Image Processing and Visual Communications

Two-Dimensional Discrete Fourier Transform (2D-DFT)

Zhou Wang

Dept. of Electrical and Computer Engineering
University of Waterloo

Summary of FT, FS, DTFT/DSFT, DFS, DFT and FFT

Fourier Transform (FT):

$x(t)$ (continuous) $\rightleftarrows$ $X(\Omega)$ (continuous)

$$X(\Omega) = \int_{-\infty}^{\infty} x(t) e^{-j\Omega t} dt$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\Omega) e^{j\Omega t} d\Omega$$

Fourier Series (FS):

$x(t)$ (continuous, periodic) $\rightleftarrows$ X_m (discrete)

$$X_m = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-jm\Omega_0 t} dt$$

$$\Omega_0 = \frac{2\pi}{T}$$

$$x(t) = \sum_{m=-\infty}^{\infty} X_m e^{jm\Omega_0 t}$$

Discrete Time/Space Fourier Transform (DTFT/DSFT):

$x(n)$ (discrete) $\rightleftarrows$ $X(\omega)$ (continuous, periodic)

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n}$$

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) e^{j\omega n} d\omega$$

Summary of FT, FT, DTFT/DSFT, DFS, DFT and FFT

Discrete Fourier Series (DFS):

$$\tilde{X}_m = \sum_{n=0}^{N-1} \tilde{x}(n) e^{-j2\pi mn/N}$$

$\tilde{x}(n)$ (discrete, periodic) $\rightleftarrows$ $\tilde{X}_m$ (discrete, periodic)

$$\tilde{x}(n) = \frac{1}{N} \sum_{m=0}^{N-1} \tilde{X}_m e^{j2\pi mn/N}$$

Discrete Fourier Transform (DFT):

$$X(m) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi mn/N}$$

$x(n)$ (discrete, finite) $\rightleftarrows$ $X(m)$ (discrete, finite)

$$x(n) = \frac{1}{N} \sum_{m=0}^{N-1} X(m) e^{j2\pi mn/N}$$

Fast Fourier Transform (FFT): Fast algorithm for computing DFT

2D DFT and Inverse DFT (IDFT)

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \exp\left[-2\pi i \left(\frac{xu}{M} + \frac{yv}{N}\right)\right]$$

$$f(x,y) \rightleftarrows F(u,v)$$

$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) \exp\left[2\pi i \left(\frac{xu}{M} + \frac{yv}{N}\right)\right]$$

M, N: image size

x, y: image pixel position

u, v: spatial frequency

often used short notation:

$$W_N = e^{-j2\pi/N}$$

The Meaning of DFT and Spatial Frequencies

- **Important Concept**

  Any signal can be represented as a **linear combination** of a set of **basic components**

$$f(x,y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)\exp\left[2\pi i\left(\frac{xu}{M}+\frac{yv}{N}\right)\right]$$

  - **Fourier components**: sinusoidal patterns
  - **Fourier coefficients**: weighting factors assigned to the Fourier components

- **Spatial frequency:** The frequency of Fourier component

- **Not to be confused with electromagnetic frequencies** (e.g., the frequencies associated with the colors of light)

Real Part, Imaginary Part, Magnitude, Phase, Spectrum

Real part: $R = \mathrm{Real}(F)$

Imaginary part: $I = \mathrm{Imag}(F)$

Magnitude-phase representation:

$$F(u,v) = |F(u,v)|e^{-j\phi(u,v)}$$

Magnitude (spectrum):

$$|F(u,v)| = \left[R^2(u,v) + I^2(u,v)\right]^{1/2}$$

Phase (spectrum):

$$\phi(u,v) = \tan^{-1}\left[\frac{I(u,v)}{R(u,v)}\right]$$ for $R > 0$

Power Spectrum:

$$P(u,v) = |F(u,v)|^2$$

2D DFT Properties

Mean of image/ DC component:

$$\bar{f}(x,y) = F(0,0) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)$$

Highest frequency component:

$$f(x,y)(-1)^{x+y} \Leftrightarrow F(u - M/2, v - N/2)$$

"Half-shifted" Image:

$$f(x - M/2, y - N/2) \Leftrightarrow F(u,v)(-1)^{u+v}$$

Conjugate Symmetry:

$$F(u,v) = F^*(-u,-v)$$

Magnitude Symmetry:

$$|F(u,v)| = |F(-u,-v)|$$

2D DFT Properties

Spatial domain differentiation:

$$\frac{\partial^n f(x, y)}{\partial x^n} \Leftrightarrow (ju)^n F(u, v)$$

Frequency domain differentiation:

$$(-jx)^n f(x, y) \Leftrightarrow \frac{\partial^n F(u, v)}{\partial u^n}$$

Distribution law:

$$\Im[f_1(x, y) + f_2(x, y)] = \Im[f_1(x, y)] + \Im[f_2(x, y)]$$

Laplacian:

$$\nabla^2 f(x, y) \Leftrightarrow -(u^2 + v^2)F(u, v)$$

Spatial domain Periodicity:

$$f(x, y) = f(x + M, y) = f(x, y + N) = f(x + M, y + N)$$

Frequency domain periodicity:

$$F(u, v) = F(u + M, v) = F(u, v + N) = F(u + M, v + N)$$

Computation of 2D-DFT

Fourier transform matrices: remember $W_N = e^{-j2\pi/N}$

$$\mathbf{F}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N & W_N^2 & \cdots & W_N^{N-1} \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)^2} \end{bmatrix} \qquad \mathbf{F}_N^* = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \cdots & W_N^{1-N} \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ 1 & W_N^{1-N} & W_N^{2(1-N)} & \cdots & W_N^{-(N-1)^2} \end{bmatrix}$$

relationship: $\mathbf{F}_N^{-1} = \frac{1}{N}\mathbf{F}_N^*$

In particular, for $N = 4$:

$$\mathbf{F}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \qquad \mathbf{F}_4^* = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix}$$

Computation of 2D-DFT

- To compute the 1D-DFT of a 1D signal **x** (as a vector):

$$\tilde{\mathbf{x}} = \mathbf{F}_N \mathbf{x}$$

To compute the inverse 1D-DFT:

$$\mathbf{x} = \frac{1}{N} \mathbf{F}_N^* \tilde{\mathbf{x}}$$

- To compute the 2D-DFT of an image **X** (as a matrix):

$$\tilde{\mathbf{X}} = \mathbf{F}_N \mathbf{X} \mathbf{F}_N$$

To compute the inverse 2D-DFT:

$$\mathbf{X} = \frac{1}{N^2} \mathbf{F}_N^* \tilde{\mathbf{X}} \mathbf{F}_N^*$$

Computation of 2D-DFT: Example

- A 4x4 image

$$\mathbf{X} = \begin{bmatrix} 1 & 3 & 6 & 8 \\ 9 & 8 & 8 & 2 \\ 5 & 4 & 2 & 3 \\ 6 & 6 & 3 & 3 \end{bmatrix}$$

- Compute its 2D-DFT:

$$\tilde{\mathbf{X}} = \mathbf{F}_4 \mathbf{X} \mathbf{F}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 & 3 & 6 & 8 \\ 9 & 8 & 8 & 2 \\ 5 & 4 & 2 & 3 \\ 6 & 6 & 3 & 3 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

MATLAB function: *fft2*

$$= \begin{bmatrix} 21 & 21 & 19 & 16 \\ -4-3j & -1-2j & 4-5j & 5+j \\ -9 & -7 & -3 & 6 \\ -4+3j & -1+2j & 4+5j & 5-j \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

lowest frequency component

$$= \begin{bmatrix} 77 & 2-5j & 3 & 2+5j \\ 4-9j & -11+8j & -4-7j & -5-4j \\ -13 & -6+13j & -11 & -6-13j \\ 4+9j & -5+4j & -4+7j & -11-8j \end{bmatrix}$$

highest frequency component

Computation of 2D-DFT: Example

$$\tilde{\mathbf{X}} = \begin{bmatrix} 77 & 2-5j & 3 & 2+5j \\ 4-9j & -11+8j & -4-7j & -5-4j \\ -13 & -6+13j & -11 & -6-13j \\ 4+9j & -5+4j & -4+7j & -11-8j \end{bmatrix}$$

Real part:

$$\tilde{\mathbf{X}}_{real} = \begin{bmatrix} 77 & 2 & 3 & 2 \\ 4 & -11 & -4 & -5 \\ -13 & -6 & -11 & -6 \\ 4 & -5 & -4 & -11 \end{bmatrix}$$

Imaginary part:

$$\tilde{\mathbf{X}}_{imag} = \begin{bmatrix} 0 & -5 & 0 & 5 \\ -9 & 8 & -7 & -4 \\ 0 & 13 & 0 & -13 \\ 9 & 4 & 7 & -8 \end{bmatrix}$$

Magnitude:

$$\tilde{\mathbf{X}}_{magnitude} = \begin{bmatrix} 77 & 5.39 & 3 & 5.39 \\ 9.85 & 13.60 & 8.06 & 6.4 \\ 13 & 14.32 & 11 & 14.32 \\ 9.85 & 6.40 & 8.06 & 13.60 \end{bmatrix}$$

Phase:

$$\tilde{\mathbf{X}}_{phase} = \begin{bmatrix} 0 & -1.19 & 0 & 1.19 \\ -1.15 & 2.51 & -2.09 & -2.47 \\ 3.14 & 2.00 & 3.14 & -2.00 \\ 1.15 & 2.47 & 2.09 & -2.51 \end{bmatrix}$$

Computation of 2D-DFT: Example

- Compute the inverse 2D-DFT:

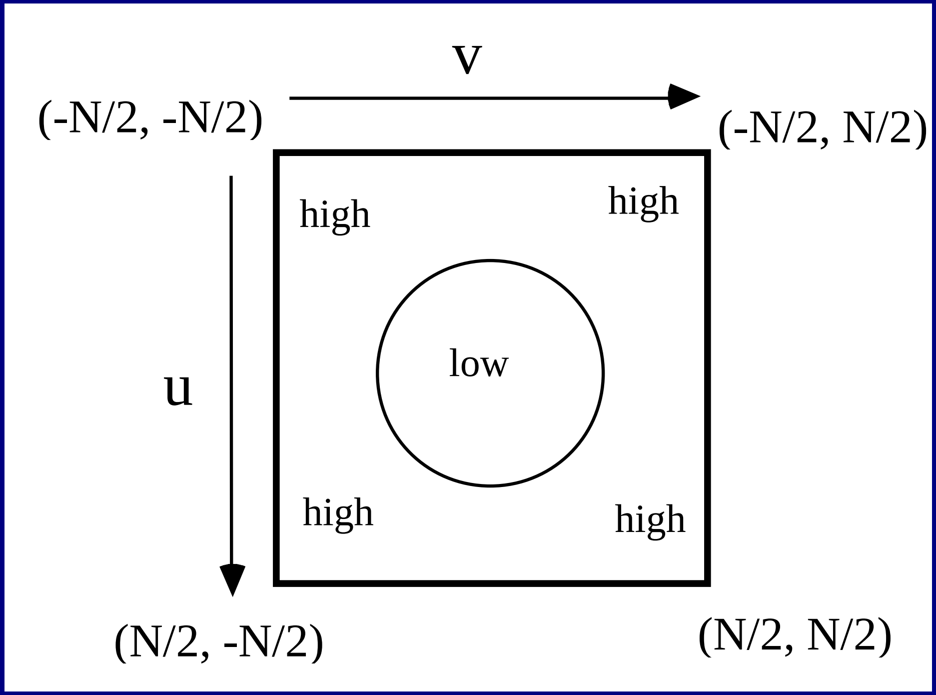
$$\mathbf{F}_4^*\tilde{\mathbf{X}}\mathbf{F}_4^* = \frac{1}{4^2}\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & j & -1 & -j\\ 1 & -1 & 1 & -1\\ 1 & -j & -1 & j\end{bmatrix}\begin{bmatrix}77 & 2-5j & 3 & 2+5j\\ 4-9j & -11+8j & -4-7j & -5-4j\\ -13 & -6+13j & -11 & -6-13j\\ 4+9j & -5+4j & -4+7j & -11-8j\end{bmatrix}\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & j & -1 & -j\\ 1 & -1 & 1 & -1\\ 1 & -j & -1 & j\end{bmatrix}$$

$$= \frac{1}{4}\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & j & -1 & -j\\ 1 & -1 & 1 & -1\\ 1 & -j & -1 & j\end{bmatrix}\begin{bmatrix}21 & 21 & 19 & 16\\ -4-3j & -1-2j & 4-5j & 5+j\\ -9 & -7 & -3 & 6\\ -4+3j & -1+2j & 4+5j & 5-j\end{bmatrix}$$

$$= \begin{bmatrix}1 & 3 & 6 & 8\\ 9 & 8 & 8 & 2\\ 5 & 4 & 2 & 3\\ 6 & 6 & 3 & 3\end{bmatrix} = \mathbf{X}$$

MATLAB function: *ifft2*

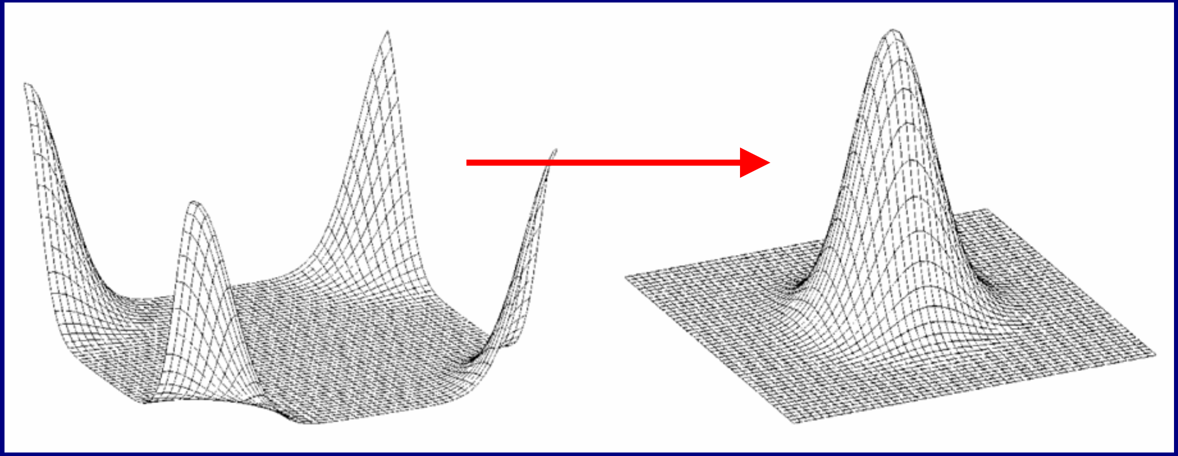
Centered Representation



MATLAB function: *fftshift*

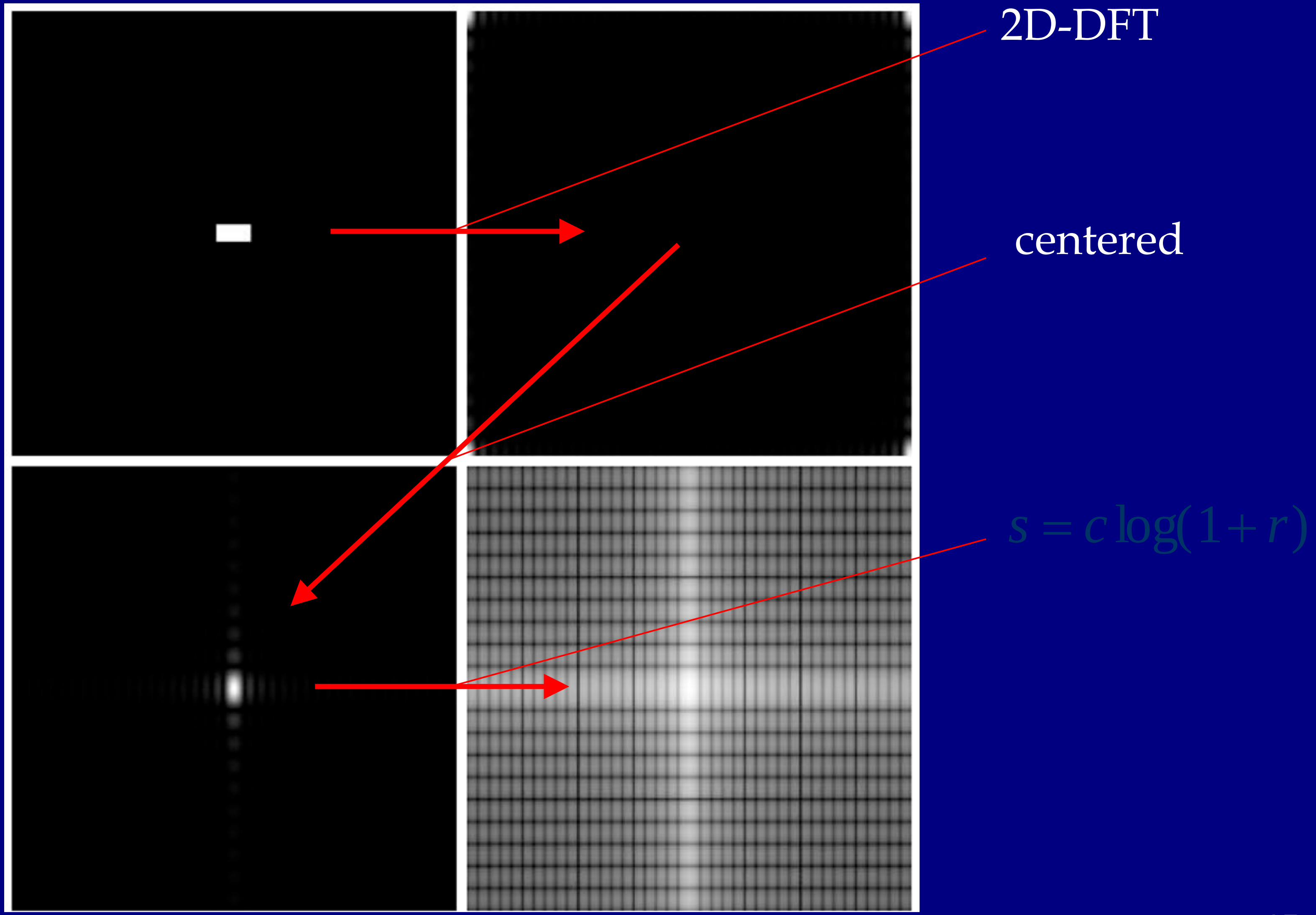
From Prof. Al Bovik

Example:



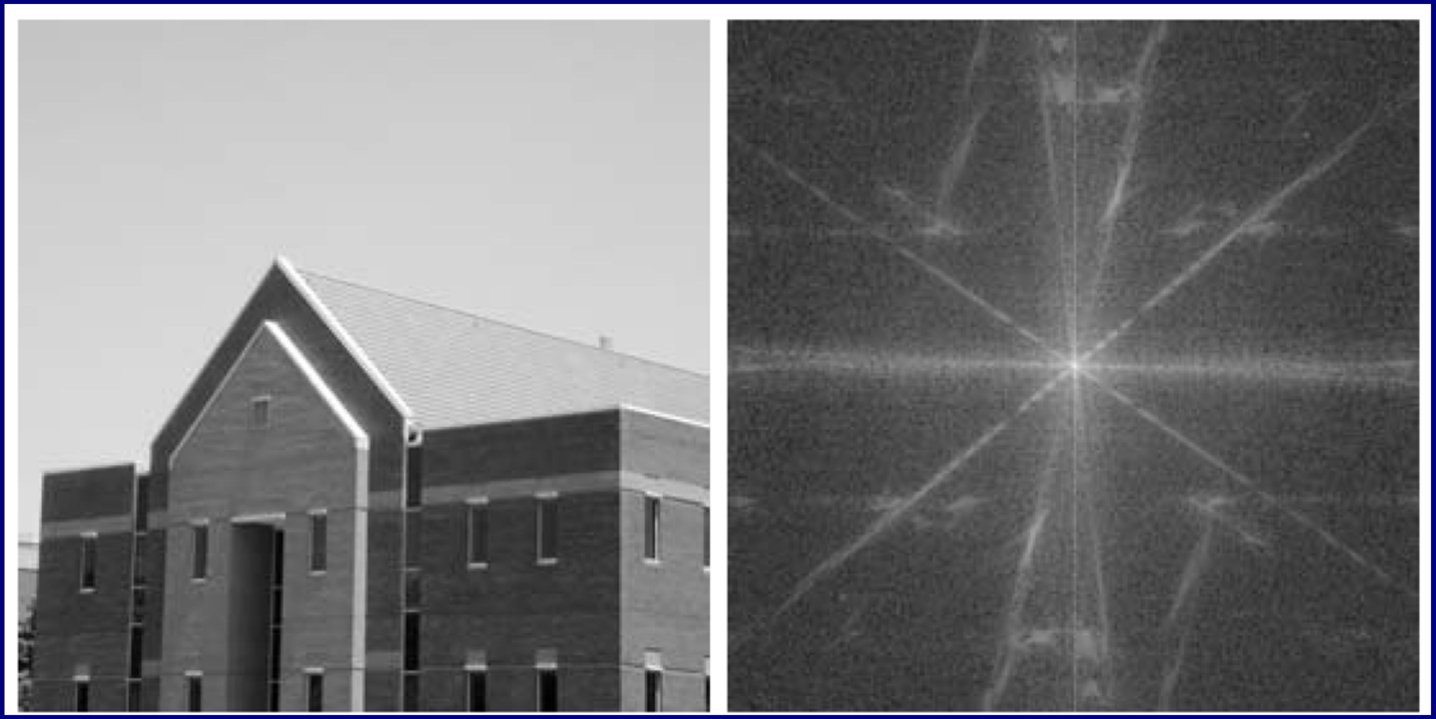
From [Gonzalez & Woods]

Log-Magnitude Visualization



From [Gonzalez & Woods]

Apply to Images



2D-DFT → centered → log intensity transformation

From [Gonzalez & Woods]

Convolution Theorem

$$f(x,y) \longrightarrow \boxed{h(x,y)} \longrightarrow g(x,y)$$

input image	impulse response (filter)	output image

$$g(x,y) = f(x,y) \otimes h(x,y)$$

DFT ↓↑ IDFT DFT ↓↑ IDFT DFT ↓↑ IDFT

$$G(u,v) = F(u,v)\,H(u,v)$$

Frequency Domain Filtering

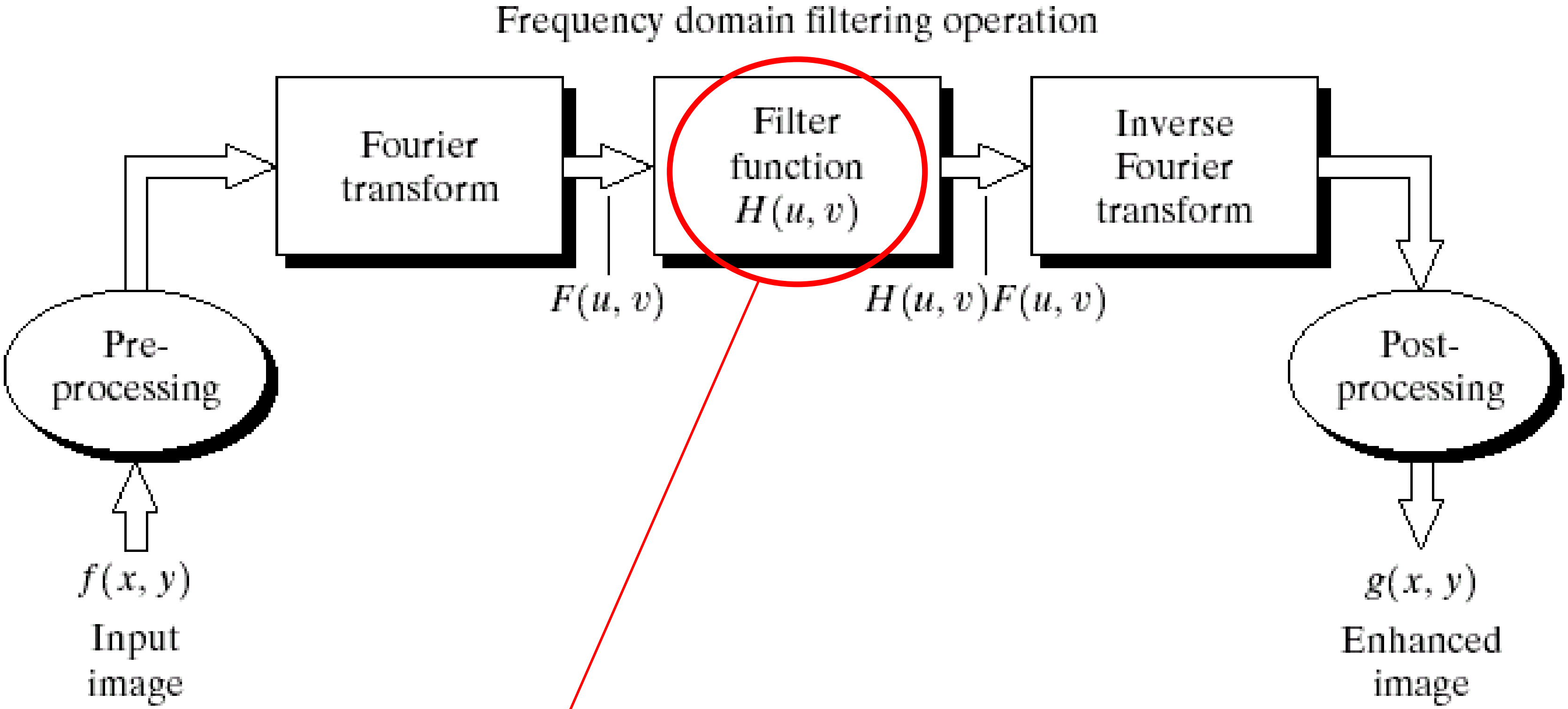


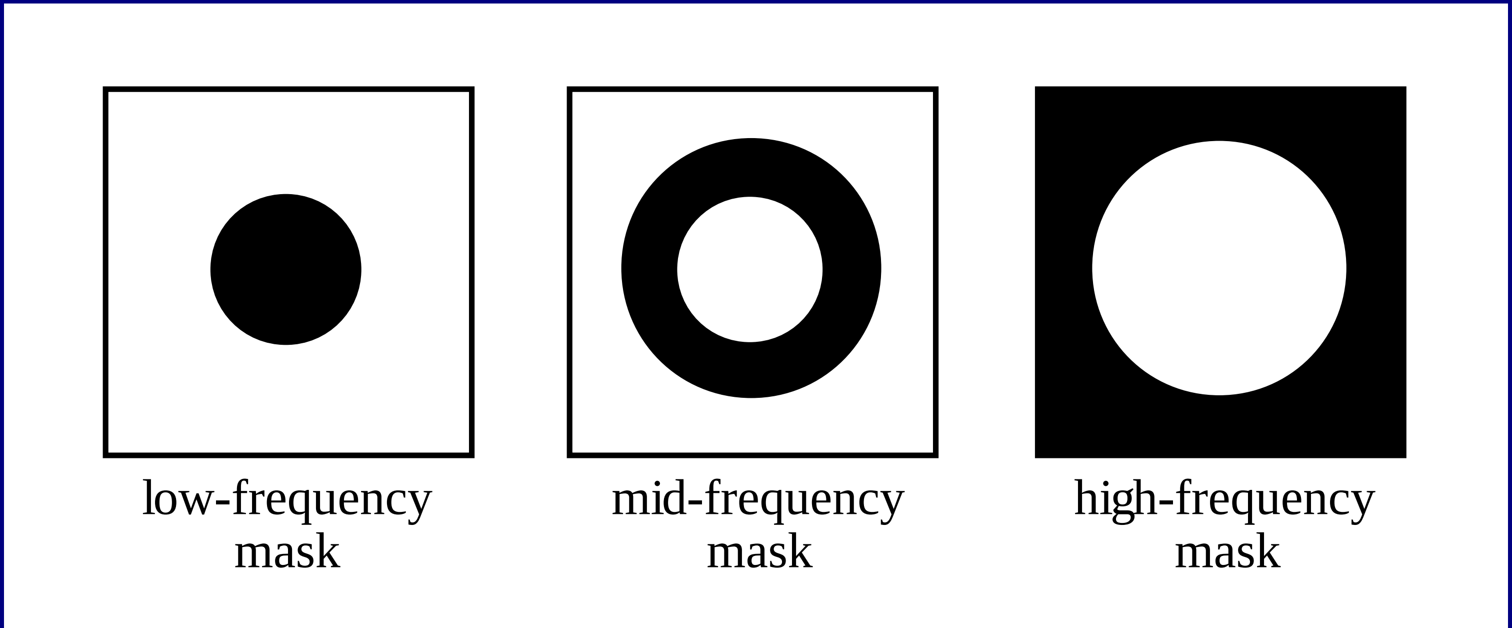
FIGURE 4.5 Basic steps for filtering in the frequency domain.

Filter design: design $H(u,v)$

From [Gonzalez & Woods]

2D-DFT Domain Filter Design

- Ideal lowpass, bandpass and highpass



From Prof. Al Bovik

2D-DFT Domain Filter Design

- Ideal lowpass, bandpass and highpass

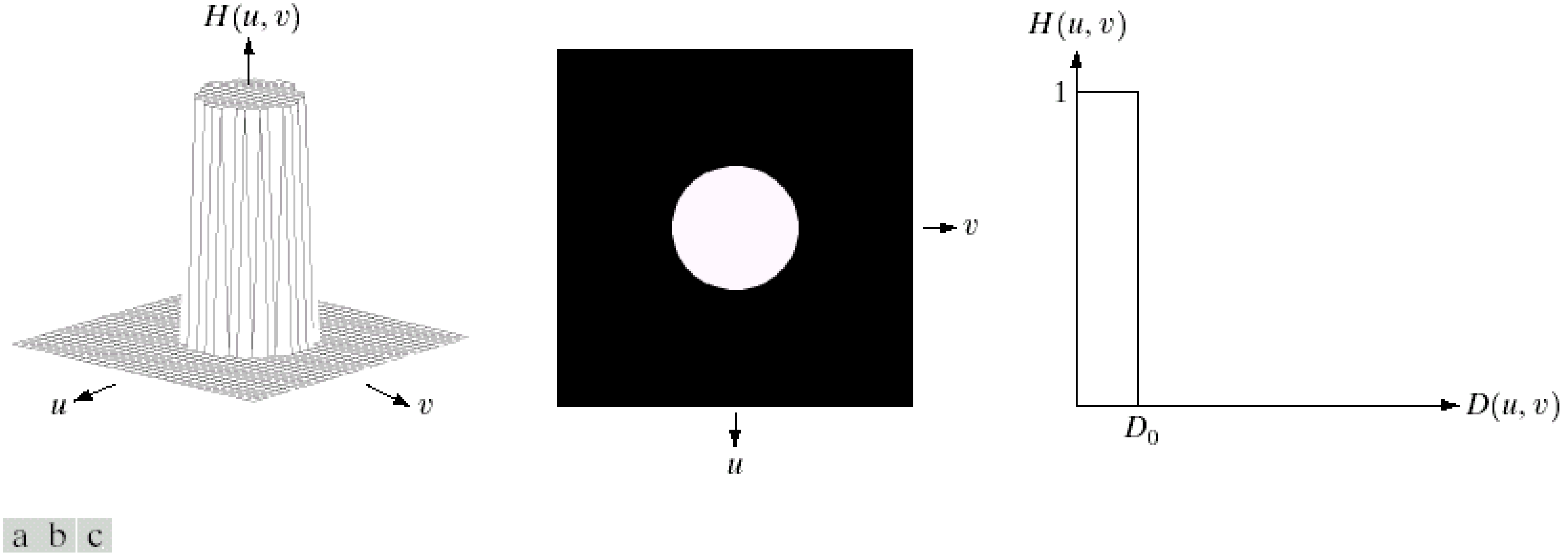
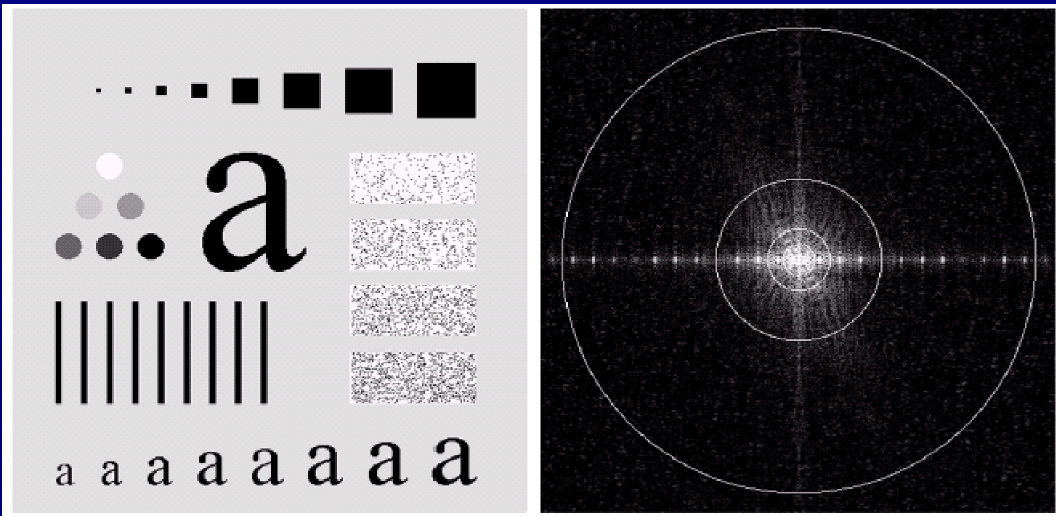


FIGURE 4.10 (a) Perspective plot of an ideal lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

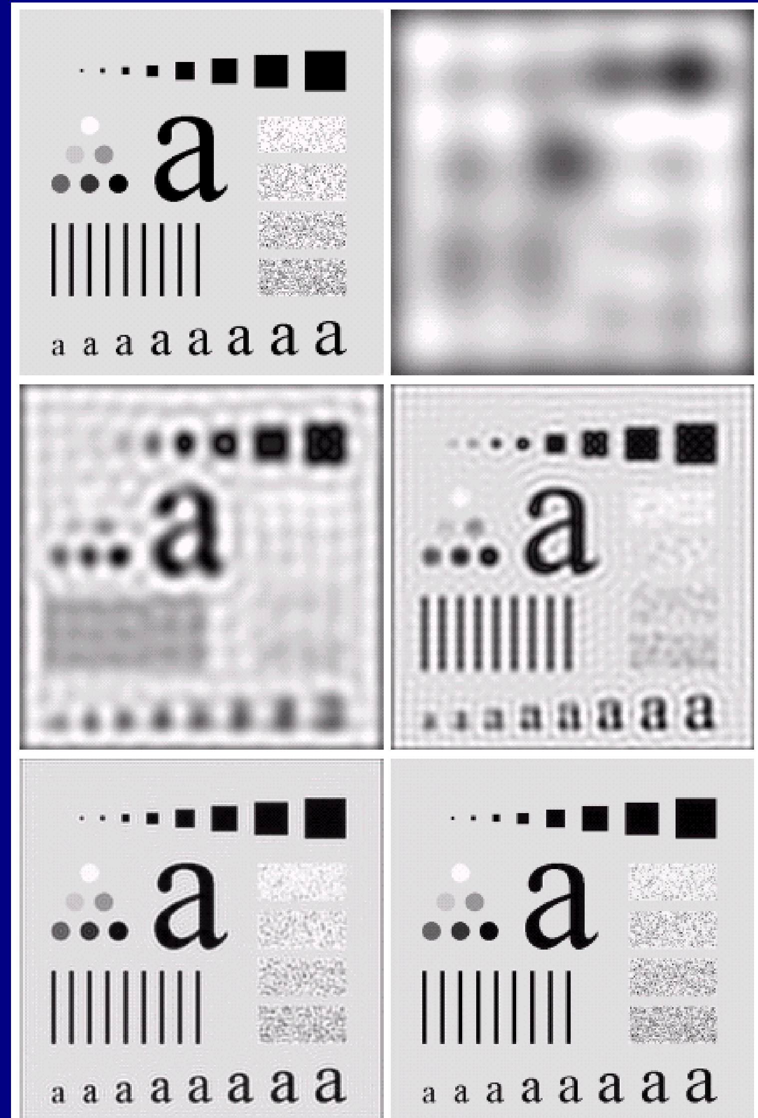
From [Gonzalez & Woods]

2D-DFT Domain Filter Design



a b

FIGURE 4.11 (a) An image of size 500 × 500 pixels and (b) its Fourier spectrum. The superimposed circles have radii values of 5, 15, 30, 80, and 230, which enclose 92.0, 94.6, 96.4, 98.0, and 99.5% of the image power, respectively.



Ideal lowpass filtering with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, respectively

From [Gonzalez & Woods]

2D-DFT Domain Filter Design

- Gaussian lowpass

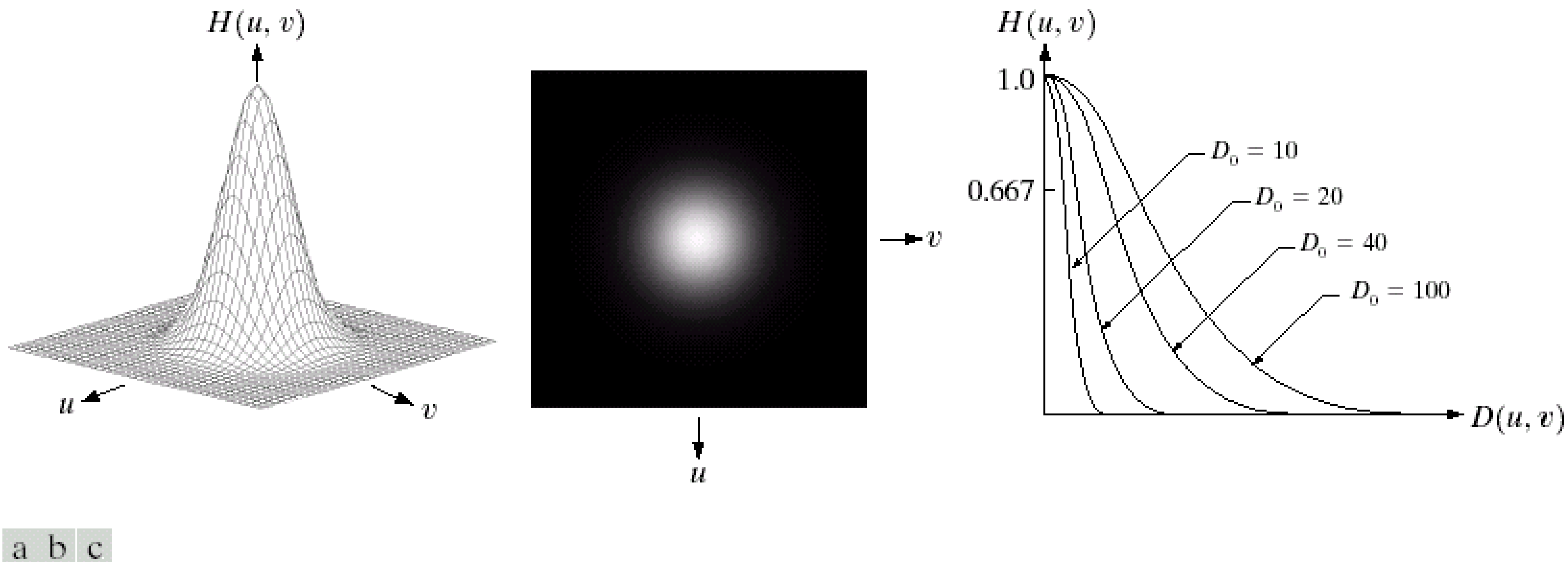
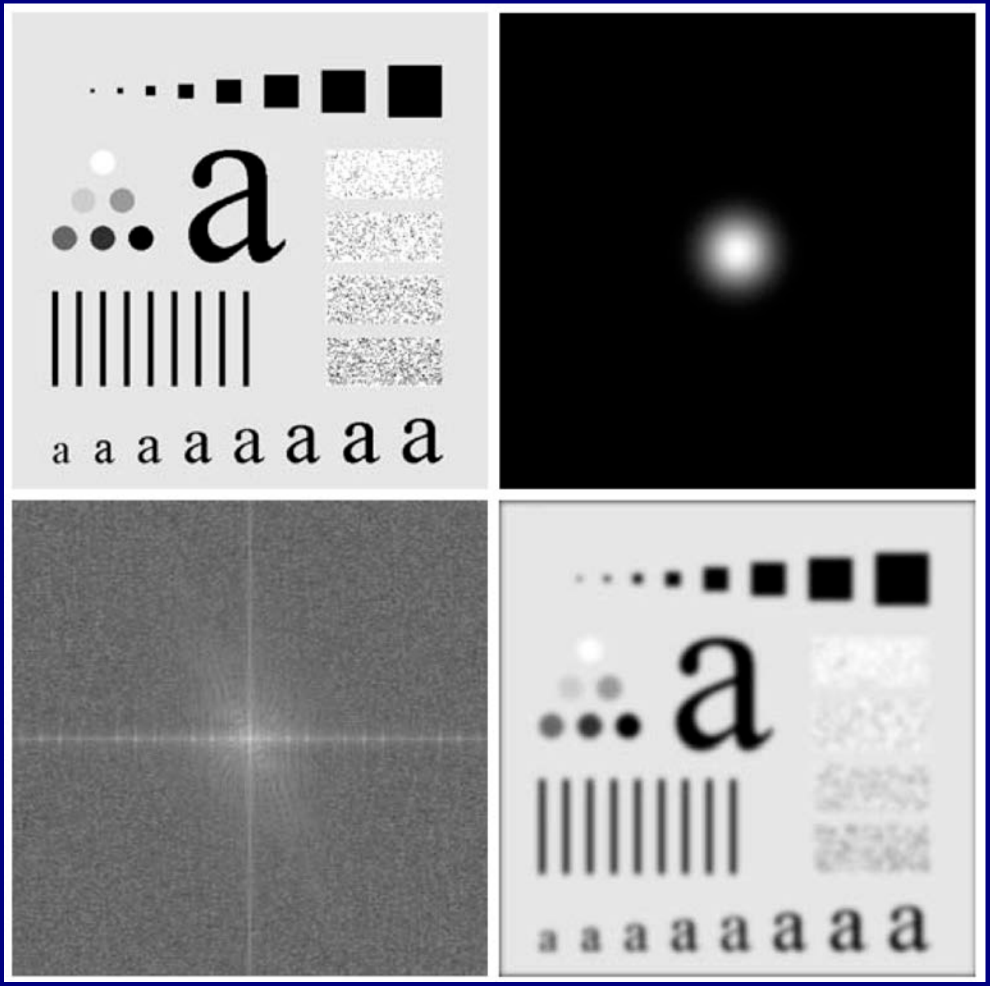


FIGURE 4.17 (a) Perspective plot of a GLPF transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections for various values of D_0.

From [Gonzalez & Woods]

2D-DFT Domain Filter Design



Effect of Gaussian lowpass filter

From [Gonzalez & Woods]

2D-DFT Domain Filter Design

Effect of Gaussian lowpass filter

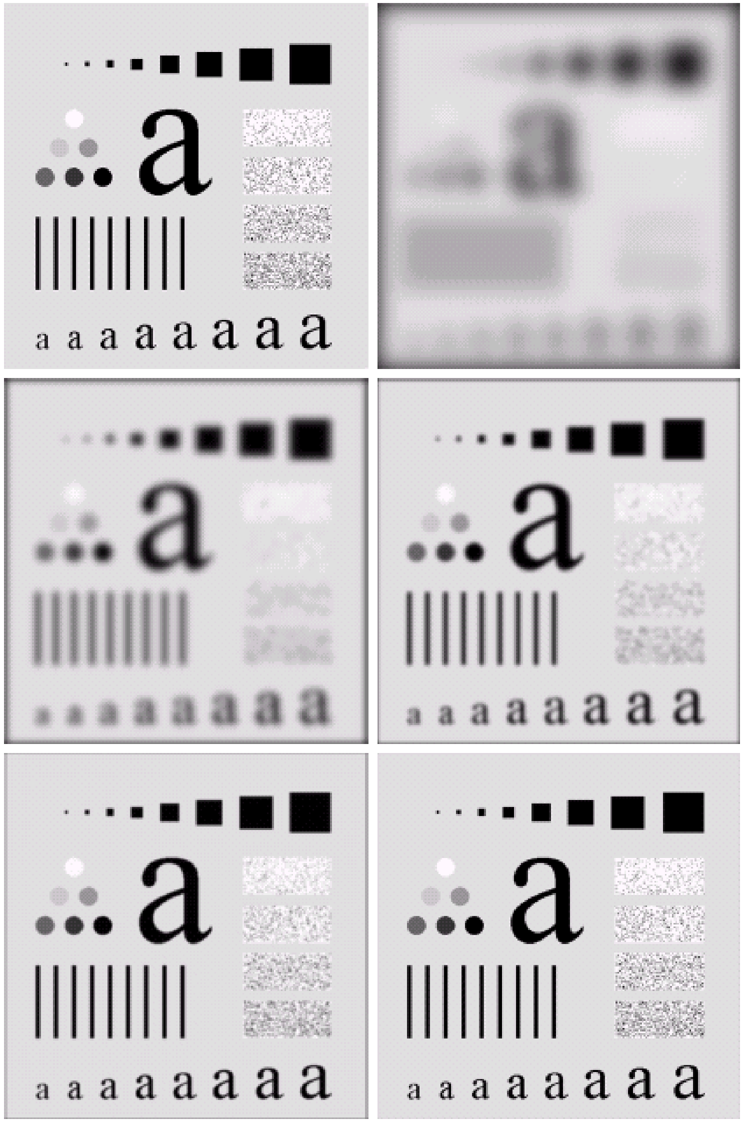
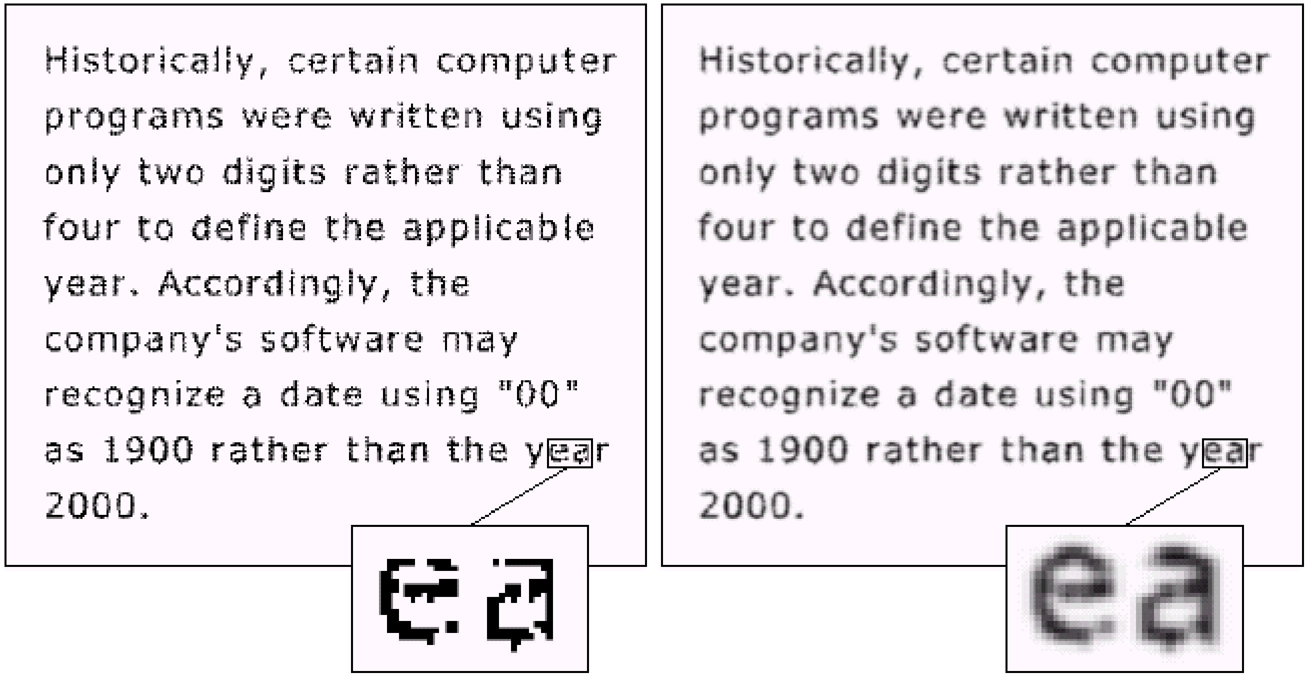


FIGURE 4.18 (a) Original image. (b)–(f) Results of filtering with Gaussian lowpass filters with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Figs. 4.12 and 4.15.

From [Gonzalez & Woods]

2D-DFT Domain Filter Design



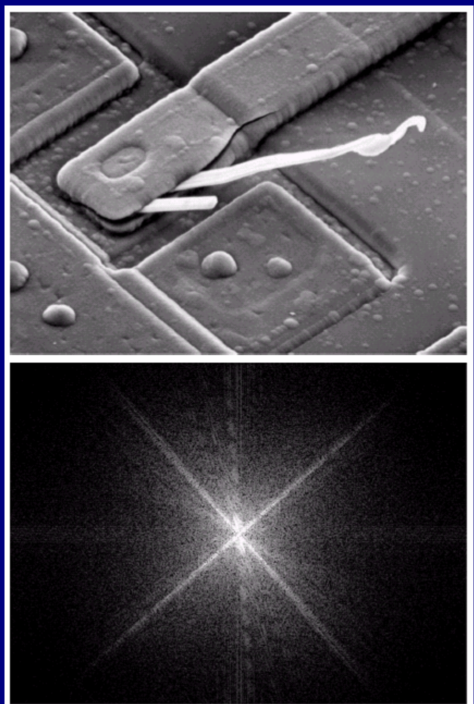
a b

FIGURE 4.19
(a) Sample text of poor resolution (note broken characters in magnified view).
(b) Result of filtering with a GLPF (broken character segments were joined).

Effect of Gaussian lowpass filter

From [Gonzalez & Woods]

2D-DFT Domain Filter Design



Gaussian lowpass filtering

Gaussian highpass filtering

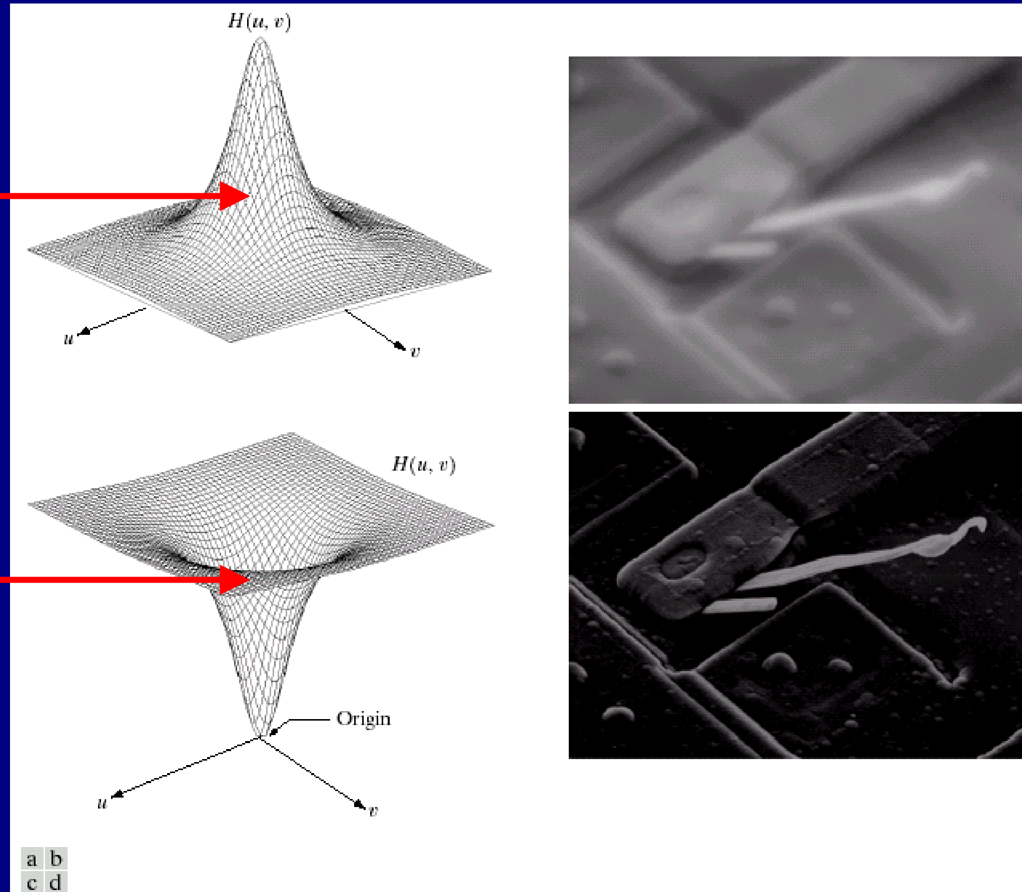


FIGURE 4.7 (a) A two-dimensional lowpass filter function. (b) Result of lowpass filtering the image in Fig. 4.4(a). (c) A two-dimensional highpass filter function. (d) Result of highpass filtering the image in Fig. 4.4(a).

From [Gonzalez & Woods]

2D-DFT Domain Filter Design

- Choices of highpass filters

Ideal Butterworth Gaussian

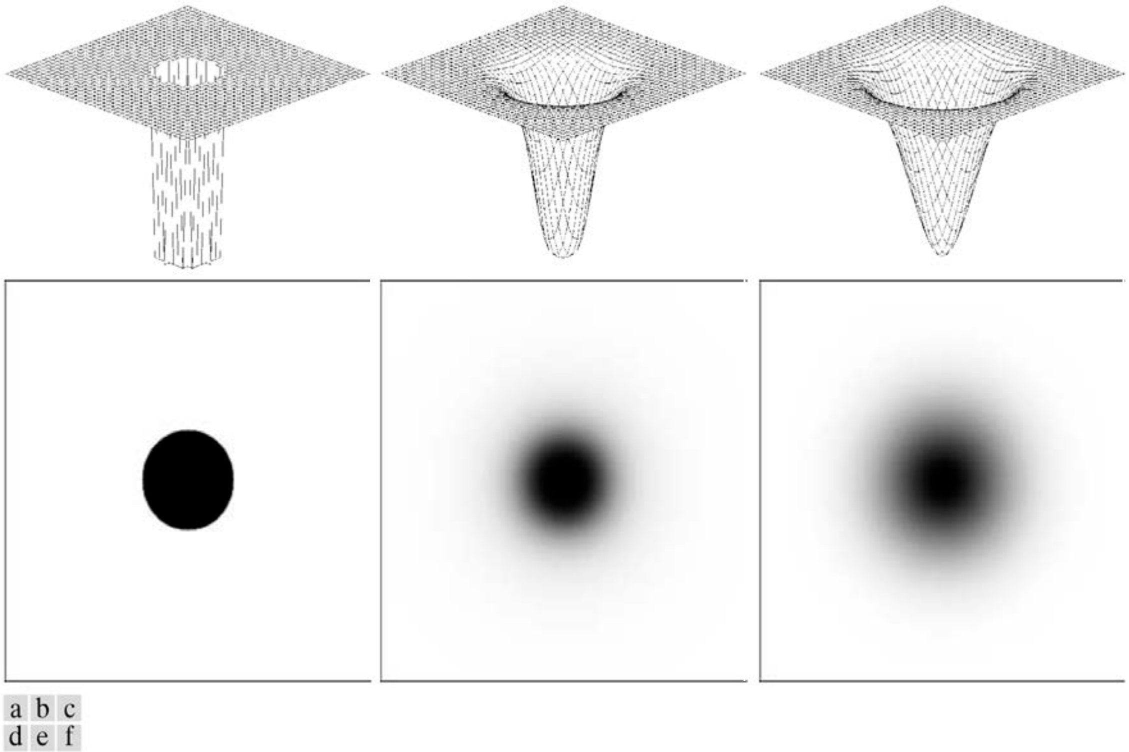


FIGURE 4.17 Top row: Perspective plots of ideal, Butterworth, and Gaussian highpass filters. Bottom row: Corresponding images.

From [Gonzalez & Woods]

2D-DFT Domain Filter Design

Obtained by applying inverse 2D-DFT to the corresponding frequency domain filters

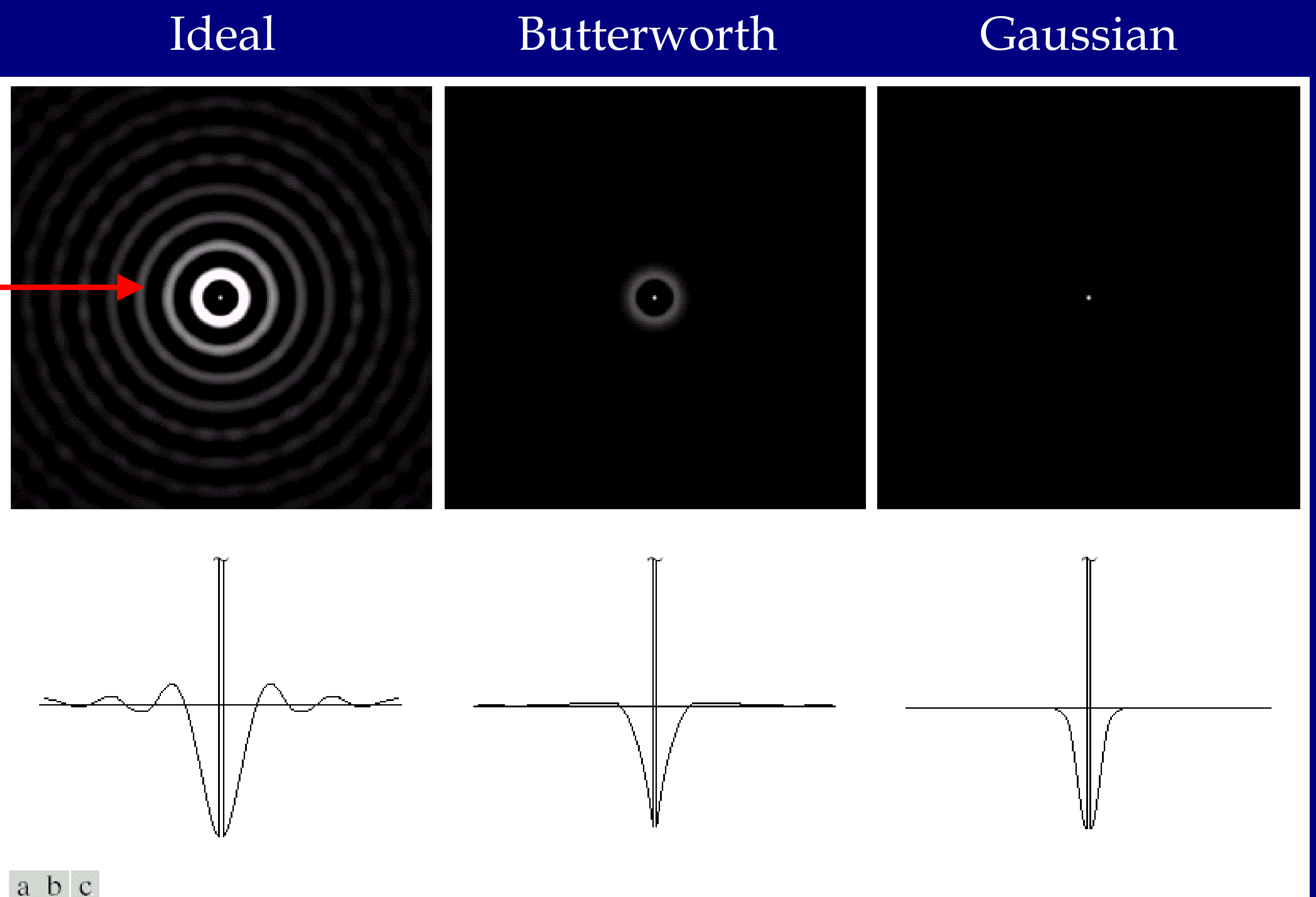
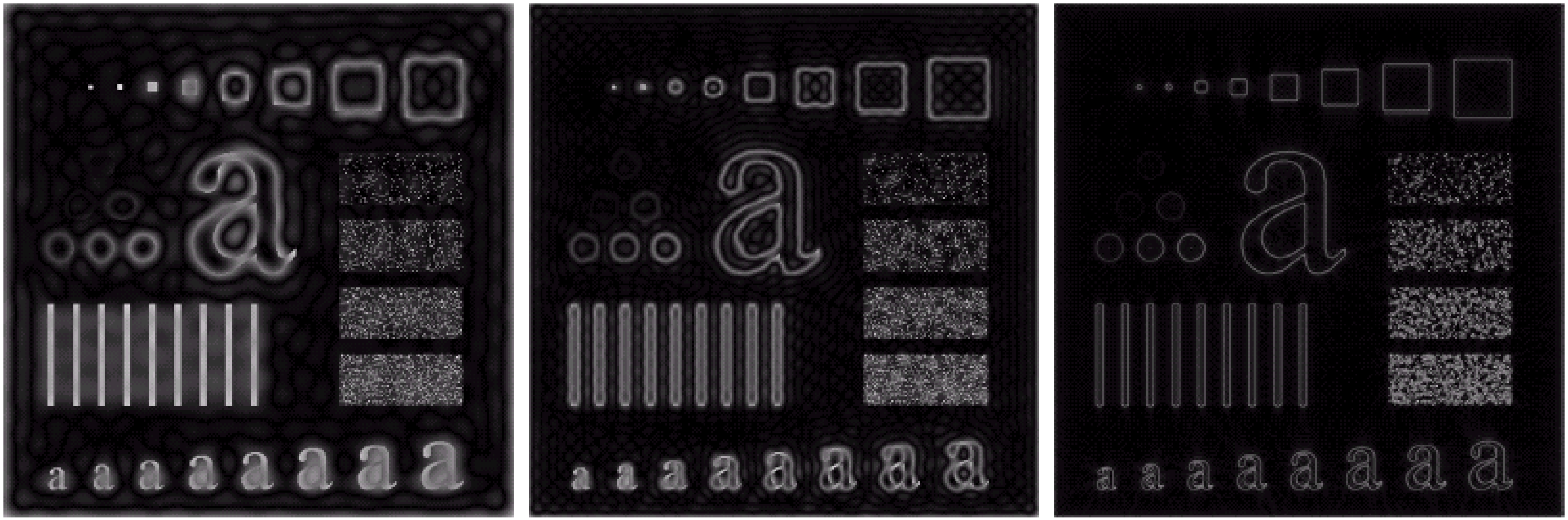


FIGURE 4.23 Spatial representations of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding gray-level profiles.

From [Gonzalez & Woods]

2D-DFT Domain Filter Design

Ideal Butterworth Gaussian



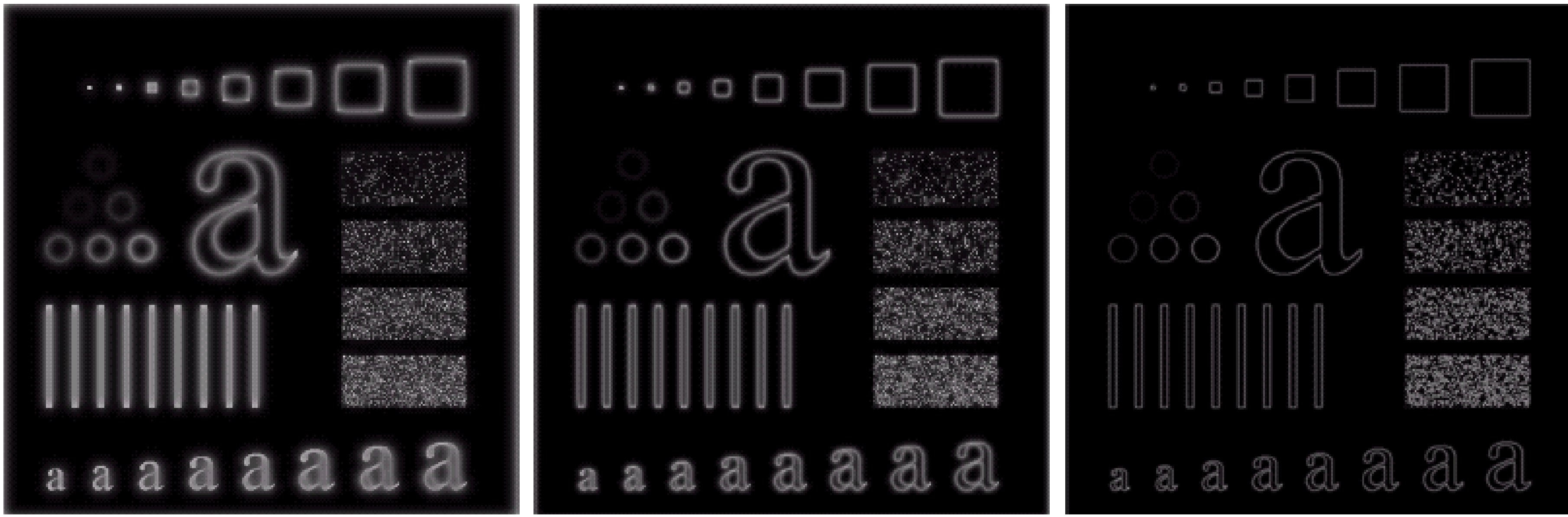
a b c

FIGURE 4.24 Results of ideal highpass filtering the image in Fig. 4.11(a) with $D_0 = 15$, 30, and 80, respectively. Problems with ringing are quite evident in (a) and (b).

From [Gonzalez & Woods]

2D-DFT Domain Filter Design

Gaussian filter with different width



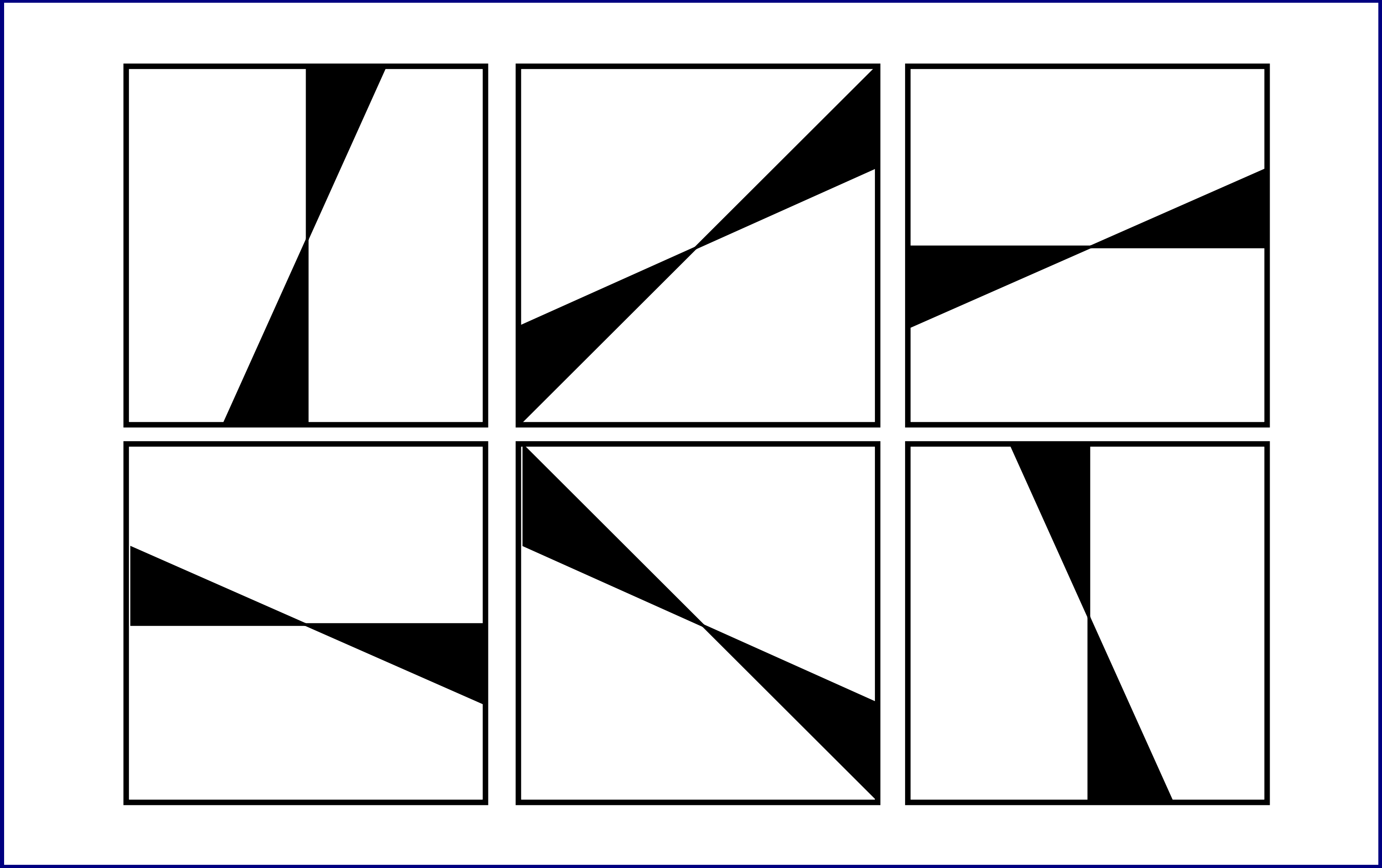
a b c

FIGURE 4.26 Results of highpass filtering the image of Fig. 4.11(a) using a GHPF of order 2 with $D_0 = 15$, 30, and 80, respectively. Compare with Figs. 4.24 and 4.25.

From [Gonzalez & Woods]

2D-DFT Domain Filter Design

- Orientation selective filters

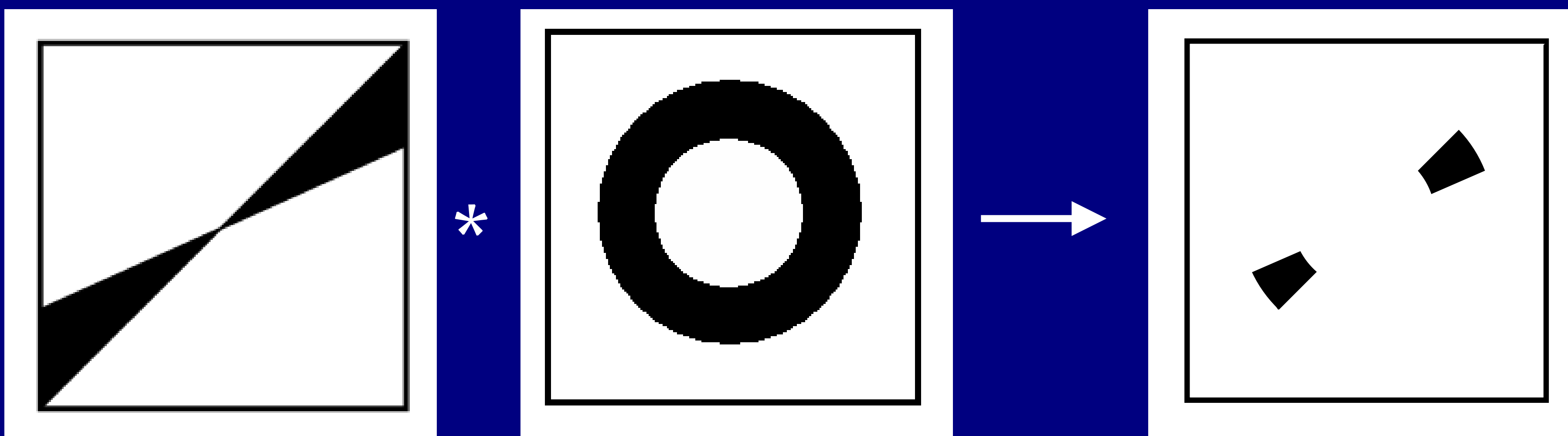


From Prof. Al Bovik

2D-DFT Domain Filter Design

- Narrowband Filtering

by combining radial and orientation selection



From Prof. Al Bovik