COL783: Digital Image Processing

Prem Kalra
pkalra@cse.iitd.ac.in
http://www.cse.iitd.ac.in/~pkalra/col783

Department of Computer Science and Engineering
Indian Institute of Technology Delhi

Image Enhancement in Frequency Domain

Fourier Transform

Main Idea: A periodic function can be decomposed into a summation of sine and cosine functions

It may be easier and natural to apply some operations in the frequency domain

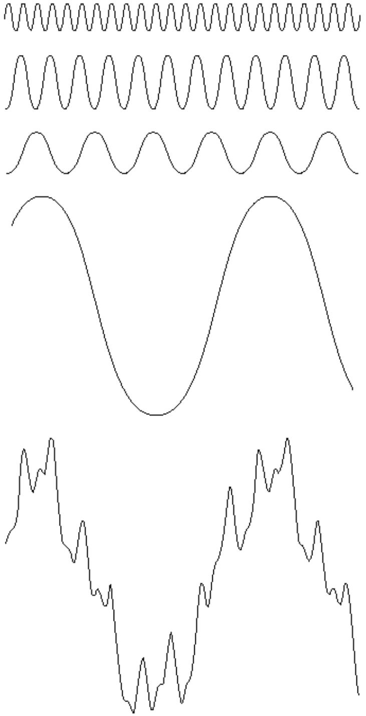


FIGURE 4.1 The function at the bottom is the sum of the four functions above it. Fourier's idea in 1807 that periodic functions could be represented as a weighted sum of sines and cosines was met with skepticism.

(Source: Gonzalez and Woods)

Image Enhancement in Frequency Domain

Fourier Transform

In continuous domain

1-D:

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

F(u) and f(x) form a Fourier pair.

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

2-D:

$$F(u,v) = \iint f(x,y) e^{-j2\pi(ux+vy)} dx\, dy$$

$$f(x,y) = \iint F(u,v) e^{j2\pi(ux+vy)} du\, dv$$

(Source: Gonzalez and Woods)

Image Enhancement in Frequency Domain

Discrete Fourier Transform

- Suppose

$$\mathbf{f} = [f_0, f_1, f_2, \ldots, f_{N-1}]$$

is a sequence of length *N*

$$\mathbf{F} = [F_0, F_1, F_2, \ldots, F_{N-1}]$$

where

$$F_u = \frac{1}{N} \sum_{x=0}^{N-1} \exp\left[-2\pi i \frac{xu}{N}\right] f_x$$

Image Enhancement in Frequency Domain

Discrete Fourier Transform

- For *M x N* matrix, forward and inverse fourier transforms can be written

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

where

- *x* indices go from *0… M – 1* (*x* cycles over distance *M*)
- *y* indices go from *0… N – 1* (*y* cycles over distance *N*)

Image Enhancement in Frequency Domain

Discrete Fourier Transform

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

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

Source: https://web.cs.wpi.edu/~emmanuel/courses/cs545/S14/slides/lecture10.pdf

Image Enhancement in Frequency Domain

Discrete Fourier Transform

- **DFT as spatial filter:** These values are just basis functions (are independent of *f* and *F*)

$$\exp\left[\pm 2\pi i\left(\frac{xu}{M}+\frac{yv}{N}\right)\right]$$

  - Can be computed in advance, put into formulas later
  - Implies each value *F(u,v)* obtained by multiplying every value of *f(x,y)* by a fixed value, then adding up all results (similar to a filter!)

Image Enhancement in Frequency Domain

Discrete Fourier Transform

2-D DFT

Often it is convenient to consider a symmetric transform:

$$v(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} u(n) W_N^{kn} \quad \text{and}$$

$$u(n) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} v(k) W_N^{-kn}$$

In 2-D: consider a NXN image

$$v(k,l) = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} u(m,n)\, W_N^{km} W_N^{ln},$$

$$u(m,n) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} v(k,l)\, W_N^{-km-ln}$$

(Source: Gonzalez and Woods)

Discrete Fourier Transform

Properties

Seperability

- Notice that Fourier transform "filter elements" can be expressed as products

$$\exp\left[2\pi i\left(\frac{xu}{M}+\frac{yv}{N}\right)\right] = \exp\left[2\pi i\frac{xu}{M}\right]\exp\left[2\pi i\frac{yv}{N}\right]$$

2D DFT   1D DFT (row)   1D DFT (column)

- Formula above can be broken down into simpler formulas for 1D DFT

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

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

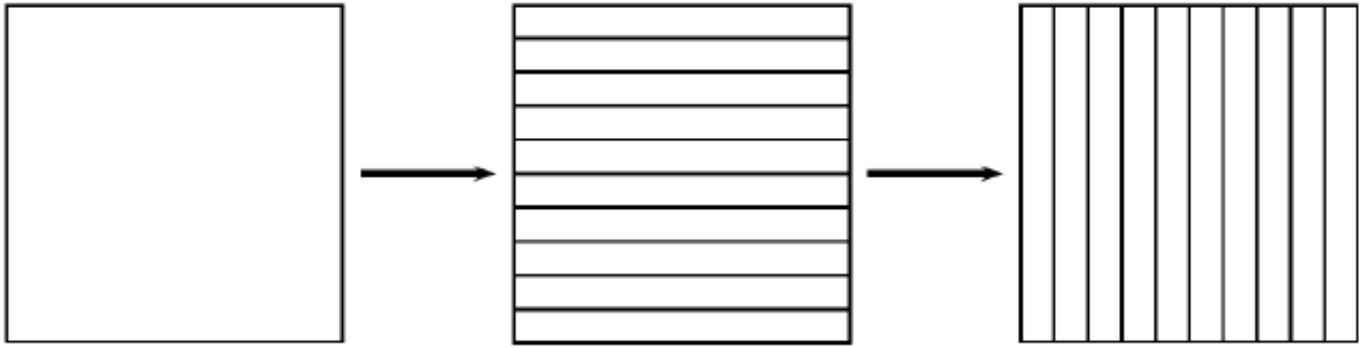
Discrete Fourier Transform

Properties

$$F(u,v) = \frac{1}{N}\sum_{x=0}^{N-1} exp[-\frac{2i\pi ux}{N}]\sum_{y=0}^{N-1} f(x,y)exp[-\frac{2i\pi vy}{N}]$$

Seperability

- Using their separability property, can use 1D DFTs to calculate rows then columns of 2D Fourier Transform



(a) Original image (b) DFT of each row of (a) (c) DFT of each column of (b)

Discrete Fourier Transform

Properties

- **Linearity:** DFT of a sum is equal to sum (or multiplication) of the individual DFT's

$$\mathcal{F}(f+g) = \mathcal{F}(f)+\mathcal{F}(g)$$

$$\mathcal{F}(kf) = k\mathcal{F}(f) \quad \textbf{\textit{k} is a scalar}$$

- Useful property for dealing with degradations that can be expressed as a sum (e.g. noise)

$$d = f + n$$

Where *f* is original image, *n* is the noise, *d* is degraded image

- We can find fourier transform as:

$$\mathcal{F}(d) = \mathcal{F}(f)+\mathcal{F}(n)$$

- Noise can be removed/reduced by modifying transform of *n*

Discrete Fourier Transform

Properties

Translation

$$f(x,y)exp[\frac{2i\pi(u_0x+v_0y)}{N}] \Leftrightarrow F(u-u_0, v-v_0)$$

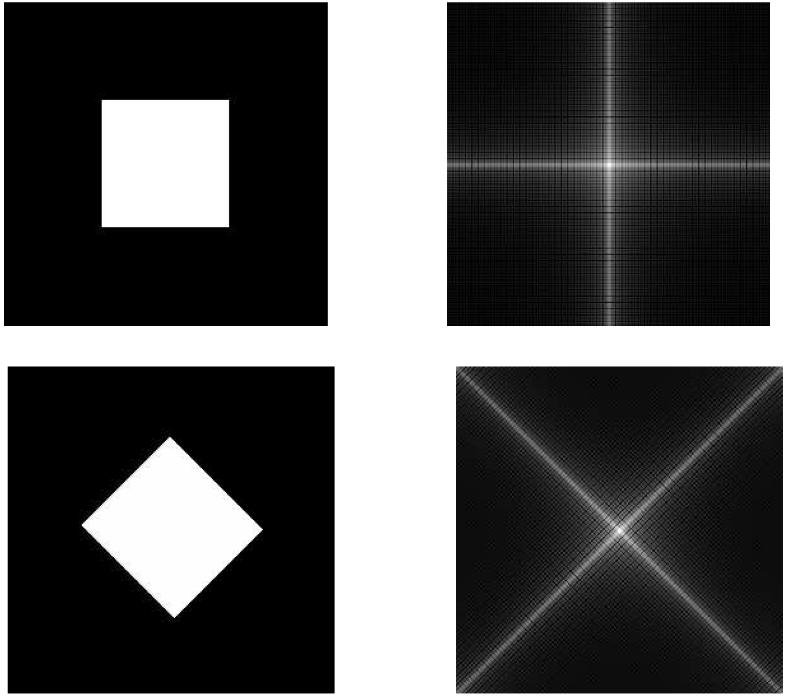
$$f(x-x_0, y-y_0) \Leftrightarrow F(u,v)exp[-\frac{2i\pi(ux_0+vy_0)}{N}]$$

Discrete Fourier Transform

Properties

Rotation

$$f(r, \theta + \theta_0) \Leftrightarrow F(w, \phi + \theta_0)$$



Source: https://web.cs.wpi.edu/~emmanuel/courses/cs545/S14/slides/lecture10.pdf

Discrete Fourier Transform

Properties

Not commutative

$$\mathcal{F}[f_1(x,y).f_2(x,y)] \neq \mathcal{F}[f_1(x,y)].\mathcal{F}[f_2(x,y)]$$

Discrete Fourier Transform

Properties

Periodicity and Conjugate Symmetry

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

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

Discrete Fourier Transform

Properties

- Recall that:

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

- The value *F(0,0)* of the DFT is called the **dc coefficient**
- If we put *u* = *v* = 0, then

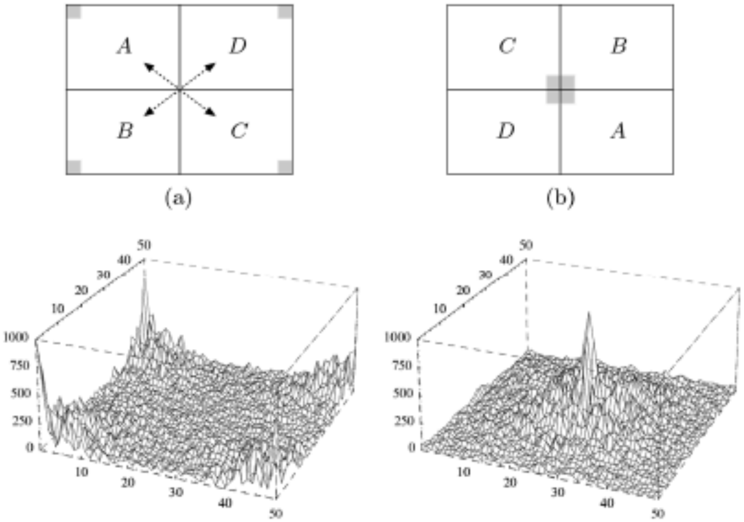
$$F(0,0) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \exp(0) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)$$

- Essentially *F(0,0)* is the sum of all terms in the original matrix

Discrete Fourier Transform

Properties

- *F(0,0)* at top left corner
- For display, convenient to have DC component in center
- Just swap four quadrants of Fourier transform



Swap 4 quadrants to center DC component

DFT spectrum after centering

Source: https://web.cs.wpi.edu/~emmanuel/courses/cs545/S14/slides/lecture10.pdf

Discrete Fourier Transform

Properties

Convolution

$$f(x) * h(x) = \int_{-\infty}^{\infty} f(\alpha)h(x-\alpha)d\alpha$$

$$f(x) * h(x) = \sum f(\alpha)h(x-\alpha)$$

$$\mathcal{F}[f(x) * h(x)] = \mathcal{F}[f(x)].\mathcal{F}[h(x)]$$

Discrete Fourier Transform

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

Discrete Fourier Transform

TABLE 4.1
(continued)

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.

Image Enhancement in Fourier Domain

- one reason for using Fourier transform in image processing is due to convolution theorem
- Spatial convolution can be performed by element-wise multiplication of the Fourier transform by suitable "filter matrix"

Image Enhancement in Fourier Domain

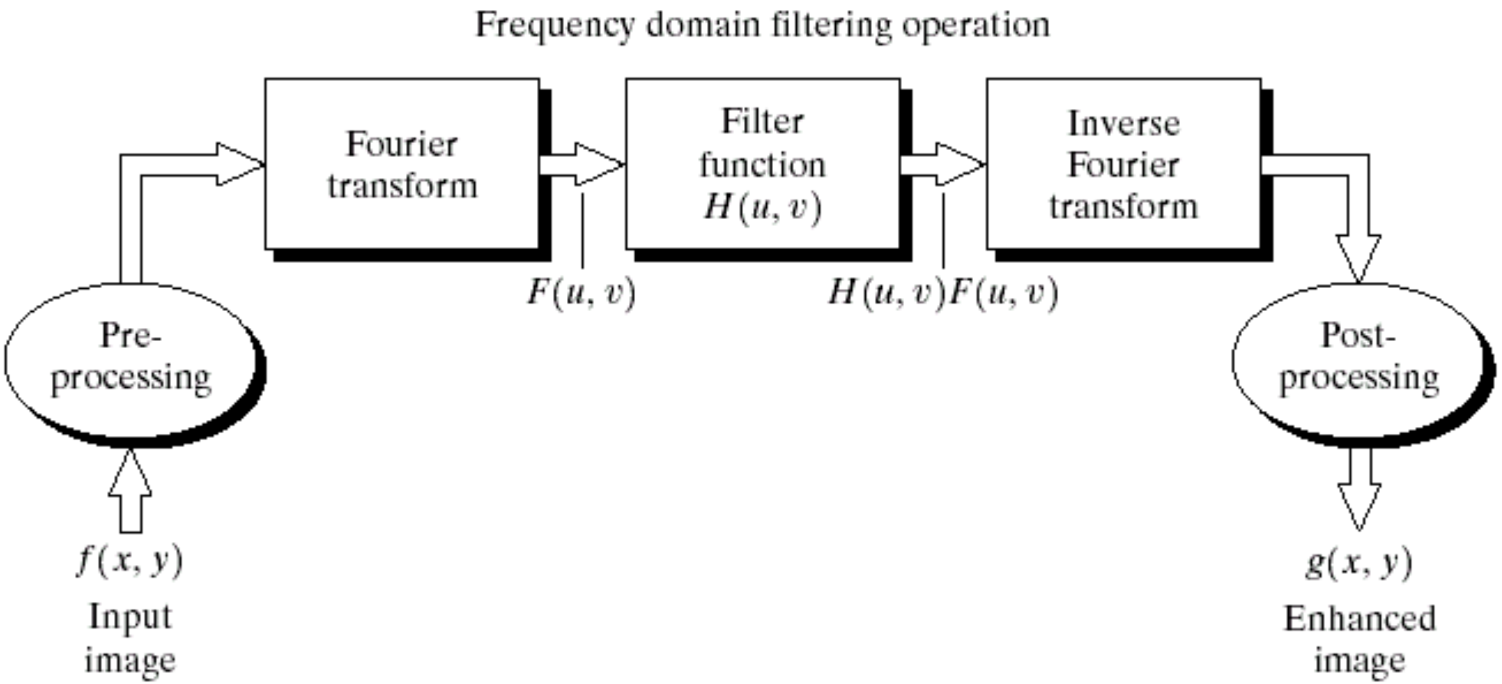


FIGURE 4.5 Basic steps for filtering in the frequency domain.

(Source: Gonzalez and Woods)

Image Enhancement in Fourier Domain

LPF :

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

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

$$D(u,v) = \sqrt{u^2 + v^2} \quad \text{(Circularly symmetric)}$$

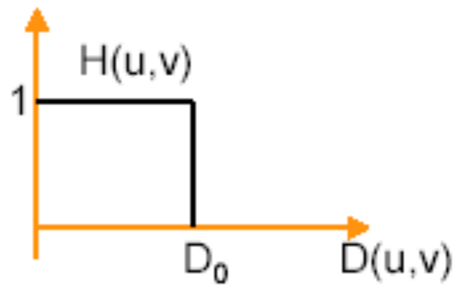
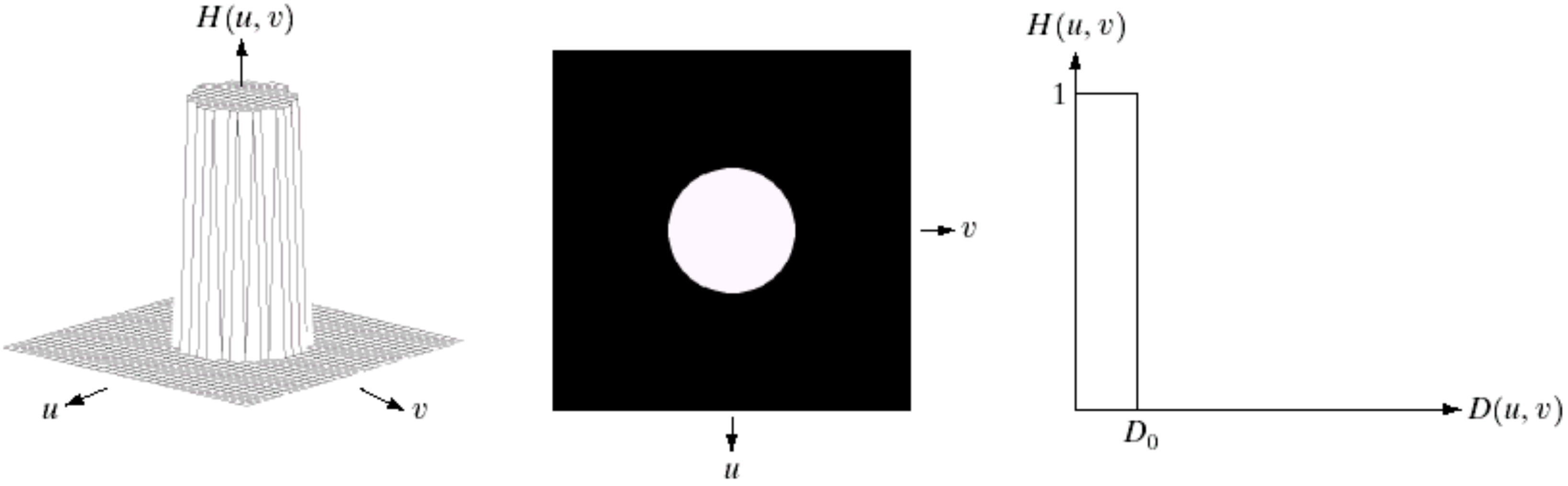


Image Enhancement in Fourier Domain

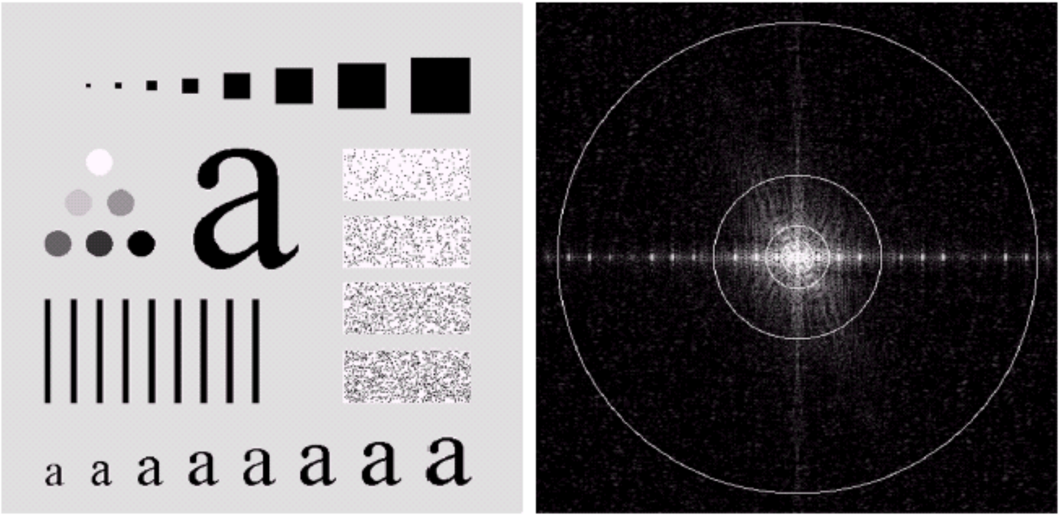


a b c

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

(Source: Gonzalez and Woods)

Image Enhancement in Fourier Domain

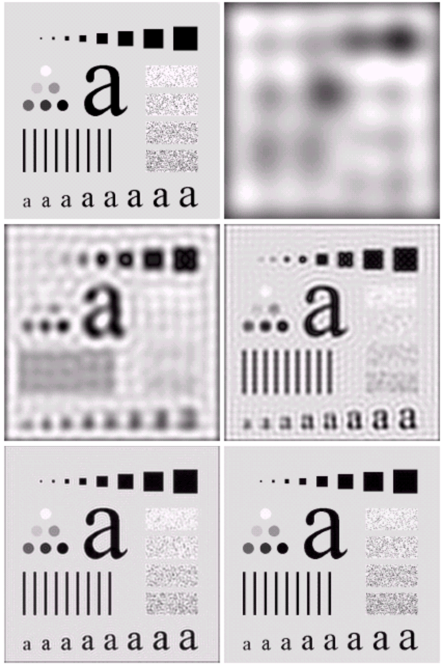


a b

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.

(Source: Gonzalez and Woods)

Image Enhancement in Fourier Domain

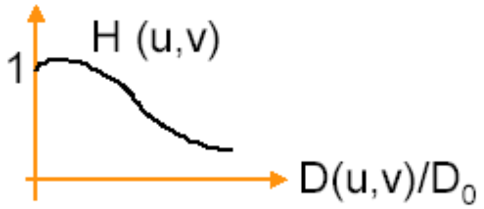


a b
c d
e f

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.

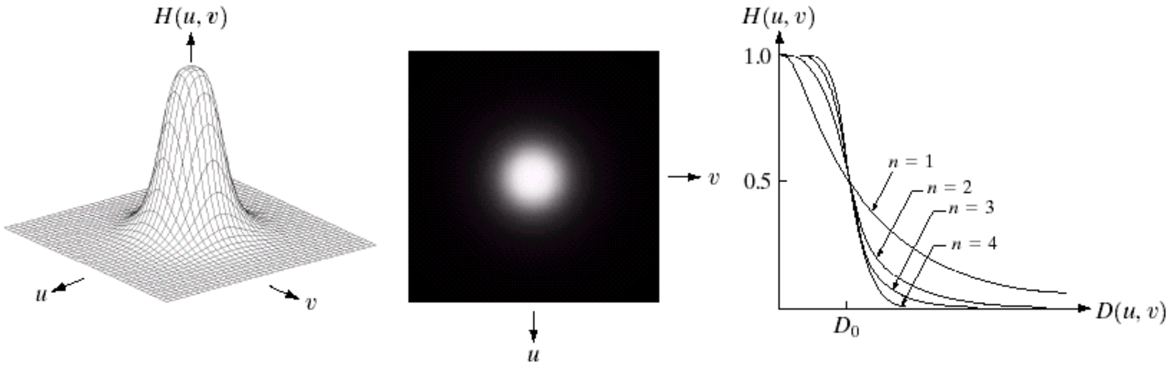
(Source: Gonzalez and Woods)

Image Enhancement in Fourier Domain



Butterworth filter:

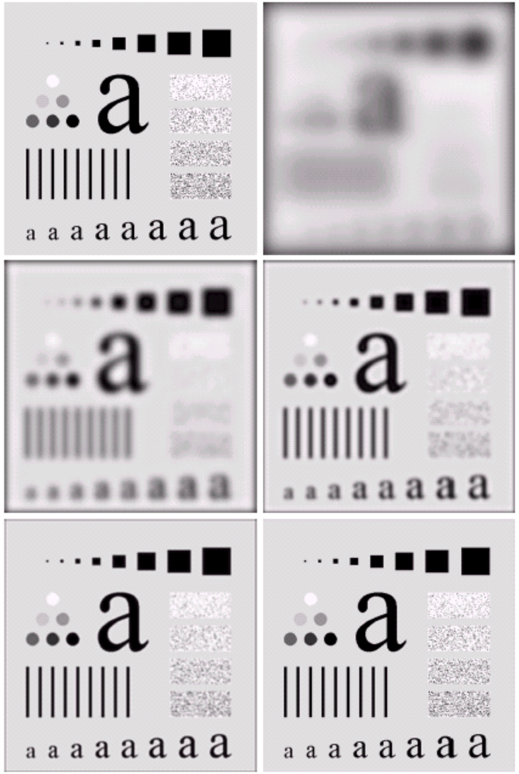
$$H(u,v) = \frac{1}{1+[D(u,v)/D_0]^{2n}}$$



a b c

FIGURE 4.14 (a) Perspective plot of a Butterworth lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.

Image Enhancement in Fourier Domain



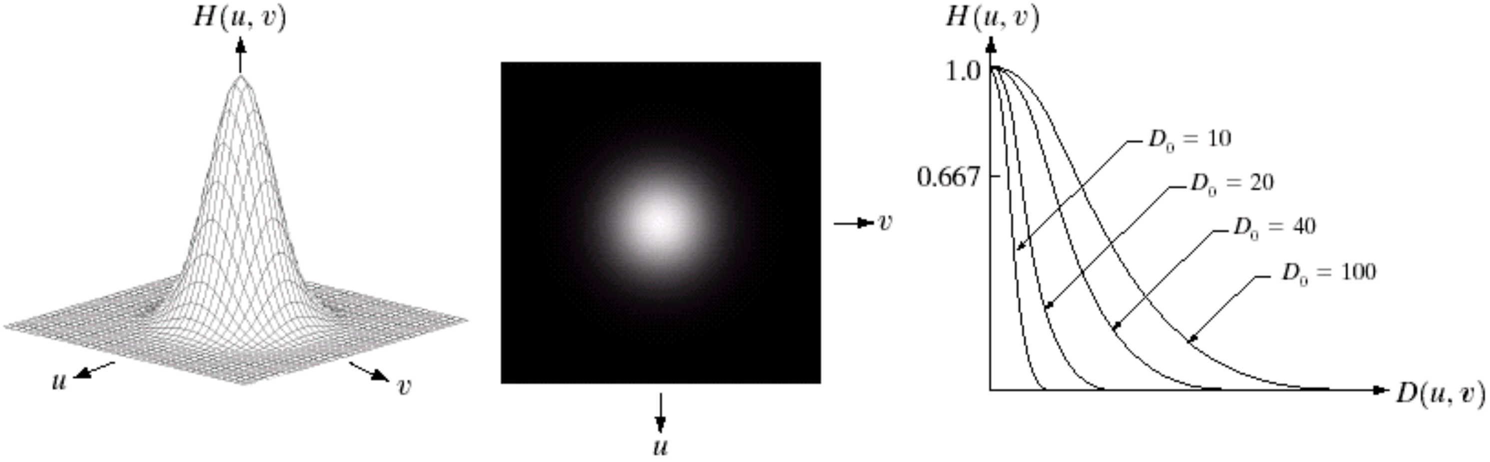
a b
c d
e f

FIGURE 4.15 (a) Original image. (b)–(f) Results of filtering with BLPFs of order 2, with cutoff frequencies at radii of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Fig. 4.12.

(Source: Gonzalez and Woods)

Image Enhancement in Fourier Domain

$$H(u,v) = \exp(-D^2(u,v)/2\sigma^2)$$



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.

Image Enhancement in Fourier Domain

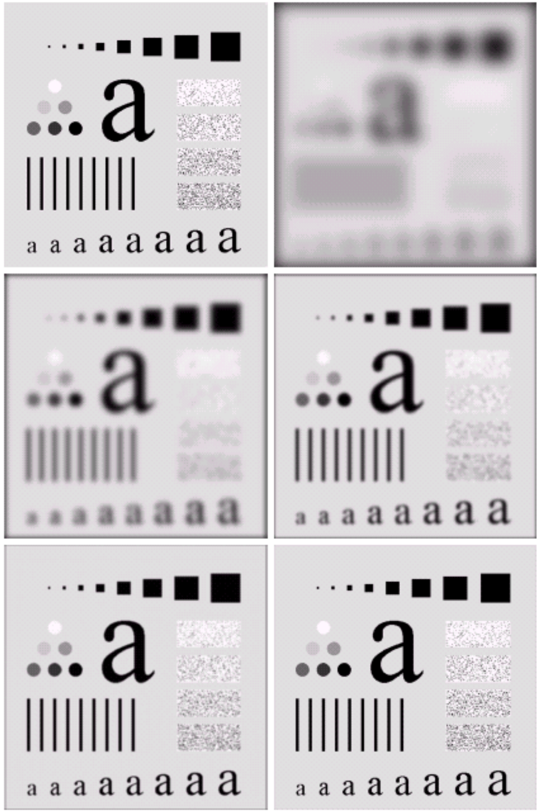


FIGURE 4.18 (a) Original image. (b)–(f) Results of filtering with Gaussian lowpass filters with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). Compare with Figs. 4.12 and 4.15.

(Source: Gonzalez and Woods)

Image Enhancement in Fourier Domain

HPF :

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

Butterworth filter:

$$H(u,v) = \frac{1}{1 + [D_0 / D(u,v)]^{2n}}$$

Image Enhancement in Fourier Domain

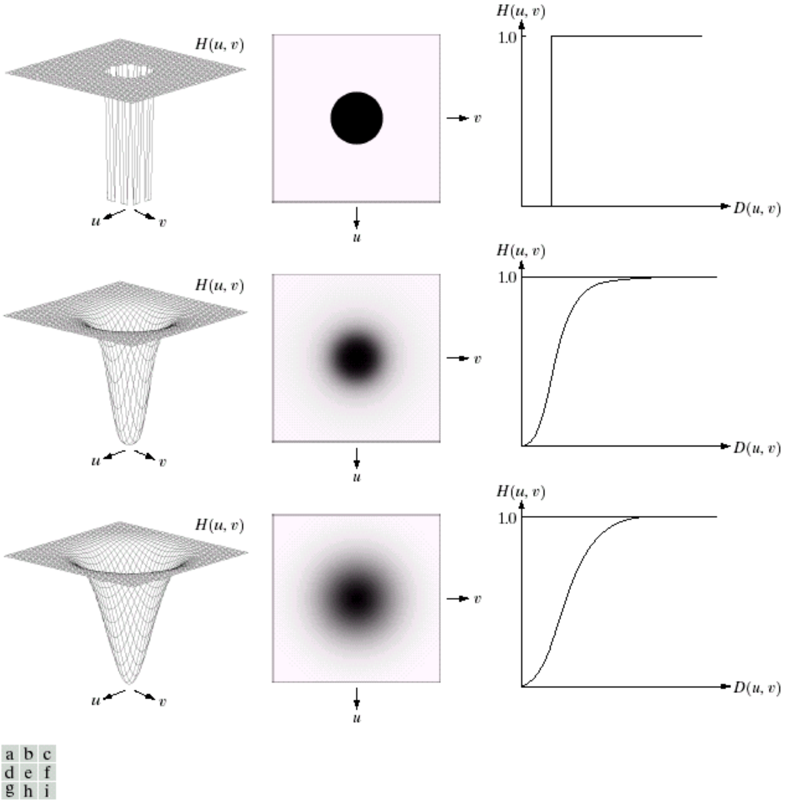
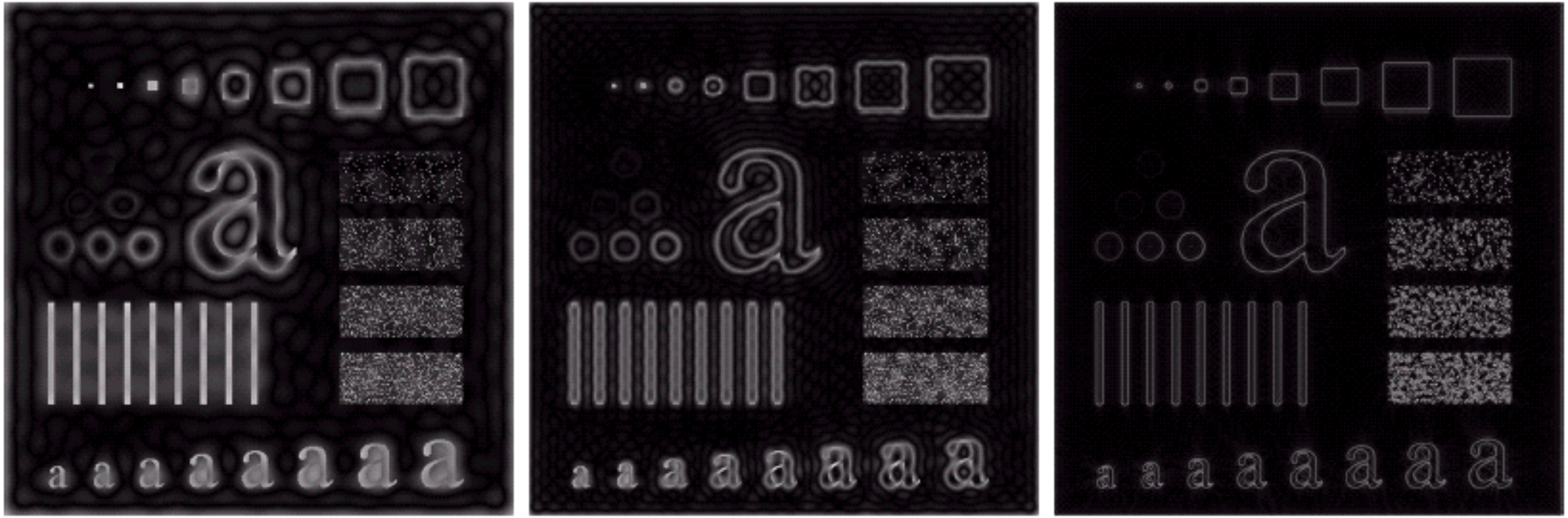


FIGURE 4.22 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

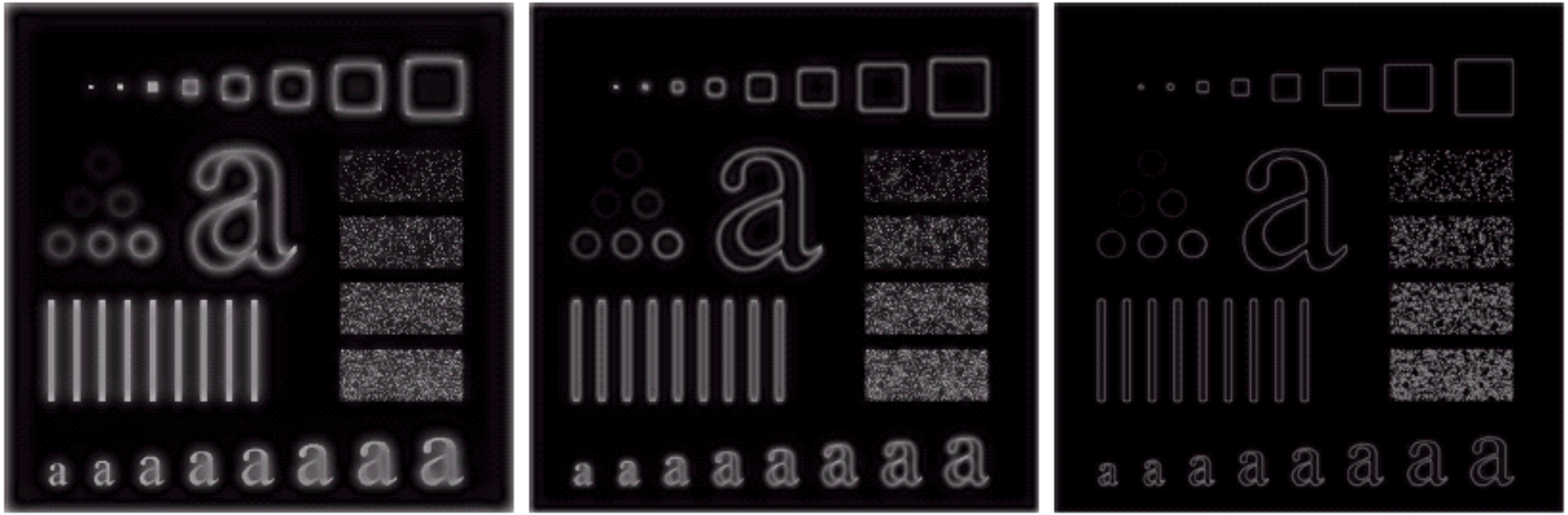
Image Enhancement in Fourier Domain



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).

Image Enhancement in Fourier Domain



a b c

FIGURE 4.25 Results of highpass filtering the image in Fig. 4.11(a) using a BHPF of order 2 with $D_0 = 15$, 30, and 80, respectively. These results are much smoother than those obtained with an ILPF.

(Source: Gonzalez and Woods)

Image Enhancement in Fourier Domain

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.

Fast Fourier Transform

- Many ways to compute DFT quickly
- Fast Fourier Transform (FFT) algorithm is one such way
- One FFT computation method
  - Divides original vector into 2
  - Calculates FFT of each half recursively
  - Merges results

Source: https://web.cs.wpi.edu/~emmanuel/courses/cs545/S14/slides/lecture10.pdf