Ph.D. Defense

Analysis and Design of Vector Error Diffusion Systems for Image Halftoning

Niranjan Damera-Venkata

Embedded Signal Processing Laboratory

The University of Texas at Austin

Austin TX 78712-1084

Committee Members

Prof. Ross Baldick

Prof. Alan C. Bovik

Prof. Gustavo de Veciana

Prof. Brian L. Evans (advisor)

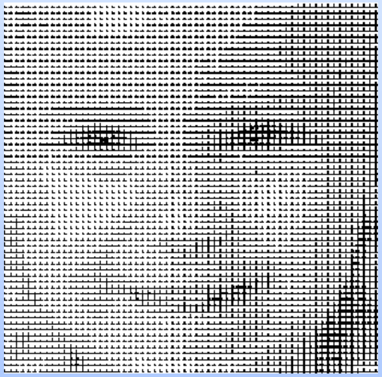
Prof. Wilson S. Geisler

Prof. Joydeep Ghosh

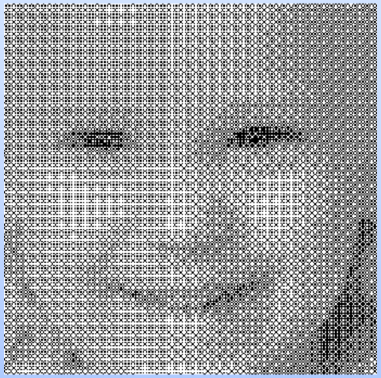
Outline

- **Digital halftoning**
  - Grayscale error diffusion halftoning
  - Color error diffusion halftoning
- ***Contribution #1:*** **Matrix gain model for color error diffusion**
- ***Contribution #2:*** **Design of color error diffusion systems**
- ***Contribution #3:*** **Block error diffusion**
  - Clustered-dot error diffusion halftoning
  - Embedded multiresolution halftoning
- **Contributions**

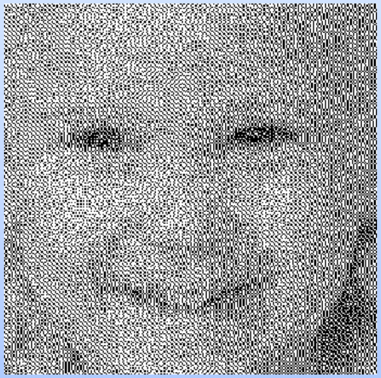
Digital Halftoning



Clutered Dot Dither
AM Halftoning

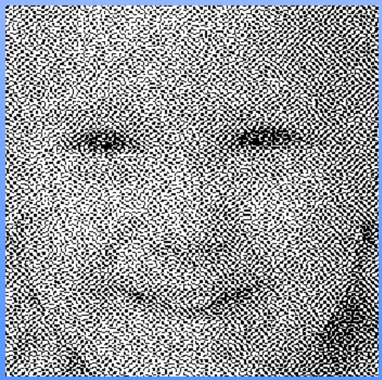


Dispersed Dot Dither
FM Halftoning

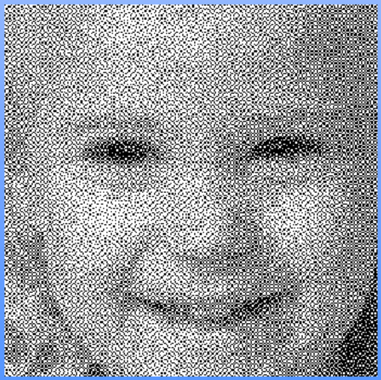


Error Diffusion
FM Halftoning 1975

Blue-noise Mask
FM Halftoning 1993



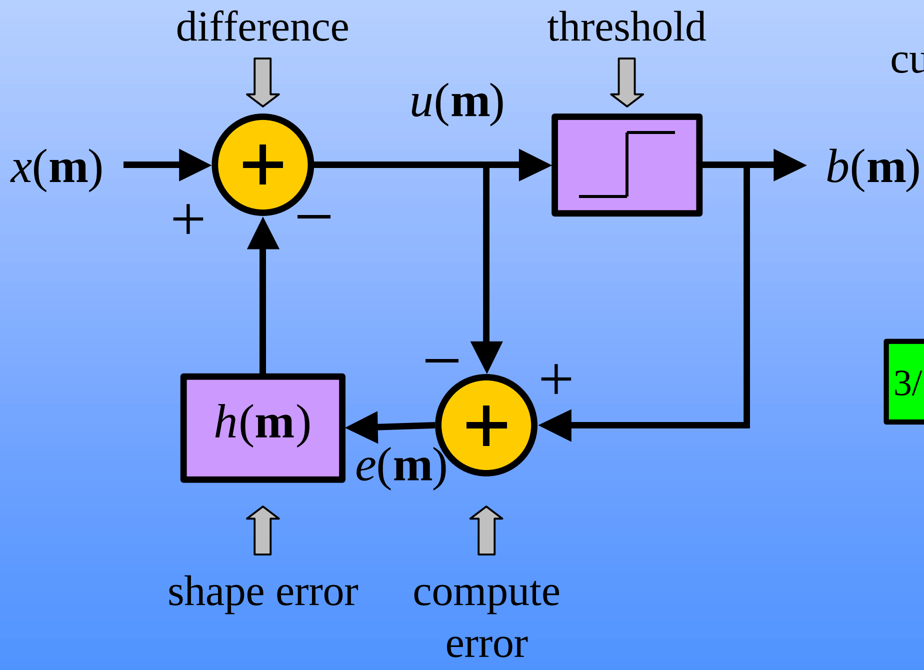
Green-noise Halftoning
AM-FM Halftoning 1992

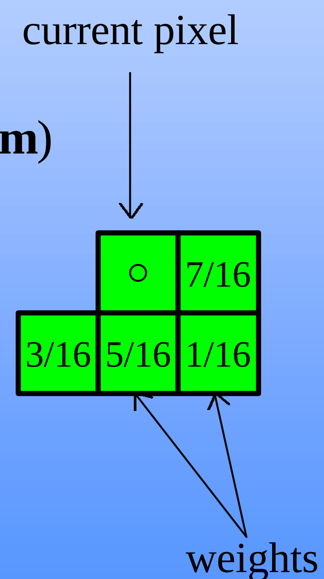


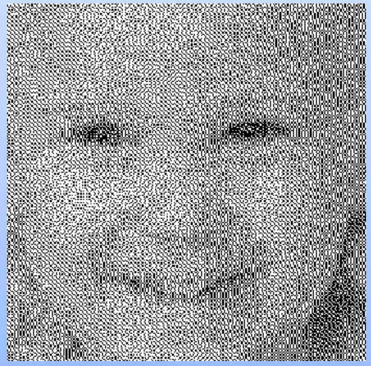
Direct Binary Search
FM Halftoning 1992

Grayscale Error Diffusion

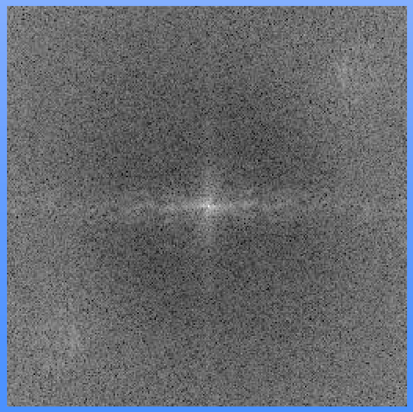
- **Shape quantization noise into high frequencies**
- **Two-dimensional sigma-delta modulation**
- **Design of error filter is key to high quality**







Error Diffusion



Spectrum

Modeling Grayscale Error Diffusion

- **Sharpening is caused by a correlated error image [Knox, 1992]**

Floyd-Steinberg

Jarvis

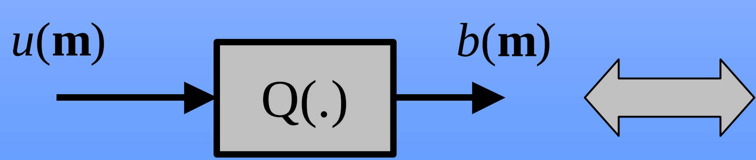
Error images

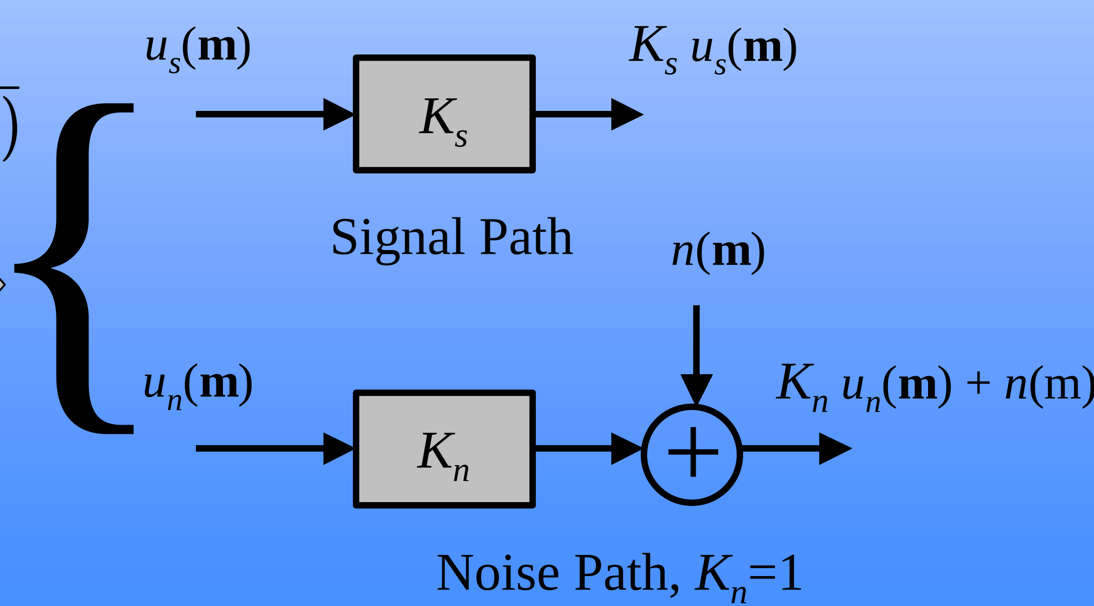
Halftones

Modeling Grayscale Error Diffusion

- **Apply sigma-delta modulation analysis to two dimensions**
  - Linear gain model for quantizer in 1-D [Ardalan and Paulos, 1988]
  - Linear gain model for grayscale image [Kite, Evans, Bovik, 2000]
  - Signal transfer function (STF) and noise transfer function (NTF)
  - 1 – H(**z**) is highpass so H(**z**) is lowpass

$$STF = \frac{B_s(\mathbf{z})}{X(\mathbf{z})} = \frac{K_s}{1 + (K_s - 1)H(\mathbf{z})}$$



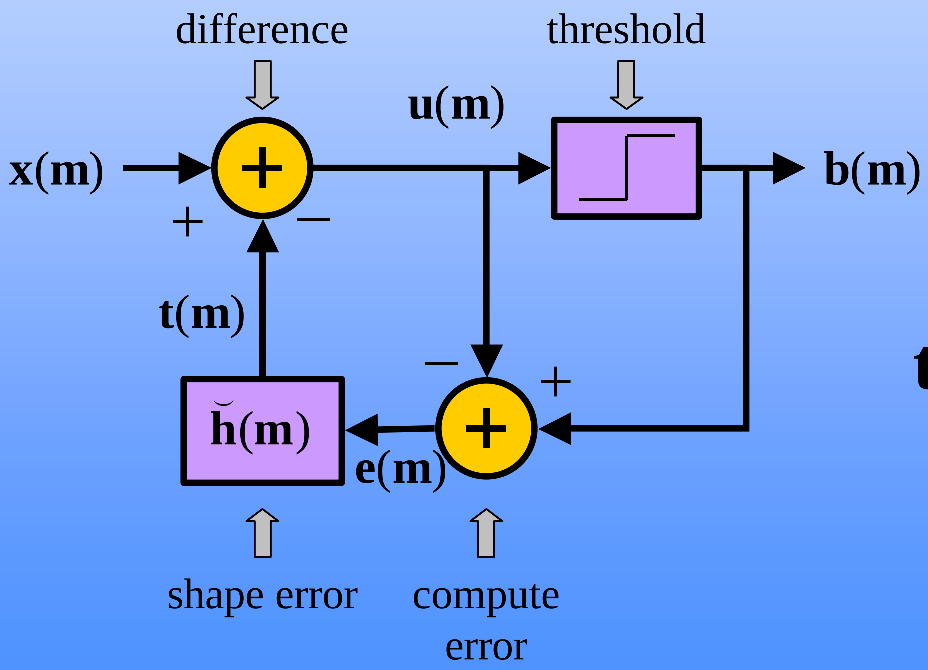


$$NTF = \frac{B_n(\mathbf{z})}{N(\mathbf{z})} = 1 - H(\mathbf{z})$$

Vector Color Error Diffusion

- **Error filter has matrix-valued coefficients**
- **Algorithm for adapting matrix coefficients**

[Akarun, Yardimci, Cetin 1997]



$$\mathbf{t}(\mathbf{m}) = \sum_{\mathbf{k} \in \wp} \underbrace{\breve{\mathbf{h}}(\mathbf{k})}_{\text{matrix}} \; \underbrace{\mathbf{e}(\mathbf{m}-\mathbf{k})}_{\text{vector}}$$

Color Error Diffusion

- **Open issues**
  - Modeling of color error diffusion in the frequency domain
  - Designing robust fixed matrix-valued error filters
  - Efficient implementation
  - Linear model for the human visual system for color images
- **Contributions**
  - “Matrix gain model” for linearizing color error diffusion
  - Model-based error filter design
  - Parallel implementation of the error filter as a filter bank

Contribution #1:

The Matrix Gain Model

- **Replace scalar gain with a matrix**

$$\breve{\mathbf{K}}_s = \arg\min_{\breve{\mathbf{A}}} E\left[\left\|\mathbf{b}(\mathbf{m}) - \breve{\mathbf{A}}\,\mathbf{u}(\mathbf{m})\right\|^2\right] = \breve{\mathbf{C}}_{\mathbf{bu}}\breve{\mathbf{C}}_{\mathbf{uu}}^{-1}$$

$$\breve{\mathbf{K}}_n = \breve{\mathbf{I}}$$

$\mathbf{u}(\mathbf{m})$ quantizer input

$\mathbf{b}(\mathbf{m})$ quantizer output

  - Noise is uncorrelated with signal component of quantizer input
  - Convolution becomes matrix–vector multiplication in frequency domain

$$\mathbf{B}_n(\mathbf{z}) = \left(\breve{\mathbf{I}} - \breve{\mathbf{H}}(\mathbf{z})\right)\mathbf{N}(\mathbf{z})$$

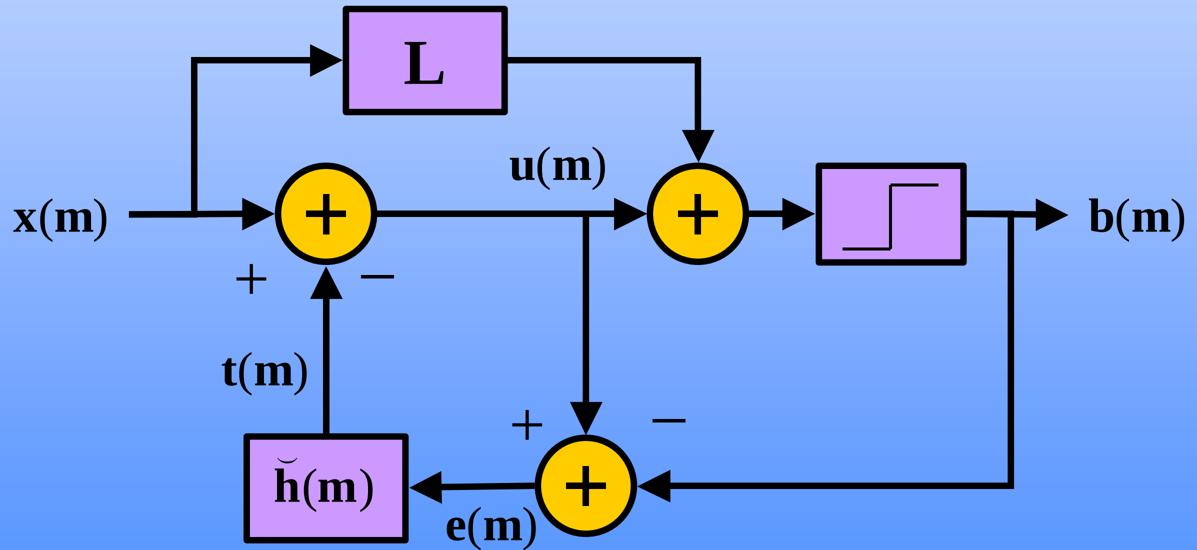
Noise component of output

$$\mathbf{B}_s(\mathbf{z}) = \breve{\mathbf{K}}\left(\breve{\mathbf{I}} + \breve{\mathbf{H}}(\mathbf{z})\left(\breve{\mathbf{K}} - \breve{\mathbf{I}}\right)\right)^{-1}\mathbf{X}(\mathbf{z})$$

Signal component of output

How to Construct an Undistorted Halftone

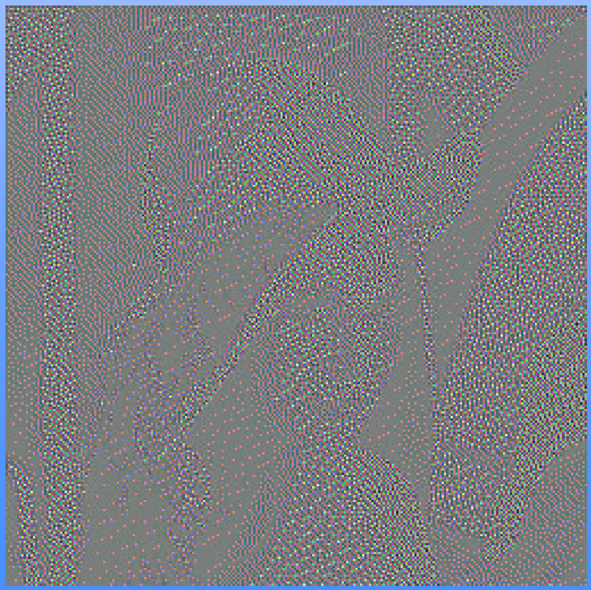
- **Pre-filter with inverse of signal transfer function to obtain undistorted halftone** $\breve{\mathbf{G}}(\mathbf{z}) = \left[\breve{\mathbf{I}} + \breve{\mathbf{H}}(\mathbf{z})\left(\breve{\mathbf{K}} - \breve{\mathbf{I}}\right)\right]\breve{\mathbf{K}}^{-1}$
- **Pre-filtering is equivalent to the following when** $\breve{\mathbf{L}} = \breve{\mathbf{K}}^{-1} - \breve{\mathbf{I}}$



Modified error diffusion

Validation #1 by Constructing Undistorted Halftone

- **Generate linearly undistorted halftone**
- **Subtract original image from halftone**
- **Since halftone should be "undistorted", the residual should be uncorrelated with the original**

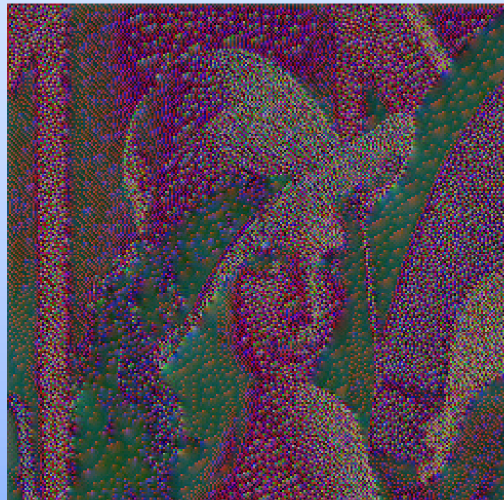


Correlation matrix of residual image (undistorted halftone minus input image) with the input image

$$\breve{\mathbf{C}}_{\mathbf{rx}} = \begin{pmatrix} 0.0006 & 0.0097 & 0.0020 \\ 0.0013 & 0.0114 & 0.0073 \\ 0.0044 & 0.0024 & 0.0043 \end{pmatrix}$$

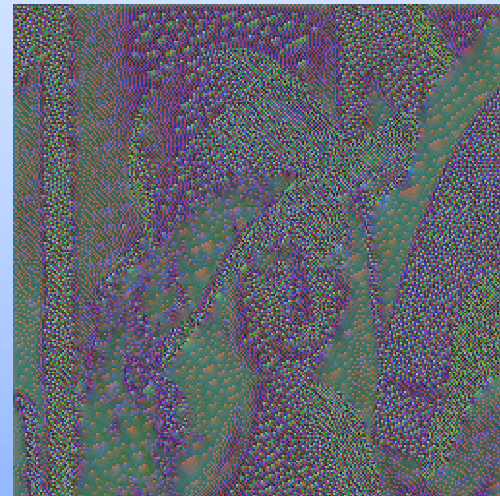
Contribution #1: Matrix Gain Model

Validation #2 by Knox's Conjecture



Correlation matrix for an error image and input image for an error diffused halftone

$$\breve{\mathbf{C}}_{\mathbf{ex}} = \begin{pmatrix} 0.3664 & 0.0019 & 0.1778 \\ 0.2661 & 0.2348 & 0.1817 \\ 0.2173 & 0.1839 & 0.1816 \end{pmatrix}$$



$$\mathbf{E}_s(\mathbf{z}) = 0$$

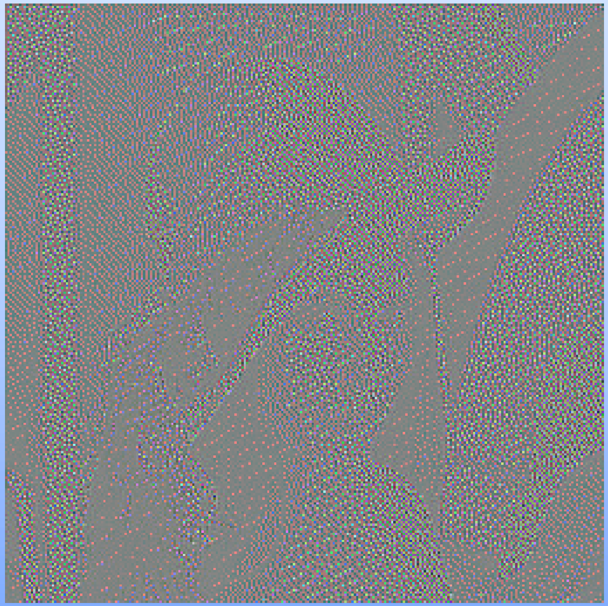
$$\mathbf{E}_n(\mathbf{z}) = \mathbf{N}(\mathbf{z})$$

Correlation matrix for an error image and input image for an undistorted halftone

$$\breve{\mathbf{C}}_{\mathbf{ex}} = \begin{pmatrix} 0.0485 & 0.0839 & 0.0082 \\ 0.0153 & 0.0550 & 0.0229 \\ 0.0160 & 0.0004 & 0.0247 \end{pmatrix}$$

Validation #3 by Distorting Original Image

- **Validation by constructing a linearly distorted original**
  - Pass original image through error diffusion with matrix gain substituted for quantizer
  - Subtract resulting color image from color halftone
  - Residual should be shaped uncorrelated noise



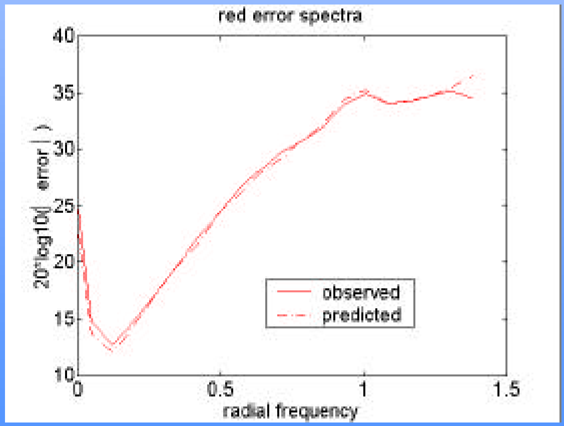
Correlation matrix of residual image (halftone minus distorted input image) with the input image

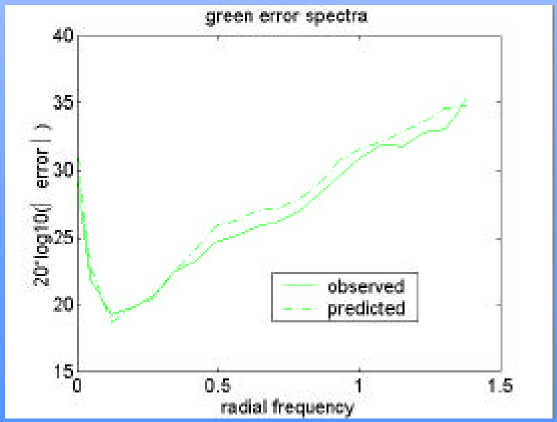
$$\breve{\mathbf{C}}_{\mathbf{rx}} = \begin{pmatrix} 0.0012 & 0.0140 & 0.0004 \\ 0.0007 & 0.0057 & 0.0126 \\ 0.0015 & 0.0101 & 0.0056 \end{pmatrix}$$

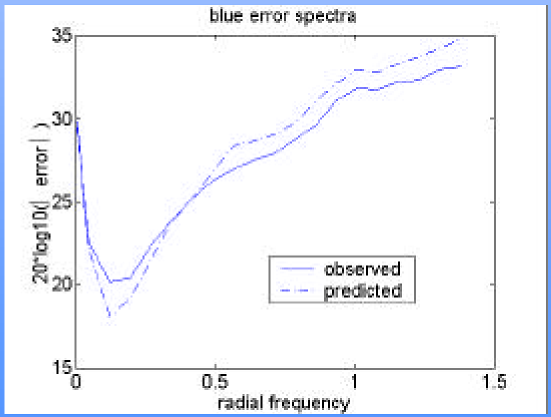
Contribution #1: Matrix Gain Model

Validation #4 by Noise Shaping

- **Noise process is error image for an undistorted halftone**
- **Use model noise transfer function to compute noise spectrum**
- **Subtract original image from modeled halftone and compute actual noise spectrum**







Contribution #2

Designing of the Error Filter

- **Eliminate linear distortion filtering before error diffusion**
- **Optimize error filter h(m) for noise shaping**

$$\min E\left[\left\|\mathbf{b}_n(\mathbf{m})\right\|^2\right] = E\left[\left\|\breve{\mathbf{v}}(\mathbf{m}) * \left(\breve{\mathbf{I}} - \breve{\mathbf{h}}(\mathbf{m})\right) * \mathbf{n}(\mathbf{m})\right\|^2\right]$$

Subject to diffusion constraints

$$\left(\sum_{\mathbf{m}} \breve{\mathbf{h}}(\mathbf{m})\right)\mathbf{1} = \mathbf{1}$$

where

$\breve{\mathbf{v}}(\mathbf{m})$ linear model of human visual system

$*$ matrix-valued convolution

Generalized Optimum Solution

- **Differentiate scalar objective function for visual noise shaping with respect to matrix-valued coefficients**

$$\frac{d\left\{E\left[\left\|\mathbf{b}_n(\mathbf{m})\right\|^2\right]\right\}}{d\,\mathbf{h}(\mathbf{i})} = \mathbf{0} \quad \forall\, \mathbf{i} \in \wp$$

- **Write the norm as a trace and then differentiate the trace using identities from linear algebra**

$$\left\|\mathbf{x}\right\| = Tr\left(\mathbf{x}\mathbf{x}'\right)$$

$$\frac{d\left\{Tr\left(\breve{\mathbf{A}}\breve{\mathbf{X}}\right)\right\}}{d\breve{\mathbf{X}}} = \breve{\mathbf{A}}'$$

$$\frac{d\left\{Tr\left(\breve{\mathbf{X}}'\breve{\mathbf{A}}\breve{\mathbf{X}}\breve{\mathbf{B}}\right)\right\}}{d\breve{\mathbf{X}}} = \breve{\mathbf{A}}\breve{\mathbf{X}}\breve{\mathbf{B}} + \breve{\mathbf{A}}'\breve{\mathbf{X}}\breve{\mathbf{B}}'$$

$$\frac{d\left\{Tr\left(\breve{\mathbf{A}}\breve{\mathbf{X}}\breve{\mathbf{B}}\right)\right\}}{d\breve{\mathbf{X}}} = \breve{\mathbf{A}}'\breve{\mathbf{B}}'$$

$$Tr\left(\breve{\mathbf{A}}\breve{\mathbf{B}}\right) = Tr\left(\breve{\mathbf{B}}\breve{\mathbf{A}}\right)$$

Generalized Optimum Solution (cont.)

- **Differentiating and using linearity of expectation operator give a generalization of the Yule-Walker equations**

$$\sum_{\mathbf{k}} \breve{\mathbf{v}}'(\mathbf{k})\breve{\mathbf{r}}_{\mathbf{an}}(-\mathbf{i}-\mathbf{k}) = \sum_{\mathbf{p}}\sum_{\mathbf{q}}\sum_{\mathbf{s}} \breve{\mathbf{v}}'(\mathbf{s})\breve{\mathbf{v}}(\mathbf{q})\breve{\mathbf{h}}(\mathbf{p})\breve{\mathbf{r}}_{\mathbf{nn}}(-\mathbf{i}-\mathbf{s}+\mathbf{p}+\mathbf{q})$$

where

$$\mathbf{a}(\mathbf{m}) = \breve{\mathbf{v}}(\mathbf{m}) * \mathbf{n}(\mathbf{m})$$

- **Assuming white noise injection**

$$\mathbf{r}_{\mathbf{nn}}(\mathbf{k}) = E\left[\mathbf{n}(\mathbf{m})\,\mathbf{n}'(\mathbf{m}+\mathbf{k})\right] \approx d\left(\mathbf{k}\right)$$

$$\mathbf{r}_{\mathbf{an}}(\mathbf{k}) = E\left[\mathbf{a}(\mathbf{m})\,\mathbf{n}'(\mathbf{m}+\mathbf{k})\right] \approx \breve{\mathbf{v}}\left(-\mathbf{k}\right)$$

Contribution #2: Error Filter Design

Generalized Optimum Solution (cont.)

- **Optimum solution obtained via steepest descent algorithm**

$$\left(\nabla J\right)_{(\breve{\mathbf{h}}(\mathbf{i}))} = -\sum_{\mathbf{k}} \breve{\mathbf{v}}'(\mathbf{k})\breve{\mathbf{r}}_{\mathbf{an}}(-\mathbf{i}-\mathbf{k}) + \sum_{\mathbf{p}}\sum_{\mathbf{q}}\sum_{\mathbf{s}} \breve{\mathbf{v}}'(\mathbf{s})\breve{\mathbf{v}}(\mathbf{q})\breve{\mathbf{h}}(\mathbf{p})\breve{\mathbf{r}}_{\mathbf{nn}}(-\mathbf{i}-\mathbf{s}+\mathbf{p}+\mathbf{q})$$

$$\breve{\mathbf{h}}^{(q+1)}(\mathbf{i}) = \mathrm{P}\left(\breve{\mathbf{h}}^{(q)}(\mathbf{i}) - a\left\{\left(\nabla J\right)_{(\breve{\mathbf{h}}^{(q)}(\mathbf{i}))}\right\}\right)$$

$$\mathrm{P}\left(\breve{\mathbf{f}}(\mathbf{i})\right) = \breve{\mathbf{f}}(\mathbf{i}) - \left(\frac{1}{3|\wp|}\right)\left(\sum_{\mathbf{m}\in\wp}\breve{\mathbf{f}}(\mathbf{m}) - \breve{\mathbf{I}}\right)\mathbf{1}$$

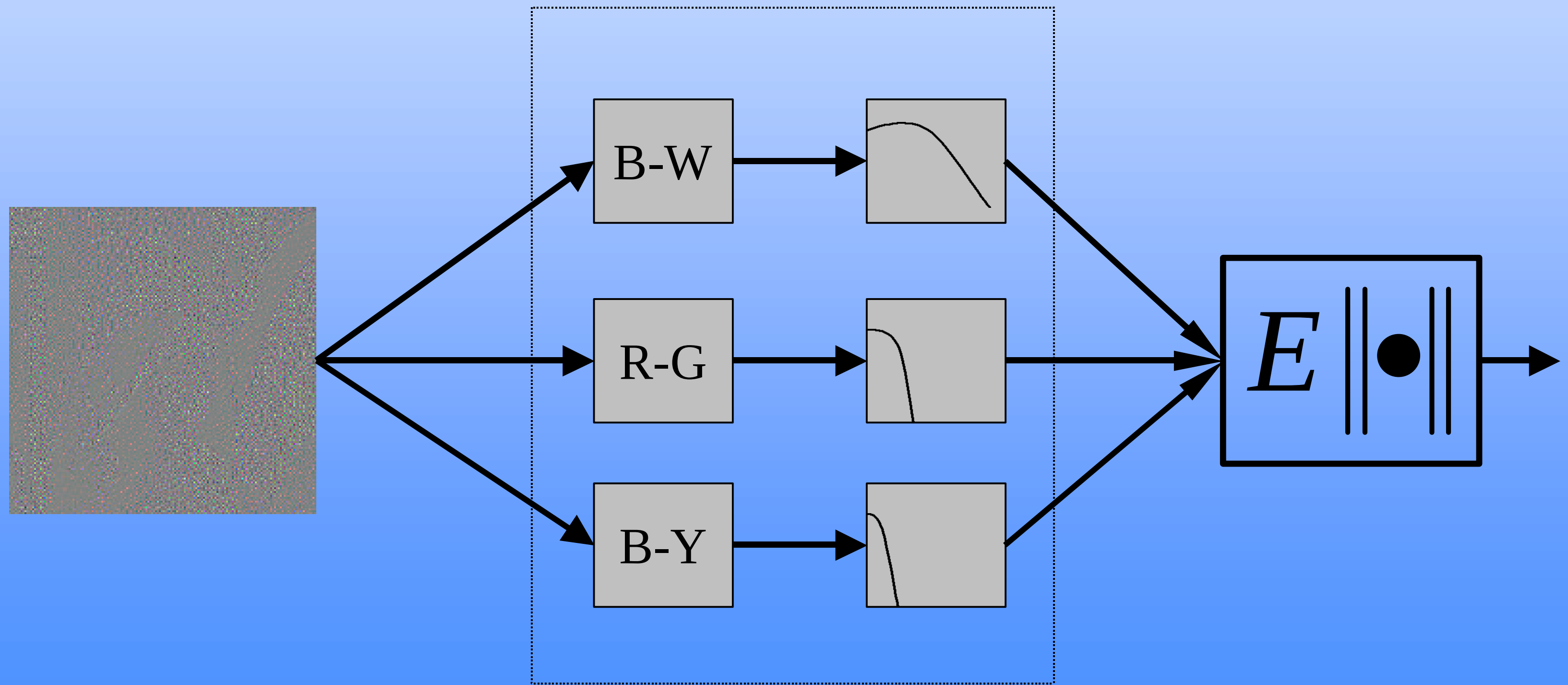
a - convergence rate parameter

P - projection operator

q - iteration number

Linear Color Vision Model

- **Pattern-Color separable model [Poirson and Wandell, 1993]**
  - Forms the basis for S-CIELab [Zhang and Wandell, 1996]
  - Pixel-based color transformation



Opponent representation

Spatial filtering

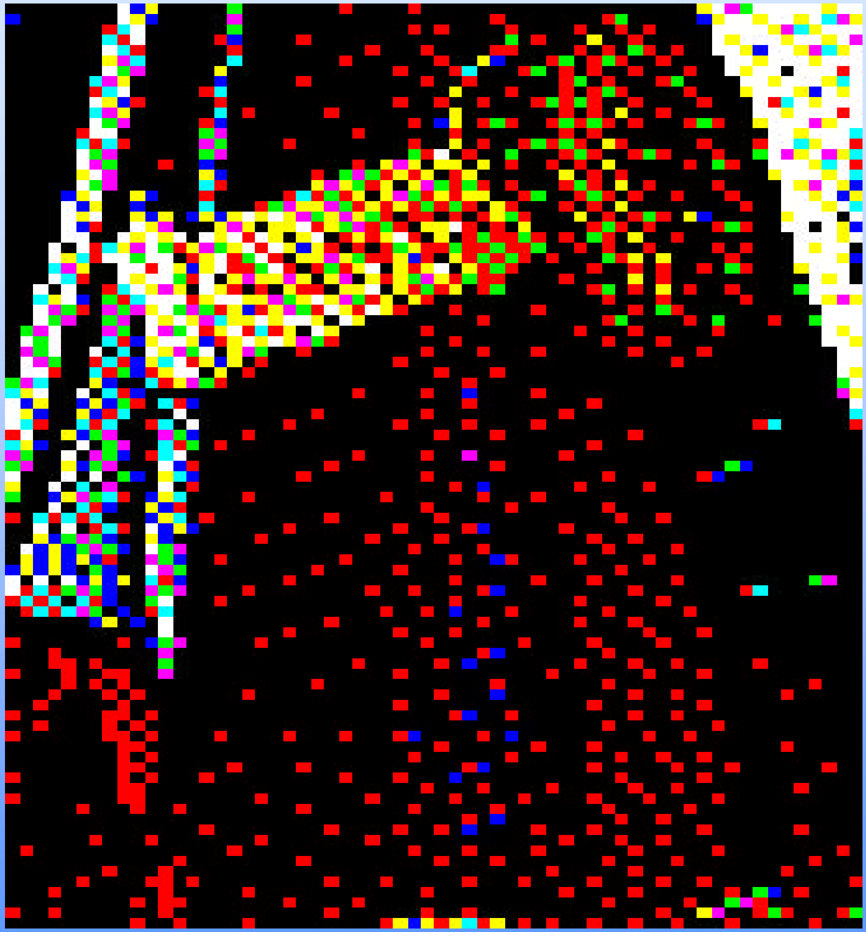
Contribution #2: Error Filter Design

Linear Color Vision Model

- **Undo gamma correction on RGB image**
- **Color separation**
  - Measure power spectral distribution of RGB phosphor excitations
  - Measure absorption rates of long, medium, short (LMS) cones
  - Device dependent transformation ***C*** from RGB to LMS space
  - Transform LMS to opponent representation using ***O***
  - Color separation may be expressed as ***T = OC***
- **Spatial filtering is incorporated using matrix filter** $\breve{\mathbf{d}}(\mathbf{m})$
- **Linear color vision model**

$\breve{\mathbf{v}}(\mathbf{m}) = \breve{\mathbf{d}}(\mathbf{m})\breve{\mathbf{T}}$ where $\breve{\mathbf{d}}(\mathbf{m})$ is a diagonal matrix

Floyd-Steinberg

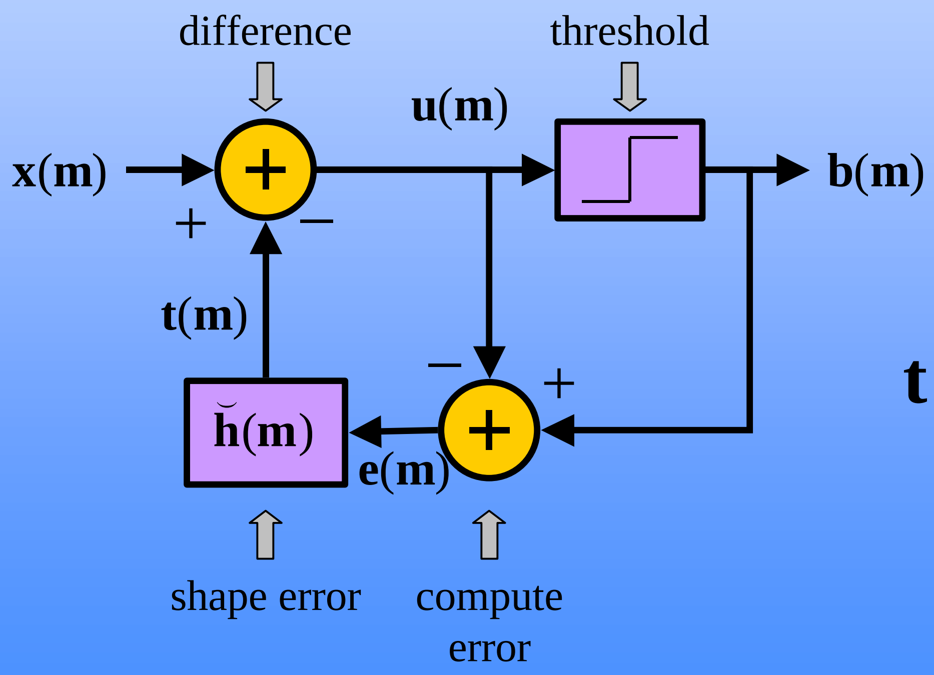


Optimum Filter

Contribution #3

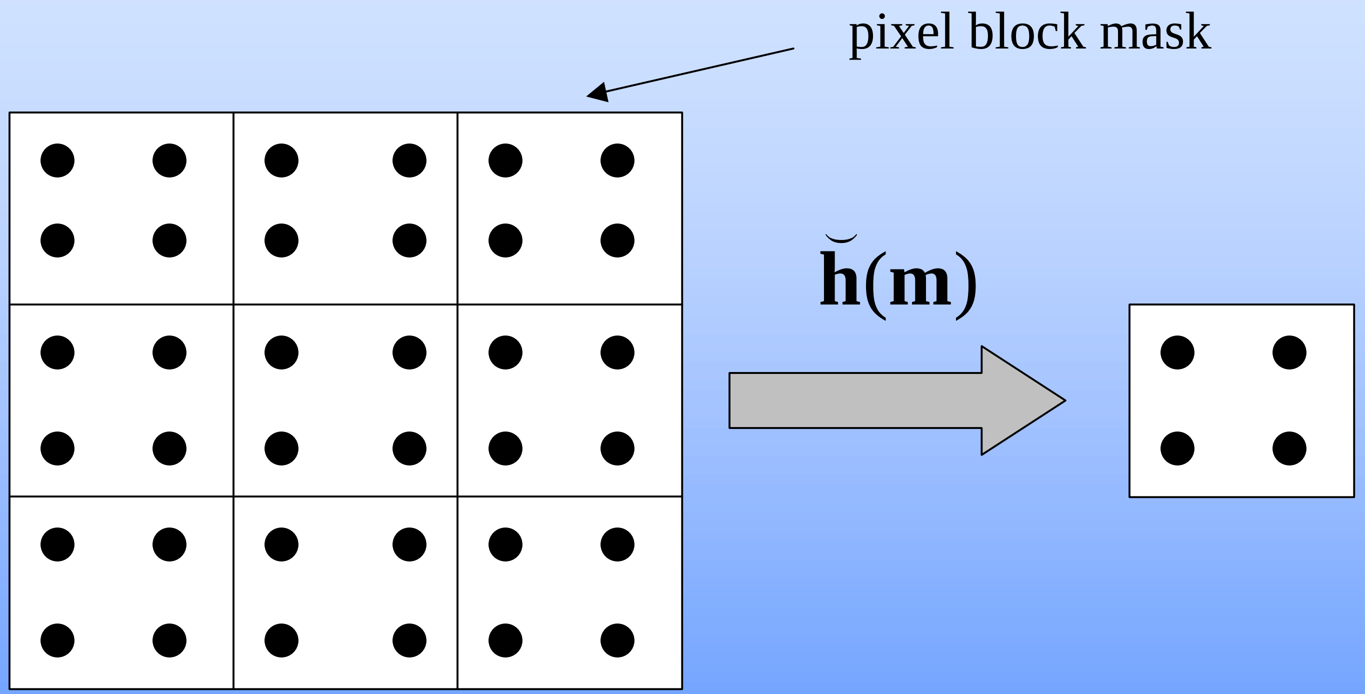
Block Error Diffusion

- **Input grayscale image is "blocked"**
- **Error filter uses all samples from neighboring blocks and diffuses an error block**



$$\mathbf{t}(\mathbf{m}) = \sum_{\mathbf{k}\in S} \breve{\mathbf{h}}(\mathbf{k})\,\mathbf{e}(\mathbf{m}-\mathbf{k})$$

Block Interpretation of Vector Error Diffusion



- **Four linear combinations of the 36 pixels are required to compute the output pixel block**

Contribution #3: Block Error Diffusion

Block FM Halftoning

- **Why not "block" standard error diffusion output?**
  - Spatial aliasing problem
  - Blurred appearance due to prefiltering
- **Solution**
  - Control dot shape using block error diffusion
  - Extend conventional error diffusion in a natural way
- **Extensions to block error diffusion**
  - AM-FM halftoning
  - Sharpness control
  - Multiresolution halftone embedding
  - Fast parallel implementation

Contribution #3: Block Error Diffusion

Block FM Halftoning Error Filter Design

- **Start with conventional error filter prototype**

$$\tilde{\mathbf{a}} = \begin{bmatrix} \frac{1}{16} & \frac{5}{16} & \frac{3}{16} & \frac{7}{16} \end{bmatrix}$$

- **Form block error filter as Kronecker product**

$$\breve{\Gamma} = \tilde{\mathbf{a}} \otimes \breve{\mathbf{D}} \qquad \breve{\mathbf{D}} \text{ diffusion matrix}$$

- **Satisfies "lossless" diffusion constraint**

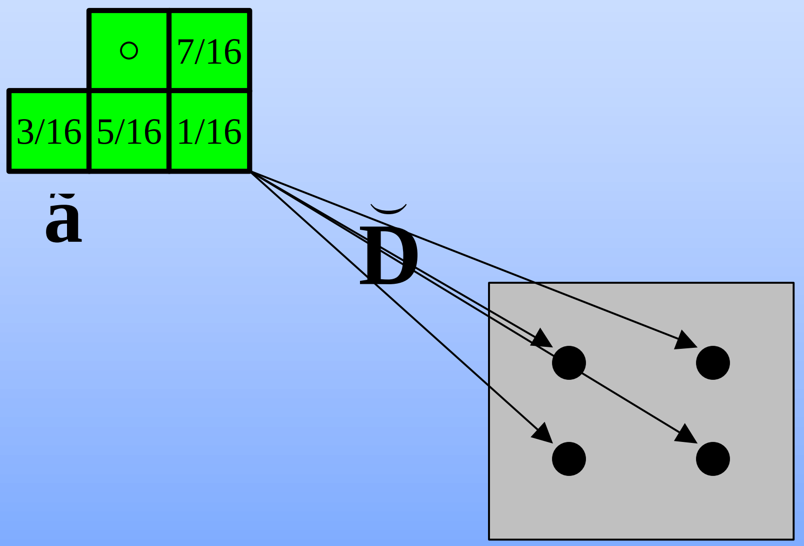
$$\breve{\Gamma}\mathbf{1} = \mathbf{1} \qquad \breve{\Gamma} \geq \breve{\mathbf{0}}$$

- **Diffusion matrix satisfies**

$$\breve{\mathbf{D}}\mathbf{1} = \mathbf{1} \qquad \breve{\mathbf{D}} \geq \breve{\mathbf{0}}$$

Block FM Halftoning Error Filter Design

- **FM nature of algorithm controlled by scalar filter prototype**
- **Diffusion matrix decides distribution of error within a block**
- **In-block diffusions are constant for all blocks to preserve isotropy**



Block FM Halftoning Results

- **Vector error diffusion with diffusion matrix**

$$\breve{\mathbf{D}} = \frac{1}{N^2}\left[\breve{\mathbf{1}}\right]$$

N is the block size

Pixel replication

Floyd-Steinberg

Jarvis

Contribution #3: Block Error Diffusion

Block FM Halftoning with Arbitrary Shapes

Plus dots

Cross dots

Contribution #3: Block Error Diffusion

Embedded Multiresolution Halftoning

- **Only involves designing the diffusion matrix**
  - FM Halftones when downsampled are also FM halftones

LMH	H	MH	H
H	H	H	H
MH	H	MH	H
H	H	H	H

Halftone pixels at Low, Medium and High resolutions

- **Error at a pixel is diffused to the pixels of the same color**

Contribution #3: Block Error Diffusion

Embedded Halftoning Results

High resolution halftone

Medium resolution halftone

Low resolution halftone

Simple down-sampling

Contributions

- **Matrix gain model for vector color error diffusion**
  - Eliminated linear distortion by pre-filtering
  - Validated model in three other ways
- **Model based error filter design for a calibrated device**
- **Block error diffusion**
  - FM halftoning
  - AM-FM halftoning (not presented)
  - Embedded multiresolution halftoning
- **Efficient parallel implementation (not presented)**

Published Halftoning Work Not in Dissertation

N. Damera-Venkata and B. L. Evans, ``Adaptive Threshold Modulation for Error Diffusion Halftoning," *IEEE Transactions on Image Processing*, January 2001, to appear.

T. D. Kite, N. Damera-Venkata, B. L. Evans and A. C. Bovik, "A Fast, High Quality Inverse Halftoning Algorithm for Error Diffused Halftoned images," *IEEE Transactions on Image Processing*,, vol. 9, no. 9, pp. 1583-1593, September 2000.

N. Damera-Venkata, T. D. Kite , W. S. Geisler, B. L. Evans and A. C. Bovik ,``Image Quality Assessment Based on a Degradation Model" *IEEE Transactions on Image Processing*, vol. 9, no. 4, pp. 636-651, April 2000.

N. Damera-Venkata, T. D. Kite , M. Venkataraman, B. L. Evans,``Fast Blind Inverse Halftoning" *IEEE Int. Conf. on Image Processing*, vol. 2, pp. 64-68, Oct. 4-7, 1998.

T. D. Kite, N. Damera-Venkata, B. L. Evans and A. C. Bovik, "A High Quality, Fast Inverse Halftoning Algorithm for Error Diffused Halftoned images," *IEEE Int. Conf. on Image Processing*, vol. 2, pp. 64-68, Oct. 4-7, 1998.

Submitted Halftoning Work in Dissertation

N. Damera-Venkata and B. L. Evans, ``Matrix Gain Model for Vector Color Error Diffusion," *IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing*, June 3-5, 2001, to appear.

N. Damera-Venkata and B. L. Evans, ``Design and Analysis of Vector Color Error Diffusion Systems," *IEEE Transactions on Image Processing*, submitted.

N. Damera-Venkata and B. L. Evans, ``Clustered-dot FM Halftoning Via Block Error Diffusion," *IEEE Transactions on Image Processing*, submitted.

Types of Halftoning Algorithms

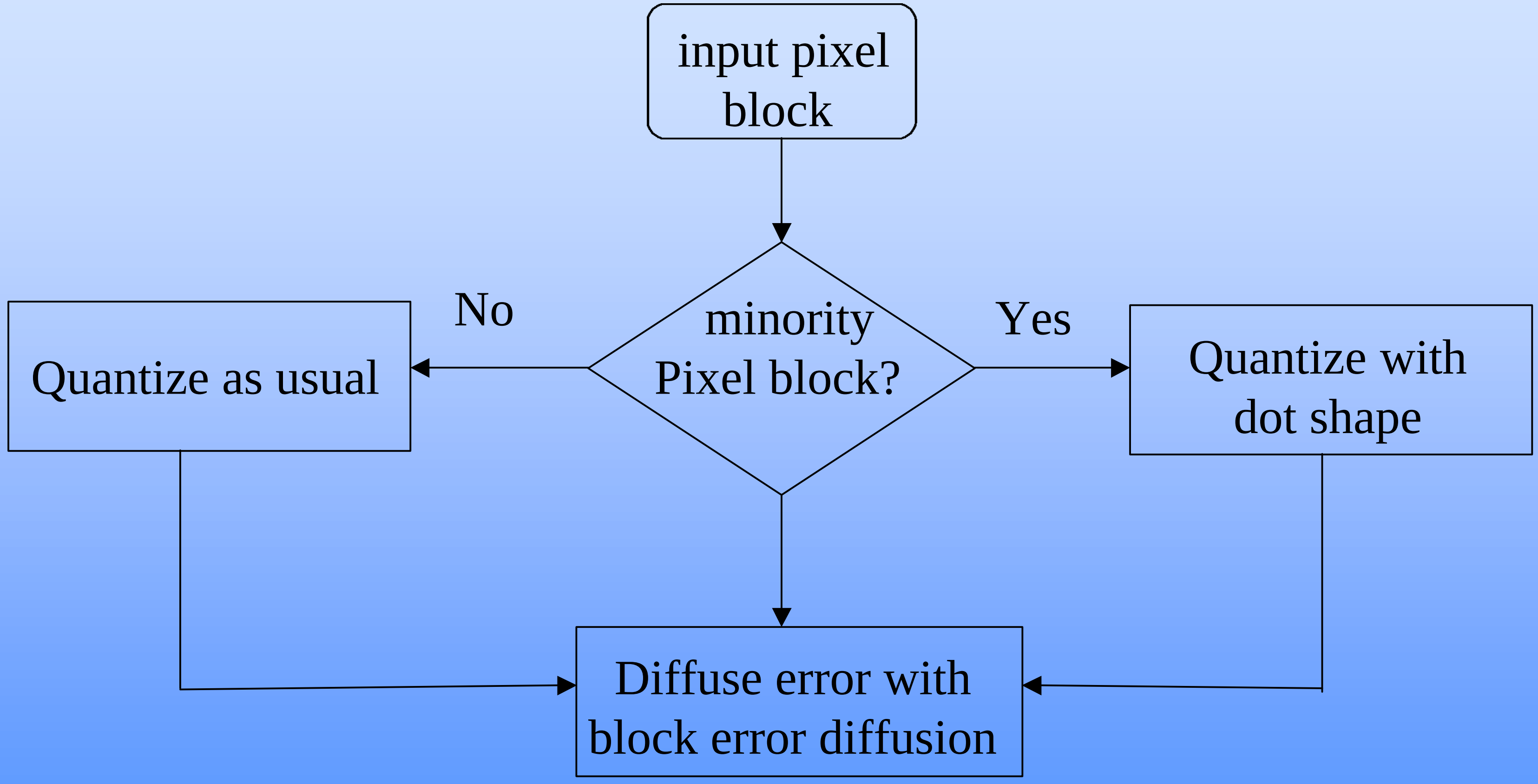
- **AM halftoning**
  - Vary dot size according to underlying graylevel
  - Clustered dot dither is a typical example (laserjet printers)
- **FM halftoning**
  - Vary dot frequency according to underlying graylevel
  - Error diffusion is typical example (inkjet printers)
- **AM-FM halftoning**
  - Vary dot size and frequency
  - Typical example is Levien’s “green-noise” algorithm [Levien 1993]

Designing Error Filter in Scalar Error Diffusion

- **Floyd-Steinberg error filter [Floyd and Steinberg, 1975]**
- **Optimize weighted error**
  - Assume error image is white noise [Kolpatzik and Bouman, 1992]
  - Use statistics of error image [Wong and Allebach, 1997]
- **Adaptive methods**
  - Adapt error filter coefficients to minimize *local* weighted mean squared error [Wong, 1996]

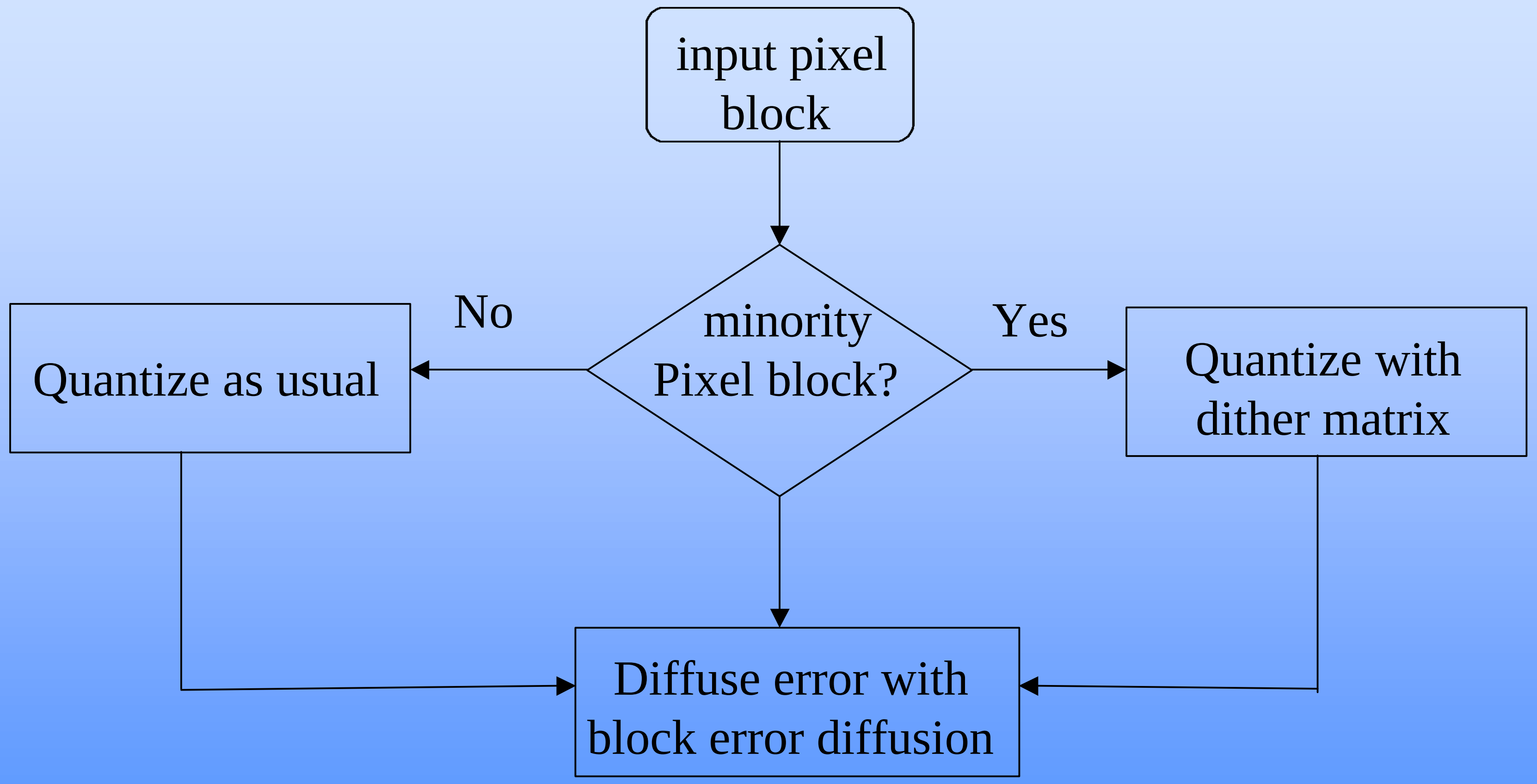
Contribution #3: Block Error Diffusion

FM Halftoning with Arbitrary Dot Shape



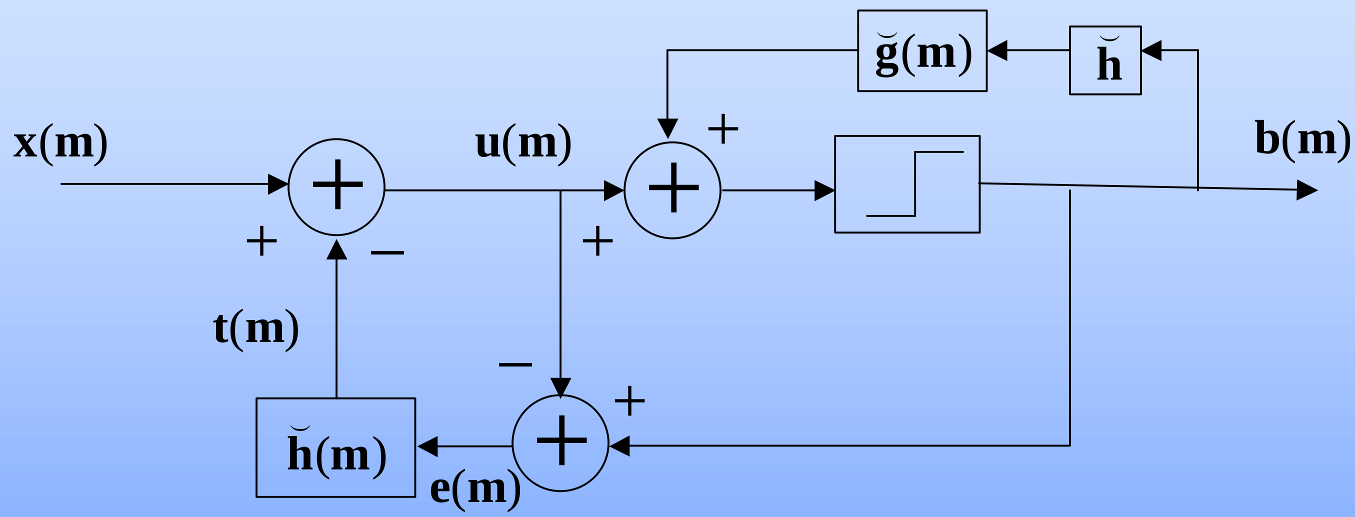
Contribution #3: Block Error Diffusion

AM-FM Halftoning with User-controlled Dot Shape



Contribution #3: Block Error Diffusion

AM-FM Halftoning with User-controlled Dot Size



Block green noise error diffusion

- **Promotes pixel-block clustering into super-pixel blocks**

Contribution #3: Block Error Diffusion

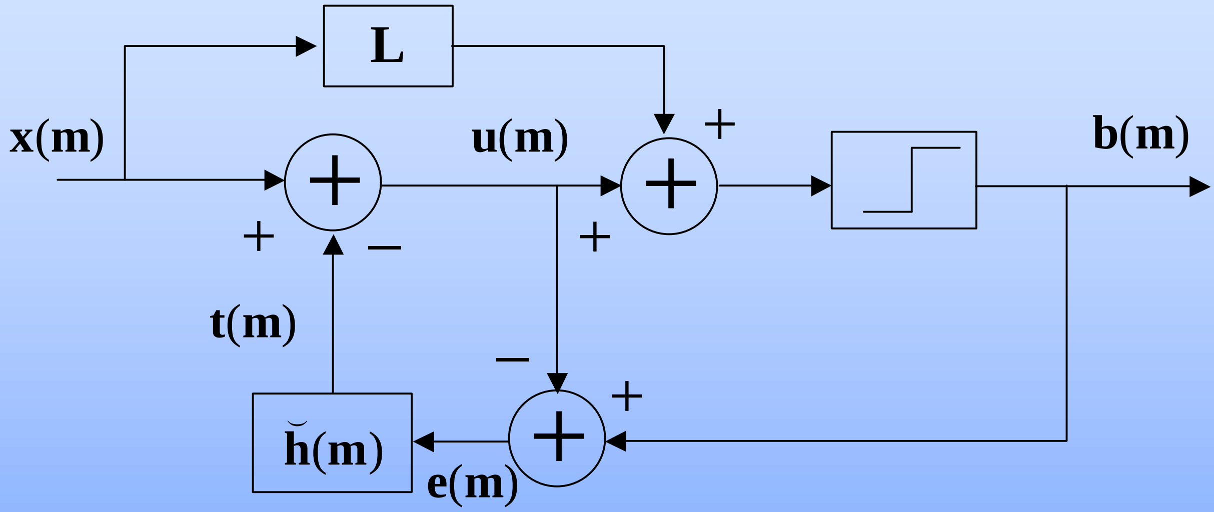
AM-FM Halftoning Results

Clustered dot dither modulation

Output dependent feedback

Contribution #3: Block Error Diffusion

Block FM Halftoning with Sharpness Control



Modified error diffusion

- **The above block diagram is equivalent to prefiltering with**

$$\breve{\mathbf{G}}_s(\mathbf{z}) = \left[\breve{\mathbf{I}} + \breve{\mathbf{H}}(\mathbf{z})\left\{\breve{\mathbf{I}} - \breve{\mathbf{H}}(\mathbf{z})\right\}\right]\breve{\mathbf{L}} + \breve{\mathbf{I}}$$

Contribution #3: Block Error Diffusion

Block FM Halftoning with Sharpness Control

$$\breve{\mathbf{L}} = 0.2\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\breve{\mathbf{L}} = 0.6\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Contribution #3: Block Error Diffusion

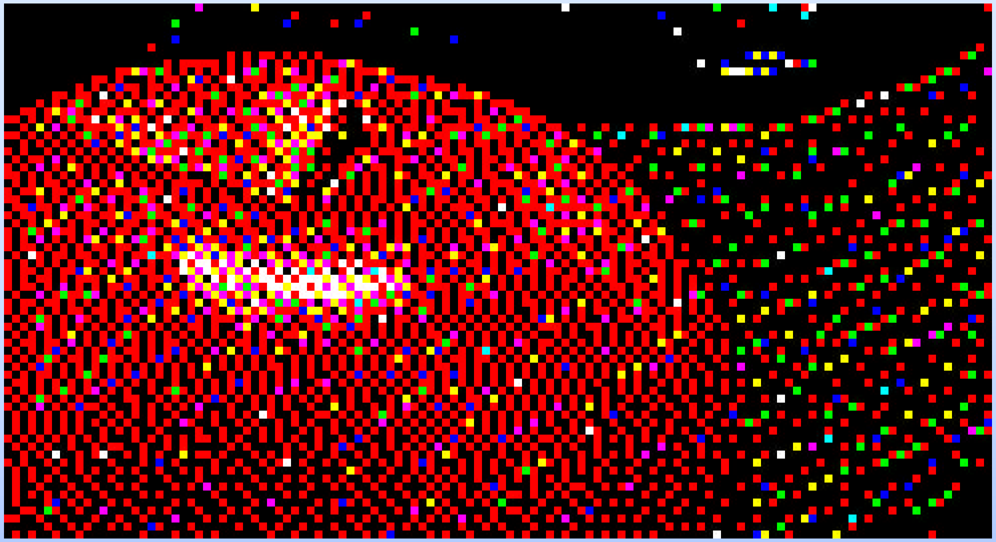
Diffusion Matrix for Embedding

$$\mathbf{l} = \begin{pmatrix}1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{pmatrix}'$$

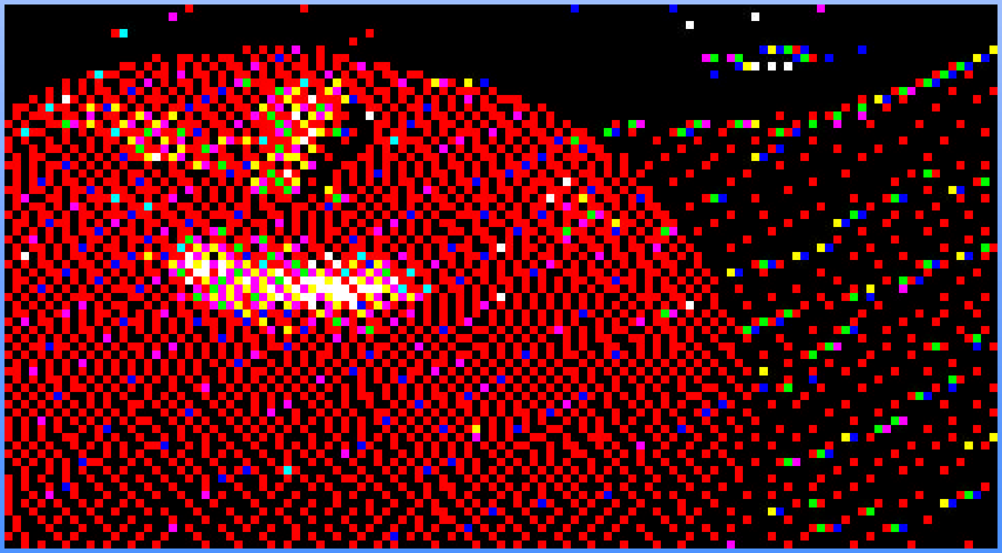
$$\mathbf{m} = \begin{pmatrix}0 & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0\end{pmatrix}'$$

$$\mathbf{h} = \begin{pmatrix}0 & \frac{1}{12} & 0 & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & 0 & \frac{1}{12} & 0 & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12}\end{pmatrix}'$$

$$\breve{\mathbf{D}} = \begin{pmatrix}\mathbf{l} & \mathbf{h} & \mathbf{m} & \mathbf{h} & \mathbf{h} & \mathbf{h} & \mathbf{h} & \mathbf{h} & \mathbf{m} & \mathbf{h} & \mathbf{m} & \mathbf{h} & \mathbf{h} & \mathbf{h} & \mathbf{h} & \mathbf{h}\end{pmatrix}'$$



Floyd-Steinberg

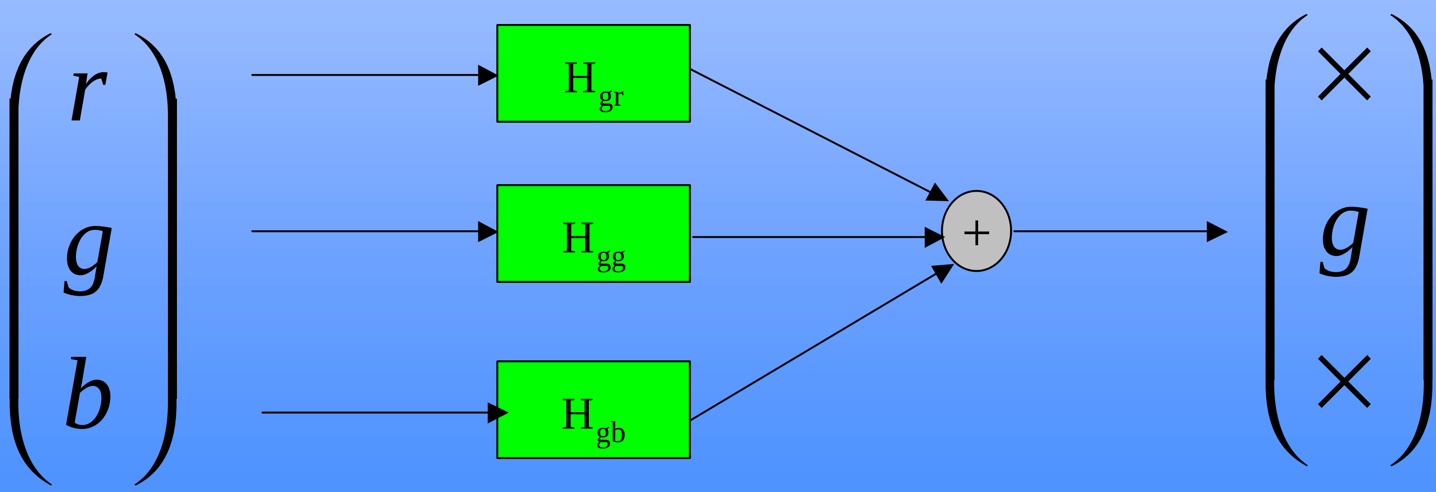


Optimum Filter

Contribution #4:

Implementation of Vector Color Error Diffusion

$$\breve{\mathbf{H}}(\mathbf{z}) = \begin{pmatrix} H_{rr}(\mathbf{z}) & H_{rg}(\mathbf{z}) & H_{rb}(\mathbf{z}) \\ H_{gr}(\mathbf{z}) & H_{gg}(\mathbf{z}) & H_{gb}(\mathbf{z}) \\ H_{br}(\mathbf{z}) & H_{bg}(\mathbf{z}) & H_{bb}(\mathbf{z}) \end{pmatrix}$$



Contribution #4:

Implementation of Block Error Diffusion