Fast Rehalftoning and Interpolated Halftoning Algorithms with Flat Low-Frequency Response

Thomas D. Kite
Audio Precision, P.O. Box 2209, Beaverton, OR 97075-2209 USA

Brian L. Evans and Alan C. Bovik
Department of Electrical and Computer Engineering
The University of Texas at Austin, Austin, TX 78712-1084 USA

E-mail: {tom,bevans,bovik}@vision.ece.utexas.edu

Abstract— **We present raster image processing algorithms for rehalftoning error diffused halftones and producing interpolated error diffused halftones. Rehalftoning converts a halftone created by one method into one created by another method. In interpolated halftoning, interpolation increases the image size before halftoning, e.g. for printing. Both rehalftoning and interpolated halftoning introduce blur and noise in the output image. To compensate for the blur, we use modified error diffusion, which has a variable gain parameter to control the sharpness. We derive optimal formulas for the sharpness control parameter to make the overall frequency response flat. The high-frequency noise is masked by error diffusion. The proposed algorithms yield halftones of high fidelity at a low computational cost.**

I. Introduction

Halftoning attempts to produce a binary (black-and-white) image from a grayscale image so that the image can be rendered on devices which cannot reproduce shades of gray. Rehalftoning converts one type of halftone into another. Interpolated halftoning resizes an image before halftoning. In this paper, we develop and optimize new algorithms for rehalftoning and interpolated halftoning for error diffused halftones.

A conventional approach for rehalftoning is to apply inverse halftoning and then halftoning. Inverse halftoning converts a halftone into a continuous tone image, e.g. by converting a 1 bit/pixel halftone into an 8 bit/pixel grayscale image. Inverse halftoning methods produce high-quality images by varying the tradeoff in resolution across the image using non-linear or spatially adaptive filters. These methods are typically computationally intensive, although fast implementations exist [1], [2], [3].

The proposed rehalftoning algorithm greatly reduces computation over a conventional approach, and is well-suited to raster image processing. We apply a simple linear finite impulse response (FIR) filter to the halftone to create a blurred, noisy pseudo-grayscale image. We then apply modified error diffusion [4] to the grayscale image to create the halftone. Modified error diffusion incorporates a parameter that determines the sharpness of the resulting halftone. We derive formulas for computing the optimal sharpness parameter to give a flat end-to-end system response; noise in the inverse halftone is masked by the quantization noise introduced by error diffusion halftoning. In the derivation, we use a linear gain model for error diffusion halftoning [5] and a polynomial approximation to the complex digital frequency $z = e^{j\omega}$. We assess the quality of rehalftoned images by using a weighted signal-to-noise ratio (WSNR) measure based on the human visual system.

The interpolated halftoning algorithm uses simplified interpolation to create high quality interpolated halftones. Computation is reduced over more complicated interpolation methods for the same visual quality. The linear gain model and the digital frequency approximation are again used to derive an optimum value for the sharpness parameter to flatten the system response.

This research was supported in part by a grant from Hewlett-Packard Laboratories, Palo Alto, CA, and an NSF CAREER Award under Grant MIP-9702707.

II. Background

In halftoning, the requirements of high visual quality and low computational cost tend to conflict, and many halftoning methods have been proposed. Halftoning methods may be divided into ordered dithering and stochastic. Ordered dithering by clustered dot screening is the most commonly used method in mass-market printed media. Stochastic halftoning, such as error diffusion, generally produces higher quality halftones than ordered dithering, but at a higher computation cost. Both clustered dot and error diffusion halftoning are used in printers and copiers.

A. Error Diffusion

Error diffusion is essentially a 2-D noise-shaped feedback coder; i.e., the output is quantized to a lower resolution and the quantization error is filtered and added to the input. Floyd-Steinberg error diffusion [6] offers extremely low complexity (four-tap 2-D filter with dyadic coefficients). Some print engines are unable to render Floyd-Steinberg halftones accurately. For these devices, another halftoning method such as clustered dot screening must be used.

Because error diffusion is a form of 2-D delta-sigma modulation [5], [7], it can be accurately modeled as a linear shift-invariant system. Its transfer function depends on the transfer function of the error filter, $H(\mathbf{z})$, and the effective gain of the quantizer, K_s [5]. For Floyd-Steinberg error diffusion, $H(\mathbf{z})$ is a four-term polynomial in the $\mathbf{z}$ variables, and based on measurements, $K_s \approx 2$.

B. Interpolation

An image often needs to be resized for printing, so that it appears at the correct size on the page. For instance, an image which is sized correctly for a printer with a resolution of 300 dpi (dots per inch) will be half the size when printed on a 600 dpi printer. In such instances, the image must be resized by interpolation before halftoning. Several interpolation methods are in common use, and are listed here in order of computational complexity [8]:

- Nearest neighbor interpolation;
- Bilinear interpolation; and
- Higher order functions, e.g. cubic splines.

Interpolation assumes that the pixel values of a spatially discretized image represent samples on an integer grid of a continuous intensity function, $I(x, y)$, that is defined over the entire plane. To resize the image, a grid of output points is constructed, and $I(x, y)$ is interpolated at these new points. The interpolation method defines how the intensity at each pixel is constructed.

Bilinear interpolation assumes that the image intensity $I(x, y)$ in the original scene varies linearly in the x and y directions over the rectilinear area whose corners are defined by four neighboring input pixels of the spatially quantized image. The assumption that $I(x, y)$ varies linearly between pixels fails at sharp edges, and the interpolated image therefore appears smoother than the original. However, bilinear interpolation is quite effective at suppressing the high frequency aliases of the baseband spectrum that appear in the passband after upsampling. These aliases lead to blockiness in the interpolated image.

III. Rehalftoning

A generic approach for the high-quality reproduction of a halftone on a particular device involves inverse halftoning followed by halftoning using a method optimized for the device in question. The combination of inverse halftoning and halftoning is rehalftoning. Rehalftoning enables operations such as rotation, scaling, and contrast adjustment to be performed on the intermediate grayscale image. In general, these operations are not possible with halftones [9], because they expand the wordlength or destroy halftone quality. In this section, we develop rehalftoning algorithms for error diffused halftones. Sometimes, an error diffused halftone is all that is available, either because the original image was stored in that form, or because it was scanned from an image that was printed using error diffusion.

A. Algorithm

The ultimate goal of a rehalftoning method is a visually pleasing *halftone*. Since halftoning inevitably discards image information because of the reduction in wordlength to one bit, it is overkill to use an inverse halftoning method that attempts to obtain grayscale images of the highest quality. The rehalftoning method described in this paper uses a linear lowpass filter to perform inverse halftoning by means of wordlength expansion. The resulting grayscale images are blurred and contain a good deal of quantization noise. However, this noise tends to be masked by the quantization noise introduced by the subsequent halftoning step. Furthermore, when error diffusion is used to produce the final halftone, the blurring of the inverse halftone can be compensated for by employing a modified halftoning method that allows the sharpness of the halftone to be adjusted with a single multiplicative parameter. The result is a halftone of high visual quality and equivalent sharpness to the original, created at very low cost in complexity. The complete rehalftoning system is shown in Fig. 1. The leftmost dashed box of Fig. 1 shows the transfer function of the halftoning method.

We use a linear filter $G(\mathbf{z})$, shown in the middle dashed box of Fig. 1, to create the intermediate inverse halftone. We design $G(\mathbf{z})$ to optimize the trade-off between reducing quantization noise and maximizing sharpness. Since we do not need to produce high-quality inverse halftones in this application, we can greatly reduce computational complexity. We design a separable $M \times M$ filter that

- is small, symmetric, FIR;
- has zeros at $(f_N, -)$, $(-, f_N)$; and
- has 6 bits of output resolution

where (f_x, f_y) is the spatial frequency vector, and f_N is the Nyquist frequency in each direction. The filter is small and has integer coefficients for computational efficiency. It is symmetric and FIR for zero phase [10]. The zeros at the band edges suppress strong tones that are common in error diffused halftones [11]. To measure the output resolution R, we compute the filter output for each of the possible 2^M binary outputs, count the number of distinct outputs N, and use $R = \log_2(N)$.

We found empirically that a 6-bit wordlength was needed to avoid pixel clumping in error diffusion halftones. This agrees with Hunt's findings [12]. Error diffusion produces an output equal to the input at pixels where the input is 0 or 1, because the feedback error from the quantizer is never large enough to force the input to cross the quantization threshold under those conditions. For any input image, the output is pre-determined to be 0 when the input is 0, and 1 when the input is 1. An image quantized to a short wordlength has a greater proportion of pixels with values 0 or 1 than if a longer wordlength is used. Thus, error diffusion has less freedom to disperse output pixels for short wordlength inputs, which leads to pixel clumping and the consequent visible artifacts.

To obtain halftones that are not blurred or sharpened with respect to the original image, we require that the system frequency response be as flat as possible. We use a modified error diffusion method due to Eschbach and Knox [4] that allows simple frequency response tailoring using a single multiplicative parameter. As shown in the rightmost dashed box of Fig. 1, the frequency response of the halftoning system is modified according to

$$1 + L(1 - H(\mathbf{z})) \tag{1}$$

where L is an adjustable scalar parameter. The halftone

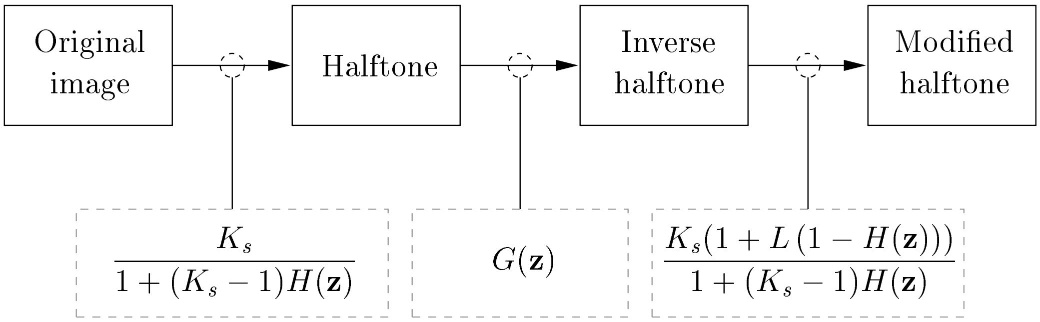


Fig. 1. Block diagram of the rehalftoning system, showing the effective signal transfer functions of each step. K_s and $H(\mathbf{z})$ are the effective signal gain of the quantizer and the z-transform of the error filter, respectively, and may be different for the two halftoning schemes. L is the sharpness parameter that is adjusted for a flat total system response.

is blurred if $L < 0$, and sharpened if $L > 0$. We can therefore compensate for the blurring introduced by the inverse halftoning step by an appropriate choice of L.

For tractability, we analyze the composite response of the rehalftoning method by using the low frequency approximation to the complex digital frequency

$$z = e^{j\omega} \approx 1 + j\omega - \frac{\omega^2}{2} \ , \tag{2}$$

which is obtained by using Euler's series formula $e^x = 1 + x + \frac{x^2}{2!} + \dots$. The expression in (2) is accurate to approximately 10% (real part) and 20% (imaginary part) at $\omega = 1$ radian/sample. Since most of the energy in natural images falls in the lower spatial frequencies, and noise power from halftoning swamps image noise at higher frequencies, the use of (2) is justified. We choose the free parameter L in Fig. 1 to give the flattest low-frequency response for the entire system. When the Floyd-Steinberg error filter is used in both halftoning stages, we obtain (after considerable algebra)

$$\begin{aligned} L &= 0.014 \quad (x \text{ direction}) \\ L &= 0.361 \quad (y \text{ direction}) \ . \end{aligned} \tag{3}$$

The optimum value of L is slightly different for the x and y directions because of the asymmetry in the halftoning error filter imposed by causality constraints. We take the average of the two values ($L = 0.188$) as the optimal L. Typically, different halftoning schemes (with different error filters) will be used for the two halftoning stages. The optimum value of L must then be re-computed using the values of K_s and $H(\mathbf{z})$ for the two halftoning schemes. However, the analysis is identical.

B. Results

Fig. 2(a) shows the original *food* image. Fig. 2(b) shows the result of rehalftoning this image, i.e. halftoning with Floyd-Steinberg error diffusion, inverse halftoning with a linear lowpass filter, and halftoning with modified error diffusion with the value of L chosen for flat low-frequency response. The resulting halftone, shown in Fig. 2(b), has a very similar sharpness to the original image, and is of high quality. The system transfer function, shown in Fig. 2(c), is largely flat at low frequencies, as expected. Because a compromise value for the sharpness parameter L has been used, the midband response rises slightly in the x direction, and falls slightly in the y direction.

Max. ω (cyc/deg)	WSNR (dB)	
	Rehalftone	Halftone
20	11.0	11.3
40	21.7	22.5
60	28.2	29.7
80	32.0	33.9

TABLE I
WSNR measurements of rehalftone of Fig. 2(b), compared to direct halftone, for four viewing distances. The 'Rehalftone' column is the WSNR of the rehalftone relative to the blurred original. The 'Halftone' column is the WSNR of the direct halftone relative to the original.

We used a WSNR measure [13] to assess the subjective quality of the rehalftones produced by this method. Before applying WSNR, we remove the signal-dependent correlation between the original image and the processed image by modeling the process used to create the modified image. To assess the quality of rehalftones, we form a model inverse halftone by filtering the *original* image with the inverse halftoning filter $G(\mathbf{z})$. The difference between this image and the rehalftone consists of noise with a very low linear correlation to the original image. This uncorrelated noise is then weighted according to a human visual system (HVS) model, to give a WSNR figure of merit for the image. The perceptibility of the noise depends on the maximum angular frequency ω subtended at the eye by the image, which in turn depends on its size and resolution, and its distance from the viewer. The response of the HVS falls to zero at $\omega \approx 60$ cycles/degree [14]. Table I indicates that at normal viewing distances, the subjective difference between the original halftone and the rehalftone is small. We obtained similar figures for other images.

(a) Original image. (b) Rehalftone. (c) System transfer function.

Fig. 2. Rehalftoning results using the system shown in Fig. 1. The images are 512×512 pixels in size. The sharpness parameter L of the modified error diffusion method is chosen to give the system the flattest low-frequency response. The system frequency response rises slightly in the x direction and falls slightly in the y direction, because of the asymmetry in the Floyd-Steinberg error filter.

C. Computational requirements

The inverse halftoning portion of the rehalftoning algorithm has a low complexity, since it consists solely of a small, fixed FIR filter. Only four rows of the image need to be stored in memory. The computational requirement of error diffusion is also small. Computation is further reduced by exploiting the fact that some operations are common to both parts of the algorithm, such as looping over the image and writing results to the output. Per pixel, the rehalftoning algorithm requires

- 34 increments (++)
- 12–28 integer additions
- 4 integer multiplications
- 2 bit shifts

The number of additions varies according to the input, and is equal to 20 on average for a mid-gray image. For an image of size 512×512 pixels, the entire rehalftoning process requires approximately 16 million operations. The C implementation takes less than 0.4 seconds to execute on a 167 MHz Sun Ultra-2 workstation for a 512×512 halftone. For c columns, this implementation requires $4(c+3)$ bytes of storage, i.e. 2060 bytes for a 512×512 image. Because all operations are local and use integer arithmetic, the algorithm is ideal for implementation in embedded software.

IV. Interpolated halftoning

The choice of interpolation method depends on the required quality of the resulting image, and the computation power available. If an image is intended for printing, it may not be necessary to perform a computationally expensive interpolation, since improvements in the resulting image will probably be masked by the halftoning process. A simple interpolation method may blur images, but modified error diffusion can then be used to re-sharpen them. This allows high quality, interpolated halftones to be produced at low computational cost. We focus on bilinear interpolation because of its simplicity and good results.

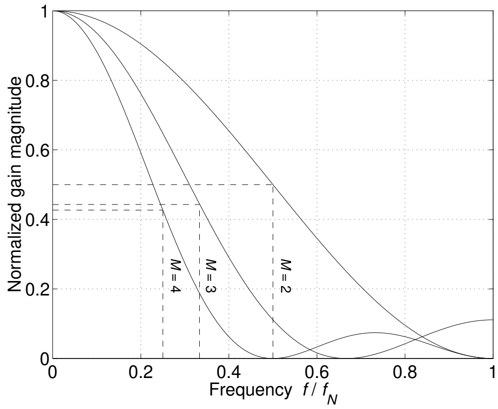


Fig. 3. Frequency responses of linear interpolation functions, for upsampling ratios of 2, 3, and 4. The passband edges and their associated gains are shown as dashed lines.

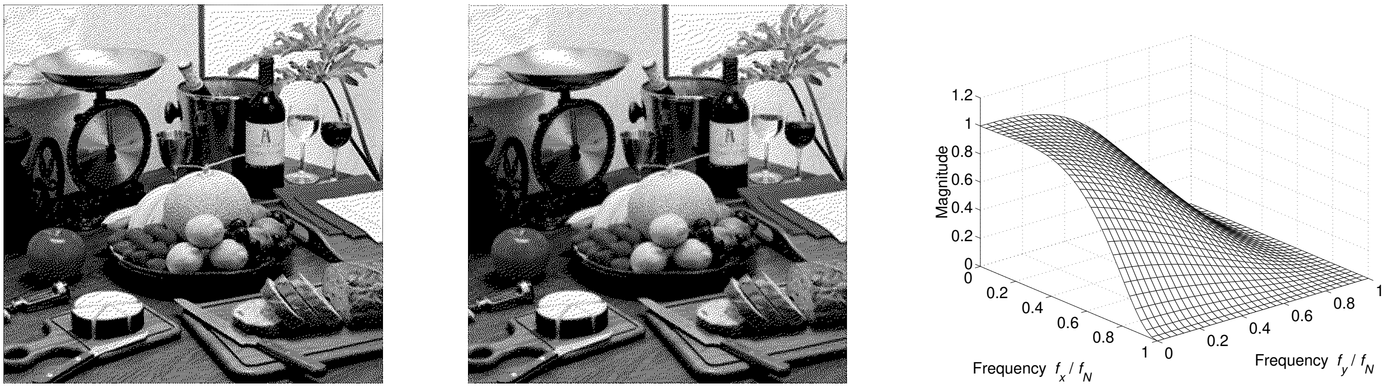
A. Algorithm

The frequency response of the one-dimensional linear interpolator with upsampling ratio M is given by

$$H_{\mathrm{LI}}(e^{j\omega}) = \left(\frac{1 - e^{-jM\omega}}{1 - e^{j\omega}}\right)^2 . \tag{4}$$

Fig. 3 plots (4) for $M = \{2, 3, 4\}$. The normalized response at the passband edge is 0.5 (-6 dB) for $M = 2$, and $4/\pi^2 \approx 0.405$ (-7.8 dB) in the limit as M becomes large. When the bilinear interpolator is applied separably in two dimensions, the gain at the passband edge is -12 dB for $M = 2$, and -15.7 dB in the limit as M becomes large. This causes blurring in the interpolated image.

We consider an interpolated halftoning system of the form in Fig. 1 and substitute the interpolation function for $G(\mathbf{z})$. We assume that $M = 2$, i.e. image size is doubled. The analysis is analogous for other scaling factors. We use the low frequency approximation of (2) to analyze the compound system. A value for the sharpness parameter L is chosen that flattens the response of the system at low frequencies, leading to accurate halftones.



(a) Halftoned, filtered image. (b) Bilinear interpolation. (c) System transfer function.

Fig. 4. Interpolated halftoning results on the *food* image. The original image (512×512 pixels, Fig. 2(a)) is halfband filtered and subsampled by 2 in each direction, bilinearly interpolated, and halftoned using modified error diffusion. The sharpness parameter L is chosen to flatten the low-frequency response of the system. An asymmetry in the x and y directions similar to that of Fig. 2(c) can be seen.

B. Results

We created an image of size 256×256 pixels by filtering and subsampling a 512×512 original. The smaller image was then scaled up by a factor of two and interpolated to obtain a 512×512 approximation to the original image. Since spectral energy above $f_N/2$ in the original image is lost in the 256×256 image, and cannot be recovered by any interpolation scheme, the interpolated image looks blurred with respect to the original. We create a halfband filtered version of the original image for comparison.

Fig. 4(a) shows the halftone created from the halfband filtered image, and Fig. 4(b) shows the corresponding interpolated halftone. They appear subjectively similar, as expected. Fig. 4(c) shows that the system transfer function is indeed flat at low frequencies. Computational cost is low because of the simple interpolation method used.

C. Computational requirements

The computational efficiency of interpolated halftoning stems from the use of a simple interpolation method. Bilinear interpolation requires 7 additions and 6 multiplications per output pixel, which reduces to an average of 1.67 additions and 1 bit shift per output pixel for interpolation by two. For this factor, the algorithm requires

- 2 increments (++)
- 9.67 integer additions
- 4 integer multiplications
- 3 bit shifts

per output pixel, and two image rows of storage. This algorithm is suited for implementation in embedded software.

V. Conclusions

Rehalftoning is necessary when scanning, processing, and reprinting documents, and interpolated halftoning is crucial for resizing images for printing and copying. We develop and optimize new, fast algorithms for rehalftoning and interpolated halftoning for error diffused halftones. The algorithms use local memory and integer arithmetic, and are therefore well-suited to the embedded hardware and software of printers and copiers. For both algorithms, we derive optimum values for the sharpness parameter in modified error diffusion to flatten the system transfer function. The interpolated halftoning method obtains high quality reproduction with only bilinear interpolation. A C implementation of the algorithms presented here is available at `http://www.ece.utexas.edu/~bevans/projects/inverseHalftoning.html`.

References

[1] T. Kite, N. Damera-Venkata, B. Evans, and A. Bovik, "A high quality, fast inverse halftoning algorithm for error diffused halftones," *Proc. IEEE Conf. Image Processing*, vol. 2, pp. 59–63, Oct. 1998.

[2] J. Luo, R. de Queiroz, and Z. Fan, "A robust technique for image descreening based on the wavelet transform," *IEEE Trans. Signal Processing*, vol. 46, pp. 1179–1194, Apr. 1998.

[3] M. Ting and E. Riskin, "Error-diffused image compression using a binary-to-grayscale decoder and predictive pruned tree-structured vector quantization," *IEEE Trans. Image Processing*, vol. 3, pp. 854–858, Nov. 1994.

[4] R. Eschbach and K. Knox, "Error-diffusion algorithm with edge enhancement," *J. Opt. Soc. Am. A*, vol. 8, pp. 1844–1850, Dec. 1991.

[5] T. Kite, B. L. Evans, A. C. Bovik, and T. Sculley, "Digital halftoning as 2-D delta-sigma modulation," *Proc. IEEE Conf. Image Processing*, vol. 1, pp. 799–802, Oct. 1997.

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

[7] T. Bernard, "From Σ-Δ modulation to digital halftoning of images," *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, pp. 2805–2808, May 1991.

[8] R. Gonzalez and R. Woods, *Digital Image Processing.* Reading, MA: Addison-Wesley, 1993.

[9] P. Wong, "Adaptive error diffusion and its application in multiresolution rendering," *IEEE Trans. Image Processing*, vol. 5, pp. 1184–1196, July 1996.

[10] T. Huang, J. Burnett, and A. Deczky, "The importance of phase in image processing filters," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 23, pp. 529–542, Dec. 1975.

[11] S. Norsworthy, R. Schreier, and G. Temes, eds., *Delta-Sigma Data Converters.* New York, NY: IEEE Press, 1997.

[12] R. Hunt, *The Reproduction of Colour.* Fountain Press, 5th ed., 1996.

[13] N. Damera-Venkata, T. Kite, W. Geisler, B. Evans, and A. Bovik, "Image quality assessment based on a degradation model," *IEEE Trans. Image Processing*, to appear.

[14] T. Cornsweet, *Visual Perception.* New York, NY: Academic Press, 1970.