

Error Diffusion and Edge Enhancement: Raster Versus Omni-Directional Processing

J. S. Arney,[▲] P. G. Anderson,[▲] and Sunadi Gunawan

Rochester Institute of Technology, Rochester, New York

Error diffusion is a well known technique for generating bi-level images and is often used instead of a halftone screen process in order to minimize the visual impact of quantization error. In addition, some kernels used for error diffusion are capable of enhancing the appearance of edge sharpness. The current study examined experimentally the effect of kernel type on the magnitude and the directional symmetry of edge enhancement. In addition, an alternative to raster image processing, involving a linear algorithm called linear pixel shuffling, was used to perform error diffusion. This allows the use of symmetrical kernels and the omni-directional diffusion of quantization error. Edge enhancement with omni-directional error diffusion was examined and found to be capable of more directionally uniform enhancement of edges than is possible with raster error diffusion.

Journal of Imaging Science and Technology 46: 359–364 (2002)

Introduction

In 1976, Floyd and Steinberg described the now familiar process of error diffusion as an alternative to halftone screening for simulating tone in bi-level images.^{1,2} The Floyd–Steinberg algorithm samples each location in an original gray scale image, compares the reflectance value, R , to a threshold value, T , and assigns a bi-level value of $m = 0$ or $m = 1$ to the output image. The resulting error, $E = (R - m)$, is distributed to neighboring, not-yet-quantized locations. If error diffusion is applied in a left-to-right raster, the error is distributed to the right, downward, and to the left in fractions defined by a kernel such as illustrated in Fig. 1. By diffusing the error to neighboring pixels, the output bi-level image simulates gray scale similarly to a halftone screen process. In addition, the diffusion process re-distributes distributes noise power, and by selecting an appropriate diffusion kernel the visual noise in the image can be reduced significantly. Moreover, the error diffusion process can enhance the visual sharpness of edges.^{3–5} Thus, the design of diffusion kernels and modifications to the Floyd–Steinberg algorithm have been active areas of research over the past two decades.^{1,2} In this report we explore experimentally the influence of the diffusion kernel on the magnitude and symmetry of error diffusion. We also describe a modification to error diffusion that replaces the usual raster sequence of processing with a pseudo-random sequence based on linear pixel shuffling.⁶ This allows error to be diffused with an omni-directional kernel. As will be shown, omni-directional error diffusion improves the directional symmetry of edge enhancement.

Edge Enhancement and Edge Amplitude

Error diffusion enhances edges by overshooting the high reflectance part of an edge and/or undershooting the low part of the edge. This is illustrated in Figs. 1 and 2 for a Floyd–Steinberg type algorithm with raster processing from left to right along each row, processing rows sequentially from top to bottom. The amount of edge enhancement can be measured experimentally by averaging the pixel gray values in each column of the bi-level image and plotting versus horizontal location as illustrated in Fig. 2. The ordinate axis R in Fig. 2 is pseudo-reflectance defined as $R = P/255$ where P is the column average pixel value 0 to 255. Edge metrics can be defined by using the four reflectance values indicated in Fig. 2. R_{HE} is the highest reflectance value on the high side of the edge and \bar{R}_H is the mean reflectance on the high side of the edge. Reflectance values R_{LE} and \bar{R}_L are similarly defined for the low side of the edge. Using these four reflectance values, edge enhancement metrics for the high and the low sides of the edge can be defined as shown in Eqs. 1 and 2.

$$E_H = (R_{HE} - \bar{R}_H) \quad (1)$$

$$E_L = (\bar{R}_L - R_{LE}) \quad (2)$$

One would expect the edge values, E_H and E_L , to depend on the magnitude of the edge, measured as ΔR_{edge} , as well as on the design of the diffusion kernel. To show the importance of ΔR_{edge} on edge enhancement, the error diffusion kernel shown in Fig. 1 was applied, in raster fashion, to edges over the range ranging $0.1 < \Delta R_{\text{edge}} < 0.9$. The results in Fig. 3 show a maximum edge effect for an edge of magnitude $\Delta R_{\text{edge}} = 0.5$. It is not surprising that the average enhancement, $(E_H + E_L)/2$, occurs at $\Delta R_{\text{edge}} = 0.5$, but it is less intuitive that both E_H and E_L individually reach maxima at $\Delta R_{\text{edge}} = 0.5$.

Original manuscript received August 6, 2001

▲ IS&T Member

©2002, IS&T—The Society for Imaging Science and Technology

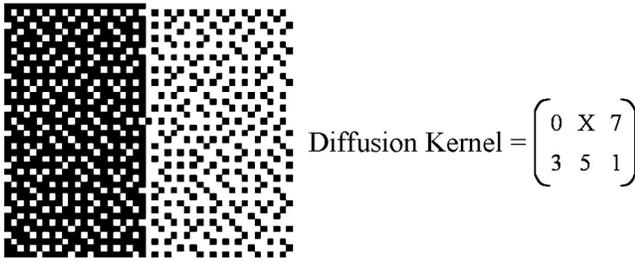


Figure 1. Example section of a 128×128 pixel image generated by raster error diffusion of a gray step image with reflectance gray levels of 0.26 and 0.73. Error is distributed in fractions $7/16$, $3/16$, $5/16$, and $1/16$, where $16 = 7 + 3 + 5 + 1$.

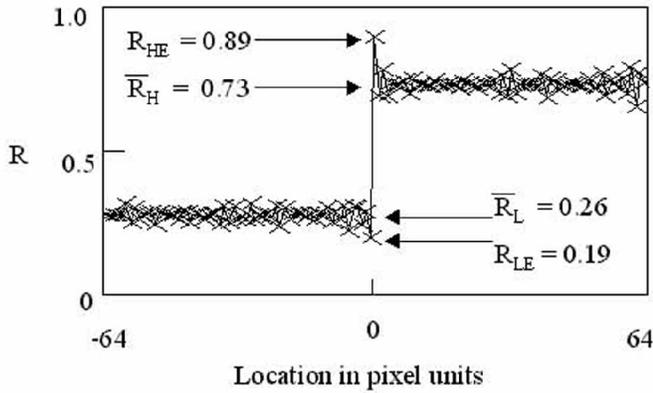


Figure 2. The average pixel gray level versus distance from the edge for the 128×128 image illustrated in Fig. 1.

Direction of Error Diffusion

The diffusion kernel allows error to be diffused in different directions, and this impacts the symmetry of the enhancement effect. This issue was explored experimentally by applying the raster process described above to an edge of amplitude $\Delta R_{\text{edge}} = 0.5$ ($R_L = 0.25$ and $R_H = 0.75$) using different diffusion kernels. The results are summarized in Table I.

A sample of the edge formed with kernel 1A in Table I is shown in Fig. 4, with the edge graph averaged vertically over 128 pixels. The diffusion of error with this simple kernel clearly resulted in the usual dot modulated gray levels, but edge enhancement was observed over the noise level characteristic of the image. However, by diffusing to the two pixels in front of the raster process, using kernel 2A of Table I, resulted in a significant edge enhancement, as shown in Fig. 5.

Intuition might have suggested that expansion of the kernel to diffuse error over a spatially larger region might decrease the magnitude of edge enhancement, but the data in Table I and Figs. 4 and 5 show this is not the case. It is common to use the analogy of noise power when describing error diffusion images, but analogy might be misleading in this case since there is no apparent conservation of edge energy associated with the structure of the kernel. This is further illustrated by considering kernels 3A and 3B. Diffusion over three pixels clearly does not decrease edge enhancement, but actually increases it. Moreover, by diffusing the error to pixel locations be-

TABLE I. Summary of Diffusion Kernels for Left-to-Right Raster Processing of Edge with $\Delta R_{\text{edge}} = 0.5$; $R_L = 0.25$; $R_H = 0.75$. Values in Parenthesis Are Zero Within Experimental Error.

Label	Kernel	E_L	E_H
1A	_ *1 000	(0.04)	(0.04)
1B	_ *0 001	(-0.02)	(-0.02)
1C	_ *0 010	(0.00)	(0.00)
2A	_ *11 0000	(-0.04)	0.25
2B	_ *00 0010 0001	(-0.03)	0.25
2C	_ *00 0100 0100	(0.00)	(0.00)
3A	_ *111 00000	(-0.05)	0.25
3B	_ *00 1110	0.25	0.25
FS	_ *70 3510	0.21	0.19

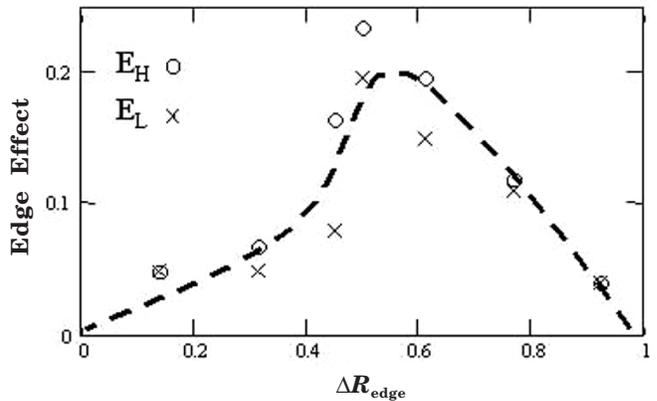


Figure 3. Edge metrics E_H (o) and E_L (x) versus edge magnitude, ΔR_{edge} , for the raster error diffusion with the kernel shown in Fig. 1.

hind the raster location X in the kernel in 3B, both sides of the edge are equally enhanced.

Direction of the Edge

The raster process used in this study was a simple left-to-right process. Although an alternating left-to-right and right-to-left raster could be applied to enhance edges in both directions, the left-to-right raster was employed in this study to examine directionality inherent to the diffusion kernel. This was investigated by applying the commonly used Floyd–Steinberg kernel, FS in Table I, to edges in the four orientations shown in Fig. 6. All original images were of amplitude $\Delta R_{\text{edge}} = 0.5$.

The pixels in the edge images were averaged over 128 pixels in each row (T and B) or column (L and R) paral-

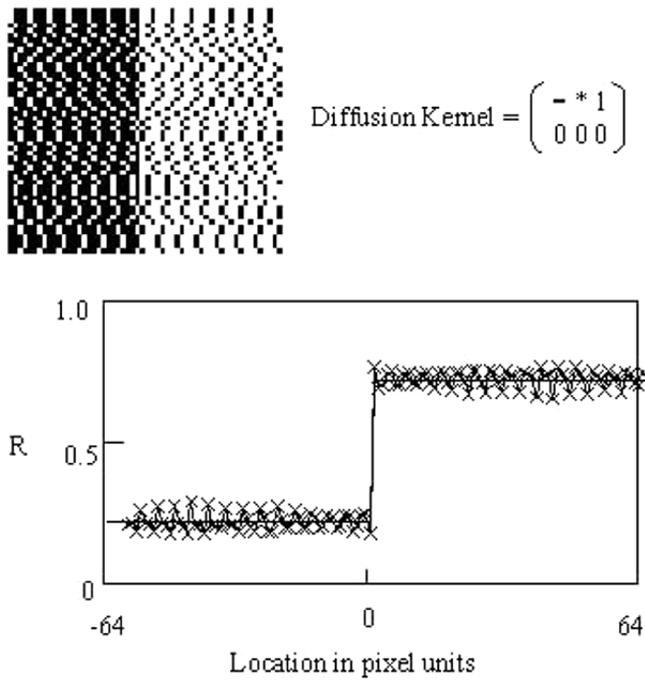


Figure 4. Raster error diffusion kernel 1A applied to edge with $\Delta R_{\text{edge}} = 0.5$; $R_L = 0.25$; $R_H = 0.75$.

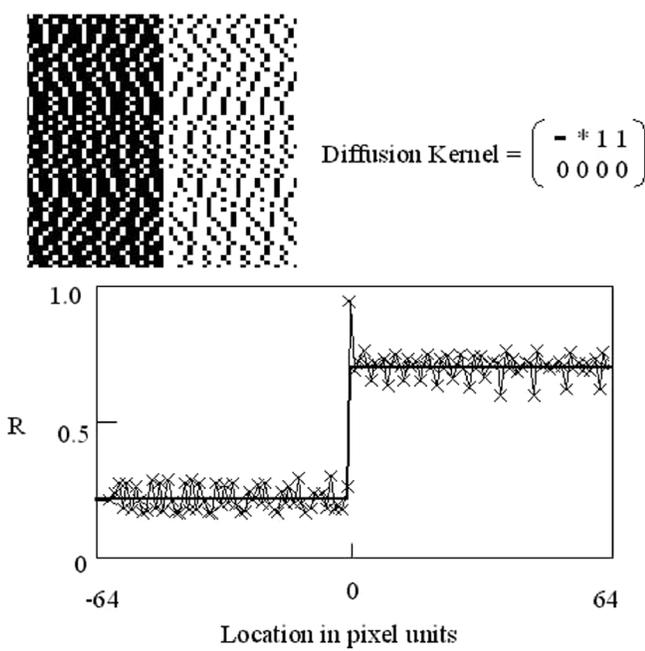


Figure 5. Raster error diffusion kernel 2A applied to edge with $\Delta R_{\text{edge}} = 0.5$; $R_L = 0.25$; $R_H = 0.75$.

labeled to the edge. The resulting average gray values were plotted to generate the edge traces illustrated in Fig. 7. The results show that the FS kernel is not entirely symmetrical in its edge enhancement characteristics. Vertical edges are enhanced equally on both sides, but horizontal edges are not. In image T the dark side of the edge is not enhanced, and in image B the light side is not enhanced. In other words, the upward side of the

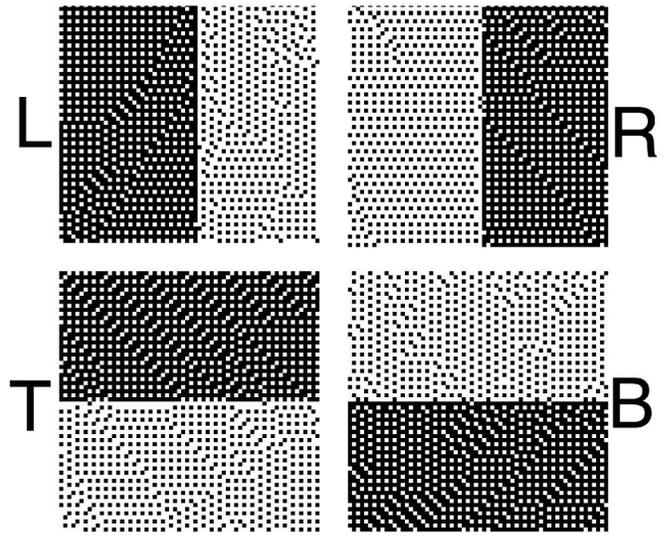


Figure 6. Edge orientations labeled with the dark side on the left (L), right (R), top (T) and bottom (B).

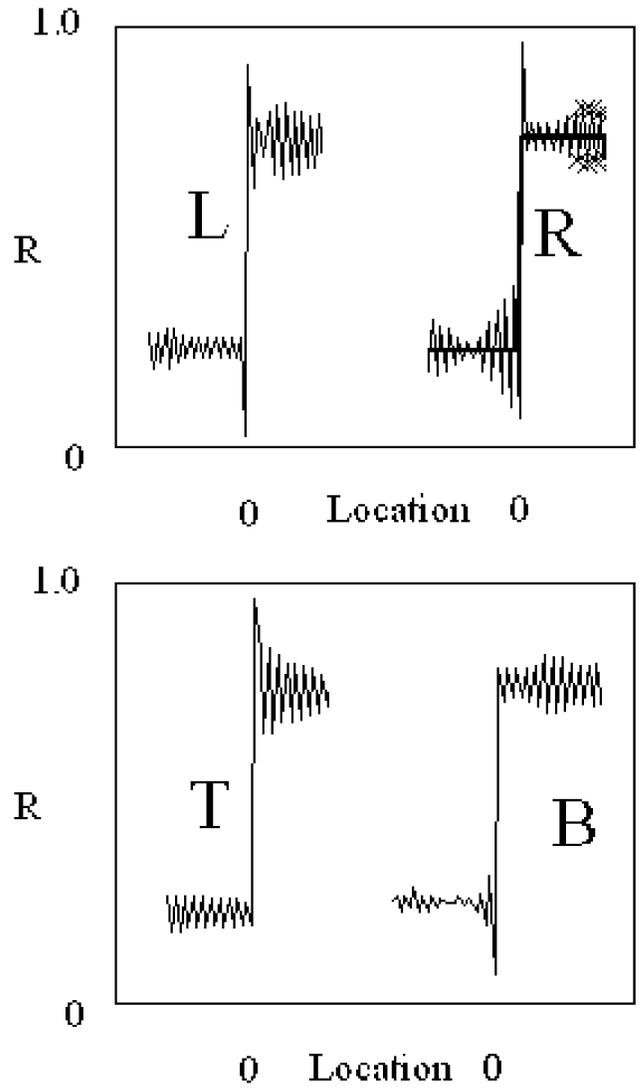


Figure 7. Edge traces corresponding to edges oriented as shown in Fig. 6.

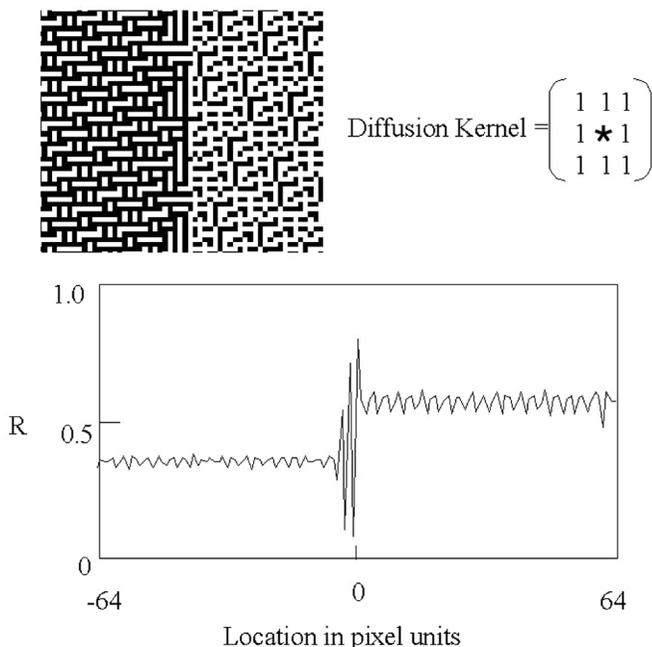


Figure 8. Omni-Directional Error Diffusion of edge of amplitude $\Delta R_{\text{edge}} = 0.2$, $R_L = 0.4$; $R_H = 0.6$.

edge, whether light or dark, was not enhanced. This is not surprising since raster processing is unable to diffuse error in the upward direction.

Omni-Directional Error Diffusion

Error diffusion is generally carried out using raster processing. However, pixels in the image can be addressed in an order other than raster order. By addressing pixels in a spatially distributed sequence, one may apply error diffusion kernels that are omni-directional. An example is shown in Fig. 8.

$$\mathbf{A} =$$

0	60	32	4	64	36	8	68	40	12	72	44	16
41	13	73	45	17	77	49	21	81	53	25	85	57
82	54	26	86	58	30	2	62	34	6	66	38	10
35	7	67	39	11	71	43	15	75	47	19	79	51
76	48	20	80	52	24	84	56	28	0	60	32	4
29	1	61	33	5	65	37	9	69	41	13	73	45
70	42	14	74	46	18	78	50	22	82	54	26	86
23	83	55	27	87	59	31	3	63	35	7	67	39
64	36	8	68	40	12	72	44	16	76	48	20	80
17	77	49	21	81	53	25	85	57	29	1	61	33
58	30	2	62	34	6	66	38	10	70	42	14	74
11	71	43	15	75	47	19	79	51	23	83	55	27
52	24	84	56	28	0	60	32	4	64	36	8	68

Figure 9. Portion of 88×88 matrix \mathbf{A} formed with $N = 14$. Elements less than 10 are shown in bold.

The algorithm for the pseudo random order of pixel processing is a derivative of linear pixel shuffling (LPS).⁽⁷⁾ First a Fibonacci-like sequence is generated with the seed numbers $G_0 = 0$, $G_1 = 1$, and $G_2 = 1$. Numbers in the sequence are extended as far as needed with Eq. (3).

$$G_N = G_{N-1} + G_{N-3} \text{ for } N \geq 3 \quad (3)$$

For example, terms G_0 through G_{14} of this sequence are: 0,1,1,1,2,3,4,6,9,13,19,28,41,60,88. We also need this sequence with negative subscripts, $N < 0$. This also is done with Eq. (3). For example, G_{-1} through G_{-14} are 0,1,0,-1,1,1,-2,0,3,-2,-3,5,1,-8.

The size of the image to be processed determines the sequence length. The value of N is selected so that G_N is equal to or greater than the largest dimension of the image. For example, a 640×480 pixel image requires $N = 20$ because $G_{19} = 595$ and $G_{20} = 870$. A 64×64 pixel image would use $N = 14$, and this will be used to illustrate the remainder of the algorithm.

After determining N , a matrix \mathbf{A} , containing elements $\mathbf{a}_{i,j}$, is generated. This is done with modulo Eq. (4). This matrix will be used to define the order of processing of pixels.

$$\mathbf{a}_{i,j} = \{iG_{N-2} + jG_{N-1}\} \% G_N \quad (4)$$

Anderson has shown that the elements of matrix \mathbf{A} are all integers $0 \leq a \leq (G_N - 1)$. Each integer in this range appears in the matrix exactly G_N times.⁶ This follows from the fact that G_N , G_{N-1} , and G_{N-2} have no common divisor greater than 1, which in turn follows from the way the number sequence is defined. A portion of the 88×88 \mathbf{A} matrix for $N = 14$ is illustrated in Fig. 9.

The utility of matrix \mathbf{A} is in its distribution of elements. Numbers that are close in value are spatially far apart, as illustrated with first decade of integers in Fig. 9. This is exactly the property needed to address pixels in a spatially distributed order. All pixels in the image at locations i,j corresponding to $\mathbf{a}_{i,j} = 0$ are processed first. In the sub-portion of matrix \mathbf{A} in Fig. 9,

$$\text{Diffusion Kernel} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & * & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

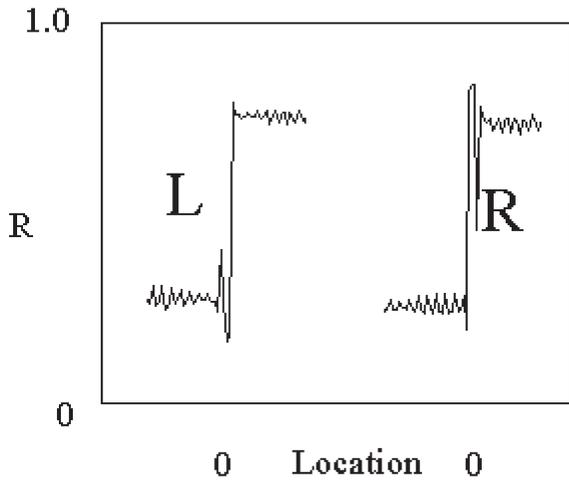


Figure 10. Omni-Directional Error Diffusion applied to edges of four orientations. $\Delta R_{\text{edge}} = 0.6$, $R_L = 0.2$; $R_H = 0.8$.

$$\text{Diffusion Kernel} = \begin{bmatrix} 1 & 2 & 1 \\ 1 & * & 1 \\ 1 & 2 & 1 \end{bmatrix}$$

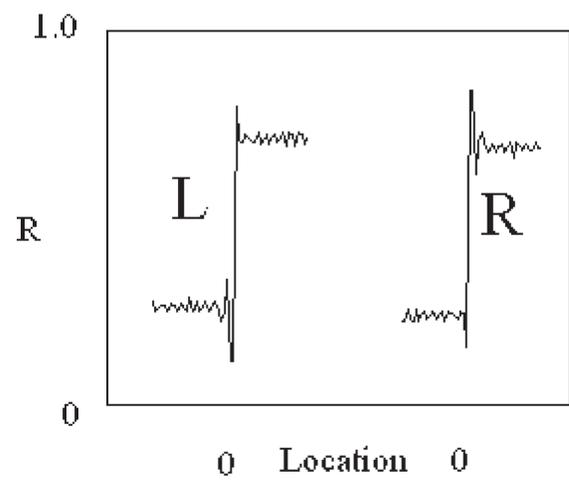


Figure 11. Omni-Directional Error Diffusion applied to edges of four orientations. $\Delta R_{\text{edge}} = 0.6$, $R_L = 0.2$; $R_H = 0.8$.

these are elements $\mathbf{a}_{0,0}$, $\mathbf{a}_{4,9}$, $\mathbf{a}_{12,5}$, etc. Next, all pixels in locations with $\mathbf{a}_{i,j} = 1$ are processed, and etc. through $\mathbf{a}_{i,j} = (G_N - 1)$.

Locating each coordinate pair (i,j) corresponding to a given value of $\mathbf{a}_{i,j} = p$ is straight forward. There are exactly G_N coordinate pairs (i,j) with $\mathbf{a}_{i,j} = p$, and these can be addressed sequentially $0 \leq q < G_N$ using modulo Eqs. (5) and (6).

$$i(p,q) = \{pG_{N+3} + qG_{N-3}\} \% G_N \quad (5)$$

$$j(p,q) = \{pG_N + qG_{N-2}\} \% G_N \quad (6)$$

If the $G_N \times G_N$ matrix is larger than the image, locations (i,j) outside the image are ignored.

Omni-directional error diffusion can be applied with kernels such as the one illustrated in Fig. 10. The algorithm is identical to raster error diffusion, but the raster sequence is in terms of coordinates (p,q) instead of (i,j) . Each (p,q) in the sequence maps uniquely to a lo-

cation (i,j) in the image, and threshold error can be diffused to neighboring pixels in any direction in the image.

Toward the end of the omni-directional error diffusion algorithm, some of the neighboring pixels to which error is diffused have already been processed. This is accommodated by adjusting the diffusion kernel so error is distributed among only non-quantized pixels. This is easy to do since the algorithm is linear and entirely deterministic. For example, consider a kernel $[1,2,3,4]$ with error fractions $1/10$, $2/10$, $3/10$, and $4/10$. If the 3rd location is already quantized, the kernel is changed to $[1,2,0,4]$ with error fractions $1/6$, $2/6$, $0/6$, and $4/6$. If locations 1 and 3 are occupied, kernel $[0,2,0,4]$ is used.

Edge Enhancement with Omni-Directional Kernels

As is the case with raster error diffusion, the behavior of omni-directional error diffusion is not entirely intuitive. The omni-directional kernel shown in Fig. 10 was applied to diffuse error in images with edges ori-

ented in the same four directions illustrated in Fig. 6. It was anticipated that edge enhancement would be symmetrical in all four orientations, but as shown in Fig. 10, this was not the case. Edges at orientations T and B were symmetrically enhanced, but L and R were not.

A number of omni-directional kernels were examined, and by trial and error the kernel illustrated in Fig. 11 was found to enhance edges symmetrically in the four orientations.

Conclusions and Opportunities

The kernel shown in Fig. 11 is not necessarily an optimum kernel for any given application. An optimum kernel would, of course, depend on the print technology and the addressability of the printer. The current study is an experimental study of the edge kernel for omni-directional processing. The results appear to be sufficiently promising that additional experimental and theoretical projects would seem justified.

Edge enhancement by error diffusion, although well known, remains a non-intuitive phenomenon. Recent theoretical studies, reported on traditional raster based processing, provide some rational for the observed non-additive nature of the relationship between kernel distributions and edge enhancement, and similar effects appear to be at play with omni-directional error diffusion. Moreover, omni-directional error diffusion has been demonstrated to be capable of enhancing edges with greater directional symmetry than is possible with raster processing. It would appear, therefore, that further study of omni-directional processing would be justified. For example, a theoretical analysis of omni-directional error diffusion, similar to studies recently reported on

raster processes,⁸⁻¹⁰ would be expected to provide useful guidelines for designing better kernels. In addition, one might suggest a statistical analysis of kernels and their effect on edge enhancement as an empirical approach to kernel optimization. Finally, a practical project to optimize an omni-directional kernel might include a study of the edge effect relative to the noise power spectrum of the halftone. Clearly much remains to be done, but the potential utility of omni-directional processing indicates such work would be fruitful. ▲

References

1. K. T. Knox, Evolution of Error Diffusion, *J. Electronic Imag.* **8**, 422 (1999).
2. Reviews of digital halftoning and error diffusion can be found in "Recent Progress in Digital Halftoning II", Reiner Eschbach, Ed., IS&T, Springfield, VA, 1999; and *Selected Papers on Digital Halftoning*, J. P. Allebach, Ed., *SPIE Milestone Series*, vol. **MS 150**, SPIE, Bellingham, WA, 1999.
3. K. T. Knox, Edge Enhancement in Error Diffusion, *SPSE 42nd Annual Conference*, SPSE, Washington, DC, 1989, p. 310,.
4. Reiner Eschbach and Keith T. Knox, Error-Diffusion Algorithm with Edge Enhancement, *J. Opt. Soc. Amer. A*, **8**, 1844 (1991).
5. J. H. Kim, T. Il Chung, H. S. Kim, K. S. Son, and Y. S. Kim, New Edge-Enhanced Error Diffusion Algorithm Based on Error Sum Criterion, *J. Electronic Imag.* **4**,172 (1995).
6. P.r G. Anderson, J. Arney, and Kevin Ayer, Linear Pixel Shuffling (I): New Paradigms For New Printers, *IS&T NIP16, International Conference on Digital Printing Technologies*, IS&T, Springfield, VA, 2000, p. 801.
7. Peter Engeldrum, unpublished results.
8. T. D. Kite, B. L. Evans and A. C. Bovik, Modeling An Quality Assessment of Halftoning by Error Diffusion, *IEEE Transactions on Image Processing*, **9**, 909 (2000).
9. K.T. Knox, Symmetric Edge Enhancement In Error Diffusion, *Proc. SPIE*, **3963**, 536 (2000).
10. Z. Fan and F. Li, Edge Behavior of Error Diffusion, *Proc. IEEE International Conference on Image Processing*, **3**, 113 (1995).