Lecture-2

Digital Image Processing

- *Slides adapted from Spring 2011 offering of ENEE 408G in the ECE Department, University of Maryland, College Park by Profs. Min Wu (minwu@umd.edu) and Ray Liu (kjiliu@umd.edu)*

Drexel UNIVERSITY

Image Fidelity Criteria

- Subjective measures
  - Examination by human subjects
  - Goodness scale: *excellent, good, fair, poor, …*
  - Impairment scale: *unnoticeable, just noticeable, …*
  - Comparative measures
    - *with another image or among a group of images*
- Objective (quantitative) measures
  - Mean square error and its variations
  - Pro: simple, less dependent on human subjects, and easy to handle mathematically
  - Con: not always reflect human perception
    - *Solution – HVS weighted MSE; combine with subjective evaluation*

MSE and PSNR

- Estimate MSE using average
  - Average of squared difference of pixel luminance between two images

$$\mathcal{E}_2 = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} |u(m,n) - u'(m,n)|^2 \quad \text{(average squared error)}$$

- Signal-to-noise ratio (SNR)
  - SNR = 10 $\log_{10}$ (σ_s^2 / σ_e^2) in unit of decibel (dB)

    σ_s^2 – image variance, σ_e^2 – variance of noise or error

  - PSNR = 10 $\log_{10}$ (A^2 / σ_e^2) with *A* being peak-to-peak value
    - *A = 255 for 8-bit grayscale image*
    - *> 40dB (unnoticeable), 30~40dB (good quality), < 30dB (low quality)*
    - *PSNR is about 12-15 dB higher than SNR*
    - *PSNR is widely used in image processing*

- Limitation: not always reflect human perception

Image Enhancement (1): Under Low Pixel Depth

8 bits / pixel

4 bits / pixel

2 bits / pixel

- Contouring effect
  - Visible contours on smoothly changing regions for uniform quantized luminance values with less than 5-6 bits/pixel
    - *human eyes are sensitive to contours*
- How to reduce contour effect at lower bits/pixel?
  - "Dithering": to break contours by adding noise before quantization

Use Dithering to Remove Contour Artifacts

31	31	32	33	33
29	30	31	32	32
28	30	31	31	33
27	28	29	30	31
27	27	28	29	30

original

24	24	40	40	40
24	24	24	40	40
24	24	24	24	40
24	24	24	24	24
24	24	24	24	24

artificial contour

quantized (step=16)

-1	+2	+3	-2	-1
+2	0	+2	-1	+2
-1	+3	-1	+1	-1
0	+2	-1	+2	0
-1	-3	+2	-1	-3

noise pattern

30	33	35	31	32
31	30	33	31	34
27	33	30	32	32
27	30	28	32	31
26	24	30	28	27

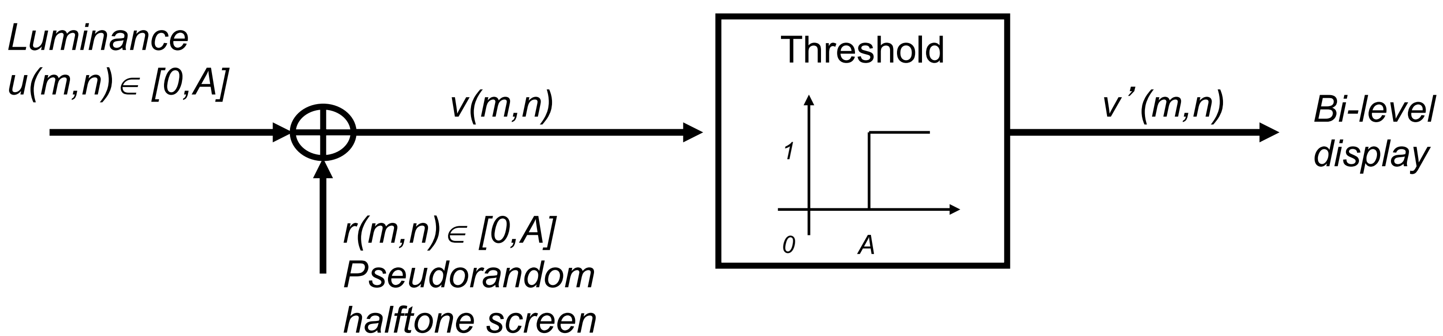
original + noise

24	40	40	24	40
24	24	40	24	40
24	40	24	40	40
24	24	24	40	24
24	24	24	24	24

quantized version of (original + noise)

Dithering for Halftone Images

- Halftone images
  - Extreme case of low bits/pixel: use two colors (b/w) to represent gray shades
- Extend the idea of pseudorandom noise quantizer
  - Upsample each pixel to form a resolution cell
  - Form dither signal by tiling halftone screen w/ same size as the image
  - Apply quantizer on the sum of dither signal and image signal
- Perceived gray level
  - Equal to the density of black dots perceived in one resolution cell



Examples of Halftone Images

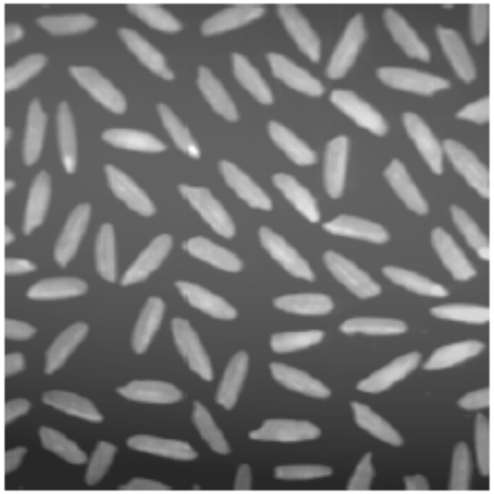
- Trade spatial resolution with pixel depth
  - Let eyes serve as a low-pass-filter
- Other approaches
  - Error diffusion
- Applications
  - Photo printing in newspaper and books
  - Image display in the old black/white monitor
  - Often use a periodic "dither screen" as noise pattern

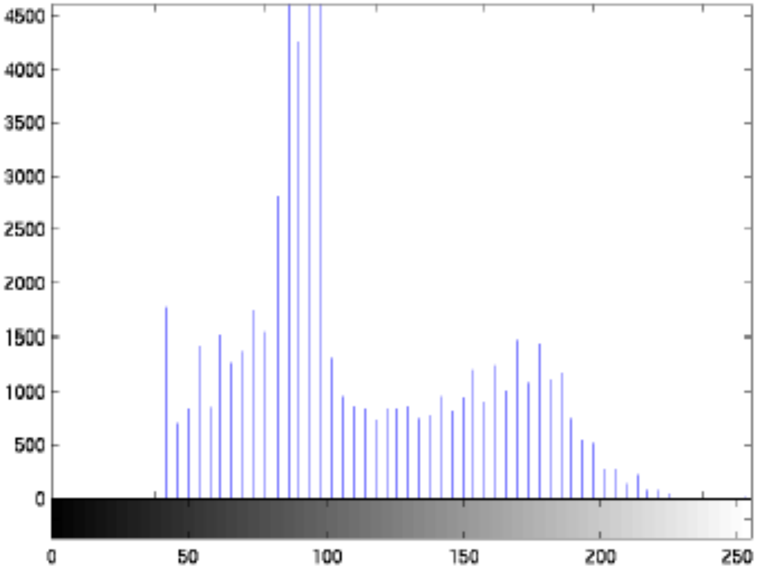
Pixel Statistics: Luminance Histogram

- For each gray level, count the # of pixels having that level
- Can group nearby levels to form a bin & count # pixels in it
- Measuring the distribution of gray levels in an image

```
I = imread('rice.tif');
imshow(I)
figure, imhist(I,64)
```

(From Matlab Image Toolbox Guide Fig.10-4)





Luminance Histogram (cont'd)

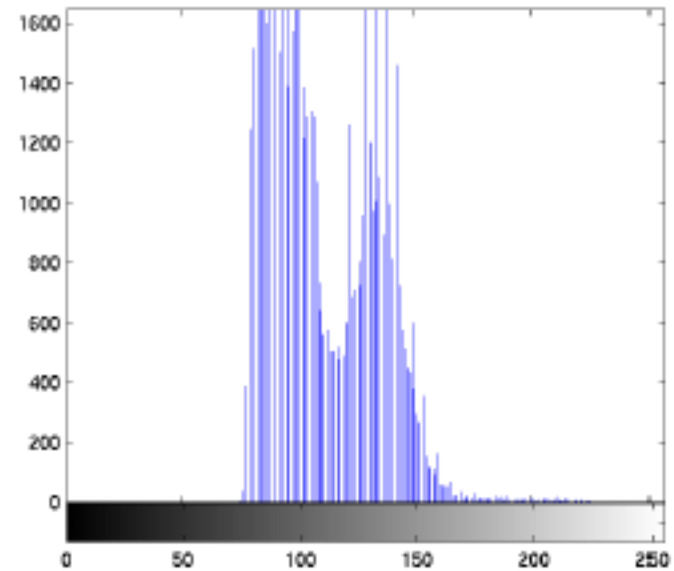
- Interpretation
  - Treat pixel values as instantiations of a random variable
  - Histogram is an estimate of the p.d.f.
- "Unbalanced" histogram doesn't fully utilize the dynamic range
  - Low contrast image ~ histogram concentrating in a narrow lum. range
  - Dark under-exposed image ~ histogram concentrating on the dark side
  - Bright over-exposed image ~ histogram concentrating on bright side

(Note: the effects of under- and over- expose depend on the film/sensor's characteristics.)

- Balanced histogram gives more pleasant look and reveals rich content

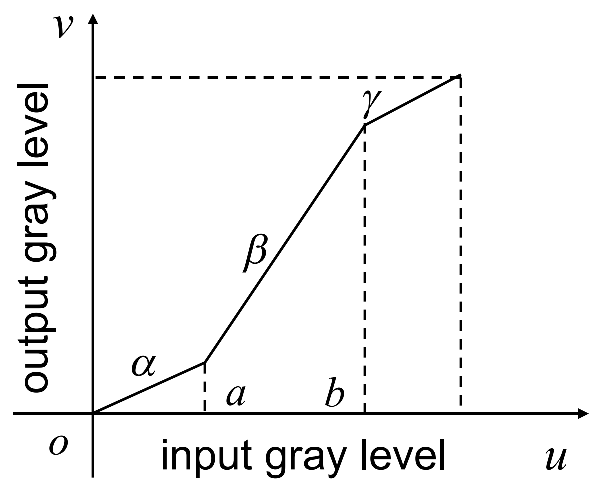
Image with Unbalanced Histogram

(From Matlab Image Toolbox Guide Fig.10-10 & 10-11)



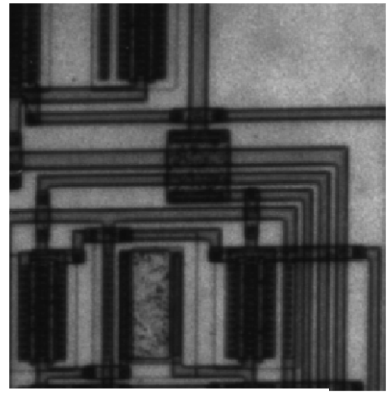
Contrast Stretching for Low-Contrast Images

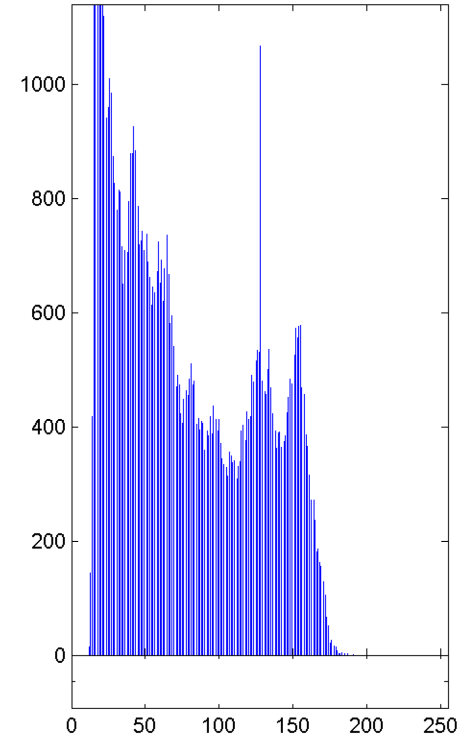
- Stretch the over-concentrated graylevels in histogram via a nonlinear mapping
  - Piece-wise linear stretching function
  - Assign slopes of the stretching region to be greater than 1



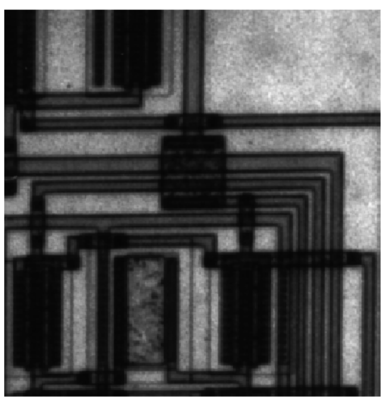
Contrast Stretching: Example

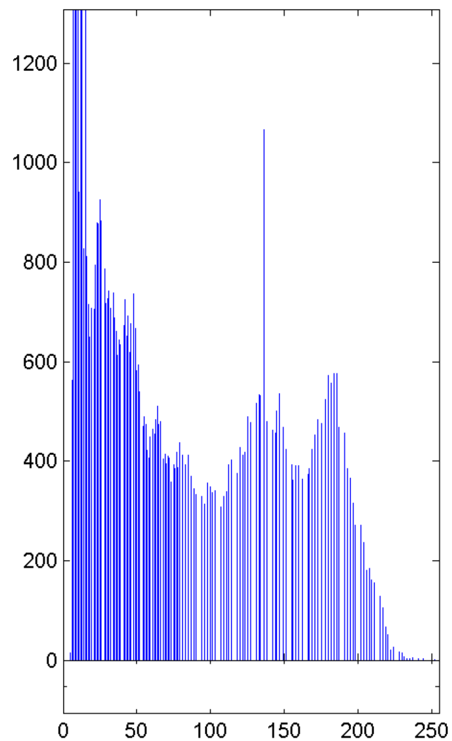
original





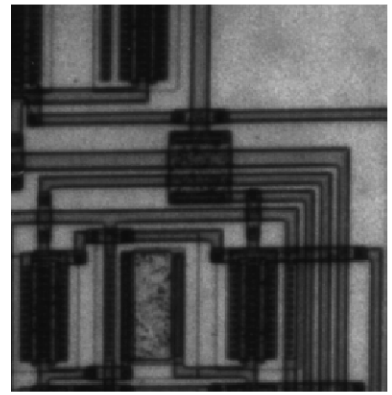
stretched

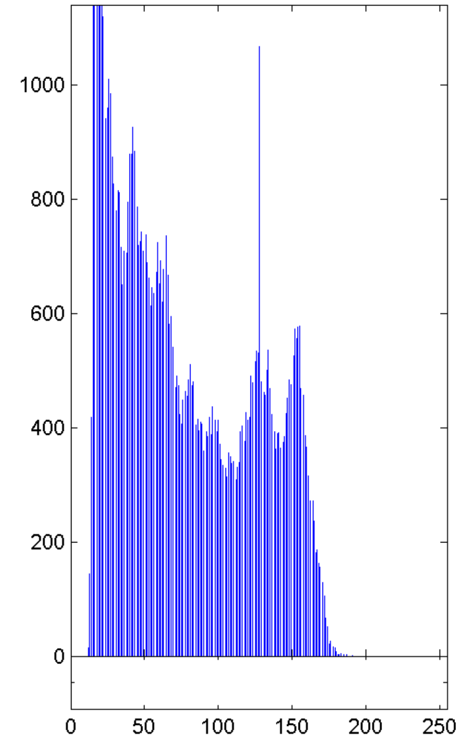




Histogram Equalization

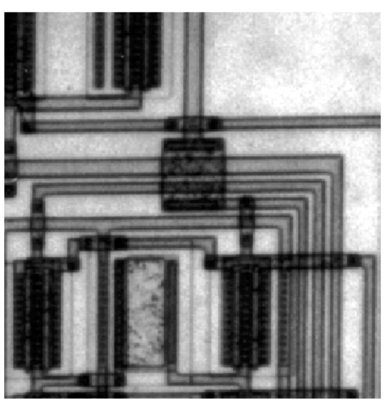
- Map the luminance of each pixel to a new value
  so that the output image has approximately uniform distribution of gray levels

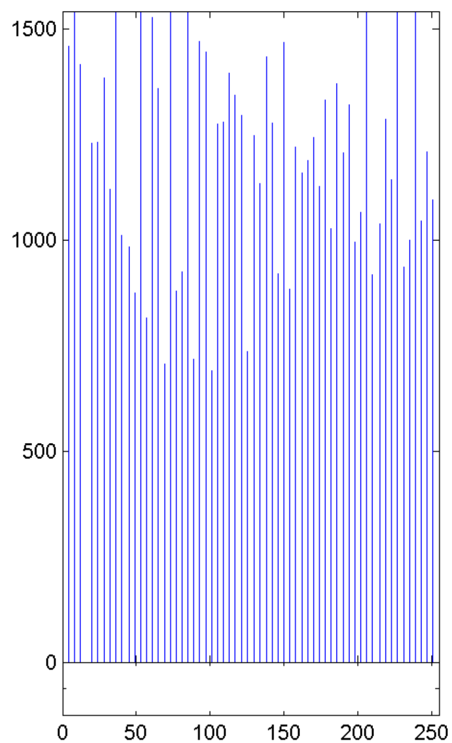




original

equalized





- No splitting

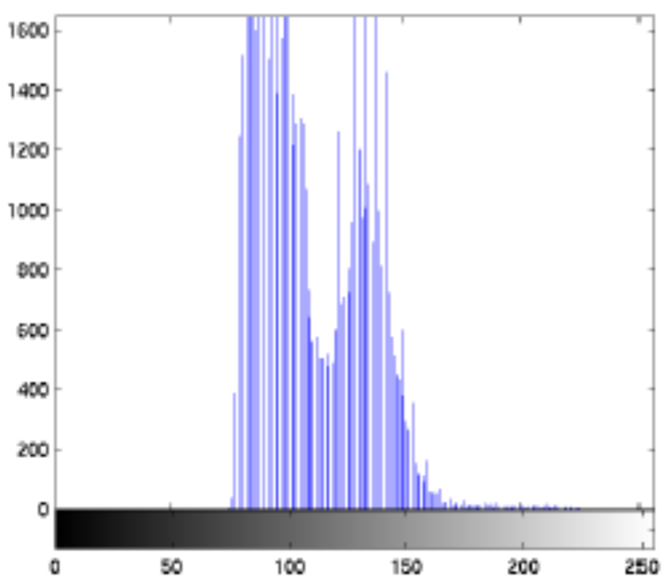
  if $X_1 = X_2 \rightarrow X_1' = X_2'$
  all pixels in one histogram bin are mapped to a new bin

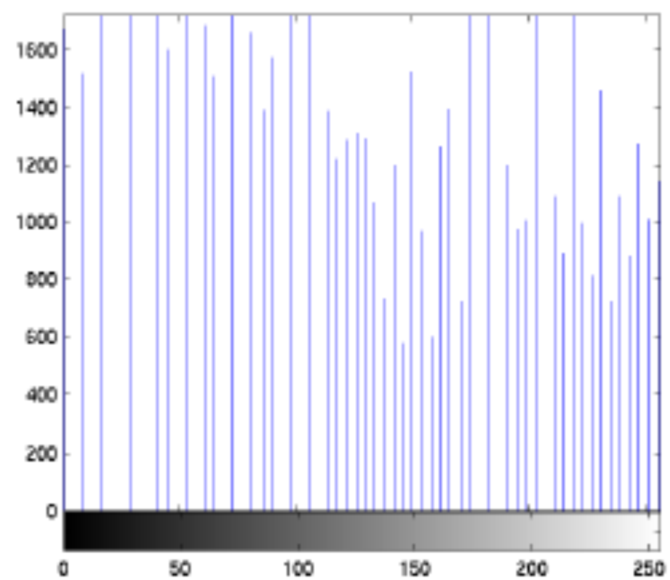
- Preserve the orders:

  if $X_1 < X_2 \rightarrow$ the new $X_1' \leq X_2'$

Examples of Histogram Equalization (cont'd)

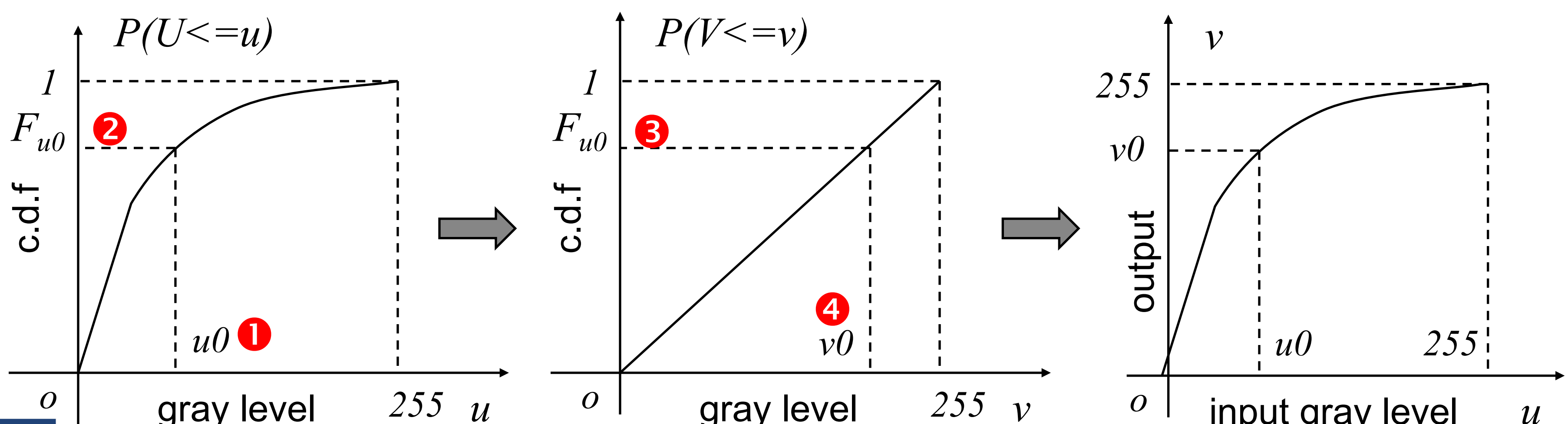
(From Matlab Image Toolbox Guide Fig.10-10 & 10-11)





How to Do Histogram Equalization?

- Basic idea:
  - Map the luminance of each pixel u to a new value v so that output image has approximately uniform distribution of gray levels
- Foundations from probability theory
  - **How to generate r.v. with desired distribution? => Match c.d.f.**
  - For u in discrete prob. distribution, the output v will be approximately uniform



Histogram Equalization: Math Foundations

- Basic idea
  - Map the luminance of each pixel to a new value so that the output image has approximately uniform distribution of gray levels
- Foundations from probability theory
  - **How to generate r.v. with desired distribution? => Match c.d.f !**
  - For r.v. U with continuous p.d.f. over [0,1], construct a new r.v. V that is monotonically increase with respect to (w.r.t.) U

$$V = F_U(u) = P[U \le u] = \int_0^u p_U(x)dx$$

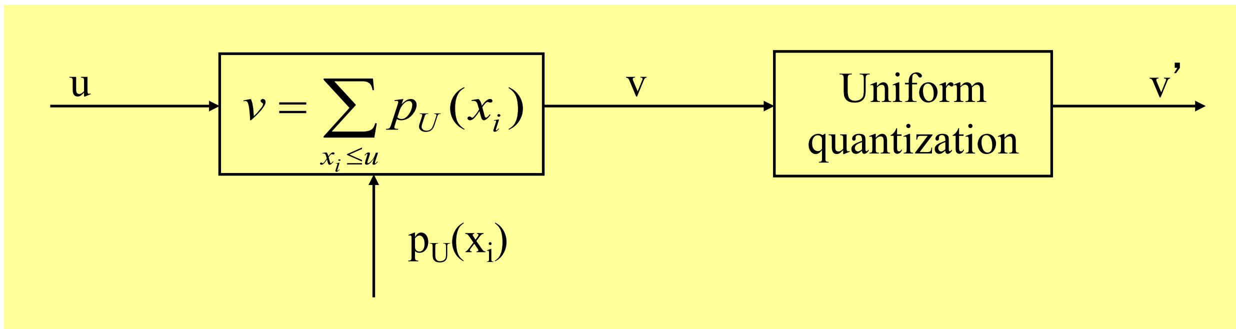
  - Random variable V is uniformly distributed over [0,1] ⇔ $F_V(v) = v$
    - $F_V(v) = P(V \le v) = P(\, F_U(u) \le v) = P(\, U \le F^{-1}{}_U(v)\,) = F_U(\, F^{-1}{}_U(v)\,) = v$
- Approach: map input luminance *u* to the corresponding *v*
  - for *u* in discrete prob. distribution, the output *v* will be approximately uniform

Drexel UNIVERSITY

Histogram Equalization: The Algorithm

- Approach: map input luminance *u* to the corresponding *v*

  for u in discrete prob. distribution, the output v will be approximately uniform



$$p_U(x_i) = \frac{h(x_i)}{\sum_{i=0}^{L-1} h(x_i)} \quad \text{for } i = 0, \ldots, L-1$$

$$v = \sum_{x_i \le u} p_U(x_i)$$

$v \in [0,1]$

$$v' = Round\left[\frac{v - v_{\min}}{1 - v_{\min}}(L-1)\right]$$

- Map discrete $v \in [0,1]$ to $v' \in \{0,\ldots,L-1\}$
- $v_{\min}$ is the smallest positive value of v
- Finally: v=1 is mapped to L-1, $v_{\min}$ is mapped to 0

2-D Fourier Transform and 2-D DFT

- 2-D Fourier Transform
  - Horizontal and vertical spatial frequencies

$$F(\zeta_x,\zeta_y) \triangleq FT[f(x,y)] \triangleq \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)\exp[-j2\pi(x\zeta_x + y\zeta_y)]dxdy,$$
$$= \int_{-\infty}^{+\infty}\left[\int_{-\infty}^{+\infty} f(x,y)\exp(-j2\pi x\zeta_x)dx\right]\exp(-j2\pi y\zeta_y)dy,$$

$$f(x,y) \triangleq FT^{-1}[F(\zeta_x,\zeta_y)] \triangleq \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} F(\zeta_x,\zeta_y)\exp[j2\pi(x\zeta_x + y\zeta_y)]d\zeta_x d\zeta_y.$$

- 2-D DFT

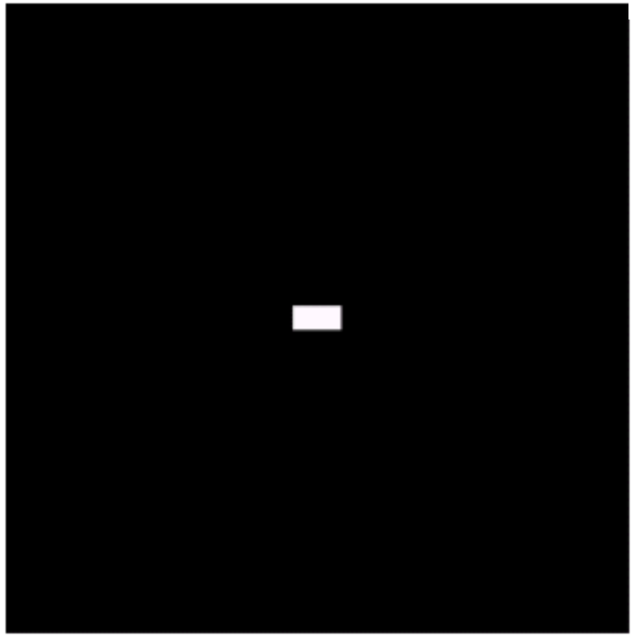
$$\begin{cases} Y(k,l) = \dfrac{1}{N}\displaystyle\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} X(m,n)\cdot W_N^{nl}\cdot W_N^{mk} \\ X(m,n) = \dfrac{1}{N}\displaystyle\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} Y(k,l)\cdot W_N^{-nl}\cdot W_N^{-mk} \end{cases}$$

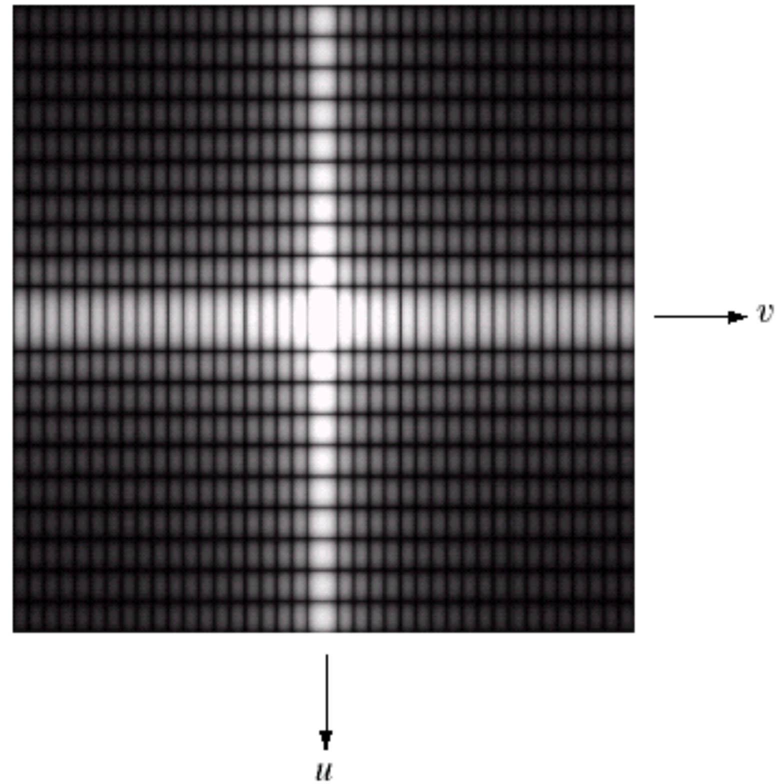
  - $W_N = exp\{-j2\pi / N\}$ complex conjugate of primitive Nth root of unity
  - Circular convolution in one domain ~ multiplication in another domain

Examples of 2-D DFT

a b

FIGURE 4.3
(a) Image of a 20×40 white rectangle on a black background of size 512×512 pixels.
(b) Centered Fourier spectrum shown after application of the log transformation given in Eq. (3.2-2 Compare Fig. 4.2.





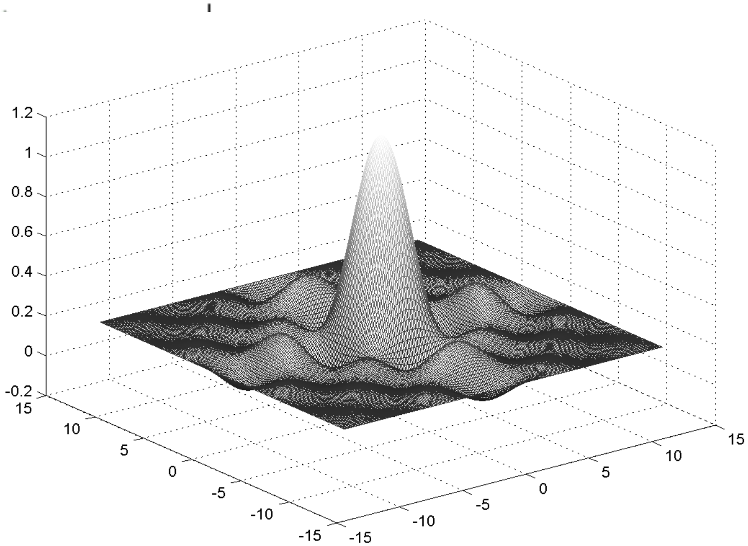
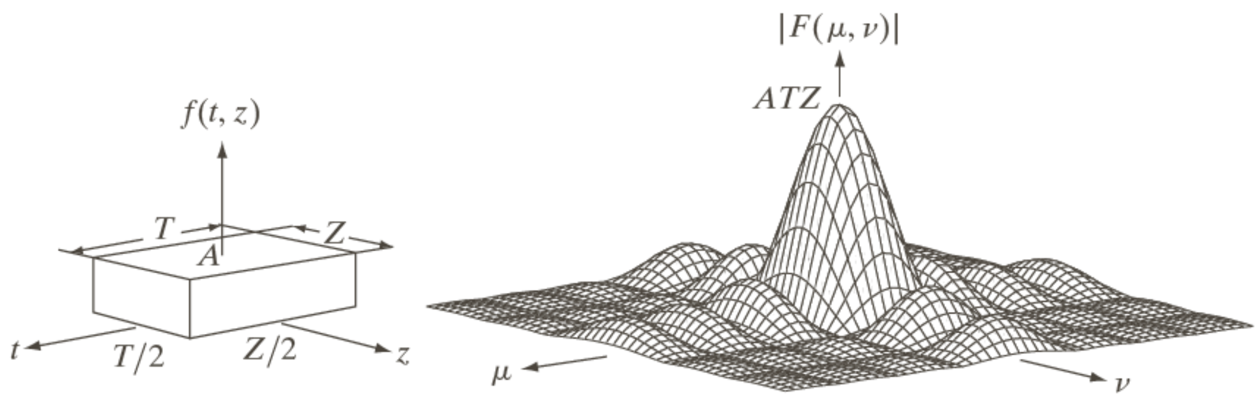


Image examples are from Gonzalez-Woods 2/e online slides.
2-D sinc function plots from B. Liu Princeton EE488 F'06.

2-D Box Function and its FT



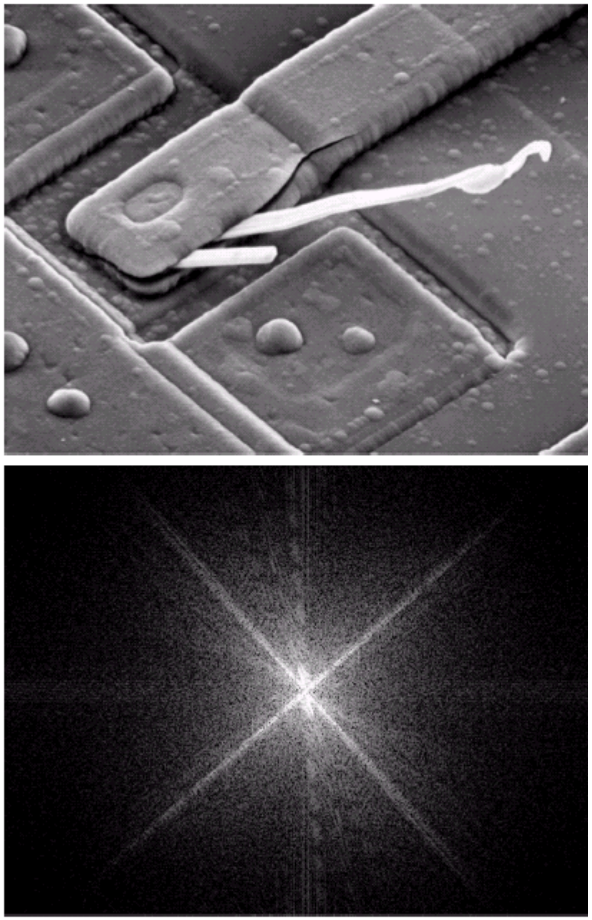
a b

FIGURE 4.13 (a) A 2-D function, and (b) a section of its spectrum (not to scale). The block is longer along the t-axis, so the spectrum is more "contracted" along the μ-axis. Compare with Fig. 4.4.

$$F[\, u, v \,] = ATZ\, [\sin(\pi uT) / (\pi uT)]\, [\sin(\pi vZ) / (\pi vZ)]$$

Image examples are from Gonzalez-Woods 3/e online slides.

Frequency Interpretation of 2-D Linear Filtering



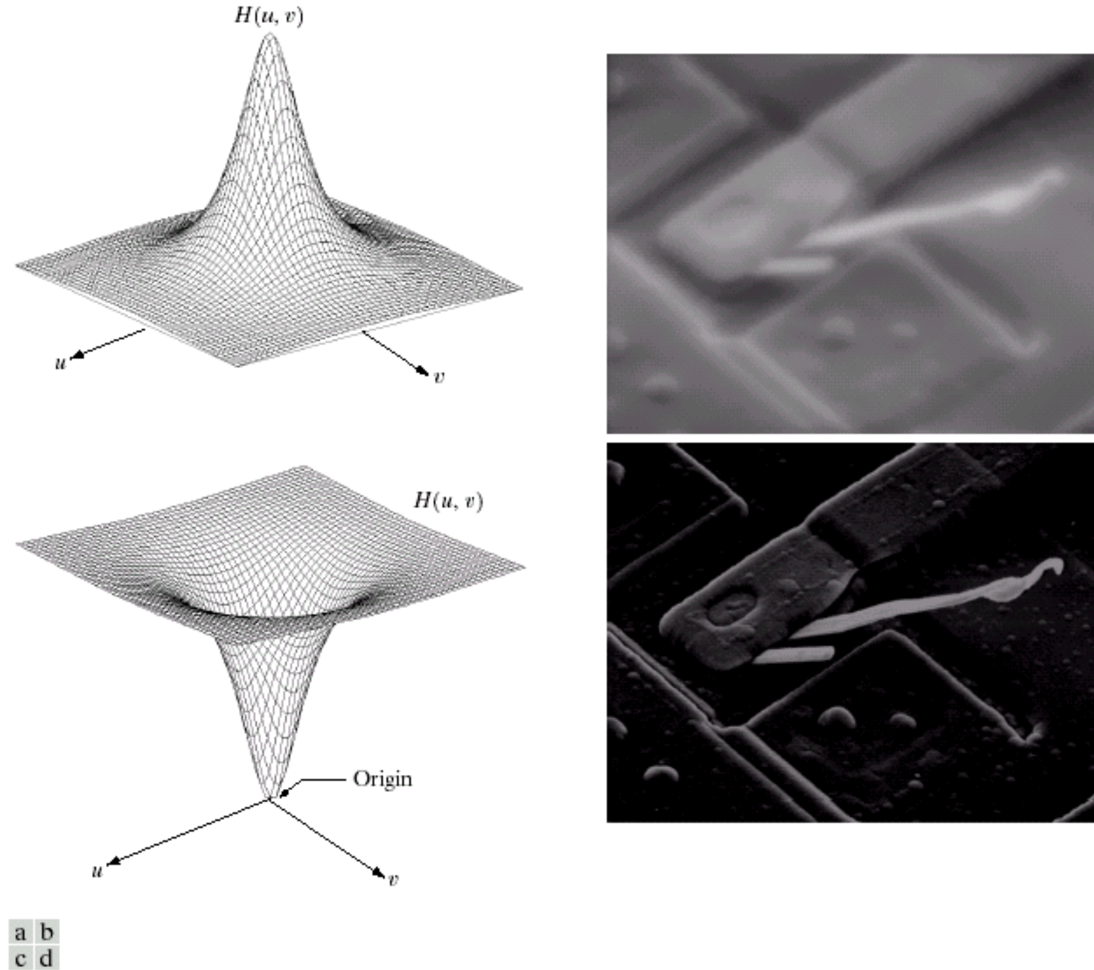


FIGURE 4.7 (a) A two-dimensional lowpass filter function. (b) Result of lowpass filtering the image in Fig. 4.4(a). (c) A two-dimensional highpass filter function. (d) Result of highpass filtering the image in Fig. 4.4(a).

Image examples are from Gonzalez-Woods 2/e online slides Fig.4.4 & 4.7.

Spatial Operations with 2-D FIR Filter

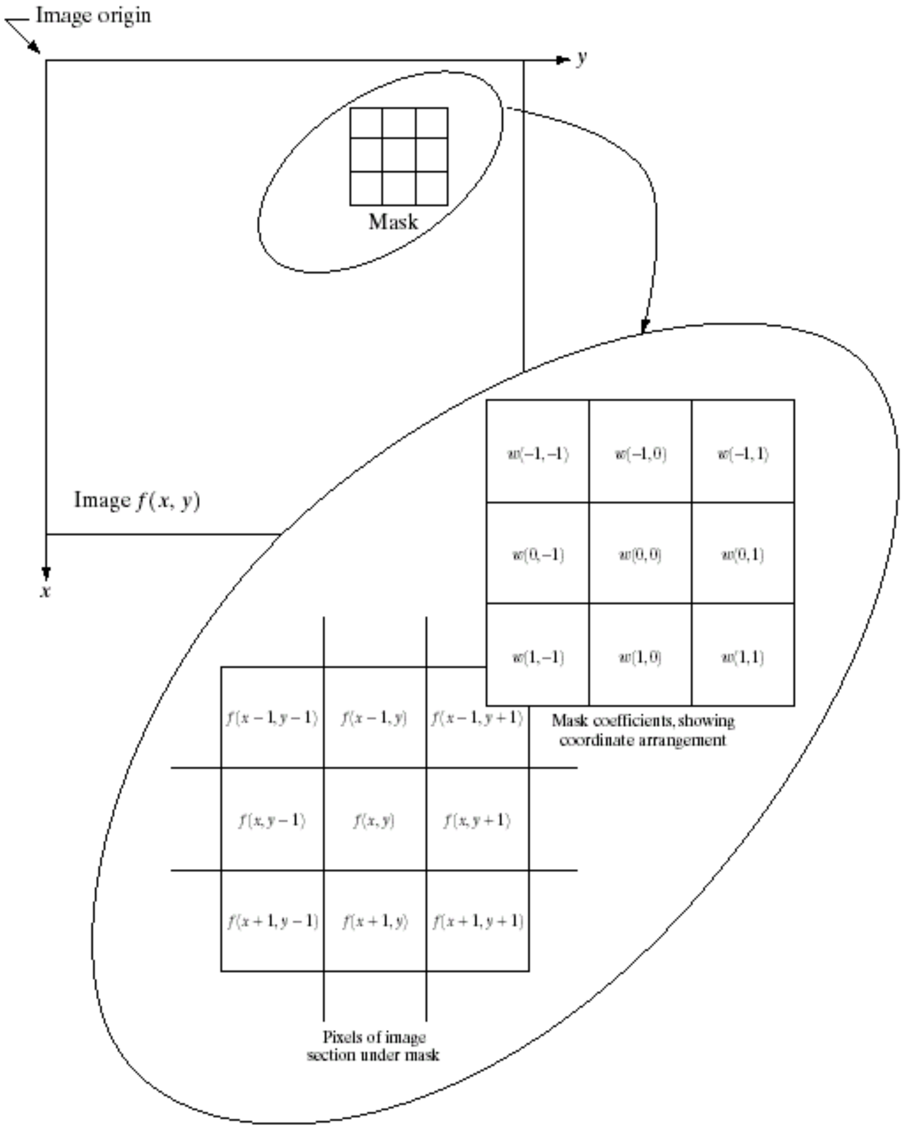


FIGURE 3.32 The mechanics of spatial filtering. The magnified drawing shows a 3×3 mask and the image section directly under it; the image section is shown displaced out from under the mask for ease of readability.

Image examples are from Gonzalez-Woods 2/e online slides Fig.3.32. Note this text defines mask as the mirrored version of the filter impulse response.

– Convolve 2-D Finite Impulse Response filter with image

$$g(m,n) = \Sigma f(m-k, n-l)\, h(k,l) = \Sigma f(m+k, n+l)\, h(-k,-l)$$

mirror w.r.t. origin, then shift & sum up

- *Typically has small support 2x2, 3x3, 5x5*

– Freq. domain interpretation

- *Multiply DFT(image) with DFT(filter)*

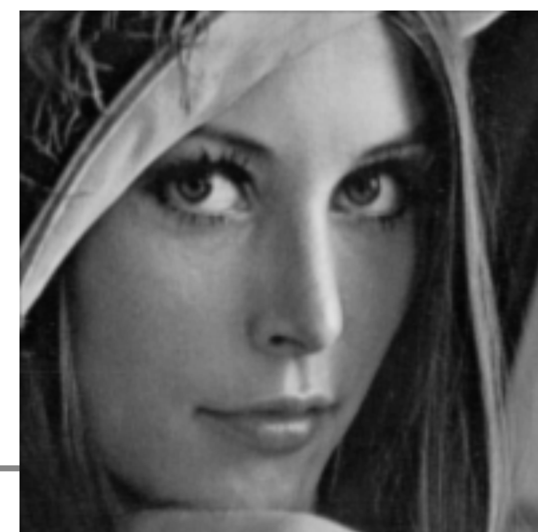
Spatial Averaging Masks

For softening and noise removal, etc.

	0	1
0	1/4	1/4
1	1/4	1/4

	-1	0	1
-1	1/9	1/9	1/9
0	1/9	1/9	1/9
1	1/9	1/9	1/9

	-1	0	1
-1	0	1/8	0
0	1/8	1/2	1/8
1	0	1/8	0



Suppressing Noise via Spatial Averaging

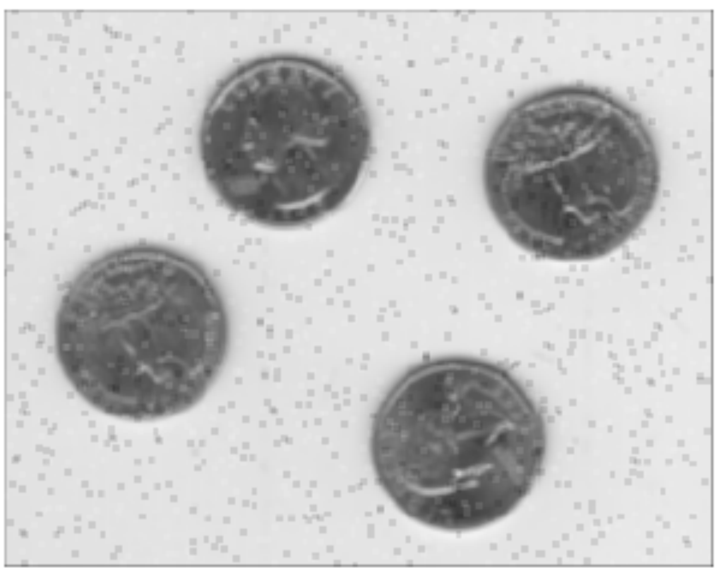
- Image with iid noise $y(m,n) = x(m,n) + N(m,n)$
- Averaged version

$$v(m,n) = (1/N_w) \Sigma\, x(m-k, n-l) + (1/N_w) \Sigma\, N(m-k, n-l)$$

- Noise variance reduced by a factor of N_w
  - N_w ~ # of pixels in the averaging window
- SNR improved by a factor of N_w if $x(m,n)$ is constant in local window
- Window size is limited to avoid blurring

Examples of Median Filtering

(From Matlab Image Toolbox Guide Fig.10-12 & 10-13)



Averaging Filter

Median Filtering

- Salt-and-Pepper noise
  - Isolated white/black pixels spread randomly over the image
  - Spatial averaging filter may incur blurred output
- Median filtering
  - Take median value over a small window as output ~ nonlinear
    - $Median\{ x(m) + y(m) \} \neq Median\{x(m)\} + Median\{y(m)\}$
  - Odd window size is commonly used
    - *3x3, 5x5, 7x7*
    - *5-pixel "+" shaped window*
  - Even-sized window ~ take the average of two middle values as output

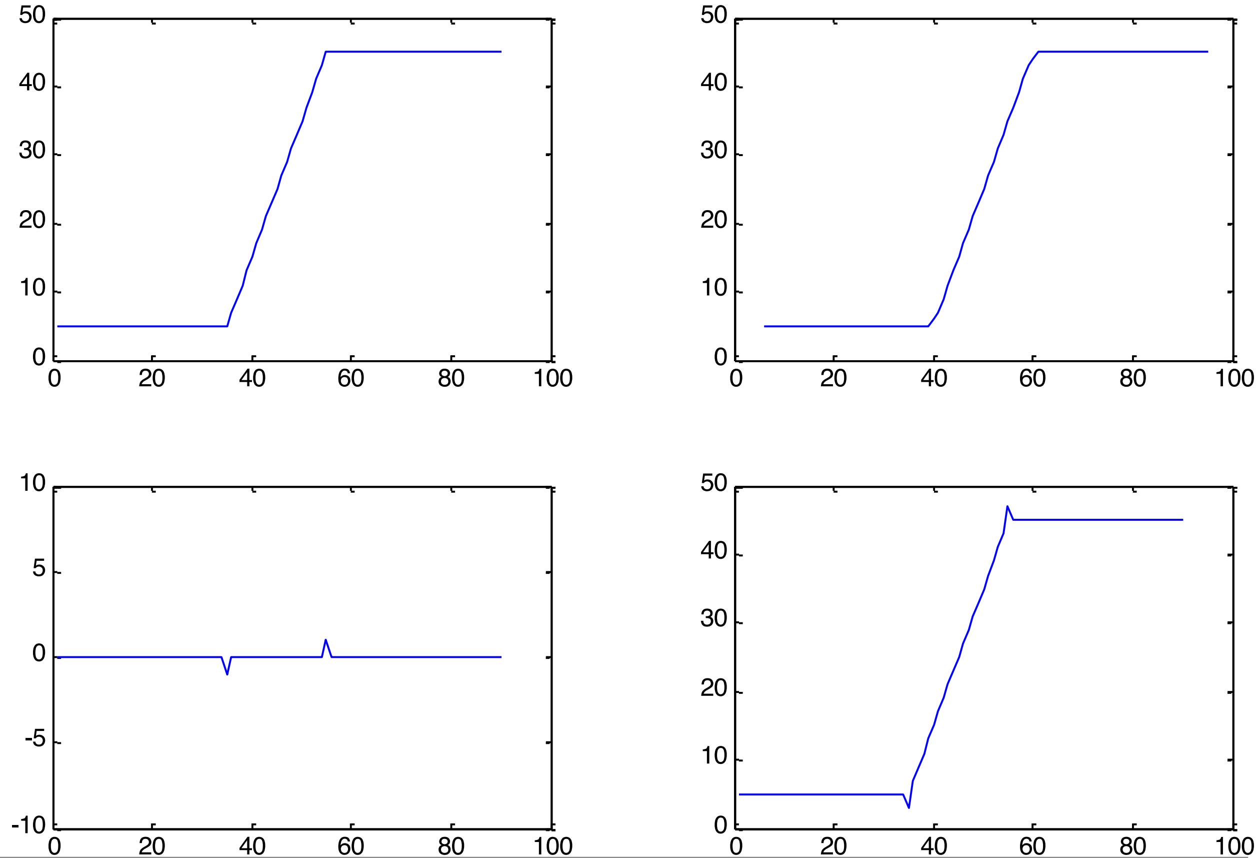
Image Sharpening

- Use LPF to generate HPF
  - Subtract a low pass filtered result from the original signal
  - HPF extracts edges and transitions
- Enhance edges

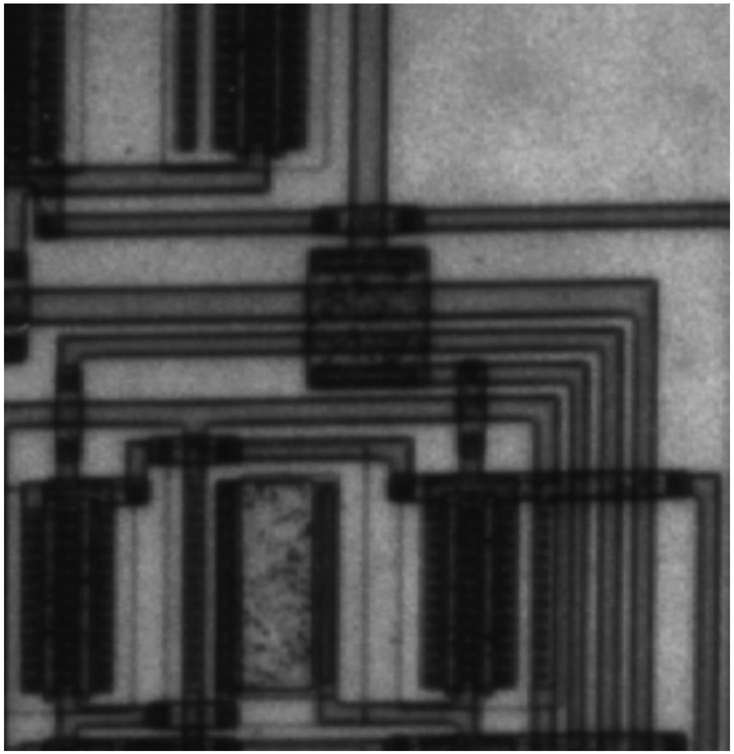
$$I_O \rightarrow I_{LP}$$

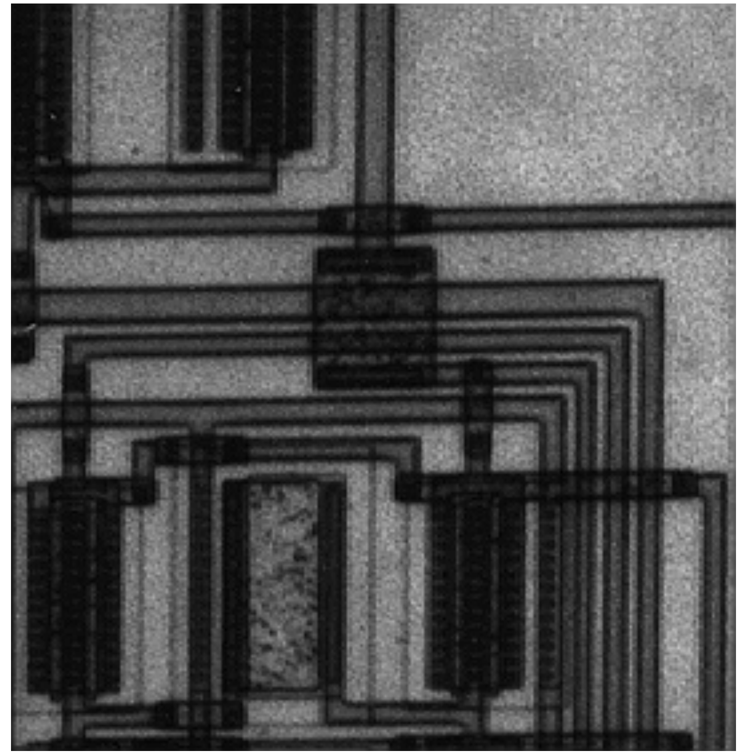
$$\rightarrow I_{HP} = I_O - I_{LP}$$

$$\rightarrow I_1 = I_O + a^* I_{HP}$$



Example of Image Sharpening





- $v(m,n) = u(m,n) + a * g(m,n)$
- Often use Laplacian operator to obtain $g(m,n)$

	-1	0	1
-1	0	-¼	0
0	-¼	1	-¼
1	0	-¼	0

Original moon image is from Matlab Image Toolbox.

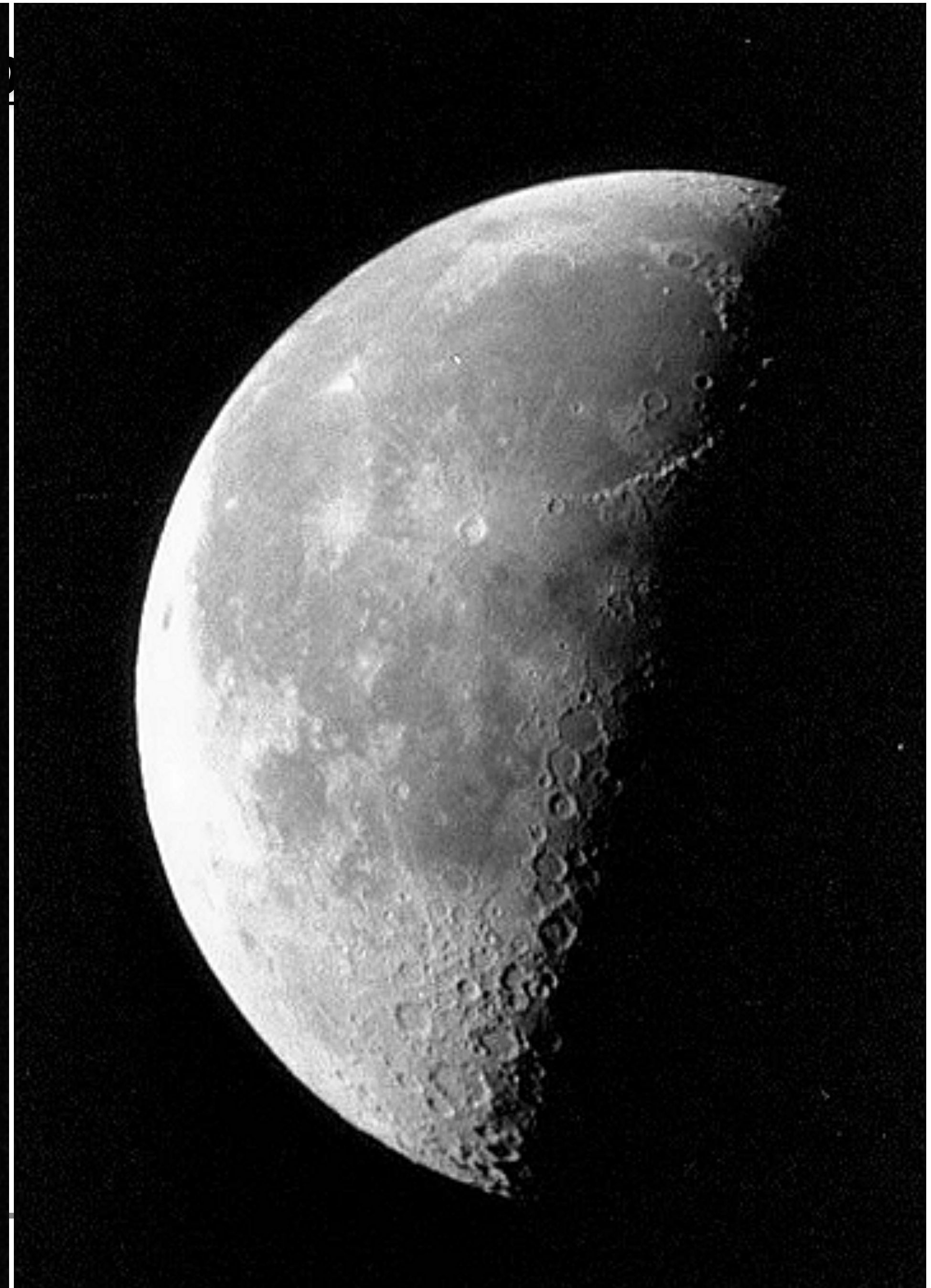


Image example is from Gonzalez-Woods 2/e online slides.

Other Variations of Image Sharpening

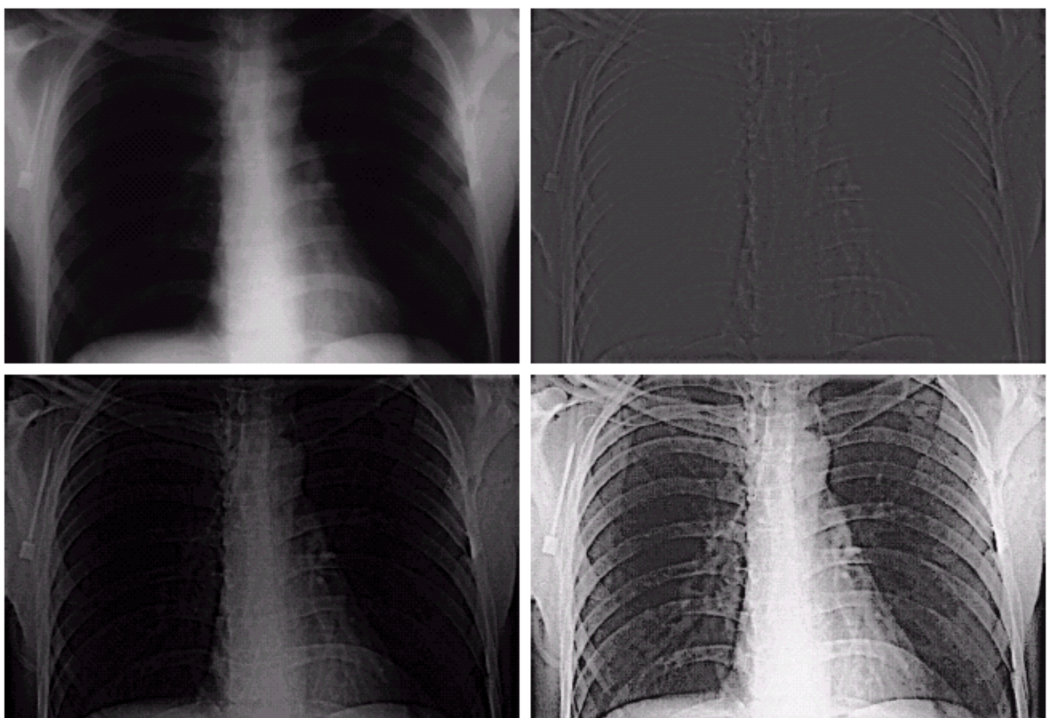
- High boost filter (Gonzalez-Woods 2/e pp132 & pp188)

  $$I_O \rightarrow I_{LP} \rightarrow I_{HP} = I_O - I_{LP} \rightarrow I_1 = (b-1)\, I_O + I_{HP}$$

  – Equiv. to high pass filtering for b=1

  – Expand or suppress original image pixel values when b≠2

- Combine sharpening with histogram equalization



a b
c d

FIGURE 4.30 (a) A chest X-ray image. (b) Result of Butterworth highpass filtering. (c) Result of high-frequency emphasis filtering. (d) Result of performing histogram equalization on (c). (Original image courtesy Dr. Thomas R. Gest, Division of Anatomical Sciences, University of Michigan Medical School.)

Spatial LPF, HPF, & BPF

- HPF and BPF can be constructed from LPF
- Low-pass filter
  - Useful in noise smoothing and downsampling/upsampling
- High-pass filter
  - $h_{HP}(m,n) = \delta(m,n) - h_{LP}(m,n)$
  - Useful in edge extraction and image sharpening
- Band-pass filter
  - $h_{BP}(m,n) = h_{L2}(m,n) - h_{L1}(m,n)$
  - Useful in edge enhancement
  - Also good for high-pass tasks in the presence of noise
    - *avoid amplifying high-frequency noise*