

***FAST REHALFTONING AND
INTERPOLATED HALFTONING
WITH FLAT LOW-FREQUENCY
RESPONSE***

Thomas D. Kite
Audio Precision, Beaverton, OR

Prof. Brian L. Evans

Prof. Alan C. Bovik

Laboratory for Image and Video Engineering
The University of Texas at Austin

<http://anchovy.ece.utexas.edu>

OUTLINE

- Introduction to halftoning
- Halftoning by error diffusion
- Rehalftoning
 - ▶ Algorithm
 - ▶ Results
- Interpolated halftoning
 - ▶ Algorithm
 - ▶ Results
- Computational requirements
- Conclusions

HALFTONING

- **Wordlength reduction for images**
 - ▶ 8-bit to 1-bit for grayscale
 - ▶ 24-bit RGB to 8-bit for color displays
 - ▶ 24-bit RGB to CMYK for color printers

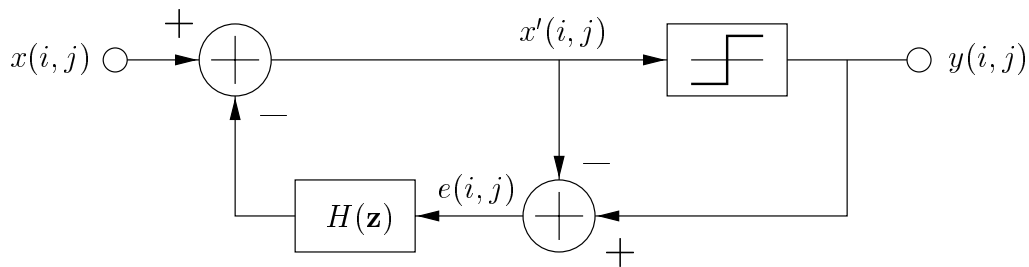
- **Applications**
 - ▶ Printers
 - ▶ Digital copiers
 - ▶ Liquid crystal displays
 - ▶ Video cards

- **Halftoning methods**
 - ▶ Screening
 - ▶ Error diffusion
 - ▶ Direct binary search
 - ▶ Hybrid schemes

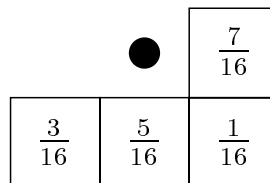
- **We consider error diffusion**

ERROR DIFFUSION

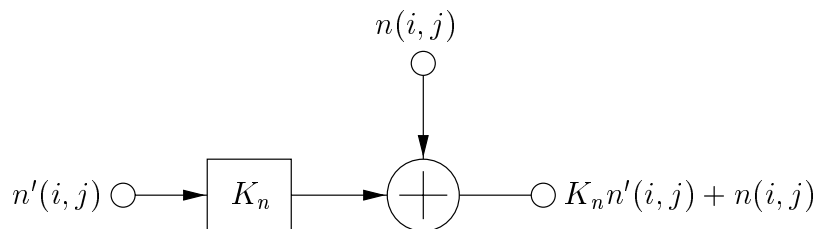
- 2-D delta-sigma modulator
- Noise shaping feedback coder



- Error filter



- Model by linearizing quantizer

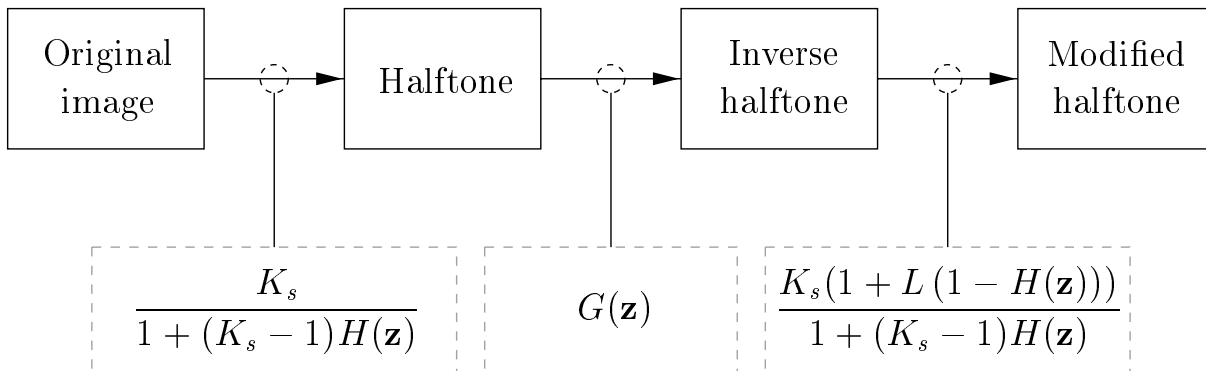


- $K_S \approx 2$ for Floyd-Steinberg filter

REHALFTONING CONCEPT

- Different devices use different halftoning methods
 - ▶ Laser printer: ordered screening
 - ▶ Inkjet printer: error diffusion
- Optimal rendering of halftone for a particular device:
 - ▶ Inverse halftone to grayscale image
 - ▶ Re-halftone grayscale image
- Inverse halftoning is expensive
 - ▶ Use simple inverse halftoning method
 - ▶ Artifacts concealed, response errors corrected in halftoning step
- Use modified error diffusion
[Eschbach and Knox, 1991]
 - ▶ Single sharpness parameter
- Assume input and output are error diffused halftones

REHALFTONING ALGORITHM



Block diagram of rehalftoning system

- Inverse halftone with 4×4 filter
 - ▶ Symmetric, FIR
 - ▶ Zeros at band edges
 - ▶ 6-bit output resolution
- Set sharpness parameter L to flatten system response at low frequencies:
 - ▶ $L \approx 0.188$ for Floyd-Steinberg filter

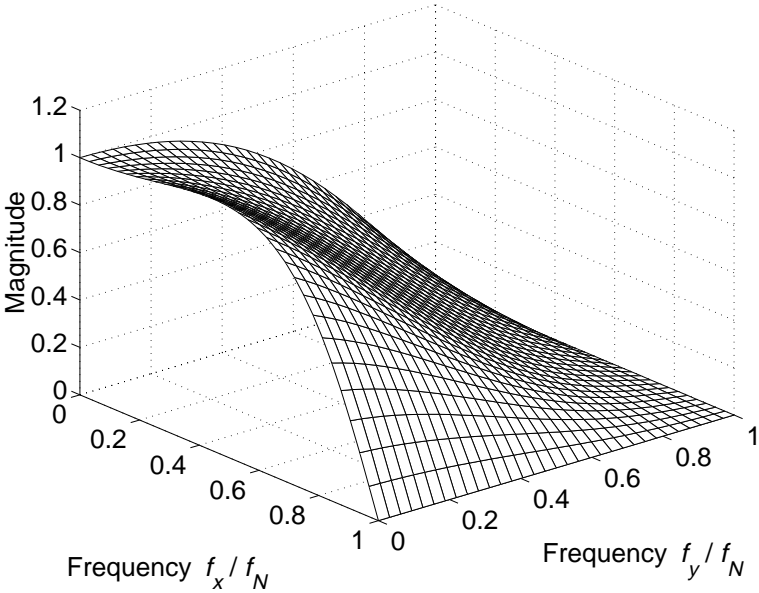
REHALFTONING RESULTS



Original image



Rehalthone



Signal transfer function

INTERPOLATED HALFTONING

- Image resizing
- Different methods (increasing cost)
 - ▶ Nearest neighbor
 - ▶ Bilinear
 - ▶ Bicubic, cubic splines, lowpass filtering
- Nearest neighbor, bilinear methods
 - ▶ Low computational cost
 - ▶ Artifacts masked by quantization noise in halftone
 - ▶ Blurring correctable by modified error diffusion
- Examine $\times 2$ interpolation; method applies to any scaling factor
- Design L for flat transfer function using linear gain model
- L constant for given interpolation scheme

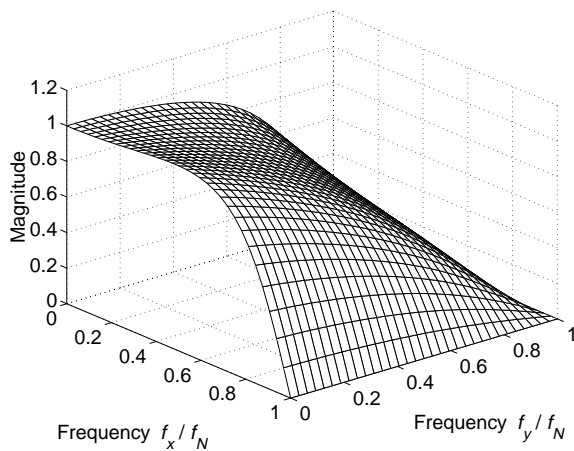
INTERPOLATION RESULTS



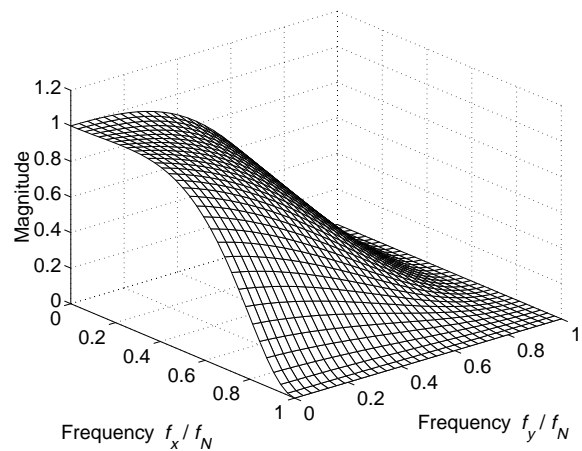
Nearest neighbor $\times 2$



Bilinear $\times 2$



Transfer function
 $L = -0.0105$



Transfer function
 $L = 0.340$

COMPUTATIONAL LOAD

- **Rehalftoning, per pixel:**
 - ▶ 34 increments (++)
 - ▶ 12–28 integer additions
 - ▶ 4 integer multiplications
 - ▶ 2 bit shifts
- **For 512×512 image:**
 - ▶ 16 million operations
 - ▶ 2060 bytes
- **$\times 2$ interpolation, per pixel:**
 - ▶ 2 increments (++)
 - ▶ 9.67 integer additions
 - ▶ 4 integer multiplications
 - ▶ 3 bit shifts
- **For 512×512 image:**
 - ▶ 5 million operations
 - ▶ 1024 bytes

CONCLUSIONS

- Rehalftoning needed for scanning, processing, re-printing
- Interpolation needed for resizing in printing, copying
- Developed fast new algorithms for error diffused halftones
 - ▶ Local memory
 - ▶ Integer arithmetic
 - ▶ Suitable for embedded implementations
- Optimized for flat response
- Web site for software and papers

`http://www.ece.utexas.edu/~bevans/projects/inverseHalftoning.html`