A PROPOSED METHOD OF SALT AND PEPPER NOISE REMOVAL IN IMAGE PROCESSING

Nguyen Thi Thanh Nhan
TNU - University of Information and Communication Technology

ARTICLE INFO

Received: 02/11/2023
Revised: 07/12/2023
Published: 07/12/2023

KEYWORDS

Salt and pepper noise
Image denoising
Image restoration
Adaptive median filter
Image processing

ABSTRACT

Salt and pepper denoising is an important pre-processing step in image processing to help improve image quality. Traditional denoising methods consider pixels with zero or maximum values as actual salt-and-pepper noise pixels. However, in many cases, these pixels are not noise pixels but rather texture pixels that lead to limited denoising effectiveness, especially when the noise ratio is high. This study overcomes the above limitation. At the noise pixel detection step, the algorithm uses an adaptively sized window to help distinguish the noise candidate points as real noise pixels or texture pixels based on the observation that the noise pixels are usually isolated pixels, whose value changes suddenly compared to the surrounding values. Then the noise pixel will be replaced by a new value based on the value of the neighboring pixels in the window. This method is only applied to grayscale images. The experimental results give the mean peak signal-to-noise ratio and structural similarity at all noise levels of 28.1631 and 0.8496, respectively. The proposed method for removing salt-and-pepper noise is very effective based on experimental results. The result is good quality images at all noise levels.

MỘT PHƯƠNG PHÁP ĐỀ XUẤT KHỬ NHIỄU MUỐI TIÊU TRONG XỬ LÝ ẢNH

Nguyễn Thị Thanh Nhàn
Trường Đại học Công nghệ thông tin và Truyền thông - ĐH Thái Nguyên

THÔNG TIN BÀI BÁO

Ngày nhận bài: 02/11/2023
Ngày hoàn thiện: 07/12/2023
Ngày đăng: 07/12/2023

TỪ KHÓA

Nhiễu muối tiêu
Khử nhiễu ảnh
Khôi phục ảnh
Lọc trung vị thích nghi
Xử lý ảnh

TÓM TẮT

Khử nhiễu muối tiêu là một bước tiền xử lý quan trọng trong xử lý ảnh để nâng cao chất lượng hình ảnh. Các phương pháp khử nhiễu truyền thống xem xét các điểm ảnh có giá trị bằng 0 hay giá trị cực đại là điểm ảnh nhiễu muối tiêu thực sự, tuy nhiên trong nhiều trường hợp những điểm ảnh này không phải là điểm ảnh nhiễu mà là điểm ảnh kết cấu dẫn đến kết quả khử nhiễu còn có hạn chế đặc biệt khi tỷ lệ nhiễu là cao. Nghiên cứu này sẽ khắc phục hạn chế trên, tại bước phát hiện điểm ảnh nhiễu, thuật toán sử dụng cửa sổ có kích thước thích nghi để phân biệt một điểm nhiễu ứng viên là điểm nhiễu thực sự hay là điểm ảnh kết cấu dựa trên quan sát rằng điểm ảnh nhiễu thường là điểm ảnh cô lập, ở đó giá trị điểm ảnh là thay đổi so với các giá trị điểm ảnh xung quanh. Sau đó điểm ảnh nhiễu sẽ được thay thế bởi một giá trị mới dựa trên các điểm lân cận. Phương pháp này chỉ áp dụng trên ảnh đa mức xám. Các kết quả thực nghiệm cho giá trị trung bình tỷ lệ tín hiệu trên nhiễu cực đại và độ tương tự về cấu trúc ở tất cả các cấp độ nhiễu lần lượt là 28,1631và 0,8496. Phương pháp đề xuất khử nhiễu muối tiêu là rất hiệu quả dựa trên các kết quả thực nghiệm. Các kết quả sau khi khử nhiễu cho chất lượng ảnh tốt tại tất cả cấp độ nhiễu.

DOI: https://doi.org/10.34238/tnu-jst.9133

Email: nttnhan@ictu.edu.vn

1. Introduction

Image denoising is an important pre-processing step in image processing to help improve image quality to serve the following image processing steps such as segmentation, edge detection, feature extraction, identification,…[1]. Digital images are often corrupted due to noise in the process image acquisition and transmission [2]. Salt-and-pepper (SaP) noise, also known as impulse noise, is a popular type of noise [2] - [4]. This noise can be caused by sharp and sudden disturbances in the image signal so it reduces image quality. In the noisy image, the noisy pixels are divided into 2 classes [5]: salt pixels (white pixels) and pepper pixels (black pixels). The salt pixels have the maximum gray value and the pepper pixels have the minimum gray value of the image. For an 256 grayscale image, the salt pixel value is 255, and the pepper pixel value is 0.

To remove SaP noise, there are two main groups of methods: traditional methods and deep learning methods. For the traditional method, there are many proposed methods, in which median filtering (MF) and adaptive median filtering (AMF) [6] are the two popular methods used a lot in the early stage. MF can restore images well when the noise ratio is low, but give poor results when the noise ratio is high [7] and the resulting image after performing the filter will be blurred, especially when applying the median filter many times [8]. AMF uses an adaptively sized window that improves the results of recovering images with a high noise ratio [9]. Current studies are also mainly based on these two methods. The second group of methods is deep learning, which is also recently interested by researchers [8], but this method depends on data and requires high computer hardware configuration.

There are two types of filters: linear filters and nonlinear filters [3], [10]. Linear filters can only remove low-density noise. They are not effective for medium-density and high-density noise. Inversely, nonlinear filters are more effective, especially for SnP noise. Among nonlinear filters, the Median Filter (MF) is a simple and effective filter for removing low-density noise (usually up to 20%).

MF is the most commonly used algorithm to remove pulse interference and SaP noise [11]. The median filter will include a window of size $m \times n$, this window will in turn move to each pixel in the image containing noise, then the window will capture the pixels of this image. The considered pixel will be replaced by the median value of the pixels obtained in the window. Noisy pixels will now be replaced by the median value of surrounding points, so this filter has the effect of reducing noise. The median value is calculated as follows: first arrange the pixel values in ascending (or descending) order. If the window size is odd, the median value is the value at position $\frac{m\times n+1}{2}$, if the window size is even, the median value is the average of the 2 middle values at position $\frac{m\times n}{2}$ and $(\frac{m\times n}{2}+1)$. Because all pixels are replaced by the median value of the points in the window, the disadvantage of this method is that the image is blurred compared to the original image.

MF is widely used as it is very effective at removing noise while preserving edges. This method of removing salt and pepper noise is very effective. Efficiency decreases when the number of noise points in the window is greater than or equal to half of the number of points in the window. Many methods based on median filtering have been proposed. In [12], a new adaptive weighted mean filter is proposed. For each pixel, expand the window size continuously until the maximum and minimum values of two consecutive window enlargements are equal. The current pixel is considered a candidate noise pixel if it is equal to the maximum or minimum value. The candidate noise pixels are then replaced with the weighted average of the current window.

The improvement studies of MF is the adaptive median filter (AMF) [6], for the MF filter that does not work effectively when the noise density is from medium to high, the AMF filter will overcome this. AMF uses variable-size windows along with adaptive conditions to determine the maximum, minimum, and median values in the window. If the conditions are satisfied, the center pixel in the window will be replaced with the median value. The larger the size of the window, the more time it takes to calculate, in many studies the maximum size of the window is often chosen to be 9.

The salt and pepper noise recovery is to use the remaining information (except for the noise pixel) to create a new, better value to replace that noisy pixel. Some previous noise reduction methods are still limited when working with images with a high noise ratio, and the resulting image is blurry.

To help improve the above problem, in this paper, an effective salt and pepper denoising method is proposed. The method is divided into two stages: salt and pepper noise detection and denoising. The propotion-based salt and pepper noise pixel detection method distinguishes noise pixels from texture pixels, thereby avoiding the treatment of texture pixels as noise pixels. Then the noisy pixels will be replaced by the mean of its neighboring pixels (pixels that are not noise pixels).

2. Method

The proposed method includes two steps: SaP noise detection and SaP noise elimination. Details are presented as follows.

In this work only 8-bit gray-level images are considered. Let $X = [x_{i,j}]_{m\times n}$, $Y = [y_{i,j}]_{m\times n}$, $R = [r_{i,j}]_{m\times n}$ be a original image, a corrupted image by SaP noise and a restored image, respectively, where m, n are the number of pixels in rows and columns.

In a corrupted image, the value of a ”salt” pixel equals to the maximum gray value 255, and the value of a ”pepper” pixel equals to the minimum value 0. Thus, $y_{i,j}$ is defined by Eq. (1).

$$y_{i,j} = \begin{cases} 0, with\ probability\ p \\ 1, with\ probability\ q \\ x_{i,j}, with\ probability\ 1-(p+q) \end{cases} \quad (1)$$

where $p, q, p+q \in [0.1]$. $W_{i,j}(r)$ is used to represent a $(2r + 1)\times(2r + 1)$ window centered at $(i; j)$ with the radius r.

2.1. SaP noise detection

With the salt and pepper noise feature, the noise pixel will receive one of two values of 0 or 255. Therefore, the candidate noise pixel $y_{i,j}$ has only one of two possibilities, $y_{min} = 0$ and $y_{max} = 255$. An indicator matrix $O = [o_{i,j}]_{m\times n}$ is a prior decision condition in noise detection.

$$o_{i,j} = \begin{cases} 1, y_{i,j} = 0\ or\ 255 \\ 0,\ otherwise \end{cases} \quad (2)$$

Matrix O is a binary matrix with value 1 corresponding to a noisy candidate pixel, and value 0 corresponding to a noiseless pixel. For grayscale images, pixels with a value of 0 or 255 can be salt and pepper pixels or texture pixels. The noisy pixel candidate can be noise or not because in many cases these pixels can be texture pixels, so it is necessary to confirm whether the noisy pixel candidate are noise or not. Noise is usually isolated pixels, whose value changes suddenly compared to the surrounding pixels, while texture pixels often have similar values compared to the surrounding. For the pixel $y_{i,j}$, if it is finally detected as noisy, it is marked with the discriminant matrix $L = [l_{i,j}]_{m\times n}$, and $l_{i,j} = 1$, else $l_{i,j} = 0$ and will not be processed. Based on this observation, the noise detection method is determined as follows:

For each pixel (i, j) in the noisy image Y and the initially restored image R, do

- Step 1: If $o_{i,j} == 0\ then\ l_{i,j} = 0, r_{i,j} = y$, break;
  Otherwise, go to step 2.
- Step 2: Initialize $r = 1, h = 1, r_{max} = 5$, where r_{max} is the maximum size of window.
- Step 3: $S_{i,j}^{sum}(r)$ is the number of pixels within $W_{i,j}(r)$ which are not equal to 0 and 255. For the pixel $y_{i,j}$ with $o_{i,j} = 1$, $S_{i,j}^{sum}(r)$ is caculated within an adaptive searching window $W_{i,j}(r)$. The

radius r of window $W_{i,j}(r)$ is initialed to 1. If $r == r_{max}$ or $S_{i,j}^{sum}(r) > 0$, the r is selected. If not, then r+1 and continue to compute. If $S_{i,j}^{sum}(r) > 0$, $y_{i,j}$ is determined noise pixel, so $l_{i,j}$=1. Otherwise go to *Step* 4.

• Step 4 : If $S_{i,j}^{sum}(r) = 0$, $y_{i,j}$ maybe considered as a texture pixel. The proportion of pixels with the same gray level as the candidate pixel $y_{i,j}$ in the $W_{i,j}(r)$ window (denoted is $S_{i,j}^{num}(r)$) to the number of pixels in the $W_{i,j}(r)$ window.

$$\theta = \frac{S_{i,j}^{num}(r)}{(2r+1)\times(2r+1)} \tag{3}$$

After a threshold T is chosen to identify the noisy pixel. If $\theta \leq T$, the candidate pixel $y_{i,j}$ is regards as a noisy pixel so $l_{i,j}$=1, else it is noiseless.

2.2. SaP Noise Elimination

Let R represent restored image, R is initialized with Y. After detecting a noise pixel $y_{i,j}$ corresponding to $l_{i,j}$=1, this pixel will be replaced with a new value $S_{i,j}^{mean}$. The formula is calculated according to the equation below:

$$\mathrm{S}_{\mathrm{i,j}}^{\mathrm{mean}} = \begin{cases} \dfrac{\sum_{(c,d)\in W_{i,j}(r)}\big(1-l(c,d)\big)*y_{c,d}}{\sum_{(c,d)\in W_{i,j}(r)}\big(1-l(c,d)\big)}, S_{i,j}^{sum}(r) \neq 0 \\ \dfrac{\sum_{(c,d)\in W_{i,j}(r)} y_{c,d}}{number\ of\ pixels\ in\ W_{i,j}(r)}, otherwise \end{cases} \tag{4}$$

$S_{i,j}^{mean}(r)$ is the mean of the noiseless pixels in $W_{i,j}(r)$ when $S_{i,j}^{sum}(r) \neq 0$, otherwise is the mean of pixels of $y_{i,j}$ in $W_{i,j}(r)$.

The detail of the proposed method are shown in Algorithm 1.

Algorithm 1. Proposed method
Input: The noisy image Y
Output: The restored image R
Compute the indicator matrix O.
For each pixel (i,j) in the noisy image Y and the initially restored image R, do
1) If $o_{i,j} == 0, l(i,j) = 0, r_{i,j} = y$, break;
Otherwise, go to step 2).
2) Initialize r= $1, h = 1, r_{max} = 7$.
3) Compute $S_{i,j}^{sum}(r)$ until r== r_{max} or $S_{i,j}^{sum}(r) > 0$;
Otherwise, r= $r + h$ and repeat step 3).
4) If $S_{i,j}^{sum}(r) > 0$, $l(i, j) = 1$, $r_{i,j} = S_{i,j}^{mean}(r)$, break;
Otherwise, go to step 5).
5) Compute θ. If $\theta \leq T$, $l(i, j) = 1$, $r_{i,j} = S_{i,j}^{mean}(r)$;
Otherwise, $l(i; j) = 0$, $r_{i,j} = y_{i,j}$

3. Results

In the experiments, proposed methods are compared with 2 methods: MF, AMF [6]. The two typical image quality metrics used are peak signal-to-noise ratio (PSNR) [13] and structural similarity (SSIM) [14] to evaluate the experimental results. PSNR can be defined as follows:

$$PSNR = 10log_{10}\left(\frac{u_{max}^2}{MSE}\right),$$
$$MSE = \frac{1}{mn}\sum_{i=1}^{m}\sum_{i=1}^{n}\left(w_{ij}-u_{ij}\right)^2 \tag{5}$$

where MSE is the mean square error, u is a noise-free image, u_{max} is the maximum gray value, for example, for grayscale image $u_{max} = 255$. A low PSNR value corresponds to low image quality, a high PSNR value corresponds to high image quality. Structural similarity (SSIM) is a qualitative metric and is proven to be a better error metric and its value is in [0, 1]. A low SSIM value corresponds to low image quality, a high SSIM value corresponds to high image quality. This metric based on the characteristic of the human vision. SSIM for image *U* and *V* can be defined as follows:

$$SSIM = \frac{(2\mu_U\mu_V + c_1) * \left(2\sigma_{U,V} + c_2\right)}{(\mu_U^2 + \mu_V^2 + c_1) * (\sigma_U^2 + \sigma_V^2 + c_2)} \tag{6}$$

where μ_U and μ_V are the average intensities of image U and V, respectively. σ_U and σ_V are standard deviations; $\sigma_{U,V}$ is the covariance; c_1 and c_2 are some constants. Here c_1 and c_2 are set to be $(0.01 * 255)^2$ and $(0.03 * 255)^2$ as in [14], respectively. Through test, threshold T = 0.8 is selected. The experiments are performed on a personal computer with Intel Core i3 3.0 GHz processor and 8 GB RAM.

The proposed method is tested on two image databases consisting of 82 images. The first database includes 14 images in the TEST IMAGES Database [15]: Barbara, Elaine, Goldhill, Man, Peppers, Yacht, Zelda, Baboon, Boat, Couple, Einstein, Face, House, and Straw. The second database include 68 images of BSD68 [16]. All images are stored in bmp, png, tiff format, grayscale and with the size of 512x512, 256x256, 321x481, 481x321 pixels.

For the first case, denoising methods are implenmented such as MF, AMF and the proposed method to remove noise of 50% on the Elaine image. Denoising results are presented in Figure 1.

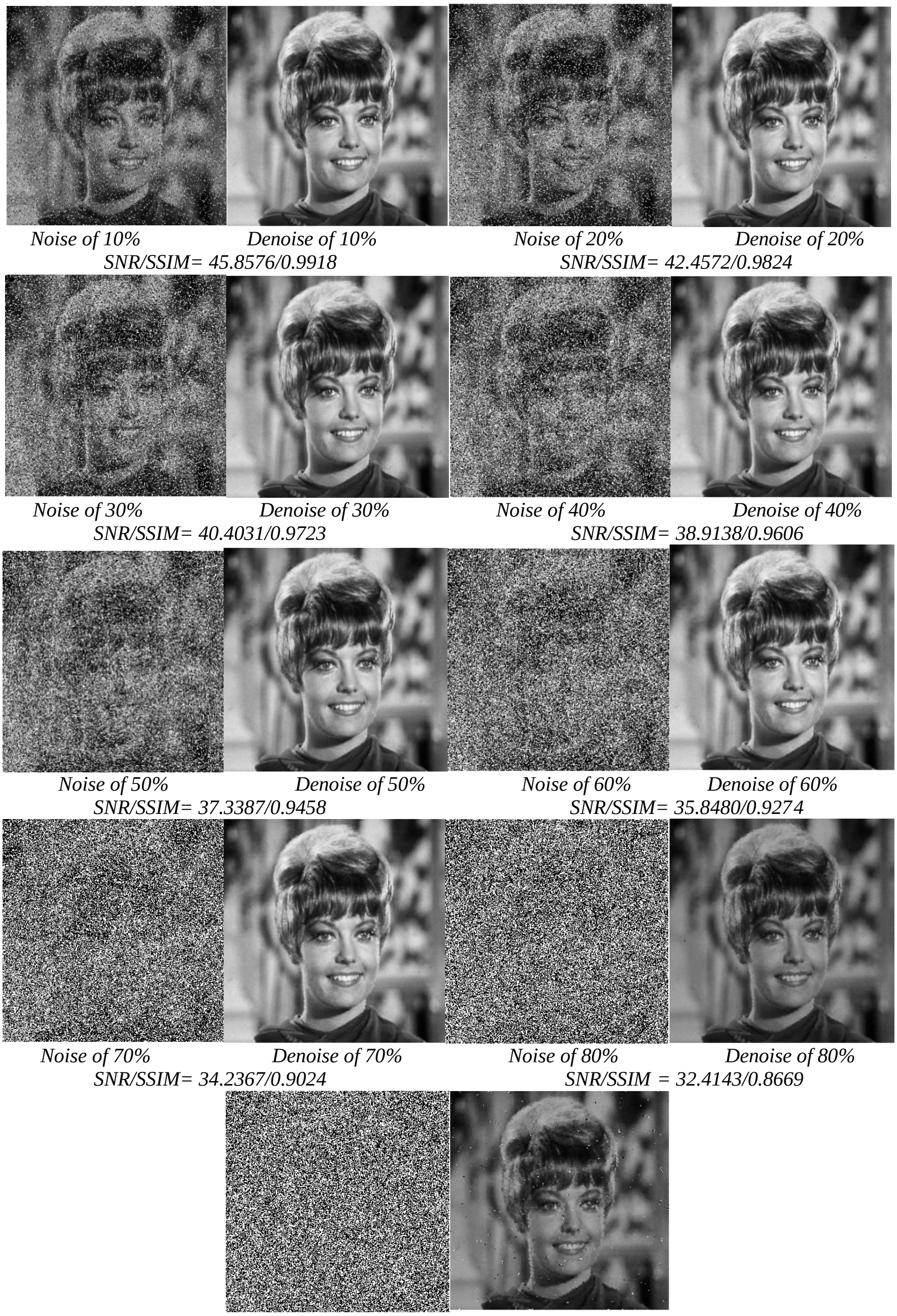
We can see that the noise has damaged Elaine's face image, making it difficult to see all the details of the face. The noise filtering results of the MF and AMF methods, the recovered image still has noise, with the proposed method, the image quality is very good, the noise has been removed, the result is not much different from the original image. By evaluating on the PSNR/SSIM measure, the proposed method received a value of 32.8821 dB/ 0.8571 giving the best result compared to the other two methods MF and AMF.

a) Original *b) Noisy* *c) MF* *d) AMF* *e) Proposed method*

Figure 1. *Experimental results of different methods for "Elaine" with SAP noise ratio of 50%. PSNR/SSIM of methods: a) Original image, b) Noisy image (8.4958 dB, 0.0223), c) MF (22.6608 dB, 0.5904), d) AMF (21.6704 dB, 0.4890), e) Proposed method (**32.8821 dB, 0.8571**)*

Figure 2 shows the results of denoising Zelda image with different levels of salt and pepper noise. The image is added salt and pepper noise with increasing rates from 10% to 90% and the corresponding denoising results when applying the proposed method. The resulting images are of very good quality for all noise levels. From the image with 60% noise ratio, we did not see any details from the original image, but the denoising results still give very good results except the 90% noise recovery image still has some noise pixels. These experimental results show the effectiveness of the proposed method.

Next, the three methods of denoising MF, AMF and the proposed method are applied on all 14 images that have been added with increasing noise ratios from 10% to 90%. The mean PSNR, SSIM values for each noise level are presented in Table 1 and Table 2 respectively. The results show that at all noise levels, the PSNR and SSIM values of the proposed method are superior to those of the MF and AMF methods. The mean value of PSNR over all noise levels of the proposed method is **28.1631**, MF is 18.3761, AMF is 19.9269. The mean value of SSIM over all noise levels of the proposed method is **0.8496**, MF is 0.4691, AMF is 0.5386.



Noise of 10% Denoise of 10%
SNR/SSIM= 45.8576/0.9918

Noise of 20% Denoise of 20%
SNR/SSIM= 42.4572/0.9824

Noise of 30% Denoise of 30%
SNR/SSIM= 40.4031/0.9723

Noise of 40% Denoise of 40%
SNR/SSIM= 38.9138/0.9606

Noise of 50% Denoise of 50%
SNR/SSIM= 37.3387/0.9458

Noise of 60% Denoise of 60%
SNR/SSIM= 35.8480/0.9274

Noise of 70% Denoise of 70%
SNR/SSIM= 34.2367/0.9024

Noise of 80% Denoise of 80%
SNR/SSIM = 32.4143/0.8669

Noise of 90% Denoise of 90%
SNR/SSIM= 25.5725/0.7518

Figure 2. *Denoising results of proposed method for the Zelda image with various noise levels (10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%)*

Table 1. *PSNR values of denoising results of the methods of 82 images with various noise levels*

Noise Level	MF	AMF	Proposed method
10%	25.7719	26.949	**36.0112**
20%	24.8264	25.1975	**32.8613**
30%	23.7692	23.2306	**30.8951**
40%	22.6328	21.3163	**29.3729**
50%	20.6438	19.538	**28.0169**
60%	17.2622	17.9068	**26.6957**
70%	13.3886	16.4040	**25.3291**
80%	9.9218	15.0125	**23.8156**
90%	7.1686	13.7876	**20.4703**
Mean	18.3761	19.9269	**28.1631**

Table 2. *SSIM values of denoising results of the methods of 82 images with various noise levels*

Noise Level	MF	AMF	Proposed method
10%	0.7142	0.8013	**0.9759**
20%	0.6992	0.7170	**0.9572**
30%	0.6809	0.6219	**0.9366**
40%	0.6534	0.5317	**0.9126**
50%	0.5820	0.4491	**0.8837**
60%	0.4187	0.3739	**0.8473**
70%	0.3780	0.3064	**0.8004**
80%	0.0728	0.2477	**0.7357**
90%	0.0229	0.2080	**0.5973**
Mean	0.4691	0.5386	**0.8496**

4. Conclusion

In this work a salt and pepper denoising method is proposed. The point of interest in this approach is to detect salt and pepper noise, distinguishing noise from texture pixels. The noisy pixels will be replaced by the mean of neighboring pixels $S_{i,j}^{mean}(r)$. Experimental results have shown the effectiveness of the proposed method on all noise levels compared with FM and AFM methods.

REFERENCES

[1] B. Goyal, A. Dogra, S. Agrawal, B. S. Sohi, and A. Sharma, “Image denoising review: From classical to state-of-the-art approaches,” *Inf. Fusion*, vol. 55, pp. 220–244, 2020.

[2] S. Esakkirajan, T. Veerakumar, A. N. Subramanyam, and C. H. PremChand, “Removal of High Density Salt and Pepper Noise Through Modified Decision Based Unsymmetric Trimmed Median Filter,” *IEEE Signal Process. Lett.*, vol. 18, no. 5, pp. 287–290, May 2011, doi: 10.1109/LSP.2011.2122333.

[3] U. Erkan and L. Gökrem, “A new method based on pixel density in salt and pepper noise removal,” *Turk. J. Electr. Eng. Comput. Sci.*, vol. 26, no. 1, pp. 162–171, 2018.

[4] D. N. Thanh, V. S. Prasath, and U. Erkan, “An improved BPDF filter for high density salt and pepper denoising,” in *2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF)*, IEEE, 2019, pp. 1–5.

[5] R. Rojas and P. Rodriguez, “Spatially adaptive total variation image denoising under salt and pepper noise,” in *2011 19th European Signal Processing Conference*, IEEE, 2011, pp. 278–282.

[6] H. Hwang and R. A. Haddad, “Adaptive median filters: new algorithms and results,” *IEEE Trans. Image Process.*, vol. 4, no. 4, pp. 499–502, 1995.

[7] R. H. Chan, C.-W. Ho, and M. Nikolova, “Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization,” *IEEE Trans. Image Process.*, vol. 14, no. 10, pp. 1479–1485, 2005.

[8] L. Liang, S. Deng, L. Gueguen, M. Wei, X. Wu, and J. Qin, “Convolutional neural network with median layers for denoising salt-and-pepper contaminations,” *Neurocomputing*, vol. 442, pp. 26–35, 2021.

[9] X. Deng, Y. Ma, and M. Dong, “A new adaptive filtering method for removing salt and pepper noise based on multilayered PCNN,” *Pattern Recognit. Lett.*, vol. 79, pp. 8–17, 2016.

[10] D. N. H. Thanh and S. Engínoğlu, “An iterative mean filter for image denoising,” *IEEE Access*, vol. 7, pp. 167847–167859, 2019.

[11] J. S. Lim, *Two-dimensional signal and image processing*, Prentice Hall, 1990.

[12] P. Zhang and F. Li, “A new adaptive weighted mean filter for removing salt-and-pepper noise,” *IEEE Signal Process. Lett.*, vol. 21, no. 10, pp. 1280–1283, 2014.

[13] A. Hore and D. Ziou, “Image quality metrics: PSNR vs. SSIM,” in *2010 20th international conference on pattern recognition*, IEEE, 2010, pp. 2366–2369.

[14] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

[15] N. Asuni and A. Giachetti, “TESTIMAGES: a Large-scale Archive for Testing Visual Devices and Basic Image Processing Algorithms,” in *STAG*, 2014, pp. 63–70.

[16] D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, IEEE, 2001, pp. 416–423.