



國立臺北科技大學

電子工程研究所

博士學位論文

從單張自然影像中去除高密度脈衝雜訊  
技術

Removing High-density Impulse Noises from a Single  
Natural Image

研 究 生：林柏雄

指導教授：黃士嘉博士

中華民國一百零九年十一月一日

# 摘要

論文名稱：從單張自然影像中去除高密度脈衝雜訊技術

頁數：60 頁

校所別：國立臺北科技大學電子工程研究所

畢業時間：一百零九學年度第一學期

學位：博士

研究生：林柏雄

指導教授：黃士嘉博士

關鍵詞：脈衝雜訊、形態學影像處理、深度學習

消除脈衝雜訊是影像處理系統中至關重要的步驟，近年來已經提出許多消除雜訊的方法來改善影像品質，然而消除高密度的脈衝雜訊仍然是主要挑戰之一。在本文中，為了解決高密度雜訊相關影像消除問題，我們提出了兩種新的消除雜訊方法，稱為形態均值濾波器 (morphological mean filter) 和輕型消除雜訊網絡 (LD-Net)。形態均值濾波器分為兩個階段執行，包括無雜訊像素計數器 (noise-free pixel counter : NPC) 模組和形態學像素擴散 (morphological pixel dilation : MPD) 模組，首先提出的方法是檢測影像中無雜訊像素的數量和位置，接下來以形態影像處理的方式將無雜訊像素為基礎進行迭代來向周圍擴張，並替換相鄰的雜訊像素直到收斂為止，通過這樣的步驟能夠去除高密度雜訊。LD-Net 分兩個階段執行，其中包括特徵增強 (feature augmentation : FA) 階段和特徵細化 (feature refinement : FR) 階段，在特徵增強階段使用反捲積層增加輸入影像的空間大小和尺寸進行有效的特徵學習，在特徵細化階段利用卷積層的序列增強圖像的紋理細節來重建無雜訊影像，實驗結果表明，與其他去雜訊方法相比，該方法能更有效地去除損壞影像中的高密度脈衝雜訊。

# Abstract

Title : Removing High-density Impulse Noises from a Single Natural Image

Pages : 60

School : National Taipei University of Technology

Department : Electronic Engineering

Time : December, 2020

Degree : Ph.D.

Researcher : Po-Hsiung Lin

Advisor : Shih-Chia Huang Ph.D.

Keywords : Impulse Noise, Morphological Image Processing, Deep Learning.

The removal of impulse noise is a crucial pre-processing step in image processing systems. In recent years, numerous noise-removal methods have been proposed to improve denoizing performance and reconstruct noise-free images. However, removing high-density impulse noise remains a major challenge. In this thesis, to address the image denoizing problem associated with high-density noise, we propose two new denoizing methods, called morphological mean filter(M-MF) and Lightweight Denoising network(LD-Net). M-MF is performed in two modules including a noise-free pixel counter (NPC) module and a morphological pixel dilation (MPD) module. First, the proposed method detects both the number and position of the noise-free pixels in the image. Next, the dilatation operation of the noise-free pixels based on morphological image processing is iteratively executed to replace the neighbor noise pixels until convergence. By doing so, the proposed method is capable to remove high-density noise and therefore reconstruct the noise-free image. LD-Net is performed in two stages including a feature augmentation stage(FA) and a feature refinement stage(FR). During the feature augmentation stage, the spatial size and dimension of the input image are increased by employing the deconvolutional layers for effective feature learning. During the feature refinement stage, the textural details of the image are enhanced for the reconstruction of the noise-free image by the utilization of a proposed sequence of three convolutional layers. Experimental results indicate that the proposed method more effectively removes high-density impulse noise in corrupted images in comparison with the other tested state-of-the-art methods.

# Acknowledgements

在北科大電子工程研究所的六年時光，即將到達一個人生的轉折，在路途中跌跌撞撞，碰到許多困難，但在老師及同學的幫助之下，逐一克服所遇到的問題。首先我要感謝我的指導教授黃士嘉博士的指導與教誨，感謝老師在這六年來對學生的付出，讓我學習到對於研究應有的態度，學生會謹記在心。誠摯感謝口試委員 XXX 教授、XXX 教授、XXX 教授、XXX 教授、XXX 教授和 XXX 教授，抽空參與學生的口試，並給予寶貴的建議及鼓勵，讓學生能順利通過口試，論文更加趨近完善。感謝這六年父母對我的栽培及鼓勵，也要謝謝研究室的學長們，適時給予我鼓勵及幫助，也因為你們讓我能堅持到最後，還有許多曾經幫助鼓勵過我的人，在此感謝你們大家。



# Contents

Abstract in Chinese . . . . .	i
Abstract in English . . . . .	ii
Acknowledgements . . . . .	iii
Contents . . . . .	iv
List of Figures . . . . .	vi
List of Tables . . . . .	viii
1 Introduction . . . . .	1
1.1 Thesis overview . . . . .	3
2 Background . . . . .	4
2.1 SM-based Filters . . . . .	5
2.1.1 AWM Filter . . . . .	5
2.1.2 MDBUTM Filter . . . . .	5
2.1.3 FEM Filter . . . . .	6
2.1.4 NAFSM Filter . . . . .	6
2.2 Learning-based Filter . . . . .	8
2.2.1 DnCNN . . . . .	8
3 Denoising Method for High-density Impulse Noises . . . . .	9
3.1 Proposed Morphological Mean Filter . . . . .	9
3.1.1 Noisefree Pixel Counter Module . . . . .	9
3.1.2 Morphological pixel dilation module . . . . .	10

3.2	Proposed Lightweight Denoising Network . . . . .	13
3.2.1	Feature augmentation stage . . . . .	14
3.2.2	Feature refinement stage . . . . .	16
3.2.3	Training method . . . . .	17
4	Evaluation . . . . .	18
4.1	Results and Discussion of The Morphological Mean Filter . . . . .	18
4.1.1	Experimental Setup . . . . .	18
4.1.2	Results . . . . .	18
4.1.3	Discussion . . . . .	21
4.2	Results and Discussion of The Lightweight Denoising Network . . . . .	21
4.2.1	Experimental Setup . . . . .	21
4.2.2	Results . . . . .	23
4.2.3	Discussion . . . . .	27
5	Conclusions . . . . .	30
	Bibliography . . . . .	31

# List of Figures

2.1	Type 1 and Type 2 windows used in the two-window based median filter.	7
3.1	The flowchart of the proposed filter. . . . .	10
3.2	Structuring element for the 8-neighborhood used in the proposed method.	11
3.3	Illustration of the image reconstructed by using the proposed method as iterations progress in which $K$ is the number of iterations after convergence: (a) is the input noisy image; (b) is the first reconstructed image; (c) is the second reconstructed image; (d) is the $K$ th reconstructed image.	11
3.4	Comparisons between the use of different values of the stopping factor $\rho$ : (a) is the input image with 90% noise density; (b) is the reconstructed image while $\rho = 0.1$ and (d) is the corresponding process of iteration involved in the pursuit of the primary objective; (c) is the reconstructed image while $\rho = 1e^{-5}$ and (e) is the corresponding process of iteration involved in the pursuit of the primary objective. . . . .	12
3.5	Overall architecture of the proposed LD-Net. . . . .	15
4.1	De-noising of the “Peppers” image. . . . .	19
4.2	De-noising of the “Aircraft” image. . . . .	19
4.3	De-noising of the “Candy” image. . . . .	20
4.4	De-noising of the “Clock” image. . . . .	20
4.5	De-noising of the “F16” image. . . . .	20
4.6	Example of “Lena” image with different high-density impulse noise levels.	22

4.7	Denoising results of LD-Net and other denoising methods on a "forest" image with 60% impulse noise corruptions. (a) Original image, (b) image corrupted by noise, (c) image reconstructed using AW-MF, (d) image reconstructed using FE-MF, (e) image reconstructed using NA-MF, (f) image reconstructed using MD-MF, (g) image reconstructed using DnCNN, and (h) image reconstructed using the proposed LD-Net. . . . .	23
4.8	Denoising results of LD-Net and other denoising methods on a "cameraman" image with 70% impulse noise corruptions. (a) Original image, (b) image corrupted by noise, (c) image reconstructed using AW-MF, (d) image reconstructed using FE-MF, (e) image reconstructed using NA-MF, (f) image reconstructed using MD-MF, (g) image reconstructed using DnCNN, and (h) image reconstructed using the proposed LD-Net. . . . .	24
4.9	Denoising results of LD-Net and other denoising methods on a "boat" image with 80% impulse noise corruptions. (a) Original image, (b) image corrupted by noise, (c) image reconstructed using AW-MF, (d) image reconstructed using FE-MF, (e) image reconstructed using NA-MF, (f) image reconstructed using MD-MF, (g) image reconstructed using DnCNN, and (h) image reconstructed using the proposed LD-Net. . . . .	25



## List of Tables

3.1	Setting of the proposed image augmentation module . . . . .	16
3.2	Setting of the proposed image refinement module . . . . .	17
4.1	The PSNR (dB) scores of our LD-Net model and the compared denoising models on 12 commonly used test images. . . . .	28
4.2	The Average PSNR (dB) results obtained by our LD-Net model and the compared denoising models on SN-LABELME test set. . . . .	29
4.3	The running time of the proposed LD-Net and competing models on SN-LABELME test set. . . . .	29



# Chapter 1 Introduction

During acquisition and transmission under adverse conditions, images are usually contaminated by high-density impulse noise. The presence of noise in the images not only reduces the visual quality but also negatively impacts the performance of many computer vision applications, including object tracking [1,2], object detection [3–5], and face image enhancement [6]. Therefore, the development of a model that can efficiently remove noise is essential. The noisy image production process of the additive noise model [7] can be described as follows:

$$I_x = J_x + n, \quad (1.1)$$

where  $I_x$  is the observed noisy image,  $J_x$  is true image, and  $n$  is noise in the image.

Various denoising approaches have been introduced, which can be divided into two main categories: filter-based methods [8–11] and learning-based methods [?, ?, 12, 13]. Filter-based denoising methods are regarded as conventional and are most popular for noise removal. The standard median (SM) filter [14] is the most basic non-linear filtering method based on the order statistic for impulse noise removal. The SM filter replaces each visited pixel by the median pixel of its neighbors within the filtering window. However, the SM filter is often inefficient if images is corrupted by high density noise (e.g., containing more than 50% noise). Therefore, several state-of-the-art SM-based methods were developed in recent years aimed at removing high-density impulse noise. Representative techniques such as Gaussian filters, mean filters, and median-type filters are often utilized because of their effectiveness and simple implementation. Zhang *et al.* [8] proposed an adaptive weighted mean filter (AW-MF) to cope with high-level impulse noise. First, the AWMF detects noise using an adaptive window size on each pixel, then removes noise candidates by replacing them with the weighted mean of the current window. A fast median filter (FE-MF) [9] has also been proposed, which captures natural pixels by utilizing prior information for image restoration. In this approach, based on the difference in the noise ratios of each image, the median is identified by employing two separate sets of masked pixels. In [10], the authors proposed a noise-adaptive fuzzy switching me-

dian filter (NA-MF) to reduce impulse noise. This method consists of two-stage designs, including a noise detection stage to identify noise pixels and a noise-filtering stage to remove the detected noise pixels. Esakkirajan *et al.* [11] introduced a modified decision algorithm that is based on an unsymmetrical trimmed median filter (MD-MF) to remove impulse noise for generating the noise-free image. Although these methods achieve high performance on images containing low-density impulse noise, they often fail to do so with images containing high-density impulse noise.

The approaches in the learning-based denoizing category have received extensive attention in recent years, including prior-learning-based methods, external/internal prior-learning-based hybrid methods, and especially deep-learning-based methods which can be mathematically defined as follows:

$$\hat{Y} = F(X; \Theta), \quad (1.2)$$

where  $X$  and  $\hat{Y}$  denote the input and output of the model, respectively, and  $F$  is a nonlinear operator function with trainable parameters  $\Theta$ . Given a dataset with  $i$  data pairs  $(X_i, Y_i)$ , we can train the model to fit ground-truth samples. DnCNNs [12] is a deep CNN-based denoising method, which modifies VGG network [13] for feature extraction of input noise images, and adopts residual learning and batch normalization for improving the training speed and noise removal performance.

Therefore, this thesis is interested in removing both SPN (salt and pepper noise) and RVIN (random-value impulsive noise) from the corrupted image. To this end, we propose a novel denoising method that addresses the challenges of SPN and RVIN suppression, respectively. The proposed method consists of a SPN suppression stage and a RVIN suppression stage and can be detailed as follows:

- In the SPN suppression stage, the proposed a novel morphological mean (MM) filter for high-density impulse noise removal. The MM filter is designed into two modules consists of two important modules: a noise-free pixel counter (NPC) module and a morphological pixel dilation (MPD) module. First, all pixels of the input image are examined in order to collect both the position and number of the noise-free pixels.

Next, the dilatation operation of these pixels is performed to fill into the neighbor noise pixels in the proposed MPD module for best recovery of the image corrupted by a high density of noise. By doing so, the proposed method is capable to remove high-density noise and therefore reconstruct the SPN-noise-free image.

- In the RVIN suppression stage, the proposed an efficient noise removal method, called LD-Net, which is a lightweight model based on a deep CNN for the effective removal of high-density impulse noise from corrupted images. To accomplish this objective, LD-Net is designed into two stages including the feature augmentation (FA) stage and feature refinement (FR) stage. Enlarging the dimensions and spatial sizes of the input features is performed in the FA stage to ensure effective feature learning, and then the textural details of the images are clarified during the FR stage to efficiently reconstruct noise-free images. LD-Net is trained end-to-end to directly produce reconstructed images and achieve fast processing speed.

The experimental results have showed that the developed image denoising method significantly outperforms existing state-of-the-art denoising methods in both SPN and RVIN removals for a large variety of scene types.

## **1.1 Thesis overview**

The rest of this thesis is structured as follows. Chapter 2 introduces the background, including AWM Filter, NAFSM Filter, and MDBUTM Filter. The proposed approach including two schemes: Morphological Mean Filter and LDNet is discussed in Chapter 3. The experiment results for the two schemes of the proposed approach over a large variety of scene types are reported and discussed in Chapter 4. We conclude this thesis in Chapter 5.

## Chapter 2 Background

The standard median (SM) filter [14] is the most basic non-linear filtering method based on the order statistic for impulse noise removal. The SM filter replaces each visited pixel by the median pixel of its neighbors within the filtering window. Several SM-based filters utilize the impulse noise detector as the pre-processing function to effectively detect both noise pixels and noise-free pixels for the input image corrupted by salt-and-pepper noise [15–18].

If the impulse noise level is above 50%, even though SM filters after processing, the result of the image isn't ideal. Under the circumstances, the image will need another smarter way to analyze the high-density noise. Recently, Deep Learning has been paying attention to its recovering ability with the corrupted image. With Deep Learning, first it can detect the useful features, and then it will keep those useful details through self-learning repeatedly. Particularly, Deep Learning is highly recommended for detecting noise. All in all, Deep Learning(sparse coding and CNN) effectively works on all different levels of high-density noise. Meanwhile, this method also allows to analyze salt-and-pepper noise and random value impulse noise [19–22].

When the visited pixel is determined by the impulse noise detector to be a noise-free pixel, the binary noise mask  $B$  is labeled as '0'. Otherwise, we label it as '1' to represent a noise pixel. This decision rule can be expressed as follow:

$$B(x) = \begin{cases} 1, & \text{if } I(x) \text{ is noise} \\ 0, & \text{otherwise} \end{cases}, \quad (2.1)$$

where  $I(x)$  is the visited pixel value at position  $x$  in the input image.

The purpose of these SM-based filters is to replace the noise pixel with a new pixel acquired from the noise-free pixels of its neighbors within the filtering window. Additionally, the detected noise-free pixel is preserved in the output noise-free image.

The purpose of Deep Learning is to extract features by training to expand a network and dictionary. The network and dictionary can be improved through multiple pieces of training in an iterative method. Deep learning adopts a trained network and dictionary to

reconstructed corrupted images.

The following is my introduction of 5 ways, including four SM-based filters and three Deep Learning. As well as describing how these operate to reach ideal results.

## 2.1 SM-based Filters

### 2.1.1 AWM Filter

The AW-MF also employs various window-sizes to find noise-free pixels in a corrupted image [8]. The size of the scanned window is continually adjusted until the noise-free pixel is found. As such, the noise-free image can be reconstructed by

$$O(x) = \begin{cases} \text{mean}(I(y)), & \text{if } B(x) = 1 \wedge s < 19 \\ \text{mean}(I(k)), & \text{if } B(x) = 1 \wedge s = 19 \\ I(x), & \text{if } B(x) = 0 \end{cases} \quad (2.2)$$

where  $s$  is the radius of a window that is iteratively increased,  $I(y)$  represents all the noise-free pixels indicated by index  $y$ , and  $B(x)$  can be attained by using Eq. (2.1).

### 2.1.2 MDBUTM Filter

The MD-MF possesses three filtering cases to restore noisy images [11]. In the first case, if the visited pixel is not a noise pixel (i.e.,  $B(x) = 0$  of Eq.(2.1)), it does not need to be processed. Additionally, if the visited pixel is a noise pixel (i.e.,  $B(x) = 1$ ), and has at least one noise-free pixel in the scanned window, the median value of all noise-free pixels is computed to be the output noise-free pixel for the second case. In the third case, when the visited pixel is a noise pixel and has no uncorrupted pixels around it within the scanned window, the output pixel value can be directly presented by the mean value of all

neighbor pixels. This decision rule can be expressed as follows:

$$O(x) = \begin{cases} I(x), & \text{if } B(x) = 0 \\ \text{median}(I(y)), & \text{if } B(x) = 1 \wedge N(x) > 0 \\ \text{mean}(I(k)), & \text{if } B(x) = 1 \wedge N(x) = 0 \end{cases} \quad (2.3)$$

where  $O(x)$  is the output noise-free image,  $N(x)$  is the number of noise-free pixels in the scanned window,  $I(y)$  are all the noise-free pixels located by index  $y$ , and  $I(k)$  are all the noise pixels located by index  $k$  in the window.

### 2.1.3 FEM Filter

Different from other noise filters, the FE-MF employs a two-window-based median filter to restore noisy images [9]. The FE-MF use two types of windows, that they call small and large windows. When the noise density is less than 50%, the method uses a "Type 1" filtering window; otherwise, it uses a "Type 2" filtering window, as shown in Fig. 2.1. As for low levels of noise, it is sufficient to examine the neighboring points along with both horizontal and vertical directions; High noise level moves around with the center of a pixel examining neighboring points along and gradually expanding outward. This decision rule can be expressed as follows:

$$O(x) = \begin{cases} I(t1), & \text{if } ND(x) < 50\% \\ I(t2), & \text{if } ND(x) \geq 50\% \end{cases}, \quad (2.4)$$

where  $O(x)$  is the output noise-free image,  $I(t1)$  is the number of noise-free pixels in the type 1 window,  $I(t2)$  is the number of noise-free pixels in the type 2 window, and  $DN(x)$  is the noise density in the image.

### 2.1.4 NAFSM Filter

The NA-MF considers that the noise-free pixels might be misjudged as noise pixels due to the use of the impulse noise detector since these noise-free pixels have a high

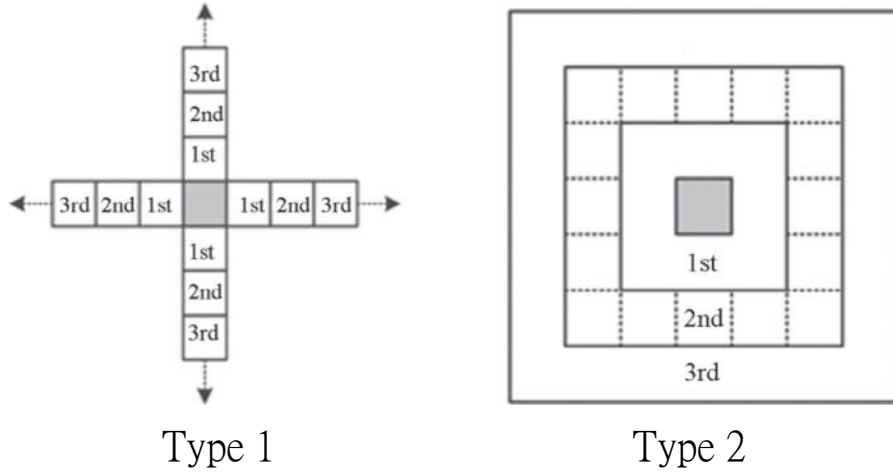


Figure 2.1: Type 1 and Type 2 windows used in the two-window based median filter.

probability of possessing similar values to noise pixels [10]. Therefore, the NAFSM filter employs the recursive double-stage methods to recover noisy images. The recursive double-stage methods employ median computation and fuzzy computation. For each visited noise pixel  $I(x)$  with s.t.  $B(x) = 1$ , the noise-free pixel can be first recovered by using median computation with various window sizes recursively as follows:

$$M(x) = \begin{cases} \text{median}(I(y)), & \text{if } B(y) = 0 \wedge s \leq 3 \\ \text{median}(I(e)), & \text{otherwise} \end{cases}, \quad (2.5)$$

where  $I(y)$  are all noise-free pixels indicated by the index  $y$  within a window,  $s = \{1, 2, 3\}$  and is the radius of a window that is iteratively increased. Otherwise, when there are no noise-free pixels within these three different window-sizes, the noise-free pixel is recovered by using the median computation of four pre-defined pixels  $I(e)$  indicated by the index  $e$ . The index  $e$  includes the top-left, left, bottom-left, and bottom position in the smallest window size.

Next, fuzzy computation is used to refine the result acquired by using the median computation in order to produce the noise-free image. The fuzzy computation  $F$  can be achieved via the fuzzy membership function as follows:

$$F(x) = \begin{cases} 0, & \text{if } D(x) < T_1 \\ \frac{D(x) - T_1}{T_2 - T_1}, & \text{if } T_1 \leq D(x) < T_2 \\ 1, & \text{otherwise} \end{cases}, \quad (2.6)$$



where  $D(x)$  is the maximum luminance difference between the visited pixel value  $I(x)$  and each 8-connected pixels.  $T_1$  and  $T_2$  are set to 10 and 30 for optimal performance according to [10]. Finally, the output noise-free image can be obtained by using the linear combination of the fuzzy computation  $F(x)$  and the median computation  $M(x)$  as follows:

$$O(x) = (1 - F(x))I(x) + F(x)M(x). \quad (2.7)$$

## 2.2 Learning-based Filter

### 2.2.1 DnCNN

DnCNNs [12] is a deep CNN-based denoising method, which modifies VGG network [13] for feature extraction of input noise images, and adopts residual learning and batch normalization for improving the training speed and noise removal performance. In [23], nonlocal self-similarity prior (NSP) was exploited for image denoising. From natural images, based on the NSP learning strategy, a patch group was proposed for learning obviously nonlocal self-similarity models to improve the denoizing performance. The authors in [24] proposed a denoizing method that exploits both internal and external data in which external priors are first learned from an independent set of clear images. Then, the learned external priors are used to guide the internal prior learning process from noisy images. Finally, the images are reconstructed using a set of orthogonal dictionaries produced from the external and internal priors. Despite having obtained promising image-reconstruction quality, most of the above learning-based denoizing approaches were designed based on multi-step learning processes, which yield complex models and involve high computation costs.

## Chapter 3 Denoising Method for High-density Impulse Noises

### 3.1 Proposed Morphological Mean Filter

In this section, we propose a novel morphological mean filter (M-MF) for high-density impulse noise removal. The flowchart of the proposed MM filter is shown in Fig. 3.1, and consists of two important modules: a noise-free pixel counter (NPC) module and a morphological pixel dilation (MPD) module.

In the proposed NPC module, all pixels of the input image are examined in order to collect both the position and number of the noise-free pixels. Next, the dilatation operation of these pixels is performed to fill into the neighbor noise pixels in the proposed MPD module for best recovery of the image corrupted by a high density of noise. We will describe these two proposed modules in the following subsections.

#### 3.1.1 Noise-free Pixel Counter Module

In contrast to the previous SM-based filters, the proposed NPC module aims to detect noise-free pixels and consequently restore the neighbor noise pixels for images with high densities of noise. First of all, the position of the noise-free pixels in the input corrupted image are detected by using Eq. (2.1); their number at the  $i$ th iteration can be obtained as follows:

$$N_i = P - \sum_{x=1}^P B_i(x), \quad (3.1)$$

where  $P$  is the total number of pixels in the input image  $I$  and  $B_i(x)$  is the  $i$ th binary noise mask at position  $x$ , which can be derived from the impulse noise detector. Note that  $i$  is the iterative generation. Algorithm 1 gives the detailed operation of the proposed NPC module.

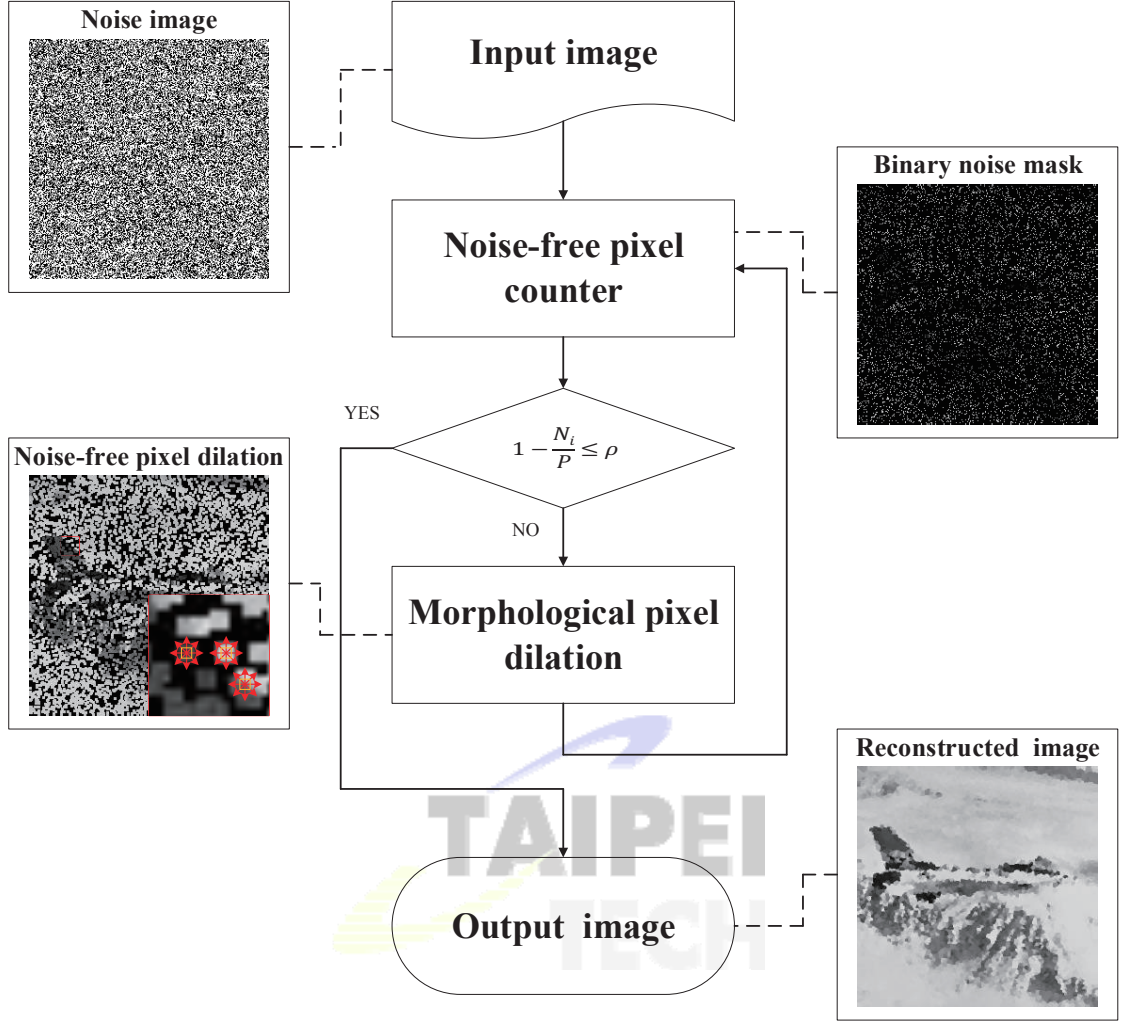


Figure 3.1: The flowchart of the proposed filter.

### 3.1.2 Morphological pixel dilation module

After the noise-free pixels of the corrupted image are detected in the NPC module, the detected pixels are then used to replace the eight-connected noise pixels within the scanned window. By doing so, the image corrupted by high densities of noise can be effectively restored. For each noise-free pixel, the  $i$ th reconstructed image  $O_i$  can be obtained by performing the dilatation operation of the noise-free pixels as follows:

$$O_i(e) = \begin{cases} I(y) \oplus N_8, & \text{if } B_i(y) = 0 \wedge \sum_{k=1}^{N_\Omega} B_i(k) \geq 1 \\ I(e), & \text{otherwise} \end{cases}, \quad (3.2)$$

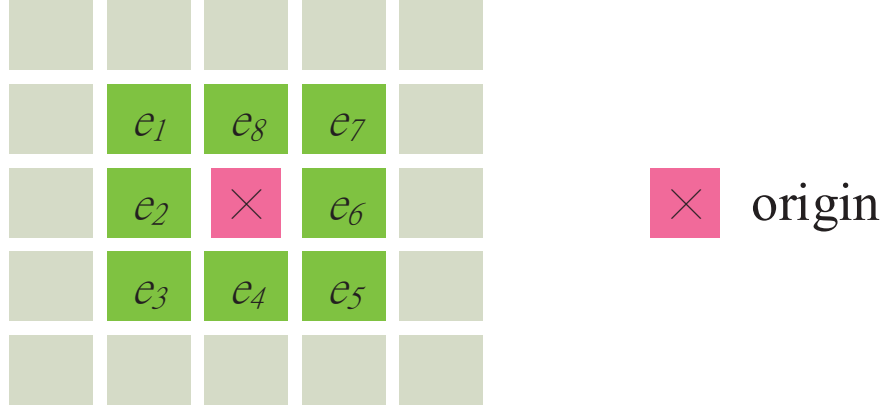


Figure 3.2: Structuring element for the 8-neighborhood used in the proposed method.

where  $\oplus$  is the dilation operation;  $N_8$  is the structuring element that allows us to define 8 neighborhood structure, as shown in Fig. 3.2;  $I(y)$  is the noise-free pixel (i.e.,  $B_i(y) =$

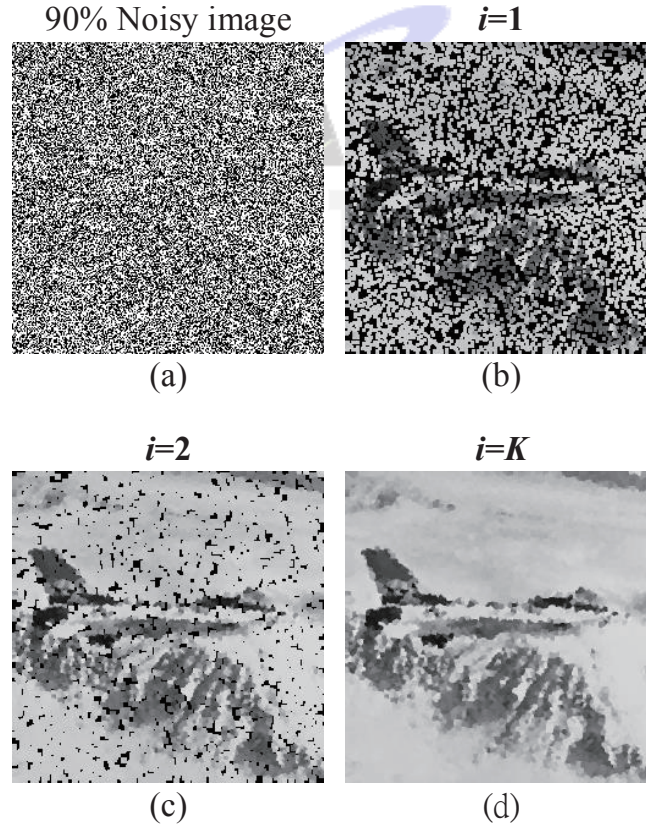


Figure 3.3: Illustration of the image reconstructed by using the proposed method as iterations progress in which  $K$  is the number of iterations after convergence: (a) is the input noisy image; (b) is the first reconstructed image; (c) is the second reconstructed image; (d) is the  $K$ th reconstructed image.

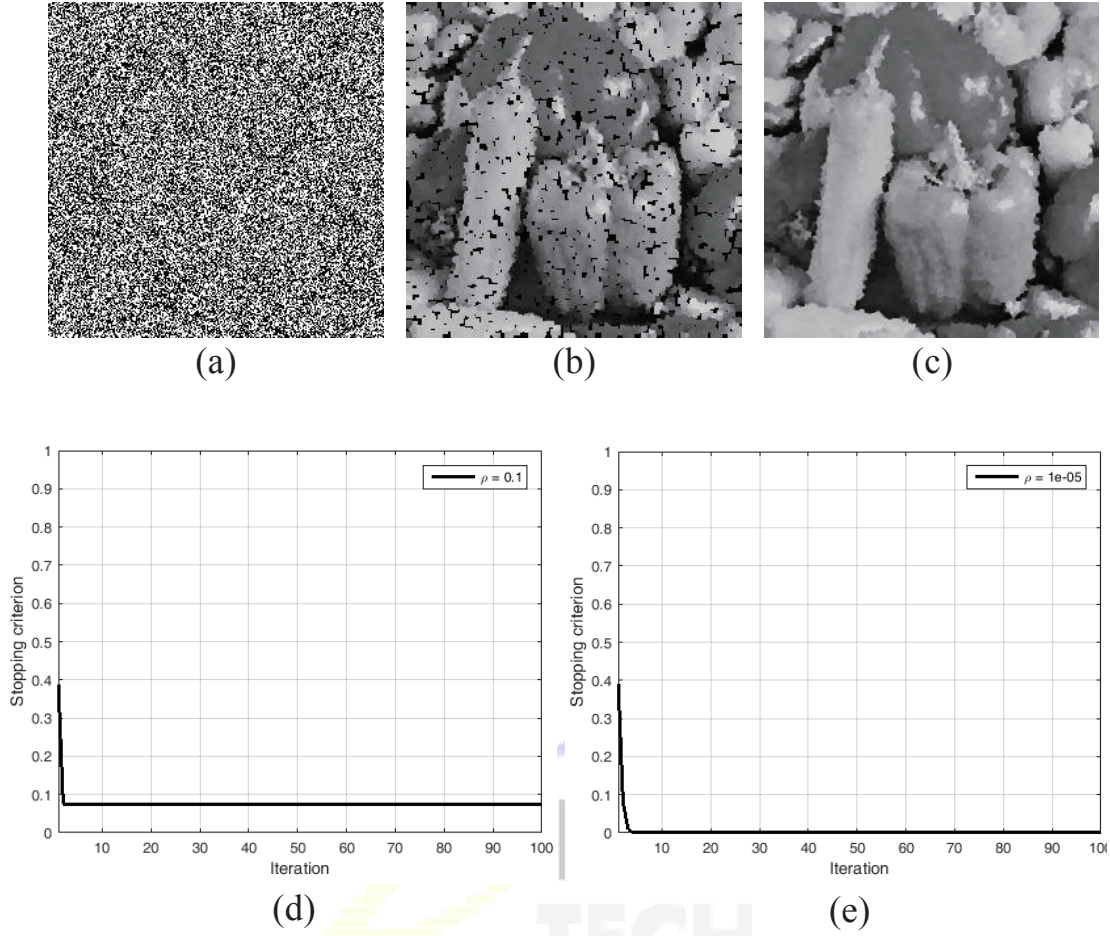


Figure 3.4: Comparisons between the use of different values of the stopping factor  $\rho$ : (a) is the input image with 90% noise density; (b) is the reconstructed image while  $\rho = 0.1$  and (d) is the corresponding process of iteration involved in the pursuit of the primary objective; (c) is the reconstructed image while  $\rho = 1e^{-5}$  and (e) is the corresponding process of iteration involved in the pursuit of the primary objective.

0), at position  $y$  in the input noise image;  $I(e)$  are eight connected pixels of the image indicated by index  $e$  including the top, top-left, left, bottom-left, bottom, bottom-right, right, and top-right positions within the window with radius  $s$ . Here, the radius  $s$  can be set as 1, meaning that the window size is  $3 \times 3$ .  $N_\Omega$  is the number of pixels within the window of radius  $s$ . Note that the additional weight map  $W$  is used to average each overlapped output value  $O_i$  to account for possible variations. The generation procedure of the weight map is described in Algorithm 2.

The processes of the proposed NPC and MPD modules are iterated until no further

---

**Algorithm 1** Noise-free pixel counter module

---

```
1: Input: Noise image  $I$ 
2: Output: Number of noise-free pixels  $N_i$ , Binary noise mask  $B_i$  Initialize:  $B_i \leftarrow 0$ 
3:    $N_i \leftarrow$  Number of pixels  $P$ 
4:    $i \leftarrow 1$ 
5: for each pixel  $x$  do
6:   if  $I(x)$  is noise pixels then
7:      $B_i \leftarrow 1$ 
8:   else
9:      $B_i \leftarrow 0$ 
10:  end if
11:   $N_i \leftarrow N_i - B_i(x)$ 
12: end for
```

---

noise pixel (i.e.,  $B_i(x) = 1$ ) is to be replaced, as shown in Fig. 3.3. The stopping criterion can be expressed as follows:

$$1 - \frac{N_i}{P} \leq \rho, \quad (3.3)$$

where  $N_i$  is the number of noise-free pixels in  $i$ th iteration and  $\rho$  is the stopping factor. Note that the different stopping factor  $\rho$  will influence the quality of output noise-free image and the processing speed for the image. A higher stopping factor  $\rho$  possesses faster convergence and results in a poorer quality of output noise-free images, as shown in Fig. 3.4. Based on this characteristic, the stopping factor  $\rho$  can be set as  $1^{-5}$  for better recovery quality.

## 3.2 Proposed Lightweight Denoising Network

In this section, we introduce our lightweight denoizing model Lightweight Denoising Network (LD-Net), which can achieve impressive performance and speed on high-density impulse noise images. Our model is very simple but effective and easy to implement, which is only composed of five convolution and deconvolutional layers followed by rectified linear units. The proposed model has two stages: a feature augmentation (FA) stage

---

**Algorithm 2** Morphological pixel dilation module

---

```
1: Input: Noise image  $I$ , The  $i$ th binary noise mask  $B_i$ 
2: Output: The  $i$ th reconstructed image  $O_i$ 
3: Initialize:  $N_\Omega \leftarrow$  Number of pixels in window of radius  $s$ 
4:   Weight map  $W_i \leftarrow 0$ 
5: for each pixel  $x$  do
6:   if  $B_i(x) = 0$  &  $\sum_{k=1}^{N_\Omega} B_i(k) \geq 1$  then
7:     for each pixel  $e$  in the window of radius  $s$  do
8:        $O_i(e) \leftarrow I(x)$ 
9:        $W_i(e) \leftarrow W_i(e) + 1$ 
10:    end for
11:   else
12:      $O_i(x) \leftarrow I(x)$ 
13:   end if
14: end for
15: for each pixel  $x$  do
16:   if  $W_i(x) > 0$  then
17:      $O_i(x) \leftarrow O_i(x) / W_i(x)$ 
18:   else
19:      $O_i(x) \leftarrow O_i(x)$ 
20:   end if
21: end for
```

---

and a feature refinement (FR) stage, as shown in Fig. 3.5. LD-Net can be learned in an end-to-end fashion, and directly generated noise-free images.

### 3.2.1 Feature augmentation stage

To effectively extract and learn features for noise removal, the spatial sizes and dimensions of the input image are first enlarged by the deconvolution layers in the FA stage. Unlike the bilinear upsampling technique, the deconvolution layer has a filter that is not fixed and can be learned during training [25]. This layer, which works as an upsampling



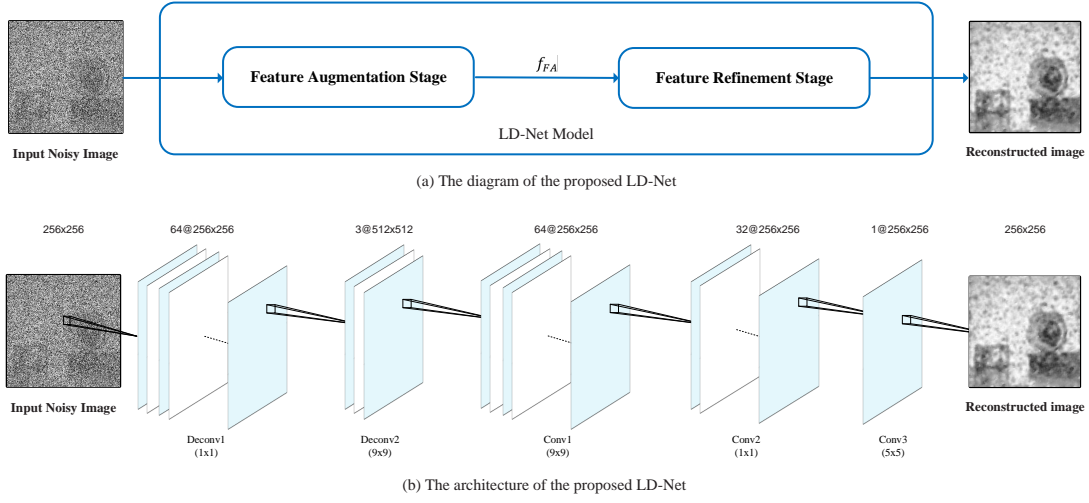


Figure 3.5: Overall architecture of the proposed LD-Net.

technique, was first used in the deconvolutional network [26] and has a wide range of applications, including image generation, semantic segmentation, and object detection. In [3], the deconvolution layers have been proven effective for human hand detection, which were used to match the resolutions of the generated feature maps in different layers. Noh *et al.* [27] employed the deconvolution layer to reverse the pooling operation and reconstruct the original size of the input image in a deconvolution network to improve the semantic segmentation performance. To visualize feature activation in the deep CNN network, deconvolution layers have also been used to produce a continuous path back to image pixels [28].

Inspired by these successful applications of deconvolution technique, we utilize two consecutive deconvolutional layers in the first stage of LD-Net. These two layers, named Deconv1 and Deconv2, have deconvolution filters  $1 \times 1$  and  $9 \times 9$  in size, respectively. Initially, the input image is fed into layer Deconv1, where 64 filters are used to produce 64 feature maps for the purpose of increasing the dimension. Then, these feature maps are sent to layer Deconv2 to enlarge their resolution by the utilization of three filters  $9 \times 9$  in size, with a stride of 2. The feature maps generated by layer Deconv2, which are  $512 \times 512$  in size, will be sent to the next layer in the next stage for image reconstruction, as shown in Fig. 3.5. The details of the proposed image augmentation module are listed in Table 3.1.



Table 3.1: Setting of the proposed image augmentation module

Layer	Activation size
Input	$256 \times 256 \times 1$
$1 \times 1 \times 1$ , deconvolutions, stride 1, pad 1, relu	$256 \times 256 \times 64$
$9 \times 9 \times 64$ , deconvolutions, stride 2, pad 1, relu	$512 \times 512 \times 3$

### 3.2.2 Feature refinement stage

During the feature refinement stage, the boundary artifacts are reduced and textural information is enhanced for the efficient reconstruction of a noise-free image. In most low-level vision applications such as haze removal, rain removal, snow removal, and noise removal, the size of the output image must remain the same as that of the input image. However, during the pooling of the CNN, the input size is often changed and must be recovered, which can produce boundary artifacts that can degrade the visual quality [12]. The method presented in [29] adopts a strategy of symmetrically padding the boundaries of input noisy images in the pre-processing step to reduce boundary artifacts. This same padding scheme is implemented before every stage in [30,31]. Unlike the abovementioned approaches, the size of the feature maps of the intermediate layers in DnCNN [12] are the same as that of the input image by directly padding them with zeros prior to convolution, which effectively prevents boundary artifacts and improves denoizing performance.

Motivated by the success reported by the authors of [12], we employ three consecutive convolution layers in the FR stage, including  $9 \times 9$  convolution,  $1 \times 1$  convolution, and  $5 \times 5$  convolution, named Conv1, Conv2, and Conv3, respectively. First, the output feature maps of the FA stage ( $f_{FA}$ ) are sent to layer Conv1, in which padding with zeros, 64 filters, and stride-2 are used to generate 64 feature maps and match the resolution of  $f_{FA}$  with that of the input image. Then, the resulting features are passed through layer Conv2, which utilizes 32 filters with size of  $1 \times 1$  to extract useful features for image reconstruction. Finally, the output of the LD-Net is computed and generated through layer Conv3, which uses one  $5 \times 5$  filter for gray images and three  $5 \times 5$  filters for color images.

Table 3.2: Setting of the proposed image refinement module

Layer	Activation size
Input	$512 \times 512 \times 3$
$9 \times 9 \times 3$ , Convolutions, stride 2, pad 1, relu	$256 \times 256 \times 64$
$1 \times 1 \times 64$ , Convolutions, stride 1, pad 1, relu	$256 \times 256 \times 32$
$5 \times 5 \times 32$ , Convolutions, stride 1, pad 1, linear	$256 \times 256 \times 1$

The detailed configurations of the proposed image refinement module are summarized in Table 3.2.

### 3.2.3 Training method

In the additive noise model presented in (1.1), noise-free images are produced by learning a mapping function  $F(I_x) = J_x$  [30]. To remove high-density impulse noise, we train LD-Net in an end-to-end fashion and adopt the mean square error as a loss function of our model, which is expressed as follows:

$$L(\Theta) = \frac{1}{N} \sum_{i=1}^N (\mathbf{Y}_i - F(X_i; \Theta))^2, \quad (3.4)$$

where  $X_i$  and  $Y_i$  are a noisy input image and a ground truth image, respectively,  $F(., \Theta)$  is a mapping function with trainable parameters  $\Theta$ , and  $N$  is the batch size.

## Chapter 4 Evaluation

This chapter demonstrates the improvement of our proposed two denoizing methods through quantitative and qualitative evaluations, respectively.

### 4.1 Results and Discussion of The Morphological Mean Filter

#### 4.1.1 Experimental Setup

In our experiments, we select five  $256 \times 256$  standard grayscale images entitled “Peppers”, “Aircraft”, “Clock”, “F16”, and “Candy” for testing the NA-MF, MD-MF, AW-MF and our MM methods. Each test image was corrupted by 90% noise density and restoration was conducted with MATLAB 2013b running on a desktop machine with a Windows 8.1 operating system and an Intel Core i7 Quad central processing unit at 2.40 GHz and 16 GB memory.

Additionally, we use the peak signal-to-noise ratio (PSNR) for quantitative measurement of output images generated by each compared method in order to verify the effectiveness of each. The PSNR computation can be formulated as follows:

$$\text{PSNR} = 10 \log_{10} \left( \frac{255^2}{\text{MSE}} \right), \quad (4.1)$$

where

$$\text{MSE} = \frac{1}{P} \sum_{x=0}^{P-1} (I(x) - O(x))^2. \quad (4.2)$$

#### 4.1.2 Results

Figures 4.1-4.5 indicate the qualitative and quantitative measurements plus the computational cost (CC) reports of our simulation results performed by each compared method.

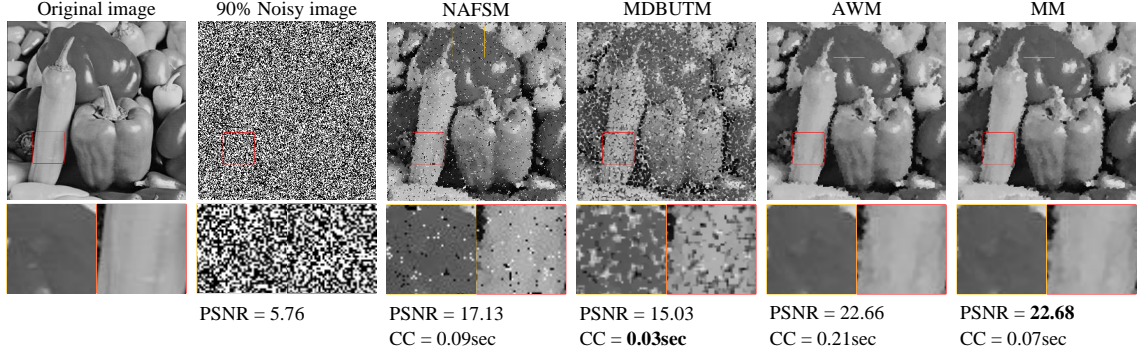


Figure 4.1: De-noising of the “Peppers” image.

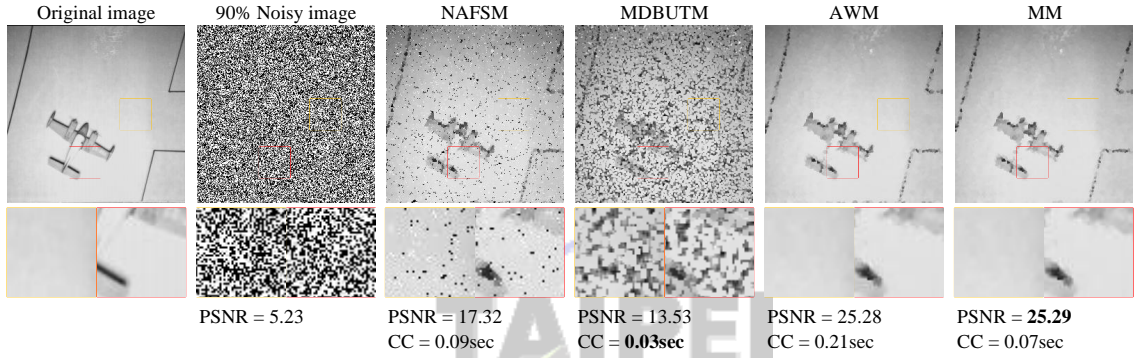


Figure 4.2: De-noising of the “Aircraft” image.

These noise-free images reconstructed by using the proposed MM method generate superior improvements in terms of visual perception in comparison with the NA-MF, MD-MF, and AW-MF methods, as demonstrated in Figs. 4.1-4.5 and their corresponding cropped windows. As can be seen, the proposed MM method provides the best PSNR values between each compared method via quantitative measurement. Specifically, because the image is corrupted with high-density noise, the training patterns cannot be produced sufficiently for each cropped window. For this reason, the images reconstructed by these two methods still contain serious noise.

Moreover, the NA-MF, MD-MF, and AW-MF methods adopt the fixed starting point along with the fixed scanned order (from the top-left of an image to the bottom-right of an image) to search the noise-free pixels around the noise pixel within a window. While high-density noises are concentrated in the scanned window, the NA-MF, MD-MF, and AW-MF methods fail to find the noise-free pixel within the window. This results in the reconstructed images still suffering from noise effects.

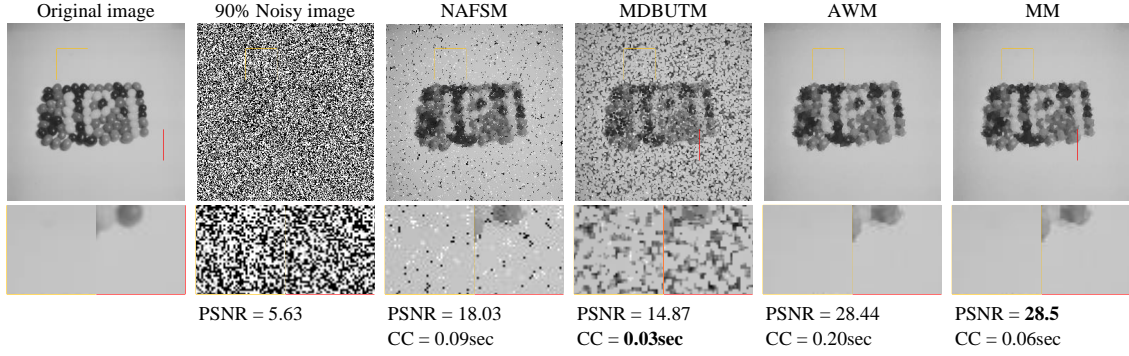


Figure 4.3: De-noising of the “Candy” image.

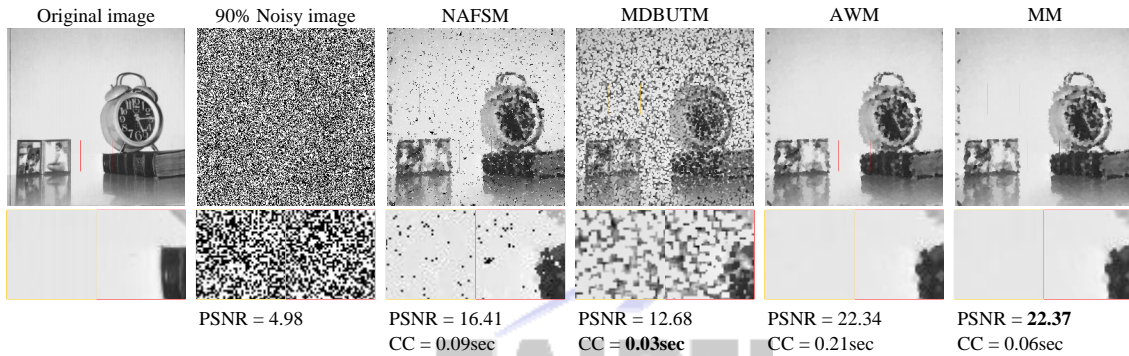


Figure 4.4: De-noising of the “Clock” image.

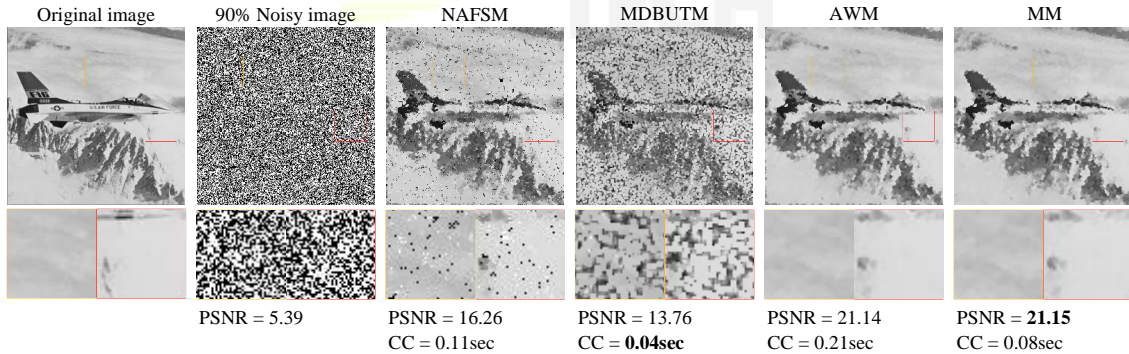


Figure 4.5: De-noising of the “F16” image.

In contrast to the NA-MF, MD-MF, and AW-MF methods, the starting point of the proposed MM filter is set on each noise-free pixel. Hence, the proposed MM filter is able to perform the morphological dilatation operation on these noise-free pixels iteratively to replace noise pixels. Therefore, the proposed MM method is capable of recovering a corrupted image with high-density noise and reconstruct a high-quality, noise-free image.



### 4.1.3 Discussion

In contrast to previous SM-based filters, to examine the noise pixels, the position and number of noise-free pixels are first filtered out by using the proposed NPC module. Subsequently, the proposed MPD module employs morphological dilatation processing for the noise-free pixels to replace the neighbor noise pixels. These two proposed modules are used iteratively until the stopping criterion is achieved. By doing so, the proposed MM filter is capable of dealing with images corrupted with high-density impulse noise. As a result, a superior noise-free image is reconstructed that achieves higher qualitative and quantitative measurement scores in comparison with the other evaluated state-of-the-art SM-based filters. The experimental results demonstrate that the proposed MM filter can more effectively removed high-density impulse noises in corrupted images in contrast to previous SM-based filters. Additionally, the proposed MM filter consumes only moderate execution time while producing the highest quality restored images and possessing the highest PSNR values.

## 4.2 Results and Discussion of The Lightweight Denoising Network

### 4.2.1 Experimental Setup

In this section, the experimental results of our proposed method in the removal of high-density impulse noise are summarized. To conduct experiments, we compose two data sets named SN-INRIA and SN-LABELME for training and testing, respectively. Quantitative and qualitative results of our model and state-of-the-art denoising models are comprehensively compared on the SN-LABELME test set. The compared models include the approaches presented by Zhang *et al.* [8] (denoted as AW-MF), Hsieh *et al.* [9] (denoted as FE-MF), Toh *et al.* [10] (denoted as NA-MF), Esakkirajan *et al.* [11] (denoted as MD-MF), and Zhang *et al.* [12] (denoted as DnCNN).

To evaluate the proposed method and compare its results with those of other noise

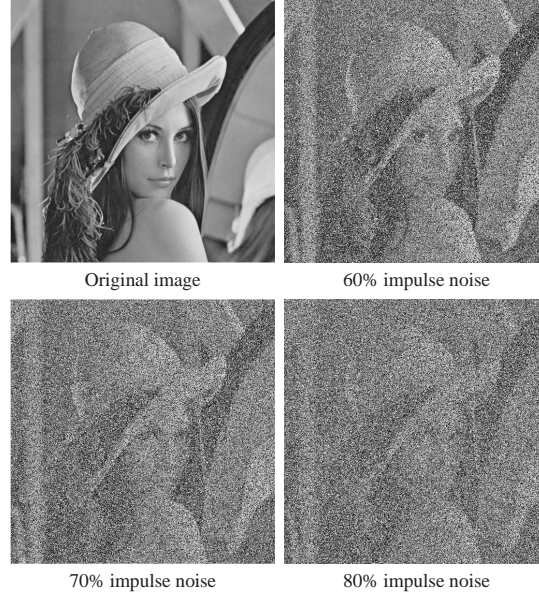


Figure 4.6: Example of "Lena" image with different high-density impulse noise levels.

removal methods, we synthesized a noisy image dataset consisting of a training set, named SN-INRIA, and a test set, named SN-LABELM. For the training models, SN-INRIA set is generated by adding impulse noise to all the images of the INRIA Holidays dataset [32]. In total, SN-INRIA comprises 159,390 gray noisy images  $256 \times 256$  in size ranging from 1% to 99% impulse noise in increments of 1%. For the testing models, we corrupted 1,629 images in the LabelMe dataset [33] by adding 60%, 70%, and 80% impulse noise to produce the SN-LABELME set with high-density impulse noise, as shown in Fig. 4.6. In total, SN-LABELME contains 4,887 noisy images of size  $256 \times 256$  in grayscale for testing.

The proposed method was implemented in the TensorFlow environment on a PC equipped with an Intel i7 2.60G Hz CPU, 16G RAM, and an Nvidia GeForce GTX 960M graphics card. The LD-Net was trained using the loss function defined in Equation (3.4) and the Adam optimizer [34] was employed to identify the optimal network parameters. During training, we set a learning rate of  $10^{-3}$  for all 200 epochs, with an epsilon value of  $10^{-8}$ , and a batch size of 8.

The compared methods were trained on the same machine by keeping unchanged the original works (i.e running environment, architecture of model, loss function, and so on).

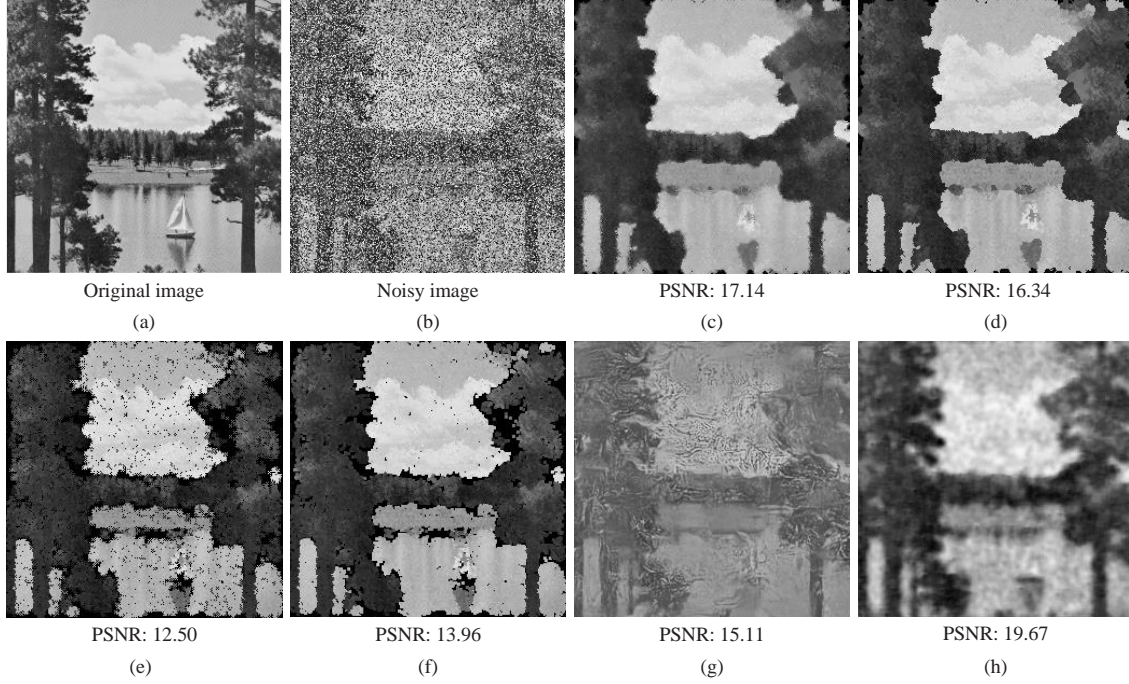


Figure 4.7: Denoising results of LD-Net and other denoising methods on a "forest" image with 60% impulse noise corruptions. (a) Original image, (b) image corrupted by noise, (c) image reconstructed using AW-MF, (d) image reconstructed using FE-MF, (e) image reconstructed using NA-MF, (f) image reconstructed using MD-MF, (g) image reconstructed using DnCNN, and (h) image reconstructed using the proposed LD-Net.

## 4.2.2 Results

The restoration quality of the proposed and state-of-the-art models was measured based on their peak signal-to-noise ratios (PSNRs), whereby the denoising model with the higher PSNR value indicates that with better visual quality of the reconstructed image.

Table 4.1 lists the PSNR values for 12 commonly used test images, and Table 4.2 shows the average PSNR values for all the images in SN-LABELME test set, in which the best PSNR results obtained by the compared models are in bold. As can be observed in Table 4.1, the proposed LD-Net obtained highest PSNR values on all the commonly used test images with high-density impulse noise levels. The second column of Table 4.2 presents the PSNR scores of compared methods on SN-LABELME test set with 60% impulse noise corruptions. We can see that LD-Net obtained 21.95dB, which is higher than those obtained by AW-MF, FE-MF, NA-MF, MD-MF, and DnCNN by 1.39dB, 2.29dB,



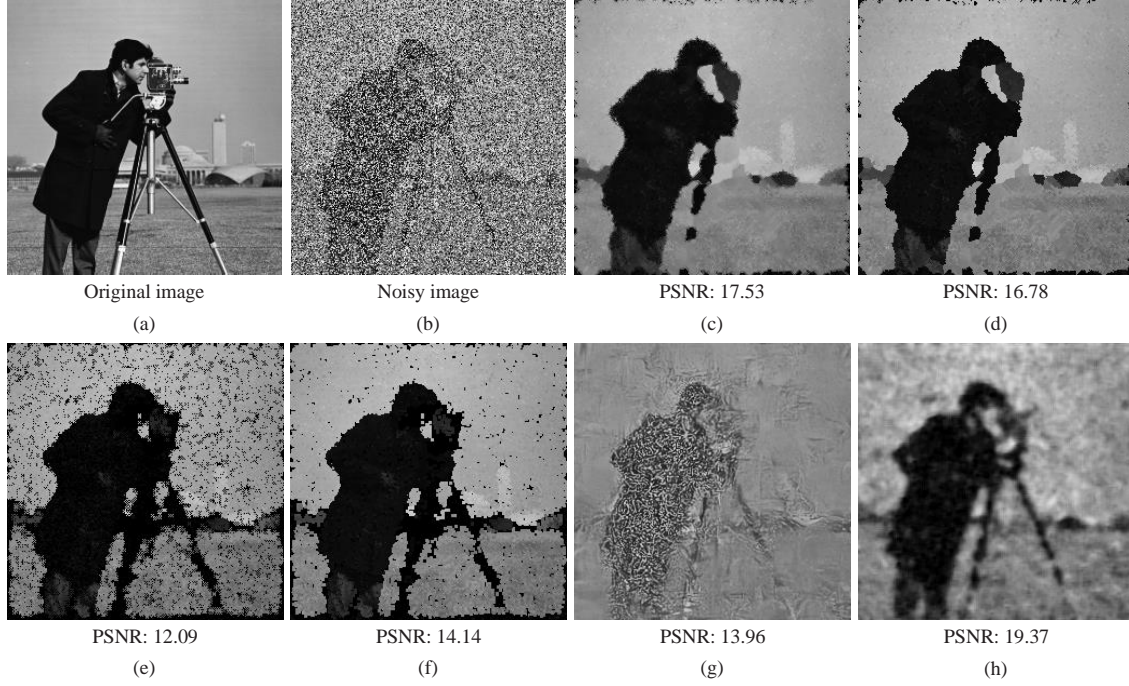


Figure 4.8: Denoising results of LD-Net and other denoising methods on a "cameraman" image with 70% impulse noise corruptions. (a) Original image, (b) image corrupted by noise, (c) image reconstructed using AW-MF, (d) image reconstructed using FE-MF, (e) image reconstructed using NA-MF, (f) image reconstructed using MD-MF, (g) image reconstructed using DnCNN, and (h) image reconstructed using the proposed LD-Net.

7.60dB, 5.36dB, and 5.93dB, respectively. The third column of Table 4.2 lists the PSNR scores of compared methods on SN-LABELME test set with 70% impulse noise corruptions. LD-Net obtained 20.83dB, which is higher than those obtained by AW-MF, FE-MF, NA-MF, MD-MF, and DnCNN by 1.27dB, 2.33dB, 8.75dB, 6.57dB, and 5.94, respectively. The fourth column of Table 4.2 shows the PSNR scores of compared methods on SN-LABELME test set with 80% impulse noise corruptions. LD-Net obtained 18.48dB, which is higher than those obtained by AW-MF, FE-MF, NA-MF, MD-MF, and DnCNN by 0.62dB, 1.63dB, 8.93dB, 7.29dB, and 4.55dB, respectively.

We visually show the reconstructed results of the proposed LD-Net and the compared methods for different images corrupted at three levels of high-density impulse noise including 60% corruption, 70% corruption, and 80% corruption. In addition, the corresponding PSNR values of the reconstructed results are also included.

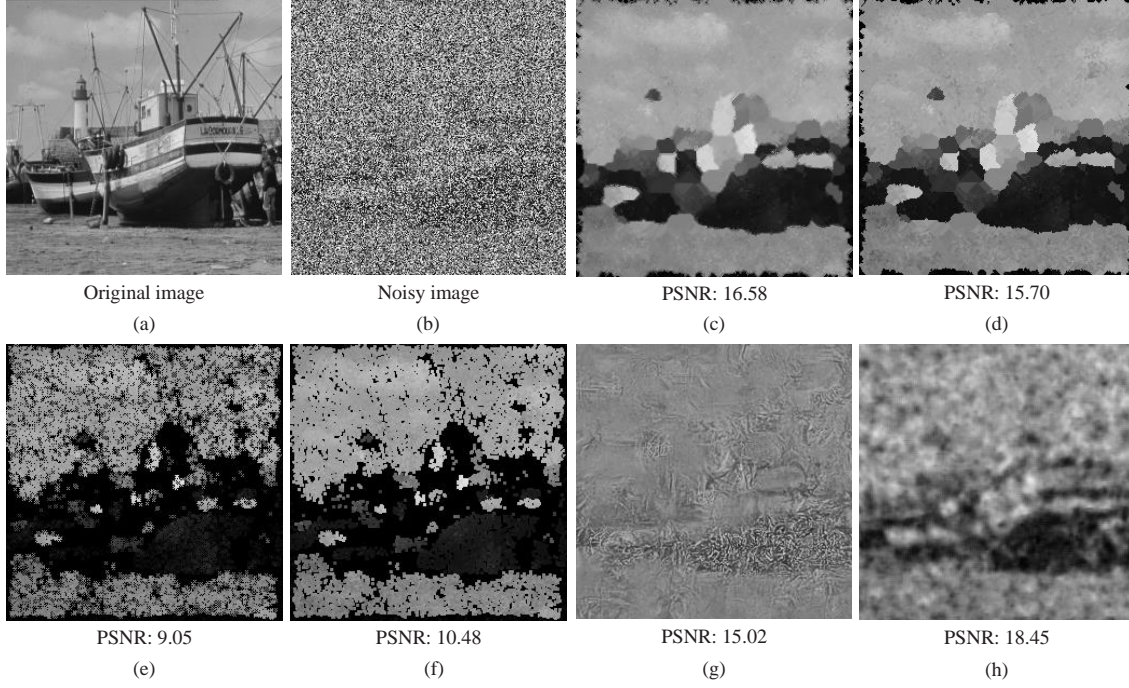


Figure 4.9: Denoising results of LD-Net and other denoising methods on a "boat" image with 80% impulse noise corruptions. (a) Original image, (b) image corrupted by noise, (c) image reconstructed using AW-MF, (d) image reconstructed using FE-MF, (e) image reconstructed using NA-MF, (f) image reconstructed using MD-MF, (g) image reconstructed using DnCNN, and (h) image reconstructed using the proposed LD-Net.

Fig. 4.7 shows the visual results of the proposed LD-Net and the compared models on a "forest" image with 60% impulse noise corruptions. As can be seen in Figs. 4.7(c)-4.7(g), the fine textures of each region of the denoising results obtained by the AW-MF, FE-MF, and DnCNN methods are considerably blurred, and a darkening effect on some pixels in the reconstructed images produced by the NA-MF and MD-MF methods are evident and would lead to visually unsatisfactory results. In contrast, the proposed LD-Net effectively recovered the texture and yielded a natural-looking reconstructed image, as shown in Fig. 4.7(h).

Fig. 4.8 shows the visual results of the proposed LD-Net and the compared models on a "cameramen" image with 70% impulse-noise corruption. As can be observed in Figs. 4.8(c)-4.8(g), the AW-MF, FE-MF, NA-MF and MD-MF methods tended to darken the pixels and present more false edges, and DnCNN generated many visual artifacts, all of which result in unnatural-looking reconstructed images. In contrast, LD-Net yielded

clearer results than those of the other methods, as shown in Fig. 4.8(h)

Fig. 4.9 shows the visual results of the proposed LD-Net and the compared models on a "boat" image with 80% impulse-noise corruption. As can be seen in the Figs 4.9(c)-4.9(g), all the competing methods achieve only a slight recovery and generated many blocking and darkened or blur artifacts, which resulted in the details of the boat in the reconstructed image being unrecognizable. The denoized result of the proposed LD-Net method, however, generated fewer artifacts and higher PSNR values than those of the compared methods, as can be observed in Fig. 4.9(h). Furthermore, the content of the original image is still easily recognizable in the reconstructed image, which demonstrates that LD-Net removes high-density impulse noise more effectively than other methods.

The superior results achieved by the proposed model are attributable to the fact that LD-Net has been optimized by adopting an end-to-end pipeline approach to noisy input images to remove noise and reconstruct images rather than estimating intermediate parameters using filters and detectors, which can intensify errors during the denoizing process. Additionally, boundary artifacts reduction is also achieved in the FR stage of the proposed model, which helps to effectively reconstruct images.

The task of noise removal is often performed as a pre-processing step in many real-world image processing systems, so the application of these systems requires denoizing at real-time speed. The proposed LD-Net model is lightweight and can achieve very fast denoizing. To compare the running time of LD-Net with those of competing models, we used all the images in the SN-LABELME test set, which were run on the same machine. Table 4.3 shows the average inference speeds of all the compared models, in which it can be seen that our model processed images of size  $256 \times 256$  in 24 ms using a single GTX 960M GPU. This result is faster than those achieved by the AW-MF [8], FE-MF [9], NA-MF [10], MD-MF [11], and DnCNN [12] models by 157ms, 77ms, 78ms, 72ms, and 37ms, respectively.

### 4.2.3 Discussion

LD-Net is trained using an end-to-end scheme and achieves real-time noise removal in two stages: 1) feature augmentation stage and 2) feature refinement stage. In the first stage, two deconvolution layers are employed to expand both the spatial size and dimensions of the image for further feature extraction. In the second stage, three convolution layers with padding zeros and different filter sizes are adopted to prevent the introduction of boundary artifacts and to enhance textural information for removing impulse noise and reconstructing a noise-free image. Our experimental results on the SN-LABELME test set demonstrated that the proposed LD-Net can provide satisfactory denoizing results. The qualitative and quantitative estimation of results obtained by our proposed model and competing models indicate that LD-Net achieves superior impulse noise removal and the highest speed.



Table 4.1: The PSNR (dB) scores of our LD-Net model and the compared denoising models on 12 commonly used test images.

Image	Boat	C.man	Clock	F16	Pepper	Lena	House	Lake	Man	Pine	Port	Tank
Noise Level	60%											
AW-MF	19.46	18.47	19.01	18.19	18.51	19.37	19.31	17.14	20.20	16.86	19.99	24.56
FE-MF	18.64	17.70	18.10	17.20	17.66	18.68	18.55	16.34	19.18	16.01	19.22	23.14
NA-MF	13.69	14.08	12.16	11.84	13.35	14.71	14.42	12.50	14.44	11.78	14.48	16.85
MD-MF	15.64	15.74	14.66	14.17	14.88	16.25	16.34	13.96	16.34	13.20	16.04	20.51
DnCNN	16.91	14.99	12.94	14.66	16.54	15.74	14.52	15.11	17.31	14.84	16.72	21.96
<b>LD-Net</b>	<b>21.69</b>	<b>20.39</b>	<b>21.75</b>	<b>20.52</b>	<b>22.16</b>	<b>21.98</b>	<b>20.41</b>	<b>19.67</b>	<b>22.55</b>	<b>20.30</b>	<b>21.45</b>	<b>24.66</b>
Noise Level	70%											
AW-MF	18.71	17.53	17.94	17.30	17.42	18.72	19.16	16.46	19.01	15.74	19.09	23.04
FE-MF	17.79	16.78	16.63	16.39	16.65	17.95	18.21	15.73	17.97	14.89	18.29	21.27
NA-MF	11.52	12.09	9.76	9.49	11.32	12.85	12.33	10.74	11.95	9.98	12.48	13.84
MD-MF	13.46	14.14	12.32	11.57	12.89	14.42	14.33	12.34	13.67	11.45	13.91	17.21
DnCNN	15.89	13.96	11.82	13.56	15.30	14.52	13.34	13.94	16.14	13.68	15.52	21.04
<b>LD-Net</b>	<b>20.48</b>	<b>19.37</b>	<b>20.75</b>	<b>19.51</b>	<b>20.92</b>	<b>20.87</b>	<b>19.82</b>	<b>18.70</b>	<b>21.12</b>	<b>19.03</b>	<b>20.54</b>	<b>23.07</b>
Noise Level	80%											
AW-MF	16.58	16.41	15.69	15.30	15.41	17.23	17.28	15.30	17.30	14.31	17.53	19.75
FE-MF	15.70	15.47	14.33	14.30	14.64	16.46	16.86	14.64	16.49	13.48	16.87	18.02
NA-MF	9.05	9.72	6.99	7.14	9.11	10.50	9.81	8.58	9.52	7.88	10.34	10.29
MD-MF	10.48	11.49	9.16	9.00	10.35	11.81	11.27	9.85	10.71	9.00	11.26	12.58
DnCNN	15.02	13.00	10.89	12.38	14.48	13.59	12.40	13.04	15.17	12.74	14.61	19.80
<b>LD-Net</b>	<b>18.45</b>	<b>17.61</b>	<b>16.95</b>	<b>17.05</b>	<b>18.45</b>	<b>18.85</b>	<b>17.36</b>	<b>16.86</b>	<b>19.11</b>	<b>16.97</b>	<b>18.53</b>	<b>20.37</b>

Table 4.2: The Average PSNR (dB) results obtained by our LD-Net model and the compared denoising models on SN-LABELME test set.

Method	Noise Level		
	60%	70%	80%
AW-MF	20.56	19.56	17.86
FE-MF	19.66	18.50	16.85
NA-MF	14.35	12.08	9.55
MD-MF	16.59	14.26	11.19
DnCNN	16.02	14.89	13.93
<b>LD-Net</b>	<b>21.95</b>	<b>20.83</b>	<b>18.48</b>

Table 4.3: The running time of the proposed LD-Net and competing models on SN-LABELME test set.

Method	Speed (ms)	Platform
AW-MF	181	Matlab
FE-MF	101	Matlab
NA-MF	102	Matlab
MD-MF	96	Matlab
DnCNN	61	Matlab
<b>LD-Net</b>	<b>24</b>	<b>PyTensorflow</b>

## Chapter 5 Conclusions

This thesis proposes a novel denoising method to restore image corrupted by high-density impulse noise, as we have surveyed in previous chapters. To this end, the proposed method can be divided into two method: a SPN suppression method and a RVIN suppression method.

In the SPN suppression method, we proposed a novel morphological mean filter to effectively remove high-density impulse noise and thereby improve image quality. In contrast to previous SM-based filters, to examine the noise pixels, the position and number of noise-free pixels are first filtered out by using the proposed NPC module. Subsequently, the proposed MPD module employs morphological dilatation processing for the noise-free pixels to replace the neighbor noise pixels. These two proposed modules are used iteratively until the stopping criterion is achieved. By doing so, the proposed MM filter is capable of dealing with images corrupted with high-density impulse noise.

In the RVIN suppression method, the lightweight noise removal model LD-Net is proposed for realizing fast and effective image denoizing in the presence of high-density impulse noise. LD-Net is trained using an end-to-end scheme and achieves real-time noise removal in two stages: 1) feature augmentation stage and 2) feature refinement stage. In the first stage, two deconvolution layers are employed to expand both the spatial size and dimensions of the image for further feature extraction. In the second stage, three convolution layers with padding zeros and different filter sizes are adopted to prevent the introduction of boundary artifacts and to enhance textural information for removing impulse noise and reconstructing a noise-free image.

Experimental results demonstrate that the advantages of the proposed denoising method over existing image denoising methods with a significant margin in both SPN and RVIN impulse noise suppression.



# Bibliography

- [1] W. Min, M. Fan, X. Guo, and Q. Han, "A new approach to track multiple vehicles with the combination of robust detection and two classifiers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, pp. 174–186, Jan 2018.
- [2] S.-C. Huang, "An advanced motion detection algorithm with video quality analysis for video surveillance systems," *IEEE transactions on circuits and systems for video technology*, vol. 21, pp. 1–14, Jan 2011.
- [3] T.-H. Le, S.-C. Huang, and D.-W. Jaw, "Cross-resolution feature fusion for fast hand detection in intelligent homecare systems," *IEEE Sensors Journal*, vol. 19, pp. 4696–4704, June 2019.
- [4] S. Huang, T. Le, and D. Jaw, "Dsnet: Joint semantic learning for object detection in inclement weather conditions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, March 2020.
- [5] T.-H. Le, D.-W. Jaw, I.-C. Lin, H.-B. Liu, and S.-C. Huang, "An efficient hand detection method based on convolutional neural network," *2018 7th International Symposium on Next Generation Electronics (ISNE)*, pp. 1–2, May 2018.
- [6] S. Anwar, F. Porikli, and C. P. Huynh, "Category-specific object image denoising," *IEEE Transactions on Image Processing*, vol. 26, pp. 5506–5518, July 2017.
- [7] F. Wang, H. Huang, and J. Liu, "Variational-based mixed noise removal with cnn deep learning regularization," *IEEE Transactions on Image Processing*, vol. 29, pp. 1246–1258, September 2019.
- [8] P. Zhang and F. Li, "A new adaptive weighted mean filter for removing salt-and-pepper noise," *IEEE signal processing letters*, vol. 21, pp. 1280–1283, June 2014.
- [9] M.-H. Hsieh, F.-C. Cheng, M.-C. Shie, and S.-J. Ruan, "Fast and efficient median filter for removing 1-99% levels of salt-and-pepper noise in images," *Engineering Applications of Artificial Intelligence*, vol. 26, pp. 1333–1338, April 2013.
- [10] K. K. V. Toh and N. A. M. Isa, "Noise adaptive fuzzy switching median filter for salt-and-pepper noise reduction," *IEEE signal processing letters*, vol. 17, pp. 281–284, March 2009.
- [11] S. Esakkirajan, T. Veerakumar, A. N. Subramanyam, and C. PremChand, "Removal of high density salt and pepper noise through modified decision based unsymmetric trimmed median filter," *IEEE signal processing letters*, vol. 18, pp. 287–290, May 2011.
- [12] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, pp. 3142–3155, July 2017.



- [13] J. Xu, L. Zhang, W. Zuo, D. Zhang, and X. Feng, "Patch group based nonlocal self-similarity prior learning for image denoising," *Proceedings of the IEEE international conference on computer vision*, pp. 244–252, 2015.
- [14] T. Nodes and N. Gallagher, "Median filters: Some modifications and their properties," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 30, pp. 739–746, October 1982.
- [15] Chung-Bin Wu, Bin-Da Liu, and Jar-Ferr Yang, "A fuzzy-based impulse noise detection and cancellation for real-time processing in video receivers," *IEEE Transactions on Instrumentation and Measurement*, vol. 52, pp. 780–784, June 2003.
- [16] S. Schulte, M. Nachtegaal, V. De Witte, D. Van der Weken, and E. E. Kerre, "A fuzzy impulse noise detection and reduction method," *IEEE Transactions on Image Processing*, vol. 15, pp. 1153–1162, May 2006.
- [17] Pei-Eng Ng and Kai-Kuang Ma, "A switching median filter with boundary discriminative noise detection for extremely corrupted images," *IEEE Transactions on Image Processing*, vol. 15, pp. 1506–1516, June 2006.
- [18] F. Duan and Y. Zhang, "A highly effective impulse noise detection algorithm for switching median filters," *IEEE Signal Processing Letters*, vol. 17, pp. 647–650, July 2010.
- [19] Yiqiu Dong, Raymond H. Chan, and Shufang Xu, "A detection statistic for random-valued impulse noise," *IEEE Transactions on Image Processing*, vol. 16, pp. 1112–1120, March 2007.
- [20] P. Civicioglu, "Removal of random-valued impulsive noise from corrupted images," *IEEE Transactions on Consumer Electronics*, vol. 55, pp. 2097–2104, November 2009.
- [21] Amarjit Roy, Lalit Manam, and Rabul Hussain Laskar, "Region adaptive fuzzy filter: An approach for removal of random-valued impulse noise," *IEEE Transactions on Industrial Electronics*, vol. 65, pp. 7268–7278, January 2018.
- [22] Y.Y. Zhou and Z.F. Ye and J.J. Huang, "Improved decision-based detail-preserving variational method for removal of random-valued impulse noise," *IET Image Processing*, vol. 6, pp. 976–985, October 2012.
- [23] J. Xu, L. Zhang, and D. Zhang, "External prior guided internal prior learning for real-world noisy image denoising," *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 2996–3010, 2018.
- [24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, vol. Y. Bengio and Y. LeCun, Eds, no. Available: <http://arxiv.org/abs/1409.1556>, 2015.
- [25] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

- [26] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, “Deconvolutional networks,” *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition*, vol. CVPR 2010, San Francisco, CA, USA, 13-18 June 2010, no. Available: <https://doi.org/10.1109/CVPR.2010.5539957>, 2010.
- [27] H. Noh, S. Hong, and B. Han, “Learning deconvolution network for semantic segmentation,” *Proceedings of the IEEE international conference on computer vision*, pp. 1520–1528, Dec 2015.
- [28] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” *European conference on computer vision*, pp. 818–833, 2014.
- [29] H. C. Burger, C. J. Schuler, and S. Harmeling, “Image denoising: Can plain neural networks compete with bm3d?,” *2012 IEEE conference on computer vision and pattern recognition*, pp. 2392–2399, 2012.
- [30] U. Schmidt and S. Roth, “Shrinkage fields for effective image restoration,” *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2774–2781, June 2014.
- [31] Y. Chen and T. Pock, “Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, pp. 1256–1272, June 2016.
- [32] H. Jegou, M. Douze, and C. Schmid, “Hamming embedding and weak geometric consistency for large scale image search,” *European conference on computer vision*, pp. 304–317, 2008.
- [33] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, “Labelme: a database and web-based tool for image annotation,” *International journal of computer vision*, vol. 77, no. 1-3, pp. 157–173, 2008.
- [34] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.