2. LSI Systems & Transforms—Fourier transform

- Let $f(x)$ be a continuous function of a single variable x and $F(u)$ be its Fourier transform, then

$$F(u) = \Im\{f(x)\} = \int_{-\infty}^{\infty} f(x)\exp(-j2\pi ux)dx$$

$$\underline{f(x)} = \Im^{-1}\{F(u)\} = \int_{-\infty}^{\infty} F(u)\underline{\exp(j2\pi ux)}du$$

where u is frequency variable and $j = \sqrt{-1}$

- The two dimensional (2-D) Fourier transform and inverse Fourier transform are given by

$$F(u,v) = \Im\{f(x,y)\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\exp[-j2\pi(ux+vy)]dxdy$$

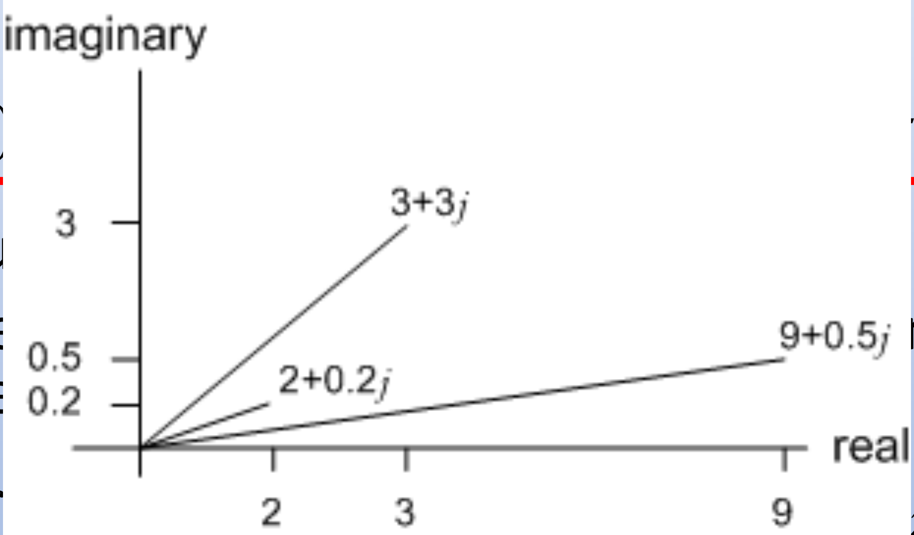
$$f(x,y) = \Im^{-1}\{F(u,v)\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)\exp[j2\pi(ux+vy)]dudv$$

where u and v are 2 frequency variables.

2. LSI Systems & Transforms—Fourier transform

$$f(x) = A\cos(2\pi ux + \varphi)$$
$$= (A\cos\varphi)\cos 2\pi ux - (A\sin\varphi)\sin 2\pi ux$$
$$= \tfrac{1}{2}A\exp(j2\pi ux + j\varphi) + \tfrac{1}{2}A\exp(-j2\pi ux - j\varphi)$$
$$= (\tfrac{1}{2}A\exp j\varphi)\exp j2\pi ux + (\tfrac{1}{2}A\exp(-j\varphi))\exp(-j2\pi ux)$$

- Let $f(x)$ … x and $F(u)$ be its F…



$f(x)$ … $ux)du$

where u is frequ…

- The two dime… nd inverse Fourier transform are

$F(u,v) = \Im\{f$ … $\pi(ux+vy)]dxdy$

$$c = a + jb = re^{j\varphi}$$
$$a = r\cos(\varphi), b = r\sin(\varphi), r = \sqrt{a^2+b^2}, \varphi = \tan^{-1}(\frac{b}{a})$$

where u and v are 2 frequency variables.

2. LSI Systems & Transforms—Fourier transform

- 2-D Fourier transform is separable

$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\exp[-j2\pi(ux+vy)]dxdy$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\exp(-j2\pi ux)\exp(-j2\pi vy)dxdy$$

$$= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(x,y)\exp(-j2\pi ux)dx\right]\exp(-j2\pi vy)dy$$

$$= \int_{-\infty}^{\infty} F_x(u,y)\exp(-j2\pi vy)dy$$

$$? = F_x(u)F_y(v) \quad \text{only if } f(x,y) = f_1(x)f_2(y)$$

- Note

$$\exp[-j2\pi(ux+vy)] = \cos[-2\pi(ux+vy)] + j\sin(-2\pi(ux+vy))$$

$$= \cos[2\pi(ux+vy)] - j\sin(2\pi(ux+vy))$$

2. LSI Systems & Transforms—Fourier transform

- Obviously, $F(u,v)$ is in general a complex function that can be represented by

$$F(u,v) = R(u,v) + jI(u,v) = |F(u,v)| \exp[j\varphi(u,v)]$$

where $|F(u,v)| = \sqrt{R^2(u,v) + I^2(u,v)}$

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

$$R(u,v) = |F(u,v)| \cos\varphi(u,v)$$

$$I(u,v) = |F(u,v)| \sin\varphi(u,v)$$

$$c = a + jb = re^{j\varphi}$$
$$a = r\cos(\varphi), b = r\sin(\varphi)$$
$$r = \sqrt{a^2 + b^2}, \varphi = \tan^{-1}\left(\frac{b}{a}\right)$$

2. LSI Systems & Transforms—definition of DFT

- The discrete Fourier transform (DFT) of a 2-D discrete function (or image) $f(x,y)$ of size mxn is defined by:

$$F(u,v) = \frac{1}{mn}\sum_{x=0}^{m-1}\sum_{y=0}^{n-1} f(x,y)\exp[-j2\pi(ux/m + vy/n)]$$

$$f(x,y) = \frac{1}{mn}\sum_{u=0}^{m-1}\sum_{v=0}^{n-1} F(u,v)\exp[j2\pi(ux/m + vy/n)]$$

- where u and v are also discrete variable.

- Comparing with continuous Fourier transform:

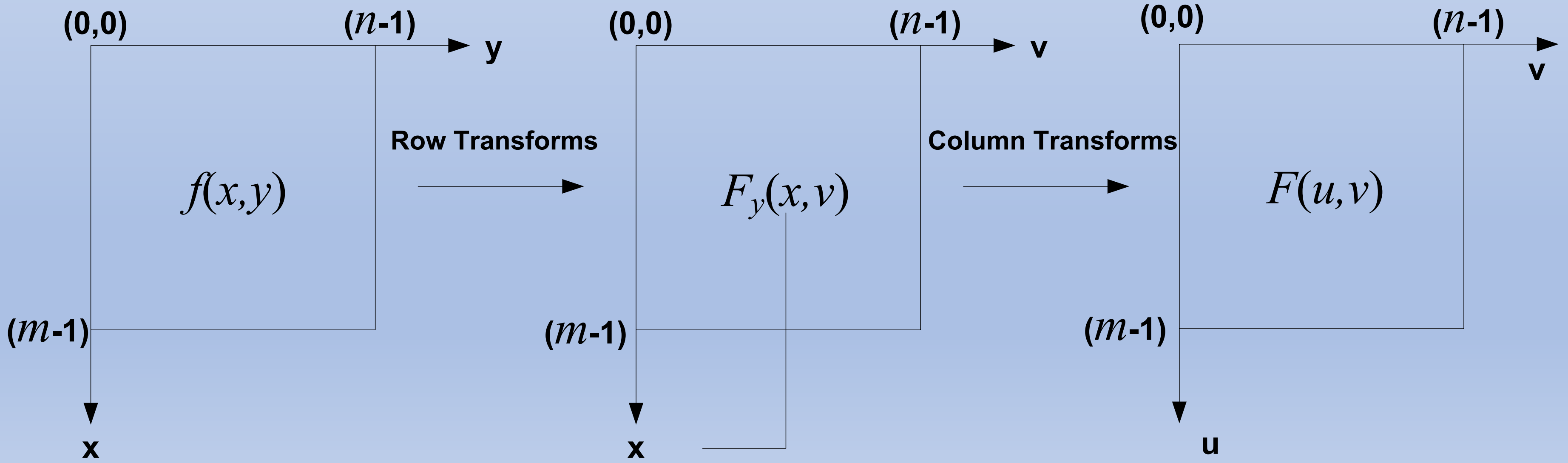
$$F(u,v) = \Im\{f(x,y)\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\exp[-j2\pi(ux+vy)]dxdy$$

2. LSI Systems & Transforms—properties of DFT

- The 2-D DFT is also separable:

$$F(u,v) = \frac{1}{mn}\sum_{x=0}^{m-1}\exp[-j2\pi ux/m]\sum_{y=0}^{n-1} f(x,y)\exp[-j2\pi vy/n]$$

$$f(x,y) = \frac{1}{mn}\sum_{u=0}^{m-1}\exp[j2\pi ux/m]\sum_{v=0}^{n-1} F(u,v)\exp[j2\pi vy/n]$$



2. LSI Systems & Transforms—properties of DFT

- Periodicity:

$$F(u,v) = F(u+m,v) = F(u,v+n) = F(u+m,v+n)$$

- Conjugate symmetry for real image $f(x,y)$

$$F(u,v) = F^*(-u,-v), \qquad |F(u,v)| = |F(-u,-v)|$$

- Linearity and scaling:

$$\Im\{\alpha f_1(x,y) + \beta f_2(x,y) + \cdots\} = \alpha F_1(u,y) + \beta F_2(u,v) + \cdots$$

$$\Im\{f(\alpha x, \beta y)\} = \frac{1}{|\alpha\beta|} F(u/\alpha, y/\beta)$$

- Convolution theorem:

For contunuous function: $f(x,y) * g(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta) g(x-\alpha, y-\beta) d\alpha d\beta$

$$f(x,y) * g(x,y) \Leftrightarrow F(u,v)G(u,v)$$

$$f(x,y)g(x,y) \Leftrightarrow F(u,v) * G(u,v)$$

- Be careful to apply this in digital image. Apply zero padding!

2. LSI Systems & Transforms—properties of DFT

- Translation

$$f(x-x_0, y-y_0) \Leftrightarrow F(u,v)\exp[-j2\pi(x_0u/m+y_0v/n)]$$
$$F(u-u_0, v-v_0) \Leftrightarrow f(x,y)\exp[j2\pi(u_0x/m+v_0y/n)]$$

- Rotation:

$$\text{Let: } x=r\cos\theta, y=r\sin\theta, u=\omega\cos\varphi, v=\omega\sin\varphi$$
$$f(r,\theta+\theta_0) \Leftrightarrow F(\omega,\varphi+\theta_0)$$

- Rotation Invariant Transform:

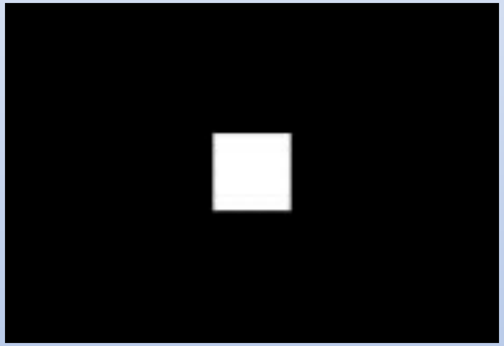
$$g(u,v)=\frac{1}{\pi}\int_0^{2\pi}\int_0^1 e^{-j(2\pi ur^2+v\theta)} f(r,\theta)drd\theta$$

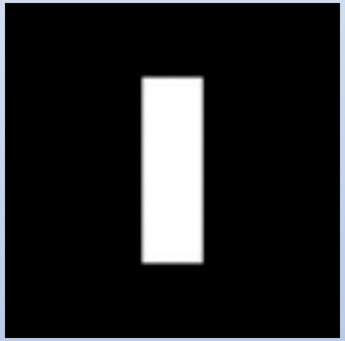
P. Yap, X.D. Jiang and A. Kot, "Two Dimensional Polar Harmonic Transforms for Invariant Image Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1259-1270, July 2010.

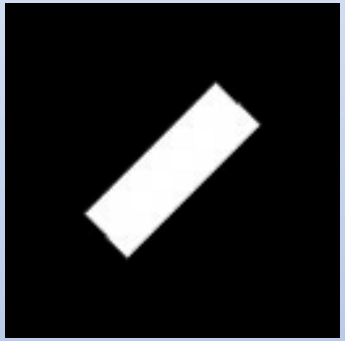
2. LSI Systems & Transforms—properties of DFT

➢ Examples:

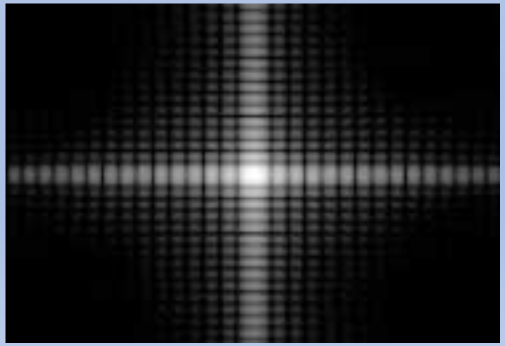
$f(x,y)$

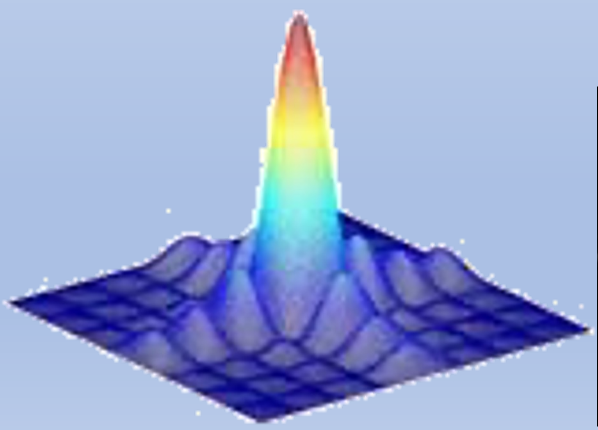


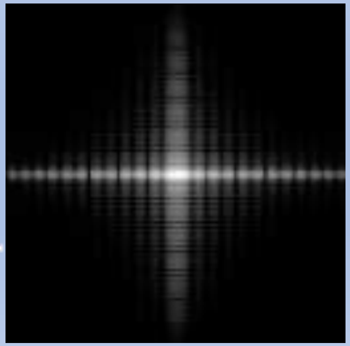


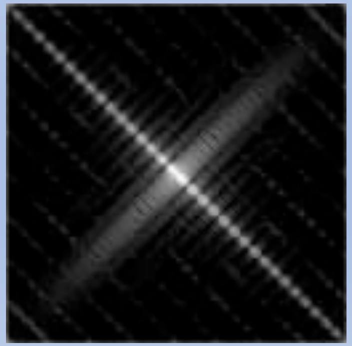


$|F(u,v)|$









2. LSI Systems & Transforms—properties of DFT

TABLE 4.1
Summary of some important properties of the 2-D Fourier transform.

Property	Expression(s)
Fourier transform	$F(u, v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x, y)e^{-j2\pi(ux/M+vy/N)}$
Inverse Fourier transform	$f(x, y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u, v)e^{j2\pi(ux/M+vy/N)}$
Polar representation	$F(u, v) = \lvert F(u, v)\rvert e^{-j\phi(u,v)}$
Spectrum	$\lvert F(u, v)\rvert = \left[R^2(u, v) + I^2(u, v)\right]^{1/2}$, $R = \text{Real}(F)$ and $I = \text{Imag}(F)$
Phase angle	$\phi(u, v) = \tan^{-1}\left[\frac{I(u, v)}{R(u, v)}\right]$
Power spectrum	$P(u, v) = \lvert F(u, v)\rvert^2$
Average value	$\bar{f}(x, y) = F(0, 0) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x, y)$
Translation	$f(x, y)e^{j2\pi(u_0x/M+v_0y/N)} \Leftrightarrow F(u - u_0, v - v_0)$ $f(x - x_0, y - y_0) \Leftrightarrow F(u, v)e^{-j2\pi(ux_0/M+vy_0/N)}$ When $x_0 = u_0 = M/2$ and $y_0 = v_0 = N/2$, then $f(x, y)(-1)^{x+y} \Leftrightarrow F(u - M/2, v - N/2)$ $f(x - M/2, y - N/2) \Leftrightarrow F(u, v)(-1)^{u+v}$

2. LSI Systems & Transforms—properties of DFT

Conjugate symmetry	$F(u,v) = F^*(-u,-v)$ $\lvert F(u,v)\rvert = \lvert F(-u,-v)\rvert$
Differentiation	$\dfrac{\partial^n f(x,y)}{\partial x^n} \Leftrightarrow (ju)^n F(u,v)$ $(-jx)^n f(x,y) \Leftrightarrow \dfrac{\partial^n F(u,v)}{\partial u^n}$
Laplacian	$\nabla^2 f(x,y) \Leftrightarrow -(u^2+v^2)F(u,v)$
Distributivity	$\Im[f_1(x,y) + f_2(x,y)] = \Im[f_1(x,y)] + \Im[f_2(x,y)]$ $\Im[f_1(x,y)\cdot f_2(x,y)] \neq \Im[f_1(x,y)]\cdot\Im[f_2(x,y)]$
Scaling	$af(x,y) \Leftrightarrow aF(u,v),\ f(ax,by) \Leftrightarrow \dfrac{1}{\lvert ab\rvert}F(u/a,v/b)$
Rotation	$x = r\cos\theta \quad y = r\sin\theta \quad u = \omega\cos\varphi \quad v = \omega\sin\varphi$ $f(r,\theta+\theta_0) \Leftrightarrow F(\omega,\varphi+\theta_0)$
Periodicity	$F(u,v) = F(u+M,v) = F(u,v+N) = F(u+M,v+N)$ $f(x,y) = f(x+M,y) = f(x,y+N) = f(x+M,y+N)$
Separability	See Eqs. (4.6-14) and (4.6-15). Separability implies that we can compute the 2-D transform of an image by first computing 1-D transforms along each row of the image, and then computing a 1-D transform along each column of this intermediate result. The reverse, columns and then rows, yields the same result.

2. LSI Systems & Transforms—properties of DFT

Property	Expression(s)
Computation of the inverse Fourier transform using a forward transform algorithm	$\frac{1}{MN} f^*(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F^*(u, v) e^{-j2\pi(ux/M + vy/N)}$ This equation indicates that inputting the function $F^*(u, v)$ into an algorithm designed to compute the forward transform (right side of the preceding equation) yields $f^*(x, y)/MN$. Taking the complex conjugate and multiplying this result by MN gives the desired inverse.
Convolution†	$f(x, y) * h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) h(x - m, y - n)$
Correlation†	$f(x, y) \circ h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f^*(m, n) h(x + m, y + n)$
Convolution theorem†	$f(x, y) * h(x, y) \Leftrightarrow F(u, v) H(u, v)$; $f(x, y) h(x, y) \Leftrightarrow F(u, v) * H(u, v)$
Correlation theorem†	$f(x, y) \circ h(x, y) \Leftrightarrow F^*(u, v) H(u, v)$; $f^*(x, y) h(x, y) \Leftrightarrow F(u, v) \circ H(u, v)$

2. LSI Systems & Transforms—some basic FT pairs

Some useful FT pairs:

Impulse $\delta(x, y) \Leftrightarrow 1$

Gaussian $A\sqrt{2\pi}\sigma e^{-2\pi^2\sigma^2(x^2+y^2)} \Leftrightarrow Ae^{-(u^2+v^2)/2\sigma^2}$

Rectangle $\mathrm{rect}[a, b] \Leftrightarrow ab\frac{\sin(\pi ua)}{(\pi ua)}\frac{\sin(\pi vb)}{(\pi vb)}e^{-j\pi(ua+vb)}$

Cosine $\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$

$$\frac{1}{2}\left[\delta(u + u_0, v + v_0) + \delta(u - u_0, v - v_0)\right]$$

Sine $\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow$

$$j\frac{1}{2}\left[\delta(u + u_0, v + v_0) - \delta(u - u_0, v - v_0)\right]$$

† Assumes that functions have been extended by zero padding.

2. LSI Systems & Transforms—image sampling

- An digital image $f_d(m,n)$ is obtained by sampling and quantizing a continuous analogy image $f_c(x,y)$. Here the sampling converts the continuous variable $x,\ y$ into integer $m,\ n$ and the quantization convert the continuous variable f_c into a finite set of numbers f_d . As the quantization has less impact than sampling to the image processing, we will focus more on the sampling process and theory.
- Given a continuous analogy image $f_c(x,y)$, it is very simple to get its discrete image $f_d(m,n)$ mathematically by

$$f_d(m,n) = f_c(m\Delta x, n\Delta y) = f_c(x,y)\big|_{\text{Let } x=m\Delta x,\, y=n\Delta y}$$

- Does $f_d(m,n)$ contain the same information as $f_c(x,y)$? Under what conditions is it yes? Mathematically analyzing this, however, needs abstract mathematical tools.

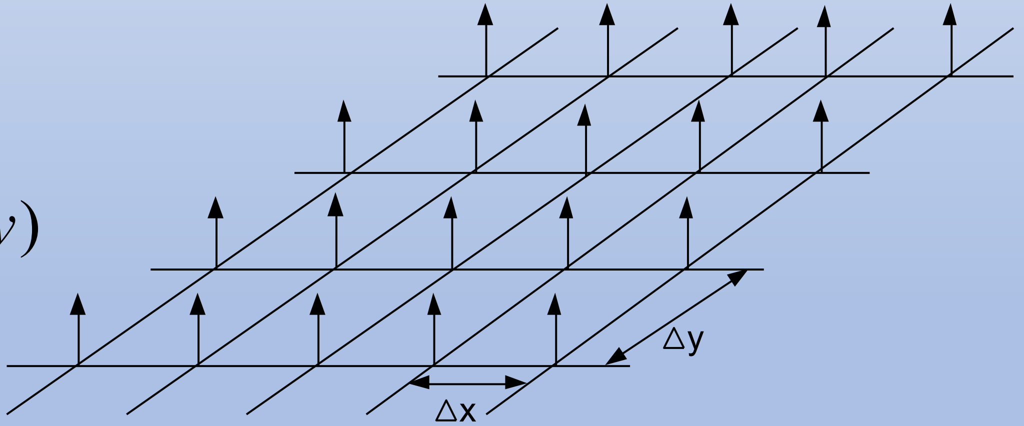
2. LSI Systems & Transforms—image sampling

- A 2-D function $f_c(x,y)$ is band-limited if its Fourier transform $F_c(u,v)$ is zero outside a bounded spatial frequency support; e.g.,

$$F_c(u,v)=0, \quad \text{for } |u|>U_0, |v|>V_0$$

- where $2U_0$ and $2V_0$ are referred to as the x and y bandwidths of the 2-D function. Why $2U_0$ and $2V_0$?
- In practice, real-world images can be well approximated by band-limited signals.
- Define a 2-D sampling function

$$s(x,y)=\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\delta(x-m\Delta x, y-n\Delta y)$$



- It has the Fourier transform of

$$S(u,v)=\frac{1}{\Delta x\Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\delta(u-m/\Delta x, v-n/\Delta y)$$

2. LSI Systems & Transforms—image sampling

- Multiply the continuous image $f_c(x,y)$ with the sampling image $s(x,y)$ yield

$$f_d(x,y) = f_c(x,y)s(x,y) = \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} f_c(m\Delta x, n\Delta y)\delta(x-m\Delta x, y-n\Delta y)$$

- The Fourier transform of the sampled image $f_d(x,y)$ in continuous domain is:

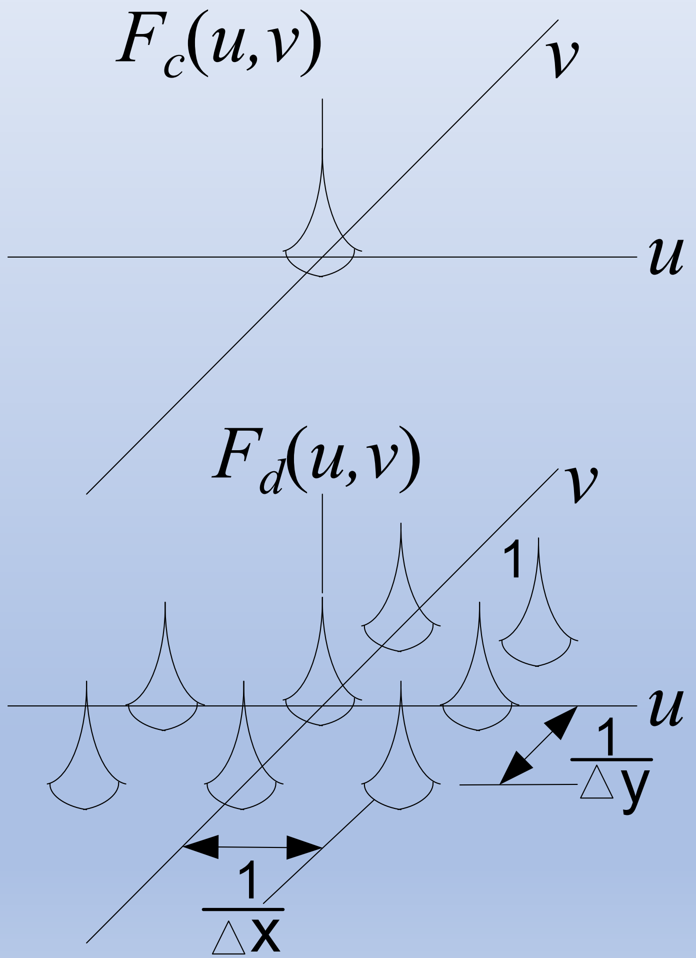
$$F_d(u,v) = F_c(u,v) * S(u,v) = \frac{1}{\Delta x \Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} F_c(u,v) * \delta(u-m/\Delta x, y-n/\Delta y)$$

$$= \frac{1}{\Delta x \Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} F_c(u-m/\Delta x, y-n/\Delta y) = \frac{1}{\Delta x \Delta y}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} F_c(u-mf_{xs}, y-nf_{ys})$$

- It is a periodic replication of $F_c(u,v)$, on a rectangular grid with spacing $(1/\Delta x, 1/\Delta y)$.

2. LSI Systems & Transforms—image sampling

- The spectrum of the sampled image $f_d(x,y)$ consists of the spectrum of the continuous $f_c(x,y)$ image (top) infinitely repeated over the frequency plane in a rectangular grid with spacing $(1/\Delta x,\ 1/\ \Delta y)$. It is a periodic replication of $F_c(u,v)$, on a rectangular grid with spacing $(1/\Delta x,\ 1/\ \Delta y)$.



2. LSI Systems & Transforms—image sampling

- If the x, y sampling frequencies are greater than the bandwidths, or if the sampling intervals are smaller than the reciprocal of bandwidths, i.e.,

$$f_{xs} = \frac{1}{\Delta x} \geq 2U_0 \;\; \& \;\; f_{ys} = \frac{1}{\Delta y} \geq 2V_0 \quad \text{or} \quad \Delta x \leq \frac{1}{2U_0} \;\; \& \;\; \Delta y \leq \frac{1}{2V_0}$$

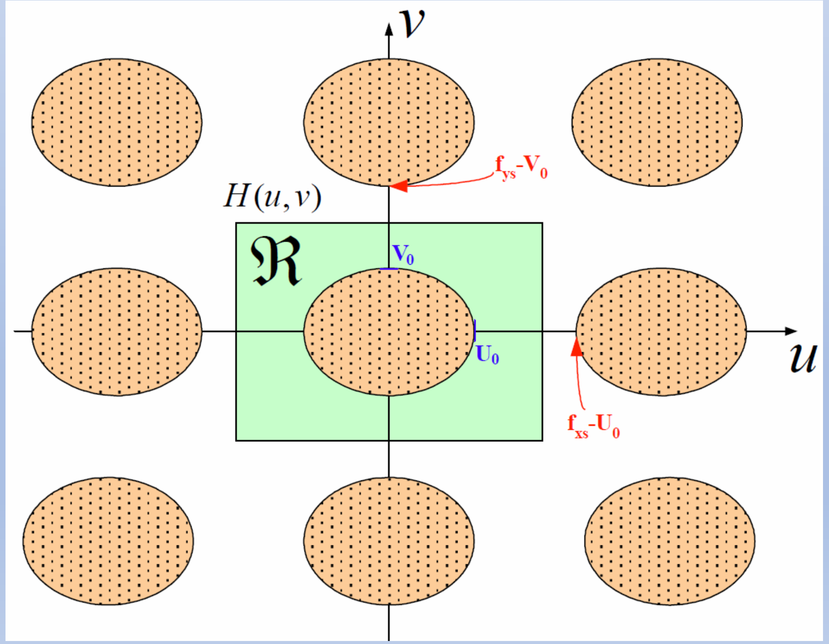
- Then $F_c(u,v)$ can be recovered from $F_d(u,v)$ by using a low-pass filter with frequency response

$$H(u,v) = \begin{cases} \Delta x \Delta y, & (u,v) \in \mathcal{R} \\ 0, & \text{otherwise} \end{cases}$$

- That is

$$F_c(u,v) = F_d(u,v) H(u,v)$$

$$f_c(x,y) = f_d(x,y) * h(x,y)$$



2. LSI Systems & Transforms—image sampling

$$f_c(x,y) = h(x,y) * f_d(x,y)$$

$$= h(x,y) * \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f_c(m\Delta x, n\Delta y)\delta(x - m\Delta x, y - n\Delta y)$$

$$= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f_c(m\Delta x, n\Delta y) h(x,y) * \delta(x - m\Delta x, y - n\Delta y)$$

$$= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f_c(m\Delta x, n\Delta y) h(x - m\Delta x, y - n\Delta y)$$

$$= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f_d(m,n) h(x - m\Delta x, y - n\Delta y)$$

- Therefore, all information of the continuous image $f_c(x,y)$ can be recovered from the discrete image $f_d(m,n)$.

2. LSI Systems & Transforms—image sampling

Sampling Theorem:

- A band-limited image $f_c(x,y)$ with bandwidths ($2\mathrm{U}_0$, $2\mathrm{V}_0$) sampled uniformly on a rectangular grid with spacing $(\Delta x,\Delta y)$ can be recovered without error from the sampled values $f_c(m\Delta x,n\Delta y)= f_d(m,n)$ provided that the sampling rates (f_{xs}, f_{ys}) are greater than the Nyquist rates.
- The lower bounds of the required sampling rates, the band width, are known as the Nyquist rates or Nyquist frequencies. Their reciprocals are known as the Nyquist intervals.
- Sampling below the Nyquist rates will cause the periodic replications of $F_c(u,v)$ to overlap, resulting in a distorted spectrum $F_d(u,v)$, in which $F_c(u,v)$ is irrevocably lost—a phenomenon that is known as aliasing.
- Aliasing can be avoided or reduced by low-pass filtering the image $f_c(x,y)$ before sampling so that its bandwidth is less than the sampling frequency. (at the expense of what?)

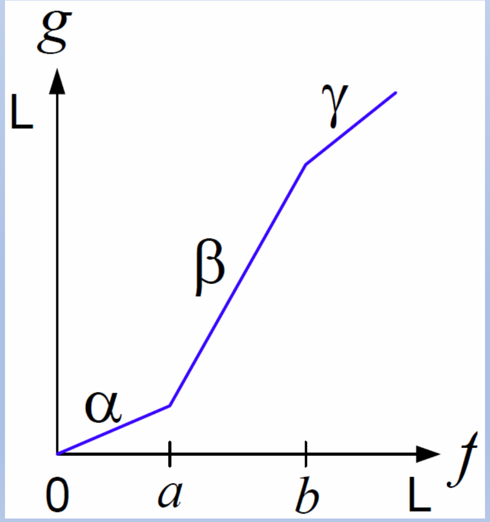
3. Image Enhancement—Outline

- Simple Point Processing Approaches
- Histogram Equalization
- Image Smoothing
- Image Sharpening
- Nonlinear Image Processing

3. Image Enhancement—point processing

- Image processing operations are designed to enhance image content or features so that they are more suitable for display or analysis.
- Many image enhancement processes are point and memoryless operations which map input image gray-level to output gray-level according to a transformation $g=T(f)$. For example:
  - Power Transformation (gamma correction): $g=c\,f^{\gamma}$
  - Log Transformation: $g=c\log(1+f)$
  - Piecewise Linear Transformation

$$g = T(f) = \begin{cases} \alpha f, & 0 \le f < a \\ \beta(f-a)+T(a), & a \le f < b \\ \gamma(f-b)+T(b), & b \le f < L \end{cases}$$



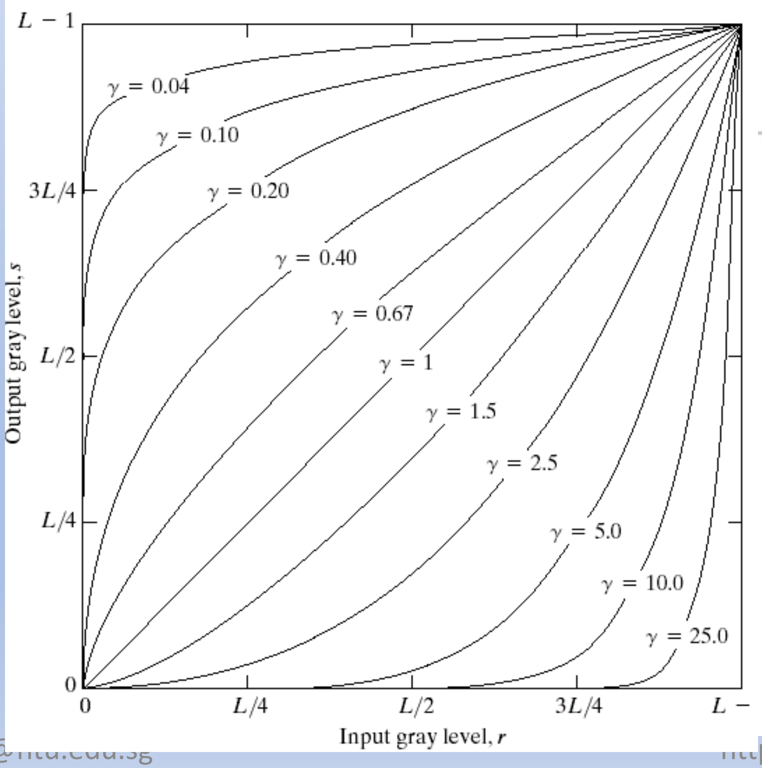
  - Histogram equalization

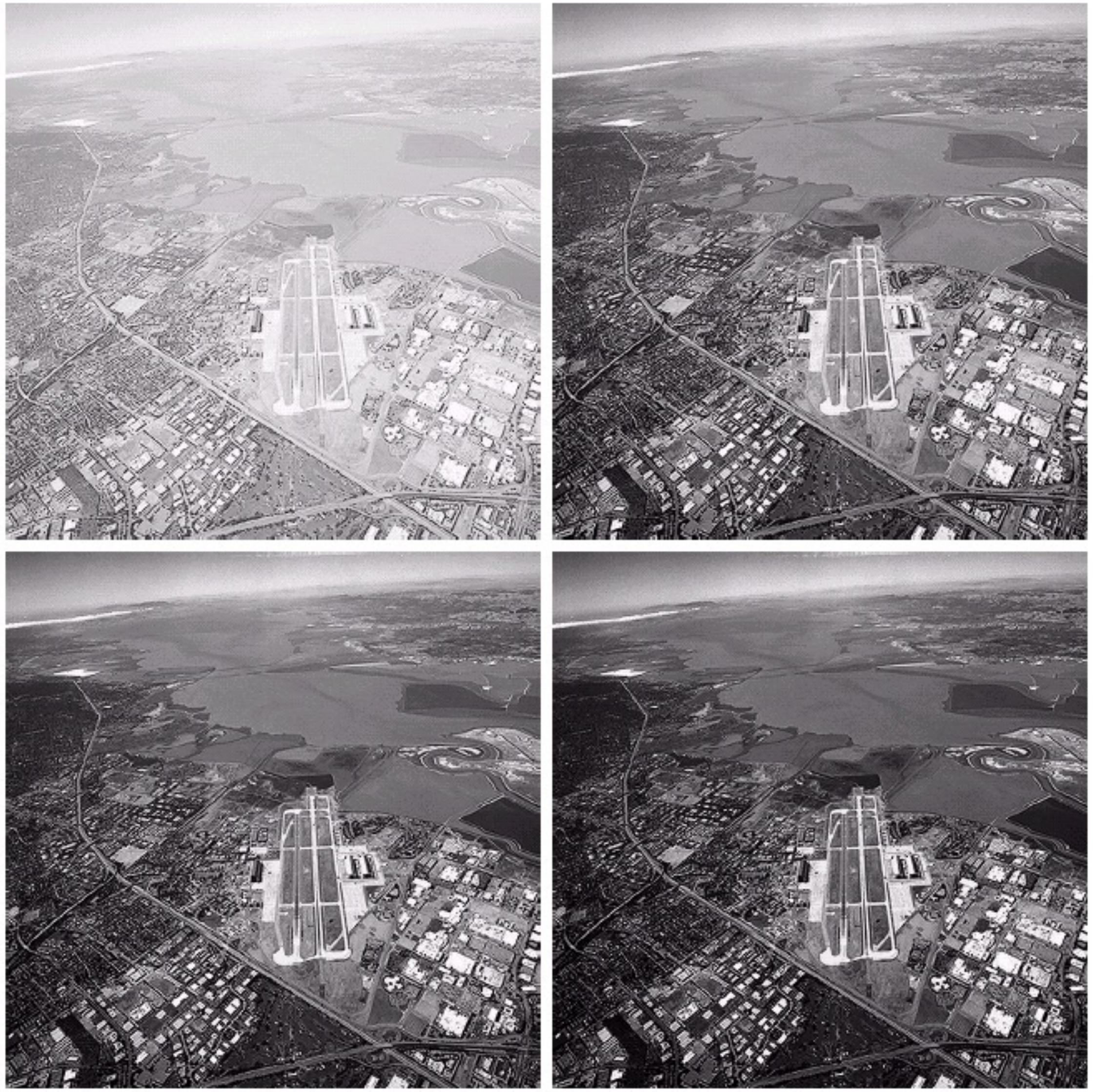
3. Image Enhancement—simple point processing

Power Transformation

gamma correction

$$g = c\,f^{\gamma}$$



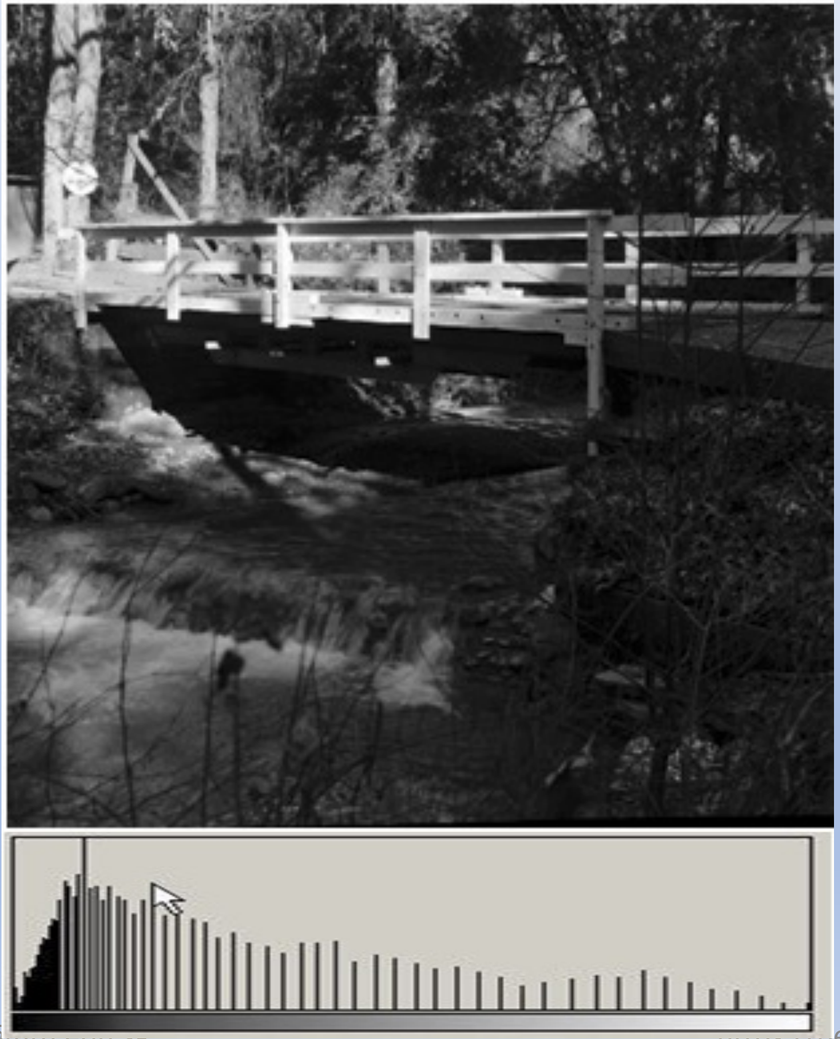


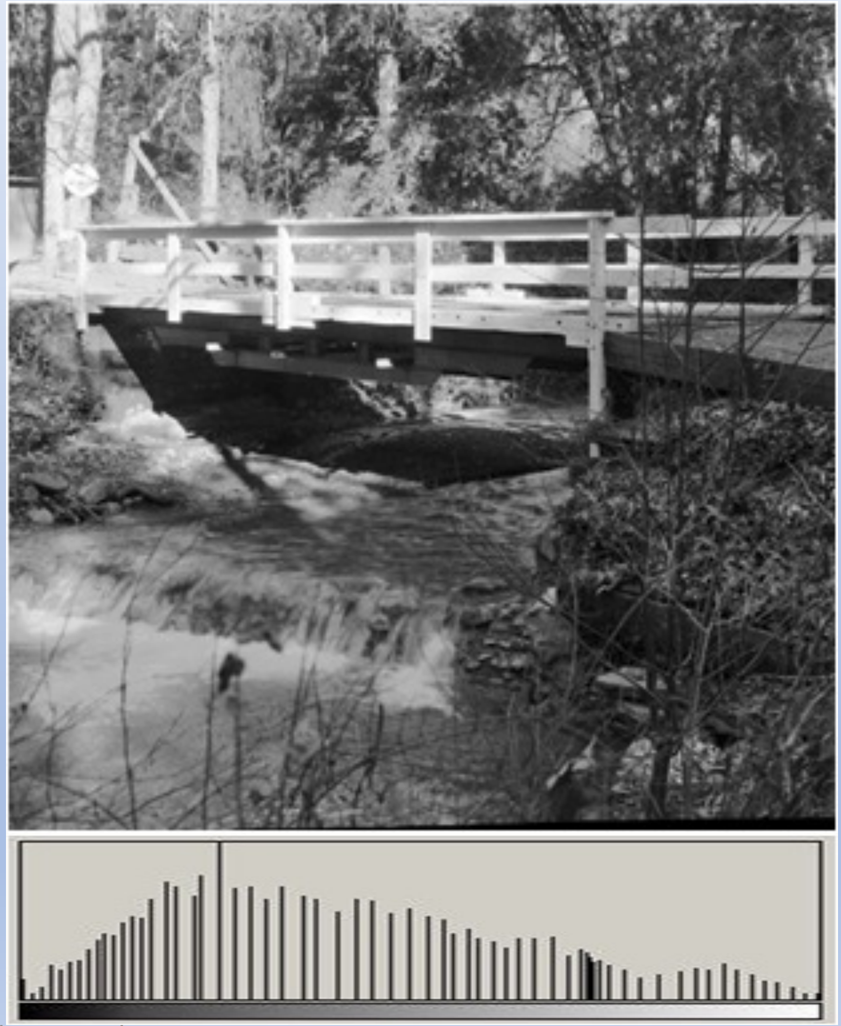
a b
c d

FIGURE 3.9 (a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0,$ and 5.0, respectively.

3. Image Enhancement—simple point processing

Log Transformation: $g = c\log(1+f)$





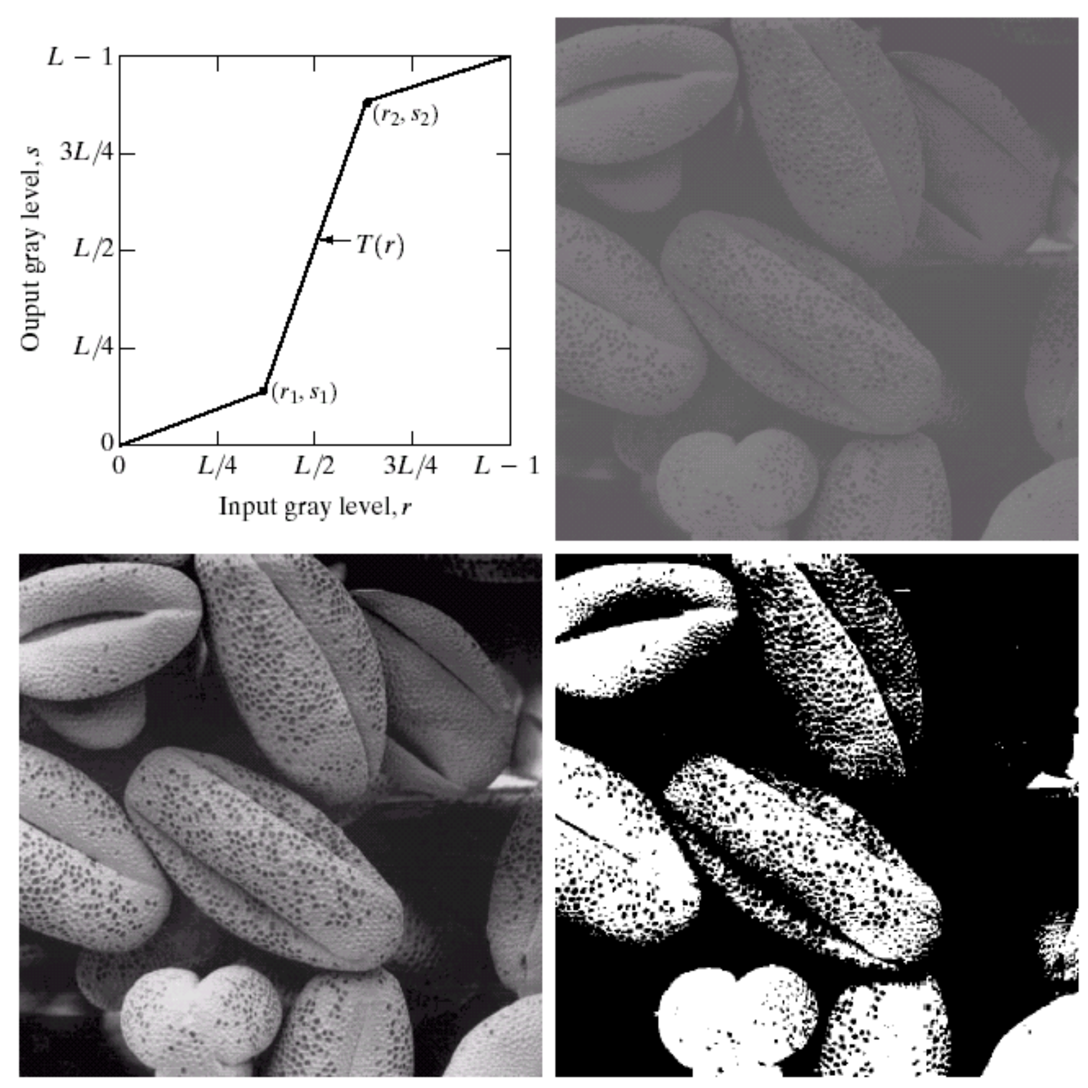
3. Image Enhancement—simple point processing

Piecewise Linear Transformation:

Contrast stretching:

$$g = T(f)$$

$$= \begin{cases} \alpha f, & 0 \le f < a \\ \beta(f-a) + T(a), & a \le f < b \\ \gamma(f-b) + T(b), & b \le f < L \end{cases}$$



3. Image Enhancement—histogram equalization

- Histogram equalization aims to obtain a uniform histogram for the output image $g(x,y)$ by transforming the gray-level f of the input image $f(x,y)$ into g

$$g = T(f)$$

- Histogram equalization algorithm:

$$c(f) = \sum_{t=0}^{f} p_f(t) = \sum_{t=0}^{f} \frac{n_t}{n}, \qquad f = 0, 1, \ldots, L$$

$$g = T(f) = round\left[\frac{c(f) - c_{\min}}{1 - c_{\min}} L\right], \quad c(f) \geq c_{\min}$$

- where t is a dummy variable of the summation. c_{min} is the smallest positive value of all $c(f)$ obtained, round[ ] rounds a real number to an integer. g is approximately uniformly distributed in [0, L] .

3. Image Enhancement—histogram equalization

- Theoretical analysis of the histogram equalization can only be done for continuous variable.
- Let f be the continuous gray value normalize to [0,1]
- Let the transform $g=T(f)$ single-valued, monotonically increasing in $0 \leq g=T(f) \leq 1$.
- The inverse transform is $f=T^{-1}(g)$ should also be single-valued and monotonically increasing.
- From probability theory, if original gray level pdf $p_f(f)$ and $T(f)$ are known, $T^{-1}(g)$ satisfies the above condition, then the transformed gray level pdf $p_g(g)$ is

$$p_g(g) = p_f(f)\frac{df}{dg}$$

3. Image Enhancement—histogram equalization

- Consider that $$g = T(f) = \int_0^f p_f(t)dt$$
- The above function is the cumulative distribution function (cdf) of f. cdf is single-valued and monotonically increasing.

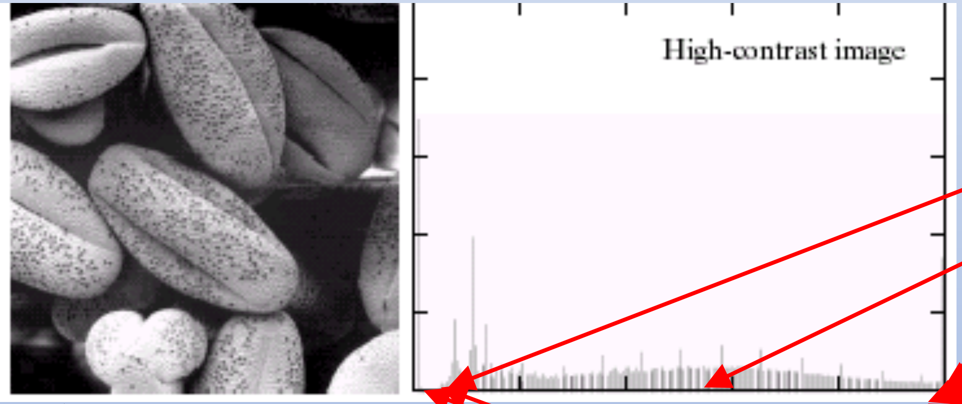
$$\because \frac{dg}{df} = p_f(f) \qquad \therefore p_g(g) = p_f(f)\frac{df}{dg} = p_f(f)\frac{1}{p_f(f)} = 1$$

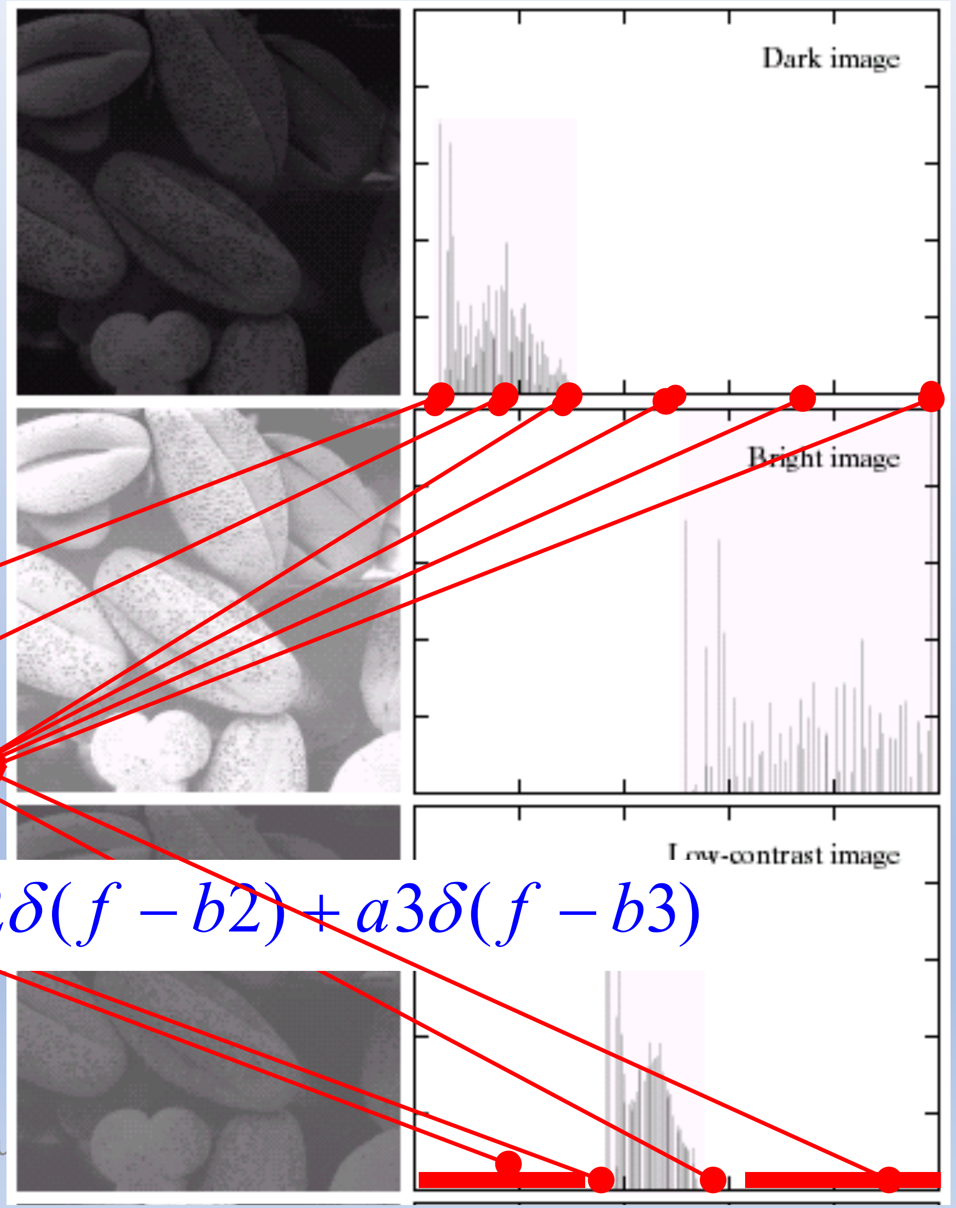
- Therefore, the transformed gray value has uniform distribution.
- The histogram equalization $$c(f) = \sum_{t=0}^{f} p_f(t) = \sum_{t=0}^{f} \frac{n_t}{n}$$
- is the discrete version of $$g = T(f) = \int_0^f p_f(t)dt$$

3. Image Enhancement—histogram equalization

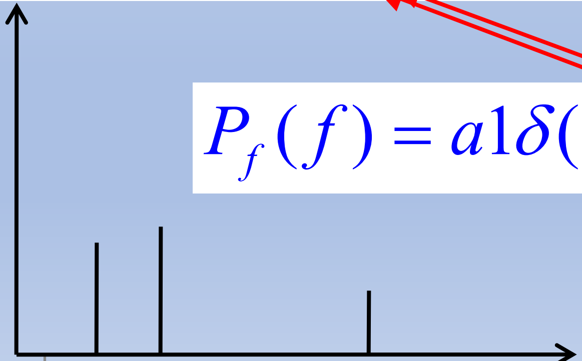
Understand the histogram equalization and its effect on image.

$$c(f)=\sum_{t=0}^{f}p_f(t)=\sum_{t=0}^{f}\frac{n_t}{n}$$

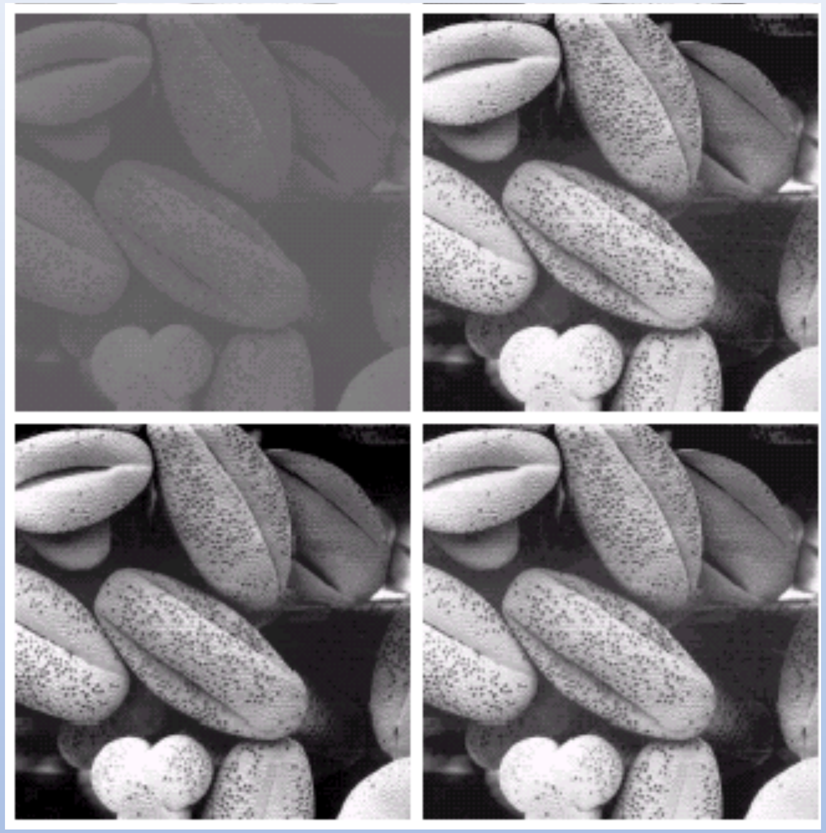




$$P_f(f)=a1\delta(f-b1)+a2\delta(f-b2)+a3\delta(f-b3)$$

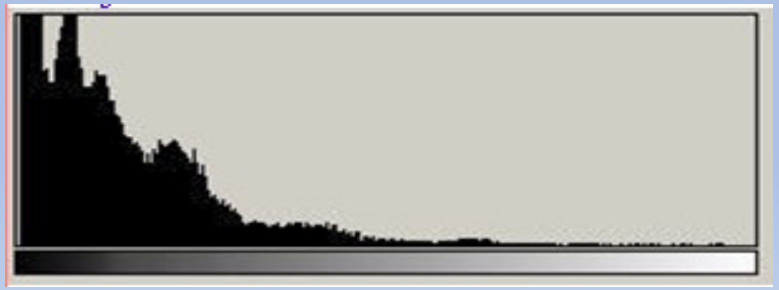


3. Image Enhancement—histogram equalization

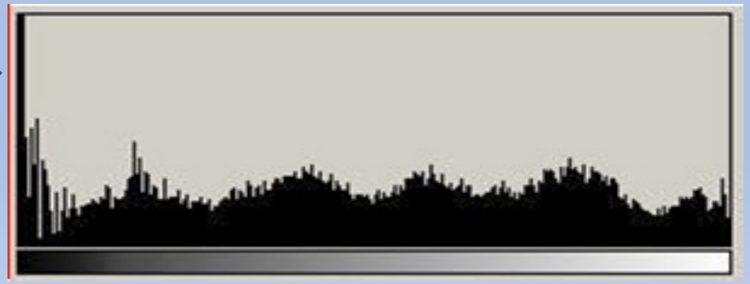


J. Ren, X. Jiang and J. Yuan, "A Chi-Squared-Transformed Subspace of LBP Histogram for Visual Recognition," ***IEEE Trans. Image Processing***, vol. 24, no. 6, pp. 1893-1904, June, 2015.

3. Image Enhancement—histogram equalization



Histogram Equalization



3. Image Enhancement—histogram equalization

- Local histogram equalization

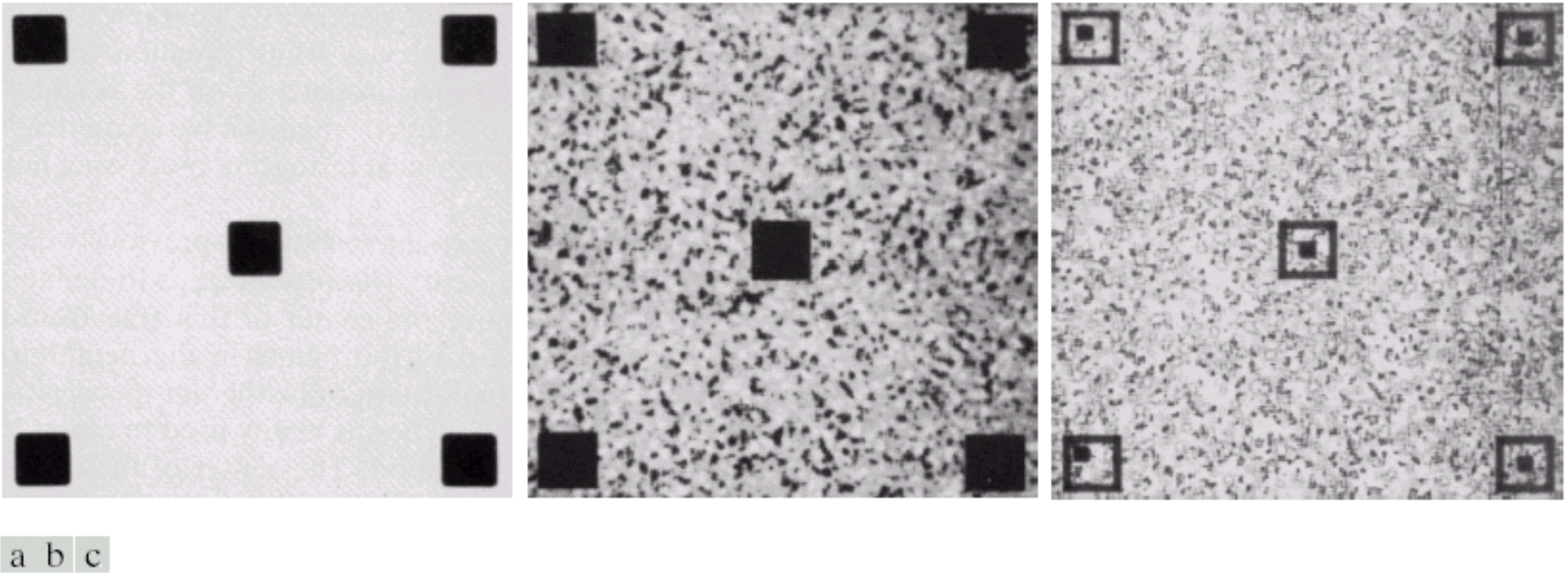


FIGURE 3.23 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a 7×7 neighborhood about each pixel.

3. Image Enhancement—image smoothing

- The basic form of linear image filtering in the spatial and frequency domain can be given as

$$
\begin{array}{ccccc}
g(x,y) & = & f(x,y) & * & h(x,y) \\
\Updownarrow & & \Updownarrow & & \Updownarrow \\
G(u,v) & = & F(u,v) & & H(u,v)
\end{array}
$$

$$g(x,y) = f(x,y) * h(x,y) = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} h(i,j) f(x-i, y-j)$$

$$= \sum_{j=-3}^{3} \sum_{i=-3}^{3} h(i,j) f(x-i, y-j), \quad \text{if } h(x,y) = 0 \quad \text{for } -3 < x, y < 3$$

- At each point (x,y) the response of the filter at that point is calculated as a sum of products of the filter coefficients and the corresponding image pixels in the area spanned by the filter mask centered at (x,y).

3. Image Enhancement—image smoothing

- Image smoothing filters are used for blurring and for noise reduction.
- These filters are also known as averaging or low pass filters.
- Ideal low-pass filter

$$H(u,v)=\begin{cases}1, & \text{if } D(u,v)\le D_0\\ 0, & \text{if } D(u,v)>D_0\end{cases} \qquad D(u,v)=\sqrt{u^2+v^2}$$

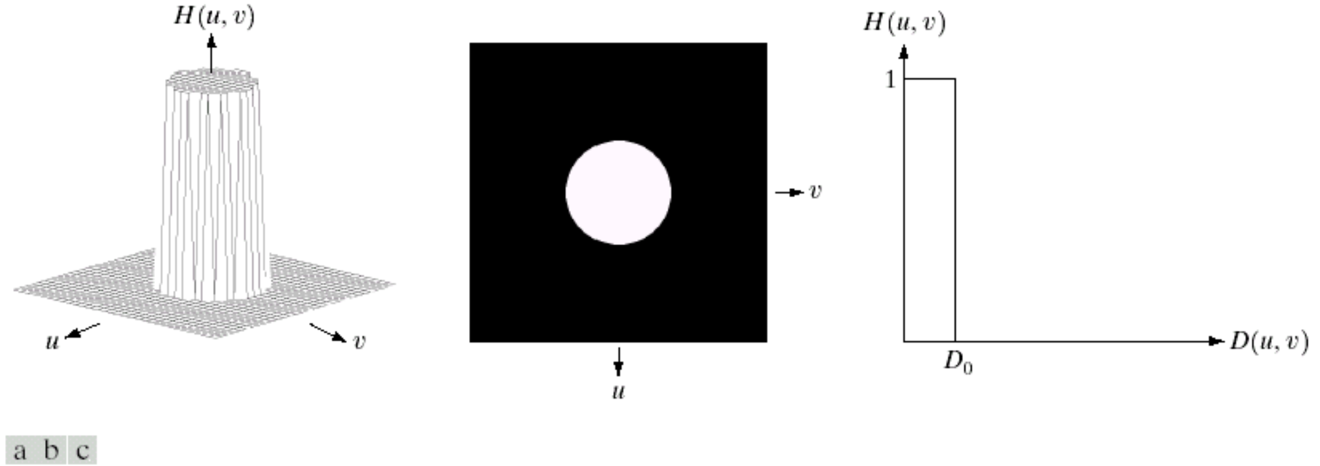


FIGURE 4.10 (a) Perspective plot of an ideal lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

3. Image Enhancement—image smoothing

- Problems of Ideal low-pass filtering

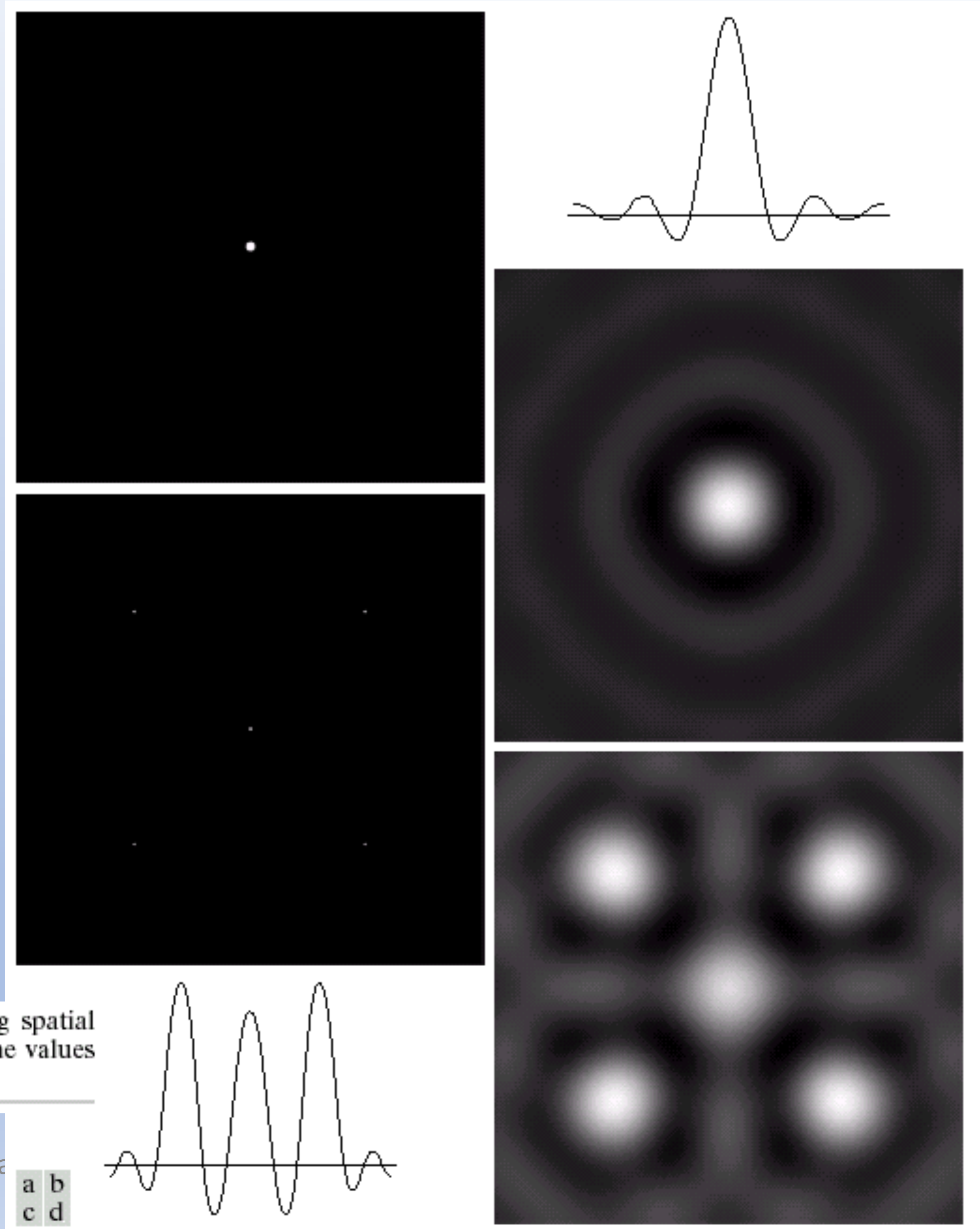
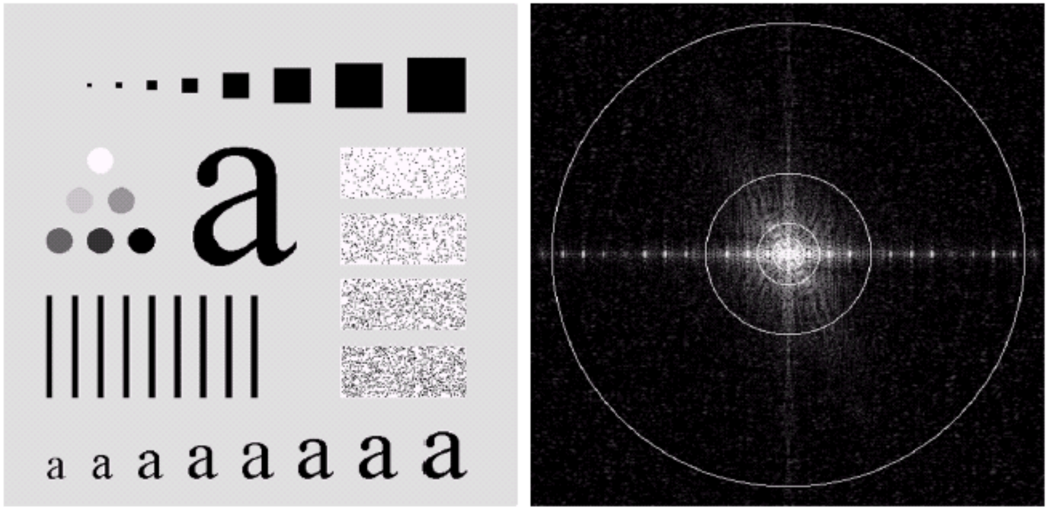


FIGURE 4.13 (a) A frequency-domain ILPF of radius 5. (b) Corresponding spatial filter (note the ringing). (c) Five impulses in the spatial domain, simulating the values of five pixels. (d) Convolution of (b) and (c) in the spatial domain.

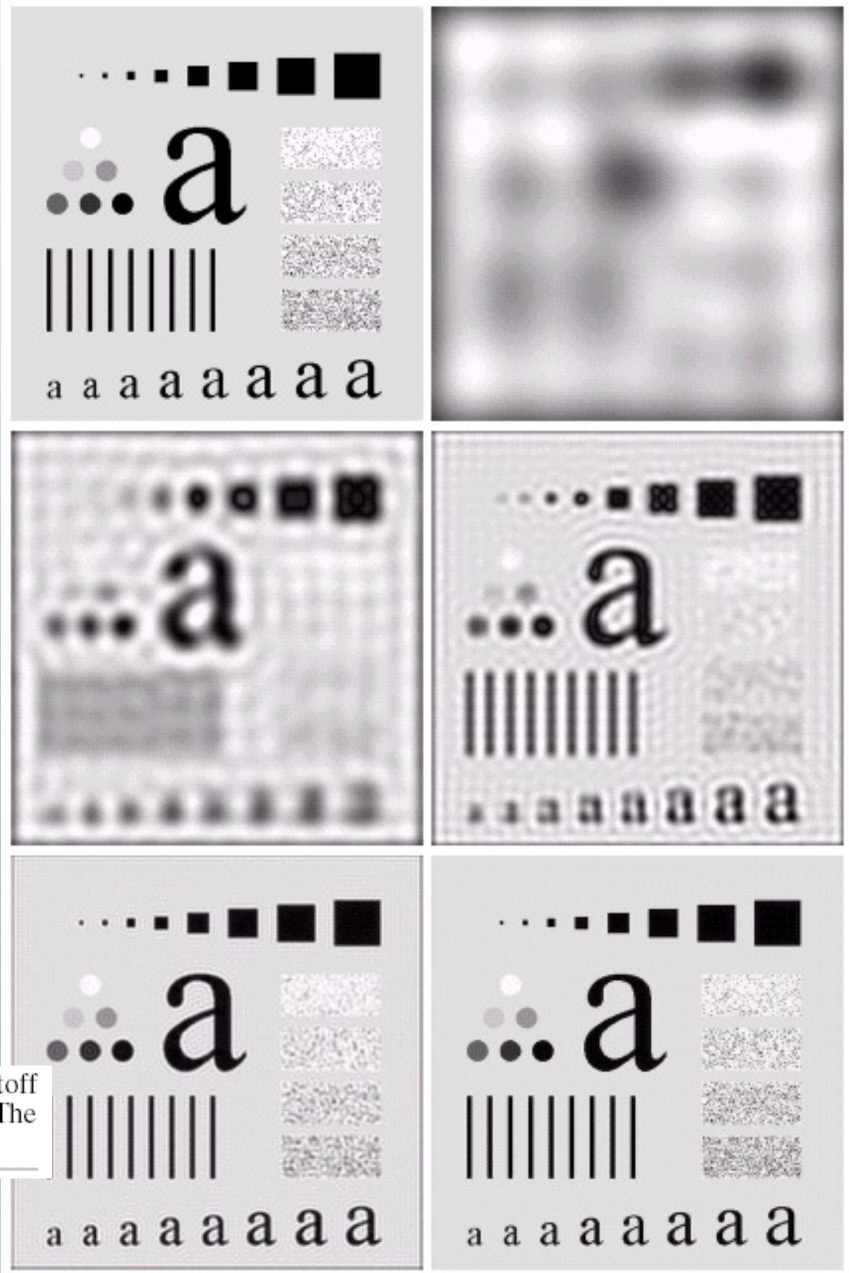
3. Image Enhancement—image smoothing

- Ideal low-pass filtering



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.



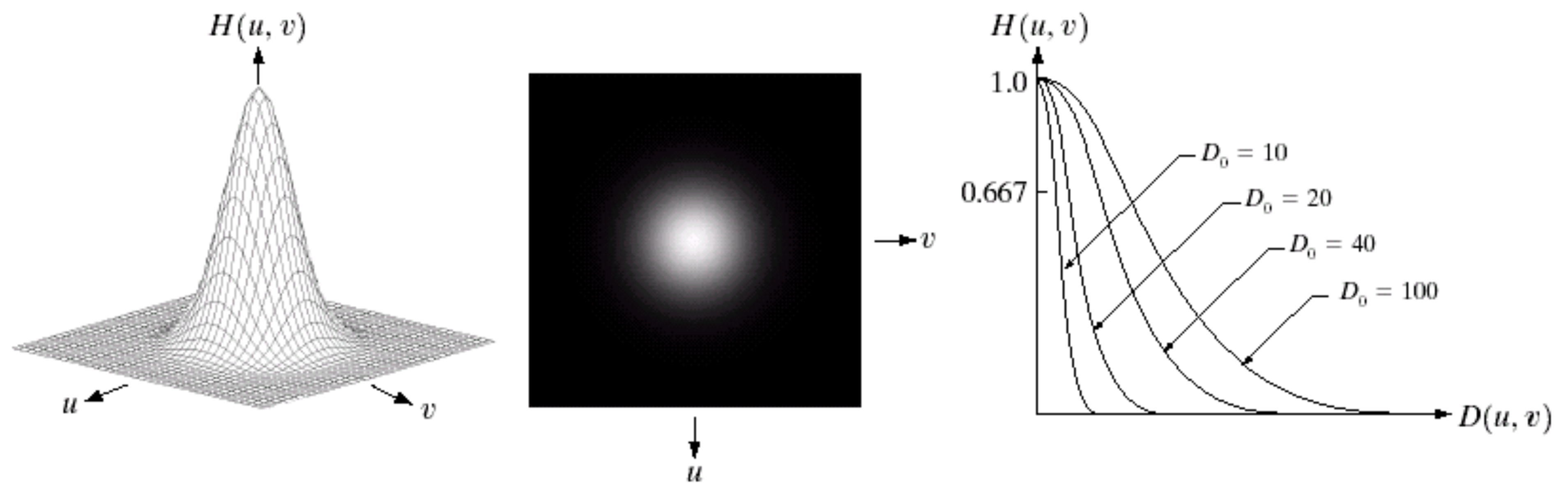
a b
c d
e f

FIGURE 4.12 (a) Original image. (b)–(f) Results of ideal lowpass filtering with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). The power removed by these filters was 8, 5.4, 3.6, 2, and 0.5% of the total, respectively.

3. Image Enhancement—image smoothing

- Gaussian low-pass filtering (GLPF)

$$G(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2D_0}}$$



a b c

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.

3. Image Enhancement—image smoothing

- Gaussian low-pass filtering (GLPF)

$$G(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2D_0}}$$

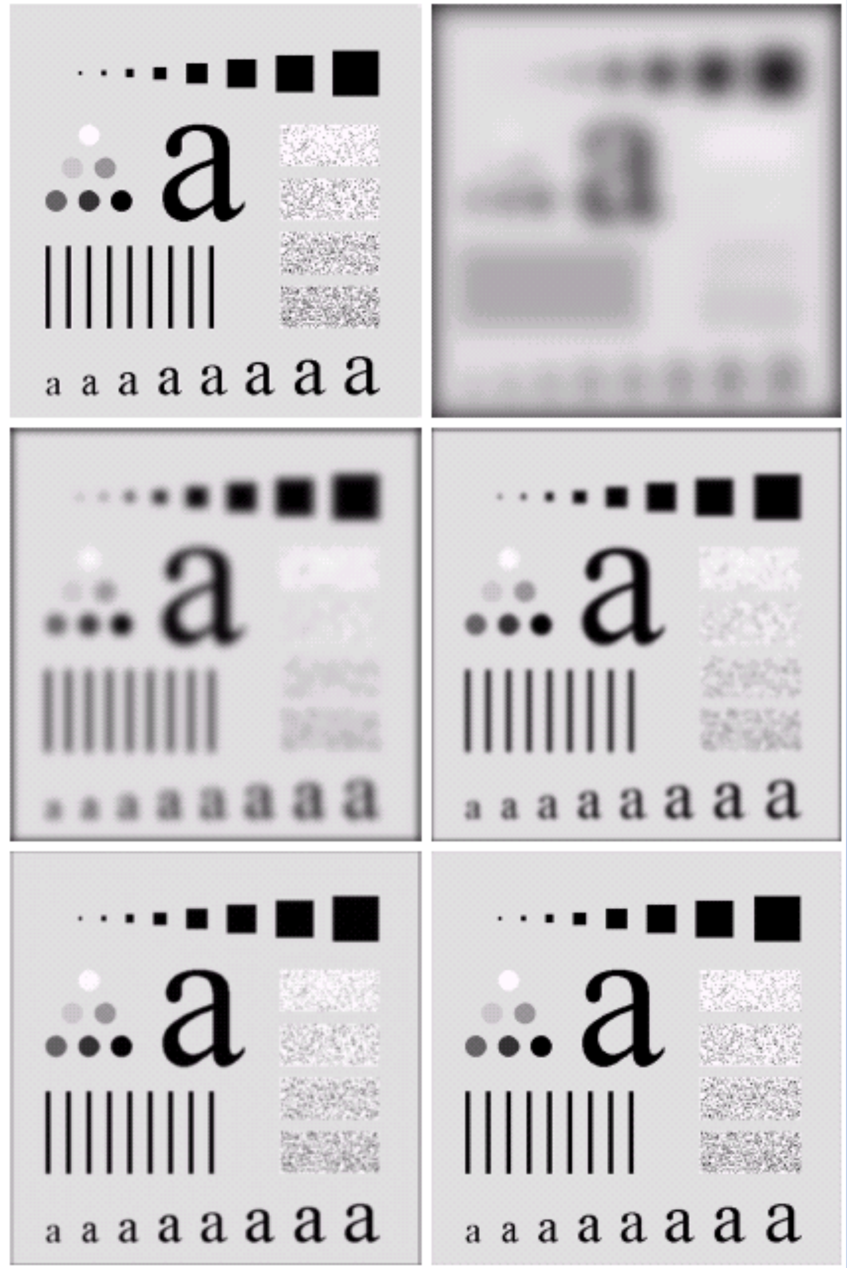
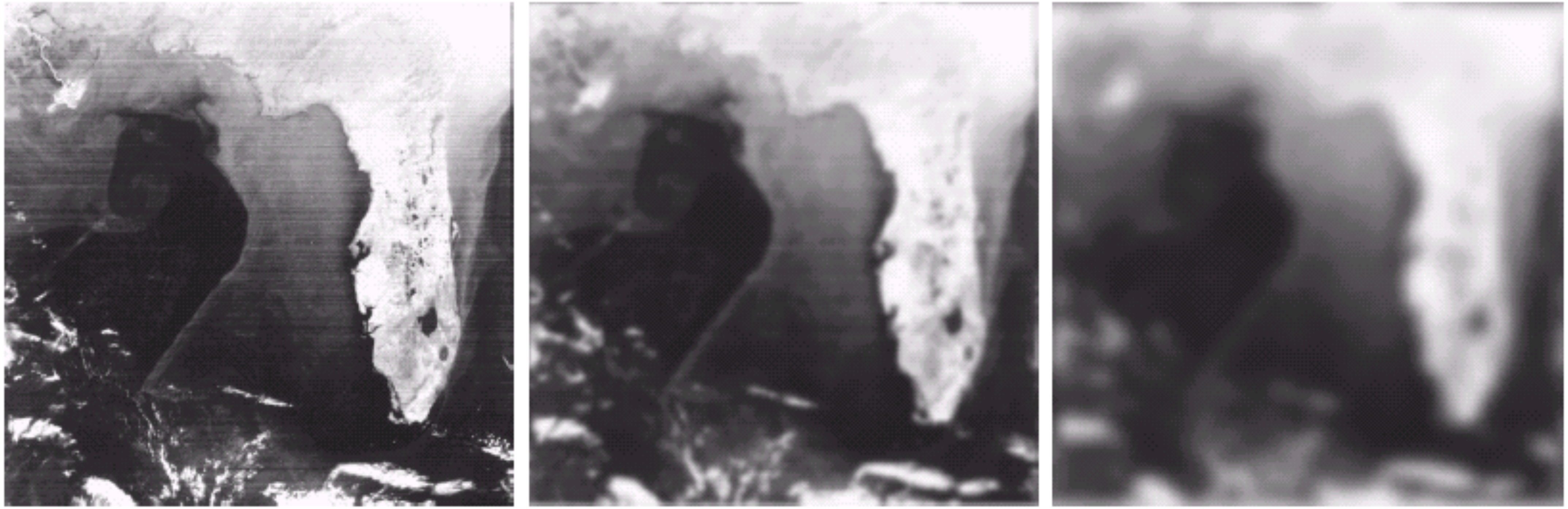


FIGURE 4.18 (a) Original image. (b)–(f) Results of filtering with Gaussian lowpass filters with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Figs. 4.12 and 4.15.

a	b
c	d
e	f

3. Image Enhancement—image smoothing



a b c

FIGURE 4.21 (a) Image showing prominent scan lines. (b) Result of using a GLPF with $D_0 = 30$. (c) Result of using a GLPF with $D_0 = 10$. (Original image courtesy of NOAA.)

3. Image Enhancement—image smoothing

a b c

FIGURE 4.20 (a) Original image (1028 × 732 pixels). (b) Result of filtering with a GLPF with $D_0 = 100$. (c) Result of filtering with a GLPF with $D_0 = 80$. Note reduction in skin fine lines in the magnified sections of (b) and (c).

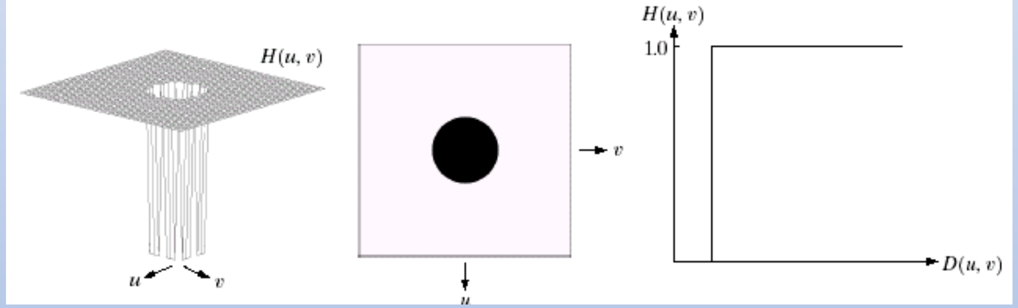
3. Image Enhancement—image sharpening

- High-pass filtering

$$H_{hp}(u,v)=1-H_{lr}(u,v)$$

- Ideal high-pass filter

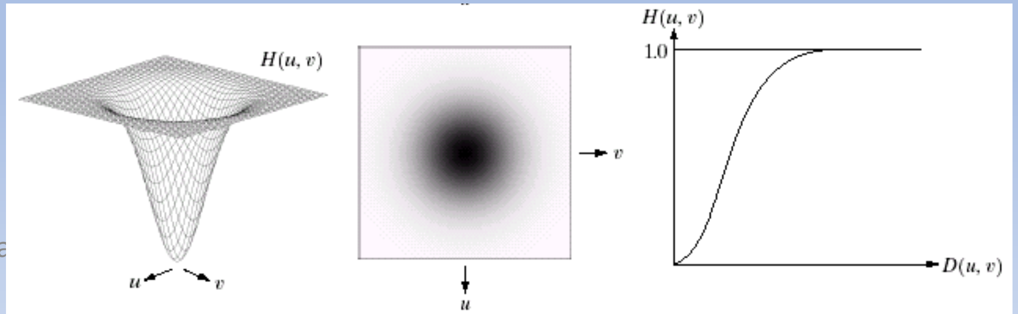
$$H(u,v)=\begin{cases}0, & \text{if } D(u,v)\le D_0\\ 1, & \text{if } D(u,v)>D_0\end{cases}$$



$$D(u,v)=\sqrt{u^2+v^2}$$

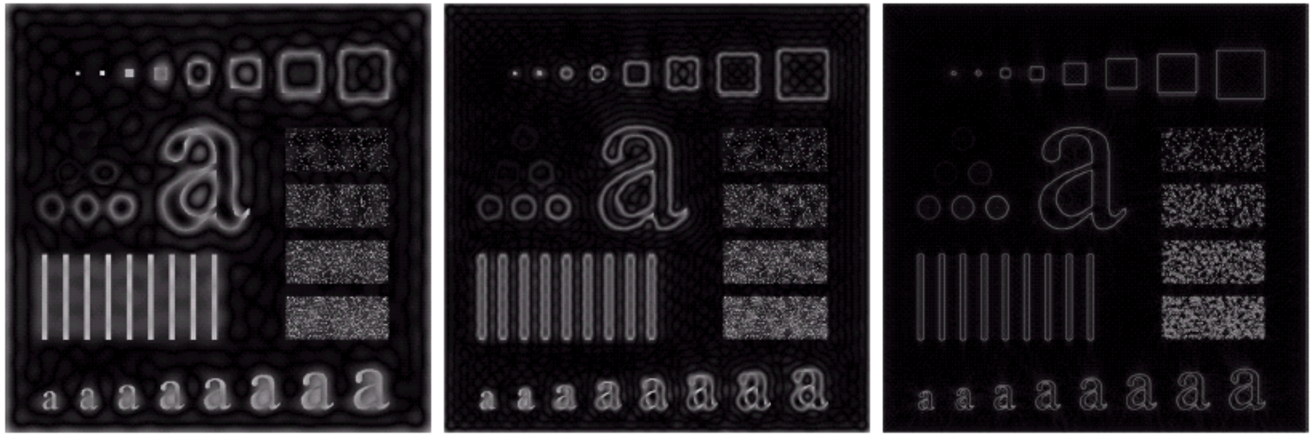
- Gaussian high-pass filter (GHPF)

$$G(u,v)=1-e^{-\frac{u^2+v^2}{2D_0}}$$



3. Image Enhancement—image sharpening

Ideal high-pass filter



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).

Gaussian High-pass filt

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.

3. Image Enhancement—image sharpening

- High-boost filtering

$$f_{hb}(x,y) = Af(x,y) - f_{lp}(x,y)$$

where $f_{lp}(x,y)$ is a smoothed version of $f(x,y)$

by a lowpass filter,

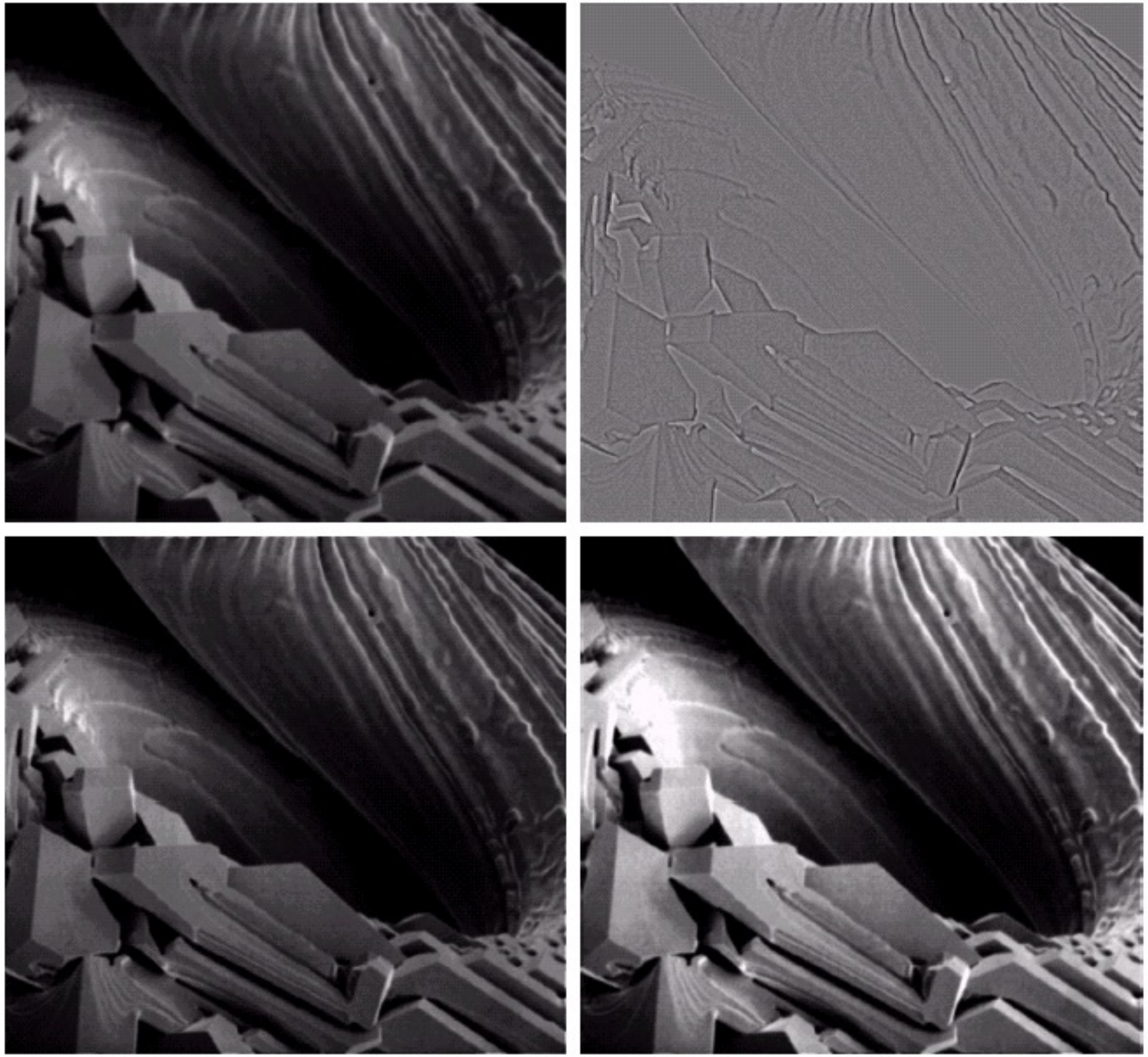
$\mathrm{A} \geq 1$

$$f_{hb}(x,y) = (A-1)f(x,y) + f(x,y) - f_{lp}(x,y)$$

$$f_{hb}(x,y) = (A-1)f(x,y) + f_{hp}(x,y)$$

3. Image Enhancement—image sharpening

High-boost filtering



a b
c d

FIGURE 4.29 Same as Fig. 3.43, but using frequency domain filtering. (a) Input image. (b) Laplacian of (a). (c) Image obtained using Eq. (4.4-17) with $A = 2$. (d) Same as (c), but with $A = 2.7$. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

3. Image Enhancement—image sharpening

Examples of combining image enhancement

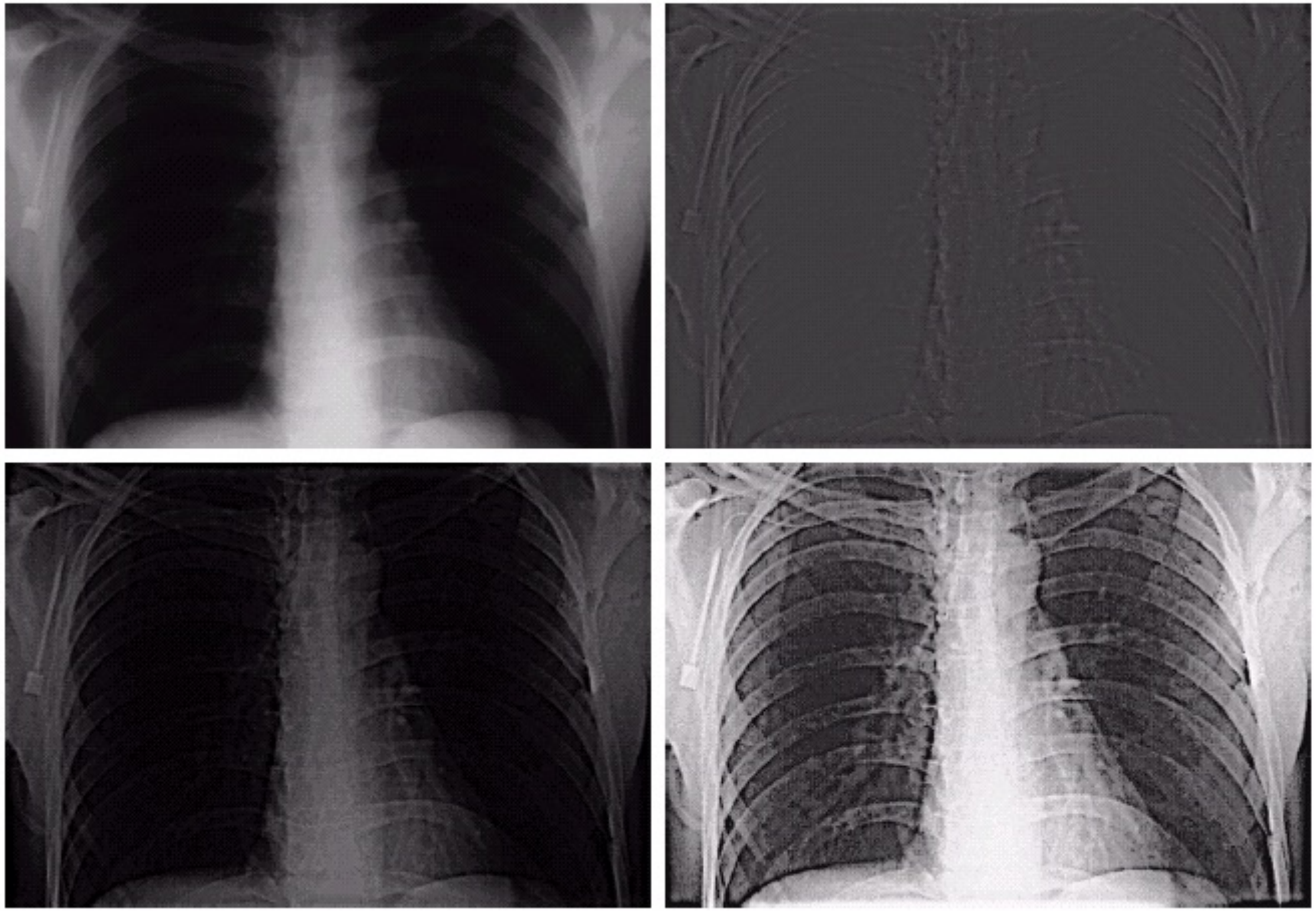


FIGURE 4.30
(a) A chest X-ray image. (b) Result of Butterworth highpass filtering. (c) Result of high-frequency emphasis filtering. (d) Result of performing histogram equalization on (c). (Original image courtesy Dr. Thomas R. Gest, Division of Anatomical Sciences, University of Michigan Medical School.)

3. Image Enhancement—image sharpening

Examples of combining image enhancem ent