https://doi.org/10.2352/ISSN.2470-1173.2017.18.COLOR-044

Edge-Preserving Error Diffusion for Multi-Toning Based on Dual Quantization

Takuma Kiyotomo*, Keisuke Hoshino**, Yuki Tsukano**, Hiroki Kibushi**, Takahiko Horiuchi*
****Graduate School of Advanced Integration Science, Chiba University, Chiba, Japan***
*****Research & Development Department, TOKYO KIKAI SEISAKUSHO, LTD, Chiba, Japan***

Abstract

Digital multi-toning is a technique for converting a continuous-tone image into a multi-tone image for its reproduction with a multi-tone output device. It is becoming important as printers now have the ability to print dots of different intensities. Error Diffusion (ED) is an algorithm that has been shown to be effective for the multi-toning process and has been widely applied to digital printing tasks. However, in the actual printing process, conventional ED techniques for digital multi-toning are often unable to print a sufficiently good-quality image because of physical or mechanical dot gain. In particular, distortion of the contour of printed letters is noticeable. Black letters against a white background appear enlarged, whereas white letters on a black background appear faded. In this paper, we propose an edge-preserving ED algorithm to improve the quality of printed images. We prepare different quantization thresholds between the edge and other regions to allow for stable detection of the edge regions. In addition, we propose a multi-step edge-detection algorithm to avoid printed artefacts. Experimental results using printed images showed that the proposed algorithm improved the quality of printed images in comparison with conventional techniques.

Introduction

Digital half-toning is one of the oldest applications of image processing since it is essential for the printing process. Digital half-toning simulates a grey-scale image by using only black and white tones. A variety of half-toning methods have been proposed [1]-[3], and are commonly used together with error diffusion (ED) [4]. Printers have greatly improved in recent years and are able to offer more than two quantization levels. Multi-toning is an extension of half-toning, in that it adopts more than three tone levels for improving the similarity between the original image and the converted image.

During the actual printing process, many kinds of noise (e.g., ink spread and ink bleed) are added to the image. Figure 1 shows examples of images that were printed using 2-bit ED and a newspaper printing machine (TKS® JETLEADER® 1500). Black letters on a white background appear enlarged, whereas white letters on a black background appear reduced in size and faded. This is a major concern for printing press experts for whom the quality of images is vital. Thus, reducing this adverse effect is of critical importance.

In this paper, we develop an edge-preserving ED algorithm for improving the quality of printed images. The design of our algorithm is based on the assumption that in the printing process the distorted areas are located at the edges. Therefore, we consider an edge-preserving approach to be effective for removing printed distortion. In this work, we prepared different quantization thresholds between the edge and other regions to enable stable detection of the edge regions. In addition, we propose a multi-step edge-detection algorithm to prevent printed artefacts from occurring.

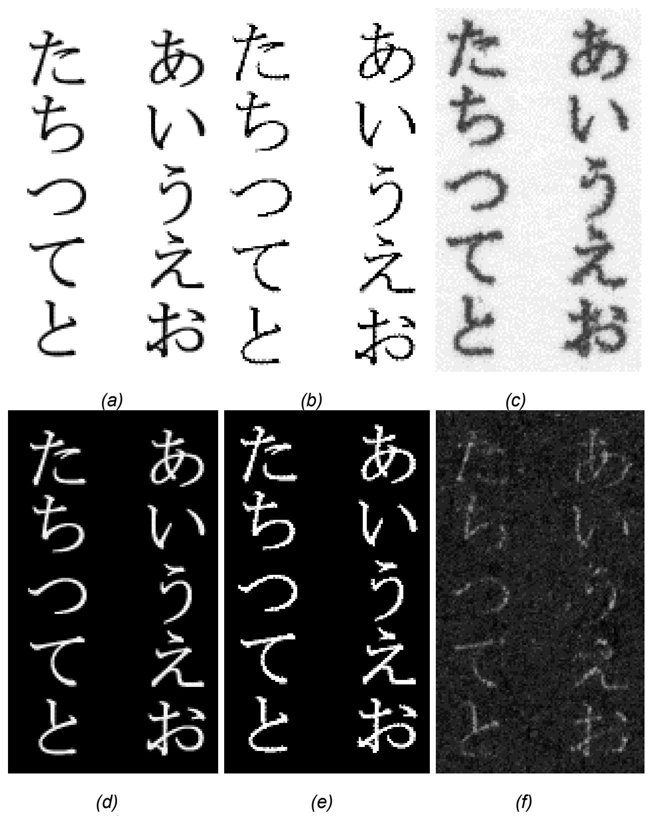


Figure 1. (a) Original image (black letters). (b) 2-bit quantized image by conventional ED. (c) Printed images at 600 dpi. (d) Original image (white letters). (e) 2-bit quantized image by conventional ED. (f) Printed images at 600 dpi. Each printed image was scanned at 2400 dpi

Related Work

ED is one of the most popular half-toning methods proposed by Floyd and Steinberg [4]. Error diffusion is known to be an algorithm that is simple and fast, and which produces good results. There are a variety of ED masks designed to enhance Floyd's method (e.g., by Stucki [5], and by Shiau and Fan [6]). Additionally, there are many other approaches to improving error diffusion. One of the simplest techniques for improving the half-toning quality is to change the scanning direction and error diffusion masks. This approach, alternately scanning left to right and right to left, is called serpentine scanning. There are also ways to reduce banding effects by adding random noise.

Some research has focused on quality loss at the edges due to the ED algorithm. Fung and Chan [7] and Li [8] used algorithms that focused on edges and led to edge-preserving error diffusion techniques. In these studies, edge information was used directly. In another approach to improve the quality of quantized images by using edge information, Shi and Li [9] and Lee et al. [10] focused on weak textures vanishing in the quantization process and proposed half-toning algorithms to preserve these weak textures using edges. Their methods were designed to prevent edge loss by ED. Thus, these methods do not mitigate the adverse effects caused by the printing process.

Because of increasing printer performance, a way to produce a quantized image with more than two quantization levels is critically needed. Multi-toned images were developed by Sarailidis and Latsavounidis [11], Guo et al. [12], and Fung and Chan [7], who modified the half-toning ED algorithm to include multi-toning. However, these previous studies neither evaluated actual printed images nor considered the adverse effect of expansion of black areas due to machine noise.

There are some half-toning algorithms using printer models. Pappas [13], Lee [14], and Lai [15] proposed half-toning algorithms that focused on dot gain and overlap of ink. Yampolskiy et al. [16] proposed a special half-toning algorithm based on the genetic algorithm. Their methods were proposed for natural images, and these methods mainly improve tone reproduction. Therefore, their goals were different from ours.

Proposed Algorithm

Multi-toning algorithm based on error diffusion

First, we modify the ED technique from half-toning to multi-toning. X, Y, and E represent an 8-bit input image, the quantized output image, and the error image, respectively. $X(i,j)$ denotes the (i,j)-th element of the image matrix. We quantize the pixels from left to right, in descending order as

$$Y(i,j) \Leftarrow Q(X(i,j+E(i,j))) \tag{1}$$

where $Q(x)$ is a quantization function, which we altered to achieve multi-toning. For half-toning, this function is

$$Q_2(x) = \begin{cases} 0 & x < 128 \\ 255 & 128 \le x \end{cases} \tag{2}$$

For 2-bit multi-toning, the quantization function becomes

$$Q_4(x) = \begin{cases} 0 & x < 64 \\ 85 & 64 \le x < 128 \\ 170 & 128 \le x < 192 \\ 255 & 192 \le x \end{cases} \tag{3}$$

By replacing the quantized function, we are able to easily to move from half-toning to multi-toning. Then we can calculate the quantization error as

$$e \Leftarrow X(i,j) + E(i,j) - Y(i,j) \tag{4}$$

The error is diffused by applying the mask **M** to neighborhood pixels. In our study, we use Shiau and Fan's mask:

$$\mathbf{M} = \frac{1}{16}\begin{bmatrix} 0 & 0 & 0 & -16 & 8 \\ 1 & 1 & 2 & 4 & 0 \end{bmatrix} \tag{5}$$

$$E[i-1,i,i+1; j-1,j,j+1] \Leftarrow E[i-1,i,i+1; j-1,j,j+1] + e \times M \tag{6}$$

Upon renewing the error image E, we apply the procedure successively to the next pixels.

Edge detection

In principle, the ED algorithm explained in the previous section is intended for use when unevenness occurs at the edge. In this paper, as a pre-processing step, we detect edge information from the original image and preserve edge information through the ED.

There are a number of methods to detect edges, and all have their distinctive characteristics. The most popular edge detecting algorithms are Sobel's [17] Sobel operator and Canny's [18] Canny operator. Another method is a high quality edge-detecting algorithm using circular filters proposed by Susan [19]. In this paper, we detect edges using the Sobel operator. The method calculates edge strength faster than other edge detecting methods. Figure 2 illustrates a typical edge-detection process. $I_{Eh}(i,j)$, and $I_{Ev}(i,j)$ represent edge images calculated by the Sobel operator designed for horizontal direction edges and vertical direction edges, respectively. In general, the edge image has positive or negative values according to its direction. To achieve edge information that has normalized positive values and all edge directions, we calculate $I_{Ehv}(i,j)$ as

$$I_{Ehv}(i,j) \Leftarrow \begin{cases} 1 & 255 < \sqrt{I_{Eh}(i,j)^2 + I_{Ev}(i,j)^2} \\ 0 & otherwise \end{cases} \tag{7}$$

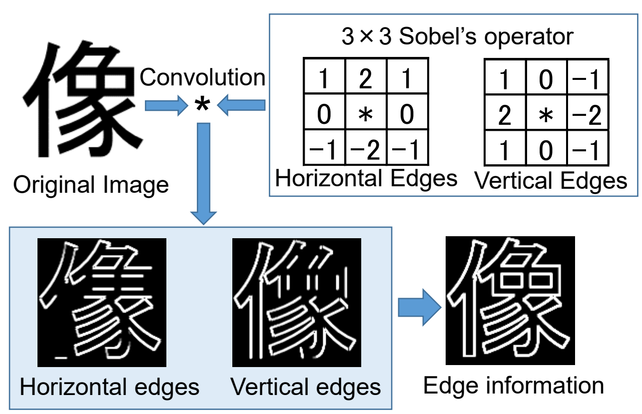


Figure 2. Outline of edge detection using Sobel operator

Edge Selection

Edge information detected by the Sobel operator is computed quickly, but with low precision. We confirmed that the operator often generated pseudo-edges on the input image, and they caused extensive artefacts on the multi-toned result. For this reason, we remove the edges that cause unwanted artefacts. We named this process "Edge Selection". Figure 3 shows the outline of our edge-selection process.

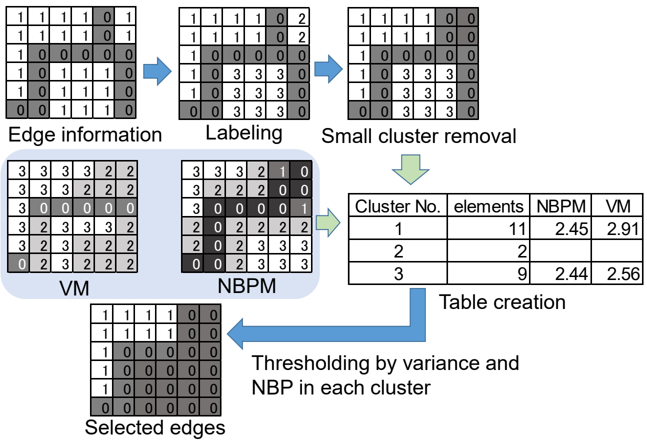


Figure 3. Outline of the poposed edge selection process

In this process, we consider adjacent pixels as one cluster, and remove clusters comprising a small number of elements as noise. Figure 4 illustrates the outline of our clustering algorithm. We can easily implement this algorithm using a recursive algorithm. This algorithm contributes gives accurate clustering results but uses a lot of computing resources. In the case of limited computer resources, we recommend setting an upper limit on cluster size.

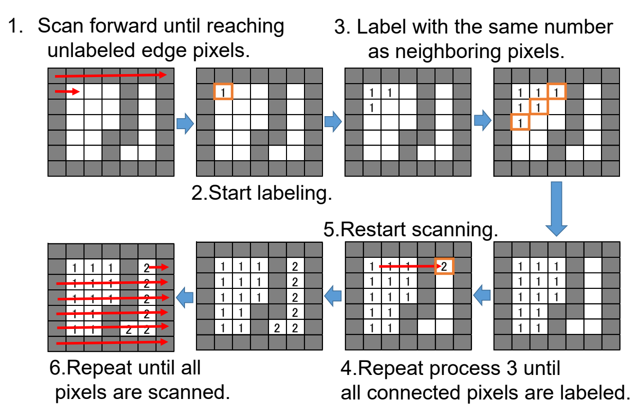


Figure 4. Outline of the proposed clustering algorithm

In addition, we use the detected edges as candidate edges from which final edges are detected based on statistical characteristics. First, we construct the variance map (VM), calculated as the variance of neighborhood pixels. We can easily obtain the VM using a box filter. Second, we configure the number of black pixels map (NBPM) by counting the neighborhood black pixels on the binarized input image. NBPM contains information of marginal density of black pixels. We binarize the original image by simple thresholding to distinguish black pixels. Then we can configure NBPM by applying a box filter to the binarized image.

We next select "useful" edges from the edge information $I_{Ehv}(i,j)$ using VM and NBPM. We calculate the average value of NBPM and VM for each cluster of edges and threshold edge clusters using this averaged value. New edge information $I_E(i,j)$ is removed using extra information (i.e. matter of extensive artefacts) from the old edge information.

Algorithm 1 presents the pseudo-code for this edge selection process. $NPBM(i,j)$ is an image representation of NPBM and $VM(i,j)$ is that of VM. $f_{mean}(A)$ denotes the process of applying the box filter to image A. The operation ".*" indicates element-wise multiplication; $thresholding(X|a)$ refers to image thresholding on image X with threshold value a.

Algorithm 1 Pseudo-code of edge selection

$I_{Ehv} \Leftarrow SobelEdgeDetection(X)$
$mean \Leftarrow f_{mean}(X)$
$corr \Leftarrow f_{mean}(X.*X)$
$VM \Leftarrow corr - mean * mean$
$BM \Leftarrow thresholding(X|thr_b)$
$NBPM \Leftarrow fmean(BM)$
$I_E \Leftarrow EdgeSelection(I_{Ehv}, NPBM, VM)$

Edge preserving error diffusion

In the error diffusion process, we quantize the edges using a quantization function that differs from the function used for other areas. There are two significant characteristics of our method. First, we exclude quantized error diffused by neighborhood pixels. Second, we use a special quantization function at the edge pixels. We name this process "Dual Quantization" (DQ). Figure 5 illustrates our multi-toning process. Detected edge information I_E is expressed as a matrix with the same size as that of the input image. The component $I_E(i,j) \in \{0,1\}$ is assigned a value of 1 when the pixel (i,j) is located at the edge. Then the DQ is performed as

$$Y(i,j) \Leftarrow \begin{cases} Q_E(X(i,j)) & I_E(i,j)=1 \\ Q(X(i,j)+E(i,j)) & I_E(i,j)=0 \end{cases} \tag{8}$$

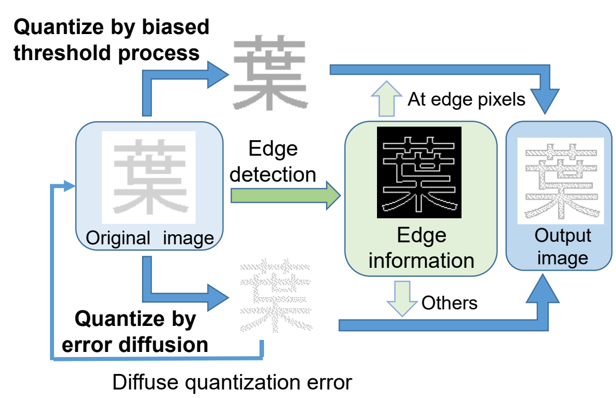
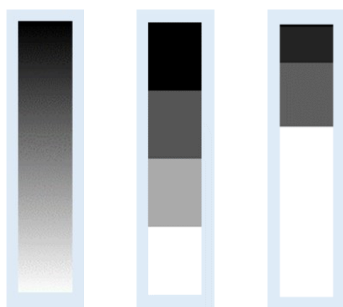


Figure 5. Outline of the proposed multi-toning process based on DQ

The quantization function Q_E is a biased threshold function. As shown in Fig. 6, Q_E assigns a wider range of high-concentration values than dark-concentration values. At the same time, we quantize pixel values by excluding errors at the edges. By using this biased thresholding and error exclusion process, a pixel has a high probability of constantly being assigned a bright value at the edge. Users have to design this biased quantization function based on attributes of their printer. In the simplest case, we determine this value from the budget of ink drops for each quantization level. We need to consider ink bleed. At the same time, we need to consider unwanted dot patterns due to this edge exclusion process at the edges. The edge exclusion process results in loss of characteristics of error diffusion masks. Therefore, we retain features of the ED mask by calculating the quantization error *e* as follows:

$$e \Leftarrow X(i,j) + E(i,j) - Q(X(i,j) + E(i,j)) \tag{9}$$



(a) (b) (c)

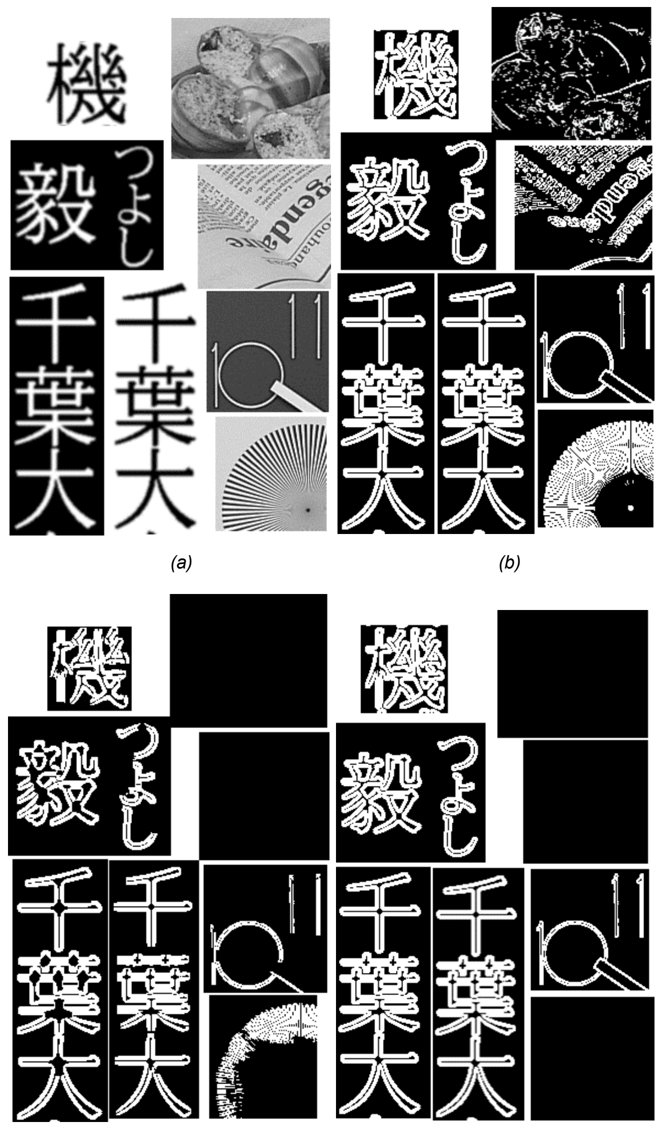
Figure 6. (a).Original image gradation (8-bit), (b) Quantized by uniform threshold (2-bit), (c)Quantized by biased threshold (2-bit)

Experimental Result

We tested our algorithm using natural images (ISO/JIS-SCID, and SHIPP) and some printed text obtained from actual newspapers as character images. Natural images have color information, but newspaper usually uses gray scale images. Therefore, we converted color images to gray scale images and used the gray scale images for our experiments. In the same manner, we used gray scale character images in this paper. Figure 7 shows a selection of the images we tested.

Figure 7. A selection of test images

Figure 8 shows the results of the edge detection process. As shown in Fig. 8(b), all of the edges of letters were correctly detected. Figures 8(c) and (d) show the result of edge selection using pixel-based and cluster-based information, respectively. By comparing these with Fig. 8(b), inappropriate edges that might cause unwanted artefacts were removed from the natural images. In Fig. 8(c), unnecessary edges remain. Unwanted artefacts in the quantized and printed images may occur if using this edge information. These results show the importance of the clustering process. We show artefacts due to unselected edges later.



(a) (b)

(c) (d)

Figure.8 Results of edge detection. (a).Original images. (b).Results of edge detection process. (c).Results of edge selection based on pixel information. (d).Results of edge selection based on cluster information

To test the performance of our methods, we implemented the all of the proposed algorithms and actually printed each of the test images using existing methods and our method. Figure 9 shows partial results using the proposed multi-toning approach based on

DQ and the multi-toning method of Shiau and Fan. We printed the results using a newspaper printing machine (TKS® JETLEADER® 1500 with 600 dpi). Figure 10 shows these printed images scanned at a resolution of 2400 dpi. As evident in the figures, the result of our algorithm has smoother edges and a wider white area than the result produced by the existing algorithm. In addition, our method produces thinner black characters than the existing method. Furthermore, our methods reduce the loss of white area caused by printer machine noise. Therefore, the sharpness of both the black and white letters is improved. This result demonstrates the effectiveness of the proposed DQ algorithm.

(a)

(b)

(c)

Figure.9 Partial results of image quantization in test images. (a).Original images (8-bit). (b). Printed image using extended Shiau and Fan's ED (2-bit). (c). Printed image using our algorithm (2-bit)

To demonstrate the importance of the edge selection process, we printed the results in the absence of this process. Figure 11 shows the printing result quantized by using selected edges and unselected edges. The result of multi-toning using unselected edges includes some artefacts. These results confirm the effectiveness of edge selection.

We implemented our algorithm and conventional methods and measured the computing time. Table 1 lists specifications of our system. In this paper, we exclude processing time for I/O communication from computing time. Table 2 lists computing time measured by each method. We used size 2048 × 2560 natural images and size 2339×1654 character images in the experiments. Note that our algorithm depends strongly on the number of clusters and its elements in the process of edge selection, so computing time varies widely in response to the input images.

Our method requires more computing time than conventional methods due to the edge detection and edge selection processes. However, our methods are composed of simple operations such as box filtering and calculating each pixel. We can easily implement our algorithm on the GPU, and can speed up this process. In addition, users can keep computing time within desired limits by setting appropriate limits of clustering size.

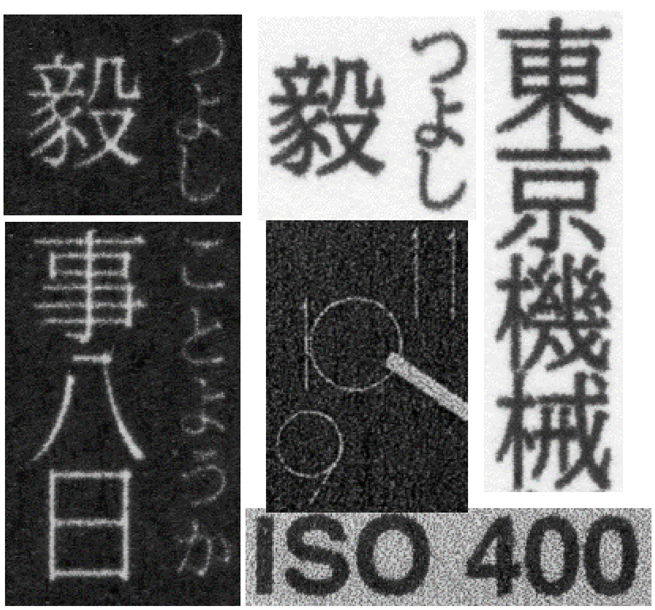
Table 1. Specifications of our system.

OS	Windows10 64bit
CPU	Intel® Core i5-4590 3.30GHz
RAM	8.00[GB]
Programming language	C Language
Compiler	gcc (ver.5.4.0)

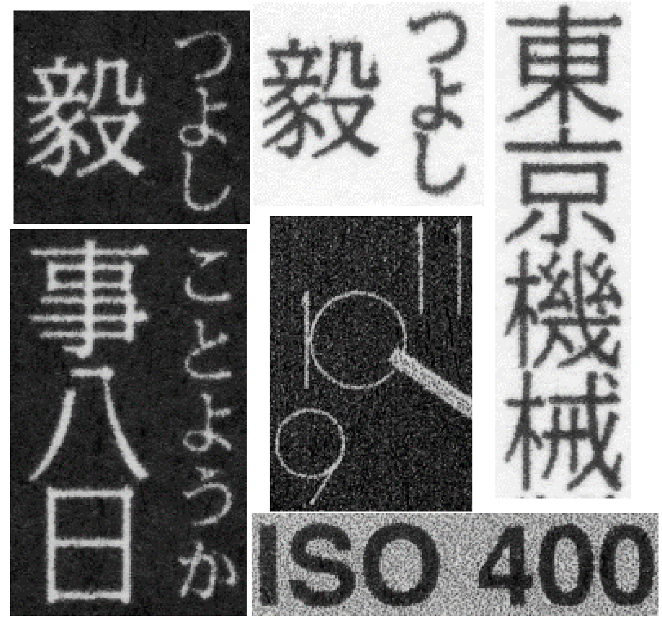
Table 2. Computing times

	Natural image [sec]	Character imaga [sec]
Floyd & Steinberg's (Half-toning)	0.67	0.48
Shiau&Fan's (Multi-toning, 2bit)	0.77	0.56
Our method (Multi-toning, 2bit)	1.97 (Edge Detection:1.19)	1.28 (Edge Detection:0.72)

(a)

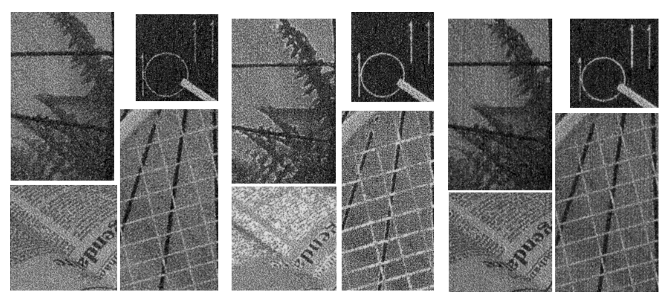


(b)



(c)

Figure 10. Printed results of Fig. 9. (a).Original images (8-bit). (b). Printed image using extended Shiau and Fan's ED (2-bit). (c). Printed image using our algorithm (2-bit)



(a) (b) (c)

Figure 11. Comparison of artefacts. (a). Printed image using previous ED method. (b). Printed image using unselected edge information. (c). Printed image using selected edge information

Conclusion

We proposed a multi-toning method that improves the quality of printed images by edge preserving error diffusion. Our method uses two characteristic approaches. One is to select edge information using variance and the number of black pixels calculated from neighborhood pixels. This process prevents the appearance of unwanted artefacts in quantized images. Another is to exclude error values when quantizing the edge pixels. In addition, at the edges we use biased thresholds, which are designed based on printer characteristics. By excluding quantized error and using biased thresholds at the edges, we improved the sharpness and reduced the loss of image quality caused by machine noise. We applied our methods to gray scale natural images and character images and confirmed their effectiveness. As a result, we produced images with reduced loss of white area caused by printer noise. Additionally, we removed artefacts at the edges using edge selection. Therefore, we were able to obtain high quality printed images.

In future work, we will extend our methods to color images. The simplest approach that modifies gray scale multi-toning to color images is to decompose the original image to CMYK images and apply the same multi-toning method to each channel. In addition, we will improve the labeling algorithm to limit computing costs.

Acknowledgments

The authors would like to thank Dr. Hiroaki Kotera at Kotera Imaging Laboratory and Dr. Ryoichi Saito at Chiba University for many suggestions, and Toshiya Sato and Hiroki Sezaki at Tokyo Kikai Seisakusho, Ltd. for useful discussions.

References

[1] R. A. Ulichney, "Digital halftoning", MIT Press, Cambridge, MA, 1987

[2] B. Bayer, "An optimum method for two-level rendition of continuous-tone pictures.", IEEE International Conference on Communications, pp. 11-15, 1973

[3] O. Deussen, S. Hiller, C. Overveld, and T. Strothotte, "Floating points: A method for computing stipple drawings", Computer Graphics Forum 19, vol. 3, pp. 40-51, 2000

[4] R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial greyscale", Proc. Soc. Inf. Display, pp. 36–37, 1975.

[5] P. Stucki, "MECCA-a multiple-error correcting computation algorithm for bilevel image hardcopy reproduction", IBM Res. Lab., Zurich, Switzerland, Res. Rep. RZ1060, 1981.

[6] J. Shiau and Z. Fan, "A set of easily implementable coefficients in error diffusion with reduced worm artifacts", Proc. SPIE, 2658, pp.222-225, 1996.

[7] Y.H. Fung and Y.H. Chan, "Edge preserving multiscale error diffusion algorithm for green noise digital halftoning", IEEE ICIP, pp. 557-560, 2010.

[8] X. Li, "Edge-directed error diffusion halftoning" ,IEEE Signal Processing Letters, Vol.13, No.11, pp. 557-560, 2010.

[9] S. Shi and X. Li, "An improvement error diffusion algorithm based on visual difference", IEEE, ICIP, 2014

[10] H. S. Lee, K. K. Kong, and K. S. Hong, "Laplacian based structure-aware error diffusion", IEEE, ICIP, pp. 525-528, 2010

[11] G. Sarailidis and I. Latsavounidis, "A multiscale error diffusion technique for digital multitoning", IEEE Trans of Image Processing, Vol.21, No.5, pp. 2697-2705, 2012.

[12] J. M. Guo, J. Y. Chang, Y. F. Liu, G. H. Lai, and J. D. Lee, "Tone-replacement error diffusion for multitoning", IEEE Trans of Image Processing, Vol.24, Issue.11, pp. 4312-4321, 2015.

[13] T. N. Pappas, J. P. Allebach, and D. L. Neuhoff, "Model-based digital halftoning", IEEE Trans of Image Processing, Vol.20, Issue.5, pp. 14-27, 2003

[14] J. H. Lee and J. P. Allebach, "Inkjet printer model-based halftoning", IEEE Trans of Image Processing, Vol. 14, Issue.5, pp.674-689

[15] J. Z. H. Lai and C. C. Chen, "Color image halftoning with the dot overlap printer model", IEEE, ICIP, pp. 333-337, 1999

[16] R. V. Yampolskiy, P. G. Anderson, J. S. Arney, V. Misic, and T. R. H. Clarke, "Printer model integrating genetic algorithm for improvement of halftone patterns", IEEE Signal Processing Society, Rochester, 2004

[17] D. Shaked, N. Arad, A. Fitzhugh, and I. Sobel, HP labs technical reports, HP Laboratories Israel, HPL-96-128(R.1), 1999

[18] J. Canny, "A computational approach to edge detection", IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986.

[19] S. M. Smith and J. M. Brady, "SUSAN—a new approach to low level image processing", International Journal of Computer Vision, vol.23, pp.45-78, 1997

Author Biography

Takuma Kiyotomo received the B.E. degree from Chiba University in 2016. He is currently a master course student in Chiba University. He is interested in quality improvement of printed images.

.