Least-Squares Model-Based Halftoning

Thrasyvoulos N. Pappas, Senior Member, IEEE, and David L. Neuhoff, Fellow, IEEE

Abstract—A least-squares model-based (LSMB) approach to digital halftoning is proposed. It exploits both a printer model and a model for visual perception. It attempts to produce an optimal halftoned reproduction, by minimizing the squared error between the response of the cascade of the printer and visual models to the binary image and the response of the visual model to the original gray-scale image. It has been shown that the one-dimensional (1-D) least-squares problem, in which each row or column of the image is halftoned independently, can be implemented using the Viterbi algorithm to obtain the globally optimal solution. Unfortunately, the Viterbi algorithm cannot be used in two dimensions. In this paper, the two-dimensional (2-D) least-squares solution is obtained by iterative techniques, which are only guaranteed to produce a local optimum. Experiments show that LSMB halftoning produces better textures and higher spatial and gray-scale resolution than conventional techniques. We also show that the least-squares approach eliminates most of the problems associated with error diffusion. We investigate the performance of the LSMB algorithms over a range of viewing distances, or equivalently, printer resolutions. We also show that the LSMB approach gives us precise control of image sharpness.

Index Terms—Facsimile, halftoning, least-squares, model-based, printer, visual perception.

I. INTRODUCTION

DIGITAL halftoning is the process of generating a pattern of binary pixels that create the illusion of a continuous-tone image. The term spatial dithering is also used to refer to this process. Digital halftoning is necessary for display of gray-scale images in media in which the direct rendition of gray tones is impossible. The most common example is printing of gray-scale images on paper. In this paper, we propose a least-squares model-based (LSMB) approach to digital halftoning that exploits both a printer model and a model for visual perception to produce high quality images using standard laser printers (typically 300 dpi).

Model-based halftoning can be especially useful in transmission of high quality documents using high fidelity gray-scale image encoders [1], [2]. In such cases, halftoning is performed at the receiver just before printing. Apart from coding efficiency, this approach permits the halftoner to be performed at the receiver just before printing. Apart from scale image encoders [1], [2]. In such cases, halftoning is impossible. The most common example is printing of the black portion of a printed binary pattern is proportional to the intended fraction of black dots (pixels) in the pattern. This means that the area occupied by each black dot is roughly the same as the area occupied by each white dot. Thus, the “desired” shape for the black dots produced by a printer would be $T \times T$ squares, where $T$ is the dot spacing. However, most printers produce approximately circular dots [4]. Their radius must be at least $T/\sqrt{2}$ so that they are capable of blackening a page entirely. Thus, black dots cover portions of adjacent spaces, causing the perceived gray level to be darker than intended. We refer to this fact as “dot overlap.” Moreover, most printers produce black dots that are larger than the minimal size, which further distorts the perceived gray level. The most commonly used digital halftoning techniques, such as clustered ordered dither, protect against dot overlap by clustering black dots so the percentage effect on perceived gray level is reduced. Unfortunately, such clustering constrains the spatial resolution of the perceived images and increases the low-frequency artifacts. The proposed method exploits printer distortions in order to increase both gray-scale and spatial resolution. A key element in such a method is an accurate printer model.

Several authors have previously proposed model-based halftoning approaches. Allebach [5] proposed halftoning so as to minimize a visual model based distortion measure, but did not give specific methods for doing so. He also suggested the use of a display model but did not give a specific one. Anastassiou [6] proposed halftoning so as to minimize a frequency weighted squared error between the binary and original images. The latter is equivalent to our formulation except that we incorporate a printer model and permit nonlinearity in the eye model. Anastassiou and Kollias [6], [7] also described a neural network technique that approximately minimized the frequency weighted squared error. Roetting and Holladay [8] proposed a dot-overlap printer model, like the one used in this paper and [9], [10], but used it only to modify ordered dither so that it results in a linear gray scale. Allebach [11] also proposed a modification of ordered dither that takes into account dot overlap to improve the gray scale of the printed images. However, ordered dither
is a highly constrained method that cannot achieve the high spatial resolution, increased gray-scale resolution, and visually pleasant textures of the proposed model-based approach, nor of the modified error diffusion mentioned in the next paragraph. Another technique that takes into account dot overlap was proposed by Pryor et al. [12]. Its performance is also inferior to the proposed method.

In [9] and [10] it was shown that printer models, especially the dot-overlap model, can be used to modify error diffusion to account for printer distortions. The resulting modified error diffusion (MED) algorithm provides high quality reproductions with reasonable complexity. However, error diffusion does not make use of an explicit eye model, and this results in some well known artifacts and asymmetries. The first to propose the use of a dot-overlap model to account for printer distortions in error diffusion was Stucki [13], [14]. In [10], it was found that Stucki’s algorithm is more efficient computationally, while the MED algorithm has better performance. Recently, Rosenberg [15] used our dot-overlap printer model to derive a gray-scale transformation that can be applied to the image before halftoning to compensate for printer distortions. This approach works only when the printer introduces a monotonic nonlinearity, and cannot match gray levels as precisely as the model-based approaches examined here and in [10].

Finally, in [16] and [17], it was shown that the parameters of sliding-window printer models can be identified from measurements of the reflectance of various test patterns. These were then used in the MED and LSMB approaches.

The LSMB halftoning approach attempts to produce 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 gray-scale image. In [18] and [19], we showed that the one-dimensional (1-D) least-squares problem, in which each row or column of the image is halftoned independently, can be implemented with the Viterbi algorithm. The Viterbi algorithm provides an efficient way to search the solution space and leads to a global optimum in a finite number of steps. Unfortunately, the Viterbi algorithm cannot be used in two dimensions. In this paper, the two-dimensional (2-D) least squares solution is obtained by iterative techniques, which are only guaranteed to produce a local optimum. The visual quality of the resulting halftone images depends on the starting point and the optimization strategy. Experiments show that LSMB halftoning can produce images with high gray-scale and spatial resolution and pleasing textures. We show that the least-squares approach eliminates most of the problems associated with error diffusion. However, despite these problems, the performance of the MED algorithm is remarkably good. As we will see, MED provides the best starting points for the LSMB algorithms which mostly preserve its texture. In fact, when one takes into consideration both computational complexity and image quality, the MED algorithm provides the best performance. The main value of the LSMB approach is as a tool for learning how well halftoning can do under various circumstances. It gives us the ability to study how the optimal halftoning patterns vary with eye and printer models. For example, we investigate the performance of the LSMB algorithms over a range of viewing distances (or equivalently, printer resolutions). As the viewing distance increases, the cutoff frequency of the eye model decreases, which means it is less sensitive to the “coarseness” of the halftone texture. Thus, more coarsely textured halftone patterns can be considered. This raises the question of whether more coarsely textured patterns actually are needed for optimality. If so, then the resulting halftones will depend on the viewing distance, while otherwise, they will be fairly robust. We also show that the LSMB approach gives us precise control of image sharpness.

The error criterion of the LSMB approach provides a measure of halftone image quality that takes into account the characteristics of both the display device (printer) and the human visual system and can be used to evaluate the performance of any halftoning technique. Our results indicate that the LSMB error metric agrees remarkably well with visual evaluations of image quality but we also point out its limitations. Overall, it provides a powerful tool for algorithm development and evaluation.

In addition to [6], the LSMB approach has been recently pursued by several authors. The LSMB approach, including both eye and printer models, was presented by the present authors in [22] and [23]. In similar efforts, Mulligan and Ahumada [24] assumed perfect printing, and Analoui and Allebach [25] considered a different display model. Carrara et al. [26] considered variations of the scheme presented in [25] and also adopted our dot-overlap printer model, while Flohr et al. [27] investigated computational issues. Zakhor et al. [28], [29] also assumed perfect printing, but the performance of the approach they proposed is inferior to the above techniques because it breaks the image into small blocks and halftones them independently. Another approach that breaks the image into blocks and uses an eye model and perfect printing was presented by Vander Kam et al. [30]. Finally, a related technique using Markov random fields was presented by Geist et al. [31].

The paper is organized as follows. Section II discusses eye models. The printer models are presented in Section III. Section IV presents the least-squares model-based halftoning approach. The experimental results are presented in Section V and the conclusions are summarized in Section VI.

II. MODELS OF VISUAL PERCEPTION

As mentioned in the introduction, halftoning works because the eye acts as a spatial lowpass filter. In this section we review the spatial frequency characteristics of the eye and introduce a more detailed model that will be exploited by the least-squares approach of Section IV. A similar discussion can be found in [18] and [19]. Parts of that discussion are repeated here for completeness.

1 Wong [20] used the Viterbi algorithm to solve a 2-D halftoning problem but the solution is not optimal.

2 Strictly speaking, changing the viewing distance is equivalent to changing the printer resolution only if the characteristics of the printed dots are not affected when the printer resolution is changed.

3 This question was also addressed in [21].
Numerous researchers have estimated the spatial frequency sensitivity of the eye, often called the modulation transfer function (MTF). Typical of such is the estimate that Mannos and Sakrison [3] found to be good for predicting the subjective quality of coded images

\[ H(f) = 2.6(0.0192 + 0.114 f) \exp\left\{-(0.114 f)^{1.1}\right\} \]  

where \( f \) is in cycles/degree. This MTF is plotted in Fig. 1 in dotted line. As indicated by (1), the eye is most sensitive to frequencies around 8 cycles/degree (others have variously estimated the peak sensitivity to lie between 3 and 10 cycles/degree [32, p. 55]). The decrease in sensitivity at higher frequencies is generally ascribed to the optical characteristics of the eye, (e.g., pupil size) [33]. The decrease in sensitivity at low frequencies accounts for the “illusion of simultaneous contrast” (a region with a certain gray level appears darker when surrounded by a lighter gray level than when surrounded by a darker) and for the Mach-band effect (when two regions with different gray levels meet at an edge, the eye perceives a light band on the light side of the edge and a dark band on the dark side of the edge).

The eye is more sensitive to horizontal or vertical sinusoidal patterns than to diagonal ones [34]. Specifically, it is least sensitive to 45° sinusoids, with the difference being about 0.6 dB at 10 cycles/degree and about 3 dB at 30 cycles/degree. This difference is not considered to be large, but it is used to good effect by the “Classical” halftoning technique [35], which produces patterns with a diagonal structure.

Besides the MTF, many other characteristics of the human visual system have been studied. A recent, comprehensive reference is [36]. Based on such characteristics, a variety of visual models have been proposed for use in various image processing tasks. The simplest, arguably, include just a filter, for example the filter of (1). The next simplest include also a memoryless nonlinearity, followed by a filter. Such nonlinearities account for Weber’s law, which says that the smallest noticeable change in intensity is proportional to intensity. Most commonly it is represented as a logarithm or power law. More complex models include, for example, a filter before the nonlinearity or a bank of filters.

In many cases, practical considerations dictate a finite impulse response (FIR) filter. Indeed, for the least-squares halftoning of Section IV, we require a discrete-space model of the form

\[ z_{i,j} = M(x_{k,l}, k = i-m, \ldots, i+m, l = j-m, \ldots, j+m) \]  

where the \( x_{k,l} \)'s are samples of the image, the \( z_{i,j} \)'s are the model outputs (upon which cognition is based), and \( M \) is some sliding-window function (\( m \) is a nonnegative integer). Such a model can easily incorporate a memoryless nonlinearity and an FIR filter. We experimented with models of the form

\[ z_{i,j} = n(x_{i,j}) * h_{i,j} \]  

where \( n(\cdot) \) is a memoryless nonlinearity, \( h_{i,j} \) is the impulse response of an FIR filter, and \( * \) denotes convolution. One way to obtain a 2-D filter \( h_{i,j} \) as is as a separable combination of 1-D FIR filters. Fig. 1 shows (in solid lines) the frequency and impulse responses of a 1-D FIR approximation to the MTF of (1) [18], [19]. Note that the impulse response is specified in degrees of visual angle. The spacing of the dots is given by

\[ \tau = 2 \tan^{-1}\left(\frac{1}{2 R d}\right) \approx \frac{1}{R d} \text{ radians} = \frac{180}{R d} \pi \text{ degrees} \]  

where \( R \) is the printer resolution in dpi and \( d \) is the viewing distance. The sampling frequency is \( f = 1/\tau \). The filter of Fig. 1 was designed for a printer resolution of 300 dpi and a viewing distance of 30 in.

Note that the frequency response of the filter of Fig. 1 does not decrease at low frequencies, that is, it does not model the Mach-band effect. This would require a considerably longer FIR filter which would increase the complexity of the algorithms we consider. Moreover, as we will see in Section V, higher order filters, that do capture the Mach-band effect, have very little effect in the resulting images. In any case, it is the lowpass characteristic of the eye that is necessary for halftoning to work. The fact that the eye is not sensitive to very low frequencies does not have to be used. Note also that the halftoning algorithm should not compensate for the Mach-band effect, i.e., the eye should perceive the same Mach-band effect from the halftoned images as it does from the original continuous-tone image.

Similar FIR filters can be designed for different printer resolutions and viewing distances. Figs. 2 and 3 show the impulse and frequency responses of a sequence of such filters designed to match the (flattened) Mannos and Sakrison MTF for samples taken at 300 dpi and viewing distances of 12, 24, and 36 in. Since the sampling is coarser at shorter distances, the order of the FIR filter is lower, and more importantly, the approximation is cruder. We will refer to these filters as the 1-D best-fit models.
We have found that the impulse responses of the 1-D eye filters we designed above are well approximated by a Gaussian shape with appropriate standard deviation. In fact, as was pointed out in [37], at 300 dpi and a viewing distance of 30 in, the impulse response of the 1-D eye filter and a Gaussian filter with $\sigma = 1.5$, $\tau = 0.0095^\circ$ are virtually identical. This is convenient because one can simply scale the standard deviation to obtain Gaussian filters for different viewing distances. Equivalently, one can sample (at multiples of $\tau$) a Gaussian with standard deviation fixed at $\sigma = 0.0095^\circ$. Note that the 2-D Gaussian filters are circularly symmetric.

In addition to the 1-D best-fit models, Figs. 2 and 3 show the impulse and frequency responses of such a sequence of (truncated) Gaussian filters that match the (flattened) Mannos and Sakrison MTF for samples taken at 300 dpi and viewing distances of 12, 24, and 36 in. At 12 in, the Gaussian filter does not match the Mannos and Sakrison MTF as well. This is due to aliasing; the sampling is too coarse at this viewing distance. A better approximation can be obtained by using the separable combination of the 1-D best-fit models. However, our experiments with the techniques of Section IV indicate that the two filters result in very similar halftone images.

Circularly symmetric Gaussian filters were also used in [24]. (They were truncated to $5 \times 5$ with standard deviations ranging from 0.2–1.6 pixels.) Finally, Gaussian filters were used in the void and cluster method for blue-noise screen design [38]. The standard deviation of the filter was chosen to be 1.5 pixels, which interestingly, corresponds to a viewing distance of 30 in at 300 dpi.

The frequency response of the eye filters we consider here is also very close to the MTF proposed by Daly [39], [40]. This can be seen in Fig. 1, where Daly’s MTF is plotted as a dashed line. Daly’s model was used in [25]–[27]. Daly’s MTF also models the variations in the visual MTF as a function of viewing angle (pattern orientation). The FIR filter of (3) could be designed to account for such variations. However, unlike [41], our experiments with the techniques of Section IV did not show a significant effect on the resulting halftone patterns when a filter with angular dependence is used. Thus, the

Note that Daly’s model (like Mannos and Sakrison’s) is specified in the frequency domain; the FIR filter specification in the spatial domain was obtained by the inverse 2-D fast Fourier transform (FFT) of the sampled frequency domain function.
circularly symmetric Gaussian filters seem to be a reasonable simplification. In the remainder of this paper we will use the Gaussian approximation because it is easier to adapt to different viewing conditions. We will assume that the printer resolution is fixed and will consider a sequence of eye filters corresponding to different viewing distances. Our approach is also appropriate for a fixed viewing distance and various printer resolutions.

Finally in [18] and [19], we considered a nonlinearity in the form of a power law, and found that it offers no significant improvements. As we will see in Section V, the same is true in this case.

III. PRINTER MODELS

In this section, we review the printer models that were described in [9], [10]. The printer models are independent of the characteristics of the human visual system. Our work is oriented to high resolution laser printers. We have used “write-black” laser printers with 300 dpi resolution (such as the HP LaserJet II) as our test vehicles. However, the discussion and methods are intended to apply directly, or be adaptable, to a wide variety of printers.

To a first approximation, such printers are capable of producing black spots (usually called dots) on a piece of paper, at any and all sites of a Cartesian grid with horizontal and vertical spacing of $T$ in. The reciprocal of $T$ is generally referred to as the printer resolution in dpi. The printer is controlled by an $N_W \times N_H$ binary array $b_{i,j}$, where $b_{i,j} = 1$ indicates that a black dot is to be placed at site $(i, j)$ which is located $iT$ inches from the left and $jT$ inches from the top of the image, and $b_{i,j} = 0$ indicates that the site is to remain white.

As discussed in the introduction, printers produce black dots that are more round than square. Fig. 4 illustrates the most elementary distortion introduced by most printers: their dots are larger than the minimal covering size, as if “ink spreading” occurred. A variety of other distortions are present in actual printers, caused by the heat finishing, reflections of light within the paper, and other phenomena. As a result, the gray level produced by the printer in the vicinity of site $(i, j)$ depends in some complicated way on $b_{i,j}$ and neighboring bits. The purpose of a printer model is to predict this dependence. In [9] and [10], we proposed a discrete space printer model, the input of which is the binary array $b_{i,j}$ and the output an array $p_{i,j}$ of gray-scale values, where $p_{i,j}$ represents the darkness of site $(i, j)$. Specifically, the model has the form

$$p_{i,j} = \mathcal{P}(W_{i,j}) \quad 1 \leq i \leq N_W, \quad 1 \leq j \leq N_H$$

where $W_{i,j}$ denotes the bits in some neighborhood of $b_{i,j}$ and $\mathcal{P}$ is some function thereof. This discrete space model offers considerable computational advantages. The underlying assumptions of the above discrete space model can be made more concrete when one considers a continuous space version of the model, as was done in [10].

For the least-squares approach, it is essential that $p_{i,j}$ be entirely determined by the bits in a finite window around $b_{i,j}$, e.g., a $3 \times 3$ window. In this case, the possible values of $\mathcal{P}$ can be listed in a table, e.g. with $2^9$ elements. The individual elements of this table might be derived from a detailed physical understanding of the various phenomena affecting gray level or from measurements of the gray level due to various dot patterns. An example of a printer model based on the first approach is given below. Alternative printer models based on the second approach are explored in [16] and [17].

A simple printer model that accounts for the “dot-overlap” distortion illustrated in Fig. 4 is the circular dot-overlap model [9], [10]

$$p_{i,j} = \mathcal{P}(W_{i,j}) = \begin{cases} 1, & \text{if } b_{i,j} = 1 \\ f_1\alpha + f_2\beta + f_3\gamma, & \text{if } b_{i,j} = 0 \end{cases}$$

where the window $W_{i,j}$ consists of $b_{i,j}$ and its eight neighbors, $f_1$ is the number of horizontally and vertically neighboring dots that are black, $f_2$ is the number of diagonally neighboring dots that are black and not adjacent to any horizontally or vertically neighboring black dot, and $f_3$ is the number of pairs of neighboring black dots in which one is a horizontal neighbor and the other is a vertical neighbor, and where $\alpha$, $\beta$, and $\gamma$ are the ratios of the areas of the shaded regions shown in Fig. 5 to $T^2$.

The parameters $\alpha$, $\beta$, and $\gamma$ can be expressed in terms of the ratio $\rho$ of the actual dot radius to the ideal dot radius $T/\sqrt{2}$ [9], [10]. Note that the model is not linear in the input bits, which is due to the fact that the paper saturates at black intensity. For a printer with the ideal dot size, $\rho = 1$, the minimum value, and $\alpha = 0.143$, $\beta = 0$, and $\gamma = 0$. For the HP printer, our experience indicates that $\rho \approx 1.25$, which results in $\alpha = 0.33$, $\beta = 0.020$, and $\gamma = 0.008$. Fig. 6 illustrates how the dot pattern in Fig. 4 is modeled with these values.
In [9] and [10], we showed that (assuming that the above printer model is valid) dot overlap can be exploited to obtain more gray levels than could be obtained without it. Moreover, as we will show in the next sections, this can be done without sacrificing spatial resolution.

IV. LEAST-SQUARES MODEL-BASED HALFTONING

As mentioned earlier, the least-squares model-based (LSMB) halftoning approach attempts to minimize 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. The least-squares approach is noncausal. That is, the decisions at any point in the image depend on “past” as well as “future” decisions. In standard error diffusion, the decisions at any point in the image depend on the “past” only. It is this noncausality of the least-squares approach that gives it the freedom to make sharp transitions and track edges better than previously described approaches.

The 2-D least-squares halftoning problem can be formulated as follows. Suppose we are given a gray-scale image \( x_{i,j} \), with \( i = 1, \ldots, N_W \) and \( j = 1, \ldots, N_H \), where \( N_W \) is the width of the printable lines in dots and \( N_H \) is the number of printable lines. We assume that the image has been sampled so there is one pixel per dot to be generated.\(^6\) The gray level \( x_{i,j} \) of each pixel varies from 0 (white) to 1 (black). We are also given a printer model with the sliding-window form of (5), and an eye model of the form specified in Section II with a memoryless nonlinearity \( n(\cdot) \) followed by an FIR filter with impulse response \( h_{i,j} \). In the LSMB approach we seek the binary approximation \( \tilde{b}_{i,j} \) that minimizes the squared error

\[
E = \sum_{i=1}^{N_W} \sum_{j=1}^{N_H} (z_{i,j} - w_{i,j})^2
\]

where, as illustrated in Fig. 7

\[
\begin{align*}
z_{i,j} &= y_{i,j} * h'_{i,j} = n(x_{i,j}) * h'_{i,j} \quad (8) \\
w_{i,j} &= v_{i,j} * h_{i,j} = n(p_{i,j}) * h_{i,j} \quad (9) \\
p_{i,j} &= P(W_{i,j}) \quad (10)
\end{align*}
\]

\(^6\)When the number of samples of a given image is different from the number of dots to be generated, interpolation is necessary. *Bilinear* and *spline* interpolation can produce an image of any size. However, for sampling rate conversions by a rational factor, the best results are obtained by an *expander* followed by an *equiripple* lowpass FIR filter [42 pp. 105–109].

These boundary conditions assume that no ink (toner) is placed outside the image borders.

Note that we have allowed different impulse responses \( h_{i,j} \) for the eye filters corresponding to the continuous-tone and halftone images. We found that when we do not filter the gray-scale image, the resulting halftone images are sharper.\(^7\) This was suggested by Anastassiou in [6]. In fact, as we will see in the next section, one can consider a whole range of filters \( h_{i,j} \) that result in different amounts of sharpening.

In principle, the optimal solution can be obtained by an exhaustive search over all possible binary patterns for the entire image. This approach is not computationally feasible, however. The number of possible patterns for an \( N_H \times N_W \) image is \( 2^{N_H N_W} \) (e.g., \( 1.16 \times 10^{17} \) just for a 16 \( \times \) 16 image). Thus, the 2-D least-squares solution is obtained by iterative optimization techniques. Such techniques find a solution that is only a local optimum. They assume that an initial estimate of the binary image \( \tilde{b}_{i,j} \) is given. This could be a trivial image, e.g., a constant or random image, or the output of any halftoning algorithm including the MED [9], [10] and the 1-D LSMB algorithm [18]. The visual quality of the resulting halftone images depends on the starting point and the optimization strategy.

A simple iterative scheme, initially presented in [22] and [23], updates the values of the binary image one pixel at a time, in the following way. Given an initial estimate \( \tilde{b}_{i,j} \), for every image site \( (i,j) \) (in some fixed or random order, usually

\[^7\]In effect, we are prefiltering the continuous-tone image with the inverse of the eye filter, i.e., a highpass filter.
(a) (b) (c)

Fig. 9. Details of 300 dpi printouts (magnified by 3). (a) Classical screen. (b) MP-MED with FS filter. (c) MP-MED with JJN filter.

a raster scan), we find the binary value $b_{i,j}$ that minimizes the squared error

$$E_{i,j} = \sum_{(k,l) \in B_{i,j}} (z_{k,l} - w_{k,l})^2$$

(13)

where $B_{i,j}$ is a neighborhood of the point $(i, j)$. If the eye filter is FIR and the neighborhood $B_{i,j}$ is chosen large enough, then choosing $b_{i,j}$ to minimize $E_{i,j}$ is equivalent to choosing $b_{i,j}$ to minimize the overall error $E$. In other words, when the eye filter is FIR, the binary value of each pixel affects only the model outputs $w_{k,l}$ in its neighborhood, and thus, the error need only be computed locally. Here we should mention that the value of $E_{i,j}$ should not be computed from scratch each time; the value of $E_{i,j}$ never changes and only a few terms in the convolution sum that determines $w_{k,l}$ must be updated (those that involve $b_{k,l}$ and its neighbors).

An iteration is complete when the minimization is performed once at each image site. The number of iterations depends on the starting point and the effective filter width. Usually, the wider the eye filter, the more the required iterations. We will refer to the above optimization procedure as the toggle-only scheme, because in comparison to schemes described later, the only permitted change at each image site is toggling the value of $b_{i,j}$.

As we will see in the next section, when the toggle-only scheme is used, the dependence on the starting point is strong.

A similar scheme was proposed by Mulligan and Ahumada in [24]. In [6], Anastassiou proposed a similar optimization procedure using neural nets. Note that, even though the error is computed locally, there is a strong dependence on the neighboring pixels that can propagate over large areas. Hence breaking up the image into blocks and optimizing them independently (as in [28] and [30]) could result in significant deviations from optimality.

Selecting a trivial neighborhood $B_{i,j}$ that consists of just the point $(i, j)$ reduces the amount of computation but is not guaranteed to reduce the overall error, especially for narrow eye filters. For wider eye filters, i.e., at 300 dpi and viewing distance of 30 in, the results are almost as good as those obtained with the large neighborhood [23]. In the remainder of this paper, however, we will assume that the neighborhood $B_{i,j}$ is large enough to guarantee that minimizing $E_{i,j}$ is equivalent to minimizing $E$, as discussed above.

In order to obtain lower errors and better results, we considered a variation of the above iterative technique, whereby, each minimization is performed over a small set of adjacent pixels. That is, given an initial estimate $b_{i,j}$, for each image site $(i, j)$, we find the binary values $b_{i,j}$ in a neighborhood $B_{i,j}$ that minimize the squared error $E_{i,j}$, as expressed in (13). However, the amount of computation increases exponentially with the number of pixels in $B_{i,j}$.

In order to control the amount of computation, Carrara et al. [26], suggested testing only a small subset of all possible binary patterns in the neighborhood $B_{i,j}$. Specifically, they chose $B_{i,j}$ to be a $3 \times 3$ window surrounding $(i, j)$, and tested only ten of the $2^9$ possible binary patterns: the current pattern as is, the pattern formed by toggling the value of $b_{i,j}$, and the eight patterns formed by swapping the value of $b_{i,j}$ with the value of one of its eight neighbors in $B_{i,j}$.

guaranteed to reduce the overall error, especially for narrow eye filters. For wider eye filters, i.e., at 300 dpi and viewing distance of 30 in, the results are almost as good as those obtained with the large neighborhood [23]. In the remainder of this paper, however, we will assume that the neighborhood $B_{i,j}$ is large enough to guarantee that minimizing $E_{i,j}$ is equivalent to minimizing $E$, as discussed above.

In order to obtain lower errors and better results, we considered a variation of the above iterative technique, whereby, each minimization is performed over a small set of adjacent pixels. That is, given an initial estimate $b_{i,j}$, for each image site $(i, j)$, we find the binary values $b_{i,j}$ in a neighborhood $B_{i,j}$ that minimize the squared error $E_{i,j}$, as expressed in (13). However, the amount of computation increases exponentially with the number of pixels in $B_{i,j}$.

In order to control the amount of computation, Carrara et al. [26], suggested testing only a small subset of all possible binary patterns in the neighborhood $B_{i,j}$. Specifically, they chose $B_{i,j}$ to be a $3 \times 3$ window surrounding $(i, j)$, and tested only ten of the $2^9$ possible binary patterns: the current pattern as is, the pattern formed by toggling the value of $b_{i,j}$, and the eight patterns formed by swapping the value of $b_{i,j}$ with the value of one of its eight neighbors in $B_{i,j}$.

8 Let $S_h$ be the support of the eye filter $h$ and $S_p$ be the support of the printer model. Then $B_{i,j}$ must be at least as large as $S_h \oplus S_p$, where $\oplus$ denotes dilation and $S_h$ is the transpose of the set $S_h$.

9 Again, $B_{i,j}$ must be large enough to guarantee that minimizing $E_{i,j}$ is equivalent to minimizing $E$. 

<table>
<thead>
<tr>
<th>Image</th>
<th>Bank</th>
<th>Lena</th>
<th>6386</th>
<th>Ramp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classical screen</td>
<td>1476.4</td>
<td>1699.8</td>
<td>1294.0</td>
<td>1272.7</td>
</tr>
<tr>
<td>Blue-noise screen</td>
<td>68.8</td>
<td>60.7</td>
<td>129.9</td>
<td>54.4</td>
</tr>
<tr>
<td>MED (FS filter)</td>
<td>512.2</td>
<td>687.8</td>
<td>292.6</td>
<td>481.0</td>
</tr>
<tr>
<td>MED (JN filter)</td>
<td>149.2</td>
<td>155.0</td>
<td>170.4</td>
<td>123.7</td>
</tr>
<tr>
<td>MP-MED (FS filter)</td>
<td>64.2</td>
<td>70.6</td>
<td>66.8</td>
<td>71.1</td>
</tr>
<tr>
<td>MP-MED (JN filter)</td>
<td>54.2</td>
<td>38.4</td>
<td>107.3</td>
<td>34.7</td>
</tr>
<tr>
<td>LSMB with random start</td>
<td>36.0</td>
<td>40.3</td>
<td>62.7</td>
<td>41.8</td>
</tr>
<tr>
<td>LSMB with MP MED (JN filter) start</td>
<td>21.1</td>
<td>22.0</td>
<td>41.2</td>
<td>24.0</td>
</tr>
</tbody>
</table>

TABLE I

PER PIXEL SQUARED ERROR FOR DIFFERENT HALFTONING TECHNIQUES FOR 300 DPI PRINTER RESOLUTION AND VIEWING DISTANCE OF 30 IN
This scheme is called the toggle/swap scheme. Analoui and Allebach [25] considered swaps only. Flohr et al. [27] showed that the toggle/swap scheme produces higher quality images than the toggle-only scheme. Our results presented in the next section are consistent with this. They also indicate weaker dependences on the starting point.

As we pointed out above, the images produced using the above methods are only local minima of the least-squares problem. There could in fact be several local minima for a given image, depending on the starting point, as we show in the next section. More sophisticated (and computationally intensive) schemes use simulated annealing [24], [37] but have not yet shown any significant improvements in image quality.

Finally, in an attempt to minimize the dependence on the starting point, we tried the following “progressive” scheme: we used the LSMB estimate at a given viewing distance as a starting point for the next higher viewing distance, starting at 12 in with 6-in increments. We used the multipass MED \(^{10}\) with a Floyd–Steinberg (FS) filter as the original starting point, and the simple iterative scheme with toggle/swap at each viewing distance. Even though this scheme requires a considerable amount of computation, we use it in an attempt to find alternative minima and to investigate the sensitivity of the results to the optimization strategy.

The error criterion of the LSMB algorithm can be used to evaluate the performance of any halftoning technique (see the next section and [37]). It provides a measure of halftone image quality that takes into account the characteristics of both the display device (printer) and the human visual system. The LSMB image quality metric can be computed over the whole image or over a small segment of the image to provide a localized measure of performance. Thus, we can define

\[
E_A = \frac{1}{N} \sum_{(i,j) \in A} (z_{i,j} - w_{i,j})^2
\]

where \(z_{i,j}\) and \(w_{i,j}\) are given by (9) and (10) and \(A\) is a subset of the image containing \(N\) pixels. Usually \(A\) does not include the boundary pixels to avoid biases due to edge artifacts. In

\(^{10}\)The multipass MED will be defined in the next section.
the next section, this error criterion is used to compare other methods to LSMB halftoning.

V. EXPERIMENTAL RESULTS

We tested the LSMB approach on several images and compared it to other schemes. Fig. 8 shows two of those images. The resolution of the Lena image is $512 \times 512$ pixels and the resolution of the double gray-scale “Ramp” is $788 \times 80$ pixels. The gray-scale resolution of the original images is 8 bits/pixel. The images in Figs. 9–14 are magnified details (by a factor of three) of the halftone images as they would be printed on a 300 dpi laser printer. They were printed using
simulated dot overlap with $\rho = 1.25$ at one third of the printer resolution.

We evaluate the performance of the LSMB technique in terms of 1) spatial resolution, (i.e., sharpness), 2) texture, (i.e., visibility of halftone patterns), 3) gray-scale resolution, (i.e., number of perceived gray levels), and 4) gray-scale distortion of the halftoned images. As we saw in [19], some of these attributes can be interdependent.

Fig. 9(a) shows a magnified detail of one of the test images halftoned using a clustered ordered dither scheme (“classical”), and Fig. 9(b) and (c) show the image halftoned with the modified (to compensate for dot-overlap) error diffusion algorithm (MED) [9], [10], using the FS and Jarvis–Judice–Ninke (JJN) filters, respectively. The figure shows the results of the multipass version of the modified error diffusion algorithm (MP-MED) which was introduced in [16] and [17]. As was shown in [16] and [17], the MED algorithm produces images that are slightly (almost imperceptibly) darker than the original continuous-tone image, while the multipass MED algorithm produces the correct gray level. The values of the LSMB image quality metric for these techniques are given in Table I. (Note, that the table includes data from two images that are not shown in Fig. 8.) Observe that “classical” screening, which is the worst visually, results in the highest errors. The bias of the MED algorithm is reflected in significantly higher error values than the corresponding values for the multipass MED algorithm.

The performance of the LSMB method depends strongly on the choices of printer and eye models. As we will see shortly, it also depends on the starting point and the optimization strategy. We evaluated the results of the method with the following “baseline” choices: circular dot-overlap printer model with $\rho = 1.25$, a Gaussian eye filter for a viewing distance of 30 in at 300 dpi, and no eye nonlinearity. We used the toggle-only scheme, described in the previous section, that updates one pixel at a time. We also assumed that the two filters $h_{i,j}$ and $H_{i,j}$ are equal. Fig. 10 shows the results of the LSMB approach with two different starting points. The starting point for Fig. 10(a) was a random binary image, and for Fig. 10(b) the MP-MED algorithm with the JJN filter. Observe that there is a strong dependence on the starting point. The scheme tends to preserve the original texture for the MP-MED starting point, while the random start results in patterns that look considerably grainier. Table I shows that the LSMB algorithm with MP-MED starting point results in the lowest error; the error for the random start is significantly higher. Thus, the values of the LSMB image quality metric agree with the visual evaluation. Using a blue-noise screen as a starting point also results in higher errors and grainy images [37]. In general, our experiments indicate that this scheme either preserves the original texture or converges to grainy patterns like those of Fig. 10(a).

As was shown in [23], the LSMB algorithm results in grainy images even when the (one-pass) MED output is used as a starting point. The reason for this is that, as we mentioned above, the MED algorithm produces images that are slightly

---

11 Our experiments indicate that, for the given set of images, the eye nonlinearity offers no significant improvements.
original motivation for the introduction of the multipass MED algorithm by Dong in [16].

As indicated in the previous section, the number of iterations depends on the starting point. The MP-MED starting point requires the fewest iterations. This can be explained by the fact that the toggle-only algorithm tends to preserve the original MP-MED texture.

We now consider variations in the eye filters to account for different viewing distances. A similar experiment was carried out by Mulligan and Ahumada in [24]. We choose $h_{x,y} = h_{x,y}$ and consider eye models that correspond to viewing distances of 1, 1.5, 2, 2.5, 3, 4, 5, and 6 ft at 300 dpi. Table II lists the per pixel squared errors for the whole range of viewing distances and a number of different starting points. The lowest error for each column is shown in boldface characters. The table demonstrates that the toggle-only scheme is quite sensitive to the starting point, i.e., there is considerable variation in the per pixel squared errors within each of the columns, as can be seen by the standard deviation divided by the mean that is listed at the bottom of each column. Observe that different starting points are tuned to different distance ranges. At short viewing distances (1–1.5 ft), the FS starting point results in the lowest error, while at medium distances (2–3 ft), the JJN start results in the lowest error. At even higher viewing distances (4–6 ft), the lowest error was obtained with the JJN filter with random weight perturbations [35]. Fig. 11 shows an image halftoned with LSMB using Gaussian eye filters ($h = h'$) with standard deviations 0.6, 1.2, and 1.8 pixels, corresponding to viewing distances of 1, 2, and 3 ft (3, 6, and 9 after the magnification). The starting points were those that result in the lowest error in Table II. As the viewing distance increases, one expects an increase in the number of perceived gray levels at the expense of coarser textures (which become less visible) and reduced image detail. This is indeed what we observe. For example in (b) and (c) the nose has more gray tones, the overall texture is coarser, and feathers have less detail. However, since the results are only approximate, we cannot be sure that a completely different texture will not produce a lower error. In 1-D LSMB halftoning, where the optimal solution can be obtained using the Viterbi algorithm, it was shown that the optimal textures do get coarser as the viewing distance increases [19]. The number of iterations required for convergence increases with viewing distance from about 7 to about 17, for all starting points with the exception of the random start, which requires about twice as many iterations.

Fig. 12 shows a ramp halftoned with LSMB using Gaussian eye filters ($h = h'$) with standard deviations 0.6, 1.2, and 2.4 pixels, corresponding to viewing distances of 1, 2, and 4 ft (3, 6, and 12 after the magnification). The starting points are also shown in the figure and were those that result in the lowest error in Table III, which lists the per pixel squared errors for the whole range of viewing distances and various starting points. Observe that for the given MP-MED starting points the algorithm preserves most of the texture patterns, each of which is matched to the viewing distance. Again, Table III demonstrates that different starting points are tuned to different distance ranges. Notice also the asymmetries in the nearly white regions of the MP-MED images of Fig. 12:
The slanting strings of dots are caused by the raster-scan processing. In addition, the slanting is different when the white end of the ramp is at the top or at the bottom of the image. Such asymmetries are eliminated by the LSMB algorithm. Finally, notice that, as Mulligan and Ahumada [24] demonstrated in a similar experiment, when the viewing distance decreases (i.e., the eye filter becomes narrower), large white areas appear in the lighter parts of the image. This is because the LSMB metric penalizes dots that are farther apart than the distance over which the eye model can average. Similarly, the extent of the black areas in the darker parts of the image increases with decreasing viewing distance. Notice, however, that the black areas are not as large as the white ones because of the asymmetry in the printer model.

We also tried the toggle/swap scheme [26]. As mentioned in the previous section, Flohr et al. [27] showed that this scheme produces higher quality images than the toggle-only scheme. Our results also show that the toggle/swap scheme produces halftone images of higher quality and lower error than the toggle-only scheme, and indicate a weaker dependence on the starting point for viewing distances of 1–2 ft. This is demonstrated in Table IV, which lists the per pixel squared errors for the Lena image for the whole range of viewing distances and the different starting points. Notice the similarity of the elements within each of the first few columns of the table, in comparison to the variability in the elements of the columns of Table II. The MP-MED starting points still result in the lowest error, however, with the FS filter slightly better at short viewing distances (1–1.5 ft) and the JJN filter slightly better at medium distances (2–3 ft). The differences in visual quality with different starting points are small and agree with the metric predictions. At higher viewing distances the dependence on the starting point becomes stronger, which indicates that the toggle/swap scheme is not as effective, in the sense that it does not produce a result as close to the local optimum found by exhaustive search. Again, as the viewing distance increases, the number of perceived gray levels increases at the expense of coarser textures and reduced image detail, and the number of iterations for convergence increases. The numbers are comparable to the toggle-only case. Here, however, the random start does not require as many iterations (only about 25% more than the other starting points).

In an attempt to challenge the conjecture that coarser textures are necessary for an increase of perceived gray levels, we tried the progressive scheme described in the previous section. That is, we used the LSMB estimate at a given viewing distance as a starting point for the next viewing distance, starting at 12 in with 6-in increments. We used the MP-MED with the FS or JJN filter as the original starting point, and the toggle/swap scheme at each viewing distance. The results are shown in Table V and Fig. 13. Observe that the errors are lower than those for the earlier results, and the changes in texture with increased viewing distance are a lot less dramatic. However, one can still detect slightly coarser textures as the viewing distance increases. Clearly, the use of this progressive scheme has a significant effect on the resulting images, both visually and in terms of the error metric. In this case, the number of iterations for convergence decreases slightly with viewing distance from 8 to 4; however, the whole sequence has to be computed.

We now show how we can control the amount of sharpening by varying the eye filter corresponding to the continuous-tone image. Fig. 14 shows an image halftoned with LSMB using a fixed Gaussian filter $h$ with standard deviation 1.5 pixels, corresponding to a viewing distance of 2.5 ft (7.5 ft after magnification), and different Gaussian filters $h'$ with standard deviations 1.5, 0.9, and 0 pixels (the last filter is a unit impulse). As expected, the narrower the impulse response of the filter $h'$, the sharper the halftone image. Thus, the choice of the filter $h'$ can be used to control the amount of smoothing (or sharpness) of the halftone images. In fact, the MED result

<table>
<thead>
<tr>
<th>Viewing distance (feet)</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random pattern</td>
<td>833.3</td>
<td>108.8</td>
<td>49.1</td>
<td>17.5</td>
<td>7.09</td>
<td>1.68</td>
<td>0.57</td>
<td>0.310</td>
</tr>
<tr>
<td>Blue-noise screen</td>
<td>836.6</td>
<td>156.1</td>
<td>41.6</td>
<td>13.6</td>
<td>5.29</td>
<td>1.22</td>
<td>0.43</td>
<td>0.226</td>
</tr>
<tr>
<td>MP-MED (FS filter)</td>
<td><strong>826.5</strong></td>
<td><strong>155.4</strong></td>
<td><strong>41.5</strong></td>
<td><strong>13.7</strong></td>
<td><strong>5.36</strong></td>
<td><strong>1.23</strong></td>
<td><strong>0.43</strong></td>
<td><strong>0.233</strong></td>
</tr>
<tr>
<td>MP-MED (JJN filter)</td>
<td>835.1</td>
<td>102.8</td>
<td>41.7</td>
<td><strong>12.7</strong></td>
<td><strong>4.78</strong></td>
<td>1.09</td>
<td>0.39</td>
<td>0.211</td>
</tr>
<tr>
<td>MP-MED (Stucki filter)</td>
<td>829.1</td>
<td>157.5</td>
<td>40.6</td>
<td>12.8</td>
<td>4.93</td>
<td>1.14</td>
<td>0.40</td>
<td>0.222</td>
</tr>
<tr>
<td>MP-MED (FS filter with rand. pert.)</td>
<td>829.8</td>
<td>160.5</td>
<td>43.5</td>
<td>14.1</td>
<td>5.49</td>
<td>1.22</td>
<td>0.42</td>
<td>0.230</td>
</tr>
<tr>
<td>MP-MED (JJN filter with rand. pert.)</td>
<td>831.0</td>
<td>171.5</td>
<td>49.0</td>
<td>15.2</td>
<td>5.54</td>
<td>1.13</td>
<td><strong>0.37</strong></td>
<td><strong>0.205</strong></td>
</tr>
<tr>
<td>normalized standard deviation</td>
<td>0.01</td>
<td>0.09</td>
<td>0.20</td>
<td>0.29</td>
<td>0.34</td>
<td>0.39</td>
<td>0.36</td>
<td>0.37</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Viewing distance (feet)</th>
<th>1</th>
<th>1.5</th>
<th>2</th>
<th>2.5</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>MP-MED (FS filter)</td>
<td>826.5</td>
<td>142.0</td>
<td>36.0</td>
<td>11.4</td>
<td>4.24</td>
<td>0.88</td>
<td>0.285</td>
<td>0.154</td>
</tr>
<tr>
<td>MP-MED (JJN filter)</td>
<td>835.1</td>
<td>142.3</td>
<td>35.4</td>
<td>11.0</td>
<td>4.13</td>
<td>0.87</td>
<td>0.285</td>
<td>0.153</td>
</tr>
</tbody>
</table>

The slanting strings of dots are caused by the raster-scan processing. In addition, the slanting is different when the white end of the ramp is at the top or at the bottom of the image. Such asymmetries are eliminated by the LSMB algorithm. Finally, notice that, as Mulligan and Ahumada [24] demonstrated in a similar experiment, when the viewing distance decreases (i.e., the eye filter becomes narrower), large white areas appear in the lighter parts of the image. This is because the LSMB metric penalizes dots that are farther apart than the distance over which the eye model can average. Similarly, the extent of the black areas in the darker parts of the image increases with decreasing viewing distance. Notice, however, that the black areas are not as large as the white ones because of the asymmetry in the printer model.

We also tried the toggle/swap scheme [26]. As mentioned in the previous section, Flohr et al. [27] showed that this scheme produces higher quality images than the toggle-only scheme. Our results also show that the toggle/swap scheme produces halftone images of higher quality and lower error than the toggle-only scheme, and indicate a weaker dependence on the starting point for viewing distances of 1–2 ft. This is demonstrated in Table IV, which lists the per pixel squared errors for the Lena image for the whole range of viewing distances and the different starting points. Notice the similarity of the elements within each of the first few columns of the table, in comparison to the variability in the elements of the columns of Table II. The MP-MED starting points still result in the lowest error, however, with the FS filter slightly better at short viewing distances (1–1.5 ft) and the JJN filter slightly better at medium distances (2–3 ft). The differences in visual quality with different starting points are small and agree with the metric predictions. At higher viewing distances the dependence on the starting point becomes stronger, which indicates that the toggle/swap scheme is not as effective, in the sense that it does not produce a result as close to the local optimum found by exhaustive search. Again, as the viewing distance increases, the number of perceived gray levels increases at the expense of coarser textures and reduced image detail, and the number of iterations for convergence increases. The numbers are comparable to the toggle-only case. Here, however, the random start does not require as many iterations (only about 25% more than the other starting points).

In an attempt to challenge the conjecture that coarser textures are necessary for an increase of perceived gray levels, we tried the progressive scheme described in the previous section. That is, we used the LSMB estimate at a given viewing distance as a starting point for the next viewing distance, starting at 12 in with 6-in increments. We used the MP-MED with the FS or JJN filter as the original starting point, and the toggle/swap scheme at each viewing distance. The results are shown in Table V and Fig. 13. Observe that the errors are lower than those for the earlier results, and the changes in texture with increased viewing distance are a lot less dramatic. However, one can still detect slightly coarser textures as the viewing distance increases. Clearly, the use of this progressive scheme has a significant effect on the resulting images, both visually and in terms of the error metric. In this case, the number of iterations for convergence decreases slightly with viewing distance from 8 to 4; however, the whole sequence has to be computed.

We now show how we can control the amount of sharpening by varying the eye filter corresponding to the continuous-tone image. Fig. 14 shows an image halftoned with LSMB using a fixed Gaussian filter $h$ with standard deviation 1.5 pixels, corresponding to a viewing distance of 2.5 ft (7.5 ft after magnification), and different Gaussian filters $h'$ with standard deviations 1.5, 0.9, and 0 pixels (the last filter is a unit impulse). As expected, the narrower the impulse response of the filter $h'$, the sharper the halftone image. Thus, the choice of the filter $h'$ can be used to control the amount of smoothing (or sharpness) of the halftone images. In fact, the MED result
of Fig. 9(c) is sharper than the LSMB result of Fig. 14(a), which indicates that error diffusion introduces some amount of sharpening. This last observation agrees with the findings of Eschbach and Knox [43], [44], who demonstrated the presence of an inherent amount of sharpening in the standard error diffusion algorithm, and showed that the amount of sharpening introduced by error diffusion can be controlled by the use of an input-dependent threshold. A more detailed study of the edge behavior of error diffusion can be found in [45]. Another approach to increase the amount of sharpening in error diffusion was presented in [46].

Finally, we tried using an eye filter (31st order FIR) that captures the Mach-band effect, i.e., the filter response decreases at low frequencies. We found that, if the two filters \( h \) and \( h' \) are not identical, then the halftone images come out blurred and unpleasant. For example, dropping the filter \( h' \) is equivalent to prefiltering the continuous-tone image with the inverse of the filter \( h \), i.e., with a band-stop filter. In other words, the algorithm tries to compensate for the Mach-band effect. Since there is no compensation for the Mach-band effect in the original continuous-tone image, the Mach-band compensated halftone image is unpleasant to the eye. On the other hand, when both images are filtered with the same filter, then the decrease at low frequencies has very little effect on the halftone images.

Before we close this section, we would like to discuss the issue of computational complexity. The goal of this paper was to develop an approach that can lead to the best possible halftone images for a given set of viewing conditions and display parameters. Hence, we did not try to optimize the computational efficiency of the LSMB algorithms in any significant way. This issue was considered extensively by Flohr et al. [27], who examined various ways for speeding up the LSMB algorithms, including table lookups and selective pixel updates during each iteration. However, as an indication of the amount of computation involved, we consider the toggle-only scheme applied to the Lena image with the FS MP-MED starting point. The execution time on an SGI O2 workstation ranged from about one minute for the shortest viewing distance to about 20 m for the longest (or about 14–70 s per iteration).

As we discussed in the previous section, the LSMB approach provides a metric of halftone image quality that takes into account the characteristics of both the display device (printer) and the human visual system. The results of this section indicate that the LSMB metric predictions agree with visual evaluations of image quality. However, it also has serious limitations because it is very difficult to capture image quality using a single number. For example, this metric is affected by image sharpness, gray-scale resolution, texture artifacts, geometric and gray-scale distortions, etc. In addition, a halftone image that has been displaced by a few pixels may have higher error than an image that has been halftoned by a clearly inferior technique, even though the first image is visually superior. Similarly, as we saw in this section, the error increases significantly with a small uniform shift in brightness, while the perceived quality is barely affected. One has to be especially careful when using this metric to evaluate the performance of techniques, like error diffusion, that introduce slight shifts in the image edges (to be more precise, it enhances the edges asymmetrically [45]) that are not objectionable to the eye but affect the error metric. On the other hand, when this metric is used carefully, it agrees remarkably well with human perception and provides a powerful tool for algorithm development and evaluation.

VI. CONCLUSION

We presented a new least-squares model-based approach to digital halftoning. It exploits both a visual and a printer model to produce an “optimal” halftoned reproduction. The 2-D least-squares solution is obtained by iterative techniques. We showed that the visual quality of the resulting halftone images depends on the starting point and the optimization strategy. Our experiments demonstrate that LSMB halftoning can produce images with high gray-scale and spatial resolution and pleasing textures. We considered the performance of the LSMB algorithms over a range of viewing...
distances. Overall the least-squares model-based approach offers a substantial improvement over conventional halftoning techniques, and also eliminates the problems associated with error diffusion. Our results also indicate that error diffusion with the Floyd–Steinberg filter is better suited for shorter viewing distances or lower display resolutions (typical CRT displays), while the Jarvis–Judge–Ninke filter is more suitable for longer viewing distances or higher display resolutions (300 dpi printers and higher). Finally, we showed that the LSMB approach gives us precise control of image sharpness.

We showed that the error criterion of the LSMB approach provides a measure of halftone image quality that takes into account the characteristics of both the display device (printer) and the human visual system and can be used to evaluate the performance of any halftoning technique. Our results indicate that, despite its limitations, the LSMB error metric agrees remarkably well with visual evaluations of image quality and provides a powerful tool for algorithm development and evaluation.

REFERENCES


Thrasyvoulos N. Pappas (M’87–SM’95) received the S.B., S.M., and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, in 1979, 1982, and 1987, respectively. He has been a Member of Technical Staff at Bell Laboratories, Murray Hill, NJ, since 1987. His research interests are in image and multidimensional signal processing. He has worked on analysis of medical images and algorithms for image and video segmentation. He has also worked on numerical analysis, optimization, and control. His recent work has been on perceptual image and video compression, model-based halftoning, color printing, and video/audio integration for teleconferencing.

Dr. Pappas is an Associate Editor and Editor for Electronic Abstracts for the IEEE TRANSACTIONS ON IMAGE PROCESSING. He is a Member of the IEEE Image and Multidimensional Signal Processing Technical Committee. He is also co-chair for the Conference on Human Vision and Electronic Imaging, sponsored by SPIE and IS&T.

David L. Neuhoff (M’74–SM’83–F’94) was born in Rockville Centre, NY, in 1948. He received the B.S.E. degree from Cornell University in 1970 and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 1972 and 1974, respectively.

In 1974, he joined the University of Michigan, Ann Arbor, MI, where he is now Professor of electrical engineering and computer science. From 1984 to 1989, he was an Associate Chairman of the Systems Science and Engineering Division, Electrical, Electronics, and Computer Science Department. He spent from September 1989 through June 1990 and from January through May 1997 on sabbatical at Bell Laboratories, Murray Hill, NJ. His research and teaching interests are in communications, information theory, and signal processing, especially data compression, quantization, image and video coding, source-channel coding, Shannon theory, high-resolution quantization theory, data synchronization, and halftoning.

Dr. Neuhoff was an Associate Editor for Source Coding for the IEEE TRANSACTIONS ON INFORMATION THEORY, from 1986 to 1989. He served on the Board of Governors of the IEEE Information Theory Society from 1988 to 1990. He chaired the IEEE Southeastern Michigan Chapter of Division I in 1978, co-chaired the 1986 IEEE International Symposium on Information Theory, Ann Arbor, and served as tutorial chair for ICASSP’95.