FM HALFTONING VIA BLOCK ERROR DIFFUSION

Niranjan Damera-Venkata

Halftoning and Image Processing Group
Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304
E-mail: damera@exch.hpl.hp.com

Brian L. Evans

Embedded Signal Processing Laboratory
Dept. of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084
E-mail: bevans@ece.utexas.edu

ABSTRACT

Error diffusion halftoning is a popular method of producing frequency modulated (FM) halftones. In FM halftoning the dot size and shape is fixed (equal to one pixel) and the dot frequency is varied in accordance to the graylevel values of the underlying grayscale image. We generalize error diffusion to produce FM halftones with user controlled dot size and shape using block quantization and a block filter in the feedback loop. We call this modified quantization and feedback process block error diffusion. The block filters are designed from well known scalar error filter prototypes and retain their properties. Further, we show that choosing a structured block filter results in an efficient parallel implementation of block error diffusion.

1. INTRODUCTION

Digital halftoning may be classified into three categories–AM (amplitude modulation), FM (frequency modulation) and AM-FM hybrid halftoning. In AM halftoning the dot size is varied depending on the graylevel value of the underlying grayscale image while the dot frequency is held constant. A typical example of this type of halftoning is clustered-dot ordered dither. FM halftones have a fixed dot size and shape, but the frequency of the dots varies with the graylevel of the underlying grayscale image. Conventional digital FM halftones have a fixed dot size of one pixel. Typical examples of FM halftones include halftones produced with dispersed-dot ordered dither and error diffusion halftoning [1]. AM-FM halftones have variable dot shape/size and variable frequency of dots depending on the graylevel value to be reproduced. Typical examples of halftones that are in this category are the green-noise halftones of Levien [2, 3], halftones generated by Velho and Gomez [4] using halftoning on space filling curves, and halftones generated by Scheermesser and Bryngdahl [5] with texture control.

Conventional FM halftones (single pixel halftone dots) however, suffer from physical printer imperfections such as dot gain [6]. Dot gain is caused from the increase in printed dot size from the intended dot size. This results in dot overlap, which results in a reduction in tonal range that

causes loss of definition in the reproduced image [3]. Lau, Arce and Gallagher [3] further note that if the size/shape variation from printed-dot to printed-dot is small, then the effects of dot gain can be mitigated by dot gain compensation techniques. If however, there is a large variation in the size/shape of the printed dots, clustering of the digital halftone dots adds robustness to the halftoning process, and in many cases is a necessity [7].

This paper generalizes conventional error diffusion halftoning to produce FM halftones with user controlled dot size and shape. Unlike conventional FM halftones, dot clusters of more than one pixel are allowed. The generated FM halftones can be designed to have very low dot size/shape variation, while their spacing is modulated depending on the underlying grayscale image. This is different from the AM-FM methods discussed above that have variable dot shape/size. Furthermore the method presented in this paper is single pass unlike [5] and uses the conventional raster scan unlike [2, 4]. Unlike the above methods, the method presented in this chapter has user tunable sharpness control and multiresolution embedding capability built into the framework. We refer to the framework as block error diffusion, since the notion of a pixel in conventional error diffusion is replaced with the concept of a pixel block in our frame work. The quantization error at each pixel in the pixel block is diffused to suitably selected pixels in neighboring blocks in suitably selected proportions. Thus a whole block of quantization error is diffused.

Section 2 introduces the basic framework of block error diffusion by formulating the basic equations. Section 3 demonstrates the effect of designing the block error filter from well known scalar error filter prototypes. We also discuss how FM halftones with user-controlled dot shape and size may be produced. Section 4 derives an efficient parallel implementation for block error diffusion. Finally, Section 5 concludes the paper by summarizing the results.

2. BLOCK ERROR DIFFUSION

Fig. 1 shows the block diagram illustrating block error diffusion. Although the basic diagram resembles conventional error diffusion halftoning there are key differences. The input is an $N \times N$ block of pixels (called a pixel block) as opposed to a single pixel in conventional error diffusion and is assumed to be normalized to the range $[0, 1]$. The sam-
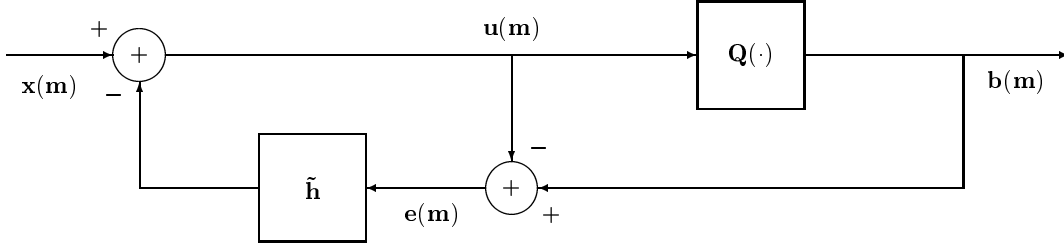
Figure 1: System block diagrams for block error diffusion halftoning where $\tilde{\mathbf{h}}$ represents a fixed 2-D nonseparable FIR error filter with matrix-valued coefficients. The vector $\mathbf{m}$ represents the 2-D index (m_1, m_2).

ples of the input pixel block may be organized into a vector $\mathbf{x}(\mathbf{m})$ at each location $\mathbf{m} = (m_1, m_2)$.

The quantizer output for each sample in a pixel block is exactly one element from the discrete set $\mathcal{O} = \{0, 1\}$. Here 0 represents black and 1 represents white. We quantize each pixel block using the 1-bit quantizer $\mathbf{Q}(\cdot)$. The quantization is performed by simply thresholding each sample of the modified input pixel block $\mathbf{u}(\mathbf{m})$ at mid gray.

The filter in the feedback loop has matrix-valued coefficients. The filter operates on the quantization error sequence $\mathbf{e}(\mathbf{m})$ to produce the feedback signal sequence $\mathbf{f}(\mathbf{m})$ according to

$$\mathbf{f}(\mathbf{m}) = \sum_{\mathbf{k} \in \mathcal{S}} \tilde{\mathbf{h}}(\mathbf{k}) \mathbf{e}(\mathbf{m} - \mathbf{k}) \qquad (1)$$

where $\tilde{\mathbf{h}}(\cdot)$ is an $N^2 \times N^2$ matrix-valued sequence, and $\mathcal{S}$ is the filter support.

In terms of block filtering, the operation described by (1) can be described with the help of Fig. 2 which illustrates a block error filter operating on pixel blocks of 2×2 pixels. The output pixel block is computed by forming 4 different linear combinations of all pixels in the pixel block mask which consists of 16 pixels. Each linear combination produces a single output pixel of the output pixel block.
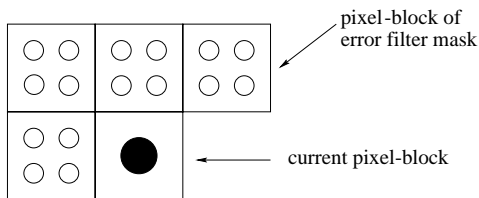

Figure 2: Block error filter operating on pixel blocks of 2×2 pixels. The shaded circle indicates the current pixel block. The unfilled circles indicate the error image pixels underlying the block filter mask. The pixels in the output pixel block are computed using 4 linear combinations of all 16 error pixels within the error filter mask.

3. FM HALFTONING VIA BLOCK ERROR DIFFUSION

The block error filter in the feedback loop governs how quantization error is diffused to the neighboring pixel blocks. For conventional error diffusion, one only needs to decide how much of the quantization error is to be diffused to each neighboring pixel under the constraint that all the quantization error be diffused. The case of block error diffusion also requires that all of the quantization error be diffused. This imposes the constraints

$$\tilde{\mathbf{\Gamma}} \mathbf{1} = \mathbf{1}, \quad \tilde{\mathbf{\Gamma}} \geq 0 \qquad (2)$$

where $\mathbf{1}$ represents an $N^2 M^2 \times 1$ column vector with all of its elements equal to one. These conditions correspond to the assertion that the elements of the matrix-valued error filter coefficients be non-negative and that their rows sum to unity.

3.1. Error filter design

In designing the coefficients $\tilde{\mathbf{\Gamma}}$, we start with the coefficients of a conventional error filter and map them into corresponding block filters. By starting with a scalar filter with the same support (say $M \times M$) as the multifilter or block filter, and by representing its coefficients by the row vector $\tilde{\gamma}$, where

$$\tilde{\gamma} = \begin{bmatrix} g(0) \mid g(1) \mid \ldots \mid g(M^2 - 1) \end{bmatrix} \qquad (3)$$

a multifilter $\tilde{\mathbf{\Gamma}}$ may be derived from $\tilde{\gamma}$ as follows

$$\tilde{\mathbf{\Gamma}} = \tilde{\gamma} \bigotimes \tilde{\mathbf{D}} \qquad (4)$$

where $\bigotimes$ denotes the Kronecker product operation. $\tilde{\mathbf{D}}$ is an $N^2 \times N^2$ matrix which we call the diffusion matrix. Since the element of $\tilde{\gamma}$ are the coefficients of a conventional error filter they are non-negative and sum to one. Thus to satisfy the constraints imposed by (2), the diffusion matrix must satisfy the constraints

$$\tilde{\mathbf{D}} \mathbf{1} = \mathbf{1}, \quad \tilde{\mathbf{D}} \geq 0 \qquad (5)$$

where $\mathbf{1}$ represents an $N^2 \times 1$ column vector with all its elements equal to one. Thus, by imposing structure on $\tilde{\mathbf{\Gamma}}$ we only need to design the $N^2 \times N^2$ diffusion matrix $\tilde{\mathbf{D}}$.

We will now show that the decomposition of (4) is a natural and intuitive way of designing suitable error filters to generate FM halftones via block error diffusion.
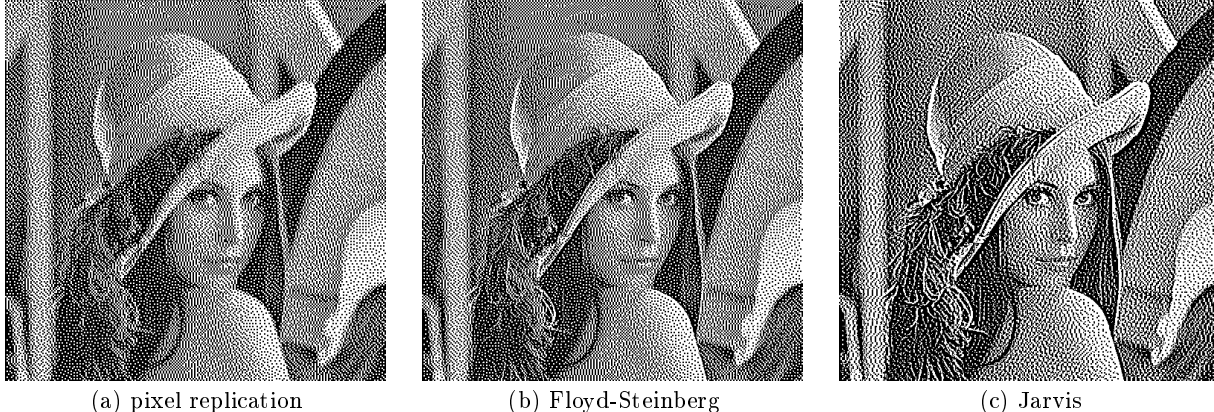
(a) pixel replication   (b) Floyd-Steinberg   (c) Jarvis

Figure 3: Halftones generated by (a) pixel replication induced block clustering and error diffusion with block error filters derived from conventional (b) Floyd-Steinberg and (c) Jarvis filters. Note the improved performance of the block error filters over pixel replication induced block clustering.

The physical meaning of deriving the block filter from a given conventional error filter via (4) is that the quantization error incurred at the current pixel block is diffused to the neighboring pixel blocks in the same proportions that a conventional error filter diffuses error to its neighboring pixels. The diffusion matrix $\tilde{\mathbf{D}}$ governs the proportions in which errors are to be distributed within the samples of a pixel block. According to our proposed structure these proportions are constant independent of the relative position of the pixel blocks to which errors are diffused. This enforces a local isotropy constraint. The constraints on the diffusion matrix simply indicate that *all* of the quantization error that is diffused to a pixel block must be diffused among pixels that make up the block. Thus the pixel blocks in the block-error diffusion framework are made to behave like pixels in conventional error diffusion and the block errors are diffused in much the same way as pixel errors in conventional error diffusion.

3.2. FM halftoning

This section will produce FM-halftones with dot clusters greater than one pixel. One method of achieving dot clustering would be to halftone a downsampled version of the grayscale image and then using pixel replication to get a halftone of the same size as the grayscale image. For 2×2 minority pixel dot clusters, one could filter the original grayscale image with a halfband filter, downsample the original grayscale image by retaining every alternate sample in the horizontal and vertical directions, halftone the downsampled grayscale image, and interpolate it to the resolution of the original grayscale image by pixel replication. This process is identical to halftoning the filtered image after replicating the leftmost uppermost sample in each block to all samples and using the identity diffusion matrix $\tilde{\mathbf{D}} = \tilde{\mathbf{1}}_{4 \times 4}$. Fig. 3(a) shows a halftone obtained using halftone pixel replication. Clearly the spatial resolution of the image suffers due to the pixel replication and pre-filtering.

Our approach to FM halftoning relies on forming mi-

nority pixel dot clusters by diffusing the quantization error from each pixel block, equally to neighboring pixel blocks. The error diffused to each block within the block error filter mask is governed by the corresponding conventional error filter coefficients $\tilde{\gamma}$. We use the diffusion matrix $\tilde{\mathbf{D}} = \tilde{\mathbf{1}}_{4 \times 4}$ The motivation for this is that the error at any sample within the current pixel block that is diffused to an adjacent pixel block will be spread to all the samples within the pixel block equally. This means that the quantization decisions of all pixels within the modified pixel block will be biased in the same direction. Intuitively this should result in the halftoned samples of that pixel block organizing themselves into a pixel block cluster.

Fig. 3(b) and Fig. 3(c) show sample halftones obtained using block error diffusion with the above diffusion matrix and using error filter prototypes $\tilde{\gamma} = \begin{bmatrix} \frac{1}{16} & \frac{5}{16} & \frac{3}{16} & \frac{7}{16} \end{bmatrix}$ and $\tilde{\gamma} = \begin{bmatrix} \frac{1}{48} & \frac{3}{48} & \frac{5}{48} & \frac{3}{48} & \frac{1}{48} & \frac{3}{48} & \frac{5}{48} & \frac{7}{48} & \frac{5}{48} & \frac{3}{48} & \frac{5}{48} & \frac{7}{48} \end{bmatrix}$ respectively. This corresponds to the well known Floyd-Steinberg [1] and Jarvis [8] error filters respectively. There is no need to use a prefilter to prevent spatial aliasing. By visual inspection of Fig. 3 the spatial resolution of the grayscale image is not compromised. The halftones even exhibits sharpening, characteristic of conventional Floyd-Steinberg and Jarvis error diffusion [9]. The dots are clustered into 2×2 blocks.

Using the method described above, it is possible to cluster the halftone dots into user defined shapes and sizes. Halftones with dot clusters of 3×3 and 4×4 are produced using 9×9 and 16×16 diffusion matrices having all their elements equal to $\frac{1}{9}$ and $\frac{1}{16}$ respectively. Fig. 4(a) shows a halftone with minority-block dot clusters that are in the shape of a "plus". Fig. 4(b) shows a detail view of the corresponding dot structure.

Fig. 4(a) was produced using the diffusion matrix $\frac{1}{9}\tilde{\mathbf{1}}_{9 \times 9}$ similar to square block clustered dot FM halftones described earlier.

However, we quantize the minority pixel blocks by replacing them with the desired dot shape, while the majority pixel blocks are quantized as usual at mid-gray. The quantization error is computed at each pixel block location as the vector difference between the output shape (if we are

(a) "plus" dots
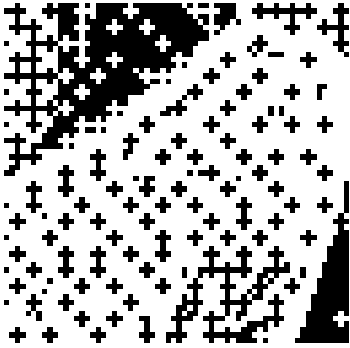
(b) Detail of hat from (a)

Figure 4: Block error diffused halftones with user controlled dot shapes.

quantizing a minority pixel) or the standard output pixel block (if we are quantizing a majority pixel). Thus at each pixel block location we need to determine if the current pixel block to be quantized will form a minority pixel block or a majority pixel block. This may be estimated by comparing the majority pixel type (1 or 0) in the current modified input pixel block against the majority pixel type in current input pixel block (the majority pixel type within a block is 1 if the majority of the samples in the block have values above 0.5 and 0 otherwise). If the two are not equal, then the pixel block is considered a minority block and the output pixel block is replaced with the desired dot shape. If the two are equal, then the current pixel block is considered a majority pixel block and the output pixel block is simply the quantization of the current pixel block at mid-gray.

4. EFFICIENCY ANALYSIS

By converting (1) to the frequency domain,

$$F_i(\mathbf{z}) = \sum_{k=0}^{N^2-1} H_k^{(i)}(\mathbf{z}) E_k(\mathbf{z}) \qquad (6)$$

Each of the filters H_j^i, for $i, j = 0, 1, \cdots, N^2 - 1$ represents a conventional scalar filter that can be derived from the

filter coefficients of the multifilter $\tilde{\Gamma}$ using the polyphase decomposition [10]. Since $\tilde{\Gamma}$ is fixed, the polyphase components of its rows may be precomputed. N^4 polyphase filters are required for the implementation. The result is a set of conventional filters with scalar coefficients, which enables the components of the error vector sequence to be buffered and filtered independently in parallel. Due to the redundancy in the diffusion matrix further savings in computation can be achieved. This is because in the case where the diffusion matrix has all its elements equal, $F_0(\mathbf{z}) = F_1(\mathbf{z}) = \cdots = F_{N^2-1}(\mathbf{z})$. Thus only one set of parallel scalar filtering operations is required. In fact, each parallel filter is identical to the scalar error filter prototype used to design the block filter.

5. CONCLUSIONS

We have introduced a general framework for producing FM halftones with user controlled dot shape. The block filter governs how pixel block errors are diffused to neighboring blocks. In this paper, we impose structure on the error filter coefficients to encapsulate the advantages of well known conventional scalar error diffusion filters and enable an efficient parallel implementation. Design of the diffusion matrix for specific applications and extension of the framework to produce AM-FM halftones will be investigated in future work.

6. REFERENCES

[1] R. Floyd and L. Steinberg, "An adaptive algorithm for spatial grayscale," *Proc. Soc. Image Display*, vol. 17, no. 2, pp. 75–77, 1976.

[2] R. Levien, "Output dependent feedback in error diffusion halftoning," *Proc. IS&T Imaging Science and Technology*, vol. 1, pp. 115–118, May 1993.

[3] D. L. Lau, G. R. Arce, and N. C. Gallagher, "Greennoise digital halftoning," *Proceedings of the IEEE*, vol. 86, no. 12, pp. 2424–2442, Dec. 1998.

[4] L. Velho and J. M. Gomez, "Digital halftoning with space filling curves," *Computer Graphics*, vol. 25, pp. 81–90, July 1991.

[5] T. Scheermesser and O. Bryngdahl, "Control of texture in image halftoning," *J. Opt. Soc. Am. A*, vol. 13, pp. 81–90, Aug. 1996.

[6] M. A. Coudray, "Causes and corrections of dot gain on press," *Screen Printing: The Journal of Technology and Management*, vol. 86, pp. 18–26, Aug. 1996.

[7] R. Ulichney, *Digital Halftoning*, MIT Press, 1987.

[8] J. Jarvis, C. Judice, and W. Ninke, "A survey of techniques for the display of continuous tone pictures on bilevel displays," *Computer Graphics and Image Processing*, vol. 5, pp. 13–40, May 1976.

[9] T. D. Kite, B. L. Evans, and A. C. Bovik, "Modeling and quality assessment of halftoning by error diffusion," *IEEE Trans. Image Processing*, vol. 9, no. 4, pp. 909–922, May 2000.

[10] G. Strang and T. Q. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, 1996.