

AM/FM Halftoning: A Method for Digital Halftoning Through Simultaneous Modulation of Dot Size and Dot Placement

Zhen He and Charles A Bouman
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN, 47907-1285

July 17, 2002

Abstract

Conventionally, digital halftoning is accomplished by either changing the size of printed dots or changing the relative density of dots on the page. These two approaches are analogous to amplitude modulation (AM) or frequency modulation (FM) used in communications. A typical AM halftoning method, such as cluster dot screening, has very low computational requirements and good print stability. However, it suffers from low spatial resolution and Moiré artifacts. Alternatively, popular FM halftoning methods, such as error diffusion, can achieve high spatial resolution and are free of Moiré artifacts but lack the print stability required for electro-photographic printing.

In this paper, we present a new class of halftoning algorithms that simultaneously modulate both the size and density of printed dots. We call this new class of algorithms AM/FM halftoning. The major advantages of AM/FM halftoning are:

- Better stability in shadow area than dispersed dot methods through the formation of larger dot clusters.
- Better Moiré resistance than clustered dot screens through irregular dot placement.
- The ability to systematically optimize dot size and density to produce the best possible print quality at each gray level.

A specific implementation of AM/FM halftoning is developed for use with electro-photographic printers having pulse width modulation (PWM) technology. We present results using dot size and dot density curves obtained through measurement-based optimization, and demonstrate that AM/FM halftoning achieves high spatial resolution, smooth halftone textures, good printing stability, and Moiré resistances.

1 Introduction

Digital halftoning methods basically fall into two categories: those that modulate dot size or amplitude (AM), and those that modulate dot spacing or frequency (FM).

In AM halftoning, the density of dot clusters, which we define as the number of clusters per unit area, is fixed. Tone rendition is achieved by varying the size of each dot. The most commonly used AM halftoning algorithm is clustered dot screening [1]. Cluster dot screening has the advantages of low computational load, stable dot formation, and resistance to such printer artifacts as dot gain and banding. Thus, it is widely used in electro-photographic (laser) printers where a single isolated dot may not develop stably. One drawback of cluster dot screening is its limited ability to render fine detail. Moreover, the regular dot placement also makes it vulnerable to Moiré patterns when periodic patterns in the image are similar to the clustered dot frequency [2]. Thus, it is not suitable for halftoning images scanned from printed material.

In FM halftoning, dot size is fixed and the tone rendition is achieved by varying the dot density. Commonly used FM halftoning algorithms include disperse dot screening [3, 4, 5], error diffusion [6, 7, 8], and search-based

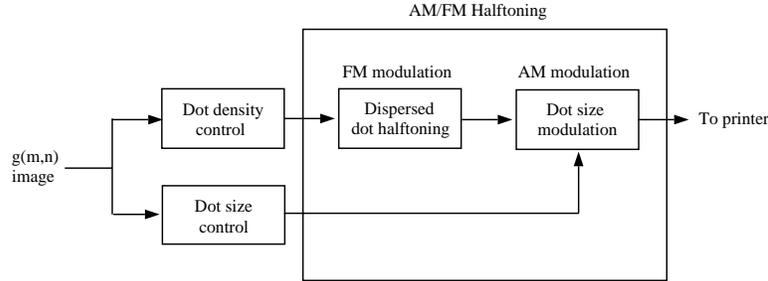


Figure 1: Diagram of the AM/FM halftoning algorithm.

halftone methods such as direct binary search (DBS) [9]. Error diffusion is by far the most popular FM halftoning algorithm. Though it needs more computation compared to screening, it is still very efficient. In general, FM halftoning achieves higher spatial resolution than AM halftoning and is free of Moiré artifacts. However, it may lack of the print stability required by electro-photographic printing. Clustered dot error diffusion was proposed in [10] to increase the print stability of error diffusion. The spectrum of such methods has been characterized as green noise [11].

In this paper, we present a new class of halftoning algorithms which combine the advantages of both AM and FM halftoning methods. These methods, which we call AM/FM halftoning, are distinct because they simultaneously modulate the dot size and dot density to produce the best quality halftone at each gray level. To halftone an image, AM/FM halftoning first determines the position of each dot using a dispersed dot halftoning algorithm and a dot density curve that relates the input gray level to the density of dots on the page. The size of each dot is then modulated according to a dot size curve. A measurement-based parameter design system is developed to optimize both the dot size and dot density curves for a particular printer.

A specific implementation of AM/FM halftoning is developed for use with electro-photographic printers having subpixel modulation ability such as Hewlett-Packard’s pulse width modulation (PWM) technology [12]. Dot size and density curves are designed via the minimization of a regularized cost function which takes both print quality and smooth texture transition into account. Experiments demonstrate that AM/FM halftoning achieves high spatial resolution, smooth halftone texture, good printing stability, and Moiré resistance.

2 AM/FM Halftoning Framework

In the following section, we present a general framework for AM/FM halftoning. Once we adopt this framework, we must then determine how to choose the dot size and dot density curves in a manner which produces the best quality rendering. We address this question in Section 2.1 by introducing a parameter design methodology which uses empirical measurements of printer response to optimize the AM/FM algorithm’s behavior.

Figure 1 illustrates how the AM/FM halftoning algorithm functions. Two look-up tables (LUT’s) are first used to determine the dot density and dot size at each pixel. The dot density is then used as the input to a dispersed dot halftoning algorithm which determines the position of dots on a discrete printing grid. The size of each printed dot is then independently modulated based on the computed dot size at that position. The resulting output is then printed.

The essential attribute of AM/FM halftoning is that it simultaneously modulates both the dot density (i.e. spacing between dots) and the dot size. Importantly, neither the dot density nor the dot size independently control the printed tone. Instead the printed tone is controlled through a combination of the two. This allows one to incorporate tone correction directly into the dot density and size LUT’s. In general, the FM part can be any disperse dot halftoning algorithm. The specific method used to modulate dot size will depend on the particular printing technology.

The dot density and dot size LUT’s are critical parameters of any AM/FM halftoning algorithm. These LUT’s must be selected to obtain the desired absorptance for each input gray level value $g(m, n)$. However, this leaves an additional degree of freedom which may be used to optimize a variety of printing attributes including print

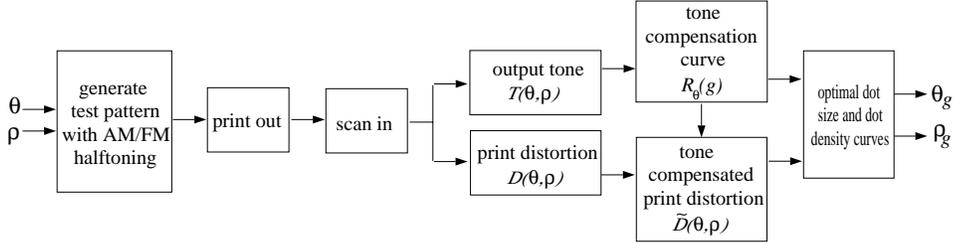


Figure 2: Diagram showing process used to design optimized LUT’s for AM/FM halftoning algorithm. The process is based on direct measurement of printed halftone quality.

quality or print stability. Therefore, it is possible to achieve the best quality rendering of a desired absorbance by either varying the dot size or the dot density.

2.1 Parameter Design for AM/FM Halftoning Algorithm

In this section, we present a general procedure for designing dot size and dot density LUT’s in order to maximize print quality while producing the desired tone response. Suppose we have selected a disperse dot halftoning algorithm to generate an FM pattern and a method to modulate the dot size. Then the disperse halftoning algorithm is first applied to generate a FM pattern with dot density ρ , and then each FM dot is modulated with dot size θ .

There are two important quantities which must be measured in order to design AM/FM LUTs. The first quantity is output tone level, $T(\theta, \rho)$. Here we assume the $T(\theta, \rho)$ is linear in absorbance, so that $T(\theta, \rho) = 1$ is a perfect black and $T(\theta, \rho) = 0$ is white. Intermediate values of $T(\theta, \rho)$ vary linearly with luminance Y . The second quantity is print distortion, $D(\theta, \rho)$, which measures the visual difference between the rendered halftone pattern and an ideally rendered gray level. For now, we only assume that $D(\theta, \rho)$ is positive, and that it decreases with improving print quality. Our objective will be to minimize the print distortion $D(\theta, \rho)$ subject to a constraint that the tone $T(\theta, \rho)$ is correct.

Figure 2 shows the general approach used for measurement based design of the dot size and dot density LUT’s. We first generate halftone patterns by printing gray patches with varying dot sizes, θ , and dot densities ρ . The print-out is then scanned in as a gray level image, and $T(\theta, \rho)$ is estimated by averaging the absorbance values of the pixels in each patch. We also compute $D(\theta, \rho)$ using a metric of print distortion that is appropriate for our problem.

The next step is to compute the tone compensation curve with respect to the desired tone curve of the printer. Let g be an integer value between 0 and 255 that specifies the nominal input gray level to the printer. The desired tone curve is denoted by $A(g)$ and is equal to the desired printed absorbance for input g . We assume that the desired tone curve is chosen so that $A(0) = A_W$ and $A(255) = A_B$, where A_B and A_W are the black and white points of the printer, respectively. For a given dot size θ and input gray level g , we may find the dot density ρ that achieves the desired absorbance by solving the equation

$$T(\theta, \rho) = A(g) . \quad (1)$$

Notice that for each fixed dot size θ , $T(\theta, \rho)$ is a monotone increasing function of the dot density ρ . Therefore, one can invert $T(\theta, \rho)$ with respect to ρ . We denote this inverse function by $T_\theta^{-1}(\cdot)$. We then compute the dot density curve which achieves the desired tone level as

$$\rho = R_\theta(g) \triangleq T_\theta^{-1}(A(g)) . \quad (2)$$

We call this the tone compensation curve for dot size θ . For each input gray value g and dot size θ , $\rho = R_\theta(g)$ is the dot density required to produce the absorbance $A(g)$. In practice, some absorbance levels may not be achievable for certain dot size values θ , even when the dot density is chosen to be its maximum value, ρ_{max} . In this case $A(g) > T(\theta, \rho_{max})$, and we set $R_\theta(g) = \rho_{max}$. After determining $T_\theta(g)$, we may compute the print

distortion of gray level g using dot size θ as

$$\tilde{D}(\theta, g) \triangleq D(\theta, \rho) \Big|_{\rho=R_\theta(g)} = D(\theta, R_\theta(g)) . \quad (3)$$

We call this the tone compensated print distortion. For any gray level g with the property that $A(g) > T(\theta, \rho_{max})$, we set $\tilde{D}(\theta, g) = \infty$, so that print distortion is infinite.

The question remains of how to choose θ and ρ for each gray level g . Denote the dot size and the dot density selected to render gray level g as θ_g and ρ_g respectively. In order to optimize the choice of (θ_g, ρ_g) , we define a cost function as an optimization objective. One choice is to use a minimum print distortion criterion, which minimizes $\tilde{D}(\theta, g)$ for each input gray level g . This optimization objective may be expressed as

$$\theta_g = \arg \min_{\theta \in [\theta_{min}, \theta_{max}]} \tilde{D}(\theta, g) \quad (4)$$

where the arg min operator returns the value of the argument that minimizes the cost function, and $[\theta_{min}, \theta_{max}]$ is the interval of possible values for θ . Equation (4) yields the value of θ_g that results in the best quality rendering for each individual gray level g . However, it may produce abrupt changes of dot size and dot density for adjacent input gray levels, which will cause visible contouring artifacts when rendering continuous tone images. To achieve smooth tone and texture transitions, we enforce smoothness constraints in the dot size and dot density curves by using regularization techniques. More specifically, we augment the cost function with quadratic penalty functions that favor solutions that are smooth. The dot size penalty function has the form

$$\frac{1}{2\sigma_1^2} \sum_{g=1}^{255} (\theta_{g-1} - \theta_g)^2 . \quad (5)$$

This function discourages large changes in dot size from one gray level to the next. The value σ_1^2 is used to control the amount of smoothing applied. When σ_1 is small, the penalty term is heavily weighted and the resulting θ_g will be very smooth. Similarly, the quadratic function applied to penalize the change of dot density from one gray level to the next is given by

$$\frac{1}{2\sigma_2^2} \sum_{g=1}^{255} (\rho_{g-1} - \rho_g)^2 \quad (6)$$

where σ_2 controls the regularization or smoothness of ρ_g . We would like to formulate our overall optimization problem in terms of $\{\theta_g\}_{g=0}^{255}$, so we can rewrite (6) using the relationship of (2) as

$$\frac{1}{2\sigma_2^2} \sum_{g=1}^{255} (R_{\theta_{g-1}}(g-1) - R_{\theta_g}(g))^2 . \quad (7)$$

Combining the expressions in (4), (5), and (7) results in the cost function

$$\mathcal{C}(\theta_0, \vec{\theta}, \theta_{255}) = \sum_{g=1}^{255} \left\{ \tilde{D}(\theta_g, g) + \frac{1}{2\sigma_1^2} (\theta_{g-1} - \theta_g)^2 + \frac{1}{2\sigma_2^2} (R_{\theta_{g-1}}(g-1) - R_{\theta_g}(g))^2 \right\} \quad (8)$$

where $\vec{\theta} \triangleq [\theta_1, \dots, \theta_{254}]$. The optimal $\vec{\theta}$ is obtained by minimizing (8)

$$\vec{\theta}^* = \arg \min_{\vec{\theta} \in [\theta_{min}, \theta_{max}]^{254}} \mathcal{C}(\theta_0, \vec{\theta}, \theta_{255}) \quad (9)$$

where $[\theta_{min}, \theta_{max}]^{254}$ is the set of feasible values for the vector $\vec{\theta}$. Notice that in (9), the boundary points θ_0 and θ_{255} are fixed in the minimization. Generally, θ_0 is set to θ_{min} to insure high quality rendering in the highlight area, and θ_{255} is set to θ_{max} to achieve the full darkness in the shadow area. When $\sigma_1 = \sigma_2 = \infty$, the cost function degenerates to the minimum print distortion criterion in (4).

Minimization of (8) is, in general, a difficult problem because the function is not convex. In practice, we have found that this cost function has complex structure with many local minima, Therefore, we need a robust optimization method.

2.2 Multi-resolution Iterative Coordinate Decent Algorithm (MICD)

One commonly used optimization method is iterative coordinate decent (ICD). This method works by iteratively updating the coordinates to minimize a cost functional. For our problem, each element of the parameter vector $\vec{\theta}$ is sequentially updated to minimize the cost function of (8). The procedure is iterated until no further decrease in the cost function is achieved. A disadvantage of ICD is that it can become trapped in local minima, as will be demonstrated in the Section 4. Here we propose a more robust method for optimization, which we call multi-resolution iterative coordinate decent (MICD).

The MICD algorithm works by optimizing the cost function after transformation of the parameter vector $\vec{\theta}$ by a series of coarse-to-fine transformations. These transformations range from a coarsest resolution of K to a finest resolution of 1. At each resolution k , ICD optimization is applied, and the result is used as an initial condition to the next finer resolution.

Consider the optimization problem for resolution k . Let $x^{(k)}$ be a vector of length $254 - k$ which is related to the vector $\vec{\theta}$ by the transformation

$$\vec{\theta} = \vec{\theta}_{int}^{(k)} + \mathbf{M}^{(k)} x^{(k)}$$

where $\vec{\theta}_{int}^{(k)}$ is the initial value of $\vec{\theta}$ at resolution k , and the elements of $\mathbf{M}^{(k)}$ are given by

$$\mathbf{M}_{ij}^{(k)} = \begin{cases} 1 & \text{if } j < i \leq j + k \\ 0 & \text{otherwise} \end{cases} .$$

Intuitively, the matrix $\mathbf{M}^{(k)}$ is chosen so that $x_j^{(k)}$ perturbs k components of $\vec{\theta}$ ranging from $\vec{\theta}_{j+1}$ to $\vec{\theta}_{j+k}$. The coarse resolution cost function is then given by

$$\mathcal{C}^{(k)}(\theta_0, x^{(k)}, \theta_{255}) = \mathcal{C}(\theta_0, \vec{\theta}_{int}^{(k)} + \mathbf{M}^{(k)} x^{(k)}, \theta_{255}) . \quad (10)$$

The MICD algorithm works by optimizing the cost function $\mathcal{C}^{(k)}(\theta_0, x^{(k)}, \theta_{255})$ at resolution k using ICD optimization. The result is then used as an initial condition for ICD optimization at resolution $k - 1$. Formally, the MICD algorithm is as follows:

1. Set $\vec{\theta}_{int}^{(K)}$ to the minimum distortion solution of (4).
2. For $k = K$ to 1

(a) Apply ICD optimization to the variable $x^{(k)}$

$$x^{(k)} \leftarrow \text{ICD arg min}_{x^{(k)}} \mathcal{C}(\theta_0, \vec{\theta}_{int}^{(k)} + \mathbf{M}^{(k)} x^{(k)}, \theta_{255})$$

(b) If $k \neq 1$, $\vec{\theta}_{int}^{(k-1)} = \vec{\theta}_{int}^{(k)} + \mathbf{M}^{(k)} x^{(k)}$

3. Set $\vec{\theta} \leftarrow \vec{\theta}_{int}^{(0)}$

Generally, we use $K = 10$ as our coarsest resolution in the optimization.

3 Detailed Implementation of AM/FM Halftoning

In this section, we describe a specific implementation of AM/FM halftoning that is suitable for the electro-photographic printers using pulse width modulation (PWM) technology [12]. PWM technology allows the size of printed dots to be modulated and is therefore ideally suited to the AM/FM method.

3.1 AM/FM Halftoning Using Pulse Width Modulation

In a conventional EP printer, the laser is either “on” or “off” during the entire period it passes a pixel’s location. This results in either a fully exposed dot, or no dot. In PWM technology, the dot can be modulated by changing

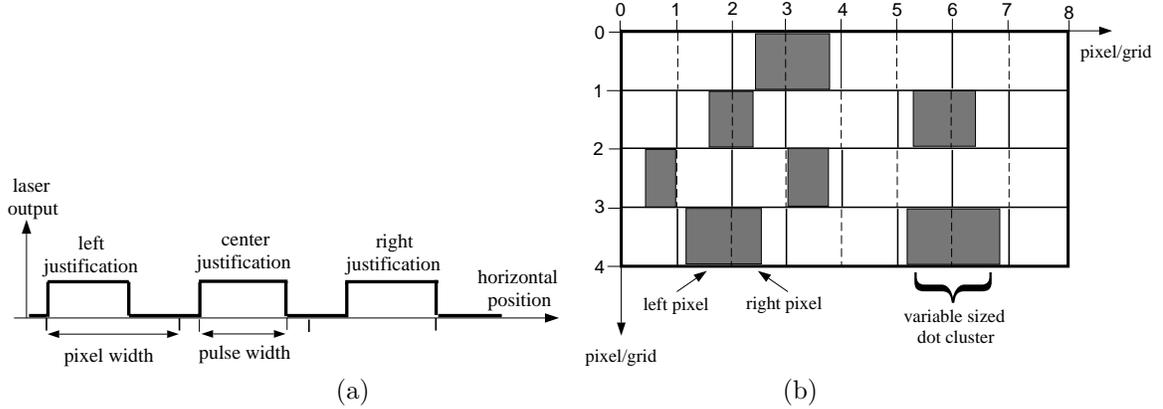


Figure 3: a) Example illustrating how the laser output can be modulated to control dot size and placement. b) Illustration of how PWM technology is used to form clustered pixel pairs arranged on a diagonal grid.

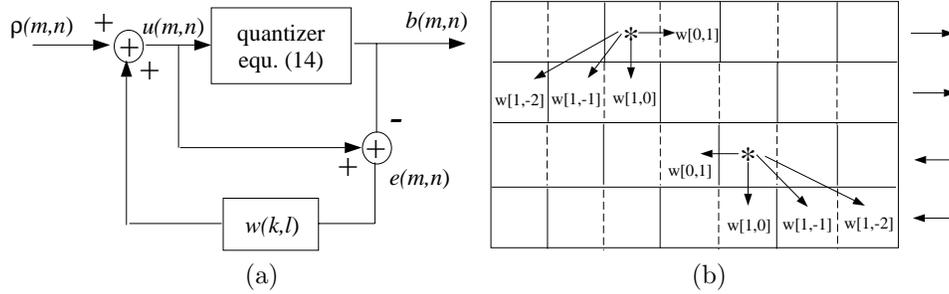


Figure 4: Modified error diffusion algorithm for the FM component of the AM/FM algorithm: a) Basic flow diagram, b) Positioning of weights and pixel processing order.

both the time that the laser is on, and the registration of the exposure in the pixel grid. Figure 3 illustrates the typical laser output patterns generated by a PWM system. For each pixel, two attributes can be controlled, the pulse width and the pulse justification. A greater pulse width increases exposure and therefore creates a darker printed pixel. The pulse justification determines whether a pixel is printed in the left, center, or right positions of the pixel grid.

In electro-photographic printing, small isolated dots tend to be highly unstable. their development can vary substantially with environmental conditions or small changes in roller speed or charge voltage. This type of instability can result in a variety of defects, including color/tone shifts, and banding artifacts. PWM technology can be used to form a more stable cluster of variable size by exposing pixel pairs as shown in Figure 3. For each pixel pair, the left pixel is right justified, and the right pixel is left justified. This results in pixels that are clustered together in pairs. The size of each pixel is also independently modulated using a pulse width value consisting of an integer between 0 and 63. Thus, the total cluster size ranges from a minimum dot size of $\theta = 0$ (total PWM value of 0) to a maximum dot size of $\theta = 2$ (total PWM value of 126).

Figure 3 shows how the 2-pixel clusters are arranged in a diagonal grid to fill all the locations on a rectangular lattice. Pairs of adjacent pixels are grouped together, but each row is offset relative to adjacent rows. Notice that the size of each pixel is modulated independently. This allows for more precise representation of edge detail. Experimentally, we have found that this diagonal packing of the pixel clusters yields effective dot placement with no overlap between adjacent clusters.

The FM modulation component determines which of the pixel pair clusters are enabled or “turned on”. For this purpose, we use a modification of conventional error diffusion which only allows dots to be fired on the diagonal grid associated with the cluster positions. The modified error diffusion algorithm is illustrated in

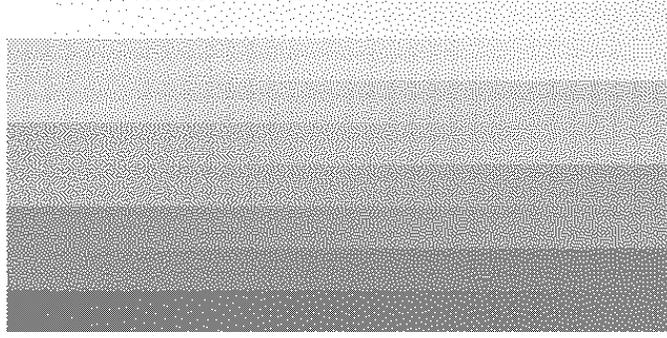


Figure 5: Serpentine ramp of the modified ED

Figure 4. Let the values (m, n) denote the row and column of a pixel respectively. Then

$$\theta(m, n) \triangleq \theta_{g(m, n)} \quad (11)$$

$$\rho(m, n) \triangleq \rho_{g(m, n)} \quad (12)$$

are the dot size and dot density at position (m, n) that results from application of the dot size and dot density look-up-tables. The modified error diffusion algorithm is then specified by the following three equations

$$u(m, n) = \rho(m, n) + \sum_{l>0} w(0, l)e(m, n-l) + \sum_l w(1, l)e(m-1, n-l) \quad (13)$$

$$b(m, n) = \begin{cases} 1 & \text{if } (u(m, n) \geq t(m, n, \rho(m, n))) \text{ and } ((m+n) \bmod 2 == 0) \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

$$e(m, n) = u(m, n) - b(m, n) \quad (15)$$

where $w(k, l)$ are error diffusion weights that normally sum to one. The major modification compared to conventional error diffusion is in the quantization step of (14) where the additional condition that $(m+n) \bmod 2 == 0$ is added. And in general, the threshold value $t(m, n, \rho(m, n))$ will be dependent on the location and the input dot density value at current pixel. The positions such that $(m+n) \bmod 2 == 0$ correspond to the left hand pixels in the pixel pair clusters of Figure 3. So this quantizer suppresses every other dot and only turns on pixels on diagonal grid. Thus the maximum fraction of pixels that can be turned on is 50%. Consequently, the input dot density is limited to the range $[0, 0.5]$.

In our experiments, both the filter weights and threshold values are dependent on input dot density $\rho(m, n)$. They are optimized for each input gray level with the method described in [13]. Using the threshold modulation method developed in [13], the threshold function $t(m, n, \rho(m, n))$ is given by

$$t(m, n, \rho(m, n)) = t_0(\rho(m, n)) + t_1(\rho(m, n))p(m, n/2; 0.5) . \quad (16)$$

Here $p(m, n; 0.5)$ is a DBS midtone pattern with 45 degree rotation to match the diagonal grid. $p(m, n; 0.5)$ takes binary value 0 or 1 and is periodic in both directions with period $M = 64$. In (16), $t_0(\rho(m, n))$ serves as basic threshold value, while $t_1(\rho(m, n))$ controls the intensity of threshold modulation. The addition of second term is to break up the possible regular halftone pattern in the area near dot density $1/8, 1/4, 3/8$. The error diffusion runs in 2-row serpentine order. And the positioning of filter weights is also shown in Figure 4. The halftone result for a serpentine ramp is given in Figure 5. Overall, it achieves fairly uniform dot placement.

The values of the error diffusion output, $b(m, n)$, at the positions corresponding to $(m+n) \bmod 2 == 0$ determine whether each pixel pair is enabled or disabled. Partial dotting is applied to compute the output PWM code at position (m, n) , which is given by

$$PWM(m, n) = \lfloor \frac{63}{2} \theta(m, n) \rfloor * b(m, n - ((m+n) \bmod 2)) \quad (17)$$

This improves resolution by allowing fine edge detailed to be rendered more accurately.

3.2 Measurement of Tone Curve and Print Distortion.

Both output tone level $T(\theta, \rho)$, and the print distortion $D(\theta, \rho)$ are measured for N_1 discrete values of dot size denoted by $\{\theta^{(i)}\}_{i=0}^{N_1-1}$, and N_2 discrete values of dot density denoted by $\{\rho^{(i)}\}_{i=0}^{N_2-1}$.

The test pattern is produced by printing an AM/FM halftoned gray level test image using a fixed dot size of $\theta^{(i)}$, and a linearly varying dot density LUT, $\rho_g = g/254$. The typical test pattern is broken into 8 sub-patterns. Each sub-pattern consists of a 16×8 array of gray patches with dot densities ranging from the minimum of 0 to the maximum of 0.5. The randomization of the patches is critical to avoid the measurement of systematic tone variations across the printed page. We have also found that the use of sub-patterns improves accuracy by increasing the spatial localization of the measurements.

For each dot size $\theta^{(i)}$, the tone curve, $T(\theta^{(i)}, \rho)$, and the print distortion curve, $D(\theta^{(i)}, \rho)$, are computed by averaging measurements over M scanned test patches, each with dot density ρ . Let $s^{(j)}(m, n)$ be the $N \times N$ sampled version of the j^{th} scanned test patch where $0 \leq m < N$ and $0 \leq n < N$. Here we assume that $s^{(j)}(m, n)$ is measured in units of absorptance which are linear with reflected energy. Further, let $S^{(j)}(k, l)$ be the DFT of $s^{(j)}(m, n)$ computed using a 2-D Hanning window to minimize boundary effects.

The initial value of the tone curve, denoted by $\tilde{T}(\theta, \rho)$, is then computed by averaging the absorptance values of the M test patches.

$$\tilde{T}(\theta^{(i)}, \rho) = \frac{1}{M N^2} \sum_{j=0}^{M-1} \sum_{(m,n)} s^{(j)}(m, n) \quad (18)$$

The distortion metric we use is based on a linear shift-invariant low-pass filter proposed in [14] with the form of

$$H(u, v) = a \exp(-b\sqrt{u^2 + v^2}) \quad (19)$$

where u and v are horizontal and vertical spatial frequency in units of cycles per degree, and a and b are constants that are dependent on average luminance of the image. Experimentally, b has been measured to be 5.169 *degree/cycle*; and for our purposes, we may assume that $a = 1$. The print distortion is computed by

$$\tilde{D}(\theta, \rho) = \sqrt{\frac{1}{M} \sum_{j=0}^{M-1} \sum_{(k,l) \neq (0,0)} F(k, l) |S^{(j)}(k, l)|^2} \quad (20)$$

where $F(k, l)$ is the filter given by

$$F(k, l) = H\left(\frac{\pi df_s}{180N} \left[\left(\left(k + \frac{N}{2}\right) \bmod N\right) - \frac{N}{2}\right], \frac{\pi df_s}{180N} \left[\left(\left(l + \frac{N}{2}\right) \bmod N\right) - \frac{N}{2}\right]\right), \quad (21)$$

d is the viewing distance in inches, f_s is the sampling frequency for $s(m, n)$ in samples per inch, and N is assumed even.

The initial tone and distortion curves are then appropriately smoothed in the ρ domain using a combination of median and linear filters. The smoothed tone curve may not be completely monotonic. To insure monotonicity, we apply the following operation.

$$T(\theta^{(i)}, \rho) \leftarrow \max_{x \leq \rho} T(\theta^{(i)}, x). \quad (22)$$

Since $T(\theta, \rho)$ and $D(\theta, \rho)$ are computed for discrete values of θ and ρ , the intermediate values of these functions must be smoothly interpolated to allow effective optimization of the cost function defined in (8). Generally, N_2 is large, so we use linear interpolation to compute intermediate values of ρ . Once this is done, the functions $R_\theta(g)$ and $\tilde{D}(\theta, g)$ may also be computed using (2) and (8). Then for each i , we then have the functions $R(\theta^{(i)}, g)$ and $\tilde{D}(\theta^{(i)}, g)$ for any g in the interval $[0, 255]$. We evaluate these functions for intermediate values of θ using cubic interpolation since this reduces interpolation error for a sparse sampling of dot size values.

4 Experimental Results

All experiments were performed on an HP4000 printer that was modified to allow pulse width modulation as described in Section 3.1. The dot size was uniformly sampled from a minimum PWM value of $\theta = 28$ to a

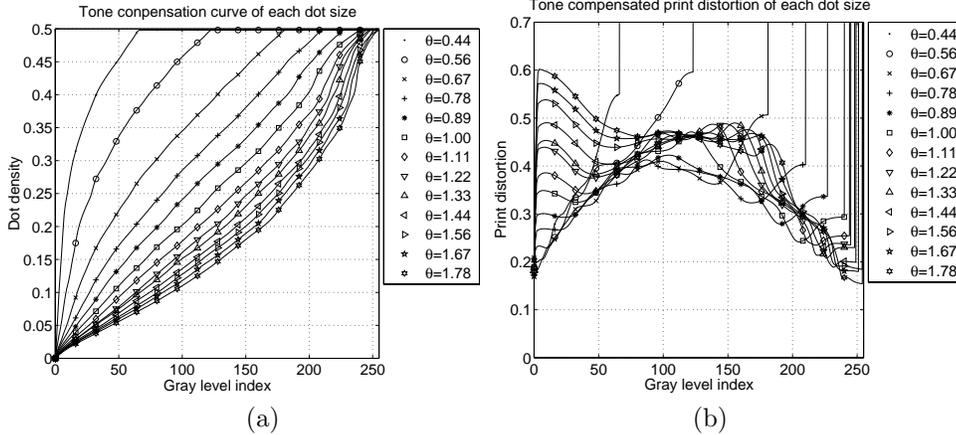


Figure 6: Measurement results of parameter design for AM/FM error diffusion: a) Tone compensation curves $\tilde{D}_{\theta^{(i)}}(g)$, b) Tone corrected print distortion curves $R_{\theta^{(i)}}(g)$.

maximum PWM value of $\theta = 122$ using a sample interval of size 7. Although the 122 is less than the maximum possible PWM value of 126, it produced a full back due to the dot gain of the printer. The test pattern contained $0.25in \times 0.25in$ test patches and was scanned at 600dpi resolution using an HP Scanjet 6100c whose output was calibrated using a GreTag reflection spectrophotometer. The scans were then converted to a linear reflectance scale. The viewing distance for the human visual system model in all experiments was chosen to be $d = 6$ inches, and the target tone curve was chosen to be

$$A(g) = A_B + (A_W - A_B) \left(1 - \frac{g}{255}\right)^\gamma$$

where the parameter γ approximately corresponds to the gamma-correction of the printer's calibration. We found that the value $\gamma = 1.5$ worked well in all our experiments.

4.1 AM/FM LUT Design Results

Figure 6 shows the tone compensation curves $R_{\theta^{(i)}}(g)$ and the tone compensated print distortion curves $\tilde{D}_{\theta^{(i)}}(g)$. As expected, greater dot density is needed for a smaller dot size to achieve a given absorbance, and it may not be possible to achieve full darkness for smaller dot sizes. Importantly, it is clear that at each gray level, g , the minimum distortion is achieved for varying values of dot size $\theta^{(i)}$.

Figure 7 shows the dot size and dot density curves that result from different choices of smoothing parameters and optimization algorithms. A smaller value of σ_1 or σ_2 constrains the dot size or dot density curve to be smoother as indicated in (8). However, as with any regularized optimization problem, there is a tradeoff between smoothness of the dot size/density curves and maximization of halftone quality at individual gray levels. For this experiment, we found that $\sigma_1 = 0.05$ and $\sigma_2 = 0.1$ yielded the best overall subjective quality.

Notice that the smaller dot size is selected in the highlight area. Intuitively, this is because the small dots are less visible in highlights. In the midtone, the dot size stays near a value of 0.70. In our experiments, we observed that dot size 0.70 achieved the highest overall quality halftone texture over a range of dot sizes for this particular printer. In the dark regions, larger dot size is preferable because this creates large "holes" which are more stable for printing.

For the case $\sigma_1 = 0.05$ and $\sigma_2 = 0.1$, we also compared the results of MICD and ICD optimization. Both the optimal dot size and dot density curves of MICD are smoother compared with those of ICD. The smooth dot size and dot density transition is crucial to avoiding abrupt halftone texture change for adjacent gray levels, which would otherwise be very visible. The converged cost function value is 86.45 for MICD optimization versus 86.74 for ICD optimization. Thus, ICD optimization is trapped in a local minimum. Generally, we have observed that the MICD optimization results in consistently lower values of the cost function.

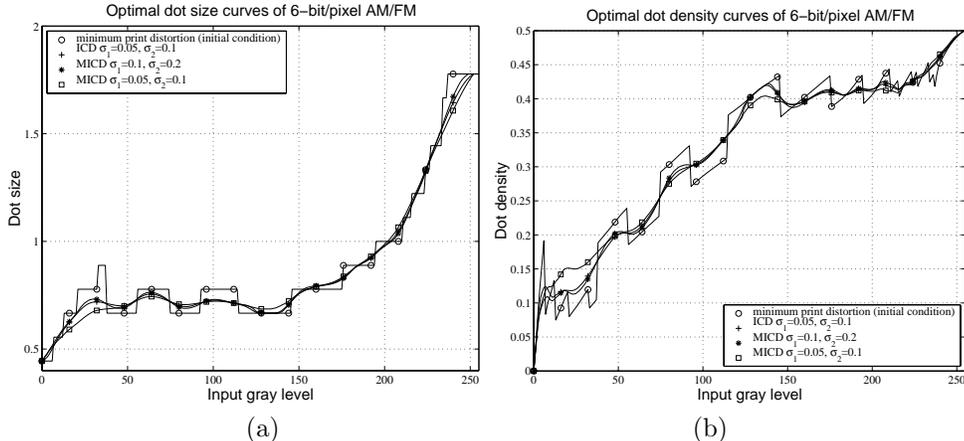


Figure 7: Plots of a) optimal dot size curves and b) optimal dot density curves for AM/FM error diffusion with 6-bit PWM codes. Each plot represents a different choice of smoothing parameters or different optimization algorithm.

4.2 Print Samples

In this section, we compare the print-outs produced using Floyd-Steinberg error diffusion and AM/FM halftoning. All figures show small sections of the original 600dpi print-outs that were subsequently scanned for inclusion in this document. Therefore the figures show the actual quality and defects that were generated in the final prints. The scan resolution was 1200dpi and all the images are rendered at 150dpi. Floyd-Steinberg error diffusion was performed at 600dpi and used PWM codes of 0 and 56 as “on” and “off”. The Floyd-Steinberg error diffusion results were also tone compensated to match the desired target tone scale.

First, we compare several gray level patches from a ramp. Figure 8 shows the results produced by Floyd-Steinberg error diffusion, and Figure 9 shows the results produced by AM/FM halftoning. In the highlight region, AM/FM halftoning achieves superior print quality because isolated dots are much less visible. Floyd-Steinberg error diffusion produces coarser textures due to the visible isolated dots and wormy halftone pattern. In the midtone region, the halftone texture transition for AM/FM halftoning is smooth, and dots are more uniformly distributed. Floyd-Steinberg error diffusion tends to produce regular directional halftone texture and also has abrupt halftone texture changes at certain gray levels. It is interesting to point out that the regular halftone texture generated by Floyd-Steinberg error diffusion makes its actual 600dpi print-out less grainy at certain midtone regions compared with that of AM/FM halftoning. In the shadow regions, AM/FM halftoning creates desirable “hole” structures which increase the printing stability. Floyd-Steinberg error diffusion produces relatively little halftone graininess in the dark area. But it also tends to show worse banding artifacts in practice due to reduced stability.¹

We next compare the spatial resolution and Moiré resistance of the halftoning algorithms. The test image consisted of scanned horizontal and vertical bars with line screen frequencies 100 and 140. Figure 10 and Figure 11 show the results of Floyd-Steinberg error diffusion and AM/FM halftoning respectively. Both algorithms achieve high spatial resolution and are free of Moiré artifacts. We think Floyd-Steinberg achieves slightly better spatial resolution. This is because we group pixels into pairs in the current implementation of AM/FM halftoning.

To compare the rendering of scanned text, we halftoned scanned text images with both algorithms. The results are shown in Figure 12. The AM/FM halftoning gives a clean rendering of text. For Floyd-Steinberg error diffusion, the result looks muddy. However, some text edges of AM/FM halftoning are not as sharp as those of Floyd-Steinberg error diffusion. This may be due to Floyd-Steinberg error diffusion’s inborn sharpening affect at edges. For this implementation of AM/FM halftoning, the error diffusion in the FM part was trained to remove the sharpening effect. Furthermore, the sharpening effect may also be weakened by the selection of

¹Banding artifacts are streaks that run along the direction of the laser (In this paper’s print-outs, the streaks are in the horizontal direction). They are generally produced by fluctuations in the movement of the OPC or drum as the laser scans.

larger dot size to render the shadow region.

5 Conclusions

In this paper, we present both a general theory and a specific implementation for AM/FM halftoning. AM/FM halftoning uses a combination of dot size and dot density modulation to produce the best possible print quality at each gray level. Our approach to designing the required dot size and dot density curves is based on regularized optimization of print quality as measured from actual printed and scanned halftones. We propose a multi-resolution iterative coordinate decent optimization algorithm for robustly optimizing the resulting nonconvex functional, and show that it performs better than a fixed scale method.

The results show that AM/FM halftoning generates high quality halftone images on electrophotographic printers. By using small dot sizes in highlights, it reduces or eliminates visible isolated dots; and by using larger dots in dark regions, it increases the stability of printed output.

References

- [1] J. c. Stoffel and J. F. Moreland, "A survey of electronic techniques for pictorial reproduction," *IEEE Trans. on Communications*, vol. 29, pp. 1898–1925, 1981.
- [2] J. P. Allebach and b. Liu, "Analysis of halftone dot profile and aliasing in the discrete binary representation of images," *J. Optical Society. America*, vol. 67(9), pp. 1147–1154, September 1977.
- [3] B. E. Bayer, "An optimum method for two-level rendition of continuous-tone pictures," *IEEE International Conference on Communications*, vol. 1, June 11-13 1973, pp. 11–15.
- [4] L. R. J. Sullivan and R. L. Miller, "Design of minimum visual modulation halftone patterns," *IEEE Trans. on Systems Man and Cybernetics*, vol. 21(1), pp. 33–38, January 1991.
- [5] R. L. M. K. E. Spaulding and J. Schidkraut, "Methods for generating blue-noise dither matrices for digital halftoning," *Journal of Electronic Imaging*, vol. 6(2), pp. 208–230, April 1997.
- [6] R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial greyscale," *Journal of the Society for Information Display*, vol. 17, no. 2, pp. 75–77, 1976.
- [7] R. A. Ulichney, "Dithering with blue noise," *Proc. of the IEEE*, vol. 76, pp. 56–79, January 1988.
- [8] B. Kolpatzik and C. A. Bouman, "Optimized error diffusion for image display," *Journal of Electronic Imaging*, vol. 1, no. 3, pp. 277–292, July 1992.
- [9] M. Analoui and J. Allebach, "New results on reconstruction of continuous-tone from halftone," *Proc. of IEEE Int'l Conf. on Acoust., Speech and Sig. Proc.*, vol. 3, March 23-26 1992, San Francisco, CA, pp. 313–316.
- [10] R. Levian, "Output dependent feedback in error diffusion," *IS&T's Eighth Int. Congr. on Adv. in Non-Impact Printing Tech.*, pp. 280–282, 1992.
- [11] R. A. D. Lau, A. Gonzalo and N. Gallagher, "Green-noise digital halftoning," *Proc. of the IEEE*, vol. 86, no. 12, pp. 2424–2444, December 1998.
- [12] C. Tung, "Resolution enhancement technooogy in hewlett-parkard laser jet printers," *Proc. SPIE Color Hard Copy and Graphic Arts II*, pp. 440–448, 1993.
- [13] P. Li and J. P. Allebach, "Tone dependent error diffusion," *IEEE Trans. on Image Processing*, submitted, 2001.
- [14] R. Nasanen, "Visibility of hafttone dot textures," *IEEE Trans. on Systems Man and Cybernetics*, vol. 14, no. 6, pp. 920–924, November/December 1984.

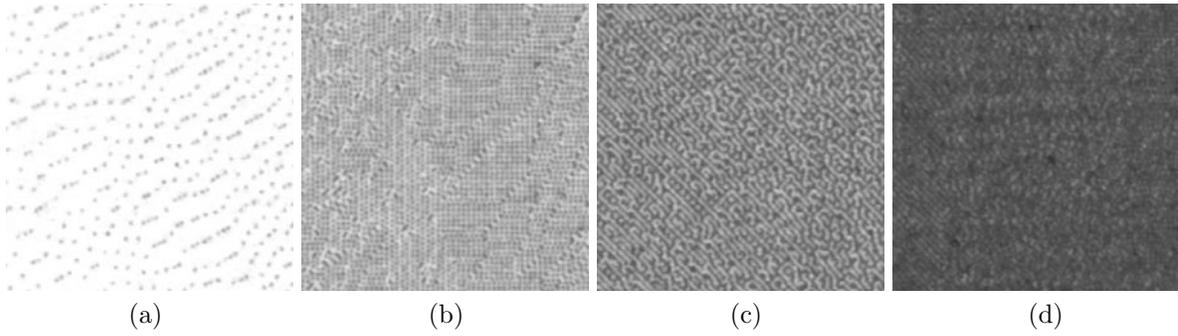


Figure 8: Scans from print-out of a ramp rendered with Floyd-Steinberg error diffusion (a). Highlight region (b). Lighter midtone region (c). Darker midtone region (d). Shadow region

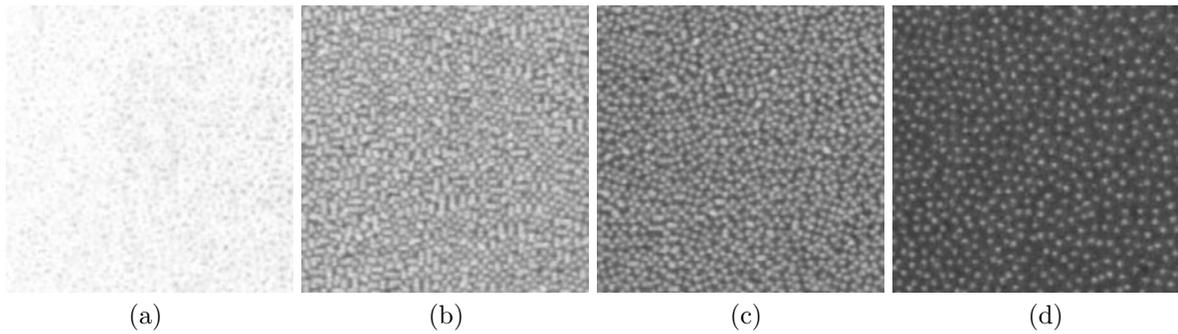


Figure 9: Scans from print-out of a ramp rendered with AM/FM halftoning (a). Highlight region (b). Lighter midtone region (c). Darker midtone region (d). Shadow region

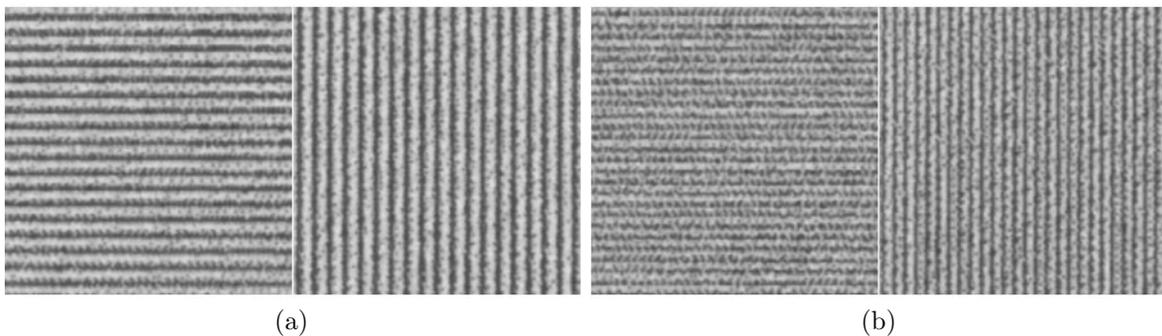


Figure 10: Scans from print out of bars rendered with Floyd-Steinberg error diffusion (a). 100 line screen frequency (b). 140 line screen frequency

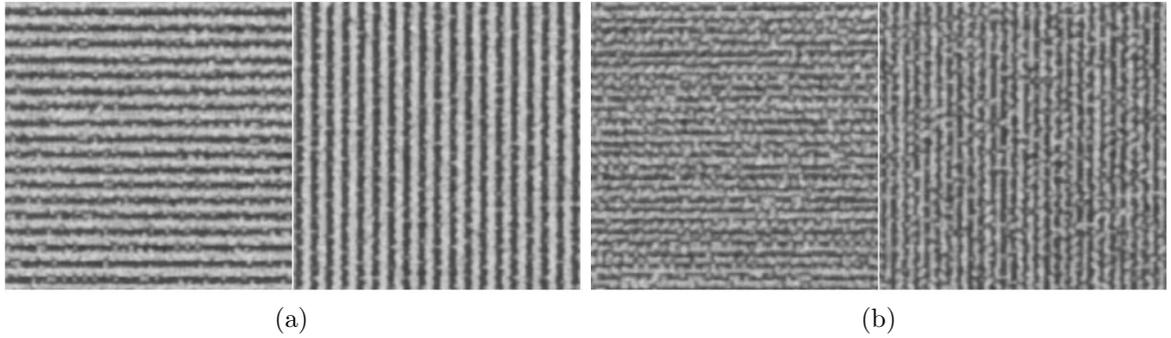


Figure 11: Scans from print out of bars rendered with AM/FM halftoning (a). 100 line screen frequency (b). 140 line screen frequency

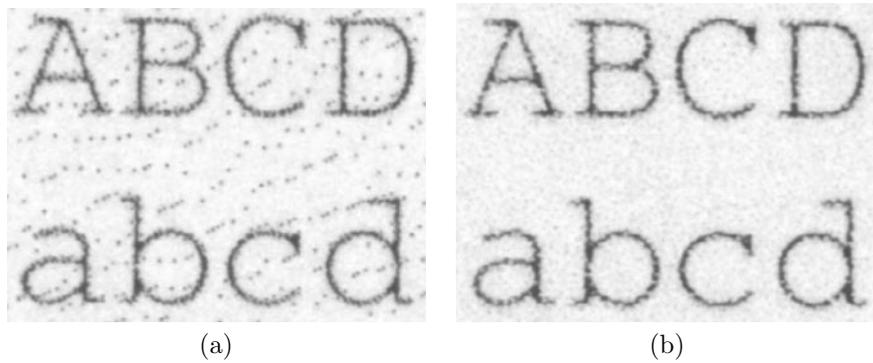


Figure 12: Comparison of rendering quality of scanned text data (a) Floyd-Steinberg error diffusion (b) AM/FM halftoning