Removing Salt-and-Pepper Noise in Text/Graphics Images

Krisana Chinnasarn

Faculty of Science, Burapha University, Bangsaen, Chonburi 20131, Thailand
E-mail: krisana@www.sci.buu.ac.th

Yuttapong Rangsanseri and Punya Thitimajshima

Department of Telecommunications Engineering, Faculty of Engineering
King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand
E-mail: kryuttha@kmitl.ac.th, ktpunya@kmitl.ac.th

Abstract

Document containing text and graphics components are usually acquired as binary images for computer processing purposes. Salt-and-pepper noise is a prevalent artifact in such images. Removing this noise usually requires iterative or multiple-pass processing, some techniques even cause distortions in document components. In this paper, we propose a novel method based on the kFill algorithm [1][2] that can be accomplished in single-pass scan over the image. The algorithm is capable to remove simultaneously both salt noise and pepper noise of any sizes that are smaller than the size of document objects. Results of the proposed method are given in comparison with the well-known morphological operations.

1. Introduction

The first step in document image analysis is to capture a paper document into a binary image, usually performed by optical scanning and by mean of thresholding [3]. The resulting image can be contaminated by salt-and-pepper noise during the conversion process and also from dirtiness in the document itself. This appears as randomly distributed black dots on a white background and white dots on a black document component. Each noise dot can be either an isolated pixel or composed of more than one pixel. Noise removal is a very important preprocessing to facilitate other subsequent operations. Isolated white and black pixels can be easily located by passing a 3 x 3 pixel window over the image, while noise larger than one pixel usually requires iterative or multiple-pass processing.

The techniques based on mathematical morphology [4] perform quite well in general for removing noise in binary images. In case of document images, however, they may significantly damage the sharpness of the text and graphics components. The kFill algorithm is designed specifically for document images, but has many restrictions in real applications. All these algorithms are achieved in several iterations over the entire image. In this paper, we present a novel technique developed from the kFill algorithm that can be accomplished in single-pass raster scan over the image.

2. Mathematical morphology

Mathematical morphology is a mathematics process for studying of image structure by using a Set theory [4]. Most of its operations, utilized for noise removal in binary images, are composed from two basic operations: *erosion* and *dilation*. The erosion operation, denoted by a mathematics notation +, is the reduction in size of ON regions, most simply performed by peeling off a single pixel from the outer boundary of all ON regions. Dilation is the opposite operation and denoted by a mathematics notation -. In each dilation step, ON value layer are added to boundaries to increase their size. To increase the performance of the mathematical morphology, we combined the two basic operations together, both are called *opening* and *closing*. *Opening*, composed by one or more iterations of erosion followed by the same number of iterations of dilation, has effect only on pepper noise (ON-pixel noise). The morphological dual of opening is *closing*, resulting just in salt noise (OFF-pixel noise) removal. For document images where salt-and-pepper noise is present, a cascade

0-7803-5146-0/98/$10.00 ©1998 IEEE.

of opening and closing operations may be necessary. This implies that at least four iterations over the entire image are required.

3. kFill algorithm

The kFill algorithm [1][2] is designed to reduce salt-and-pepper noise while maintaining readability. It take advantage of the known characteristics of the text and graphics components. It performs by moving a k x k pixel window over the image in the raster-scan direction. The window size (k) should be selected appropriate to the known size of noise to be removed. The process of noise removal is by filling the core, an inside $(k\text{-}2)^2$ region, with its opposite value. The decision whether or not to fill requires that all core pixels must be the same value and depends on three variables determined from the neighborhood, the surrounding $4(k\text{-}1)$ pixels. For a fill value equal to ON (OFF), the n variable is the number of ON (OFF) pixels, the c variable is the number of connected groups of ON (OFF) pixels, and the r variable is the number of corner pixels that are ON (OFF). Filling occurs when the following conditions are met:

$$(c = 1) \text{ AND } \{ (n > 3k\text{-}4) \text{ OR } [(n = 3k\text{-}4) \text{ AND } (r = 2)] \} \quad (1)$$

The conditions on n and r are set as functions of the window size k. The condition $c = 1$ ensures that filling does not change connectivity, i.e. does not join two image objects together, or separate two parts of the same connected object. Noise reduction is performed iteratively on the image until no filling occurs. Each iteration consists of two subiterations, one performing ON-fills, and the other OFF-fills.

4. Proposed algorithm

Most document image processing systems perform noise reduction by passing 3 x 3 pixel window size across the image to locate isolate ON and OFF pixel. In Fig. 1, we show a core region and neighbor pixels for the window size equal to 3 x 3 and 4 x 4 respectively. Closer inspection of many real document images, we found that salt-and-pepper noise components are frequently larger than one pixel. In such case, the window size larger than 3 x 3 pixel should be used, and the kFill algorithm will never fill the noise components smaller than the core size. Fig. 2 demonstrates this problem where an isolate pixel cannot be removed by using $k = 4$. In this example, the kFill algorithm will not fill the core region neither with ON or with OFF, because all core pixel are not same value. Therefore, we propose a modification of the kFill algorithm as shown in Fig. 3. In this algorithm, we fill the core with OFF when the majority of pixels are ON and the equation (1) is verified or when the majority of pixels are OFF and the equation (1) is not verified. The reciprocal conditions are also applied for the ON filling.

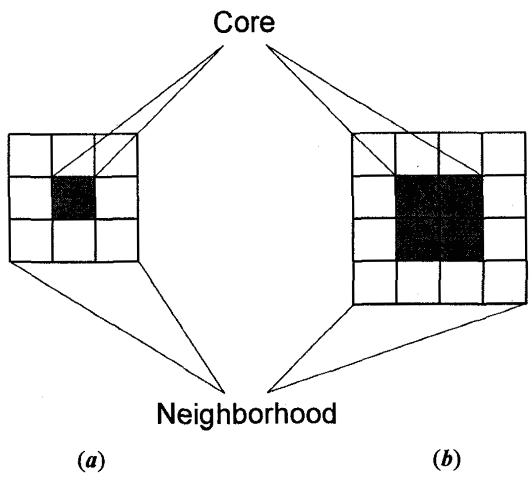

Figure 1 The core region and the neighborhood for the window size of k x k where k =3 and 4 in (*a*) and (*b*) respectively.

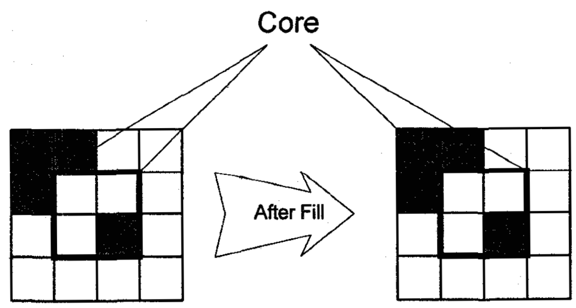

Figure 2 The isolated pixel cannot be removed by the kFILL algorithm with k = 4 or larger.

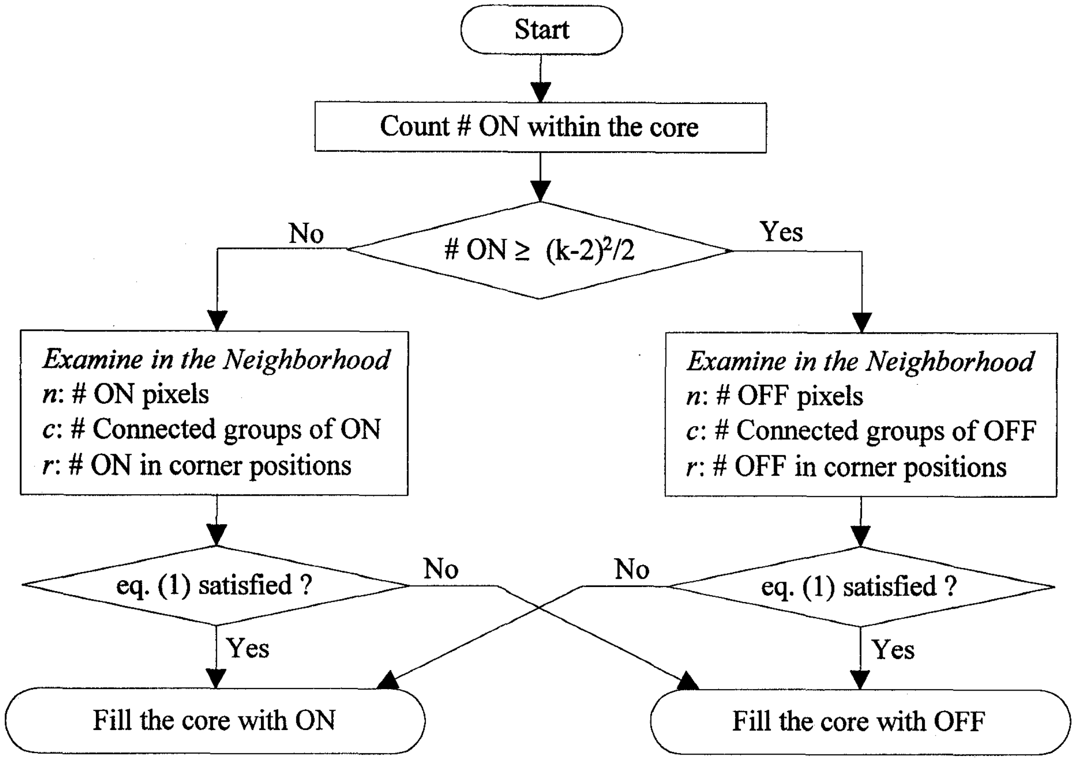

Figure 3 Flowchart of the proposed algorithm.

5. Experimental results

The proposed algorithm was tested on text images in comparison with morphological filters. An illustrative result is given in Fig. 4. The original image, shown in Fig. 4(*a*), is scanned with 300-dpi resolution. Salt-and-pepper noise of different size and shape is artificially introduced in Fig. 4(*b*) by using an image editor. Fig. 4(*c*) and (*d*) illustrate the use of morphological filters with 3 x 3 structuring element. Fig. 4(*c*) shows the result of opening followed by closing while Fig. 4(*d*) shows the result of closing followed by opening. In both cases, almost noise can be removed but the character sharpness has been significantly distorted. Fig. 4(*e*) and (*f*) show the results of the original kFill algorithm with only single iteration, when k equals 4 and 5 respectively. Finally, Fig. 4(*g*) and (*h*) show the results obtained from the proposed algorithm with the same values of k, 4 and 5 respectively. Comparing the results in the last two rows, it was clear that our algorithm outperforms the original kFill. On the other hand, the window size used in our algorithm is not critical, both results in Fig. 4(*e*) and (*f*) are quite similar, with the exception that few noise components still not removed with $k = 4$, in Fig. 4(*e*). Using $k = 5$, in Fig. 4(*f*), provides more promising results in general.

6. Conclusion

We have presented a novel algorithm for removing salt-and-pepper noise in text/graphics images. This algorithm is fast and effective. It can remove noise of different size and shape while maintaining the sharpness of the text and graphics components.

Figure 4 Results of noise removal algorithms (*a*) original image; (*b*) noisy image; (*c*), (*d*) filtered images using opening-closing and closing-opening; (*e*), (*f*) filtered images using single-iteration kFill algorithm with $k = 4$ and 5 (*g*), (*h*) filtered images using the proposed algorithm with $k = 4$ and 5 respectively.

References

[1] G.A. Story, L. O'Gorman and D. Fox, "The RightPages Image-Based Electronic Library for Alerting and Browsing", *Computer*, vol. 25, pp. 17-26, Sept. 1992.

[2] L. O'Gorman. "Image and Document Processing Techniques for the RightPages Electronic Library System", *Proc 11th IAPR Int' l Conf. Pattern Recognition, vol. II,* IEEE CS Press, Los Alamitos, Calif., pp. 260-263, 1992.

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

[4] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, New York, 1983.