] is modified in order to perform specific image denoising task and depth of network is set based on previous state-of-art methods.
2. Model learning - Residual learning along with Batch normalization was formulated in order to achieve fast and stable training and better denoising performance.

### 2.4.1 Network Architecture

For better performance and efficiency, the main task is to set a proper depth in architecture design. DnCNN follows the architecture of VGG network, so the size of convolution filter is set to 3 x 3 but remove the pooling layers. As increase in the receptive field size uses the context information in larger image region, the receptive field of DnCNN with  $d$  is  $(2d+1) \times (2d+1)$ .

From [1] [2], the receptive field size of neural network for denoising methods correlates with the patch size of an image. Very high noise level requires effective patch size to capture context information. Different state-of-art techniques are analyzed in order to find the effective patch size. This is done by fixing the noise level  $\sigma = 25$ .

Below is a table that summarizes different methods and their patch sizes.

Table 1: Effective patch size of different methods with noise level  $\sigma = 25$

Methods	BM3D	WNNM	EPLL	MLP	CSF	TNRD
Effective Patch size	49 x 49	361 x 361	36 x 36	47 x 47	61 x 61	61 x 61

DnCNN is verified with the receptive field size similar to EPLL if it can compete against the leading denoising methods. So, the receptive field size of DnCNN is set to  $35 \times 35$  with depth  $d = 17$  for a gaussian denoising with certain noise level.

Input for any DnCNN architecture is  $y = x + v$ . DnCNN adopts residual learning formulation to train a residual mapping  $\mathcal{R}(y) \approx v$ . Whereas, MLP and CSF aim to learn a mapping function to predict latent clean image.

From residual mapping, we have  $x = y - \mathcal{R}(y)$ . The MSE for clean image and estimated image from noisy input is treated as the loss function to learn trainable parameters  $\theta$  in DnCNN.

$$\ell(\Theta) = \frac{1}{2N} \sum_{i=1}^N \|\mathcal{R}(y_i; \Theta) - (y_i - x_i)\|_F^2$$

$\theta$  trainable parameters

$\{(y_i, x_i)\}_{i=1}^N$  represents N pairs of clean and noisy training pairs.

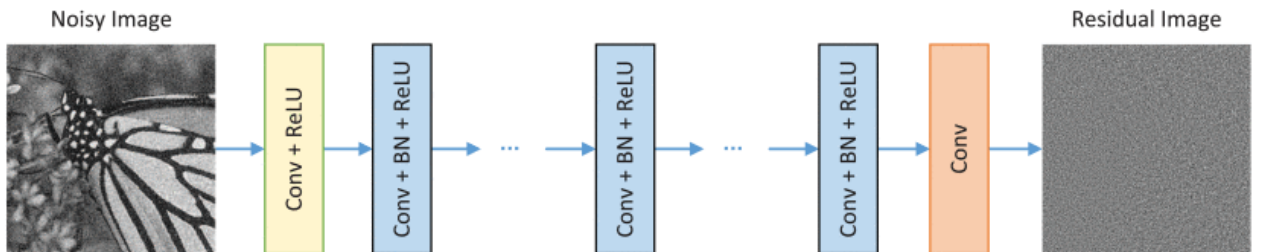


Figure 3: Proposed architecture of DnCNN



## Deep Architecture

There are three layers in the architecture, shown in the above figure.

1. Conv + ReLU: 64 filters of size  $3 \times 3 \times c$  are used to generate 64 feature maps and ReLU is used for nonlinearity.  $C$  is the number of color channels ( $c = 1$  for gray images;  $c = 3$  for color image).
2. Conv + BN + ReLU: 64 filters of size  $3 \times 3 \times 64$  are used and batch normalization is added between convolution and ReLU
3. Conv:  $c$  filters of size  $3 * 3 * 64$  are used to reconstruct the output.

This DnCNN has 3 main features, done residual learning is adopted to learn  $\mathcal{R}(y)$  and batch normalization is incorporated to speed up the training process and boost denoising performance. By adding ReLU with Conv, this network separated the noisy image from the hidden layers. This is similar to the iterative noise removal strategy adopted in different methods like WNNM and EPLL.

## Boundary Artifacts

The image deconvolution often produces undesirable artifacts in deconvolved images. These boundary artifacts depend on the type of deconvolution method involved. The output image size should be the same as input image, leading to boundary artifacts. In DnCNN method, zeros are directly padded to input image before convolution to make sure that feature map of the middle layer has the same size as the input image. By this simple padding strategy boundary artifacts are avoided.

### 2.4.2 Model Learning

Generally, networks can train wither the original mapping or residual mapping. When the noisy image  $y$  is more like the latent clean image  $x$ , residual image  $y$  is much easier to be optimized. Original mapping is closer to identity mapping than the residual mapping and residual learning is more suitable for image denoising.

Below graph shows the average PSNR values obtained using these learning formulations with/without batch normalization and with/without residual learning. It is adopted in two gradient based optimization algorithms: SGD and Adam.

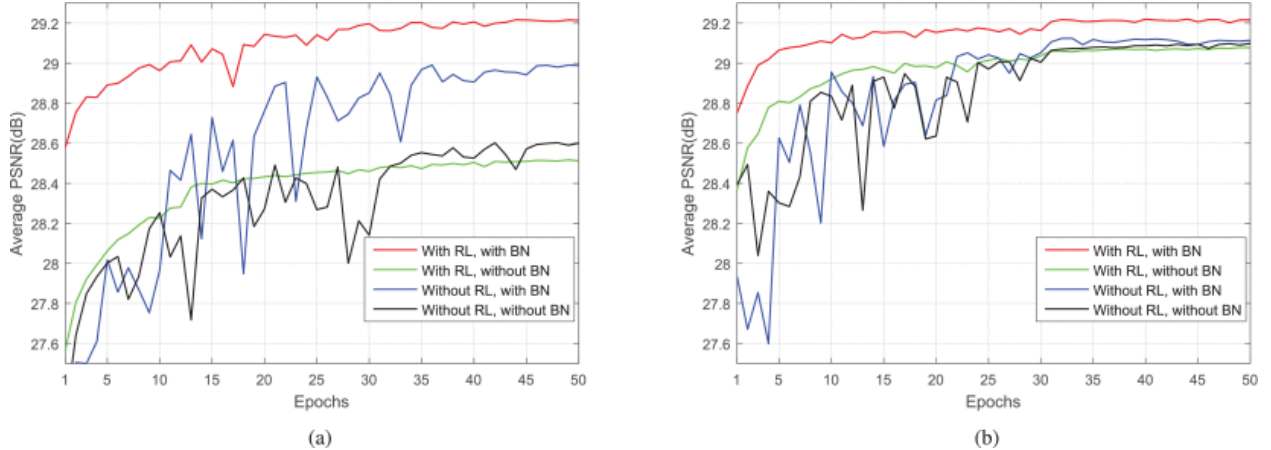


Figure 4: The Gaussian denoising results of four specific models under two gradient-based optimization algorithms, i.e., (a) SGD, (b) Adam, with respect to epochs. The four specific models are in different combinations of residual learning (RL) and batch normalization (BN) and are trained with noise level 25. The results are evaluated on 68 natural images from Berkeley segmentation dataset.

It is observed from the graph in both SGD and Adam algorithms, the combination of Residual learning and batch normalization yields the best result. We can conclude by saying instead of optimization algorithms, it is the combination of residual learning and batch normalization that increases the denoising performance of the network. Residual learning and batch normalization, both benefit from each other for Gaussian Denoising.

To sum up, the combination of residual learning and batch normalization can speed up and stabilize the training process and also boost the denoising performance.

### 2.4.3 DnCNN with unknown noise levels

Most of the state-of-art denoising techniques like Multi-Layer Perceptron, TNRD are all trained for specific noise levels. When applied to Gaussian denoising for a dataset with unknown noise level dataset is provided, the common way is to first estimates the noise level of the image and then apply the denoising algorithm. This affects the accuracy of the denoising algorithm due to noise estimation.

DnCNN in connection with TNRD, even if the noise is not gaussian or noise level is unknown in a distributed network, we can still obtain the residual mapping with the existing gradient descent inference step. So, it doesn't depend on whether the noise is Gaussian or non-Gaussian noise distribution like SISR and JPEG deblocking. In [15], it is demonstrated by extending the algorithm providing the dataset with unknown noise level ranging from  $\sigma \in [0, 55]$  to train a DnCNN algorithm. The given algorithm was able to denoise the test set without noise level estimation. Due to this functionality, DnCNN is considered as a single model to solve three specific tasks like Blind Gaussian denoising, SISR and JPEG deblocking. Several experiments are conducted by providing Gaussian noise dataset with unknown noise level, JPEG images with different quality factors to train the DnCNN network.

Extensive experiments on DnCNN denoising network with given noise level yields better results than state-of-art technologies like BM3D, TNRD. Even with the unknown noise levels, DnCNN results can still outperform BM3D and TNRD trained for a specific noise level. Moreover, DnCNN can be effective in training a single denoising network for general image denoising tasks like blind Gaussian denoising network, SISR and JPEG deblocking with different quality factors.

## 2.5 DnCNN with Microscopy Fluorescence Images

Traditional denoising methods often make idealistic assumption with the noise like Gaussian or Poisson noise. But noise in real-world such as high-ISO photos and Microscopic fluorescence are much more complex. Microscopic fluorescence images are not only much noisier than photography, but Poisson noise is the dominant noise source. A desirable effective denoising algorithm is required to get clean fluorescence microscopic images. DnCNN can perform better performance than many other denoising algorithms such as BM3D with unknown noise level.

Finally, the contributions of DnCNN denoising is summarized as below

1. An end-to-end CNN network is proposed which adopts the residual learning strategy instead of directly estimating the latent clean image from noisy observation.
2. Residual learning and batch normalization benefit the CNN by speeding up the training and increasing the denoising performance. DnCNN with certain noise levels outperforms the state-of-art methods in terms of visual quality.
3. We can train a single DnCNN network to solve general image denoising tasks like blind Gaussian denoising, SISR and JPEG deblocking.

Experimental results for real, synthetic and microscopic fluorescence can be discussed in section V.

# Chapter 3

## Unsupervised Learning

Unsupervised learning is a type of machine learning algorithm, which is used to draw an inference from the unlabeled data. This method cannot be applied directly to regression or classification problem but can be used to discover the underlying structures of image data. Expensive and time-consuming task to generate training labelled data promises the potential of unsupervised learning algorithms.

Challenges of unsupervised learning algorithm includes the underlying structure of a dataset, summarizing and grouping it most usefully and how to effectively represent data in compressed format when an unlabeled data is provided.

Compared to supervised learning, it is not always easy to come up with a metric that how well an unsupervised learning is working. Performance is always subjective and domain specific. Unsupervised learning tasks include clustering into groups and reducing dimensionality to compress data while maintaining the structure. Task of clustering is to group data points such that similar data points stay in a cluster and dissimilar datapoint stay away. Examples of clustering includes K-means clustering and hierarchical clustering. Dimensionality reduction is about reducing the complexity of the data while keeping the relevant structure possible. Two common practice techniques in dimensionality reduction is principal component analysis (PCA) and singular value decomposition (SVD).

Noise2Noise algorithm is considered, which is an unsupervised learning for training a neural network. In this algorithm we use the statistical reasoning to signal data to map corrupted objects to clean images. According to this method, it is possible to learn to restore images by only looking at noisy examples.

### 3.1 Introduction

Lehtinen introduced Noise2Noise training where pairs of corrupted images are used for training a network. It was observed that when certain statistical conditions are achieved this

network maps corrupted image pairs to output the average image. For large group of images, the target is a per-pixel statistics such as mean, median or mode over the stochastic process. Therefore, this Noise2Noise model can be supervised which used the noisy data by choosing the suitable loss function to recover the denoised image. This method eases the pain of collecting the noisy-clean pairs. This method still requires at least two independent realizations of the corrupted images.

Reconstructing a signal from noisy corrupted images is an important field of statistical data analysis. Recent advances in neural networks is to avoid using the traditional supervised learning approach and learn mapping corrupted observations to clean versions. A convolutional neural network is trained with a large number of pairs  $(\hat{x}_i, y_i)$  of corrupted noisy images  $\hat{x}_i$  and clean targets  $y_i$  by minimizing the empirical loss.

$$\underbrace{\operatorname{argmin}}_{\theta} \sum_i L(f_{\theta}(\hat{x}_i, y_i))$$

Where  $f_{\theta}$  is a parametric family of CNN mappings

L is a Loss function

We use  $\hat{x} \sim p(\hat{x}, x_i)$  which is random variable distributed according to the clean target. In this data model, training data includes pairs of short and long exposure images, incomplete or complete magnetic resonance images, microscopic fluorescence images, synthetic scenes etc., This algorithm provides advances in Gaussian denoising, JPEG deblocking and text removal. These results are compared with the state-of-art supervised methods like BM3D and DnCNN. At times it outperforms the supervised learning methods or likelihood models which uses labelled data for few sets of images in image denoising.

## 3.2 Technical background

Assume having a set of unreliable room temperatures  $(y_1, y_2, y_3...)$ . Common methodology to estimate the right room temperature is to find the smallest average deviation from the set of given temperatures with minimum loss. Function for loss L

$$\underbrace{\operatorname{argmin}}_{\theta} E_y \{L(z, y)\}$$

Loss L1 (Least Absolute Deviations) function used to minimize the error that is the sum of the absolute differences between noisy and clean observations.

$$\text{L1LossFunction} = \sum_{i=1}^n |y_{true} - y_{predicted}|$$

Loss L2 (Least Square Errors) function used to minimize the error that is the sum of all the squared differences between noisy and clean observations.

$$\text{L2LossFunction} = \sum_{i=1}^n (y_{true} - y_{predicted})^2$$

For L2 loss, the minimum is found at the arithmetic mean of the observations. Whereas, for L1 loss the optimum is at the median of the observations.

$$z = E_y\{y\}$$

Statistically, summary estimation using loss functions is an ML estimation by calculating a loss function as a negative log. Training neural networks is a kind of point estimation procedure. Observe a training example with a task for set of input-target pairs  $(x_i, y_i)$ , where  $f_\theta(x)$

$$\underbrace{\text{argmin}}_{\theta} E(x, y) \{L(f_\theta(x), y)\}$$

Removing the dependency of input data and using a trivial learned scalar, it reduces to (2). This whole training task is turned out to be minimization problem at every given sample. (4) is now equivalent to

$$\underbrace{\text{argmin}}_{\theta} E_x \{E_{y|x} \{L(f_\theta(x), y)\}\}$$

This minimize problem is solved using the point estimation problem for each given input sample. Thus, this property of loss is inherited into neural network training.

The usual process of training input-target pair is manipulated by mapping each input to multiple values instead of 1:1 mapping from input to target. It can be explained by an example. In any super resolution task over images, the low-resolution image can be explained by several high-resolution images as the exact positions and orientations are lost in decimation.  $p(y|x)$  is the highly complex distribution of low-resolution images. So, training a network with low- and high-resolution images using L2 loss, the neural network learns to average all the edges results spatial blurriness for the network prediction. This can be defined as learned discriminator functions as losses.

This property of the trained neural network has an unexpected benefit. One of the properties of L2 minimization is that the estimation of a target remains unchanged even when the input is changed with random number whose value matches the target. From equations (3) and (5), we can say that the equations remain unchanged even if distribution of input and output are changed as the arbitrary distributions having the same expected values. We can corrupt the training data of neural network with zero-mean noise without change in the learning data. From equation (1), we can say that input and targets are now from a noisy image such that the unobserved clean target is  $\mathbb{E}\{\hat{y}_i|\hat{x}\} = y_i$ . We do not need any explicit  $p(\text{noisy}|\text{clean})$  or  $p(\text{clean})$  to train a neural network when we have the data distributed accordingly.

With the above information, image restoration tasks can be solved using the corrupted set of images instead of clean targets. A long exposure, noise-free photo in low light photography is the average of several short, independent, noise exposures. This can be an example of getting expensive or potentially long exposure photographs by removing the photon noise using pairs of noisy images. Similar observations can be made with l1 losses as well.

### 3.3 Noise2Noise Training

Training a Noise2Noise is different than traditional method that can be denoised without any ground truth data available. Pairs of noisy images  $(x^j, x'^j)$  are inputs for Noise2Noise method.

$$x^j = s^j + n^j \text{ and } x'^j = s^j + n'^j$$

Where  $n^j$  and  $n'^j$  are noise components and are independent samples from the same distribution. Patch based perspective is again applied on training data pair  $(x_{RF(i)}^j, x'^j_i)$

$x_{RF(i)}^j$  extracted from noisy input  $x^j$

$x'^j_i$  is noisy target from  $x'^j$  at position i

Similar to traditional method, parameters are tuned to reduce the training loss just that noisy target is different from ground truth provided in traditional method. Though the mapping is learned from noisy target, the Noise2Noise method will still provide a clean image as output. The important fact is the expected value of noisy input is same as the clean image.



## 3.4 Deep Architecture

Two different architectures were used in Noise2Noise to determine that clean targets are unnecessary in this application.

1. RED30
2. U-Network

### 3.4.1 RED30

A baseline state-of-art method called “RED30” is used for few experiments. It is constructed a 30-layer hierarchical residual network with 128 feature maps. To train this network, IMAGENET dataset has been used. We obtain 256 X 256-pixel crops drawn from 50k images with different noise level ranging from  $\sigma \in [1,50]$ .

In this blind denoising network, the RED30 has to estimate the noise level while removing it. Experiments with different datasets like BSD300, SET14, KODAK has been performed. When compared with the state-of-art BM3D method, RED30 achieves  $\sim 0.7\text{dB}$  better results and the denoising performance still hold good.

### 3.4.2 U-Network

U-network is used for much deeper network and is roughly 10 times faster training speed and still achieves similar results.

The network architecture consists of contracting path and expansive path. The contracting path typically have a convolutional network consisting of repeated application of 3x3 convolutional networks followed by 2x2 max pooling operation with a stride of 2. Every step in the down sampling we double the number of feature channels. Expansive path contains up sampling of feature channels that halves the number of feature channels, a concatenation with the cropped feature map and two 3x3 convolutions followed by ReLU. Altogether there are 18 convolutional layers.

Except for the first experiment, this application uses U-Net for rest of them. All the basic text removal and noise removal with RGB images the output channels are set up  $n = m = 3$ .

For Monte Carlo denoising with input as RGB pixel color, 3D vector per pixel  $n = 9, m = 3$ . MRI reconstruction experiments are done using  $n = m = 1$ .

This network does not have batch normalization, dropouts or other regularization techniques. Training is done with ADAM. Experiments were conducted with a learning rate of 0.001.

NAME	$N_{out}$	Function
INPUT	n	
ENC_CONV0	48	Convolution3×3
ENC_CONV1	48	Convolution3×3
POOL1	48	Maxpool2×2
ENC_CONV2	48	Convolution3×3
POOL2	48	Maxpool2×2
ENCCONV3	48	Convolution3×3
POOL3	48	Maxpool2×2
ENC_CONV4	48	Convolution3×3
POOL4	48	Maxpool2×2
ENC_CONV5	48	Convolution3×3
POOL5	48	Maxpool2×2
ENC_CONV6	48	Convolution3×3
UPSAMPLE5	48	Upsample2×2
CONCAT5	96	Concatenate output ofPOOL4
DEC_CONV5A	96	Convolution3×3
DEC_CONV5B	96	Convolution3×3
UPSAMPLE4	96	Upsample2×2
CONCAT4	144	Concatenate output ofPOOL3
DEC_CONV4A	96	Convolution3×3
DEC_CONV4B	96	Convolution3×3
UPSAMPLE3	96	Upsample2×2
CONCAT3	144	Concatenate output ofPOOL2
DEC_CONV3A	96	Convolution3×3
DEC_CONV3B	96	Convolution3×3
UPSAMPLE2	96	Upsample2×2
CONCAT2	144	Concatenate output ofPOOL1
DEC_CONV2A	96	Convolution3×3
DEC_CONV2B	96	Convolution3×3
UPSAMPLE1	96	Upsample2×2
CONCAT1	96 + n	Concatenate INPUT
DEC_CONV1A	64	Convolution3×3
DEC_CONV1B	32	Convolution3×3
DEV_CONV1C	m	Convolution3×3, linear act

Table 2: Network architecture used in Noise2Noise.  $N_{out}$  denotes the number of output feature maps for each layer. Number of network input channels  $n$  and output channels  $m$  depend on the experiment. All convolutions use padding mode same and last layer is followed by leaky ReLU. Other layers have linear activation.

This U-Net architecture achieves performance on biomedical image segmentation applications like microscopic fluorescence images.

Few experiments are conducted by adding a synthetic noise called Poisson noise to the image. Poisson noise is the dominant source of noise in many photographs where zero-mean noise is signal independent and hard to remove it. Training is done with L2 loss and by varying the noise magnitude. Dark current and quantization are dominated by Poisson noise, can be made zero-mean and hence pose no problems for training with noisy targets.

## Monte Carlo Rendering

Monte Carlo is a path tracing method in computer graphics for rendering three-dimensional images such that the illumination is real. In this process physically accurate renderings are generated. This is the random sequence of scattering lights connecting light source, virtual sensors and their radiance in the possible paths. It is constructed such that intensity at each pixel is the sampling to the zero-mean. But there are very few sampling techniques that suits this distribution. Some effects like lighting, scene configuration vary from pixel to pixel and results in rare distributions and bright outliers.

All this effect makes it difficult to remove noise from images compared to Gaussian and Poisson. This Noise2Noise not only has luminance values but also the texture color and normal vector of surface visible at each pixel.

This combination of unbounded luminance and non-linearity is a problem in denoising. If the denoiser is trained to output a luminance values, a standard MSE loss will be dominated by outliers. Or if the denoiser is trained to output a tone mapped values, the nonlinearity would make the output image different from the expected clean image. This nonlinearity problem with tone mapped outputs exists with the metrics MSE which is used to measure the quality of HDR images. This is resolved by considering the denominator of the gradient to be zero to achieve the correct output values. It is observed from our experiments that it is always better to use tone mapped inputs instead of HDR images. This retains the target denoised as the expected value. This network is trained with a set of 860 architectural images. It takes more time to render these images on a NVIDIA tesla GPU.

### 3.4.3 MRI Images

Magnetic Resonance Images have a set of biological tissues that are essential by sampling the Fourier transform. Recent changes in MRI techniques depends on the principle of compressed sensing. They underestimate the k-space Fourier transform and apply non-linear reconstruction in a suitable transform domain.

This regression problem is trained with a convolutional neural network by a pair of noisy images with L2 loss as the Fourier transform is linear. And for additional improvement we apply Fourier transform on the frequencies of the input and then transforming it back to the domain by applying inverse transform before computing the loss. This process is trained end to end. IXI brain MRI dataset is used to train this network. For simulating the spectral sampling, we draw random samples from the FFT of the images in the dataset. Hence this is applied on a real valued with FFT built in support. Very high restoration can be observed when training this network with the noisy data.

## 3.5 Noise2Noise on Microscopic Fluorescence images

A noise2Noise model is trained with a noisy image and outperforms many state-of-the-art supervised methods on Poisson noise which need a clean and noisy image pairs. FMD dataset is trained with the noise2Noise network which dominates in Poisson noise for two-photon and confocal microscopy images and wide-field microscopy with Gaussian noise. This method shows better performance than with few traditional methods trained with real noisy images. For working with FMD dataset, few estimation parameters are required like scaling coefficient and Gaussian noise variance. Then the training is performed with Noise2Noise denoising algorithm with the above parameters. Noise2Noise network is trained with different noise levels with samples of randomly selected mini batches for 400 epochs. This training may take less than 1ms when run on a GPU that enables real time denoising up to 100 frames per second. This outperforms several traditional denoising methods. With such performance in denoising an image and high speed, Noise2Noise method can efficiently benefit real time noisy microscopic fluorescence imaging. This training confirms that Noise2Noise model has similar performance as DnCNN without providing any clean images. These results show us that the deep learning denoising algorithm that are trained with FMD dataset outperforms other methods by large margin in all the different modalities and noise levels.

# Chapter 4

## Self-Supervised learning

### 4.1 Introduction

Several architectures have been developed in order to remove noise from images by applying deep learning. Architectures like U-nets, residual networks, residual dense networks have been introduced. These models are trained as a Supervised model with noisy images as input and clean images as output. The above networks are trained to remove different kinds of noise in the images. Self-supervised learning is a kind of machine learning technique which is a promising alternative where tasks are developed that allow models to learn without explicit supervision and help to perform the task of denoising. Major benefits of using this self-supervised technique is increasing data efficiency like achieving comparable or better performance without labelled data or reinforcement learning.

This field of self-supervised learning is rapidly increasing, and performance of these methods is comparable to the fully supervised models. This kind of learning does not need access to clean image targets or pairs of noisy images. This can be applied in situation when the data is difficult to acquire or is really an expensive task. There are several algorithms designed based on this self-supervised learning approach by employing a blind spot network in receptive fields. This significantly improves aspects like image quality and training efficiency. The results obtained by such algorithms are comparable with the denoising methods that handle Gaussian noise, Poisson noise and impulse noise.

Lehtinen introduced Noise2Noise as a method that removes the need for having a noisy-clean image pairs which eases data collection significantly, large collection of poor images is still required. Observation according to this method states that when a certain statistical condition is met, a network designed to map impossible corrupted images to corrupted image learns to output the average image. This restoration model can be supervised using noisy data by choosing the appropriate loss function to recover the statistic of interest. The motivation behind introducing this Self-supervised model, how much can we learn by looking at the pairs of corrupted images. Based

on several experiments on Noise2Noise model with different noise types we can say only minor concessions in denoising performance are necessary. It was also identified that all the parameters in denoising algorithm need not be known in advance. In case when the ground truth is physically unavailable, Noise2Noise can still enable the training of the denoising algorithm. However, this algorithm requires two images capturing the same content with independent noises.

Along with these advantages, Noise2noise have few other drawbacks.

1. Noise2Noise requires pairs of noisy images.
2. Gathering such image pairs with constant  $s$  is possible only for static scenes.

To overcome these limitations of an unsupervised network Noise2Noise, a new algorithm is introduced. Noise2Noise is a self-supervised algorithm that required neither clean-noisy image pair or nor noisy image pairs.

## 4.2 Technical Background

Noise2Void is designed by inspiring from the Noise2Noise algorithm. This algorithm needs no image priors, but just uses individual noisy image as training data by the assumption that the corruption is zero-mean and independent between pixels. Noise2Void is a self-supervised method that overcomes the high-quality denoising models and can be trained without availability of clean-noisy image pairs. Unlike Noise2Noise or any other traditional denoising method, Noise2Void can be applied to data that have neither noisy-clean image pairs or noisy image pairs are available. For designing this method a few basic assumptions were made.

1. Signal in an image is not pixel-wise independent.
2. The noise in an image is conditionally pixel-wise independent when the signal  $s$  is given.

We used BSD68 dataset along with FMD dataset in order to check the performance of Noise2Void. These results are in turn compared to other unsupervised algorithm like Noise2Noise and traditional algorithm with supervised learning like DnCNN. Though Noise2Void might not outperform other methods which are trained with clean-noisy image pairs or noisy image pairs, but the results are comparable and outperforms with few noise levels.

This algorithm is also applied to FMD Microscopic Fluorescence dataset which cannot be applied to traditional denoising algorithms due to lack of clean images. Even Noise2Noise cannot be applied to few of the datasets since datasets are not available. This helps us better understand the practical importance of Noise2Void method.

Noise2Void method is based on blind-spot network where the center pixel is not included in the receptive fields. In this way we can use the same image for both training and testing. Since the center pixel is not considered in receptive fields, using the same image is similar to using different noisy image. This method is termed as self-supervised method as the same image is used to predict the clean image without separate training data.

Contributions for this method includes

- Introduction of an approach which requires a body of single noisy image in order to train a denoising network.
- Comparing the results of self-supervised network to traditionally approached supervised learning and unsupervised learning.
- A detailed description of the efficient implementation of Noise2Void method.

These deep learning methods are trained for extracting information from ground truth and then apply this model to the test data. In [1], Jain applied the convolution neural networks for denoising images. They used the set-up of denoising the regression task and the network tries to minimize the loss between the predicted image and the ground truth image provided. Later in [17], Zhang introduced a state of the art denoising results by introducing denoising convolutional neural networks based on the idea of residual learning. In this method, the CNN not only predict the clean image but instead noise at every pixel was determined. This allows to train the denoising CNN with different ranges of noise levels. Their architecture is distributed with several pooling layers. Then a complimentary very deep encoder-decoder architecture has been developed in [18] for the purpose of denoising task. This network also uses the residual learning but introduces

skip connections between corresponding encoding and decoding modules. This network also uses a single CNN for different noise levels.

In [18], Tai used recurrent persistent memory units as part of the above architecture. [19] is based on the task of image restoration in microscopic fluorescence data. For this purpose, pairs of several low and high frequency images are collected. This data collection can be difficult as the biological samples should not move in between the exposures. Noise2Void uses this method as the starting point, including their U-Net architecture.

N2V is applied on [17] and [18] as their architecture requires noisy input at each pixel. This input is masked in N2V when the gradient is calculated.

#### 4.2.1 Internal statistics method in Noise2Void

Internal statistics method will not be trained on ground truth instead it is directly applied on the test image where required information is extracted. Noise2Void follows this internal statistics method and enables the training directly on a test image. Like [20], N2V is a classic denoising algorithm that predicts pixel values based on the noisy surrounding

BM3D is classic example of Internal Statistics method. It's idea is based on the idea that noisy images usually contain repeated patterns of data. This method performs denoising by grouping similar patterns and filtering them. But this approach has high computational cost during training. This is overcome by Noise2Void method since computation is performed only during training. Once the network is trained, it can filter out similar patterns in any number of images.

Deep image prior is also a method which shows that the CNNs resonates with the distribution of natural images and it can be used for image restoration without any additional training. This network produces a regularized denoised image as output.

Finally, this Noise2Void is followed using [21] where the network might not be used for denoising, but it trains a neural network to predict the unseen pixel based on its surroundings. This network is trained to predict the probability distribution for each pixel for regression task. However, Noise2Void differs in structure of receptive fields which masks the central pixel in a square receptive field.



### 4.2.2 Image formation model

A noisy image is composed of  $x = s + n$  and its disjoint distribution

$$p(s, n) = p(s)p(n|s)$$

Where  $p(s)$  is the arbitrary distribution where  $i$  and  $j$  are two pixels within a certain radius

$$p(s_i | s_j) \neq p(s_i)$$

The pixels of  $s_i$  are not statistically independent. Assume the conditional distribution

$$p(n|s) = \prod_i p(n_i | s_i)$$

Here the pixel values of  $n_i$  is conditionally independent when signal is given. Assume the noise is zero-mean

$$E[n_i] = 0 \text{ which leads to } E[x_i] = s_i$$

If multiple images with same signal were provided with different noises and realizations the result would be equal to ground truth.

### 4.3 Noise2Void Training

In Noise2Void, both the training and test samples are derived from the single noisy image  $x^j$ . We simply extract a patch from the training data and treat it as an input and its center pixel as target, N2V will learn to identify by mapping the value at the center of the input patch to the output.

In Spite of training the network with a clean image, training N2V using a single noisy image is still possible. This learning algorithm has an architecture with some special receptive fields. In N2V algorithm we assume the receptive field  $\mathcal{X}_{RF(i)}$  for the network have a blind spot in the center.

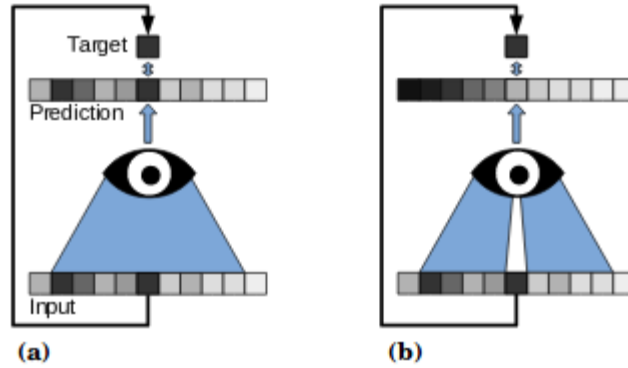


Figure 5: Conventional neural network along with the blind-spot network. (a) In a conventional neural network, the prediction for each pixel depends on the square patch of the input pixel known as receptive field of the pixel. Training a network with such noisy image as input and target, the network will degenerate and simply learn the identity. (b) In a blind-spot network, the receptive fields of each pixel excluded the center pixel to learn its identity. This will remove the pixel wise independent noise when trained on same noisy and target image.

This CNN prediction is affected by all its square patch pixels except the center pixel at very location. This type of network is termed as blind-spot network. Blind-spot network can be applied with any of the algorithms like traditional or Noise2Noise using the clean target or noise target. Since the center pixel is it considered, the network will have slightly less information for prediction. So, the accuracy of the network can be a bit less when compared to the regular network. However, this can still perform better as it is removing only one pixel out of the entire receptive fields.

The main advantage of using a blind spot network is it becomes difficult to learn the identity of the center pixel. From our basic assumption that the noise is pixel-wise independent, the neighboring pixels will have no information about the noisy pixel. This becomes difficult to produce an output better than apriori expected value (zero-mean for noisy pixel). However, the signal still contains statistical dependencies. Because of this the network can still predict the signal by looking at the neighboring pixels. As a result, blind-spot network will allow us to find the input and target patch from the same noisy image.

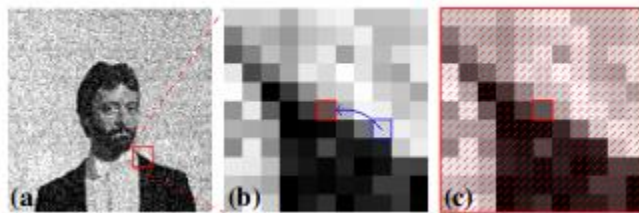


Figure 6: Blind-spot masking scheme used in Noise2Void Training. (a) Noisy training image (b) A randomly selected pixel (blue rectangle) is chosen and intensity is copied over to create blind-spot (red and striped square). It is used as input for training. (c) target patch corresponding to selected pixel. Original input with unmodified values is also used as target. Loss is calculated for blind spot images that are masked.

We have to train this network further by minimizing the empirical loss.

$$\arg_{\theta} \min \sum_j \sum_i L(f(x_{RF(i)}^j; \theta), x_i^j)$$

We should note that the target is same in Noise2Noise as well as Noise2Void which is extracted from the second noisy image. By comparing it with Noise2Noise, both the targets have equal signal and the noise is independent samples from the same distribution. Blind spot network can be trained by only using individual noisy images. A new scheme has been proposed to mask the center pixel and randomly select any pixel from the neighboring pixels such that it effectively prevents the network to learn the identity.

## 4.4 Network Architecture

Noise2Void uses U-Net architecture to which batch normalization is added before each activation function. CSB Depp framework, a toolbox for Content Aware Image Restoration (CARE) is the basis for its implementation. In this network we process an entire patch to calculate the gradients for single noisy image.

In a given noisy training image, they extract a random patches of size 64x64 pixels, which are greater than receptive field size. Within each patch few random pixels are selected, and different masking techniques are used in order to avoid clustering. Different masking techniques include Uniform Pixel Selection (UPS), Gaussian(G), Gaussian Fitting(GF), Gaussian Pixel Selection(GPS) with different kernel size, loss functions and features are applied.

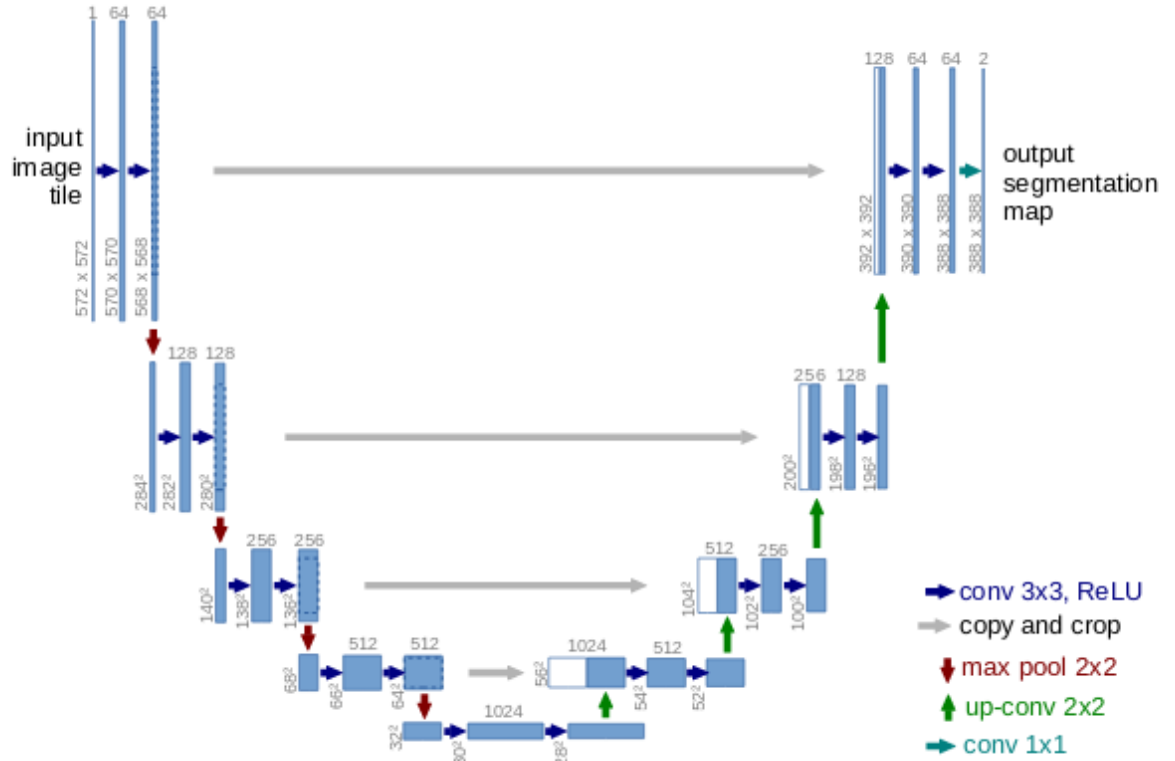


Figure 7: U-net architecture with 32x32 as the lowest resolution. Blue box denote a multi-channel feature map. Channels are denoted on top of the box. White box denotes the copied feature maps. X-y size is denoted at lower edge of the box.

This network is trained for 200 epochs with each epoch containing 400 gradient steps. An on-the-fly random sub-patch(64x64) extraction is performed. Validation loss is calculated similar to training loss for any traditional and Noise2Noise losses. They use the randomly selected masked pixels for calculating the validation loss. Simultaneously gradients are calculated while ignoring the predicted image. This training is done using keras pipeline with specialized loss function zero for all the selected pixels.

The receptive fields for training are selected based on the CARE framework. With a kernel size of 3x3 and depth 2, the receptive field size is set to be 22x22 pixels. With a kernel size of 5x5, receptive field size is set to 40x40. Different masking methods with different kernel sizes are applied on the network to check the performance of the network. BSD68 dataset is used in order to test different settings and different variants of the masking scheme. For few schemes the loss is calculated using Mean Absolute Error instead of Mean Square Error.

## 4.5 Noise2Void on Microscopic Fluorescence Images

For microscopic images, Noise2Void method is unfortunately not competitive with models that are trained with clean image data or noisy image data. So, by combining this with some suitable description of noise, a complete probabilistic noise model with per-pixel intensity distribution is generated. With this model we can obtain a noisy signal and true signal in every pixel. This method is applied on FMD dataset with broad range of noise and achieved competitive results in state-of-the-art supervised methods. In self-supervised methods like Noise2Void the assumptions are that the images are pixel wise independent and the true intensity of the pixel can be predicted from the blind-spot network mentioned above. The second assumption is not fulfilled for microscopic image data and this needs some improvement for the existing model.

This problem of self-supervised networks is addressed by Laine et al. in [22] by assuming the Gaussian noise model and predicting its intensity per pixel. This can be applied to other noise models as well and it was described well in PN2V. This model combines a general model that can be represented as histogram and distribution of pixel intensities is represented by a set of predicted samples. MMSE is used to estimate the final prediction and it consistently outperforms other supervised techniques.

Instead of using the idea of masking, PN2V trains the CNN to describe the probability distribution. By integrating over all the possible clean signals, the probability of observing a pixel the pixel value when the surrounding receptive fields are given can be considered as an unsupervised learning task. The loss function in this method is considered from the interpreted as independent samples and drawn from  $p(s_i | x_{RF(i)}; \theta)$ . This summation needs to be performed on a GPU. Minimal mean Squared Error is used to find sensible estimates for every pixel based on the probabilistic model by weighing the predicted samples with the observed likelihood and calculating the average.

This model is applied on the FMD dataset with different samples and different imaging conditions. This method uses the arbitrary noise model by analyzing the available set of images with the same noise. PN2V is challenging in low-light conditions where noise is the typical limiting factor for analysis.

## Chapter 5

### Comparison of Results and Summary

In this section, we will present the experimental settings and results of our image denoising methods and compare them for real, synthetic and microscopic fluorescence images.

#### Experimental settings for DnCNN

Datasets used: Set14, BSD68, CBSD68, LIVE1

For Gaussian denoising with unknown noise level, we followed 400 images of patch size  $50 \times 50$  to train a single DnCNN model for blind Gaussian denoising.  $128 \times 3000$  patches are cropped to train the model.



*Figure 8: The left is the ground-truth, middle is the noisy image corrupted by AWGN, the right is the denoised image by DnCNN. Average PSNR (dB)/SSIM with Noise Level 25 on BSD68 dataset is 30.4358/0.8618.*

In addition to gray scale image denoising, we also trained the blind image denoising network DnCNN with 432 color images of Berkeley Segmentation Dataset. And CBSD68 for the purpose of testing. The noise levels are set in the range of  $[0, 55]$  and  $128 \times 3000$  patches of size  $50 \times 50$  are cropped for training this model and is referred as CDnCNN



Figure 9: The left is the ground-truth, middle is the noisy image corrupted by AWGN, the right is the denoised image by DnCNN. Average PSNR (dB)/SSIM with Noise Level 35 on CBSD68 dataset is 31.5531/0.8190.

DnCNN is to learn a single model for three general imaging denoising tasks like Blind Gaussian denoising, SISR (Single Image Super Resolution) and JPEG deblocking. All these images are given as input to the network with different downscaling factors, quality factor. Totally we generated 128 x 8000 image patch pairs for training. We refer this training as DnCNN-3.



Figure 10: The left is the ground-truth, middle is the noisy image corrupted by AWGN, the right is the denoised image by DnCNN. Average PSNR (dB)/SSIM with Noise Level 25, Upscaling Factor 3, Quality Factor 20 on BSD68 dataset is 32.17/0.8995.

## Experimental settings for Noise2Noise

Datasets used: ImageNet, BSD300, SET14 and KODAK

We train our Noise2Noise network using 256 x 256 pixels crops drawn from 50k images in the IMAGENET dataset. A randomized standard deviation of [0, 50] is used for each training sample where the network has to estimate the magnitude of noise before removing it. No batch normalization, dropout or other regularized techniques are used. Training was done using ADAM.



The number of input and output channels for the colored BSD dataset is  $n = 3$ ,  $m = 3$ . KODAK dataset is used as a validation dataset.

Learning rate of 0.0003 is kept constant during the training and the same is used for all the experiments with Noise2Noise.



Figure 11: The left is the ground-truth, middle is the noisy image corrupted by AWGN, the right is the denoised image by Noise2Noise. Average PSNR (dB) on BSD dataset is 32.88

Table 3: Below is the average PSNR observed when the network is trained with different noise and different denoising methods.

Noise	Dataset	Average PSNR (dB) Noise2Noise	Average PSNR (dB) DnCNN
Gaussian	BSD300	31.04	31.553
Poisson	BSD300	30.33	-

## Noise2Noise on MRI data

Dataset used for MRI training is IXI-T1. Once the training set is downloaded, subset of this data is converted into training and validation sets. By default, Noise2Noise training is enabled for the MRI images. This network is training for 300 epochs and a learning rate of 0.0003 only on pairs of noisy images.

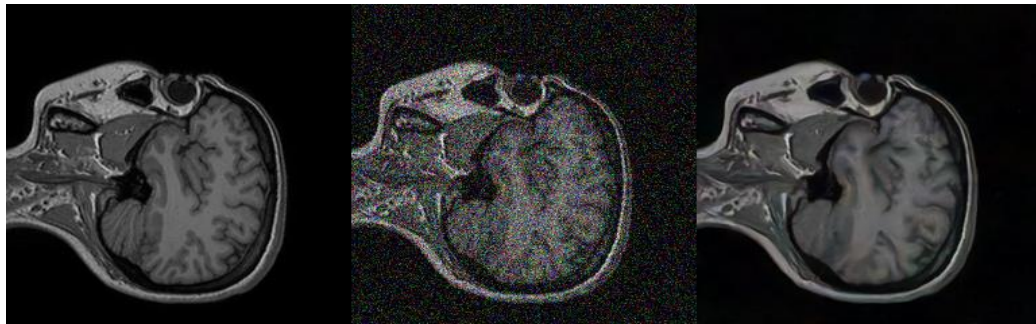


Figure 12: The left is the ground-truth, middle is the noisy image corrupted by AWGN, the right is the denoised image by Noise2Noise. Average PSNR (dB) on IXI-T1 dataset is 31.61



## Experimental settings for Noise2Void

This method does not require either ground truth image or noisy image pairs as training is not required. But training is done on the input image itself.

Noise2Void is applied on natural images, simulated biological image and microscopic fluorescence images. And the results of the network are compared to DnCNN and Noise2Noise. The network used for Noise2Void is a U-Net with depth 2 and kernel size 3 with batch normalization and linear activation function in the last layer. For Noise2Void training we use a batch size of 128 and initial learning rate of 0.0004

A noisy image with a dimension of 1100 x 2800. This model generates a patch of 64x64 and are divided into training and validation sets. This model trained with 25 epochs to execute the notebook faster. For better results we might increase the epoch size to 100 - 200 for training. Once a model is generated, prediction is done using the model created in training.



*Figure 13: Left is the noisy image, right is the denoised image by Noise2Void. PSNR (dB) is 29.31*

## Denoising Microscopic Fluorescence for FMD dataset

This network is implemented using Jupyter notebook where training and testing is performed on FMD dataset with convallaria MICE data. This experiment is done on FMD dataset with confocal modality. The FMD dataset is split into training and test sets, where training set is composed of images randomly selected from different FOVs for each image configuration and noise levels.

Considering GPU memory constraint, the network is trained with images cropped into four non overlapping patches of size 256 x 256. Instead of testing with pre-train models, we re-trained the model with same network mentioned in the architecture with similar hyper parameters on the FMD dataset from scratch.

### DnCNN

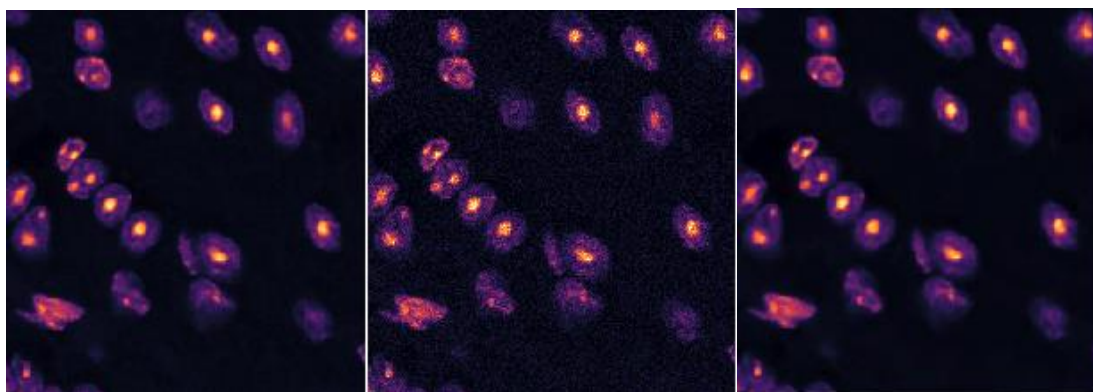


Figure 14: The left is the ground-truth, middle is the noisy image corrupted by AWGN, the right is the denoised image by DnCNN. Average PSNR (dB) on FMD dataset with CONFOCAL\_MICE data is 34.533

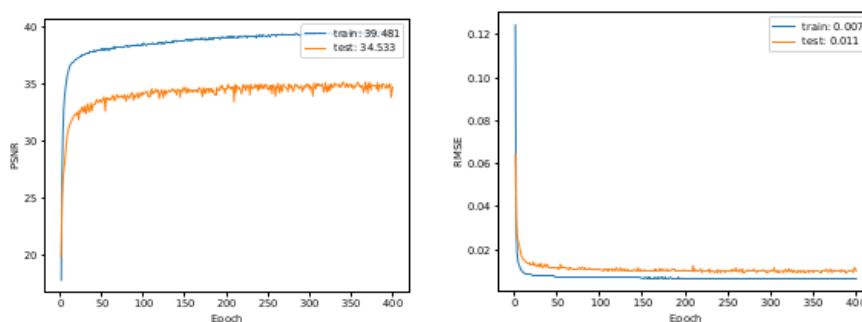


Figure 15: PSNR and MSE on the mixed test set with raw images during training. Each training epoch contains images of size 256 x 256. Batch normalization helps stabilize training DnCNN, residual learning help improve denoising.

## Noise2Noise

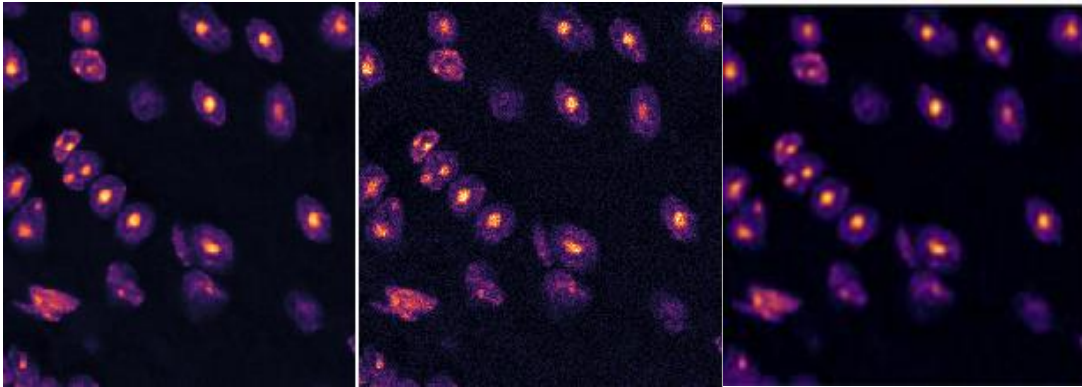


Figure 16: The left is the ground-truth, middle is the noisy image corrupted by AWGN, the right is the denoised image by Noise2Noise. Average PSNR (dB) on FMD dataset with CONFOCAL\_MICE data is 35.47

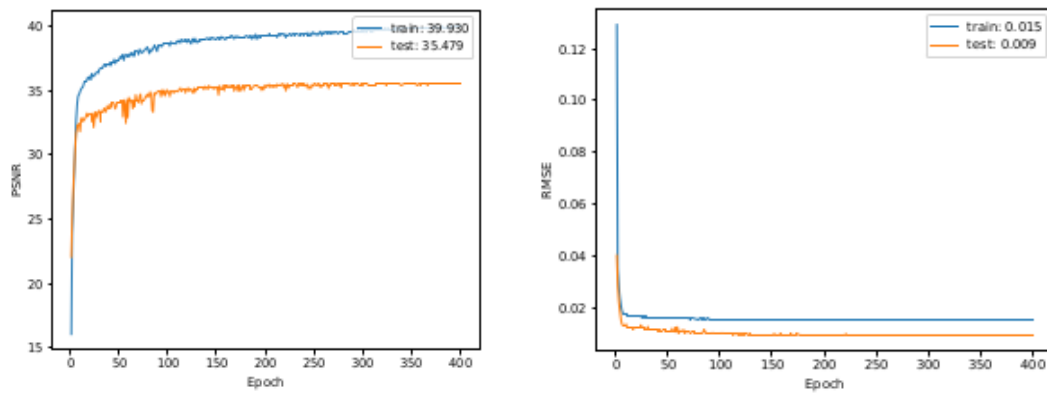


Figure 17: PSNR and MSE on the mixed test set with raw images during training. Each training epoch contains images of size 256 x 256. Batch normalization helps stabilize training Noise2Noise, residual learning help improve denoising.

## Noise2Void

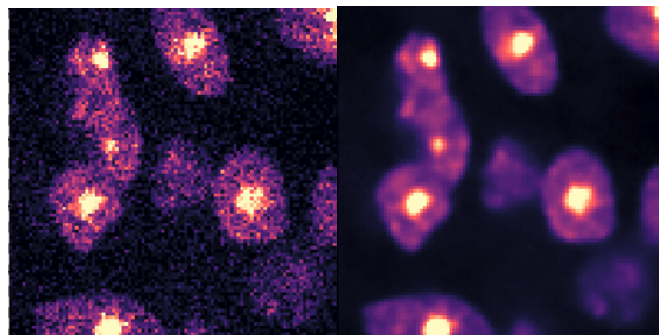


Figure 18: The left is the ground-truth, the right is the denoised image by Noise2Void. Average PSNR (dB) on FMD dataset with CONFOCAL\_MICE data is 37.57






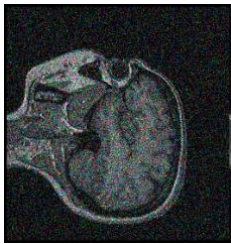
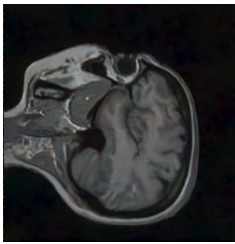

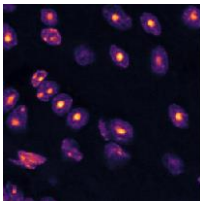
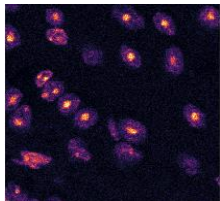
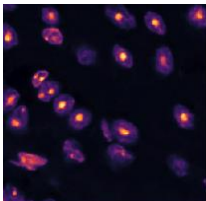
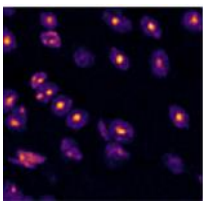
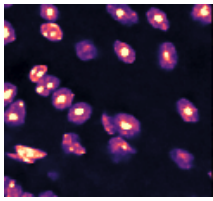
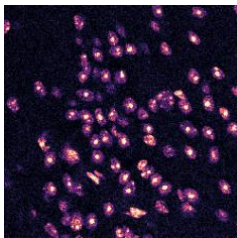
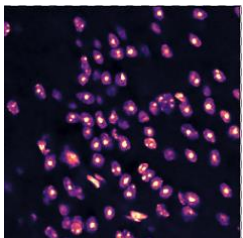
	Ground Truth	Input	DnCNN	Noise2Noise	Noise2Void
BSD300			 PSNR: 31.553dB	 PSNR:31.04dB	 PSNR:29.31dB
IXI-T1	Does not exist		Does not exist	 PSNR:31.61dB	 PSNR:29.42dB
Confocal -MICE			 PSNR:34.533dB	 PSNR:35.47dB	 PSNR:37.57dB
Two-Photon MICE	Does not exist		Does not exist	Does not exist	 PSNR:37.57dB

Figure 19: Results and average PSNR values obtained by DnCNN, Noise2Noise and Noise2Void trained denoising network. For few of the denoising methods, the datasets are not application due to unavailability of clean-noisy targets or noisy pairs for training.

## Summary

Denoising convolutional neural network (DnCNN) was proposed when residual learning is adopted in order to separate noise from clean image. DnCNN implements batch normalization and residual learning to speed up training speed and improve the denoising performance. This blind denoising method has the capacity to handle Gaussian denoising using unknown noise levels and also other denoising tasks like SISR and JPEG decompression with different quality factor. Extensive experiments were conducted on different datasets including Gray and Color images. These experiments show that this method works effectively with Gaussian noise than real-world images in biomedical sets.

Noise2Noise method shows that a simple statistical argument leads to new capabilities in learning about the signal using deep neural networks. It is always possible to recover signals just by looking at the noisy observations and still the performance levels are equal or close to using clean-noisy image pairs. With the experiments conducted, we show that the high-resolution performance of deep neural networks can be achieved entirely without clean data. This is a benefit for several applications like microscopic data that do not have a collection of clean image pairs.

The latest research on denoising noisy images is using Noise2Void. It was observed that this training requires only single noise image for denoising CNNs. This method is applied on different sets of data like general photography and fluorescence microscopy. As long as the initial predictions like pixel wise independent noise are met, this method can be compared to traditional methods and Noise2Noise networks. This can be a powerful denoising network in the field of biomedical imaging data like microscopic fluorescence structures.

# Chapter 6

## Future Work

### Convolutional Blind Denoising of Real Photographs - CBDNet

Deep Convolutional Neural Networks have achieved impressive results in image denoising with Additive White Gaussian Noise (AWGN). But their performance is limited to these particular noise types. These cannot be applied to real-world noisy images as the learned models from CNN are easy to overfit which deviates from real world noise models. In order to improve the ability of deep CNN denoisers, a new network has been developed to handle realistic noise models with real-world noisy clean image pair. Different tasks handled by this network includes signal dependent noise/signal processing pipeline is considered for synthetic realistic noise and the network is trained for real-world noisy photographs to train CBDNet. For rectifying the denoising result conveniently, a noise estimation network is also included into CBDNet.

Though gaussian denoising performance have been significantly improved, denoisers for blind noise removal for real photographs still degrades. Also, denoisers for non-blind denoising also suffers from smoothing out the details in the process. This phenomenon can largely depend on the memorizing ability of large-scale training data. So, in this CBDNet they developed a convolution blind denoising network to tackle the problem with real world noisy photographs. Generally, the performance of a denoiser depends on the distributions of real and synthetic images. So, training a network with realistic noise models in blind image denoising is the major problem. In CBDNet assumption, Poisson-gaussian distribution is considered as an alternative for AWGN raw noise modeling. And also, in-camera processing also increases the complexity of noise. Therefore, CBDNet considers both gaussian Poisson noise as well as in-camera processing into consideration for this noise model. In-camera processing includes demosaicing, gamma correction, JPEG compression etc., These processing play vital role in realistic noise models and achieves noticeable performance.



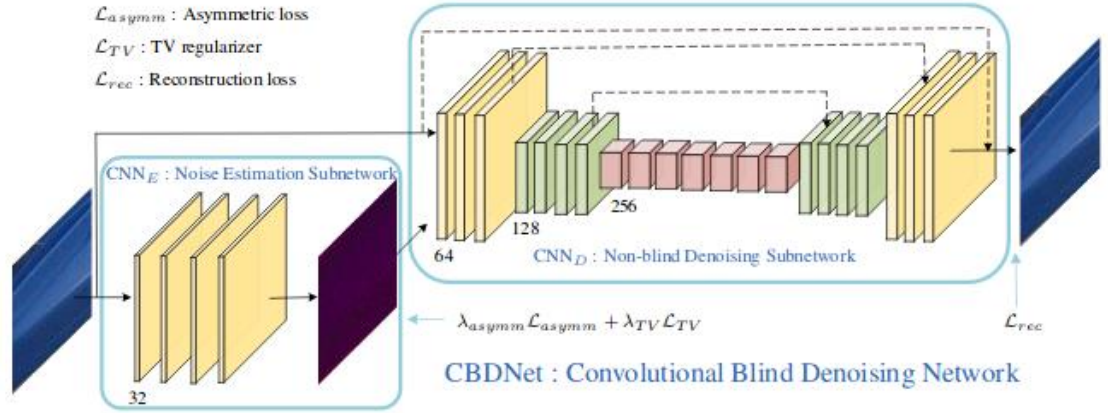


Figure 20: Illustration for CBDNet for blind denoising real-world images.

This CBDNet involves two stages according to [23], one is noise-estimation subnetwork and other a non-blind denoising subnetwork. Generally, CNN depends on the ability of memorizing the training data. Existing deep denoisers like DnCNN does not work well with the real-world noisy images as they assume the data to be AWGN while the real noise distribution is much different from that. But these models have high memorization ability to learn the real noise models. Therefore, noise models play a crucial role in the denoising performance. Noise model in CBDNet considers Gamma correction, demosaicing and JPEG compression in order to generate synthetic noisy image which is similar to real world noisy images.

This architecture includes noise estimation  $CNN_E$  and non-blind denoising subnetwork  $CNN_D$ . A noisy observation  $y$  is input for  $CNN_E$  which generates the noise level map with network parameters. This noise level map is of the same size as the input  $y$  estimated with a fully convolutional network.  $CNN_D$  takes both  $y$  and noise level input map as input to generate the denoising result.

Structure of  $CNN_E$

- 5-layer fully convolution network without pooling and batch normalization.
- Conv2D (32, (3,3), activate = ReLU)

Structure of  $CNN_D$

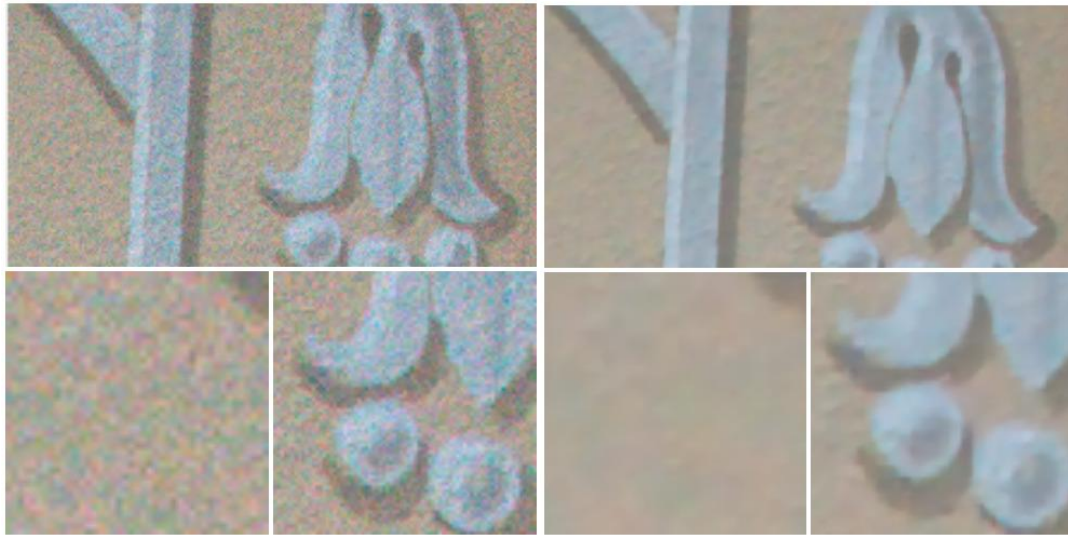
- 16-layer U-Net architecture (input:  $y$  and noise level input map)
- With symmetric skip connection, strided convolutions and transpose convolutions
- Filter size 3\*3 and ReLU nonlinearity is applied after every layer except the last one.

This network provides better results when compared to the DnCNN for real-world images due to the realistic noise models which includes Gaussian and ISP pipeline to make the model learn the synthetic images applicable to real-world noisy images. And also, its performance can be increased by including both synthetic and real images for training the network.

## Experiments

CBDNet is trained with real-world noisy image datasets like NC12, DND and Nam.

Observed results for Real-world images by applying CBDNet



*Figure 21: Denoising results of a real-world noisy image using CBDNet*

This CBDNet is tested on microscopic fluorescence images from FMD dataset, no comparable results are obtained for this kind of images.



*Figure 22: Denoising results of microscopic fluorescence images using CBDNet*



In my future work, I would like to study about the CBDNet and also modify the non-blind denoising network/noise-estimator in order to analyze the performance with microscopic fluorescence images.

# Bibliography

- [1] V.Jain and S.Seung, "Natural image denoising with convolutional networks," 2009.
- [2] H. Burger, C. Schuler and S.Harmeling, "Image: denoising: Can plain neural networks comete with BM3D?," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012.
- [3] J. Xie, L. Xu and E. Chen, "Image denoising and inpainting with deep neural networks," 2012.
- [4] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," in *IEEE Trans. Image Process.*, Dec. 2006.
- [5] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," in *IEEE Trans. Pattern Anal. Mach. Intell.*
- [6] J. Duchi, E. Hazan and Y. Singer, "Adaptive sub gradient methods for online learning and stochastic optimization," Feb. 2011.
- [7] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012.
- [8] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," 2015.
- [9] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," 2012.
- [10] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015.
- [11] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016.
- [12] D. Kiku, Y. Monno, M. Tanaka and M. Okutomi, "Residual interpolation for color image demosaicing," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2013.
- [13] R. Timofte, V. D. Smet and L. V. Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *Proc. Asian Conf. Comput. Vis.*, 2014.
- [14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015.
- [15] S. Gu and R. Timofte, "A brief review of image denoising algorithms and beyond," in *Computer Science*, 2019.
- [16] K. Zhang, W. Zuo, Y. Chen, D. Meng and a. L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," in *IEEE Transactions on Image Processing*, 2017.
- [17] Y. Zhang, Y. Zhu, E. Nichols, Q. Wang, S. Zhang, C. Smith and S. Howard, "A Poisson-Gaussian Denoising Dataset with Real Fluorescence Microscopy Images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, 2019.
- [18] Y. Tai, J. Yang, X. Liu and C. Xu, "Memnet: A persistent memory network for image restoration," in *In CVPR*, 2017.

- [19] S. Gu, L. Zhang, W. Zuo and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014.
- [20] C. S. a. Y.-B. Y. X. Mao, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connection," in *In Advances in neural information processing systems*, 2016.
- [21] M. Weigert, U. Schmidt, T. Boothe, A. M. üller, A. Dibrov, A. Jain, B. Wilhelm, D. Schmidt, C. Broaddus, S. Cul-ley, M. Rocha-Martins, F. Segovia-Miranda, C. Norden, R. Henriques, M. Zerial, M. Solimena, J. Rink, P. Tomancak and L. Royer, "Content-aware image restoration: Pushing the limits of fluorescence microscopy," in *Nature Methods*, 2018.
- [22] Buades, B. Coll and J.-M. Morel, "A non-local algorithm for image denoising," in *In CVPR*, 2005.
- [23] A. V. D. Oord, N. Kalchbrenner and K. Kavukcuoglu, "Pixel recurrent neural networks," in *In ICML*, 2016.
- [24] S. Laine, T. Karras, J. Lehtinen and T. Aila, "High-quality self-supervised deep image denoising," 2019.
- [25] S. Guo, Z. Yan, K. Zhang, W. Zuo and L. Zhang, "Toward convolutional blind denoising of real photographs," 2018.