A Study on Impulse Noise Reduction Using CNN Learned by Divided Images

Yasushi Amari[a*], Takashi Miyazaki[a], Yusuke Koshimura[a], Yasuki Yokoyama[a], Hiroaki Yamamoto[b]

[a]National Insitute of Technology, Nagano College, 716 Tokuma, Nagano city, Nagano Prefecture and 381-8550, Japan
[b]Shinshu University, 4-17-1 Wakasato, Nagano city, Nagano Prefecture and 380-0928, Japan

*Yasushi Amari: 17902@g.nagano-nct.ac.jp
Takashi Miyazaki: miya@nagano-nct.ac.jp

Abstract

Random noise, which is one of impulse noise, and fixed pattern noise are known to be generated during imaging in a CMOS image sensor. In recent years, fixed pattern noise has been decreased due to the improvement of a CMOS image sensor performance. However, the random noise generated by photon fluctuation in the process of photon detection using photodiode still remains as a problem. So far, many denoising methods have been proposed to remove the random noise in images. In addition, we have already proposed one of denoising methods. However, our method requires an approximate threshold to obtain superior image quality. In this paper, we propose a method to effectively remove the noise superimposed on digital images using Deep Learning, which attracts attention in the field of image recognition and is applied in various fields. In addition, we report results that we compared our method with other conventional denoising methods.

Keywords: Deep Learning, Image denosing, Impulse noise, Convolution, Deconvolution.

1. Introduction

Various noises, such as fixed pattern noise and random noise, may be superimposed on the image obtained by a CMOS image sensor[(1)]. Pixels with different densities are generated in the image when the image is affected by noises. This phenomenon causes a problem of lowering image quality. Fixed pattern noise appears at a spatially fixed position and is caused by a pixel defect of the light receiving portion, dark current of the photodiode, and a variation of the sensitivity of the transistor in the process of manufacturing. Random noise is caused by dark current shot noise, 1/f noise, and photon shot noise, and this noise appears at random positions of output image. In recent years, fixed pattern noise tends to decrease due to the improvement of image sensor[(2)]. However, random noise, which is randomly generated in each pixel portion due to the photon fluctuation detected by the photodiode, is still a problem at present. Recently, the amount of light income per unit area received by the light receiving part tends to decrease in a CMOS image sensor which is required to have a high resolution. Thereby the influence of random noise at low illuminance increases.

A number of researches on denosing random noise are widely performed, and one of effective methods is Median Filter (MF)[(3)]. This method is a process of replacing the pixel of interest by obtaining the median density value from the pixel of the peripheral region. However, since this process is applied to all pixels, non-noise pixels are also subject to processing, which leads to degradation of the image. In addition, extra execution time is generated accordingly. For these problems, one of denosing methods called Switching Median Filter (SMF) has been proposed[(4-8)]. In this method, by determined whether the target pixel is noise or not, only noise pixels are successfully denoised. At the same time, the complication of its process, however, has a problem that the processing speed becomes slow. Therefore, we have already proposed Multi-directional Switching Median Filter (Multi-directional SMF), which denoises with the two-step method of multi-directional scanning and average processing using 2×2 noise detection operator[(9)].

A switching type method, such as SMF and Multi-directional SMF, needs a threshold to judge whether the target pixel is noise or not. Since the optimal threshold value differs from each image, however, it is difficult to find each

DOI: 10.12792/iciae2018.020

optimal threshold value. Thus, research on this decision method is still underway(10). In this paper, we propose a novel denoising method using Deep Learning, which has been applied in various fields in recent years and can extra feature value automatically, instead of the method using the threshold.

2. Image Denoising with Deep Learning

2.1 Proposed Method

Fig. 1 shows the overview of our proposed method. The feature of this method is that split images with superimposed noise are adopted for input images to be learned. The advantages of this method are as follows; it can be adapted to arbitrary image size, it is possible to increase the number of learning images, and it has the high efficiency of denoising. This method consists of two Convolution-layers and two Deconvolution-layers. In addition, it is mean square error as loss function.

2.2 Convolution-layer

First, convolution in an image will be described(11). The subject of this examination is grayscale image, and its gray value is stored in each pixel. The image size is $N \times N$ pixels, and the pixels are represented by index $(i,j)(i = 0,\dots,N-1, j = 0,\dots,N-1)$. The value of pixel$(i,j)$ is x_{ij}, and a real value including a negative value is taken. Learning images are filtered by using the $H \times H$ size filter. The pixel of filter is represented by index$(p,q)(p = 0,\dots,\mathrm{H}-1, q = 0,\dots,\mathrm{H}-1)$, and its value is represented by h_{pq}. h_{pq} takes an arbitrary real value. Convolution of an image, which is a product-sum calculation defined between an image and a filter is given as:

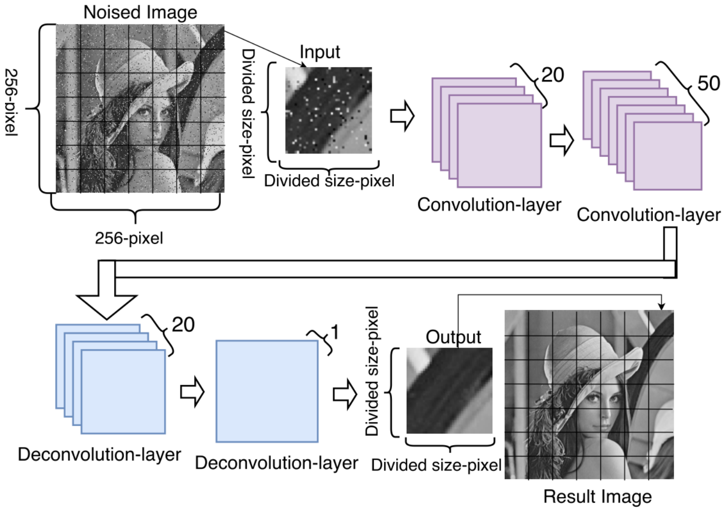


Fig. 1. Overview of Our Proposed Method.

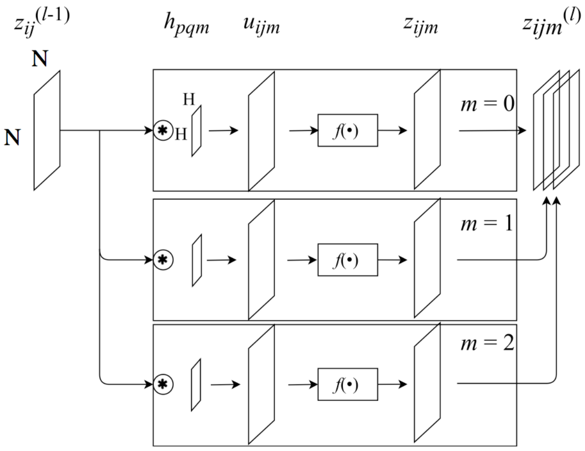


Fig. 2. Overview of Convolution-layer(11).

$$u_{ij} = \sum_{p=0}^{H-1}\sum_{q=0}^{H-1} x_{i+p,j+q}h_{pq} \tag{1}$$

The convolution-layer is a single layer network which performs the convolution operation described above. Fig.2 shows the overview of convolution-layer and the detail in the process of convoluting several filters to grayscale image. The convolution-layer of Fig. 2 receives a grayscale image $z_{ij}{}^{(l-1)}$ from the $l-1$ layer and applies $h_{pqm}(m = 0,\dots,M-1)$, $M(=3)$ kind filter, to it. The size of each filter is $H \times H$. u_{ijm}, which is executed in parallel for each filter m of $m = 0,1,2$. This calculation is expressed as:

$$u_{ijm} = \sum_{p=0}^{H-1}\sum_{q=0}^{H-1} x_{i+p,j+q}{}^{(l-1)}h_{pqm} + b_{ijm} \tag{2}$$

where b_{ijm} is added as a bias. This bias is set so that $b_{ijm} = b_m$ does not depend on the position of the pixel. We obtain z_{ijm} by applying the activation function of the following equation (3) to obtained u_{ijm}.

$$z_{ijm} = f(u_{ijm}) \tag{3}$$

The activation function is a nonlinear function or an identity function after linear transformation in a neural network. Since the ReLU function is often used for the activation function, ReLU function was also used in this paper. This ReLU function is expressed by the following as:

$$ReLU(x) = \max(0,x) \tag{4}$$

M pieces of z_{ijm} are outputted by M kind of filters, when all of them are matched, it becomes the output $z_{ijm}{}^{(l)}$ of the convolution-layer. The features of the convolution-layer are as follows; the data can be compressed while considering the relationship of the data in region, and the filter necessary for feature extraction can be learned automatically.

2.3 Deconvolution-layer

The Deconvolution-layer has the role of upsampling the input feature map[12]. The processing of the filter calculation itself is the same as that of the convolution-layer, and it is the processing of enlarging the image while maintaining dense information in the whole image after feature extraction by the encoder.

2.4 Loss Function

The loss function is a function that returns a loss with the predicted value y and correct answer $\bar{y}$ as arguments[13]. It is used in final layer of neural network, and the value of the loss function decreases when the difference between the predicted value and the correct value decreases. As loss function, the mean square error is used. The mean square error is given as:

$$L = \frac{1}{D}\sum_{n=1}^{D}\sum_{i}({y_i}^{[n]} - {\bar{y}_i}^{[n]})^2 \qquad (5)$$

where D represents learning samples, and ${y_i}^{[n]}$ represents the i-th dimension of the predicted value for n-th learning sample.

In this paper, the correct answer data is the original image and the learning data is the original image with noise added. Learning is performed so that the output image as the predicted value approaches the original image as the correct value. In order to minimize the loss, the gradient method is used for Deep Learning.

3. Experimental Method

3.1 Experiment Enviroment

Table 1 shows the experimental environment.

3.2 The framework of Deep Learning used

In order to implement the proposed method, we used Chainer, which is one of the frameworks of Deep Learning[14]. Chainer is a library that learns neural networks with Backpropagation, developed by Preferred Networks Inc. The version is 1.18.0. In addition, Chainer has following features.

- Offered as Python programing language library
- Flexible correspondence to the structure of every neural network
- Intuitive code by dynamic calculation graph construction
- Support for GPU, learning using multiple GPUs can also be described intuitively

Table 1. Experiment Enviroment.

CPU	Intel(R) Core(TM) i7-6700
GPU	GTX1070 (GIGABYTE)
Memory	32GB
OS	Ubuntu 16.04

3.3 Model of Noise

It explains random noise used in this paper. Random noise is defined as:

$$V = 0.5(U_{max} - U_{min}) \qquad (6)$$

The noise model is expressed as:

$$x(i,j) = \begin{cases} x_o(i,j) & : q = 1 - p \\ RND(U_{max}, -V) & : q = p/2 \\ RND(U_{min}, V) & : q = p/2 \end{cases} \qquad (7)$$

where $x(i,j)$ is the degradation signal, and $x_o(i,j)$ is the signal of the original image.

At this time, $RND(a,b)$ is a uniform number that takes the value of the interval $(a, a+b)$, q is the probability that the value is selected, and p is ratio of the noise.

3.4 Learning Method

For learning, 5,200 images of 256×256 pixels and 8-bit grayscale randomly extracted from ImageNet (large-scale image dataset) are used. Learning is performed with five kinds of image division sizes as follows: $8\times8, 16\times16, 32\times32, 64\times64$ and 128×128 pixels. The noise added to image during learning is random noise, which is five kinds of noise density as follows:10%, 20%, 30%, 40% and 50%. To optimize the network parameters, the Adam solver is adopted[15]. The step size is started from 0.001. The mini-batch size is set to 100. The number of learning is set at 300 times. Two points are evaluated: the division size of image suitable for image denoising, and image quality of output image when the noise density is changed as image data used for learning.

In addition, comparison is made with three kinds of conventional denoising methods, MF (median filter), SMF (switching median filter) and multi-directional SMF, which is developed in our laboratory.

3.5 Evaluation Method

PSNR (Peak Signal to Noise Ratio) is used as an index for quantitatively evaluating image quality in denoising experiments[16]. PSNR is expressed as:

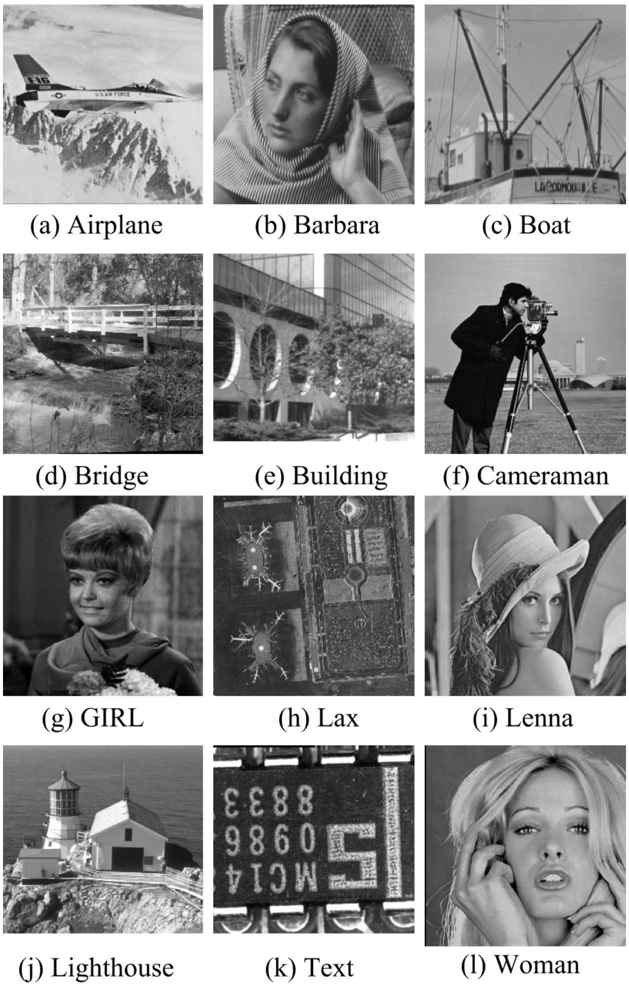
$$\mathrm{PSNR[dB]} = 10\log_{10}\frac{255^2}{\mathrm{MSE}} \qquad (8)$$

where MSE is mean square error between the original image and the restored image. M and N are the sizes of image, x_{ij} is the density of the original image and y_{ij} is the density of the restored images at (i, j).
MSE is expressed as:

$$\mathrm{MSE} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(x_{ij} - y_{ij})^2 \qquad (9)$$

Since the quality of image becomes better in inverse proportion to the loss between the original image and the restored image, the larger the value of PSNR is, the better the image quality becomes.

In this paper, twelve kinds of images of SIDBA (Standard Image Data-BAse) are used as test images. Fig. 3 shows the test images. Random noises are added to test images, and the density values of these noises are as follows: 10%, 20%, 30%, 40% and 50%. Next, test images are denoised using model, which is learned by the each division size of image and the added noise density. After these processes, the PSNR of each result image is measured and the average of PSNR is calculated.


(a) Airplane (b) Barbara (c) Boat

(d) Bridge (e) Building (f) Cameraman

(g) GIRL (h) Lax (i) Lenna

(j) Lighthouse (k) Text (l) Woman

Fig. 3. Test images.

4. Experimental Results

4.1 Result of Proposed Method

Fig. 4 shows the result of denoising images with random noise by each division size, with the above mentioned density values of 10%, 20%, 30%, 40% and 50%. Image denoising was performed by using the result of learning images with the noise added at the same rate of each random value noises.

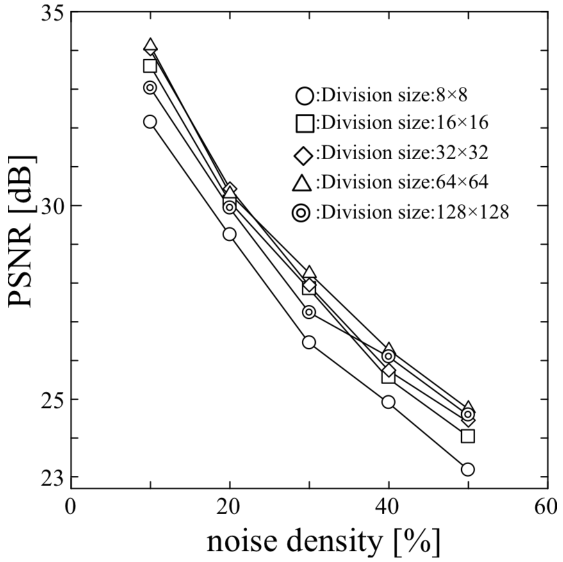


Fig. 4. Average of PSNR of proposed denoising method for each division size.

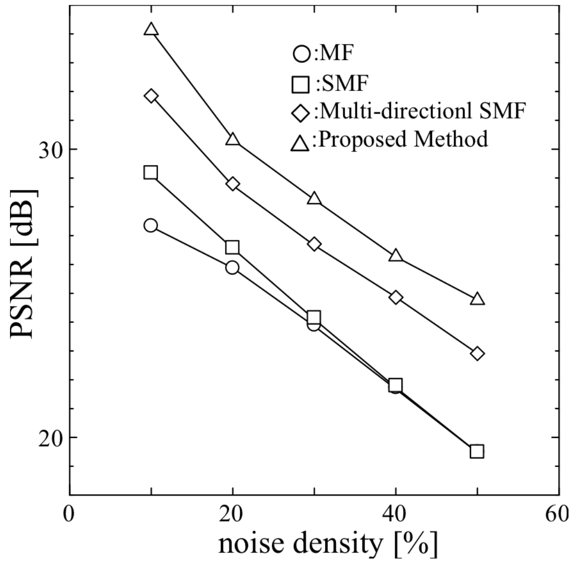


Fig. 5. Comparison of denoising performance with proposed method and conventional method.

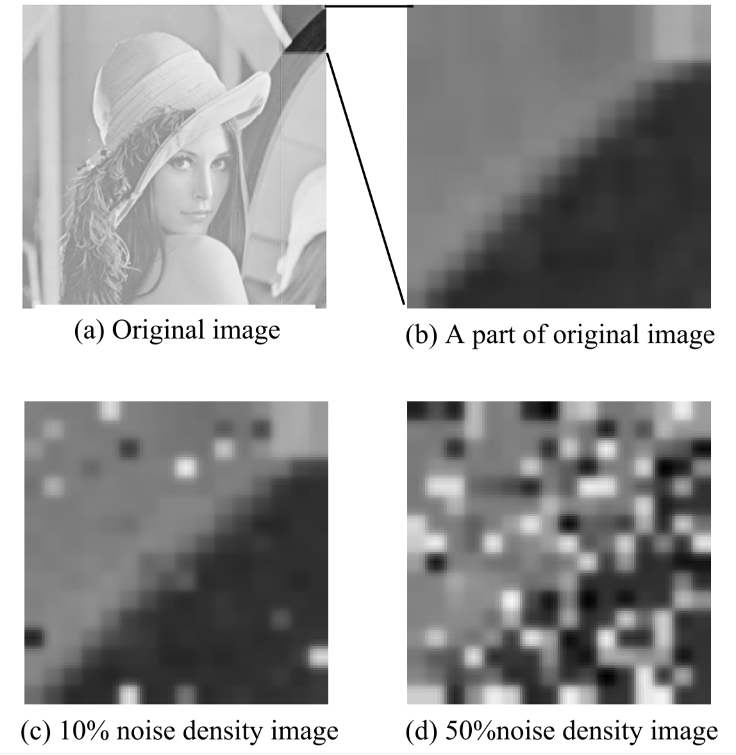

(a) Original image
(b) A part of original image
(c) 10% noise density image
(d) 50%noise density image

Fig. 6. Original image and the noise added image of Lenna are divided into16×16 pixels.

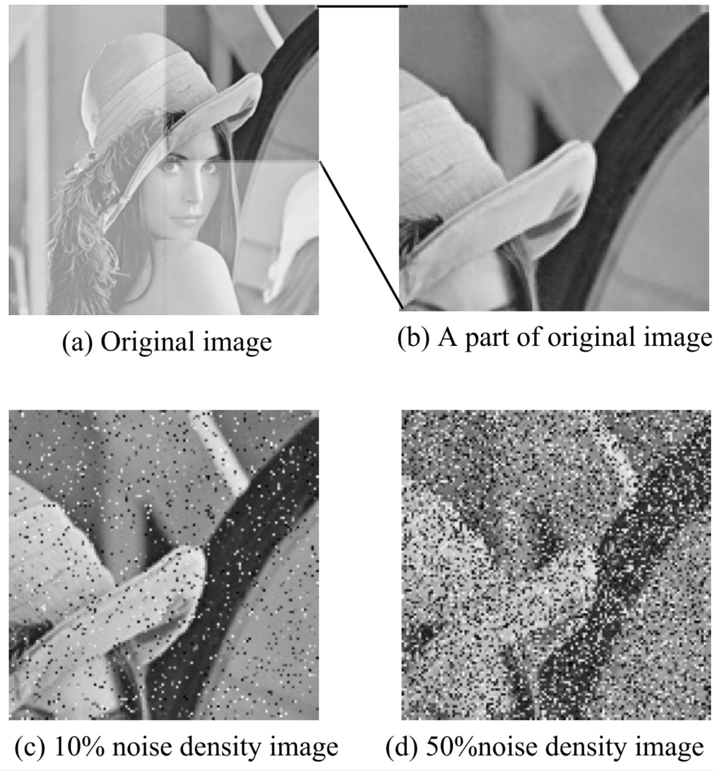

(a) Original image
(b) A part of original image
(c) 10% noise density image
(d) 50%noise density image

Fig. 7. Original image and the noise added image of Lenna are divided into 128×128 pixels.

According to Fig. 4, it is found that the average value of the PSNR of the resultant image with the division size of 8×8 pixels is the lowest. The results of two cases, division size 32×32 and 64×64 pixels, were almost the same, and the best results were obtained. Therefore, the suitable division size for image denoising is considered to be 32×32 or 64×64 pixels.

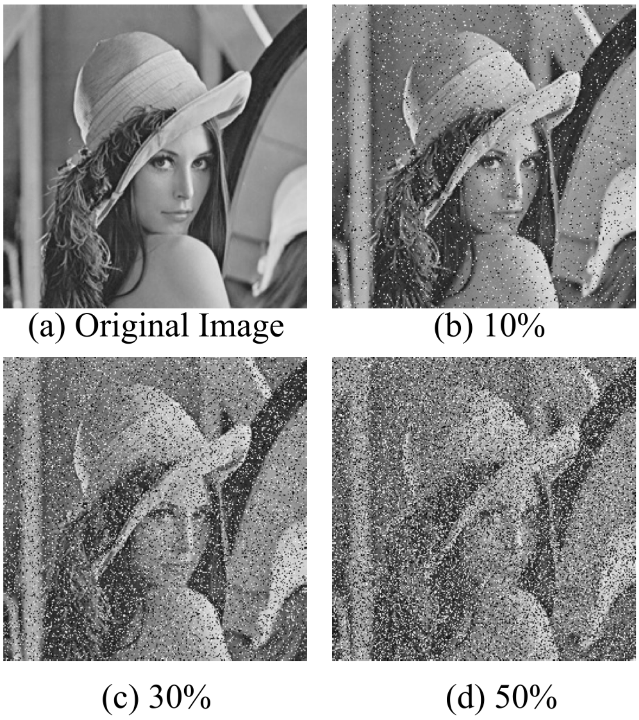
Fig. 5 shows comparison of denoising performance with proposed method and conventional methods. The result of the image division size of 64×64 pixels was used as the result of the proposed method.

According to Fig.5, the result of the proposed method is the best result. Especially, when noise density is 10%, the result of proposed method is about 3dB higher than the Multi-directional SMF, which shows the best result among the conventional methods. In addition, when noise density is 20% to 50%, the result of proposed method is about 2dB higher than Multi-directional SMF.

In addition, in the case of the low density of noises, the result is obtained that the image quality tends to be good when the division size of images used for learning is small. On the other hands, in the case of the density of noises is contrarily high, the larger division size of images to remove the noises, the image quality is better.

In order to explain this reason, the original image of the image Lenna divided into 16×16 pixels and the image added with noise are shown in Fig. 6. Similarly, the original image divided into 128×128 pixels and the image added with noise are shown in Fig. 7. According to Figures 6 and 7, in the case of the 10% of the noise density, the division size of 16×16 pixels is possible to distinguish the edge from the noise better than the division size of 128×128 pixels. However, in the case of the 50% of the noise density, it is difficult to become distinguished the edge from noise when division size of 16×16 pixels is used as image size for learning of configuring the neural network. Because the ratio of the noise added on edge part increase more compared to 10% noise. Conversely, it is considered that the division size of 128×128 pixels is easier to distinguish noises from the geometric characteristics existing in an image than the division size of 16×16 pixels.

Therefore, it is better to enlarge the division size of the image used for learning as the ratio of noise increases.



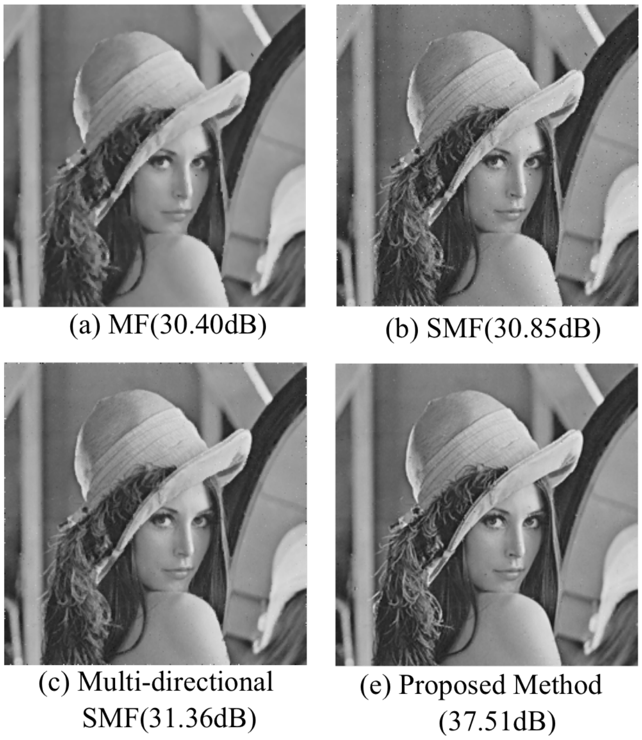
(a) Original Image

(b) 10%

(c) 30%

(d) 50%

Fig. 8. Original Image and Noised Images for the experiments.



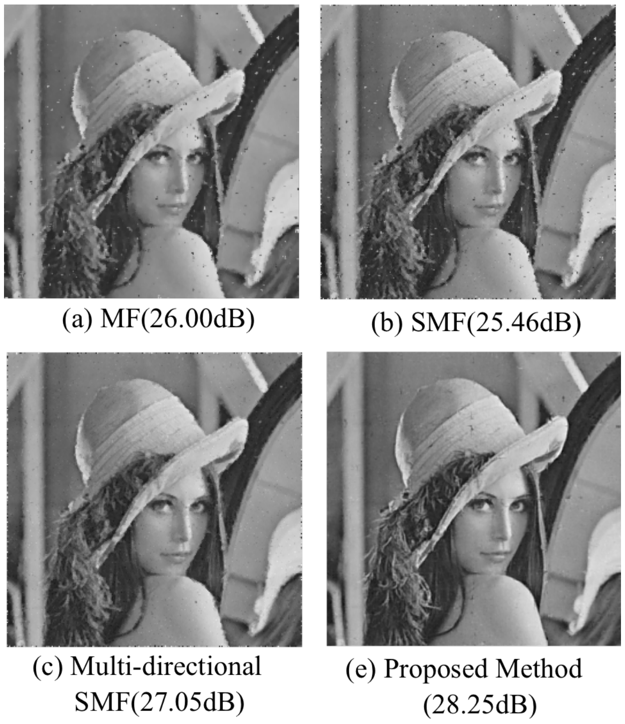
(a) MF(30.40dB)

(b) SMF(30.85dB)

(c) Multi-directional SMF(31.36dB)

(e) Proposed Method (37.51dB)

Fig. 9. Denoising results of 10% noise.



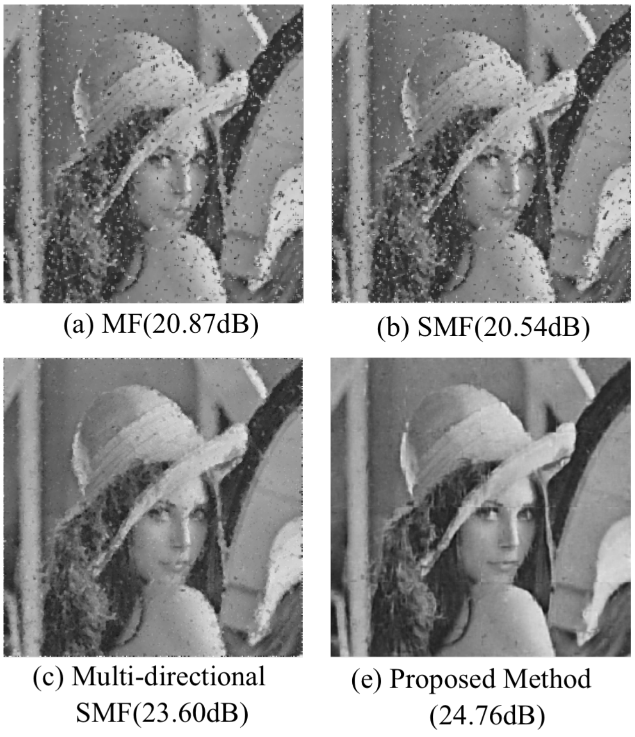
(a) MF(26.00dB)

(b) SMF(25.46dB)

(c) Multi-directional SMF(27.05dB)

(e) Proposed Method (28.25dB)

Fig. 10. Denoising results of 30% noise.



(a) MF(20.87dB)

(b) SMF(20.54dB)

(c) Multi-directional SMF(23.60dB)

(e) Proposed Method (24.76dB)

Fig. 11. Denoising results of 50% noise.

4.2 Comparison of denoising performance

Next, the image with noise reduction is to be shown. An image called Lenna was used as the representative of the test images. Fig. 8 shows original image and images with random noise, and these noise densities are as follows: 10%, 30% and 50%. Figures 9, 10 and 11 show the results of image denosing by conventional methods and the proposed method (64×64-pixels).

According to Fig. 9, in case of noise density is 10%, MF can remove almost all the noises, but the edge of image is blurred. On the other hand, SMF and Multi-directional SMF keep the form of edges clear, but the noises are not completely removed partially. However, the proposed method can remove noises while maintain the clear form edges.

According to Fig. 10, in case of noise density at 30%, Multi-directional SMF and the proposed method remove noises more effectively than MF and SMF. In addition, the proposed method hold edges clearer than SMF.

According to Fig. 11, in case of noise density at 50%, the proposed method is the best way for image denoising. At other different noise ratios, image edges are not kept clear by the proposed method, and consequently, the image becomes like a mosaic image. Therefore, it remains as a future task.

5. Conclusions

We proposed a method of image denosising using Deep Learning and compared it with conventional methods. From experimental results, it was found that the division sizes of the suitable image for image denoising are 32×32 and 64×64 pixels. Moreover, we found that the proposed method is superior to the conventional method at all noise density. Furthermore, we found that the proposed method can remove noise while maintaining the edge more than the conventional method. However, in this paper, noise is removed by using the result learned with the same addition rate as the noise added to the test image.

For that reason, we would like to discuss future cases where the noise addition ratio of the image used for learning is different from the noise addition ratio of the image that actually performs image denoising.

In addition, this paper focused on the performance of image denoising against division size of image. In the future, we will also investigate about the relationship between the learning time and the division size of image, and carry out the experiment on the speed of proposed method compared to other methods.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number 16K00260 and 17H00383.

References

(1) Hiroaki MURAKAMI, Nobuo NAKAMURA and Toshihiro NISHIMURA : “A study of noise reduction of CMOS image sensor using CNN”, IEICE Technical Report, Vol. MBE2006, No. 108, pp. 49-52, 2007.

(2) Junichi NAKAMURA : “IMAGE SENSOR and SIGNALPRCESSING for DIGITAL STILL CAMERAS,” Taylor and Fancis Group, 2006.

(3) Patrenahalli M. Narenda : “A Separable Median Filter for Image Noise Smoothing”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-3, No. 1, pp.20-29, 1981.

(4) Tong Sun and Yrjo Neuvo : “Detail-preserving median based filters in image processing”, Pattern Recognition Letters, Vol. 15, No. 4, pp.341-347, 1994.

(5) Zhou Wang and D. Zhang : “Progressive switching median filter for removal of impulse noise from highly corrupted images”, IEEE Transactions on Circuits and Systems Ⅱ : Analog and Digital Signal Processing, Vol. 46, No. 1, pp.78-80, 1999.

(6) Shuqun Zhang and M.A. Karim : “A new impulse detector for switching median filters”, IEEE Signal Processing Letters, Vol. 9, No. 11, pp.360-363, 2002.

(7) Yiqiu Dong, Raymond H. Chan and Shufang Xu : “A Detection Statistic for Random-Valued Impluse Noise”, IEEE Transactions on Image Processing, Vol. 16, No. 4, pp.1112-1120, 2007.

(8) Ryosuke KUBOTA and Noriaki SUETAKA : “Distribution Distance-based Threshold Auto-tuning Method for Switching Median Filter”, IEICE Electronics Express”, Vol. 7, No. 17, pp.1310-1316, 2010.

(9) Yasuki YOKOYAMA, Takashi MIYAZAKI, Mitsuo SONE, et al. : “Switching Median Filter Using Multi-direction Scanning 2x2 Noise Detector and Averaging Method”, IEICE Trans(A), Vol. J95-A, No. 10, pp.737-750, 2012.

(10) Yasuki YOKOYAMA, Takashi MIYAZAKI, Mitsuo SONE, et al. : “Basic Study on Threshold Calculation Based on Edge Information of Multi-directional

Switching Median Filter", ITE Technical Report, Vol. 37, No. 35, pp23-26, 2013.

(11) Takayuki OKADA : "Machine Learning Professional Series Deep Learning", Kodansha Co., Ltd, pp79-95, 2015.

(12) Naoki SHIMADA and Takeshi OOURA : "Learning with Chainer Introduction to Deep Learning", Gijutsuhyoronsha Co., Ltd, pp.108-124, 2017.

(13) Yasuki SAITO : "Make it from Scratch Deep Learning – Learning with Python Theory and Implementation of Deep Learning", O'Reilly Japan, Inc., pp.87-95, 2016.

(14) Preferred Networks, inc., Preferred Infrastructure, inc., "Chainer: A flexible framework for neural networks", Preferred Networks, inc., Preferred Infrastructure, < https://chainer.org>, retrieved 20 November 2017.

(15) D. Kingma and J. Ba. : "Adam: A method for stochastic optimization", International Conference for Learning Representations, 2015.

(16) Q. Huynh-Thu and M. Ghanbari : "Scope of validity of PSNR in image/video quality assessment", Electronics Letters, Vol. 44, No. 13, p800-801, 2008.