A Scheme for Salt and Pepper Noise Reduction on Graylevel and Color Images

NUCHAREE PREMCHAISWADI*, SUKANYA YIMNGAM**, WICHIAN PREMCHAISWADI***
*Faculty of Information Technology, Dhurakijpundit University
110/1-4 Prachachuen Road Laksi, Bangkok 10210, Thailand
nucharee@dpu.ac.th
**Faculty of Resources and Environment
Kasetsart University, Bangkok, Thailand
E-mail: yimngam@yahoo.com
***Graduate School of Information Technology, Siam University
235 Petchkasem Road, Phasi-charoen, Bangkok 10163, Thailand
wichian@siam.edu

Abstract: - Noise in images can be found in almost scanning documents. The picture and text can be contaminated by additive noise from document scanning process or dirtiness in the document itself. This paper presents a scheme for Salt and Pepper noise reduction. This scheme combines the characteristics of an Applied kFill Algorithms and Median Filter by using window size of 3x3 and 5x5, depends on size of Salt and Pepper noise. The aim of this technique is to increase PSNR of picture images in the scanning documents. The experimental results show that the proposed scheme can remove Salt and Pepper noise better than the used of Applied kFill Algorithms and Median Filter. This scheme also can be apply in binary, gray scale, and color format of documents.

Key-Words: - Image Processing, Noise reduction, Salt and Pepper Noise, kFill Algorithm, Applied kFill Algorithm and Median Filter, PSNR

1 Introduction

In the document processing, scanning is the first process to convert a paper document into image documents. The scanned images might be contaminated by additive noise and these low quality images will affect to the next step of document processing. Therefore, a pre-processing process is required to improve the quality of images before sending to next stages in the document processing [1].

There are many kinds of noises in images. One additive noise called “Salt and Pepper Noise”, the black point and white point sprinkled all over image, typically looks like salt and pepper, which can be found in almost documents. A document usually uses lightly background color to highlight text. The digitized result of these documents will generate a salt and pepper noise on the background. When closer inspection of many real document images, it can be found that salt and pepper noise components are in binary, gray scale and color images.

Noise reduction is usually used in a pre-processing stage in image analysis process to improve the quality of images [2]. There are many methods proposed for removing salt and pepper noise such as kFill Algorithm, Applied kFill Algorithm and Median Filter [3-5]. The kFill and Applied kFill Algorithms are capable to remove simultaneously both salt noise and pepper Noise. However, these methods can be used only on binary image. The Median Filter can be used to remove these noises on binary, graylevel and color image. However, this method requires a long computation time. This paper presents a scheme which is the combination of Applied kFill Algorithm and Median Filter Algorithm to remove this noise on binary, gray scale and color image, with considerably less blurring than Median Filter and preserving useful detail in the image.

2. Related works

2.1 kFill Algorithm [3-5]

The kFill filter is the scheme designed to reduce salt and pepper noise. For the text images which the information is in binary, salt and pepper noise is almost prevalented. This noise appears as isolated pixel or pixel regions of ON noise in OFF backgrounds or OFF noise (holes) within characters and other foreground ON regions. In this algorithm, black pixel is called ON while, white pixel is called OFF. The process of removing this noise called “Filling”.

In this algorithm, a window size of *k x k* pixels is moved over an image in the raster-scan direction. Inside the window, there are (k-2) x (k-2) regions, called the core, and 4(k-1) pixels on the window perimeter, called the neighborhood. The filling operation entail setting all values of the core to ON or OFF, depends on pixel values in the neighborhood. The decision on whether or not to fill with ON(OFF) requires that all core values must be OFF(ON), depends on two variables, determined from the neighborhood. For a fill-value equal to ON(OFF), the n variable is the number of ON-OFF pixel in the neighborhood, and c is the number of connected group of ON-pixel in the neighborhood [3][4].

Filling occurred only when n is greater than a threshold, n and c is equal to 1. The value of n is set as a function of window size, n=3k-4, to retain the text features described above. The stipulation that c=1 ensures that filling does not change connectivity (that is, does not join two letters together or separate two parts of the same connected letter). This method is performed iteratively on the image until no filling occurs and use only in binary image.

2.2 Applied kFill Algorithm [3-5]

In many real document images, salt and pepper noise components are frequently larger than one pixel. In such case, the window size larger than 3x3 pixel should be used, and the kFill Algorithm will never fill the noise components smaller than the core size, The kFill Algorithm will not fill the core region neither with ON or with OFF, because all core pixel are not same value. In this algorithm, it fills the core with OFF when the majority of pixels are ON or when the majority of pixels are OFF. This method is fast and effective. It can remove noise of difference size and shape while maintaining the sharpness of the text and graphics components. However, the algorithm can be used only on binary images.

2.3 Median Filter [2][6]

The median filter is a non-linear digital filtering technique, often used to remove noise from images or other signals. The idea is to examine a sample of the input and decide if it is representative of the signal. This is performed using a window consisting of an odd number of samples. The values in the window are sorted into numerical order; the median value, the sample in the center of the window, is selected as the output. The oldest sample is discarded, a new sample acquired, and the calculation repeats. Median filtering is a common step in image processing. It is particularly useful to reduce speckle noise and salt and pepper noise. Its edge-preserving nature makes it useful in cases where edge blurring is undesirable.

3. The Proposed Algorithm

The proposed algorithm is the extension of the author's work [8] to remove salt and pepper Noise on binary, gray scale and color images. In this algorithm, black pixel is defined as ON, white pixel as OFF and k as windows size, like that used in kFill algorithm. The steps of the algorithm are as the following:

1. Count ON or OFF pixels of the core ((k-2)*(k-2)).
2. If the numbers of ON or OFF pixels of the core are more than a half of all pixels in the core $(((k-2)^2/2)+1)$. The decision is to fill the core with the median of pixel values from the core and the surrounding neighborhood.
3. If the numbers of ON or OFF pixels of the core are less than a half of all pixels in the core $(((k-2)^2/2)+1)$. The decision is to fill the core with the median of pixel values from the core.
4. If all pixels in the core are not ON or OFF then fill the core with the original pixels values.

The flowchart of proposed algorithm is shown in the figure 1.

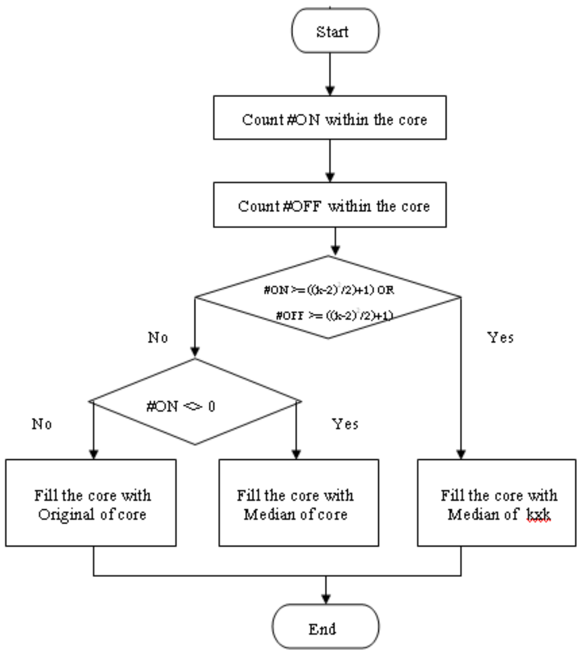


Figure 1: Flowchart of the proposed algorithm.

4. Peak signal-to-noise ratio [7]

The PSNR is most commonly used method to measure quality of reconstructed images. The calculation of the method is based on the mean squared error (MSE) which is defined as:

ISSN: 1790-5109

ISBN: 978-960-474-108-3

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}||I(i,j) - K(i,j)||^2$$

Where *I(i,j)* is original image, *K(i,j)* is the approximated version and *m*, *n* are the dimension of image.

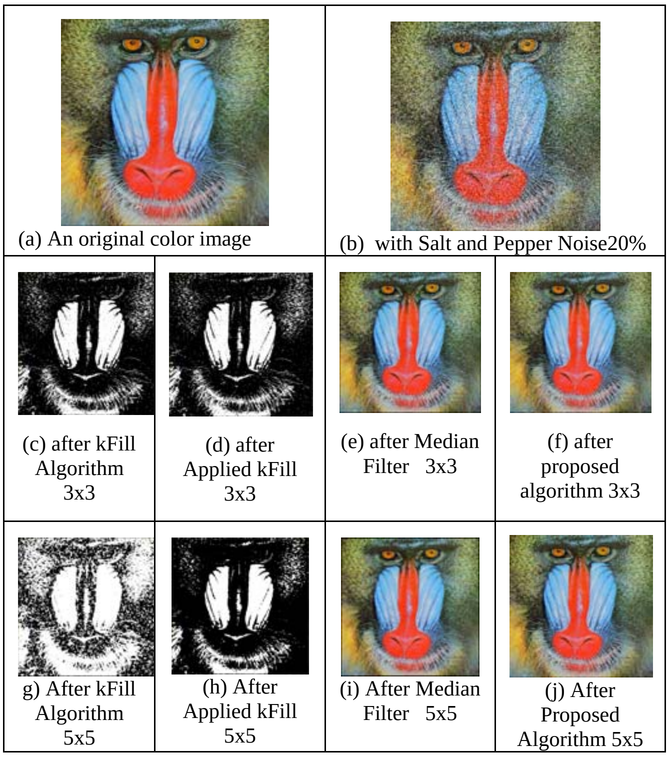
The PSNR is defined as:

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right)$$

Here, MAX_i is the maximum possible pixel value of the image. For color images with three RGB values per pixel, the definition of PSNR is the same except the MSE is the sum over all squared value differences divided by image size and by three. Typical values for the PSNR in lossy image are between 30 and 50 dB, where higher is better.

5 .Experimental results

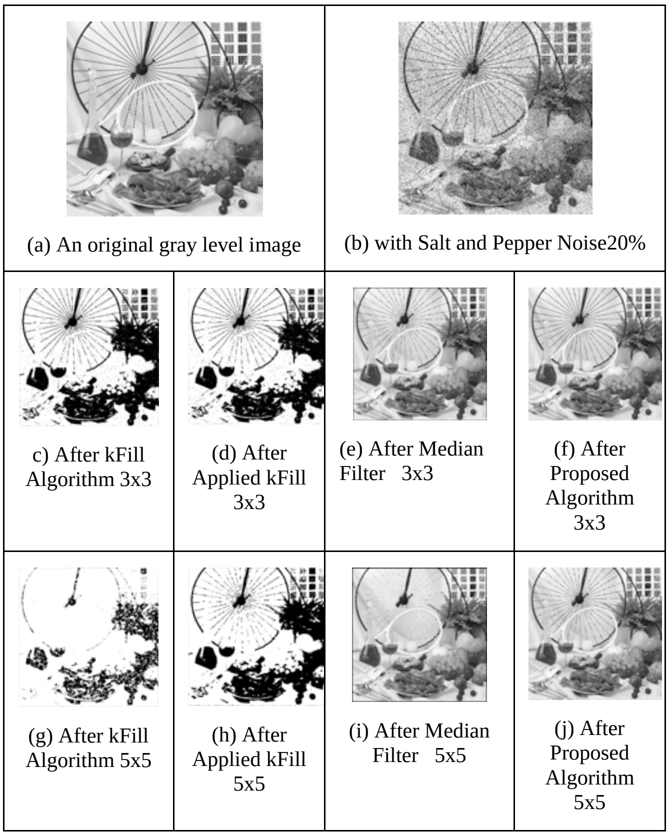
The proposed algorithm is implemented and tested with with 10 color images and 10 graylevel images which download from image database website.



(a) An original color image
(b) with Salt and Pepper Noise20%
(c) after kFill Algorithm 3x3
(d) after Applied kFill 3x3
(e) after Median Filter 3x3
(f) after proposed algorithm 3x3
g) After kFill Algorithm 5x5
(h) After Applied kFill 5x5
(i) After Median Filter 5x5
(j) After Proposed Algorithm 5x5

Figure 2: The Results of color image of a window size of 3x3 and 5x5

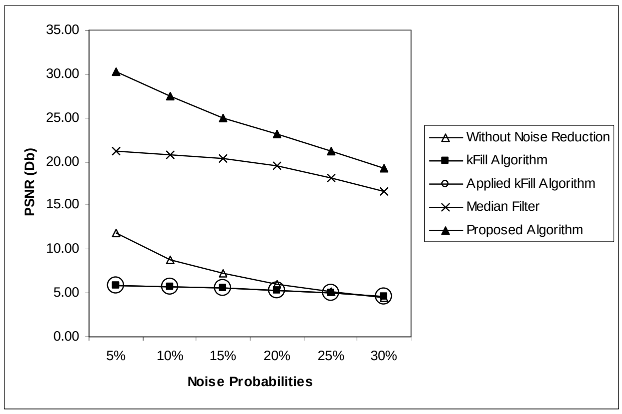
In the experiments, each image was added with salt and pepper noise with probabilities from 5% to 30%. Then, each image was tested by using window size of 3 x 3 and 5 x 5. The experimental results of the proposed are compared with kFill Algorithm, Applied kFill Algorithm, Median Filter algorithm. Some of the experiment results of the tested color and graylevel images are shown in Figure 2 and Figure 3, respectively.



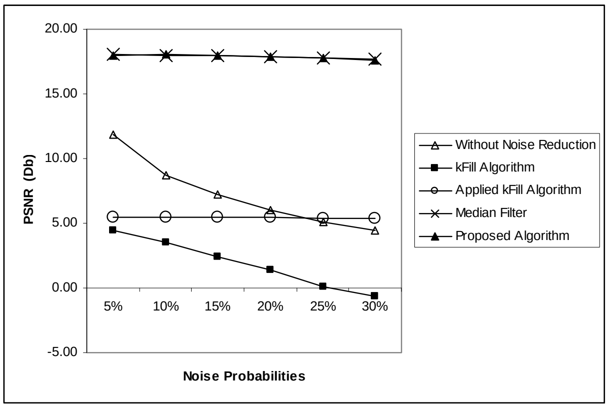
(a) An original gray level image
(b) with Salt and Pepper Noise20%
c) After kFill Algorithm 3x3
(d) After Applied kFill 3x3
(e) After Median Filter 3x3
(f) After Proposed Algorithm 3x3
(g) After kFill Algorithm 5x5
(h) After Applied kFill 5x5
(i) After Median Filter 5x5
(j) After Proposed Algorithm 5x5

Figure 3: The Results of gray image of a window size of 5x5

The PSNR of color and graylevel images of each method is shown in Table 1 and Table 2. The comparison graphs of PSNR of the proposed scheme with other methods are also shown on Figure 4 and Figure 5.

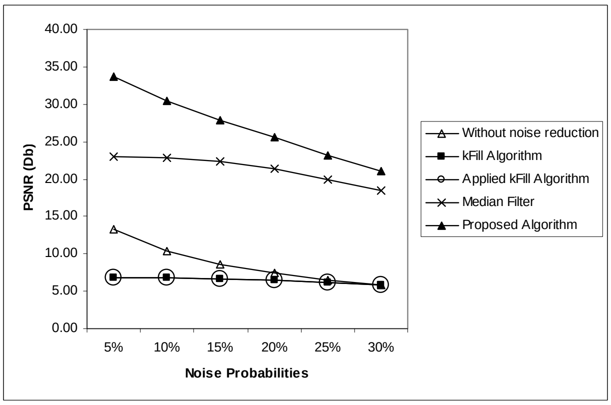


a) Window size of 3x3

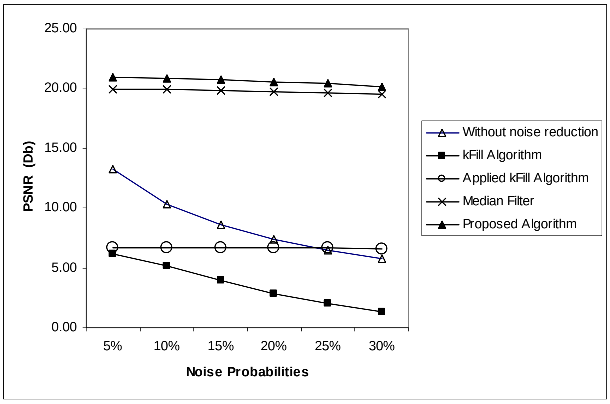


b) Window size of 5x5

Figure 4 The graph of PSNR for graylevel images



a) Window size of 3x3



b) Window size of 5x5

Figure 5 The graph of PSNR of color images

6 .Conclusion and discussion

This paper presents algorithms for salt and pepper noise reduction in image documents. This scheme is a combination of Median Filter and Applied kFill Algorithm. This proposed algorithm can remove these noises of any size that are smaller than the size of document objects. This method is fast and can be used effectively on binary, gray scale and color image, with considerably less blurring than another method and preserving useful detail in the image. The experiment results show that this proposed scheme can significantly increase PSNR of color and graylevel images.

References

[1] R.C. Gonzalez and R.E. Woods., 1992, *Digital Image Processing*, Addison-Wesley.

[2] National Institute of Standard and Technology." Noise Reduction – The Rank Filter" [Online]. Available http://www.nist.gov/lispix/imlab/noise/shottc.html. 2003.

[3] Krisana Chinnasarn, Yuttapong Rangsanseri and Punya Thitimajshima, "Removing Salt-and-Pepper Noise in Text/Graphics Images". IEEE Computer Society Press, 1998.

[4] G.A. Story, L.O'Gorman and D.Fox., "The RightPages Image-Based Electronic Library for Alerting and Browsing" Computer, Vol. 25, No.9, Sept. 1992, pp.17-26.

[5] L.O'Gorman and R. Kasturi., Document Image Analysis. IEEE Computer Society Press, 1995.

[6] Wikipedia, the free encyclopedia "Median Filter" [Online]. Available :http://en.wikipedia.org/wiki/Median_filter

[7] Wikipedia, the free encyclopedia "Peak signal-to-noise ratio" [Online]. Available : http://en.wikipedia.org /wiki/Peak_signal-to-noise_ratio

[8] Nucharee Premchaiswadi, Sukanya Yimngam and Wichian Premchaiswadi, "An Input Improvement for an Optical Character Recognition System using Noise Reduction", Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2007 (pp. 2136-2143).

Table 1 - The comparison of PSNR of color images of window size of 3x3 and 5x5

Algorithms	PSNR											
	Window size of 3x3 Noise Probability (%)						Window size of 5x5 Noise Probability (%)					
	5%	10%	15%	20%	25%	30%	5%	10%	15%	20%	25%	30%
Original image with noise	13.23	10.31	8.59	7.42	6.52	5.82	13.23	10.31	8.59	7.42	6.52	5.82
kFill Algorithm	6.84	6.77	6.63	6.42	6.14	5.83	6.16	5.14	3.96	2.87	2.00	1.35
Applied kFill Algorithm	6.84	6.77	6.63	6.42	6.14	5.83	6.72	6.71	6.69	6.67	6.64	6.58
Median Filter	23.05	22.80	22.29	21.36	19.90	18.50	19.99	19.92	19.84	19.76	19.68	19.55
Proposed Algorithm	33.69	30.50	27.91	25.57	23.12	21.10	20.92	20.87	20.74	20.58	20.40	20.12

Table 2 - The comparison of PSNR of gray level images of window size of 3x3 and 5x5

Algorithms	PSNR											
	Window size of 3x3 Noise Probability (%)						Window size of 5x5 Noise Probability (%)					
	5%	10%	15%	20%	25%	30%	5%	10%	15%	20%	25%	30%
Original image with noise	11.86	8.73	7.21	6.05	5.13	4.42	11.86	8.73	7.21	6.05	5.13	4.42
kFill Algorithm	5.81	5.69	5.57	5.33	5.01	4.66	4.43	3.52	2.41	1.38	0.10	-0.66
Applied kFill Algorithm	5.81	5.69	5.57	5.33	5.01	4.66	5.48	5.47	5.46	5.44	5.40	5.33
Median Filter	21.19	20.78	20.40	19.55	18.14	16.61	18.05	17.98	17.93	17.91	17.81	17.69
Proposed Algorithm	30.30	27.52	25.02	23.10	21.15	19.19	18.00	18.04	17.98	17.91	17.74	17.56

ISSN: 1790-5109

ISBN: 978-960-474-108-3