AN ERROR DIFFUSION TECHNIQUE WITH REDUCED DIRECTIONAL HYSTERESIS

Yuk-Hee Chan, H.T. Kwong*, K.T. Lo and C.K. Li

Centre of Digital Signal Processing for Multimedia Applications
Department of Electronic and Information Engineering
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong

*Communication and Advanced Consumer Technology Division
Motorola Semiconductors HK Ltd
Silicon Harbour Centre, Hong Kong

ABSTRACT

In this paper, a digital halftoning method is proposed to diffuse error with a more symmetric error distribution by making use of the concept of delayed decision. It can improve the diffusion performance by effectively reducing the directional hysteresis. A corresponding dot-overlap compensation scheme is also proposed to eliminate the bias in the gray scale of the printed images.

1. INTRODUCTION

Digital halftoning is a method that uses bilevel pixels (black and white pixels) to simulate a gray-scale image on a bilevel output device. Among the many available schemes we have nowadays, error diffusion[1] is believed to be one of the most effective approaches which can provide the best quality. However, directional hysteresis is unavoidable in conventional error diffusion schemes since sequential predetermined order is required to diffuse the quantization error. Numerous techniques have been developed to alleviate the problem. The simplest one might be reversing the direction of processing every scanline, but it just covers the problem without solving it. Some recently proposed schemes can reduce or even completely solve the problem. However, since they are typically either of iterative [2] or frame-oriented [3, 4, 5] nature, they may not be practical for real-time applications.

In this letter, two contributions will be made. First, a digital halftoning method eligible for real-time applications will be presented to diffuse error with a more symmetric error distribution by making use of the concept of delayed decision[6]. Second, we shall introduce a corresponding compensation scheme to remove the dot-overlap distortion when the algorithm is used to generate printer output.

This work was jointly supported by The Hong Kong Polytechnic University and Motorola Semiconductors HK Ltd (Project No. 0759.000.00.420.).

2. ALGORITHM

Consider we want to apply digital halftoning to a gray-level input image **F** whose values are within [0,1] to obtain an output binary image **B**. Let $f_{m,n}$ and $b_{m,n}$ denote the input and output at pixel (m,n), respectively. In the proposed scheme, we proceed with a raster scan strategy. Without loss of generality, consider we are now processing pixel (p,q). As raster scan strategy is exploited, all scanned pixels were processed and their output $b_{m,n}$'s were determined.

In order to determine $b_{p,q}$, we first assume the ordered sequence of the outputs of the next L pixels to be determined including $b_{p,q}$, say $\vec{s}$, to be one of the elements of $S = \{\{a_1, a_2 \cdots a_L\} | a_i \in \{0,1\}\}$.

After making an assumption, a compensation step is performed to compensate for the dot-overlap effect to be introduced in printers. Here, we assume that an ideal pixel occupies an unity square and an ink dot is circular. When a circular dot is put on pixel (p,n), some area of (p,n)'s neighbors will also be covered by the dot as shown in figure 1. Let $c_{p,n}$ be the newly covered area introduced by printing a circular ink dot at pixel (p,n) according to $b_{p,n}$. The value of $c_{p,n}$ can be defined as the output of a vector to scalar mapping of $\{b_{p-1,n-1}, b_{p-1,n}, b_{p-1,n+1}, b_{p,n-1}, b_{p,n}\}$ as given in Table 1. After performing the compensation, the error at pixel (p,n) is modulated to be

$$e_{p,n} = f_{p,n} - c_{p,n} + d_{p,n} \quad for \quad q+L > n >= q \quad (1)$$

0-7803-5471-0/99/$10.00©1999 IEEE

where

$$d_{p,n} = \sum_{i=1}^{W} \sum_{j=-W}^{W} h_{i,j} e'_{p-i,n-j} \qquad (2)$$

is the error diffused from the past scanlines, $W \times (2W+1)$ is the size of the region of interest, $h_{i,j}$'s are the filter weights of the corresponding diffusion filter and $e'_{m,n}$ is the error item at pixel (m,n) after carrying out a lateral diffusion to be discussed.

The lateral diffusion is performed among the $e_{p,n}$'s in the concerned segment of the line currently processed. This lateral diffusion is helpful to reduce the directional hysteresis. The error item at pixel (p,n) after the lateral diffusion is given as $e'_{p,n} = \sum_{j=-N}^{N} w_j e_{p,n+j}$, where $2N+1$ is the window size and w_j's are the filter weights of the non-casual lateral filter, and the cumulative distortion of the segment for the assumed future output pattern is calculated as

$$D_{b_{p,q}, b_{p,q+1} \cdots b_{p,q+L-1}} = \sum_{j=q-N}^{q+N} {e'_{p,j}}^2 \qquad (3)$$

For each possible assumption of $\vec{s} \in S$, we evaluate its corresponding $D_{\vec{s}}$ and the final output $b_{p,q}$ is determined with

$$b_{p,q} = \begin{cases} 1 & if \ \sum_{\vec{s} \in S_0} D_{\vec{s}} > \sum_{\vec{s} \in S_1} D_{\vec{s}} \\ 0 & if \ \sum_{\vec{s} \in S_0} D_{\vec{s}} < \sum_{\vec{s} \in S_1} D_{\vec{s}} \\ rand() & if \ \sum_{\vec{s} \in S_0} D_{\vec{s}} = \sum_{\vec{s} \in S_1} D_{\vec{s}} \end{cases} \qquad (4)$$

where $rand()$ is a function whose output is either 0 or 1 by random, $S_0 = \{\{0, a_2 \cdots a_L\} | a_i \in \{0,1\}\}$ and $S_1 = S \setminus S_0$.

After a decision has been made, we proceed one pixel and carry out the procedures of distortion calculation and decision for the next pixel $b_{p,q+1}$. These steps are repeated until the entire image is processed.

In practice, we let $L = 2N+1$ to minimize the steps we look ahead in order to avoid unnecessary computation and time overhead to reach a decision. Note most of the computation result obtained during the determination of $b_{p,q}$ can be reused to determine $b_{p,q+1}$. Hence, the computational effort required is much less than it appears to be.

In general, to determine a new output pixel $b_{p,q}$, a new input pixel $f_{p,q+L-1}$ will be taken into account. One has to take care of all 2^L possible output sequences in S. As there are only 2 possible non-zero values of $c_{p,q+L-1}$ when the output of the past scanline is determined, only 3 additions are required to determine all possible values of $e_{p,q+L-1}$. The computation of $d_{p,n}$ involves past scanlines only, which takes $(2W+1)W$ multiplications and $(2W+1)W-1$ additions. For each of the 2^L possible cases, it takes $2N+1$ multiplications and $2N$ additions to compute $e'_{p,q+L-1}$. The realization of eqn. (3) totally takes 2^L multiplications and $3 \times 2^{L-1}$ additions for all cases. Finally, the output $b_{p,q}$ is determined with $2^L - 2$ additions and 1 comparison with (4). In short, $(L+1)2^L + (2W+1)W$ multiplications, $(2L+3)2^{L-1} + (2W+1)W$ additions and 1 comparison are required to determine one output pixel. The effort can be further reduced by nearly half if the filters exploited are symmetric.

3. SIMULATION RESULTS

Simulations have been carried out to evaluate the performance of the proposed algorithm. Figure 3 shows some simulation results obtained with a 256 × 256 point light source pattern shown in Figure 2. The diffusion and lateral filters exploited in our method are, respectively, $\{h_{1,-1}, h_{1,0}, h_{1,1}\} = \{0.2, 0.6, 0.2\}$ and $\{w_{-1}, w_0, w_1\} = \{1/3, 1/3, 1/3\}$. This implies $L = 3$ and $W = N = 1$. The filters presented in [5] were used to simulate [1] and [5]. In simulating [2], a 3 × 3 filter given as $\frac{1}{32}\begin{bmatrix} 3 & 5 & 3 \\ 5 & 0 & 5 \\ 3 & 5 & 3 \end{bmatrix}$ was used and the iteration was initialized with the error image obtained with [1]. Table 2 shows the realization effort of these methods for comparison.

We deliberately printed the figures using a 600dpi laser printer so that the individual dots can be clearly printed and the effect of dot overlapping is not dominant. No compensation was done to get Fig.3d. One can clearly see the pattern noise in Figures 3b and 3c while hardly find any in Figures 3a and 3d. Figures 3e and 3f show the effect of dot-overlap compensation. A printer model of χ=1.0, which is defined as the ratio of the actual dot radius to the ideal dot radius $T/\sqrt{2}$, was used for compensation and generating the printed outputs. The connection among χ, α, β, and γ is given in [7]. Fig.3e shows the actual printed output of Fig.3a, which is purposely shown here for reference to show how the dot-overlap effect darkens a diffused output and the effect of the proposed compensation scheme.

4. CONCLUSIONS

In this paper, a digital halftoning method is proposed to diffuse quantization error with a more symmetric error distribution by making use of the concept of delayed decision. A corresponding dot-overlap compensation scheme is also proposed to eliminate the bias in the gray scale of the printed images. Simulation results show that the proposed method can effectively compensate for a printer's dot-overlap effect and provide

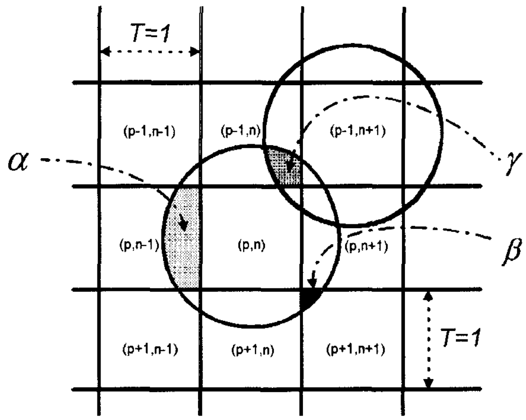

Figure 1: Circular dot-overlap model

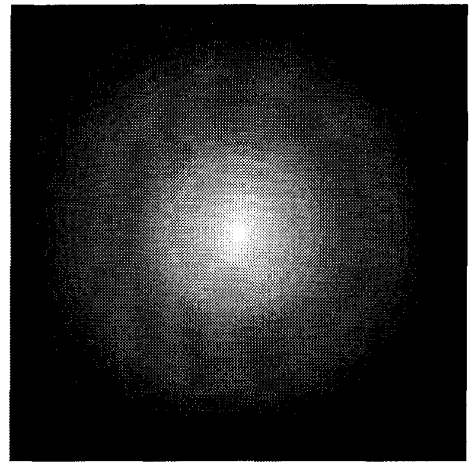

Figure 2: Original test pattern

a better diffusion result as compared with some other diffusion methods which are dedicated for reducing the directional hysteresis.

5. REFERENCES

[1] R. W. Floyd and L. Steinberg, "An Adaptive Algorithm for Spatial Gray Scale," Proc. SID International Symposium Digest of Technical Papers, Society for Information Displays, 1975, pp.36-37.

[2] A. Makur and A. Kumar, "Iterative error diffusion halftoning using a zero phase error filter," Proc., IEEE ICIP'97, Vol.1, Santa Barbara, CA, USA, pp.795-798.

[3] Y.H.Chan, "A modified multiscale error diffusion technique for digital halftoning," IEEE Signal Processing Letters, Vol.5, No.11, Nov 1998, pp.277-280.

[4] I. Katsavounidis and C.C. J. Kuo, "A multiscale error diffusion technique for digital halftoning," IEEE trans. on image processing, Vol.6, No., Mar 1997, pp.483-490.

[5] Z. Fan, "Error diffusion with a more symmetric error distribution," Proc. of SPIE, Vol.2179, 1994, San Jose, CA, USA, pp.150-158.

[6] P.W.Wong, "Error diffusion with delayed decision," Proc. of SPIE, Vol.2411, 1955, San Jose, CA, USA, pp.198-206.

[7] Y. Lin and T.C. Ko, "A modified model-based error diffusion," IEEE Signal Processing Letters, Vol. 4, No.2, Feb 1997, pp.36-38.

Table 1: Area to be covered by a new dot

Input pattern ψ^*	$c_{p,n}$
00001	$1 + 4\alpha + 4\beta$
00011	$1 + 2\alpha$
00101	$1 + 4\alpha + 2\beta - 2\gamma$
00111	$1 + 2\alpha + 2\beta - 2\gamma$
01001	$1 + 2\alpha$
01011	$1 + \gamma - \beta$
01101	$1 + 2\alpha + \beta - \gamma$
X1111	1
10001	$1 + 4\alpha + 2\beta - 2\gamma$
10011	$1 + 2\alpha + \beta - \gamma$
10101	$1 + 4\alpha - 4\gamma$
10111	$1 + 2\alpha - \beta - 3\gamma$
11001	$1 + 2\alpha + \beta - \gamma$
11011	$1 - \beta + \gamma$
11101	$1 + 2\alpha + 2\beta - 2\gamma$
XXXX0	0

$^*\psi = \{b_{p-1,n-1}, b_{p-1,n}, b_{p-1,n+1}, b_{p,n-1}, b_{p,n}\}$
$^\sharp$ X is don't care.

Table 2: Number of operations required for generating an output pixel with different methods

	[1]	[5]	ours	[2]*
MUL	4	6	35	8 × No. of iterations
ADD	5	10	39	9 × No. of iterations
CMP	1	1	1	1 × No. of iterations

* Extra computation is required for initialization.

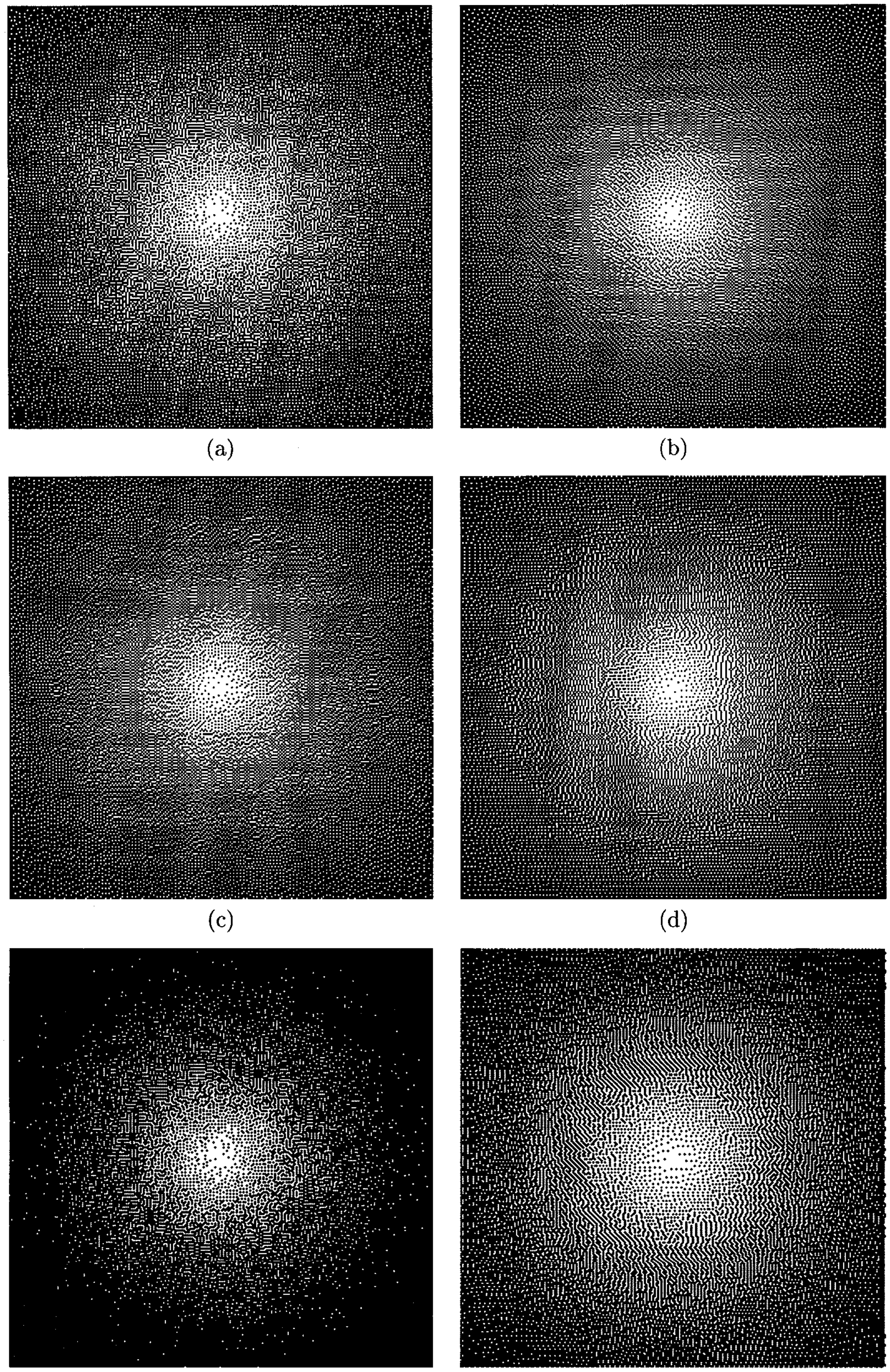

Figure 3: (a): Diffusion result of Makur & Kumar's method[2] after 10 iterations; (b): Diffusion result of standard error diffusion[1]; (c): Diffusion result of Fan's method[5]; (d): Diffusion result of the proposed scheme without printer compensation; (e): Actual printed output of Fig.3a; (f): Actual printed output of the result of the proposed algorithm with printer compensation.