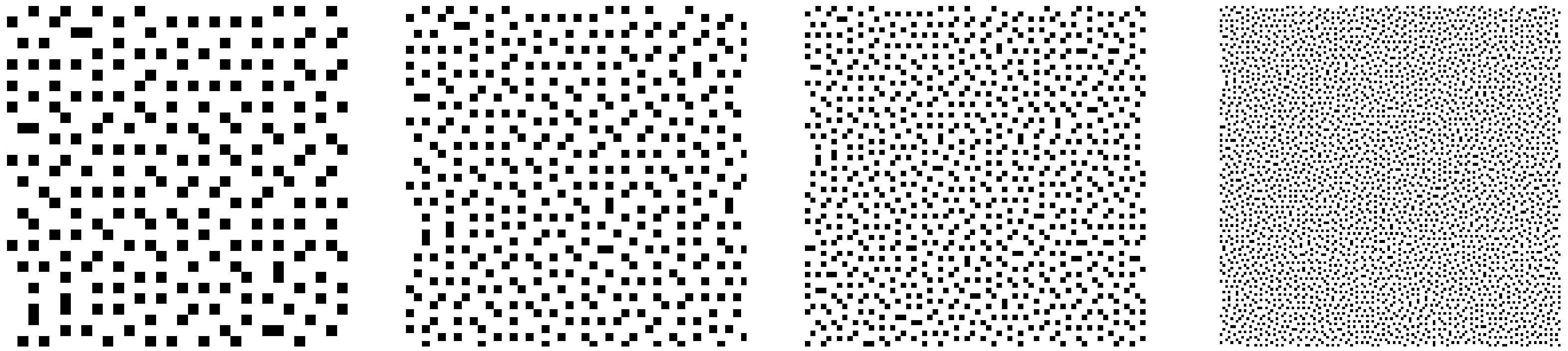
1.1 Halftoning Fundamentals

Credits/References: Prof. Jan P. Allebach, Purdue University, Lecture notes

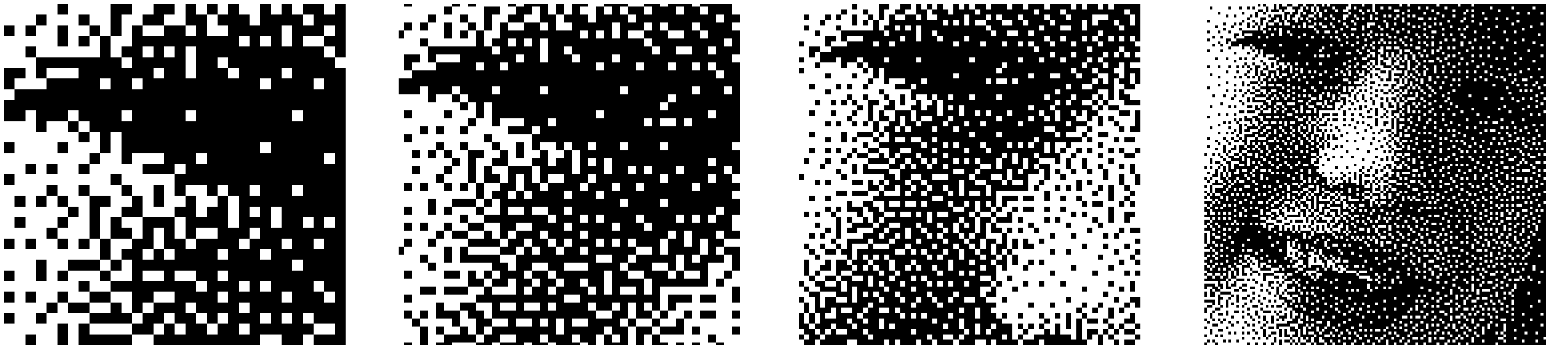
What is Digital Halftoning – How Tone is Rendered

- Digital halftoning is the process of rendering a continuous-tone image with a device that is capable of generating only two or a few levels of gray at each point on the device output surface.
- The perception of additional levels of gray depends on a local average of the binary or multilevel texture.



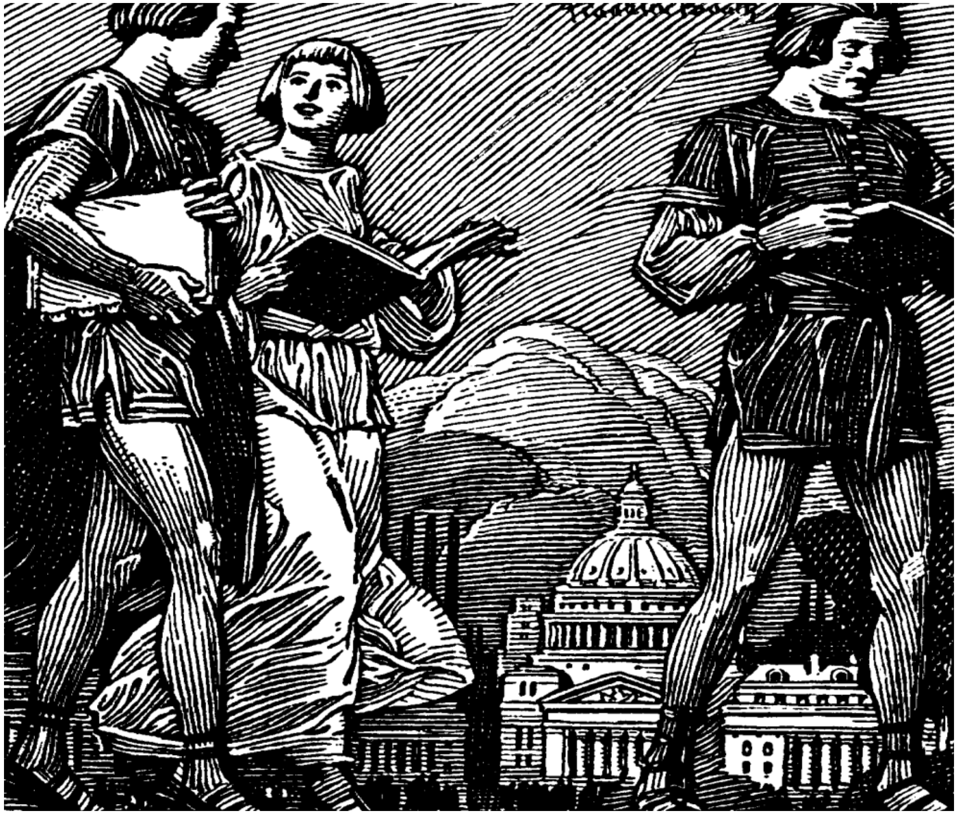
What is Digital Halftoning – How Detail is Rendered

- Detail is rendered by local modulation of the texture



History – Early Halftoning Concepts

- Fundamental concepts have been used for centuries in weaving and engraving



History – Digital Halftoning

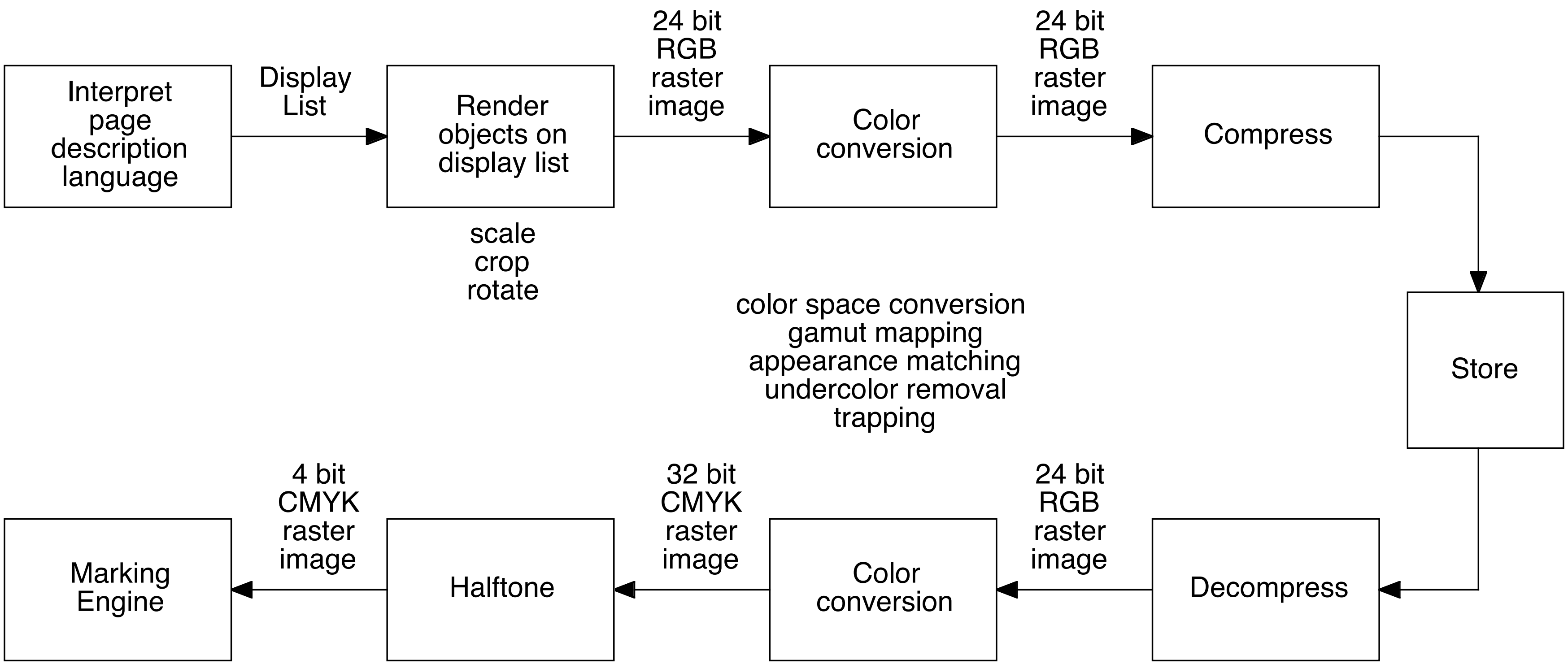
- Digital halftoning algorithms first appeared in the early 70s as computer graphics displays and hardcopy devices became more widely available.

Bayer - 1972

Floyd-Steinberg - 1976

Overview of Imaging Pipeline

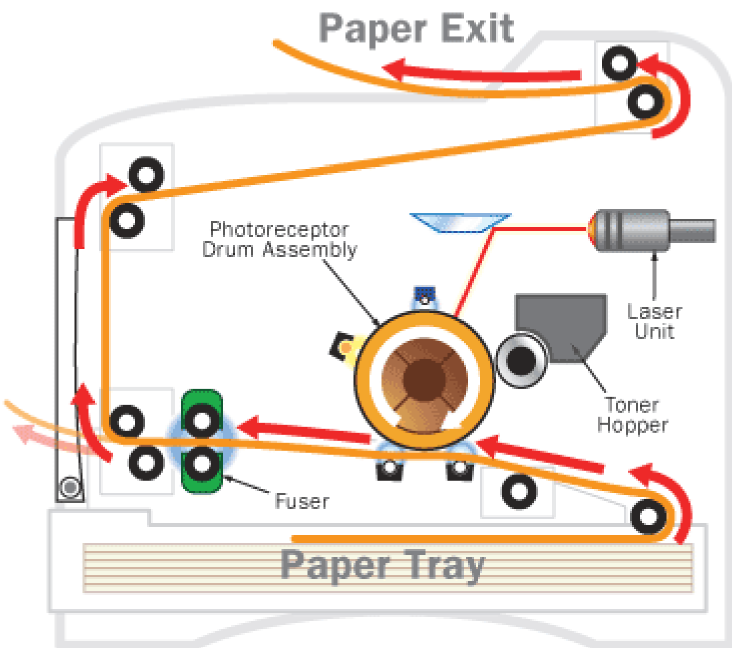
- Page description language (PDL) to print



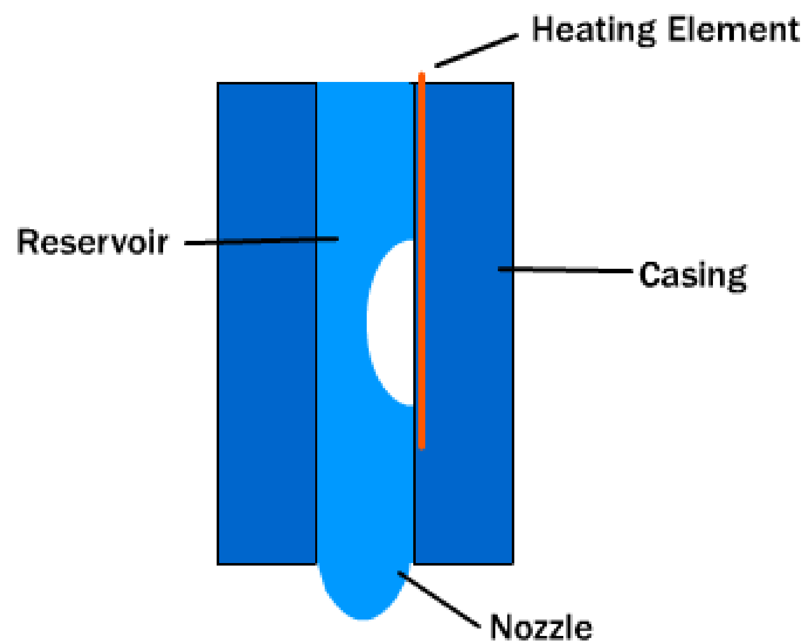
Overview of Printing Technologies

- Two dominant technologies
  - Electrophotographic (EP) process with laser or LED marking
  - Inkjet (IJ)
- Today, both these technologies span a broad range of price-points and applications
  - Low-end
    - » Home and small office applications
    - » Color EP printers < $300, color IJ printers < $50
  - Mid-range
    - » Workgroup, office, enterprise
    - » Color EP and IJ printers between $500 and $10K
  - High-end
    - » Commercial and industrial applications
    - » Color EP and IJ systems between $10K and multiples of $1M

Two Dominant Digital Printing Technologies



Laser EP Printers



Inkjet

- Traditionally:
  - EP has been used in faster, higher capacity, higher cost business platforms.
  - IJ has been used in slower, lower capacity, lower cost home/small business platforms.

Source: http://computer.howstuffworks.com/, http://www.explainthatstuff.com/inkjetprinters.html, http://www.explainthatstuff.com/laserprinters.html

Print Technology Issues

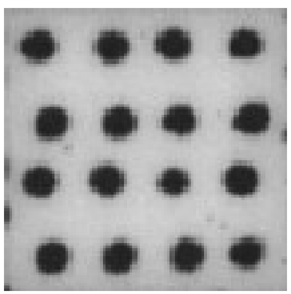
- Image quality
- Speed
- Cost of hardware
- Cost of consumables
- Permanence

Print Technology Factors that Impact Halftoning

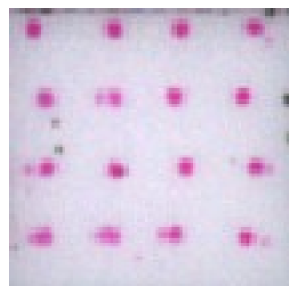
- Resolution
- Dot hardness
- Dot gain
- Dot stability
- Banding

All patterns generated with this bit map printed at 600dpi and scanned at 4000 dpi.

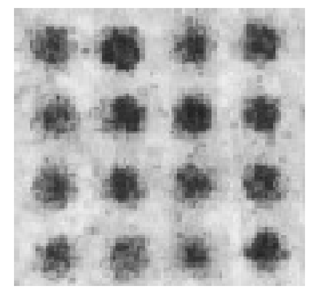
1001001
0000000
0000000
1001001
0000000
0000000
1001001



K plane - inkjet



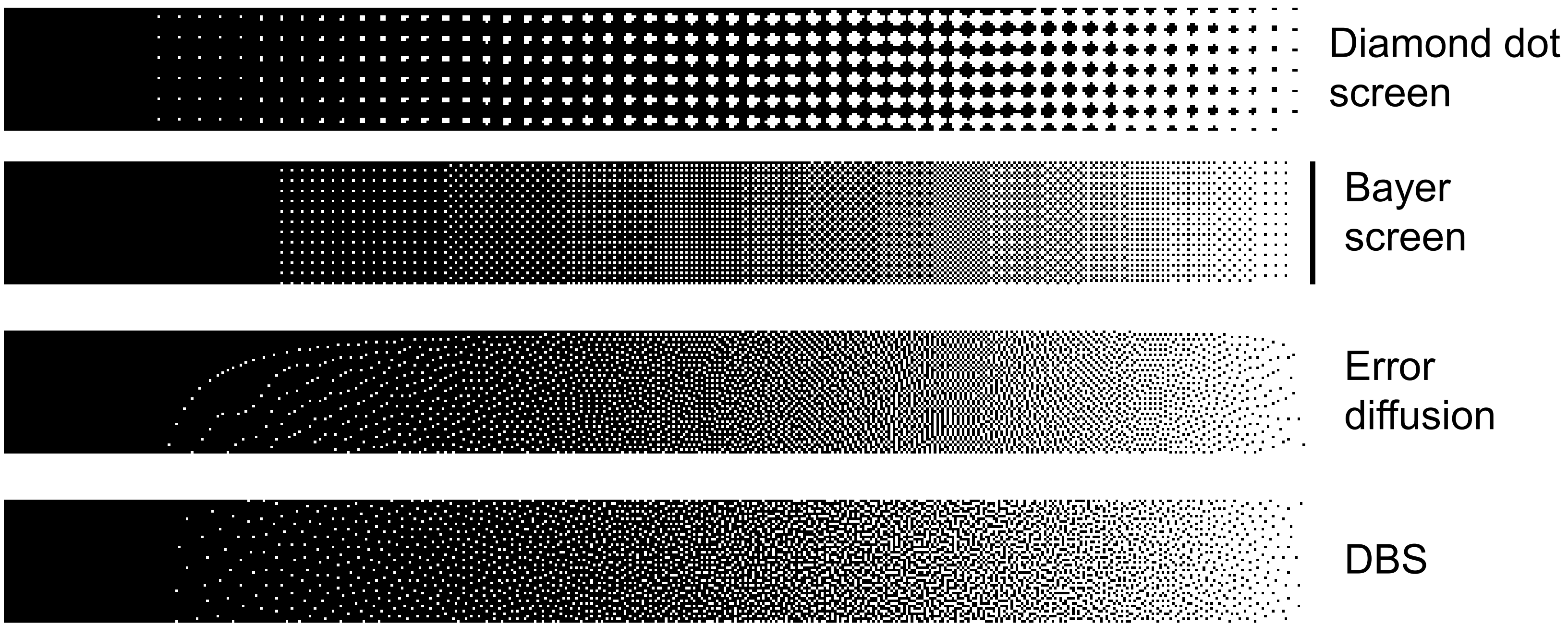
M plane - inkjet



Monochrome EP

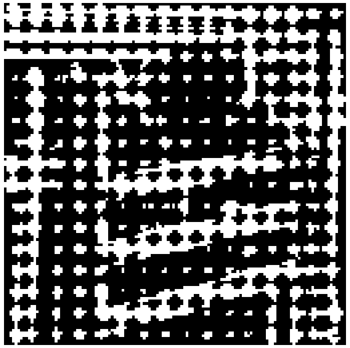
The Two Fundamental Goals of Digital Halftoning

- Representation of Tone
  - smooth, homogeneous texture.
  - free from visible structure or contouring.



The Two Fundamental Goals of Digital Halftoning (cont.)

- Representation of Detail
  - sharp, distinct, and good contrast in rendering of fine structure in image.
  - good rendering of lines, edges, and type characters.
  - freedom from moire due to interference between halftone algorithm and image content



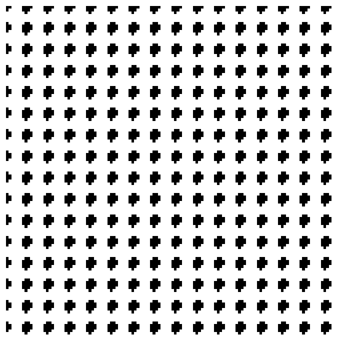
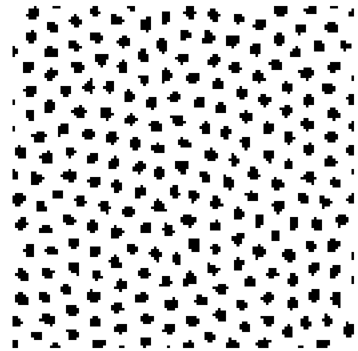
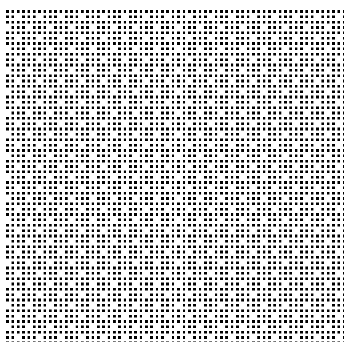
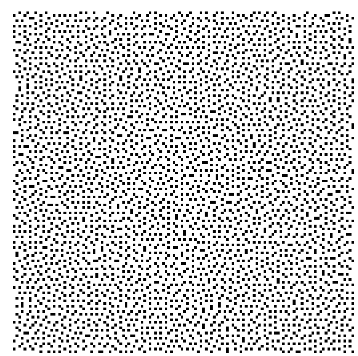
Diamond dot screen

DBS screen

Error diffusion

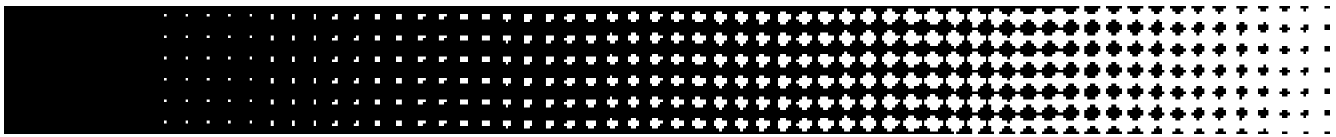
DBS

Types of Halftone Texture

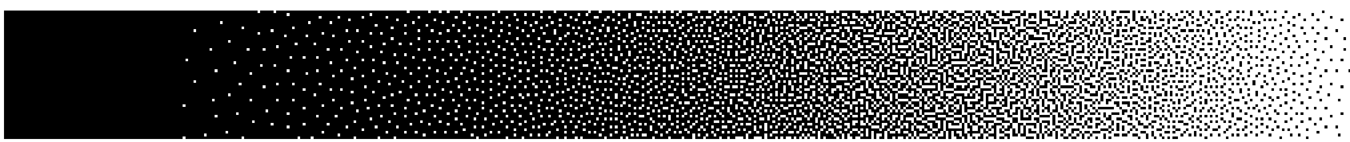
	Periodic	Aperiodic
Clustered Dot		
Dispersed Dot		

Modulation Strategies

- Amplitude modulation - dot size varies, dot spacing is fixed.



- Frequency modulation - dot spacing varies, dot size is fixed.



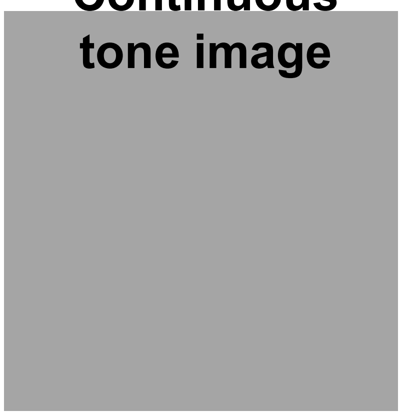
Preliminaries – Digital Halftoning

➢ **Digital Halftoning**: The process of rendering a continuous-tone image on devices, such as printers, that can only produce a small set of tone levels.

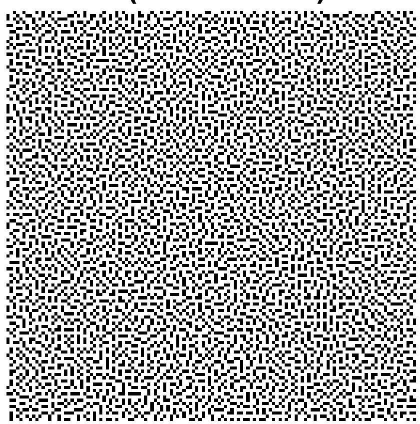
➢ Halftoning works because *human visual system* (HVS), to a first degree of approximation, acts as a *spatial low-pass filter* that blurs the rendered pixel pattern, so that it is perceived as a continuous-tone image.

Three important types of halftones

Continuous tone image

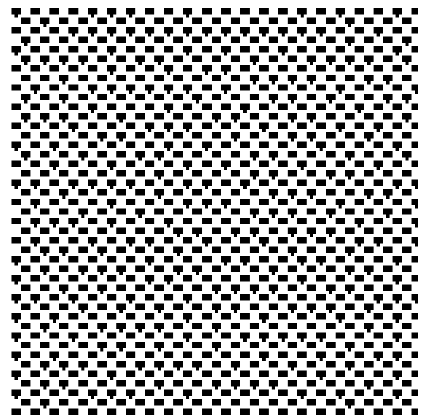


Aperiodic dispersed-dot (Blue Noise)



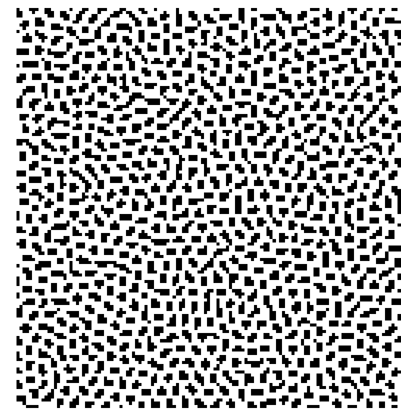
- **Good detail rendition**
- **Less stable**

Periodic clustered-dot



- **Smooth appearance**
- **Stable**
- **Visible periodic moiré**

Aperiodic clustered-dot (Green Noise)



- **Stable (Less printer-induced artifacts)**
- **No periodic moiré**

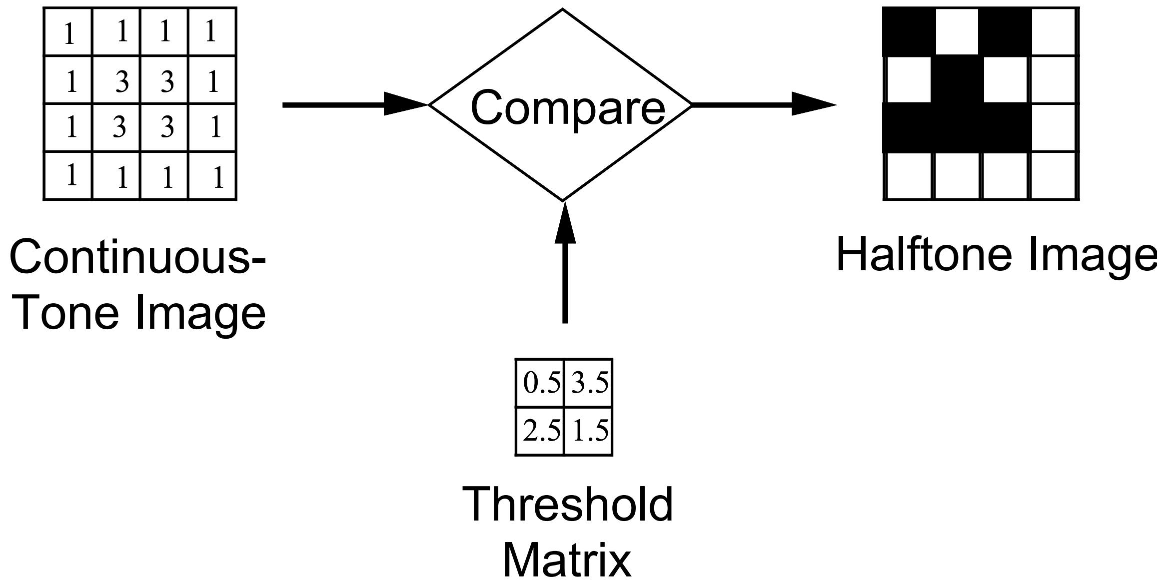
Architectures for halftoning algorithms

- There are three basic architectures for halftoning algorithms.
- Screening – point-to-point memoryless operation.
- Error diffusion – neighborhood processing.
- Search-based methods – usually are iterative.
- Computation increases in the order listed above.
- Image quality also increases in the order listed above.
- Iterative methods are useful for design of simpler, less computationally complex algorithms.

Halftoning Algorithms

	Screening Approaches	Neighboring Approaches	Iterative Approaches
Example	- Clustered-Dot Screens -- Stochastic Screen	- Error Diffusion	- DBS (Direct Binary Search) - Void and Cluster
Computational Complexity	Increasing order of computational complexity →		
Visual Performance	Increasing order of visual performance →		
Application	- Laser EP printer - High-speed digital printing	- Inkjet printer - All-in-one type (printer + scanner)	- Algorithm optimization in offline

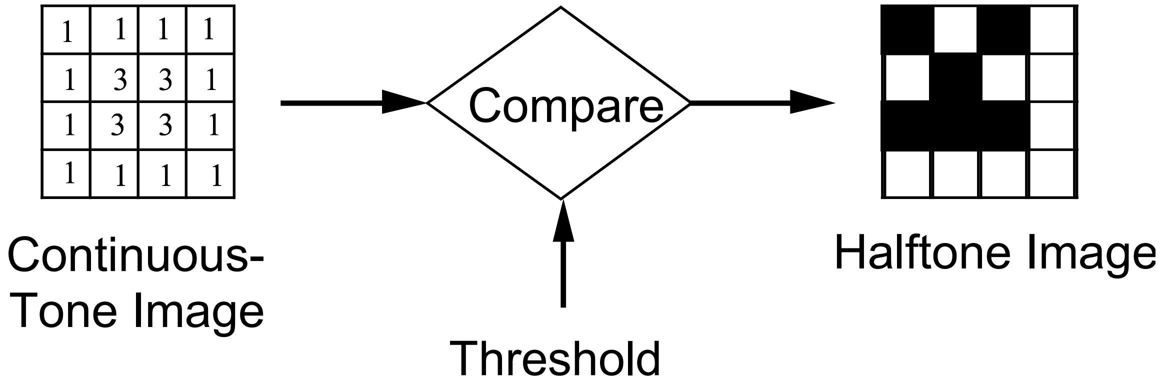
Basic Structure of Screening Algorithm



The threshold matrix is periodically tiled over the entire continuous-tone image.

Screening is a Thresholding Process

- Simple point-to-point transformation of each pixel in the continuous-tone image to a binary value.
- Process requires no memory or neighborhood information.



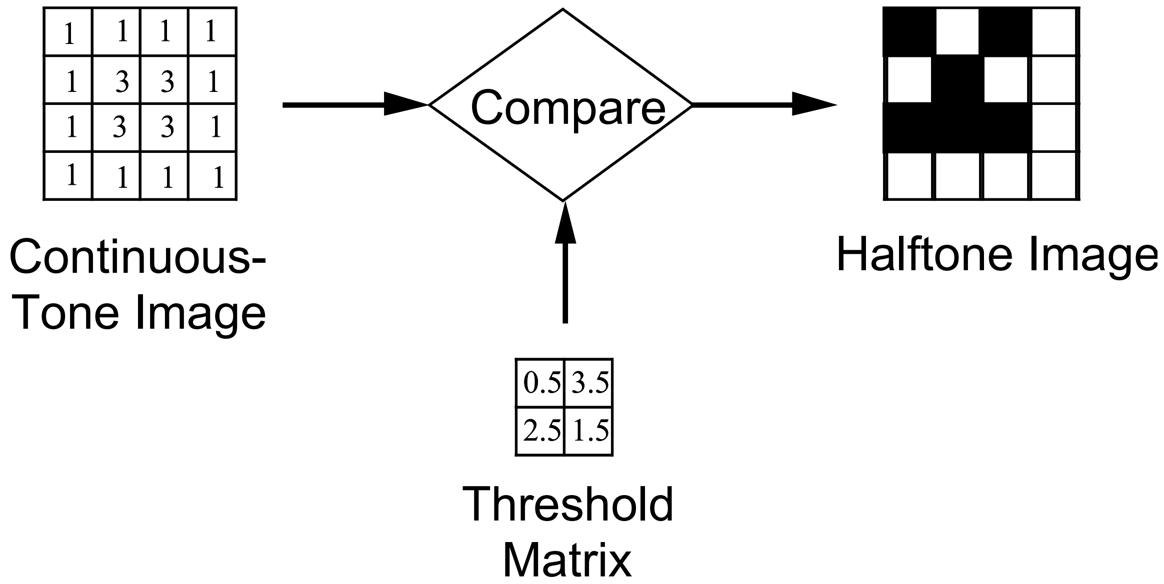
Why Not Use a Single Threshold?

- A single threshold yields only a silhouette representation of the image.
- No gray levels intermediate to white or black are rendered.
- To generate additional gray levels, the threshold must be *dithered, i.e.* perturbed about the constant value.

Continuous-tone original image

Result of applying a fixed threshold at midtone

Basic Structure of Screening Algorithm



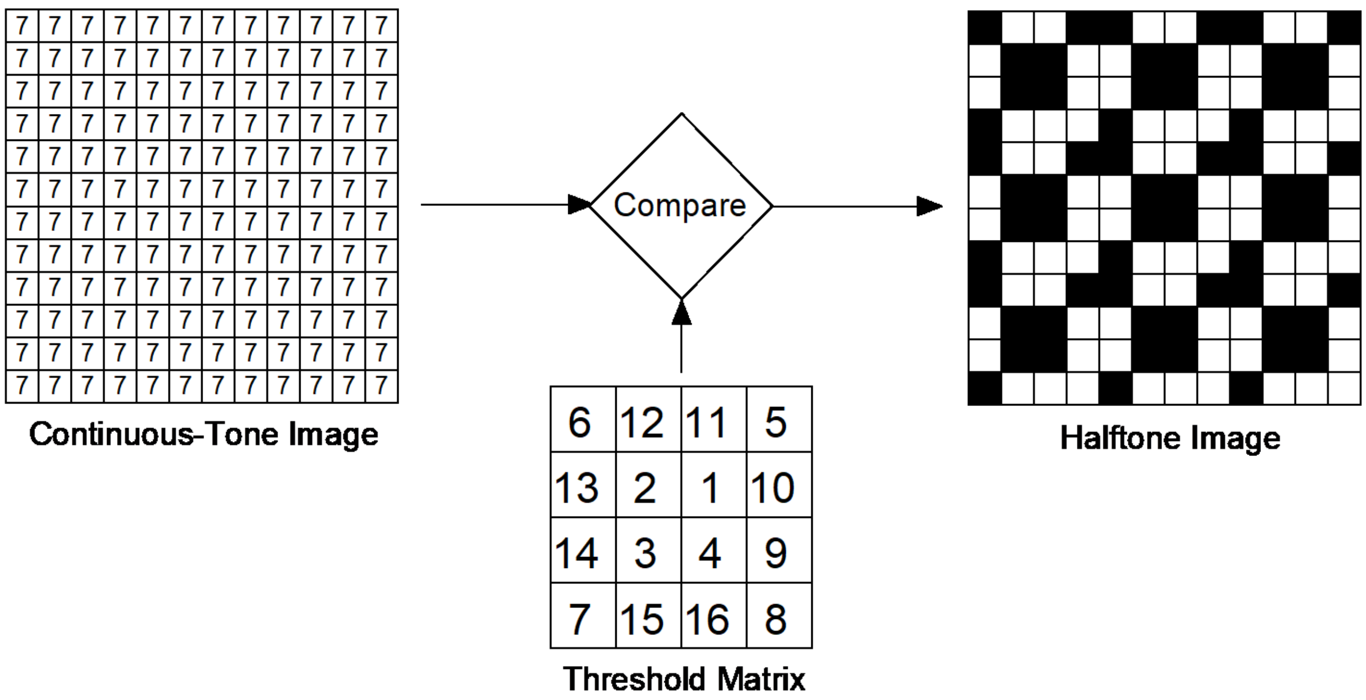
The threshold matrix is periodically tiled over the entire continuous-tone image.

Terminology

- The screening process is also called *dithering.*
- However, the term dithering is sometimes applied to any digital halftoning process, not just that consisting of a pixel-to-pixel comparision with thresholds in a matrix.
- The following are equivalent terms for the *threshold matrix*:
  - screen
  - dither matrix
  - mask

How Tone is Rendered

- If we threshold the screen against a constant gray value, we obtain the binary texture used to represent that constant level of absorptance.



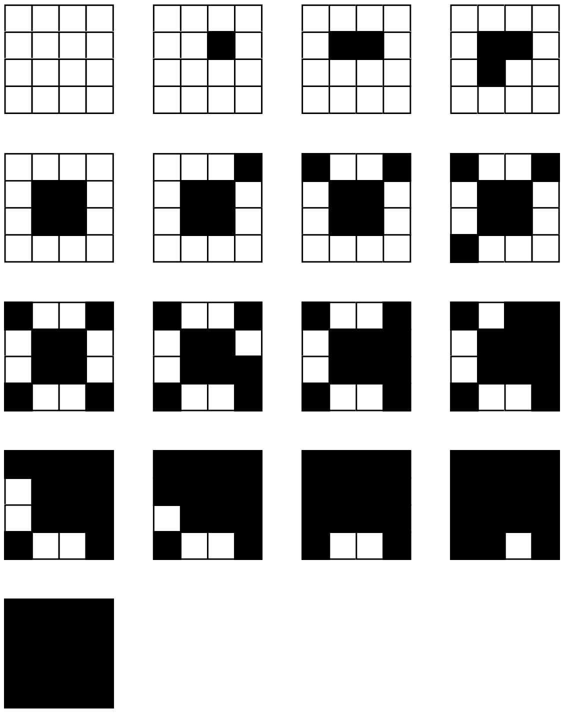
Dot Profile Function

- The family of binary textures used to render each level of constant tone is called the *dot profile function*.
- There is a one-to-one relationship between the dot profile and the screen.

Threshold Matrix

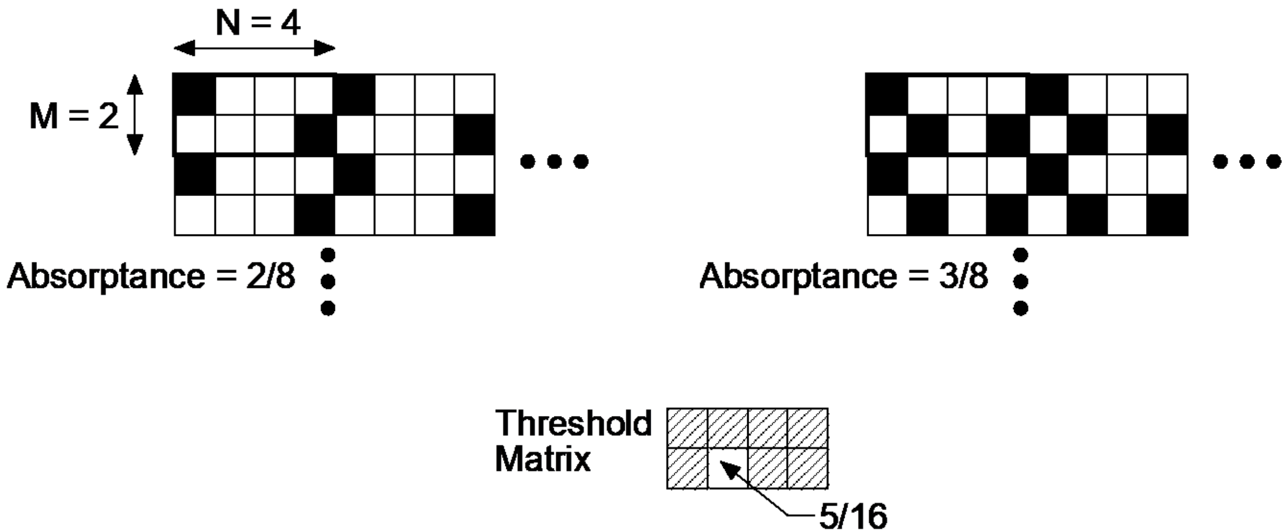
6	12	11	5
13	2	1	10
14	3	4	9
7	15	16	8

Dot Profile

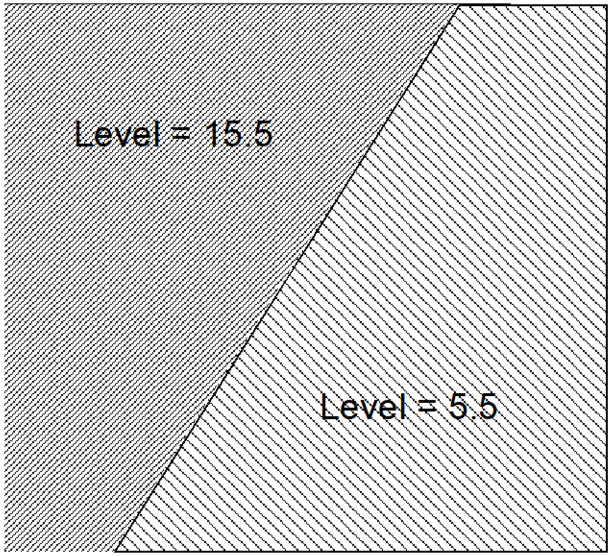


Selection of Threshold Values

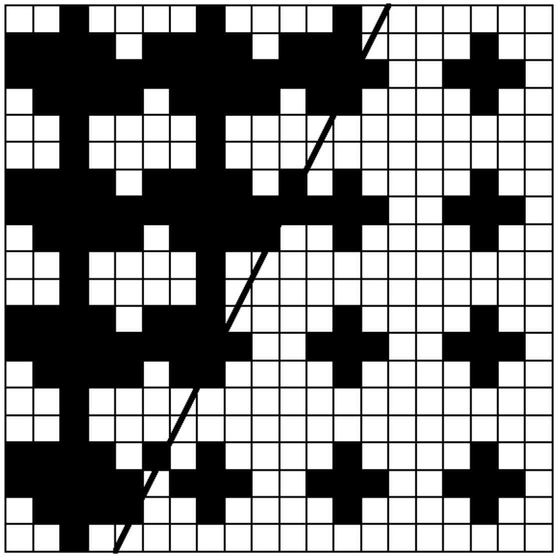
- For an MxN halftone cell,can print 0, 1, 2, …, MN dots, yielding average absorptances 0, 1/MN, 2/MN, …, 1, respectively.
- As the input gray level increases, each time a threshold is exceeded, we add a new dot, thereby increasing the rendered absorptance by 1/MN.
- It follows that the threshold levels should be uniformly spaced over the range of gray values of the input image.



Rendering of Detail - Partial Dotting



Continuous-tone image

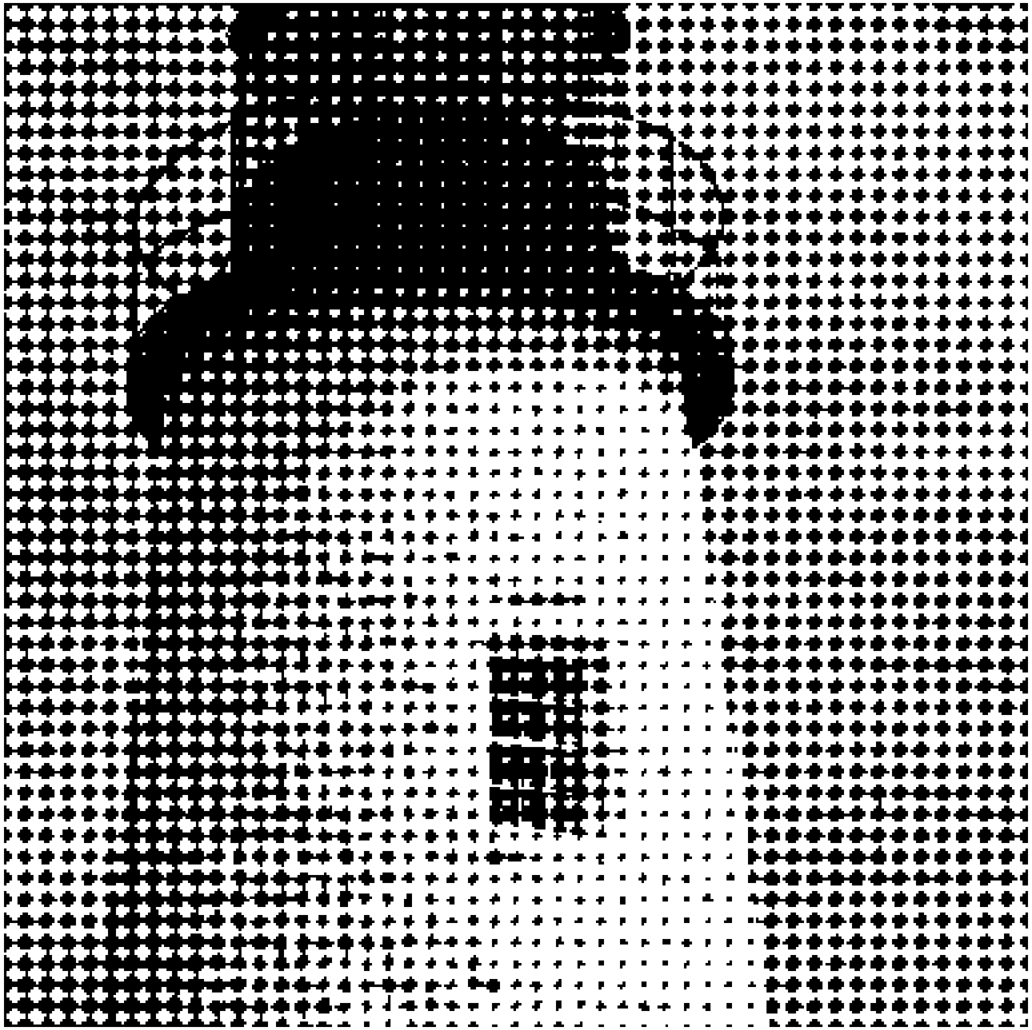


Halftone image

22	20	10	19	23
14	8	2	7	16
13	4	1	5	12
17	6	3	9	15
24	18	11	21	25

Threshold matrix

Partial Dotting - Example

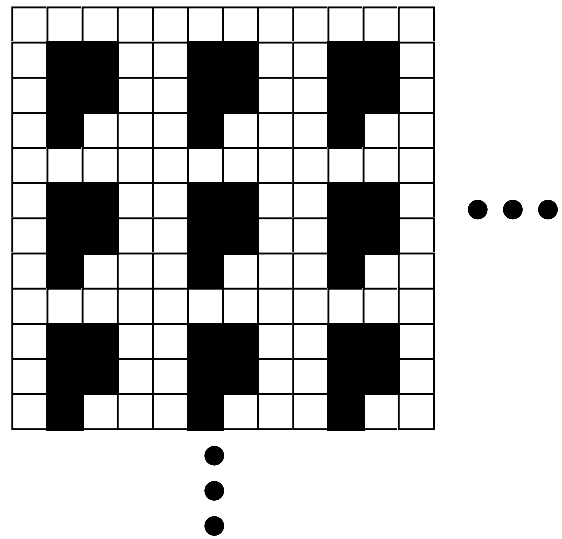


Spatial Arrangement of Thresholds

- For *clustered dot* textures, thresholds that are close in value are located close together in the threshold matrix

16	10	9	15
11	3	2	8
12	4	1	7
13	5	6	14

Threshold matrix



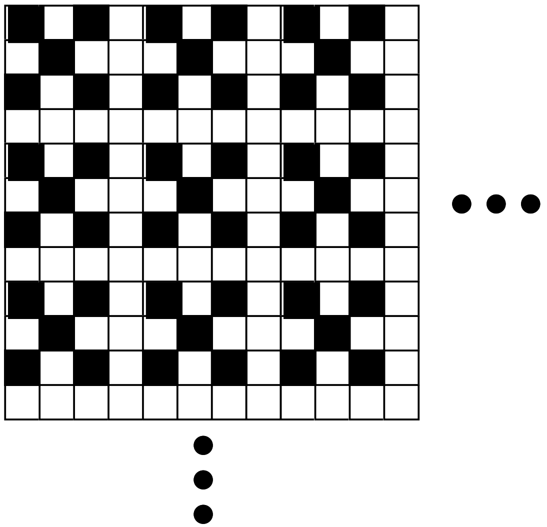
Halftone texture for gray level 5/16

Spatial Arrangement of Thresholds (cont.)

- For *dispersed dot* textures, thresholds that are close in value are located far apart in the threshold matrix.

2	10	3	13
14	5	9	7
4	12	1	11
16	6	15	8

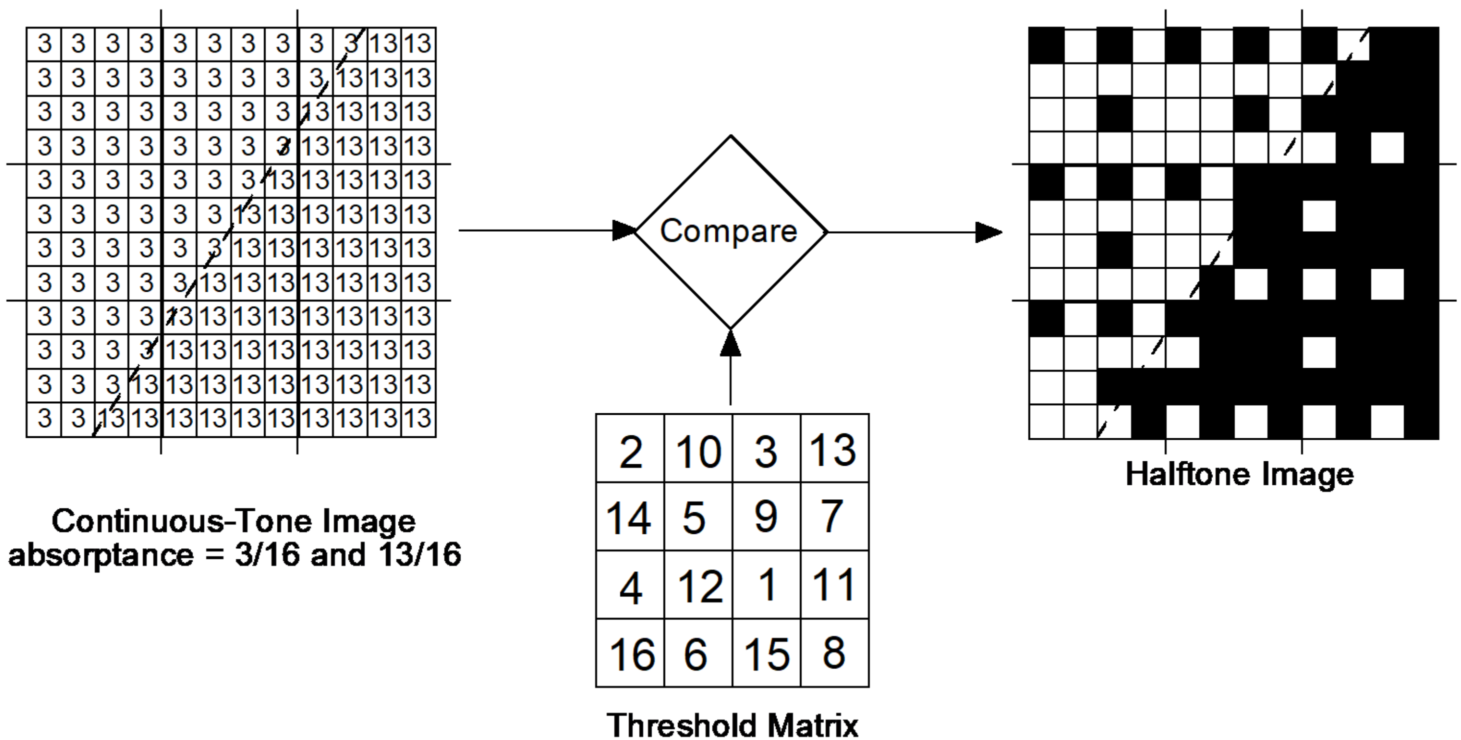
Threshold matrix



Halftone texture for gray level 5/16

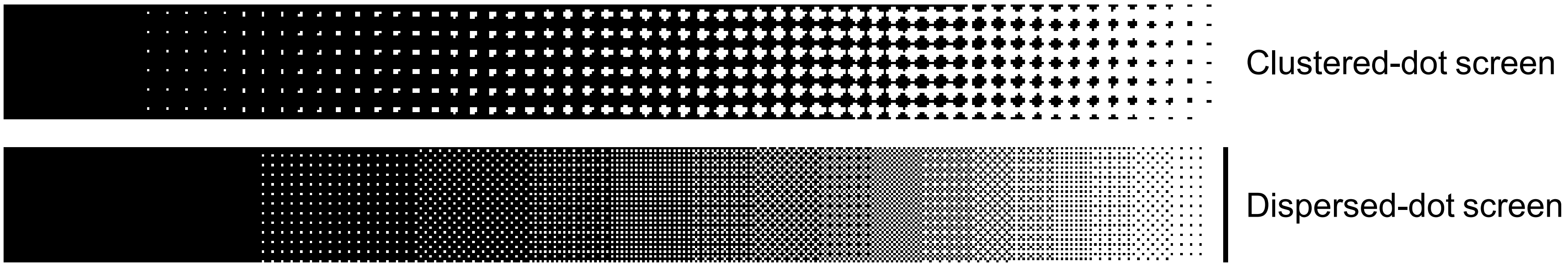
Detail Rendition with Dispersed Dot Screens

- Since the thresholds in any local neighborhood tend to be uniformly spread over the full range of gray levels, the gray level in that local neighborhood is rendered more accurately.



Clustered-Dot Screen

- Gray levels are realized by changing the clustered-dot size (AM halftoning)
- Advantage
  - Cluster is stable, robust to dot gain → widely used for electrophotographic (EP) process
- Disadvantage
  - Poor rendering details
  - Limited gray levels → contouring artifact

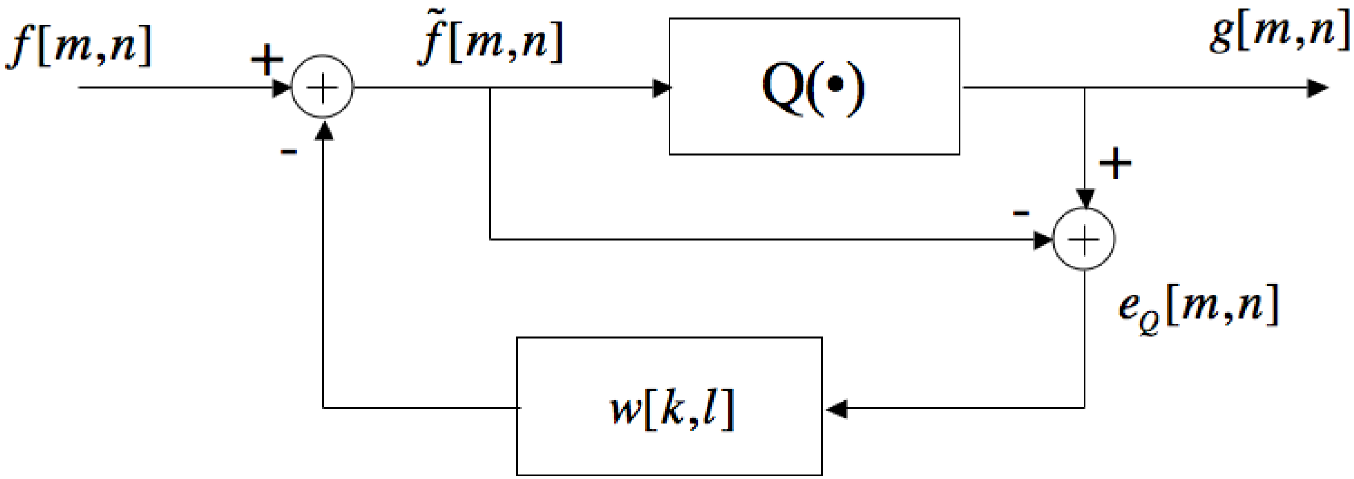


1.3 Error Diffusion – Basic Concepts

Synopsis

- Error diffusion architecture
- Error diffusion textures
- Edge enhancement effect of error diffusion
- Spectral analysis of error diffusion
- Variations on a theme
- Tone-dependent error diffusion

Error Diffusion Architecture



Definition of terms

- Continuous-tone input image – $f[m,n]$
- Modified (updated) continuous-tone image – $\tilde{f}[m,n]$
- Binary output halftone image – $g[m,n]$
- Display error – $e_D[m,n] = g[m,n] - f[m,n]$
- Quantization error – $e_Q[m,n] = g[m,n] - \tilde{f}[m,n]$

Description of the Algorithm

- Start with $\tilde{f}[m,n] \equiv f[m,n]$
- Scan pixels in image in a predetermined order, and carry out following computations
- Threshold

$$g[m,n] = \begin{cases} 1, & \tilde{f}[m,n] \geq 0.5 \\ 0, & \text{else} \end{cases}$$

- Compute quantization error

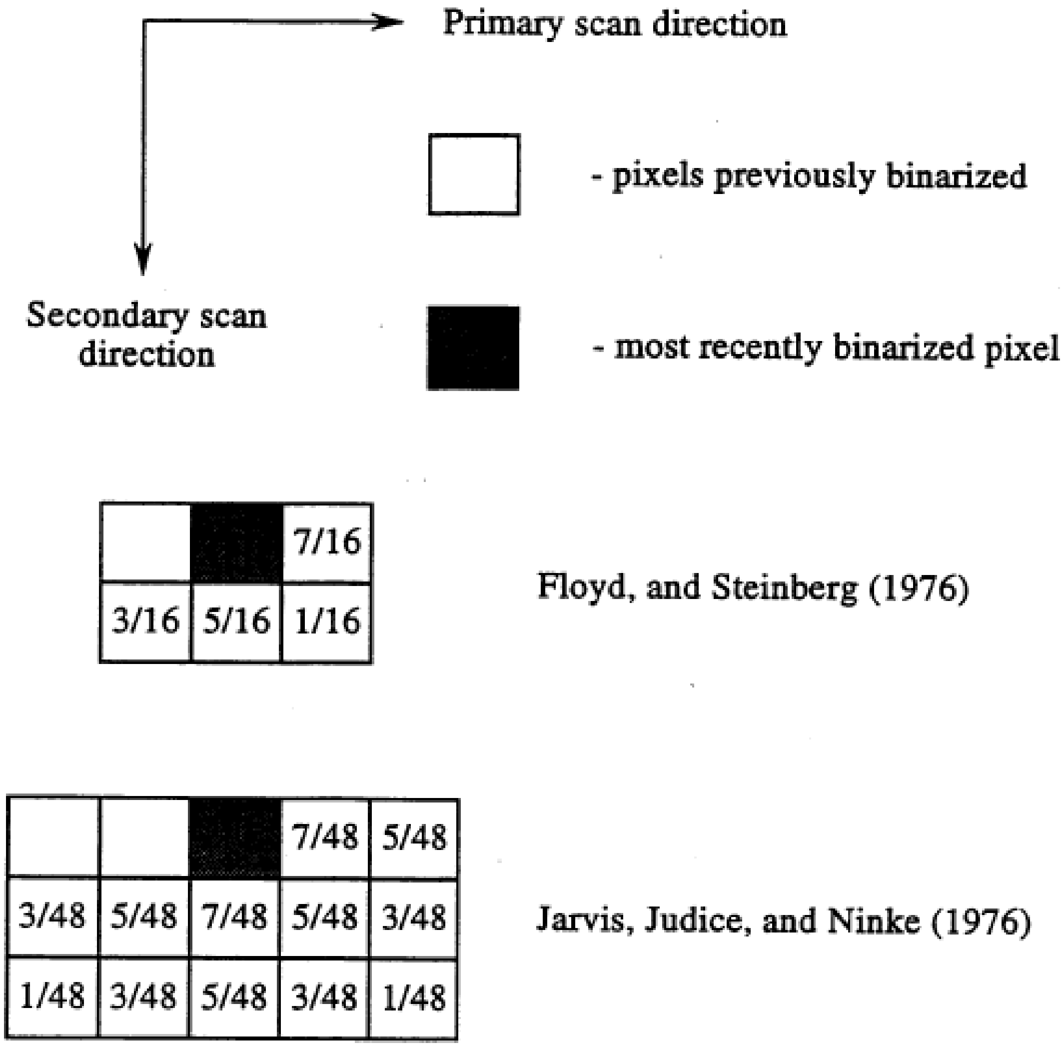
$$e_Q[m,n] = g[m,n] - \tilde{f}[m,n]$$

- Diffuse error

$$\tilde{f}[m+k,n+l] = \tilde{f}[m+k,n+l] - w[k,l]e_Q[m,n]$$

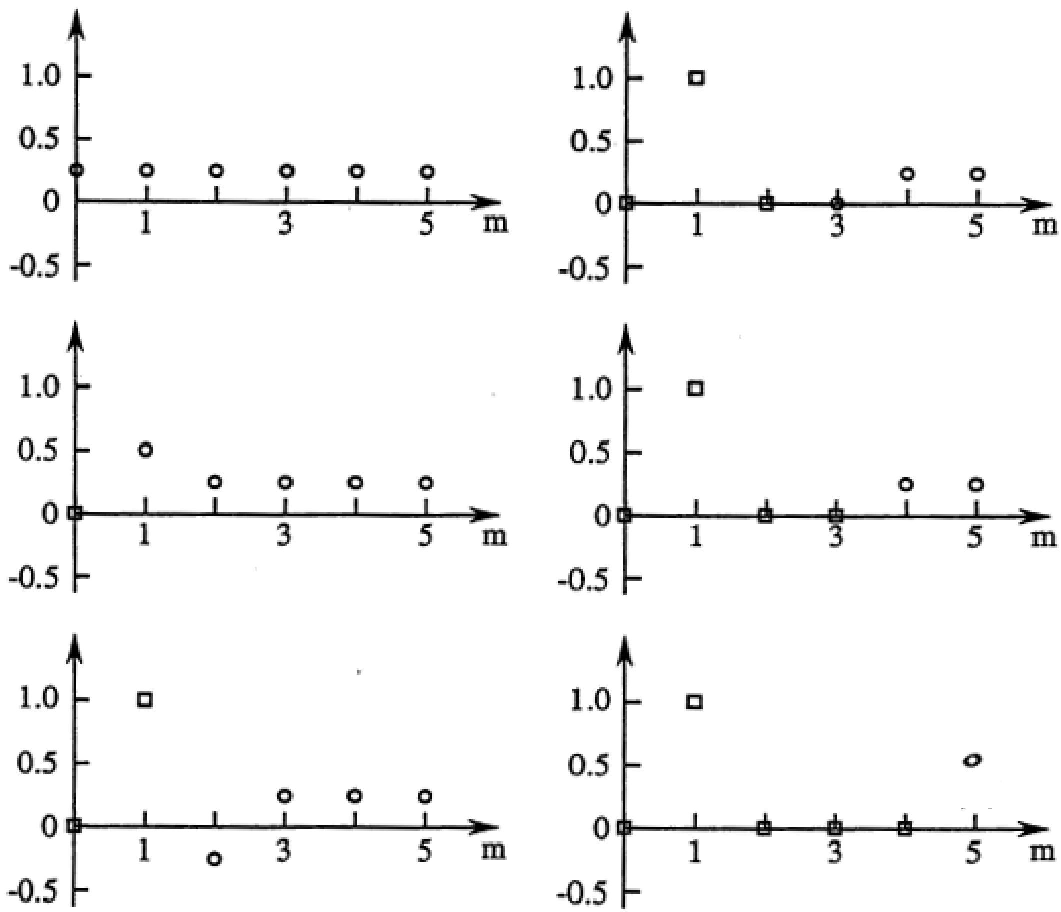
$$(m+k,n+l) \in \{\text{pixels not yet binarized}\}$$

2-D Error Diffusion Weighting Filters

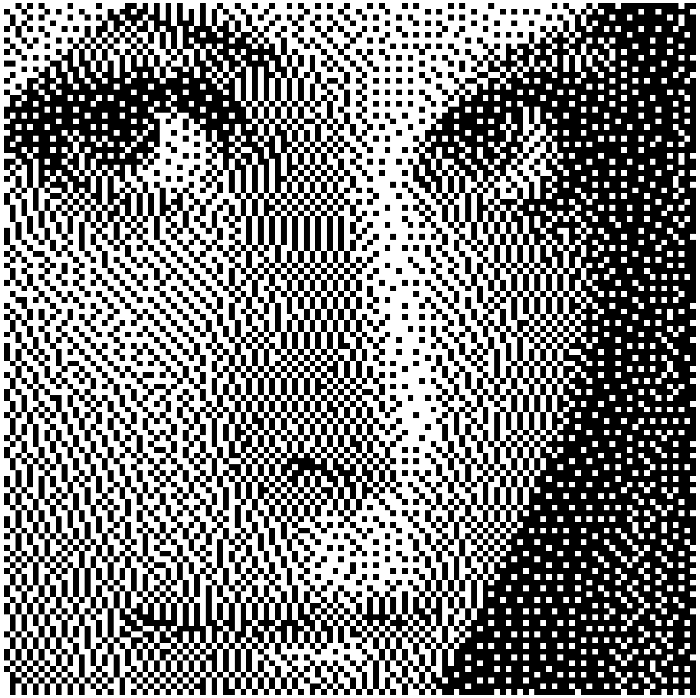


1-D Example

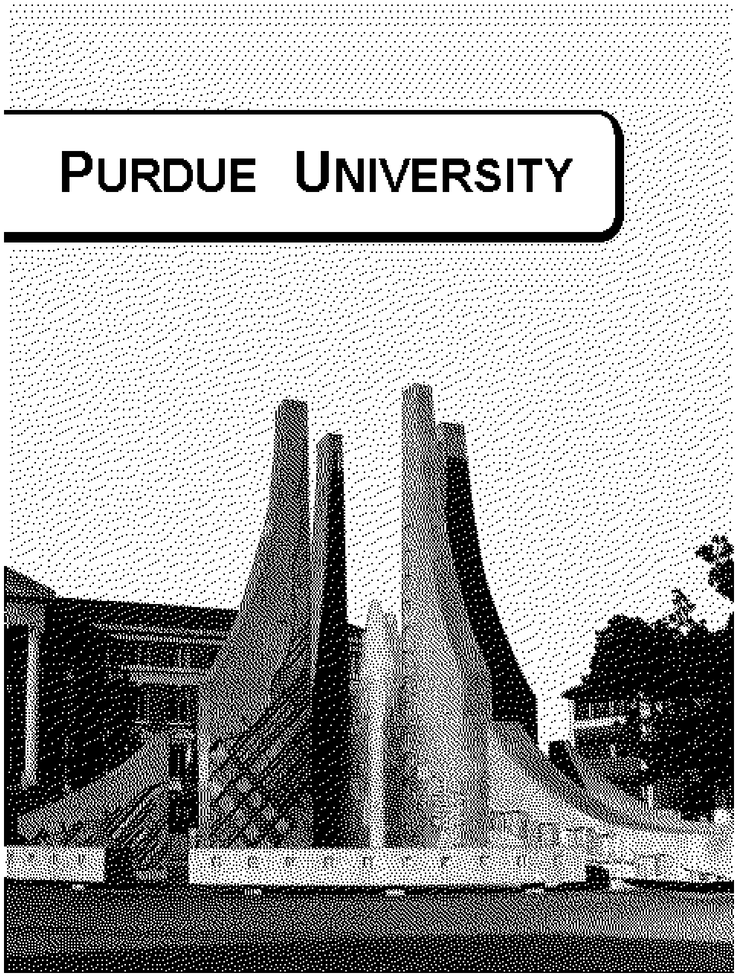
$f[m] \equiv 0.25$ $\tilde{f}[m]$ – ○ $g[m]$ – □

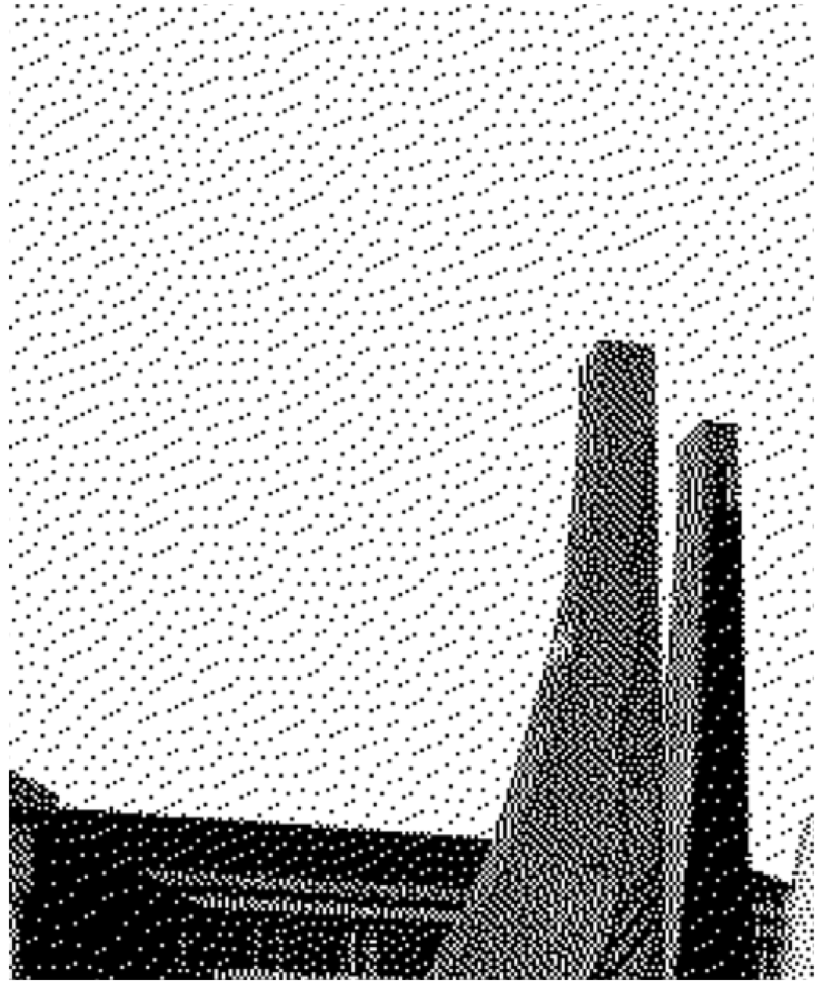


2-D Error Diffusion – Floyd-Steinberg Weights Texture Artifacts in Midtones



2-D Error Diffusion – Floyd-Steinberg Weights
Texture Artifacts in Highlights





Error Diffusion Characteristics

- At each step, error diffusion preserves local average over part of image that has been binarized and part that is yet to be binarized
- No fixed number of quantization levels
- Requires more computation than screening
- Excellent detail rendition (sharpens image)
- Generally good texture with some exceptions:
  - Texture contouring
  - Worm-like patterns in highlights and shadows
  - Texture cliques in midtones
  - Texture used to render a given gray level may be context-dependent

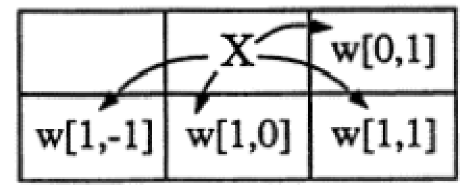
Two Views of Error Diffusion

- Diffuse error immediately after binarizing pixel to all pixels in neighborhood

$$g[m,n] = \begin{cases} 1, & \tilde{f}[m,n] \geq 0.5 \\ 0, & \text{else} \end{cases}$$

$$e_Q[m,n] = g[m,n] - \tilde{f}[m,n]$$

$$\tilde{f}[m+k,n+l] = \tilde{f}[m+k,n+l] - w[k,l]e_Q[m,n]$$

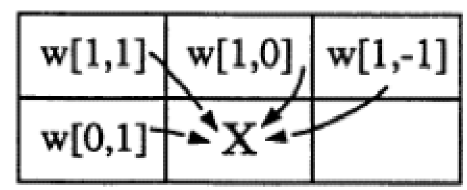


- Gather errors from neighboring pixels just prior to binarization

$$\tilde{f}[m,n] = f[m,n] - \sum_k \sum_l w[k,l]e_Q[m-k,n-l] \quad (1)$$

$$g[m,n] = \begin{cases} 1, & \tilde{f}[m,n] \geq 0.5 \\ 0, & \text{else} \end{cases} \quad (2)$$

$$e_Q[m,n] = g[m,n] - \tilde{f}[m,n] \quad (3)$$



Fourier Analysis

- Combining (1) and (3),

$$e_Q[m,n] = g[m,n] - f[m,n] + \sum_k \sum_l w[k,l]\, e_Q[m-k, n-l]$$

- In Fourier domain,

$$E_Q(\mu,\nu) = G(\mu,\nu) - F(\mu,\nu) + W(\mu,\nu) E_Q(\mu,\nu)$$

- Rearranging, we get

$$G(\mu,\nu) = F(\mu,\nu) + \bar{W}(\mu,\nu) E_Q(\mu,\nu) \quad (4)$$

where $\bar{W}(\mu,\nu) = 1 - W(\mu,\nu)$ is a high-pass filter

- This shows that the spectrum of the binary image consists of the spectrum of the continuous-tone image plus a high-pass filtered version of the spectrum of the quantization error.

Knox's Empirical Model for Quantization Error*

- We cannot find an analytical expression for $E_Q(\mu, \nu)$
- Knox proposed the following model

$$E_Q(\mu, \nu) = cF(\mu, \nu) + R(\mu, \nu) \quad (5)$$

Correlation coefficient – c

Residual – $R(\mu, \nu)$ (generally some image-dependence)

ED Weights	c
1-D	0.0
Floyd and Steinberg	0.55
Jarvis, Judice, and Ninke	0.80

***K. T. Knox, "Error Image in Error Diffusion," SPIE Vol. 1657 (1992)**

Application to Fourier Analysis

- Combining (4) and (5), we get

$$G(\mu,\nu) = F(\mu,\nu) + \bar{W}(\mu,\nu)\big(cF(\mu,\nu) + R(\mu,\nu)\big)$$
$$= \underbrace{\big(1 + c\bar{W}(\mu,\nu)\big)}_{\text{High-emphasis filter}} F(\mu,\nu) + \underbrace{\bar{W}(\mu,\nu)R(\mu,\nu)}_{\text{High-pass noise}}$$

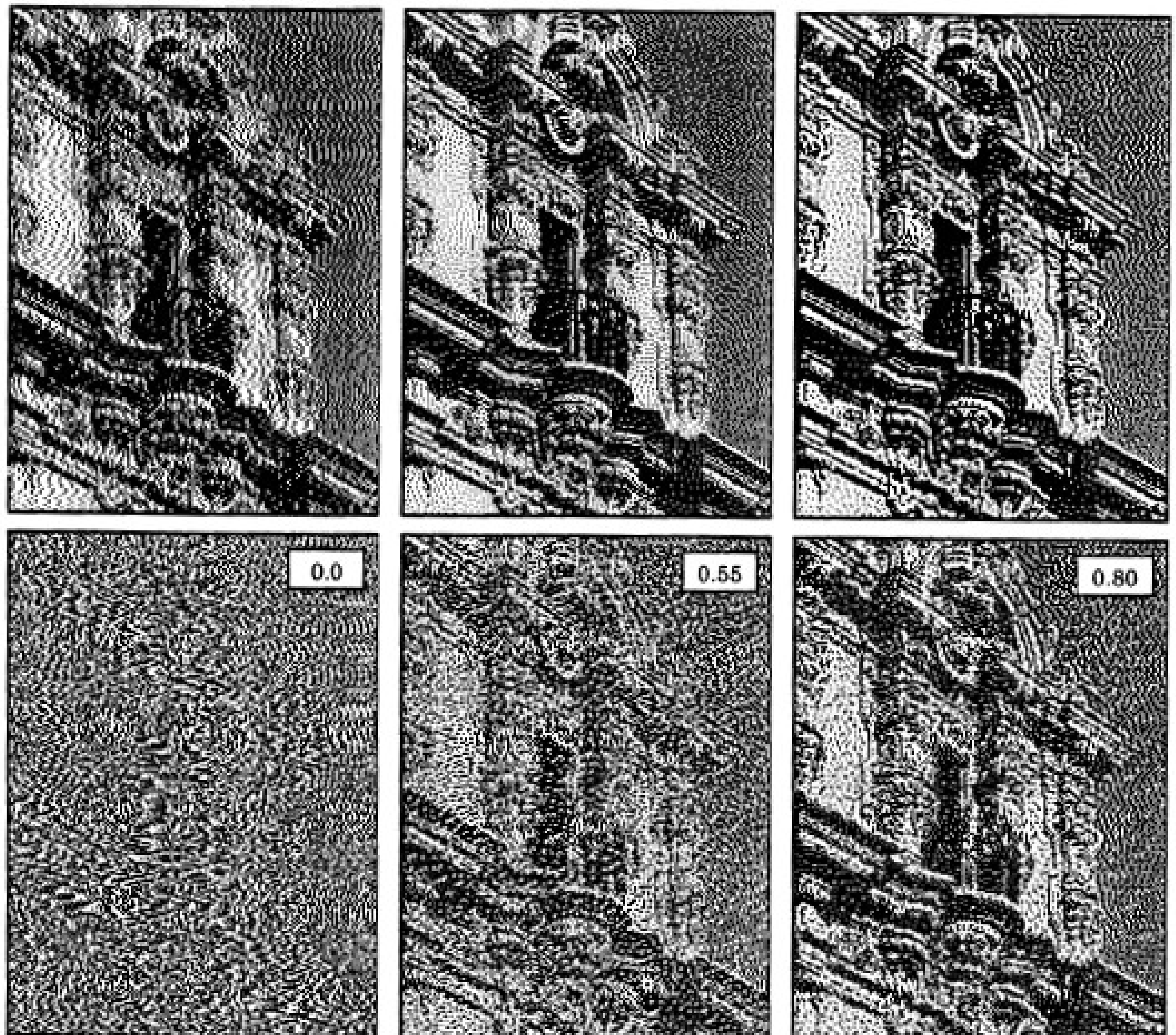
- The first term shows the origins of the sharpening effect that is characteristic of error diffusion.
- To the extent that the residual $R(\mu,\nu)$ is uncorrelated with the original image, the second term will look like blue noise.

Knox's Experimental Results

Halftone Image

Quantized Error Image

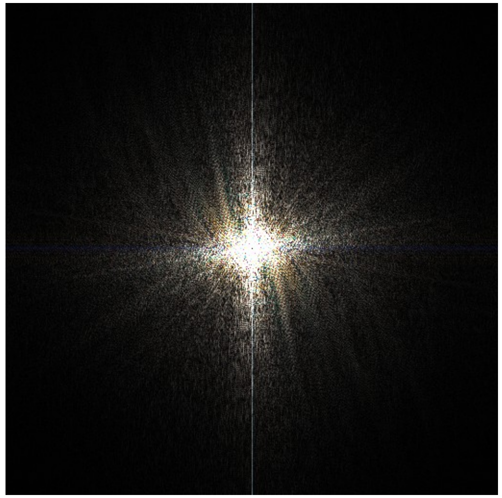
The parameter in the upper right-hand corner is the correlation coefficient c

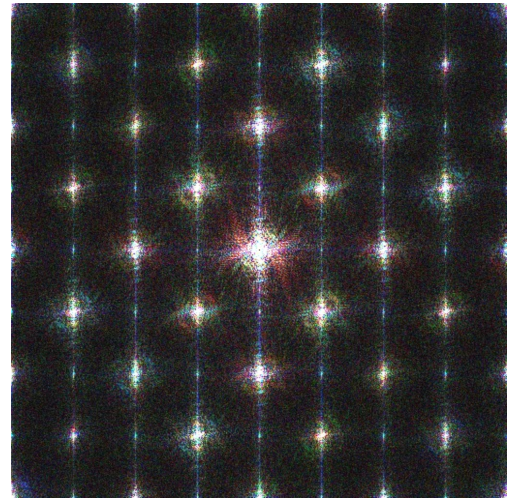


1-D **Floyd-Steinberg** **Jarvis, Judice, Ninke**

Spectral Characteristics of Error Diffusion

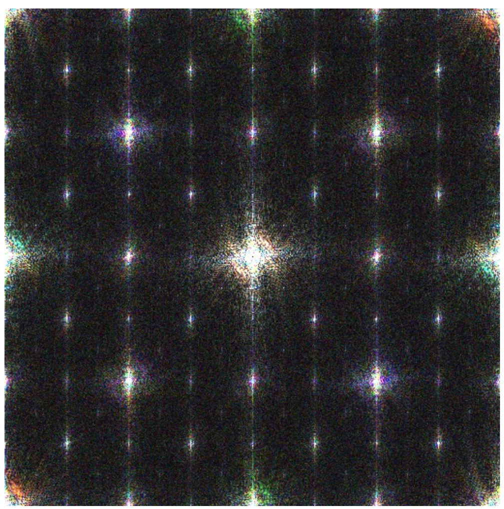
continuous tone

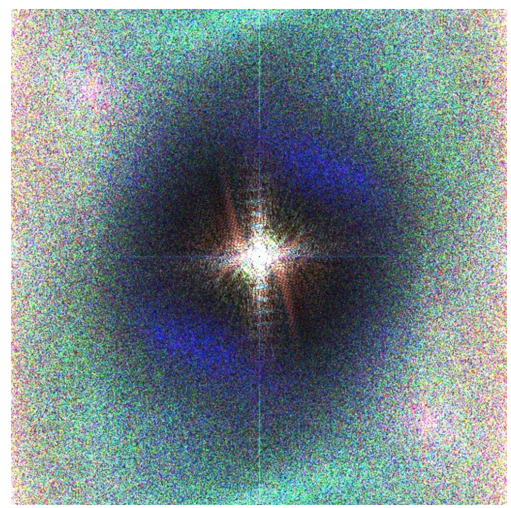




clustered dot dither

Bayer dither





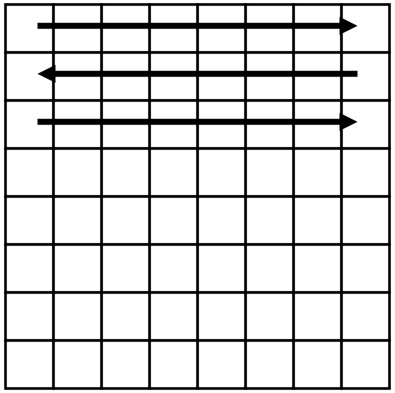
error diffusion

Synopsis

- Error diffusion architecture
- Error diffusion textures
- Edge enhancement effect of error diffusion
- Spectral analysis of error diffusion
- Variations on a theme
- Tone-dependent error diffusion

Variations on a Theme

- Non-lexicographic scan raster
  - Serpentine raster
  - Peano scan
  - Block-based scans
- Modifications to weights
  - Randomized weights
  - Adaptive weights
  - Tone-dependent weights
- Threshold modulation
  - Control sharpening
  - Encourage clustering



Serpentine Raster

Early Work

- Knox (1993) developed a spectral analysis for serpentine raster, and showed that it eliminates the asymmetry in the error weighting frequency response $W(\mu, \nu)$ that is due to the causal nature of the filter.
- Ulichney (1986, 1987) experimented with many different combinations of scan rasters, weight sets, and randomization techniques, and concluded that a serpentine raster, Floyd-Steinberg coefficients, and 50% randomization yielded the best results.

	X	$\frac{7}{16}\left(1-0.5\omega_1\right)$
$\frac{3}{16}\left(1+0.5\omega_2\right)$	$\frac{5}{16}\left(1+0.5\omega_1\right)$	$\frac{1}{16}\left(1-0.5\omega_1\right)$

Threshold Modulation

- Basic equation for threshold modulation

$$g[m,n] = \begin{cases} 1, & \tilde{f}[m,n] \geq t[m,n] + 0.5 \\ 0, & \text{else} \end{cases}$$

- Image-independent threshold modulation $t[m,n] = s[m,n]$ where $s[m,n]$ is a periodic screen function – encourages periodic clustered-dot behavior (Billotet-Hoffman and Bryngdahl, 1983)
- Image-dependent threshold modulation $t[m,n] = \kappa f[m,n]$ where κ is a constant that controls degree of edge enhancement (Eschbach, 1991)

$\kappa = -1$ Approximately cancels edge enhancement intrinsic to standard error diffusion

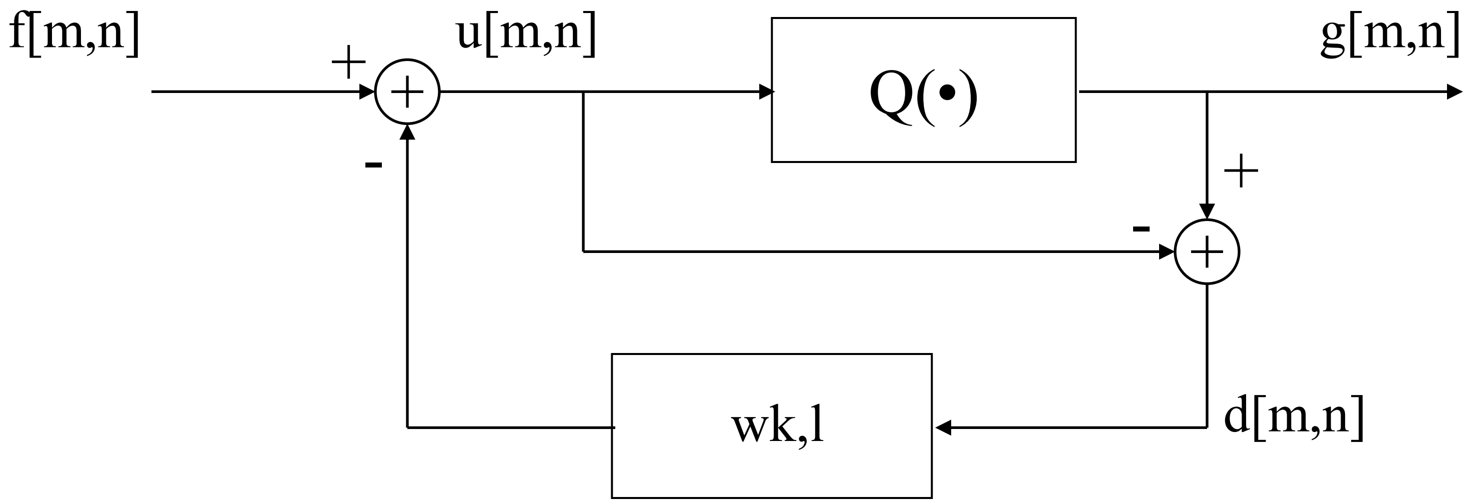
$\kappa = 0$ Yields standard error diffusion

$\kappa > 0$ Results in more edge enhancement than is provided by standard error diffusion

Tone-Dependent Error Diffusion Introduction

- Previous work related to TDED:
  - R. Eschbach, *JEI* 1993
  - J. Shu, *SID* 1996
  - V. Ostromoukhov, *US Patent* 1998
  - V. Ostromoukhov, *SIGGRAPH* 2001
- Our approach:
  - The weights and thresholds are optimized based on a HVS model.
  - Use variable weight locations to reduce worm like artifacts.

Conventional Error Diffusion



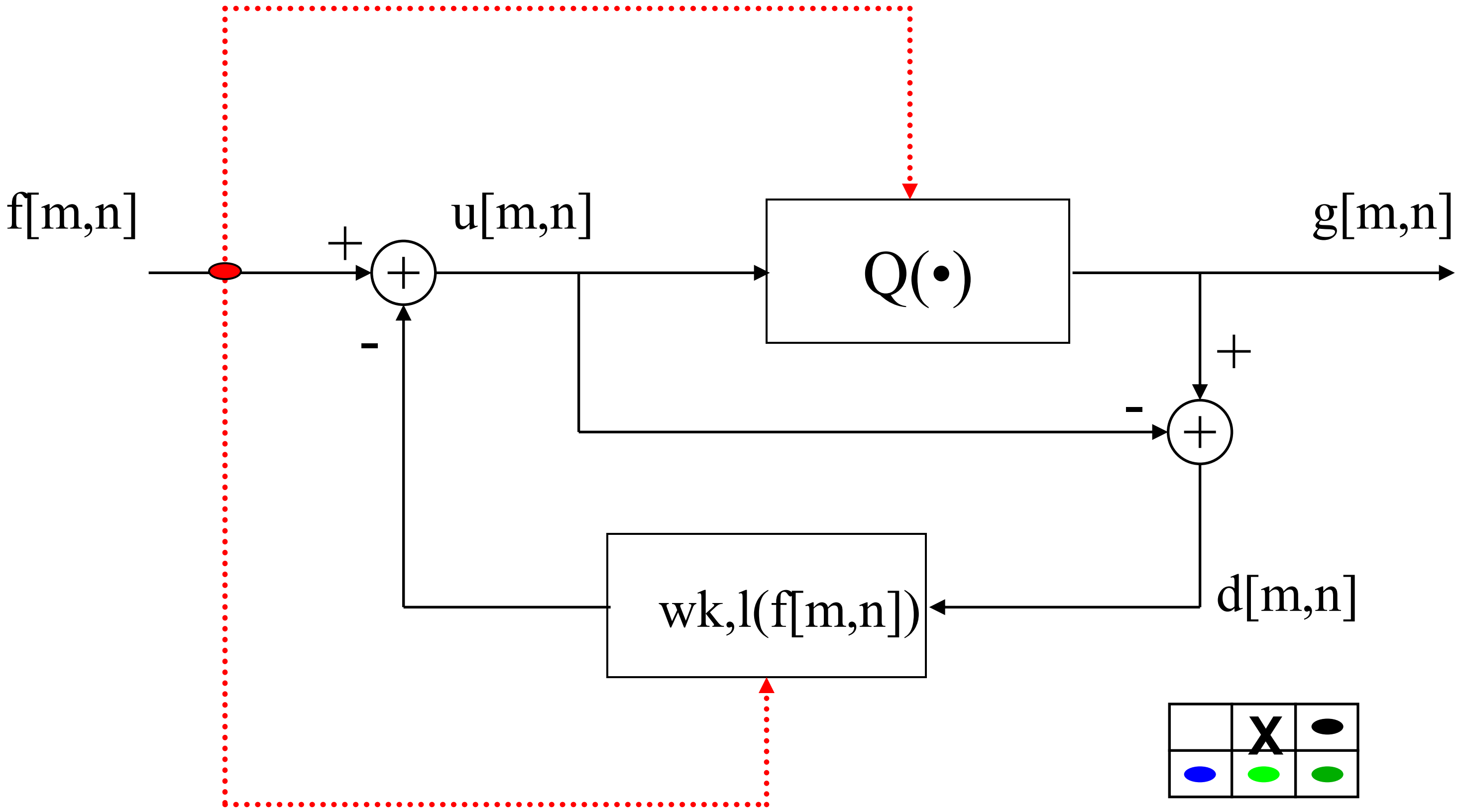
$$g[m,n] = \begin{cases} 1, & u[m,n] \geq t[m,n] \\ 0, & \text{else} \end{cases}$$

$$d[m,n] = g[m,n] - u[m,n]$$

$$u[m+k, n+l] \Box \; u[m+k, n+l] - w_{k,l} d[m,n]$$

Tone Dependent Error Diffusion [Li and Allebach, 2004]

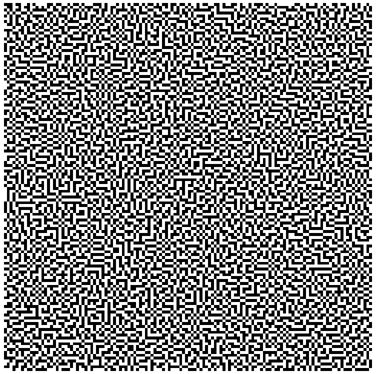
Weights and thresholds vary depending on the input



Tone Dependent Error Diffusion

Quantization process:

$$g[m,n] = \begin{cases} 1, & u[m,n] \geq t_U(f[m,n]) \\ 0, & u[m,n] < t_L(f[m,n]) \\ p[m,n;0.5] & \text{otherwise} \end{cases}$$



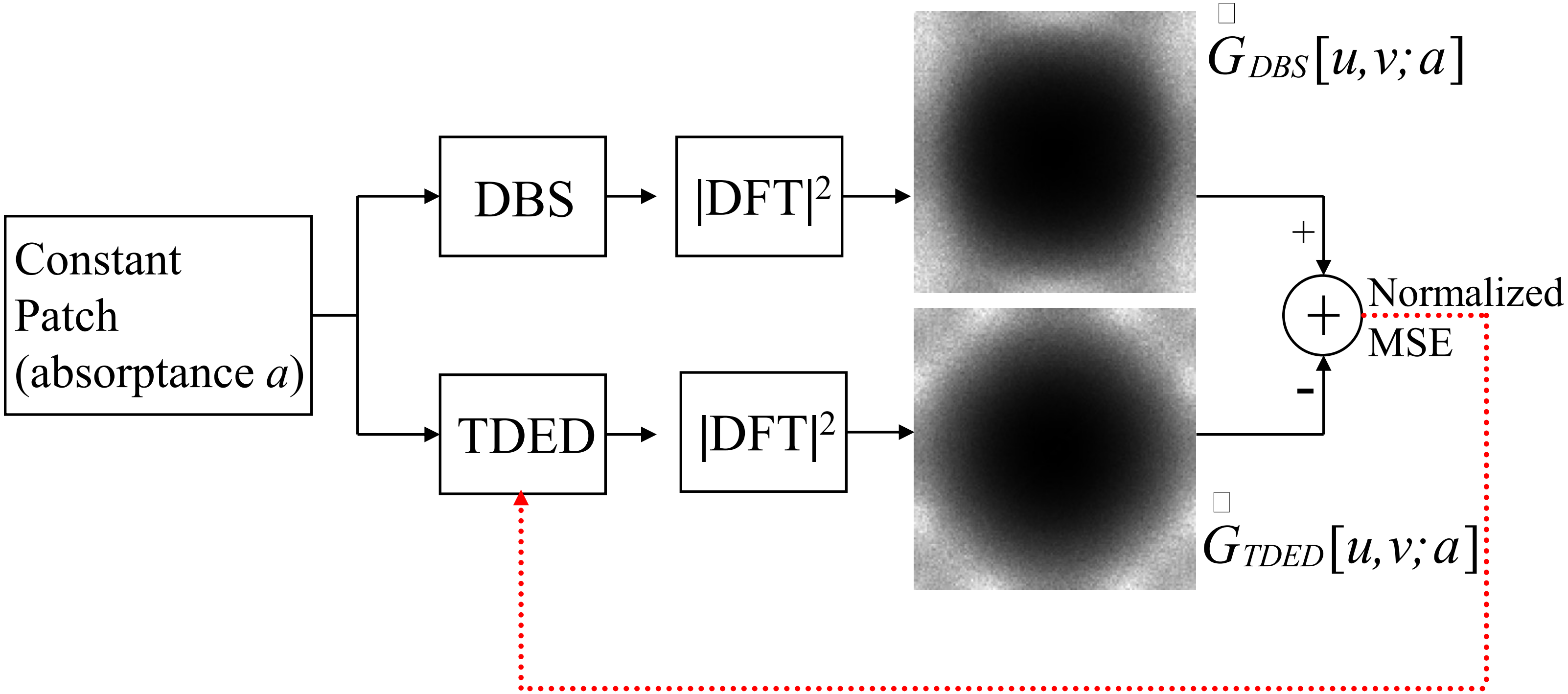
p[m,n;0.5]

Midtone halftone pattern generated with direct binary search (DBS)

Problem:

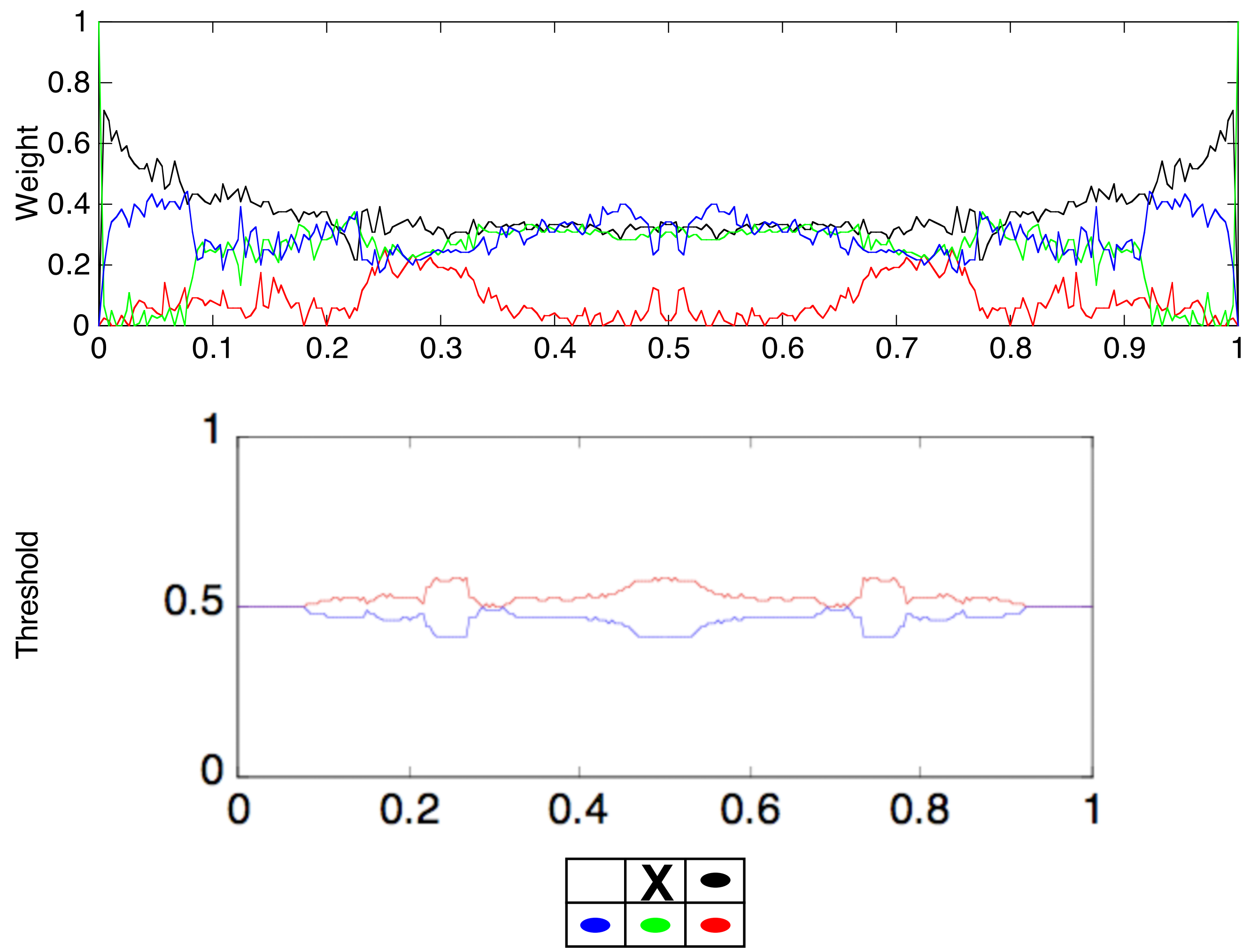
How to optimize tU, tL and wk,l for each gray level.

Optimization of TDED parameters

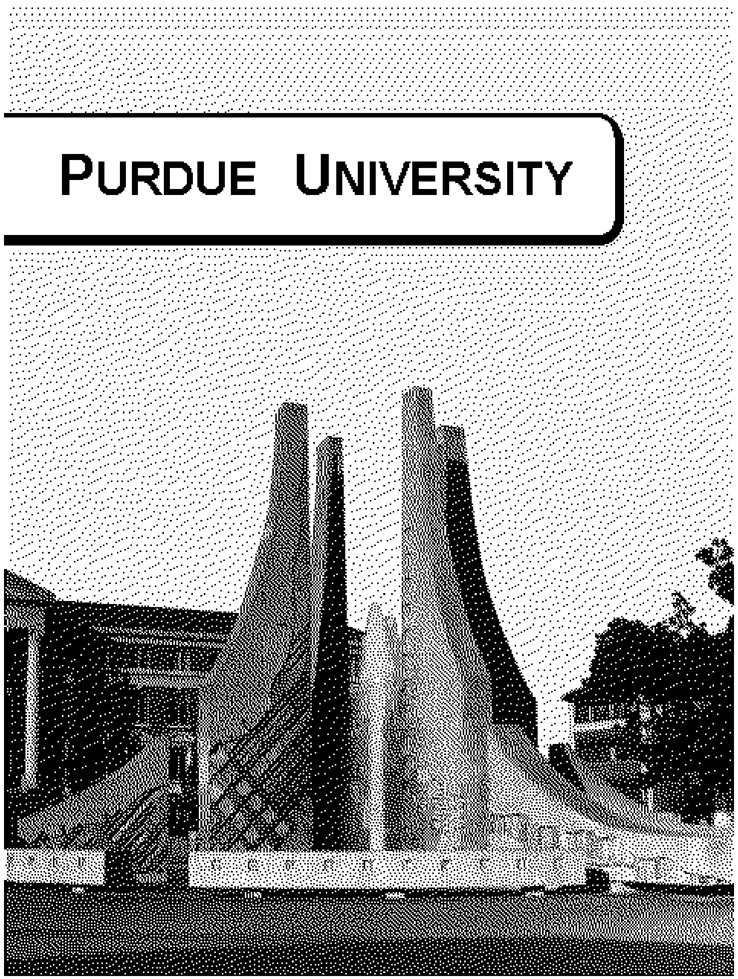


- Cost function $$\xi(a) = \sum_u \sum_v \frac{(\hat{G}_{DBS}[u,v;a] - \hat{G}_{TDED}[u,v;a])^2}{\hat{G}_{DBS}[u,v;a]^2}.$$

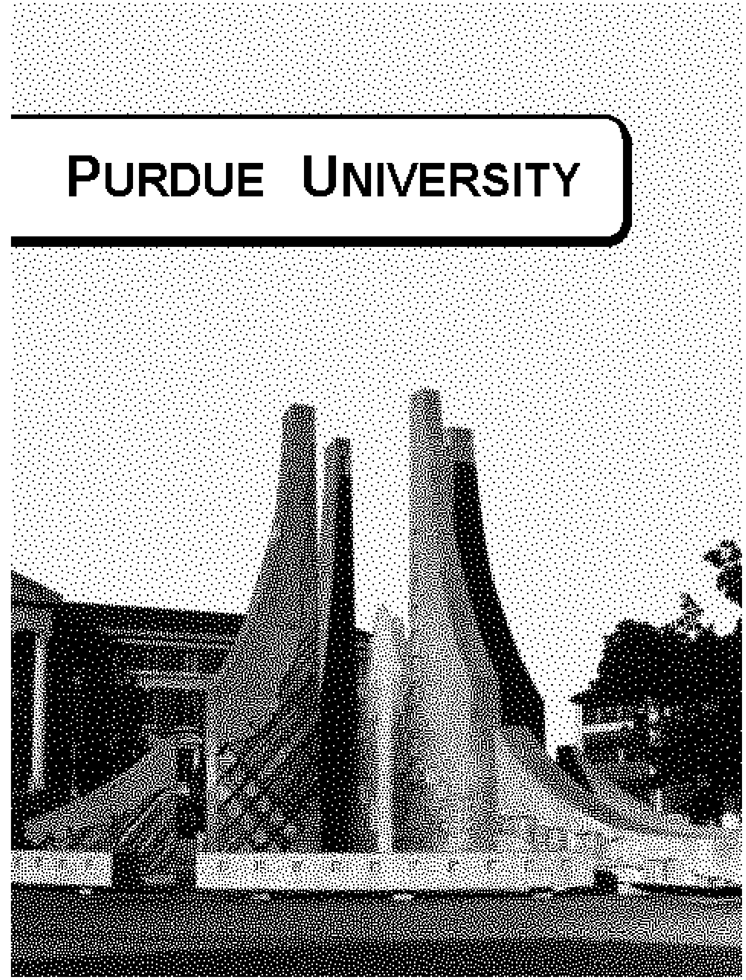
Optimal weights and thresholds



Floyd-Steinberg vs TDED

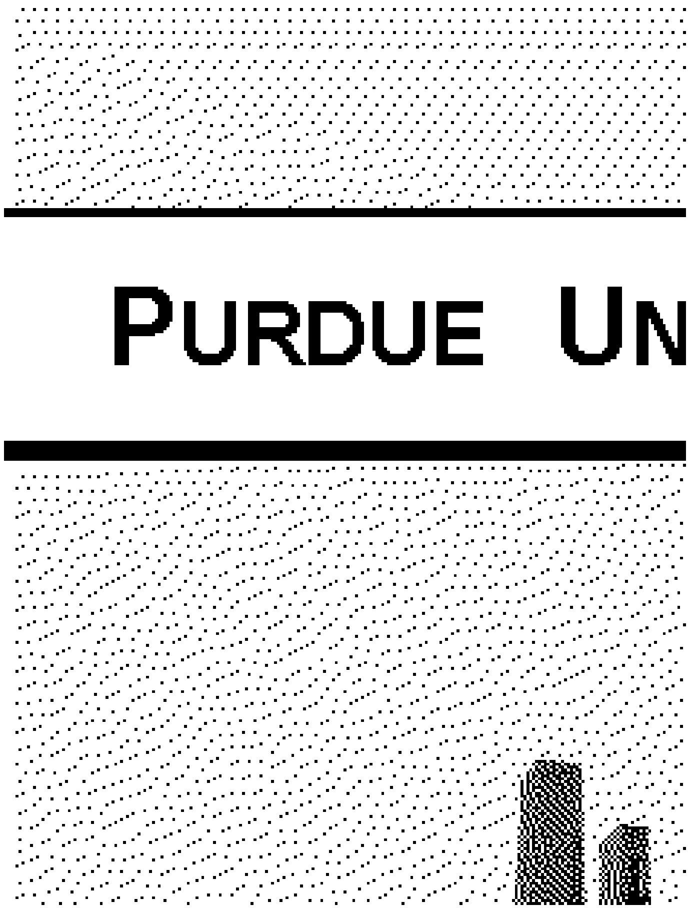


Floyd-Steinberg

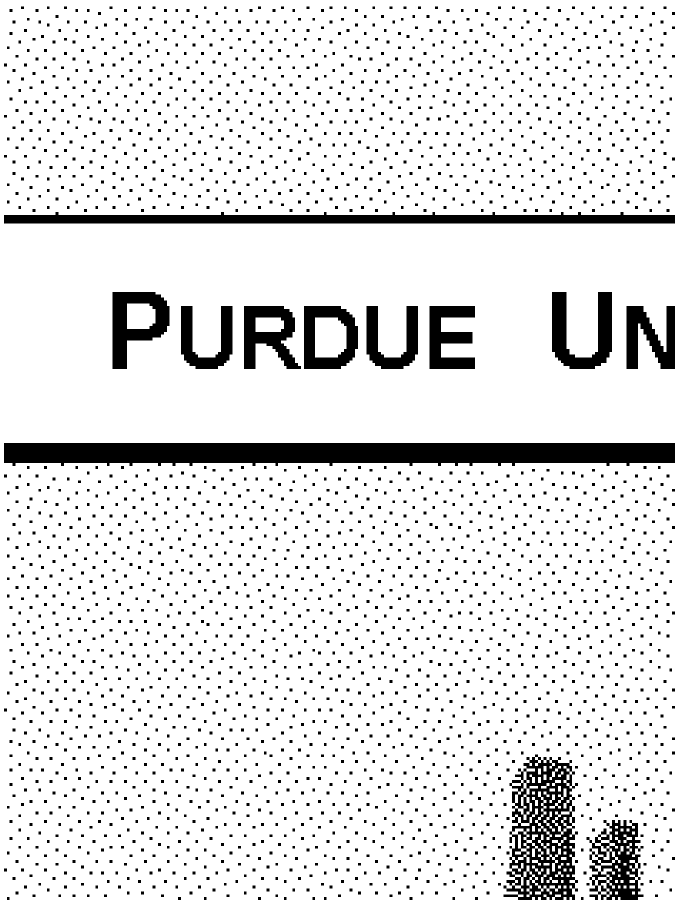


TDED

Floyd-Steinberg vs TDED

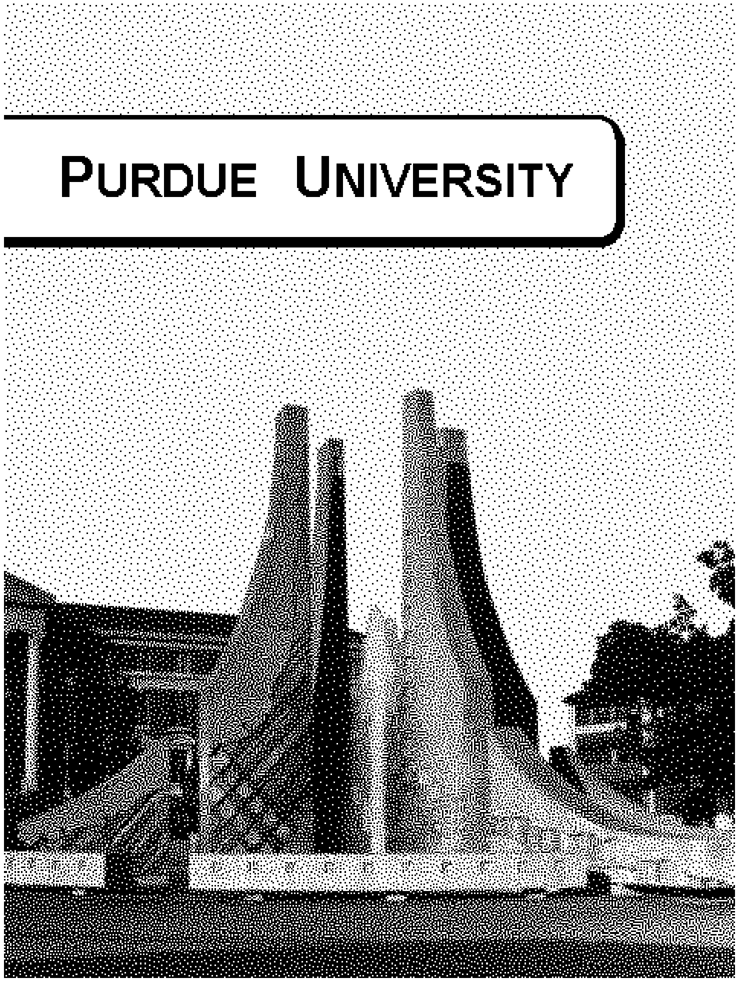


Floyd-Steinberg

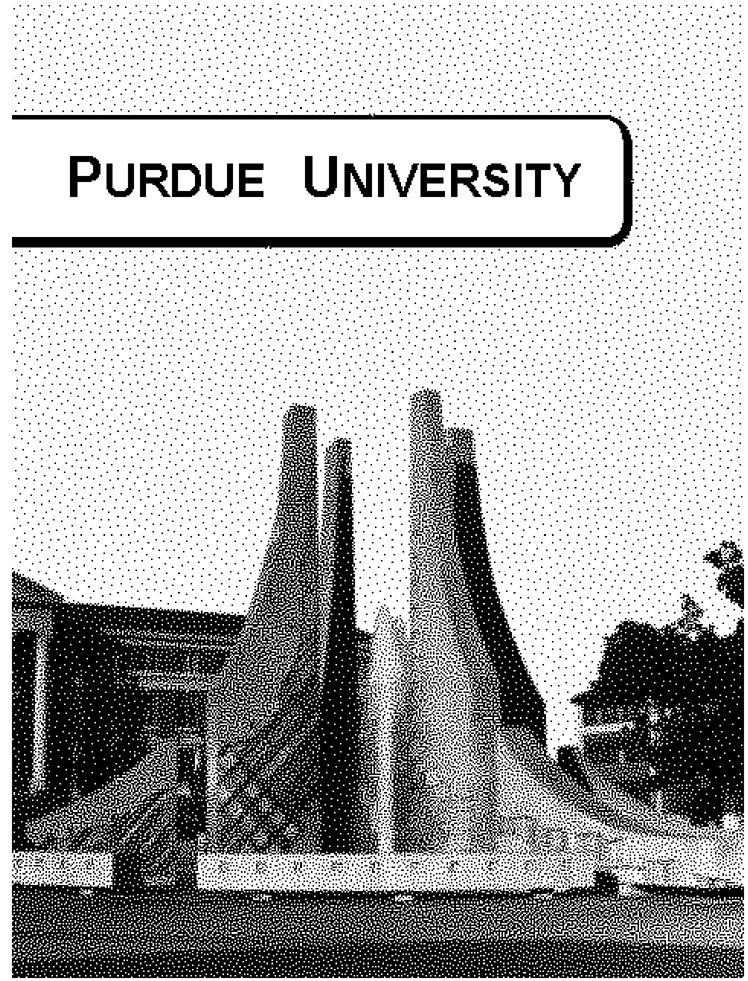


TDED

TDED vs DBS



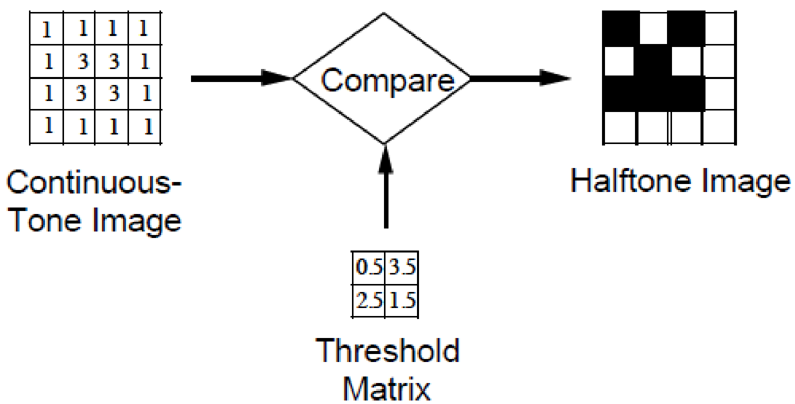
TDED



DBS

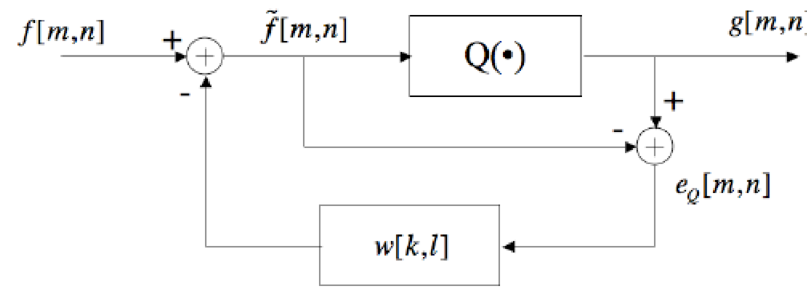
Halftoning Algorithms

Basic Structure of Screening Algorithm



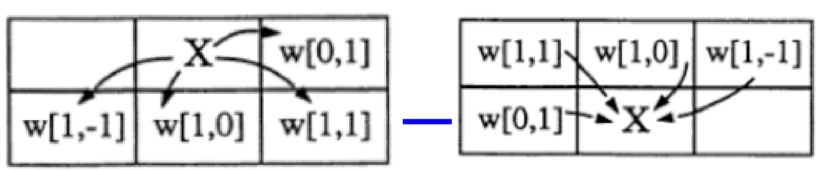
The threshold matrix is periodically tiled over the entire continuous-tone image.

Error Diffusion Architecture

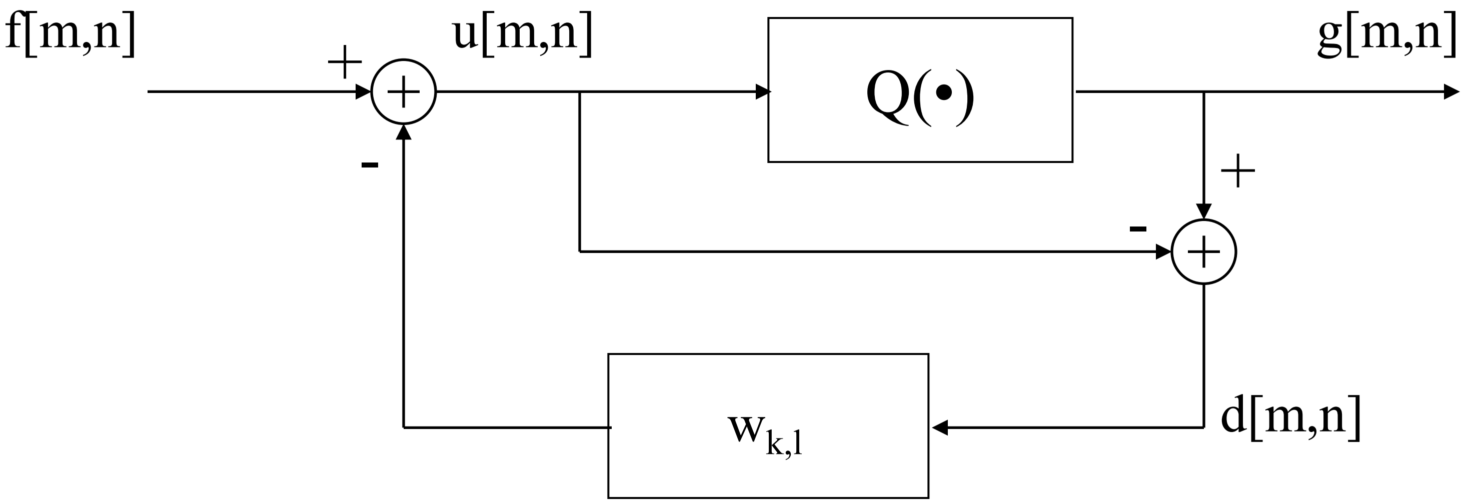


Definition of terms

- Continuous-tone input image – $f[m,n]$
- Modified (updated) continuous-tone image – $\tilde{f}[m,n]$
- Binary output halftone image – $g[m,n]$
- Display error – $e_D[m,n] = g[m,n] - f[m,n]$
- Quantization error – $e_Q[m,n] = g[m,n] - \tilde{f}[m,n]$



Error Diffusion

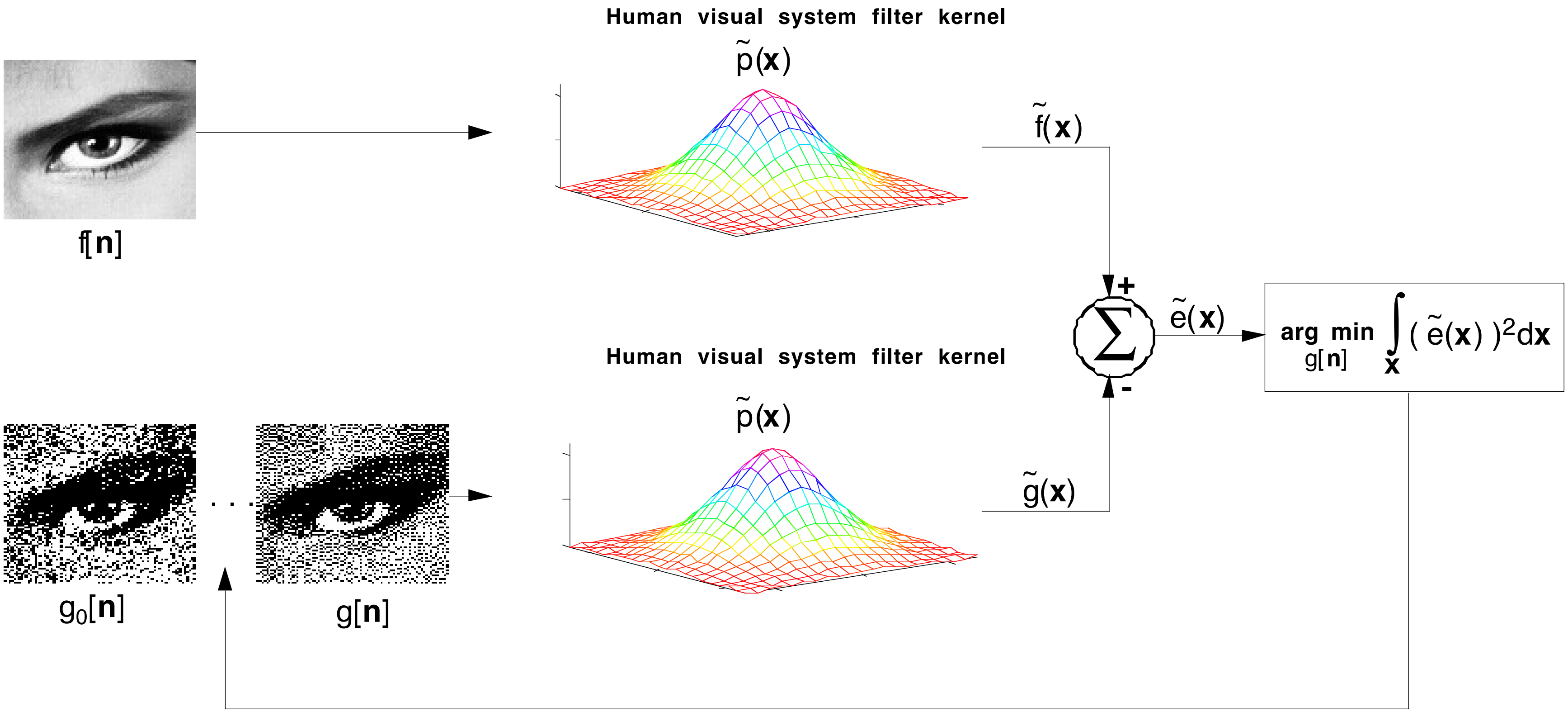


$$g[m,n] = \begin{cases} 1, & u[m,n] \geq t[m,n] \\ 0, & \text{else} \end{cases}$$

$$d[m,n] = g[m,n] - u[m,n]$$

$$u[m+k, n+l] \square \; u[m+k, n+l] - w_{k,l} d[m,n]$$

One Example of an Iterative Method: Direct Binary Search

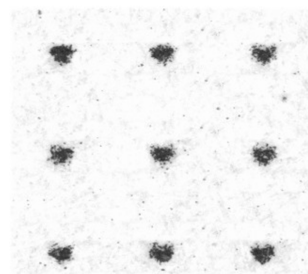


Tone Correction

- Our discussion so far has assumed an ideal printer :
  - non-overlapping, space-filling dots that have absorptance = 1
  - placed on a substrate with absorptance = 0
  - no optical scattering
- Real printers obey none of these assumptions

0	0	0
1	0	1
0	1	0

Bit map sent to 600 dpi laser printer

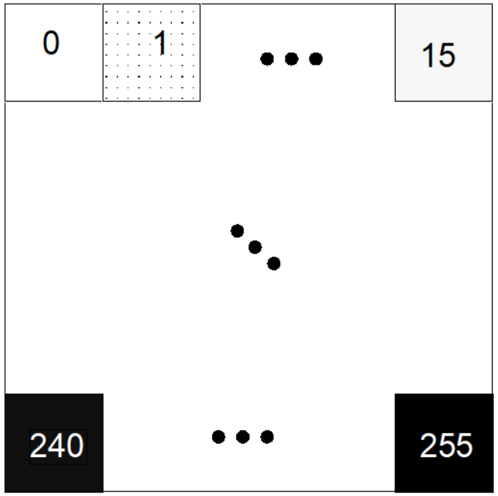


Print scanned at 4000 dpi

- To address these problems, we characterize the printer, then apply tone correction.

Printer Characterization

- Print a test target consisting of a grid of 16x16 constant tone patches, increasing in gray value from 0 to 255.
  - This implies that characterization depends on halftoning algorithm.



Test Target for Printer Characterization

- Measure printed patches using densitometer to obtain tone reproduction curve *TR(a).*

Gray Scale Conversion

- Density *D* is proportional to colorant mass deposited on substrate.
- Reflectance *r* is fraction of light reflected from paper.
- Absorptance *a* is fraction of light absorbed by paper.
- Relations between density, reflectance, and absorptance

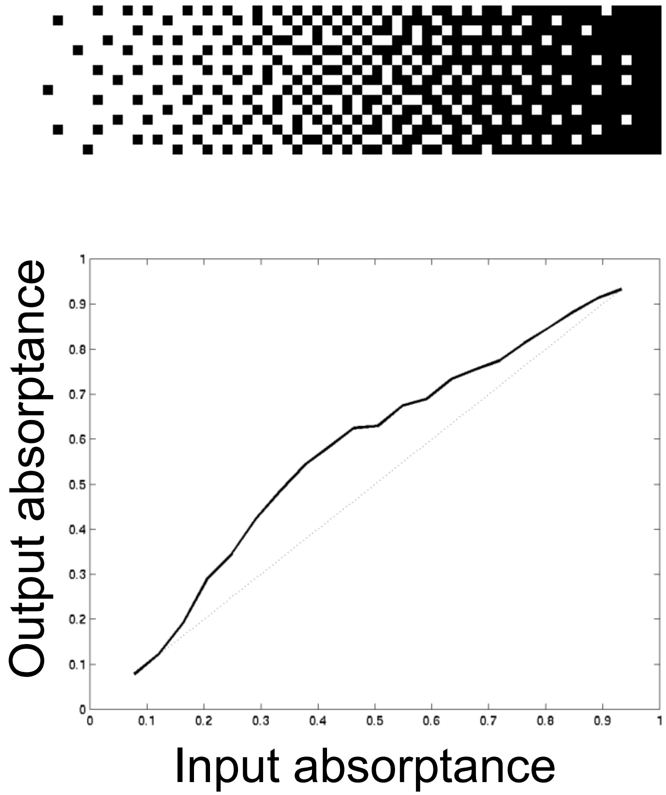
$$D = -\log_{10}(r), \quad r = 10^{-D}$$
$$a = 1 - r, \qquad r = 1 - a$$

- Examples:

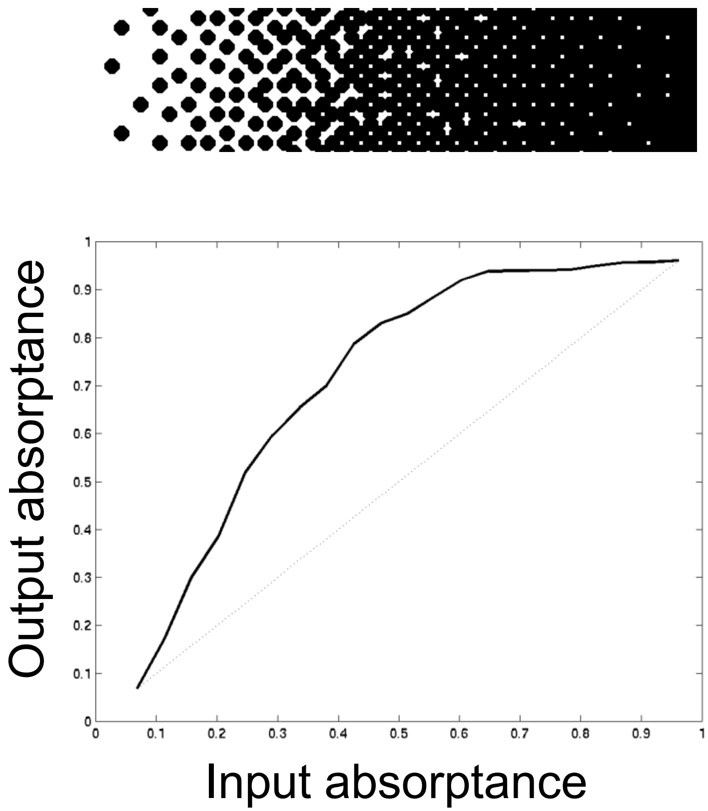
Highlight	Shadow
$a = 0.01$	$a = 0.99$
$r = 0.99$	$r = 0.01$
$D = 0.044$	$D = 2.0$

Tone Reproduction for Two Simulated Printers

Square dot printer



Circular dot printer

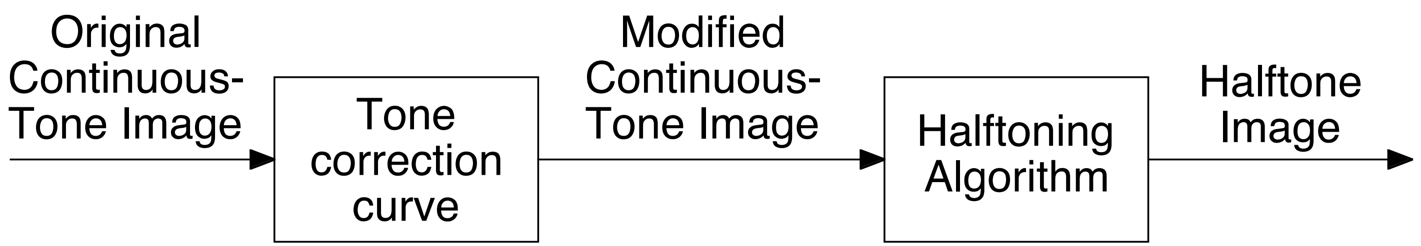


Tone Correction

- Tone correction curve is given by inverse of tone reproduction curve.

$$TC(a) = TR^{-1}(a)$$

- Process continuous-tone image pixel-by-pixel through tone correction curve prior to halftoning

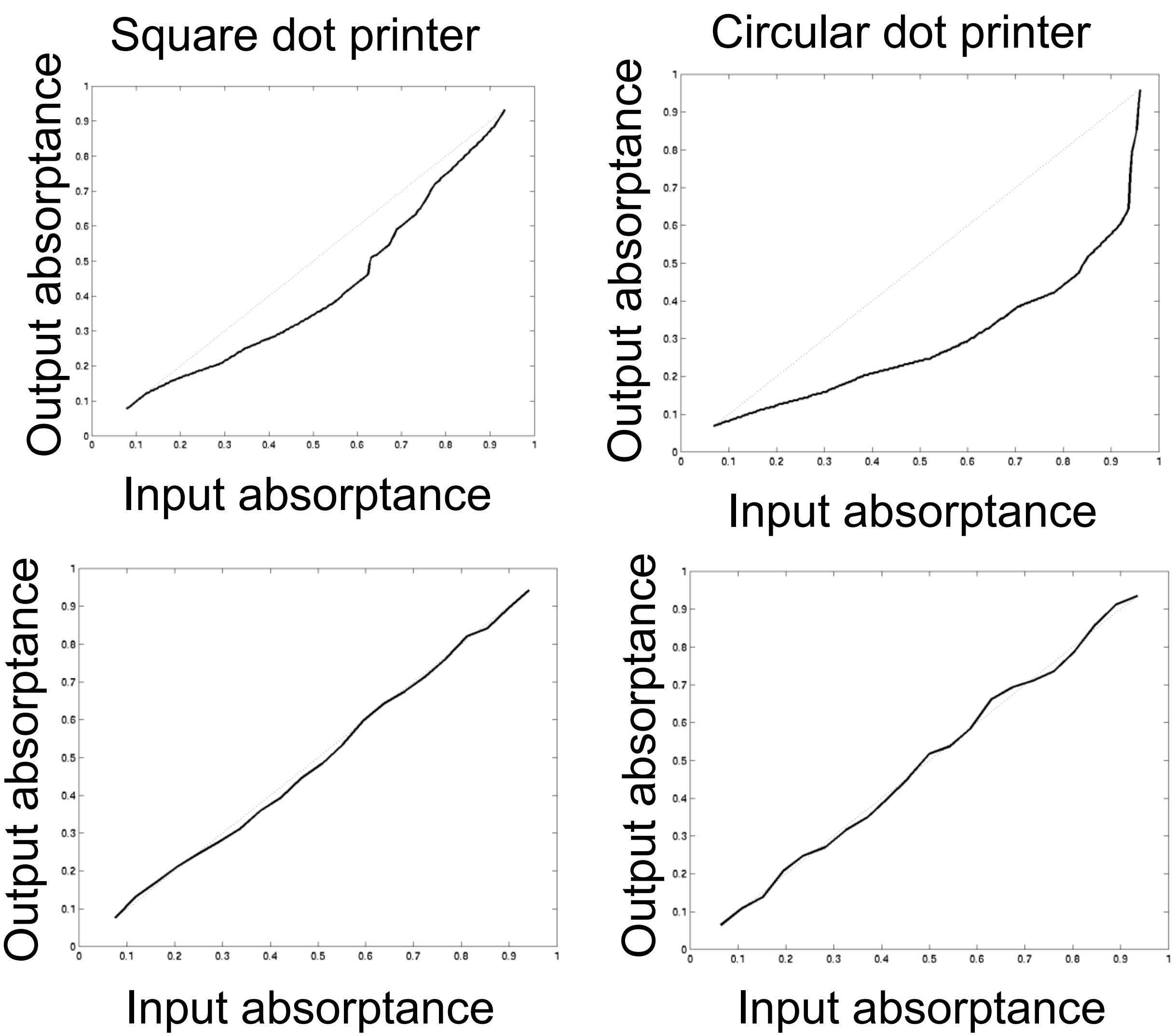


- With screening, can implement tone correction by modifying thresholds in the dither matrix.

Tone Correction for Two Simulated Printers

Tone Correction Curves

Tone Reproduction after Correction

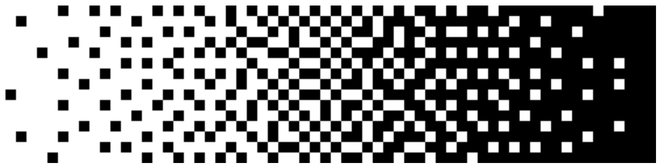


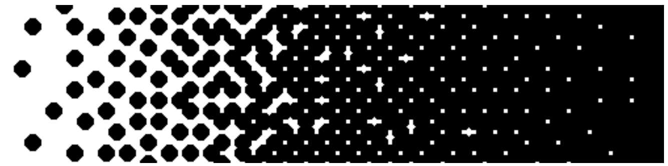
Results from Tone Correction for Two Simulated Printers

Square dot printer

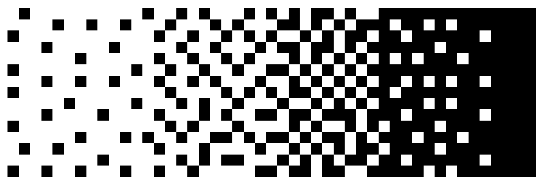
Circular dot printer

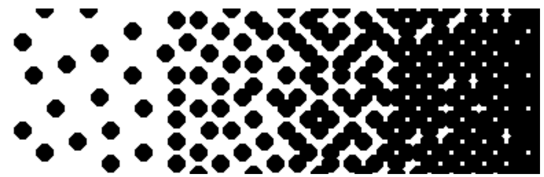
Before Correction





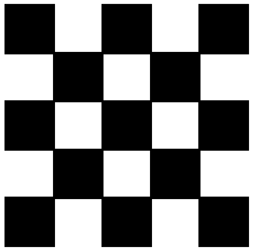
After Correction



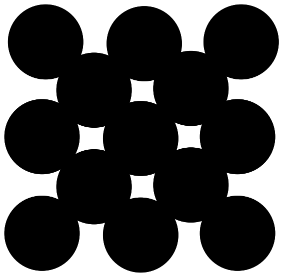


Limitations of Tone Correction

- Tone correction only assures correct average tone in the halftone image.
- It does not guarantee halftone textures that have low visibility and/or are stable.
- For example, the checkerboard shown below has absorptance 0.5 with the ideal printer and approximately 0.75 with the dot overlap printer.
- With tone correction, this texture would then be used to render an absorptance of 0.75, whereas it might be undesirable to use it at all due to instability of small holes.



Ideal Printer



Printer with Dot Overlap

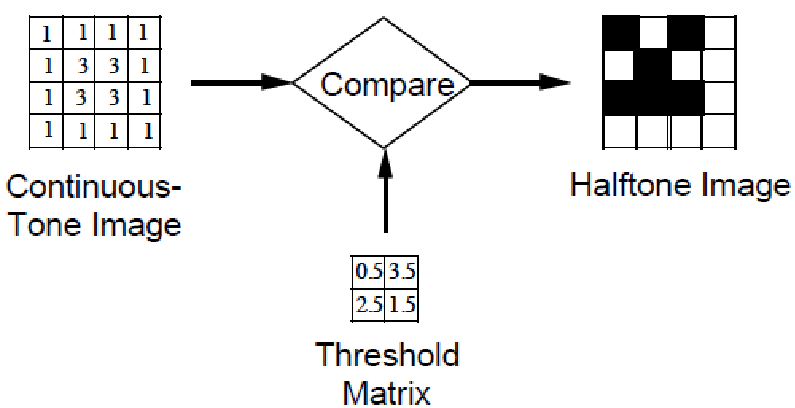
Limitations of Tone Correction (cont.)

- This problem may be overcome in one of two ways:
  - handpick the textures to be used by the halftoning algorithm,
  - Incorporate a model for the printer within the halftoning algorithm.

APPENDIX

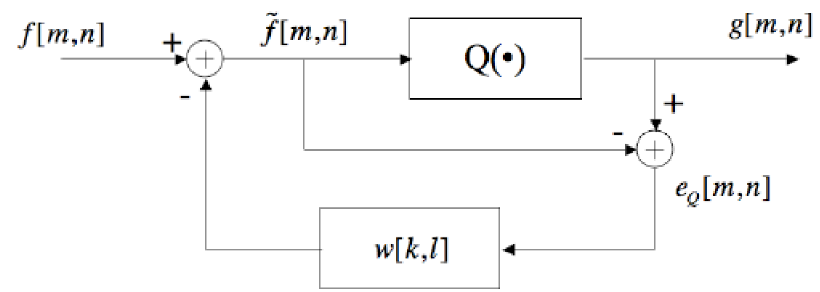
Halftoning Algorithms

Basic Structure of Screening Algorithm



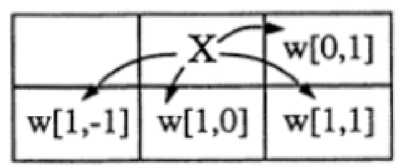
The threshold matrix is periodically tiled over the entire continuous-tone image.

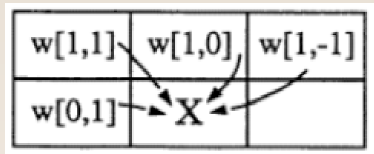
Error Diffusion Architecture



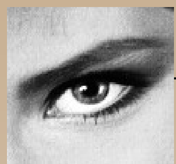
Definition of terms

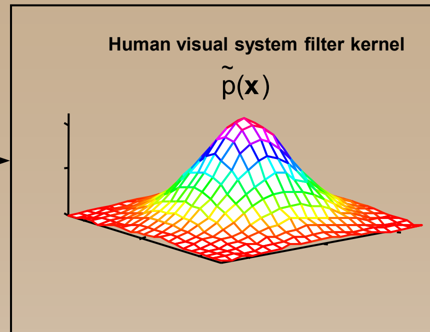
- Continuous-tone input image – $f[m,n]$
- Modified (updated) continuous-tone image – $\tilde{f}[m,n]$
- Binary output halftone image – $g[m,n]$
- Display error – $e_D[m,n] = g[m,n] - f[m,n]$
- Quantization error – $e_Q[m,n] = g[m,n] - \tilde{f}[m,n]$



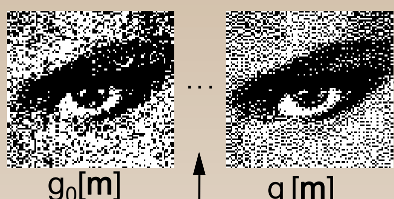


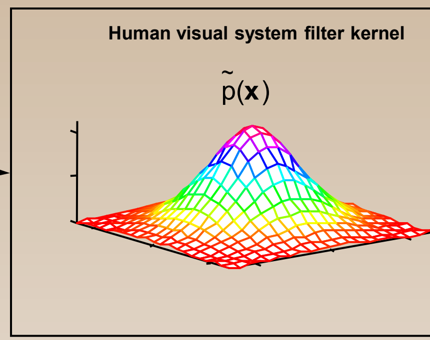
Conventional DBS Cost Metric





$\tilde{f}(\mathbf{x})$ -





$\tilde{g}(\mathbf{x})$ +

$\sum$ $\tilde{e}(\mathbf{x})$

$$\underset{g[\mathbf{m}]}{\arg\min} \int \left(\tilde{e}(x,y)\right)^2 dxdy$$

Direct binary search (DBS)

: Iterative search halftoning algorithm to minimize the mean squared perceptually filtered error image.

$\mathbf{f}[\mathbf{m}]$: *Continuous tone grayscale image*

$$\mathbf{g}[\mathbf{m}]: \textit{Halftone image} = \begin{cases} 1 & (\textit{black / dot}) \\ 0 & (\textit{white / hole}) \end{cases}$$

$e[\mathbf{m}] = g[\mathbf{m}] - f[\mathbf{m}]$: *Error image*

$c_{pp}[\mathbf{m}] =$ *Autocorrelation of HVS filter*

$$c_{pe}[\mathbf{m}] \triangleq \sum_{\mathbf{n}} e[\mathbf{n}]\, c_{\tilde{p}\tilde{p}}[\mathbf{m}-\mathbf{n}]$$

Standard DBS Error metric:

$$\phi \triangleq \int \left|\tilde{e}(x,y)\right|^2 \mathrm{d}x\mathrm{d}y$$

$$= \sum_{m,n,k,l} e[m,n]\,e[k,l]\; c_{\tilde{p}\tilde{p}}[m-k, n-l]$$

$$= \sum_{\mathbf{m}} \left(e[\mathbf{m}]\, c_{\tilde{p}\tilde{e}}[\mathbf{m}]\right)$$

Standard DBS Equations

$$\phi = \sum_{k,l,m,n} \left(e[m,n]e[k,l]c_{pp}[m-k,n-l] \right) = \sum_{\vec{m}} \left(e[\vec{m}]c_{pe}[\vec{m}] \right) = \sum_{m} \left(e[m]c_{pe}[m] \right)$$

$$c_{pe}[m] = \sum_{n} e[n]c_{pp}[n-m]$$

Update equations:

$$g'[m] = g[m] + a_o\delta[m-m_o] \qquad e'[m] = e[m] + a_o\delta[m-m_o]$$

$$c'_{pe}[m] = c_{pe}[m] + a_o c_{pp}[m-m_o]$$

Change in Error :

$$\Delta\phi = \int \left|\bar{e}'(x)\right|^2 \mathrm{dx} - \int \left|\bar{e}(x)\right|^2 \mathrm{dx} = \sum_{m} \left(e'[m]c'_{pe}[m] \right) - \sum_{m} \left(e[m]c_{pe}[m] \right)$$

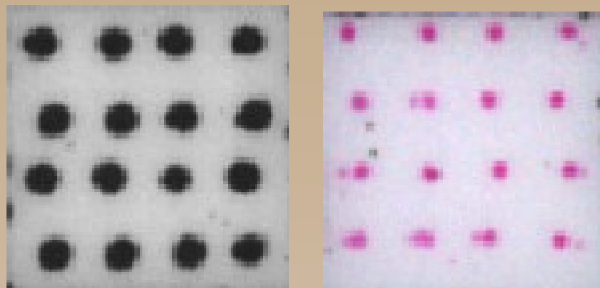
$$= \sum_{m} \left(\left(e[m] + a_o\delta[m-m_o] \right) \left(c_{pe}[m] + a_o c_{pp}[m-m_o] \right) \right) - \sum_{m} \left(e[m]c_{pe}[m] \right)$$

$$= \mathrm{a}_0^2 c_{pp}[0] + 2a_o c_{pe}[m_0]$$

$$\Delta\phi = \sum_{m} \left(e'[m]c'_{pe}[m] \right) - \sum_{m} \left(e[m]c_{pe}[m] \right) = \mathrm{a}_0^2 c_{pp}[0] + 2a_o c_{pe}[m_0] \qquad [\oplus]$$

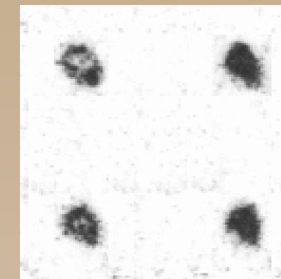
This is the change (deltaEmin) ,we try to minimize.We make the updates (in g[m], e[m], Cpe[m]) corr. to which ΔΦ is minimum

Printer Modeling Issues



Inkjet

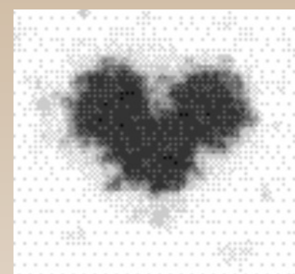
0	1	0
1	1	1
0	0	1



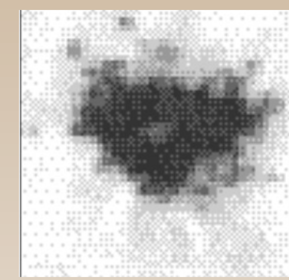
Electro-photographic (laser) prints

0	0	0	0	0
0	1	0	1	0
0	0	1	0	0
0	0	0	0	0

Bit Map



Printed at 300 dpi



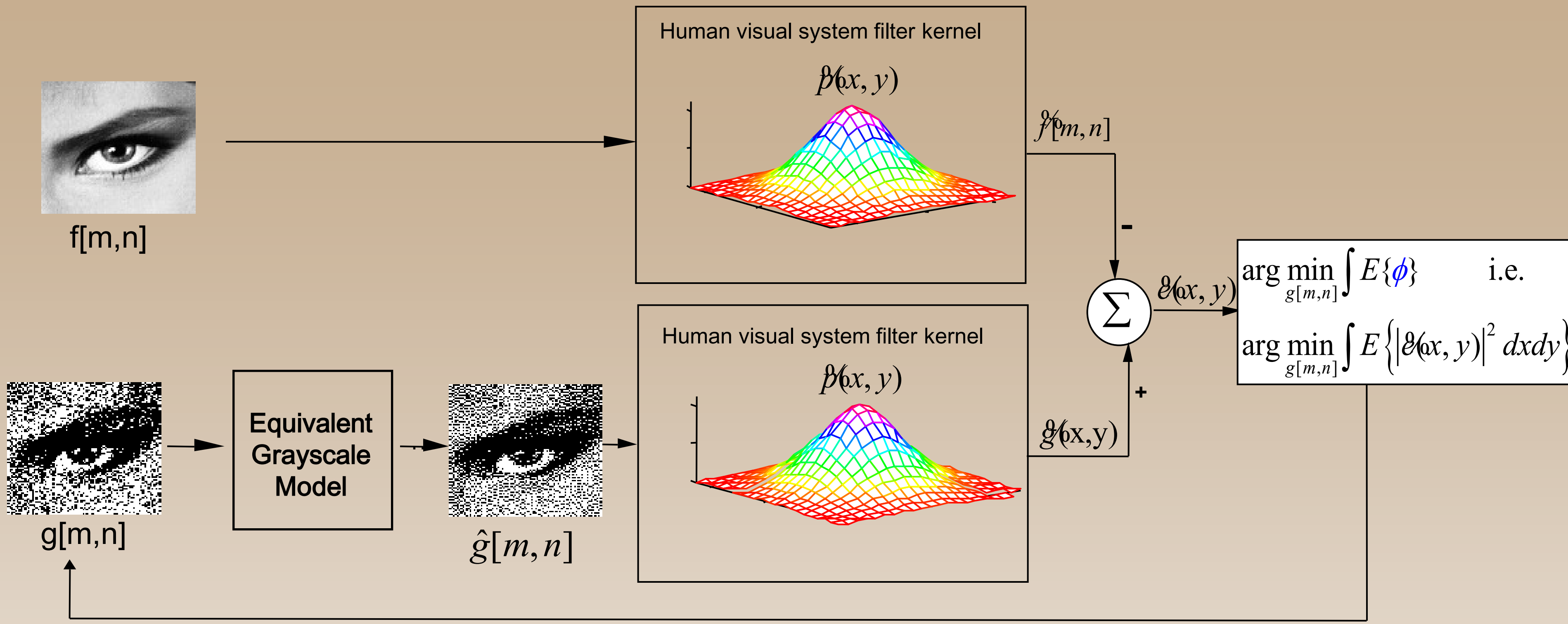
Printed at 600 dpi (2x mag.)

*

- Significant interaction among dots, particularly at higher resolutions
- Repeated occurrences of same local bit pattern will be rendered differently
- Linear, deterministic models are not adequate, particularly at higher resolutions

* F. A. Baqai and J. P. Allebach, "Computer-Aided Design of Clustered-Dot Color Screens Based on a Human Visual System Model"

Printer Model Based DBS Cost Metric



$$Error:\ \phi_g \triangleq E\{\phi\}$$

$$= \sum_{m,n,k,l} \bar{e}[m,n]\bar{e}[k,l]\, c_{\tilde{p}\tilde{p}}[m-k,n-l] \;+\; c_{\tilde{p}\tilde{p}}[0,0]\sum_{m,n}\sigma^2_{\hat{g}}[m,n]$$

where $\bar{e}[m,n] = E\{e[m,n]\} = E\{\hat{g}[m,n]\} - f[m,n]$

***E* {*Φ*} ⇒ Taking Expectation of error over the statistical model.**

Prior Work

- Physics based models
  - Analysis of optical dot gain (Yule-Nielsen effect) (Ruckdeschel and Hauser, 1978)
  - Geometric model to examine the dependence of halftone banding on imaging parameters (Loce, Lama and Maltz, 1995)
  - A model for EP dot writing process embedded in DBS (Kacker et al., 2002)
- Hard circular dot (HCD) overlap model
  - Early use by Roetling and Holladay,1979
  - Error diffusion extended to incorporate HCD model (Stucki,1981)
  - HCD model used to compensate neighborhood dot interaction (Stevenson et al. 1985)
  - Popularized and extended to tabular approach and color (Pappas et al. 1991, 1997)
  - DBS embedded with HCD model (Baqai and Allebach, 2003)
- Stochastic models
  - Micro-grid-based random toner particle models (Lin et al. 1993; Flohr et al. 1993)
  - Density-based random toner particle models (Flohr, Allebach, and Bouman, 1994)
  - Measurement of printer parameters for least squares model-based halftoning (Pappas et al. 1993,1999)
  - DBS embedded with measurement-based stochastic model (Baqai and Allebach, 2003)

Notations and Equivalent Grayscale Image

$c_{pp}[\mathbf{m}] = \textit{Autocorrelation of HVS filter}$

$e[\mathbf{m}] = g[\mathbf{m}] - f[\mathbf{m}]: \textit{Error image}$

$\mathrm{f}[\mathbf{m}]: \textit{Continuous tone grayscale image}$

$$g[\mathbf{m}]: \textit{Halftone image} = \begin{cases} 1 & (\textit{black / dot}) \\ 0 & (\textit{white / hole}) \end{cases}$$

- Image rendered by printer

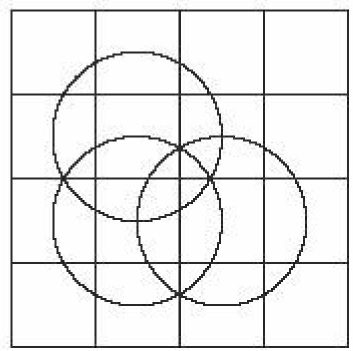
$$\hat{g}(x,y) = \sum_m \sum_n \hat{g}[m,n]\, p(x - mX, y - nX)$$

where p(x,y) is printer spot profile function and X is the distance between addressable pixels

- Equivalent grayscale image summarizes the effect of dot overlap within each device-addressable cell

0	0	0	0
0	1	0	0
0	1	1	0
0	0	0	0

Digital halftone



Rendered image

0.03	0.33	0.03	0.00
0.33	1.00	0.56	0.03
0.33	1.00	1.00	0.33
0.03	0.33	0.33	0.03

*

Equivalent grayscale Image

$$\underbrace{\hat{g}[m,n]}_{\text{Equivalent grayscale image}} = \Omega_{\text{model}}\left\{\underbrace{g[m+k, n+l]}_{\text{Digital halftone}}, (k,l) \in [-K:K] \times [-K:K]\right\}$$

where, K=1 or 2, when considering 3x3 or 5x5 neighborhood, respectively

*** F. A. Baqai and J. P. Allebach, "Halftoning via Direct Binary Search Using Analytical and Stochastic Printer Models"**

Developing Model-3 × 3: Representation for Dot Configurations

Representation for dot configurations:
Indexing the 3×3 dot configurations

2^0	2^1	2^2
2^3	2^4	2^5
2^6	2^7	2^8

0	0	0
0	0	0
0	0	0

0

1	0	0
0	0	0
0	0	0

1

0	1	0
0	0	0
0	0	0

2

1	1	0
0	0	0
0	0	0

3

....

1	1	1
1	1	1
1	1	1

511

Let χ be the operator to map 3×3 dot configuration to its index value.

$$\chi([0\ 1\ 0\ ;0\ 0\ 0\ ;\ 0\ 0\ 0]) = 2$$

Indexed image: $I_g[m,n] = \chi(g[m-1:m+1, n-1:n+1])$

Generating Equivalent Grayscale Image (using Model-3×3)

0.33	0.33	0.33	0.33	0.33	0.33
0.33	0.33	0.33	0.33	0.33	0.33
0.33	0.33	0.33	0.33	0.33	0.33
0.33	0.33	0.33	0.33	0.33	0.33
0.33	0.33	0.33	0.33	0.33	0.33
0.33	0.33	0.33	0.33	0.33	0.33

$f[m,n]$

Continuous-tone image

Halftoning

0	0	0	0	1	0
1	**0**	**1**	**0**	0	0
0	**0**	**0**	**0**	1	0
0	**0**	**0**	**0**	0	1
0	1	1	0	0	0
0	0	0	0	1	0

$g[m,n]$

Binary halftone

χ

Index	**Mean**
0	0.000
1	0.027
2	0.054
3	0.216
510	0.980
511	1.000

133	322	129	96	16	268
16	40	16	268	130	97
2	5	**2**	33	272	140
320	384	192	68	34	17
40	48	24	264	132	66
133	70	3	289	144	72

$I_g[m,n]$

Indexed image

LUT from scan analysis

0.243	0.209	0.15	0.109	0.175	0.188
0.175	0.143	0.175	0.188	0.255	0.192
0.054	0.064	0.054	0.107	0.396	0.391
0.086	0.294	0.282	0.064	0.285	0.329
0.143	0.485	0.463	0.101	0.154	0.125
0.243	0.278	0.216	0.275	0.486	0.189

$\hat{g}[m,n]$

Equivalent grayscale image