Proc. SPIE Vol. 2179 Human Vision, Visual Processing, and Digital Display V, (San Jose, CA), pp. 182–194, Feb. 1994.

Blue noise and model-based halftoning

Mark A. Schulze [1]

Department of Electrical and Computer Engineering and Biomedical Engineering Program
The University of Texas at Austin, Austin, Texas 78712

Thrasyvoulos N. Pappas

Signal Processing Research Department
AT&T Bell Laboratories, Murray Hill, New Jersey 07974

ABSTRACT

"Model-based" halftoning techniques use models of visual perception and printing to produce high quality images using standard laser printers. Two such model-based techniques are the modified error diffusion (MED) algorithm and the least-squares model-based (LSMB) algorithm. Both produce images with high spatial resolution and visually pleasant textures, but require more computation than conventional screening techniques.

Blue-noise screening is a dispersed-dot ordered dither technique which attempts to approximate the performance of error diffusion with much faster execution time. We use printer and visual system models to improve the design of blue-noise screens using the "void-and-cluster" method. We show that, even with these improvements, the performance of blue-noise screens does not match that of the model-based techniques.

We show that using blue-noise screened images as the starting point of the LSMB algorithm results in halftones inferior to those obtained with MED starting points. We also use simulated annealing to try to find the global optimum of the least-squares problem. Images found this way do not have significantly lower error than that resulting from the simple iterative LSMB technique starting with MED. This result indicates that the simple iterative LSMB algorithm with a MED starting point yields a solution close to the globally optimal solution of the least-squares problem.

1. INTRODUCTION

Digital halftoning is the process of generating a pattern of binary pixels that the eye perceives as a continuous-tone image. Digital halftoning is necessary for display of gray-scale images in media where the direct rendition of gray tones is impossible. Examples of such media include paper and binary CRT displays.

The most commonly used digital halftoning techniques are screening (ordered dither) and error diffusion [1]. In screening, the image is compared pixel by pixel with a fixed threshold array. Thus, the required amount of computation is minimal. Moreover, the computation can be carried out in parallel. Dispersed-dot screening techniques produce images with better spatial resolution and fewer halftoning artifacts than clustered-dot techniques, but are more sensitive to printer distortions. Error diffusion [2] produces images that are sharper and have better textures than those produced by conventional screening techniques. However, in error diffusion, the output at each pixel depends on previously computed values. Thus, it requires several operations per pixel which cannot be performed in parallel. The standard error diffusion algorithm is also very sensitive to printer distortions.

Blue-noise screening attempts to approximate the performance of error diffusion with much faster execution time. Two such techniques are the power-spectrum matching technique [3] and the "void-and-cluster" method [4]. As we will see, it is a dispersed-dot ordered dither technique which attempts to minimize the visibility of noise and patterns in the halftone image. Like all dispersed-dot techniques, blue-noise screening is very sensitive to printer distortions.

Recently proposed "model-based" halftoning techniques [5, 6, 7, 8] exploit models of visual perception and printing to produce high quality images using standard laser printers. One such model-based technique is the modified error

[1] Consultant at AT&T Bell Laboratories, Summer 1993.

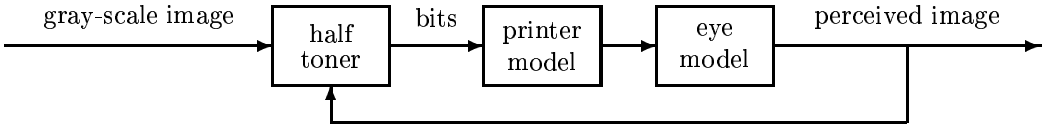


Figure 1: Model-based halftoning

diffusion (MED) algorithm [5, 6], which uses a printer model to account for printer distortions. Another technique is the least-squares model-based (LSMB) algorithm [7], which minimizes the squared error between the perceived intensity of the continuous-tone original image and the perceived intensity of the printed halftone image. Both techniques produce images with high spatial resolution and visually pleasant textures, but require considerably more computation than conventional screening techniques.

In this paper we propose using the ideas of model-based halftoning to improve the design of blue-noise screens. Specifically, we use models of the human visual system to assess the perceived "flatness" of the halftone patterns made at different gray levels when designing a screen. We also use a printer model to account for dot-overlap and other distortions introduced by laser printers. We compare the performance of model-based techniques and blue-noise screening both visually and using the error criterion of the LSMB algorithm. We show that, even with the above improvements, the performance of blue-noise screens does not match the quality of the model-based techniques, especially in the presence of significant dot-overlap. In particular, MED produces images that are sharper and have better textures than those produced by blue-noise screening.

The globally optimum solution of the LSMB problem is difficult to find, because the number of all possible halftones is too large. Least-squares techniques rely on iterative procedures that find a local optimum. Thus, the visual quality of the halftone images obtained by the LSMB algorithm depends strongly on the starting point. The halftones obtained with a MED starting point are typically better than those of most other starting points, both visually and according to the error criterion. We use the halftone image produced by the blue-noise screen (with a printer model) as a starting point for the LSMB algorithm and show that the resulting images are still inferior to those obtained with the MED starting point. Since the sharpness of the LSMB halftones does not depend on the starting point, this demonstrates that the blue-noise screen produces textures that are inferior to those produced by error diffusion.

We also use simulated annealing in an attempt to obtain the global optimum of the least-squares problem. This technique allows some increases in the squared error in early iterations, steadily decreasing the number of increases allowed until a minimum is reached. We found that simulated annealing results in a minimum error that for most images is not significantly lower than the minimum error found using the simple iterative technique starting with MED. Thus, our results indicate that the simple iterative LSMB algorithm with a MED starting point is very close to the globally optimal solution of the least-squares problem.

The remainder of this paper is organized as follows. Section 2 is a review of model-based halftoning techniques. In Section 3, we examine blue-noise screens and model-based extensions. Section 4 considers using blue-noise screens as starting points in LSMB halftoning, as well as simulated annealing solutions to the LSMB problem. The conclusions are summarized in Section 5.

2. MODEL-BASED HALFTONING

A halftoning algorithm generates a binary pattern of pixels which is printed and perceived by the eye. Model-based halftoning techniques exploit models of the printer (or other display device) and visual perception to produce high quality images. This is illustrated in Fig. 1.

2.1. Eye models

Halftoning relies on the fact that the eye acts as a spatial low-pass filter. The eye models we consider in this paper are based on estimates of the spatial frequency sensitivity of the eye, often called the modulation transfer function (MTF). A typical estimate of the MTF was obtained by Mannos and Sakrison [9]

$$H(f) = 2.6\,(0.0192 + 0.114\,f)\,\exp\left\{-(0.114\,f)^{1.1}\right\} \tag{1}$$

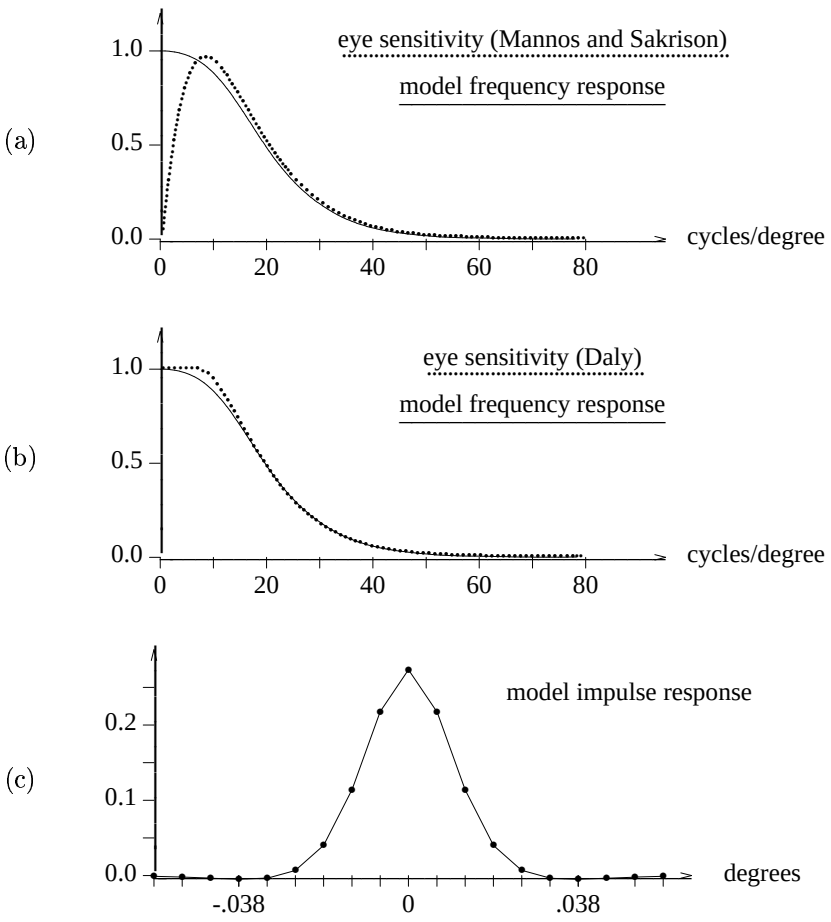


Figure 2: Eye MTF and FIR approximation

where f is in cycles/degree. This function is plotted in Fig. 2(a) by the dotted line.

A simple eye model [7] consists of a two-dimensional finite impulse response (FIR) filter. This model was obtained as a separable combination of one-dimensional approximations to the MTF of Eq. (1). The frequency response of the one-dimensional filter used in [7] is shown in Fig. 2(a) and (b) as a solid line; the impulse response is shown in Fig. 2(c). Note that the frequency response of this filter does not decrease at low frequencies, that is, it does not model the Mach-band effect. However, modeling of the Mach-band effect does not have an effect on halftoning algorithms that are based on an eye model [7]. In fact, the frequency response of this filter is very close to the MTF proposed by Daly [10, 11], which is plotted in Fig. 2(b) as a dotted line.

2.2. Printer models

Laser printers are capable of producing black dots on a piece of paper, usually on a rectangular grid. Most halftoning techniques assume that the printed black dots are square. However, most printers produce roughly circular black dots, as shown in Fig. 3. Thus, there is overlap between adjacent dots, and black dots cover adjacent space that should be white. This results in significant distortion in the printed images. Conventional methods, e.g. "classical" screening, are designed to be fairly robust to printer distortions at the expense of spatial and gray-scale resolution. Rather than trying to resist printer distortions, model-based techniques [5, 7] exploit the characteristics of each particular printer to increase both gray-scale and spatial resolution.

In this paper we will use the following notation. The printer is controlled by an $N_W \times N_H$ binary array $[b_{i,j}]$, where $b_{i,j} = 1$ indicates that an ink dot is to be printed at pixel (i, j) and $b_{i,j} = 0$ means that no ink dot is to be printed.

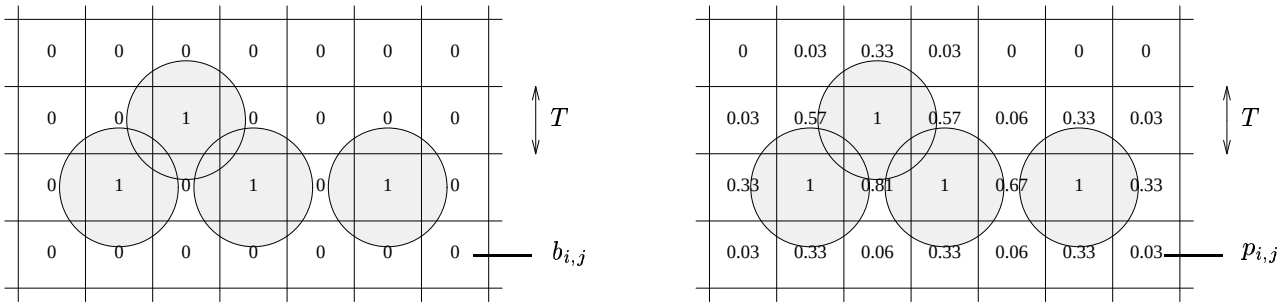


Figure 3: Circular dot-overlap model

The pixel (i, j) is located iT inches from the left and jT inches from the top of the image, where T is the dot pitch (in inches) of the printer. The resolution of the printer is $1/T$ dots per inch (dpi).

Model-based halftoning techniques depend on accurate printer models to exploit the printer characteristics. A general class of such models was introduced in [5, 6]. According to this model, the gray level produced by the printer in the vicinity of site (i, j) has a constant value $p_{i,j}$ that depends on $b_{i,j}$ and neighboring bits. Thus, the printer model takes the form

$$p_{i,j} = \mathcal{P}(W_{i,j}) \tag{2}$$

where $W_{i,j}$ denotes the bits in some neighborhood of $b_{i,j}$ and $\mathcal{P}$ is some function thereof. A simple but very effective printer model is the "circular dot-overlap" model [5, 6] which assumes that the dots are circular as shown in Fig. 3. Alternative printer models can be found in [12, 13].

We now review two model-based techniques, the *modified error diffusion (MED)* algorithm [5, 6] and the *least-squares model-based (LSMB)* algorithm [7, 8]. We will use the following notation. We use $[x_{i,j}]$ to denote a gray-scale image, where $x_{i,j}$ denotes the pixel located at the i-th column and the j-th row of a Cartesian grid. The typical gray-scale resolution is $2^8 = 256$ levels; we use a normalized scale in the interval from 0 = white to 1 = black. We assume that the image has been sampled so there is one pixel per dot to be generated. Thus the gray-scale image array $[x_{i,j}]$ and the binary image array $[b_{i,j}]$ have the same dimensions. In our experiments, we used a 300 dpi HP LaserJet II write-black printer and a 400 dpi Data Products LZR 1560 write-black printer.

2.3. Modified error diffusion

Error diffusion [2] is a halftoning technique that produces sharper images than conventional screening techniques. The standard error diffusion algorithm is very sensitive to printer distortions, however. Stucki [14] was the first to suggest using a dot-overlap model to account for printer distortions. In [5, 6] Pappas and Neuhoff showed that, by incorporating a printer model into error diffusion, it is possible not only to correct for the effects of printer distortions but also to take advantage of them to produce more gray levels. We refer to the resulting algorithm as modified error diffusion.

A block diagram of the MED algorithm [5, 6] is shown in Fig. 4. Without loss of generality, we assume that the image is scanned left to right, top to bottom. The binary image $[b_{i,j}]$ is obtained by thresholding a "corrected" value $v_{i,j}$ of the gray-scale image. The MED algorithm uses a printer model to estimate the gray level $p_{i,j}$ of the printed pixels. The difference between this gray level and the "corrected" gray scale image is defined as the error $e_{i,j}$ at the location (i, j).[2] Previous errors are low-pass filtered and subtracted from the current image value $x_{i,j}$ to obtain the "corrected" value of the gray scale image. The threshold t is typically fixed at 0.5, the middle of the gray-scale range. In the examples of this paper we use the error diffusion filter proposed by Jarvis, Judice and Ninke [15].

The modified error diffusion equations are [5, 6]:

[2] In the standard error diffusion algorithm, the errors are calculated under the assumption that the printed pixels are either perfectly black or perfectly white, and thus their gray value is equal to the binary value assigned to the output pixels.

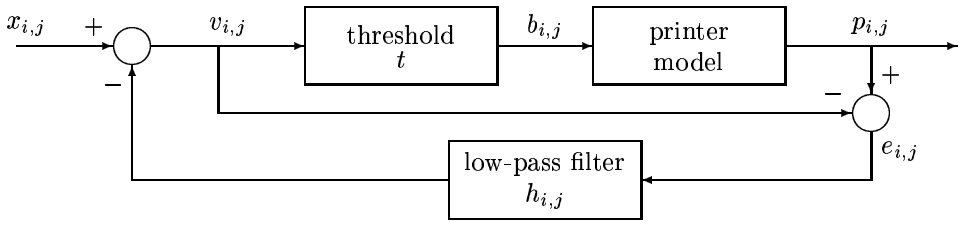


Figure 4: Modified error diffusion

$$v_{i,j} = x_{i,j} - \sum_{m,n} h_{m,n}\, e^{i,j}_{i-m,j-n} \tag{3}$$

$$b_{i,j} = \begin{cases} 1, & \text{if } v_{i,j} > t \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

$$e^{i,j}_{m,n} = p^{i,j}_{m,n} - v_{m,n} \qquad \text{for } (m,n) \prec (i,j) \tag{5}$$

where $(m,n) \prec (i,j)$ means (m,n) precedes (i,j) in the scanning order and

$$p^{i,j}_{m,n} = \mathcal{P}(W^{i,j}_{m,n}) \qquad \text{for } (m,n) \prec (i,j) \tag{6}$$

where $W^{i,j}_{m,n}$ consists of $b_{m,n}$ and its neighbors as in Eq. (2). Here the neighbors $b_{k,l}$ have been determined only for $(k,l) \prec (i,j)$; they are assumed to be zero (i.e. white) for $(k,l) \succeq (i,j)$ [5, 6]. Since only the dot-overlap contributions of the previous pixels can be used in (6), the previous errors keep getting updated as more binary values are computed. This is why the error and the printer model output depend on the location (i,j). The assumption that the undetermined pixels are white leads to a bias in the gray scale of the printed image. This bias is very small and difficult to detect in practice. However, to eliminate any bias, we use the *multi-pass* version of the modified error diffusion algorithm [12, 13].

2.4. Least-squares model-based halftoning

LSMB halftoning exploits both a printer model and a model of visual perception. It produces an "optimal" halftoned reproduction, by minimizing the squared error between the output of the cascade of the printer and visual models in response to the binary image and the output of the visual model in response to the original gray-scale image. This is illustrated in Fig. 5. The original gray-scale image is denoted by $[x_{i,j}]$. We seek the halftone image $[b_{i,j}]$ that minimizes the squared error

$$E = \sum_{i,j} (z_{i,j} - w_{i,j})^2 \tag{7}$$

where

$$z_{i,j} = x_{i,j} * h'_{i,j}, \qquad w_{i,j} = p_{i,j} * h_{i,j} = \mathcal{P}(W_{i,j}) * h_{i,j}, \tag{8}$$

$W_{i,j}$ consists of $b_{i,j}$ and its neighbors as in Eq. (2), and $*$ indicates convolution. Note that we have allowed different impulse responses $h_{i,j}$, $h'_{i,j}$ for the eye filters corresponding to the halftone and continuous-tone images. In fact, we found that when we do not filter the gray-scale image, the resulting halftone images are sharper. The boundary conditions assume that no ink is placed outside the image borders.

The least-squares solution can be obtained by iterative techniques [7]. Such techniques find a solution that is only a local optimum; the visual quality of the resulting halftone images depends strongly on the starting point. In [7], it was shown that, with an eye filter designed for a viewing distance of 30 inches and a printer resolution of 300 dpi, the starting point obtained using the multi-pass MED algorithm produces halftone images that are better both visually and according to the error criterion. In Section 4, we will consider alternative starting points. In this paper, we will use the iterative scheme that was used in [7], whereby the binary image is updated one pixel at a time, in a raster scan. At each pixel location this scheme minimizes the squared error

$$E_{i,j} = (z_{i,j} - w_{i,j})^2 \tag{9}$$

We will refer to this as the *simple iterative* scheme. In Section 4, we will also attempt to find the global optimum of the least-squares problem using simulated annealing.

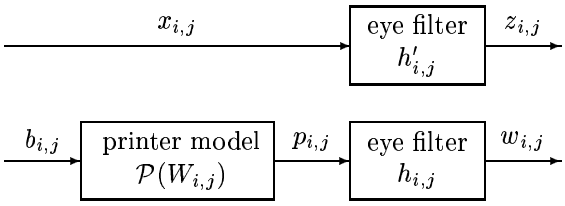


Figure 5: LSMB halftoning

2.5. Halftoning quality measure

Given an original gray-scale image $[x_{i,j}]$, a halftone image $[b_{i,j}]$, a printer model, and an eye model, we can define the following measure of halftone image quality

$$E = \frac{1}{N} \sum_{(i,j)\in\mathcal{A}} (z_{i,j} - w_{i,j})^2 \tag{10}$$

where $z_{i,j}$ and $w_{i,j}$ are given by Eq. (8) and $\mathcal{A}$ is a subset of the image containing N pixels. This is the average of the squared error that the LSMB algorithm minimizes. Usually $\mathcal{A}$ does not include the boundary pixels to avoid biases due to edge artifacts.

Note that this image quality metric takes into account the characteristics of both the display device (printer) and the human visual system. Thus, it is expected to be a much better measure of halftone image quality than plain least-squares, which does not yield meaningful results when comparing binary (halftoned) images to 8-bit original images. However, as we will see in Section 4, the halftone quality measure still has limitations, since it is very difficult to capture the quality of an image using a single number. More elaborate models have been proposed, e.g. in [16].

3. BLUE-NOISE SCREENS

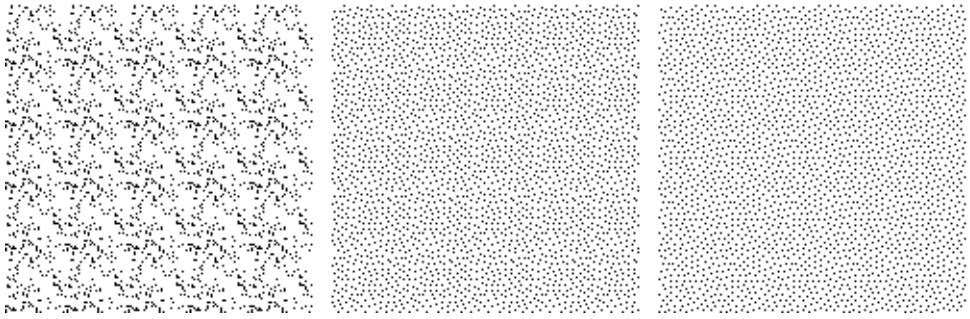
Blue-noise screening attempts to approximate the performance of error diffusion with much faster execution time. It is a dispersed-dot ordered dither technique which attempts to minimize the visibility of noise and patterns in the halftone image. First, we examine the *void-and-cluster* method in detail.

3.1. Overview of the void-and-cluster method

The void-and-cluster method for designing halftoning screens was developed by Ulichney [4]. This technique builds a screen one level at a time by adding a point to the screen in the center of the largest "void" present at any given level, or removing a point from the screen at the center of the tightest "cluster." The qualities of the screen depend heavily on the "initial binary pattern," the original pattern to which pixels are added or from which they are removed.

In [4], Ulichney forms the initial binary pattern by starting with a random pattern where 10% of the pixels are black, and the rest white. This pattern is then rearranged by successively adding a pixel to the largest void and then removing a pixel from the tightest cluster. This process is repeated until the pixel added to the largest void is immediately removed because it becomes the center of the tightest cluster. Ulichney determines the largest void by finding the white pixel location that has the highest (lightest) value after filtering the pattern with a two-dimensional Gaussian filter with standard deviation $\sigma = 1.5$ pixels. The tightest cluster, similarly, is defined as the black pixel location with the smallest (darkest) value after Gaussian filtering.

The rearranged initial binary pattern is then used to form a screen by iteratively adding black pixels. Pixels are added to the pattern one at a time at the largest void, and the threshold value of the screen at the corresponding location is the number of black pixels in the current pattern. This process is repeated until all pixels are assigned to black. The thresholds below the value of the initial binary pattern are determined by removing pixels from the initial binary pattern one at a time from the center of the tightest cluster, and assigning to that pixel location in the screen the appropriate threshold. The threshold values in the screen must be normalized to the appropriate grayscale for the images it operates on before it is used for halftoning. Actually, Ulichney inverts the pattern when the number of



(a) random start (b) Ulichney's method (c) simulated annealing

Figure 6: 32×32 initial binary patterns (no dot overlap)

black pixels becomes equal to the number of white pixels, and continues building the screen by removing black pixels instead of adding them. This is equivalent to continuing to add black pixels to the largest void because the Gaussian operator is linear; finding the location of the minimum of the filtered image is the same as finding the location of the maximum of the filtered values of the inverted image.

3.2. Initial binary pattern

An example of an initial binary pattern obtained by Ulichney's rearrangement method is shown in Fig. 6, which is printed at 100 dpi and assumes perfect printing (no dot overlap). Although the perceived gray level of this pattern is fairly constant, the periodic pattern of the screen is noticeable. In an attempt to improve the initial binary pattern used for blue-noise screen generation, we developed an algorithm using simulated annealing for finding the initial pattern based upon the work of Sullivan, Ray, and Miller [11]. Periodicities in the resulting pattern, also shown in Fig. 6, are significantly less noticeable than for the pattern rearranged by Ulichney's method.

The simulated annealing algorithm tries to match the perceived gray level of the pattern as closely as possible to a constant level that is chosen in advance. Beginning with a random pattern, the algorithm filters the image with either a Gaussian filter or eye model (see Section 3.3 below). At each point, the "current" error between the filtered (perceived) image and the desired constant image is computed, and the "new" error is computed for the case where the value of the current pixel is switched to its opposite. The difference between the new and old errors, $\Delta = (\text{new\_error} - \text{old\_error})$, is used to compute the value $q = \exp(-\Delta/\tau)$, where τ represents a "temperature" factor that is initially set quite large so that q is initially small for most values of Δ. If the new error is less than the old error ($\Delta < 0$ and therefore $q > 1$), the pixel is changed. Otherwise, a uniformly-distributed random number between zero and one is generated; if it is less than or equal to q, the change is accepted, and if it is greater than q, the change is rejected. This process is repeated for every pixel in the pattern, and then the entire process is repeated with the temperature decreased to $\kappa\tau$, where κ is a constant less than 1 (usually $0.95 \leq \kappa \leq 0.99$). At each iteration, pixel changes that increase the error between the filtered and the constant image are less likely to be accepted. The process is stopped when the number of changes made to the pattern during one iteration is zero. The values of τ and κ used to create the pattern in Fig. 6 are $\tau = 75$ and $\kappa = 0.95$.

3.3. Model-based blue-noise screens

We will now use the ideas of model-based halftoning to optimize the design of blue-noise screens (BNS). Specifically, we propose using the printer and visual system models to optimize the design of blue-noise screens.

In [4], Ulichney uses a two-dimensional Gaussian filter with $\sigma = 1.5$ as a "void-and-cluster-finding filter." The goal of his algorithm is to find the largest voids and tightest clusters, which are presumably the areas in a pattern that appear lightest and darkest, respectively. Another way to find these locations is to use the eye model described in Section 2, which is a two-dimensional FIR filter. We use this eye model as the filter for finding the initial binary pattern and for designing the screen using the void-and-cluster method. This bases the technique on an established model of visual perception, and provides justification for using the filter to find voids and clusters.

(a) Ulichney's BNS $E = 111.90$ (b) model-based BNS $E = 111.62$ (c) error diffusion $E = 94.83$

Figure 7: Perfect printer case (no dot overlap), 32×32 screens

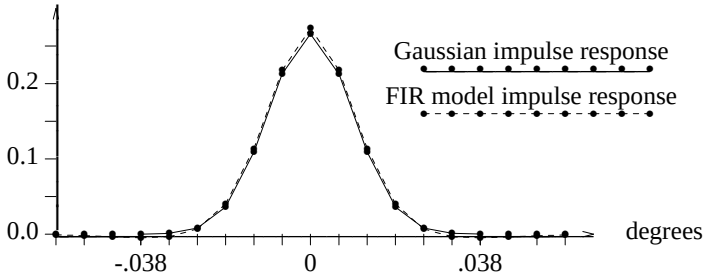


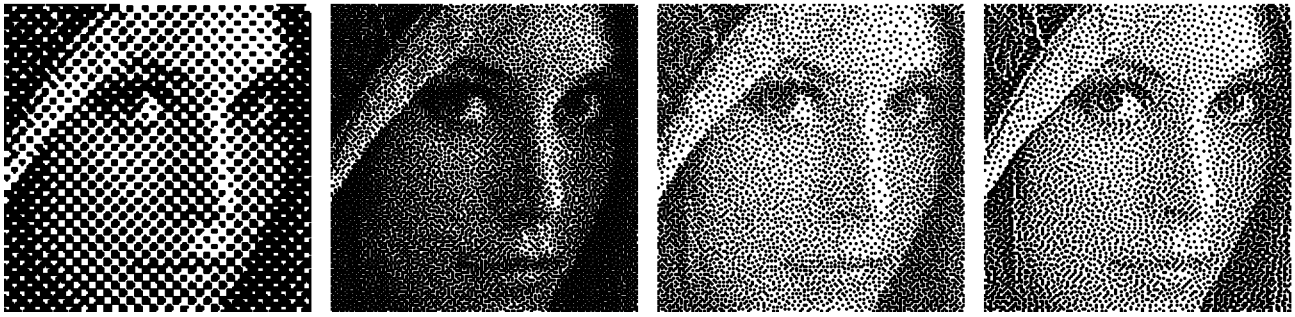
Figure 8: Comparison of Gaussian and FIR eye model

In Fig. 7, images halftoned using 32×32 blue-noise screens designed using Ulichney's method (a) and the eye model with a simulated annealing starting point (b) are compared to a halftone made by error diffusion (c). The test image is a section of the 512×512 "Lena" image. This figure is also printed at 100 dpi and assumes perfect printing. The two screened images are similar, while the error diffusion halftone (as we will discuss below) is much sharper. The halftoning quality measure (see Section 2.5) supports this conclusion, as the screened images have virtually the same value for the measure, while the error diffusion result has a significantly lower value. Fig. 8 illustrates why the results of the eye model based screen and Gaussian filter based screen are not much different. At a viewing distance of 30 inches, the impulse responses of the eye model and the Gaussian filter with $\sigma = 1.5$ are virtually identical.

Fig. 9 demonstrates the effect of printer distortions on various halftoning techniques. It is printed at 100 dpi using simulated dot overlap with dot size $\rho = 1.25$. It approximates the printer distortions of a typical 300 dpi printer magnified by a factor of 3. Fig. 9(a) shows an image that has been halftoned using a "classical" screen [1], which resists printer distortions at the expense of spatial resolution. Thus, this image indicates approximately what the correct gray level appearance should be, although image details are missing. Fig. 9(b) shows an image that has been halftoned using an 128×128 blue-noise screen without any compensation for printer distortions. This image is obviously too dark.

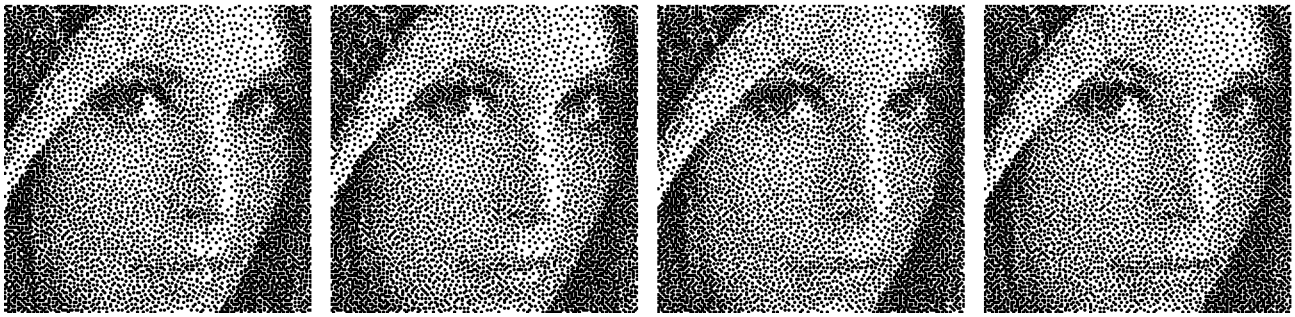
An obvious way to compensate for such printer distortions is to use the printer model of [5, 6] to modify the thresholds of an already designed screen.[3] Roetling and Holladay [17] did this, using a circular dot-overlap model. Actually, Roetling and Holladay went a step further, also using the printer model to optimize screen design. Our goal is similar, that is, to use the printer model of [5, 6] and the eye model of [7] to optimize the design of blue-noise screens using the void-and-cluster method. Note that the printer compensated screens work like regular screens; that is, there is no increase in the number of computations. A different screen is required for each printer, however.

[3]Modifying the screen thresholds is equivalent to modifying the gray levels of the original image to match the scale created by the different binary patterns that a screen produces.



(a) "classical" screen $E = 1802.32$ (b) BNS w/o printer model $E = 6837.92$ (c) model-based BNS $E = 162.87$ (d) MED $E = 112.40$

Figure 9: Actual printer case ($\rho = 1.25$ dot-overlap), 128×128 screens

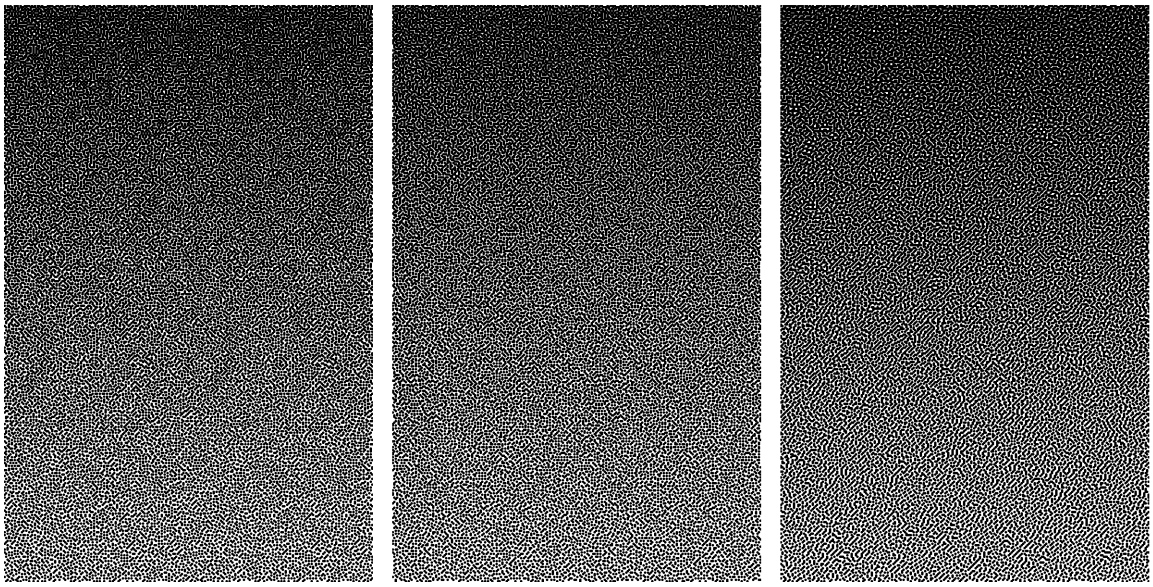


(a) Gaussian, compensated $E = 171.10$ (b) Gaussian, integral $E = 169.96$ (c) eye model, compensated $E = 170.69$ (d) eye model, integral $E = 162.87$

Figure 10: Blue-noise screen comparisons (128×128 screens, $\rho = 1.25$ dot-overlap)

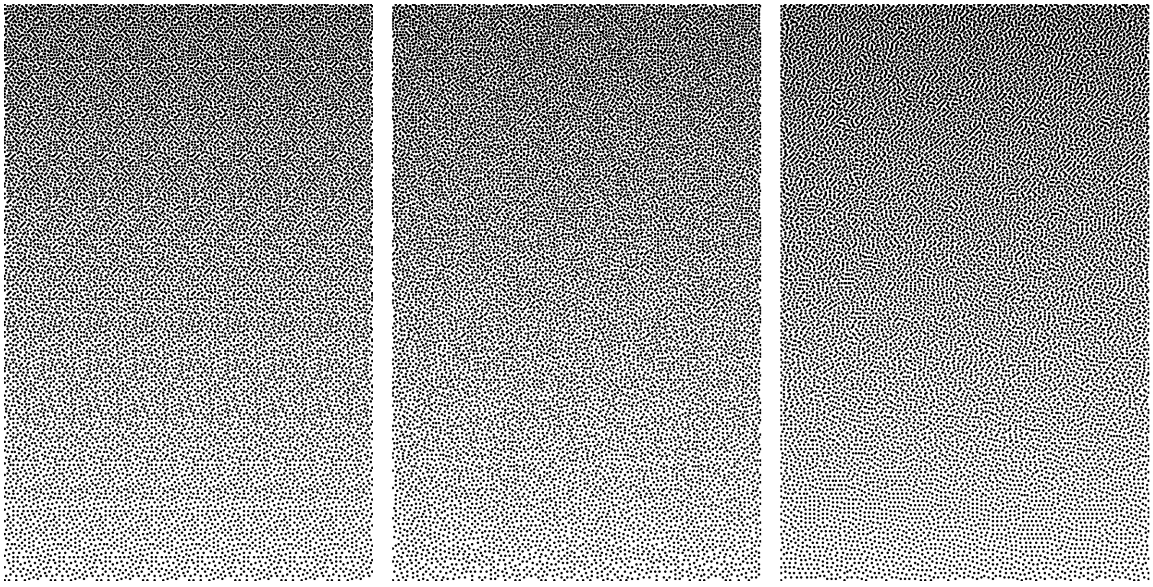
We apply the printer model described in Section 2 to the binary patterns before filtering with the eye model at each step of the void-and-cluster method, and also use the printer model to calculate the correct threshold value for each pattern in the blue-noise screen. We will refer to screens designed in this manner as *model-based blue-noise screens*. Fig. 9(c) is the result of blue-noise screening with a (128×128) screen designed using the printer model with the appropriate dot size. This image is clearly superior to both the classical screen and uncompensated blue-noise screened image; this result is backed up by an order of magnitude improvement in halftone quality measure for the compensated screened image. The MED result, shown in Fig. 9(d), is even better, however, as we discuss below.

Fig. 10 illustrates the difference between using the printer model to modify the thresholds of a blue-noise screen and integrating the printer model into the screen design. This figure is printed at 100 dpi using simulated dot overlap with $\rho = 1.25$. The halftoning quality measure is only slightly lower for the integral screen designs, and the halftoned images are visually almost indistinguishable. This example also shows that it makes little difference whether we use the eye model or the Gaussian filter suggested by Ulichney. The difference between the compensated blue-noise screens and integrating the printer model into the screen design is also illustrated in Fig. 11, which shows a gray-scale ramp halftoned by the two screens and the MED algorithm. This figure is printed at 133 dpi using simulated dot overlap with dot size $\rho = 1.25$. It approximates the printer distortions of a 300 dpi printer magnified by a factor of 2.25. Here, the quality metric indicates that the integral screen design outperforms the compensation technique. Indeed, the image generated using the "integral" screen appears to be smoother than the "compensated" screen image. However, this can only be seen in the magnified figure; at 300 dpi, the textures of the two screened images are nearly identical in appearance. Modified error diffusion creates much smoother textures than screening in this example also, and the quality metric values support this conclusion.



(a) compensated BNS $E = 48.46$ (b) integral BNS $E = 35.61$ (c) multi-pass MED $E = 20.54$

Figure 11: Model-based blue-noise screens and MED ($\rho = 1.25$ dot-overlap)



(a) 32×32 screen $E = 61.55$ (b) 128×128 screen $E = 62.31$ (c) multi-pass MED $E = 25.84$

Figure 12: Model-based blue-noise screens and MED ($\rho = 1.25$ dot-overlap)

3.4. Blue-noise screens and MED

As we saw in Fig. 7, the error diffusion halftones are much sharper than those produced by blue-noise screens. The texture that error diffusion produces is also a lot smoother. This is reflected in the halftoning quality measure of Section 2.5. Fig. 9(d) shows the result of the modified error diffusion algorithm (which also accounts for printer distortions) in the presence of $\rho = 1.25$ dot overlap. Again, the MED result is sharper and has smoother texture than the screened image, and its halftone quality measure is the lowest of the four techniques shown in Fig. 9. Fig. 11 compares the two blue-noise screens to MED. Notice that the MED image has much smoother texture and results in significantly lower (better) value of the quality measure. Since the test image is a gray-scale ramp which has no sharp boundaries, the error criterion reflects this texture difference.

Another difference between blue-noise screening and modified error diffusion is the visibility of artifacts due to the periodic nature of the screens. The visibility of such periodic artifacts depends on the printer resolution and the viewing distance, as well as on image content. An image of a grayscale ramp will often show any artifacts in a screen, as large areas have similar gray levels. Fig. 12 illustrates the periodicity artifacts of a 32×32 and a 128×128 blue-noise screen on a grayscale ramp, and compares them to the MED algorithm, which is free of such artifacts. This figure is printed at 133 dpi using simulated dot overlap ($\rho = 1.25$) and approximates the printer distortions of a 300 dpi printer magnified by a factor of 2.25. Note that the 128×128 screen is sufficiently artifact-free. We found that at 300 dpi, the period of the screen must be at least 128×128 for the artifacts to be unnoticeable. However, at 600 dpi, the artifacts of the 128×128 screen become visible; they could be eliminated with a bigger screen. Unfortunately, the halftone quality metric does not measure the visibility of periodicity artifacts, so it cannot distinguish the difference between the 32×32 and 128×128 screens. However, as we saw in the other examples, the metric does successfully predict the fact that MED creates smoother textures than blue-noise screening.

The above examples demonstrate the superiority of the MED algorithm both visually and in terms of the error criterion. This is expected, since the blue-noise screening algorithm is a lot more constrained than error diffusion. The constraints affect both the smoothness of the texture (i.e. visibility of the halftone patterns) and the sharpness of the image.

4. BLUE-NOISE SCREENS AND LEAST-SQUARES MODEL-BASED HALFTONING

In this section we consider different solutions to the LSMB problem and evaluate their performance both visually and according to the error criterion.

4.1. Starting points for LSMB halftoning

The LSMB halftoning algorithm described in Section 2.4 requires an initial binary image as a starting point. This starting point may be any binary pattern that has the same size as the desired halftone; a random starting point may be used, as may the output of a screening or error diffusion algorithm. Since the LSMB algorithm does not find the global minimum of the least-squares problem, different starting points lead to different solutions (local minima of the problem). Fig. 13 shows the results of the LSMB algorithm (simple iterative scheme) with a random starting point, a blue-noise screen starting point, and a modified error diffusion starting point. It is printed at 100 dpi using simulated dot overlap with dot size $\rho = 1.25$. The LSMB algorithm with a MED starting point results in the least error, while the quality metric values for the random start and blue-noise screen start are significantly higher and very close to each other. Visually, the random and BNS starts look grainier, considerably more at 300 dpi than at the simulated 100 dpi of the figure. This is yet another indication that the halftone textures created by the MED algorithm are better than those of blue-noise screening.

4.2. LSMB halftoning with simulated annealing

The LSMB algorithm finds only a local minimum to the least-squares halftoning problem when using the simple iterative scheme. In an attempt to find the global minimum of the problem, we have used the simulated annealing procedure described in Section 3.2.[4] A similar procedure was used to solve the LS problem in [18]. The results for different starting points are shown in Fig. 14, which is also printed at 100 dpi using simulated dot overlap with

[4] In Section 3.2, the simulated annealing procedure was applied to a constant image; here it is applied to any image.

(a) random $E = 98.39$ (b) model-based BNS $E = 101.98$ (c) multi-pass MED $E = 87.68$

Figure 13: LSMB: simple iterative scheme with different starting points ($\rho = 1.25$ dot-overlap)

(a) random $E = 89.77$ (b) model-based BNS $E = 89.79$ (c) multi-pass MED $E = 83.33$

Figure 14: LSMB: simulated annealing with different starting points ($\rho = 1.25$)

$\rho = 1.25$. The initial temperature was $\tau = 75$ and the decay constant was $\kappa = 0.95$. The three images that result are virtually indistinguishable, and the values of the quality metric are all very close. The quality metric also indicates that the simulated annealing results are better than most of the other halftoning techniques we tried, including the simple iterative LSMB scheme. However, the simple iterative LSMB algorithm with a MED starting point yields almost the same value of the quality metric as the simulated annealing results. This indicates that the simple LSMB with a MED start produces a halftone that is very close to the global optimum of the least-squares halftoning problem, without resorting to computationally expensive annealing methods. Therefore, when we perform least-squares model-based halftoning of an image starting with a modified error diffusion halftone, we can be confident that the results are about the best that can be achieved using the least-squares criterion and the eye and printer models as defined in this paper. Further improvements may be possible with changes in the halftone quality metric and the printer and eye models.

5. CONCLUSIONS

In this paper, we considered ways to refine the performance of blue noise screens for image halftoning by applying printer and visual models to the void-and-cluster screen design process. The printer model enables the design of blue noise screens that convert grayscale images directly to halftones that can be printed without distortion. Although the resulting screens perform quite well, the quality of the screened halftones still does not match that of halftones created by the modified error diffusion algorithm. Since error diffusion is able to adapt to edges, it results in noticeably sharper images than blue noise screening. In most cases the error diffusion textures are also more pleasing than the blue noise screen textures.

We also considered different solutions to the least-squares model-based halftoning problem, which uses models of

the printer and the human visual system together with a quality metric to convert the halftoning problem to a least squares problem. Solving this problem using a simple iterative scheme leads to high-quality halftones, but the quality of these halftones depends strongly on the starting point used for the algorithm. Typically, a modified error diffusion starting point gives the best results. This is another indication that error diffusion textures are superior to blue-noise screen textures. Using simulated annealing, we can approach the globally optimal solution of the least squares problem; we have found that this result is very close to the solution found using the simple iterative scheme and a modified error diffusion starting point. Further progress in halftoning using printer and eye models will require improvements in the models and in the quality metric used to evaluate the distance between the perceived halftone and the original image.

REFERENCES

[1] R. Ulichney, *Digital Halftoning.* Cambridge, MA: The MIT Press, 1987.

[2] R.W. Floyd and L. Steinberg, "An adaptive algorithm for spatial grey scale," in *Proc. SID*, vol. 17/2, pp. 75–77, 1976.

[3] T. Mitsa and K.J. Parker, "Digital halftoning technique using a blue-noise mask," *J. Opt. Soc. Am. A*, vol. 9, no. 11, pp. 1920–1929, Nov. 1992.

[4] R. Ulichney, "The void-and-cluster method for dither array generation," in *Proc. SPIE, vol. 1913, Human Vision, Visual Proc., and Digital Display IV*, San Jose, CA, Feb. 1993.

[5] T.N. Pappas and D.L. Neuhoff, "Model-based halftoning," in *Proc. SPIE, vol. 1453, Human Vision, Visual Proc., and Digital Display II*, San Jose, CA, Feb. 1991.

[6] T.N. Pappas and D.L. Neuhoff, "Printer models and error diffusion," to appear in *IEEE Trans. Image Proc.*, vol. 4, Jan. 1995.

[7] T.N. Pappas and D.L. Neuhoff, "Least-squares model-based halftoning," in *Proc. SPIE, vol. 1666, Human Vision, Visual Proc., and Digital Display III*, San Jose, CA, Feb. 1992.

[8] D.L. Neuhoff, T.N. Pappas, and N. Seshadri, "One-dimensional least-squares model-based halftoning," in *Proc. ICASSP-92, Vol. 3*, San Francisco, CA, Mar. 1992.

[9] J.L. Mannos and D.J. Sakrison, "The effects of a visual fidelity criterion on the encoding of images," *IEEE Trans. Inf. Theory*, vol. IT-20, pp. 525–536, July 1974.

[10] S. Daly, "Subroutine for the generation of a two dimensional human visual contrast sensitivity function," Technical Report 233203Y, Eastman Kodak, 1987.

[11] J. Sullivan, L. Ray, and R. Miller, "Design of minimum visual modulation halftone patterns," *IEEE Trans. Sys., Man, Cyb.*, vol. 21, no. 1, pp. 33–38, Jan./Feb. 1991.

[12] C.-K. Dong, "Measurement of printer parameters for model-based halftoning," B.S./M.S. thesis, MIT, May 1992.

[13] T.N. Pappas, C.-K. Dong, and D.L. Neuhoff, "Measurement of printer parameters for model-based halftoning," *Journal of Electronic Imaging*, vol. 2, pp. 193–204, July 1993.

[14] P. Stucki, "MECCA – a multiple-error correcting computation algorithm for bilevel image hardcopy reproduction," Research Report RZ1060, IBM Research Laboratory, Zurich, Switzerland, 1981.

[15] J.F. Jarvis, C.N. Judice, and W.H. Ninke, "A survey of techniques for the display of continuous-tone pictures on bilevel displays," *Comp. Graphics and Image Proc.*, vol. 5, pp. 13–40, 1976.

[16] S. Daly, "The visible differences predictor: an algorithm for the assessment of image fidelity," in *Proc. SPIE, vol. 1666, Human Vision, Visual Proc., and Digital Display III*, San Jose, CA, Feb. 1992.

[17] P.G. Roetling and T.M. Holladay, "Tone reproduction and screen design for pictorial electrographic printing," *Journal of Applied Phot. Eng.*, vol. 15, no. 4, pp. 179–182, 1979.

[18] J. B. Mulligan and A. J. Ahumada, Jr., "Principled halftoning based on models of human vision," in *Proc. SPIE, vol. 1666, Human Vision, Visual Proc., and Digital Display III*, (San Jose, CA), Feb. 1992.