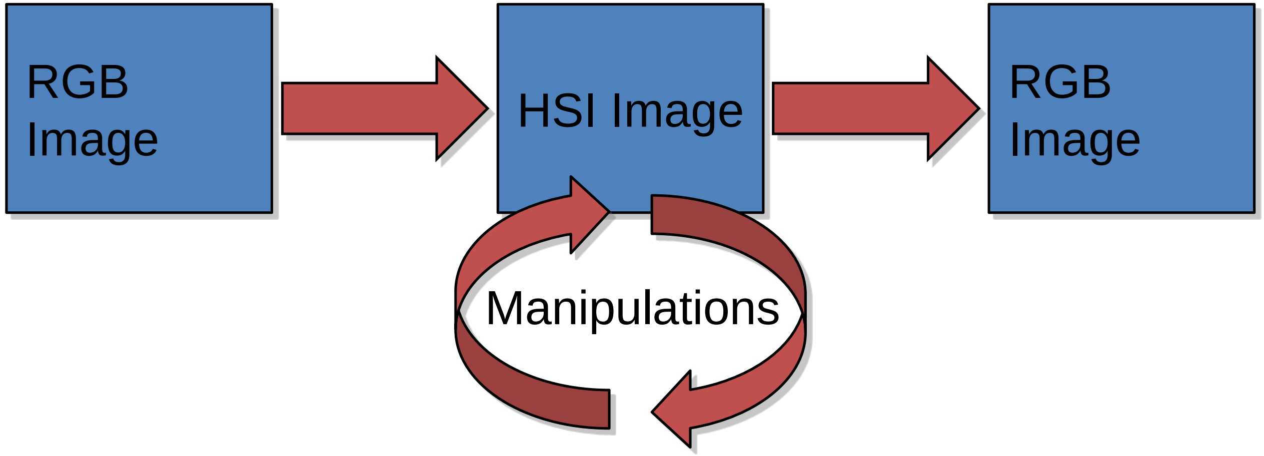
Digital Image Processing

Lecture # 10
Color Processing & Fourier Transform

Manipulating Images In The HSI Model

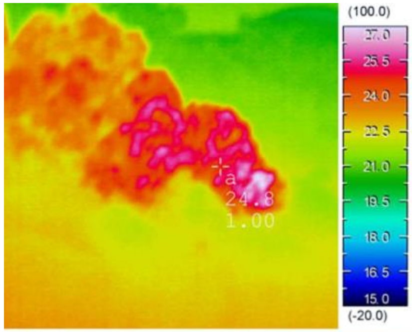
- In order to manipulate an image under the HSI model we:
  - First convert it from RGB to HSI
  - Perform our manipulations under HSI
  - Finally convert the image back from HSI to RGB

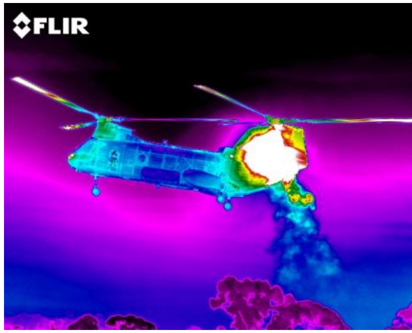


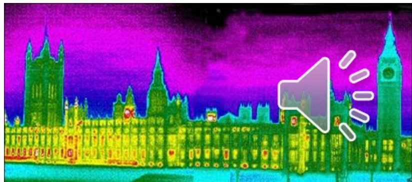
Pseudocolor Image Processing

- Pseudocolor (also called false color) image processing consists of assigning colors to grey values based on a specific criterion

- The principle use of pseudocolor image processing is for human visualization



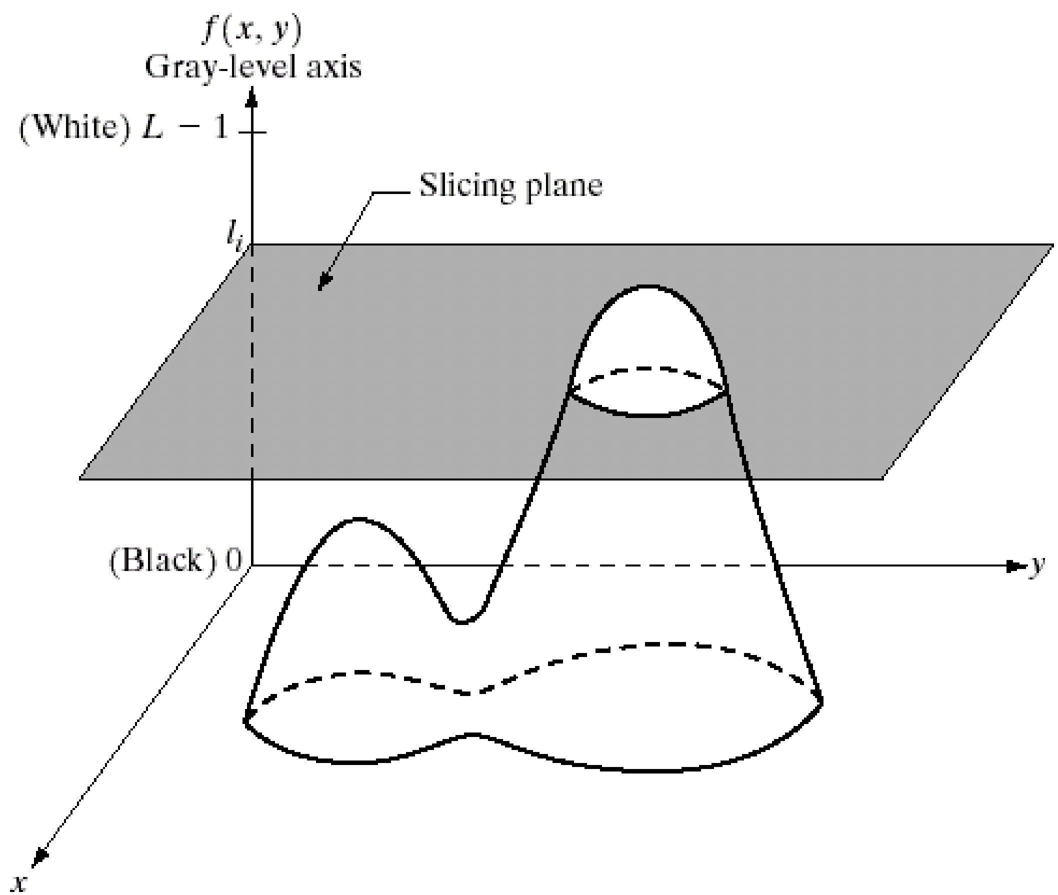




Pseudo Color Image Processing – Intensity Slicing

- Intensity slicing and color coding is one of the simplest kinds of pseudocolor image processing
- First we consider an image as a 3D function mapping spatial coordinates to intensities (that we can consider heights)
- Now consider placing planes at certain levels parallel to the coordinate plane
- If a value is one side of such a plane it is rendered in one color, and a different color if on the other side

Pseudo Color Image Processing – Intensity Slicing



Pseudo Color Image Processing – Intensity Slicing

- In general intensity slicing can be summarized as:
  - Let [$0, L-1$] represent the grey scale
  - Let l_0 represent black [$f(x, y) = 0$] and let l_{L-1} represent white [$f(x, y) = L-1$]
  - Suppose P planes perpendicular to the intensity axis are defined at levels $l_1, l_2, \ldots, l_p$
  - Assuming that $0 < P < L-1$ then the P planes partition the grey scale into $P + 1$ intervals $V_1, V_2, \ldots, V_{P+1}$

Pseudo Color Image Processing – Intensity Slicing

– Grey level color assignments can then be made according to the relation:

$$f(x,y)=c_k \qquad \text{if} f(x,y) \in V_k$$

– where c_k is the color associated with the k^{th} intensity level V_k defined by the partitioning planes at $l = k - 1$ and $l = k$

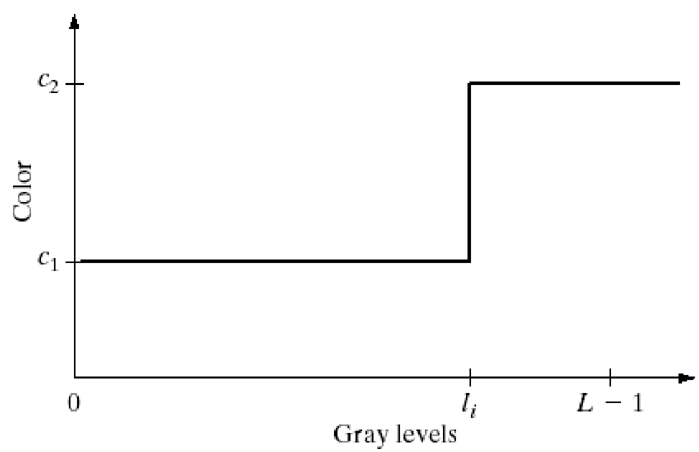
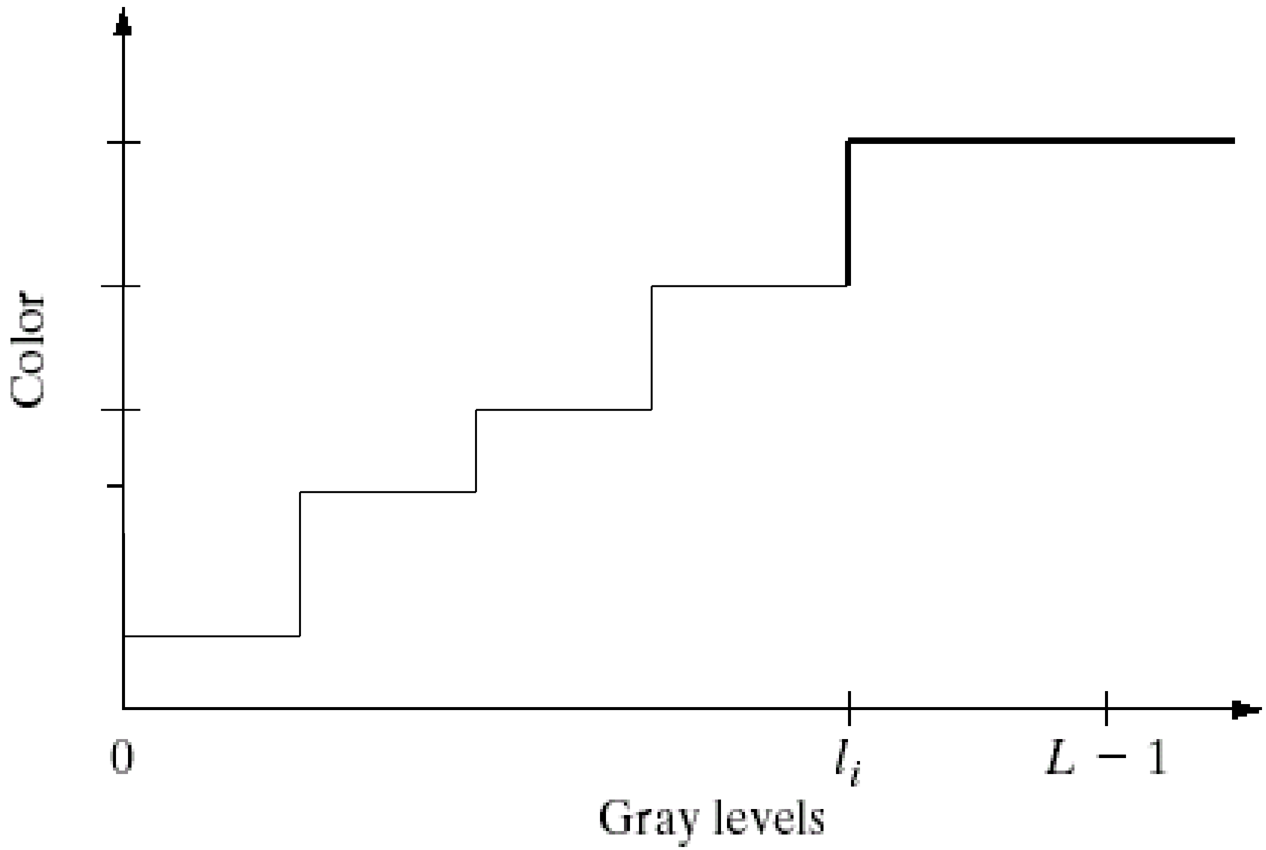


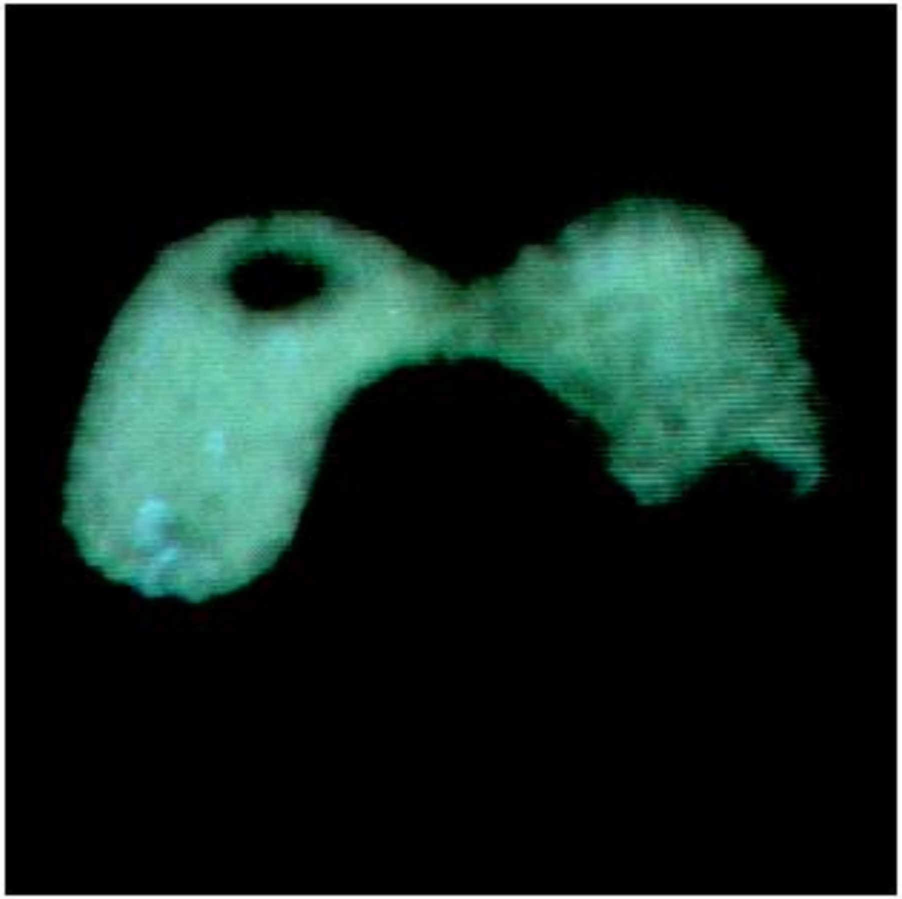
FIGURE 6.19 An alternative representation of the intensity-slicing technique.

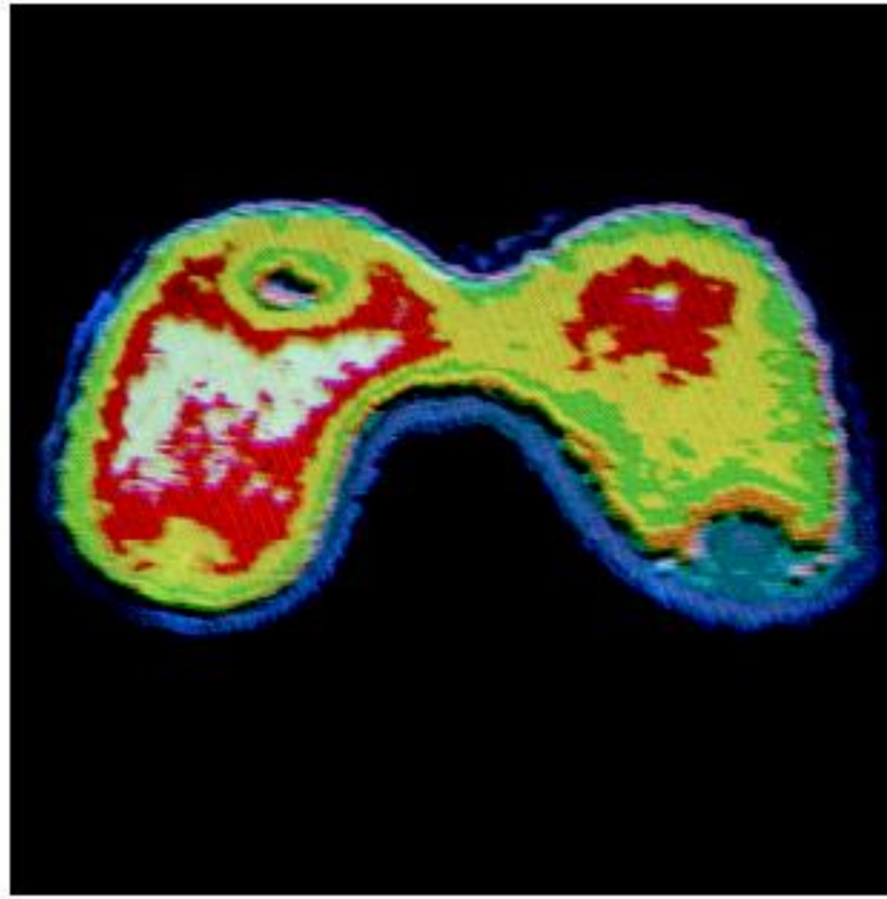
Intensity slicing (cont.)

- More slicing plane, more colors



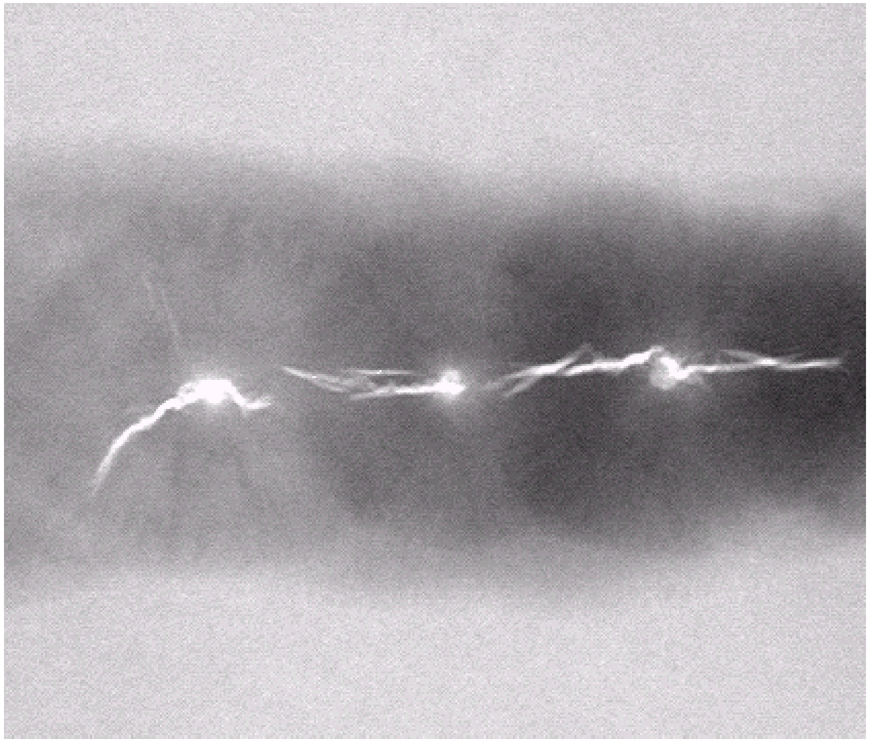
Pseudo Color Image Processing – Intensity Slicing

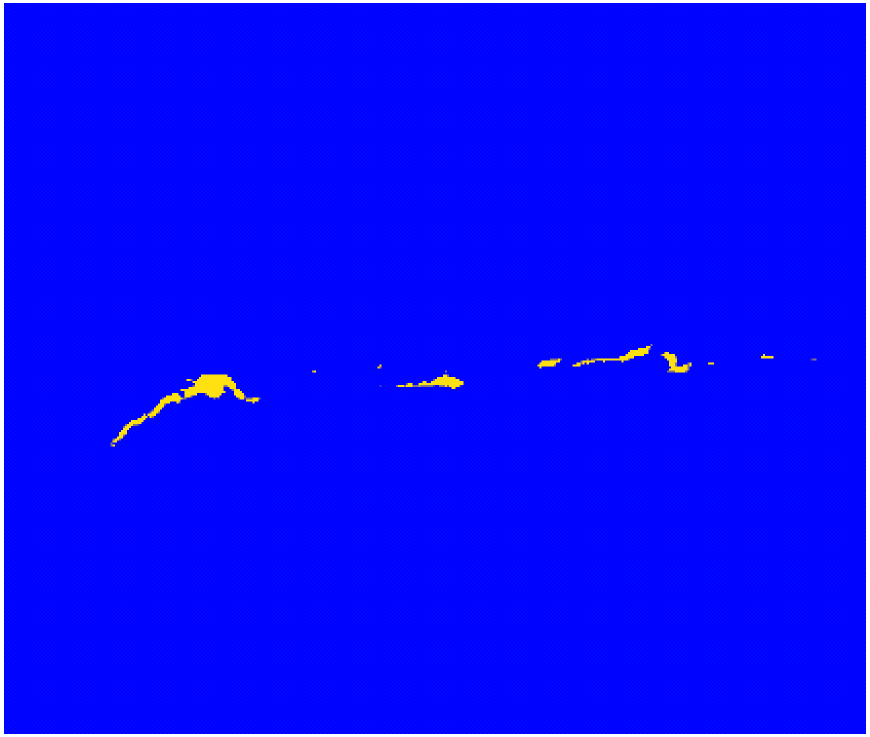




Radiation test pattern → 8 color regions

Pseudo Color Image Processing – Intensity Slicing





Gray level to color transformation

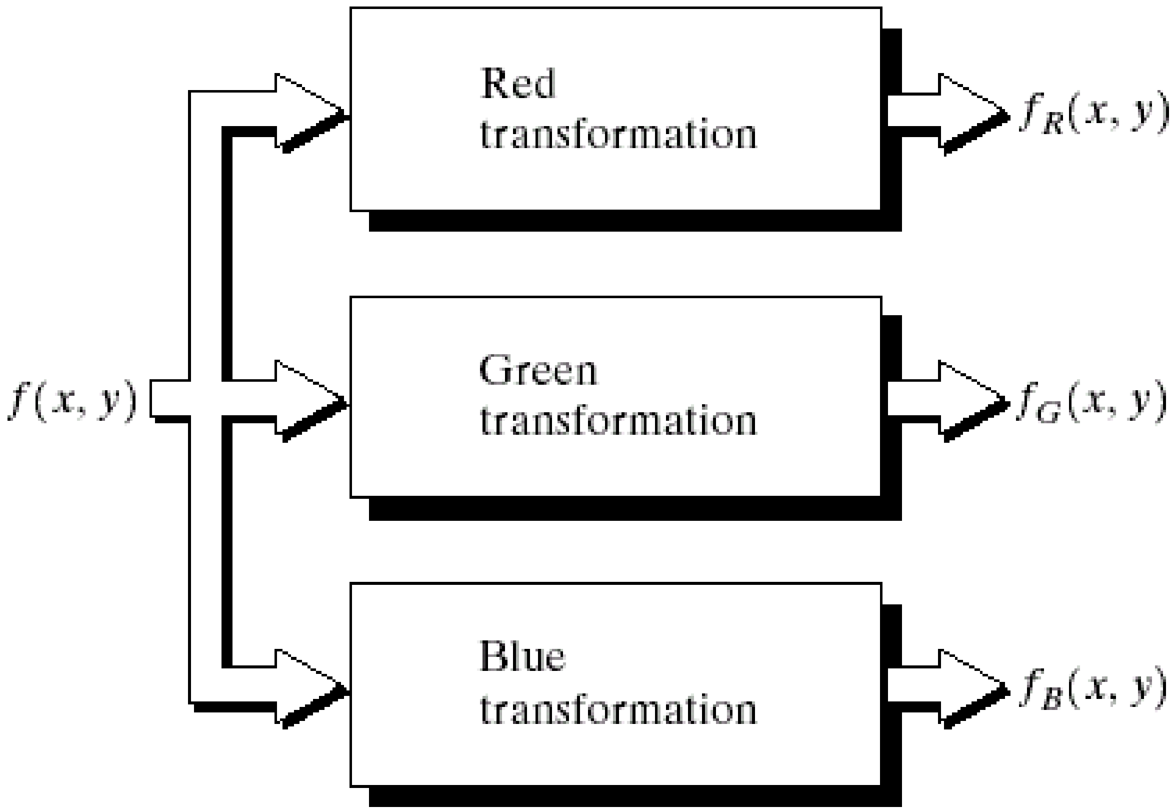


FIGURE 6.23 Functional block diagram for pseudocolor image processing. f_R, f_G, and f_B are fed into the corresponding red, green, and blue inputs of an RGB color monitor.

Color pixel

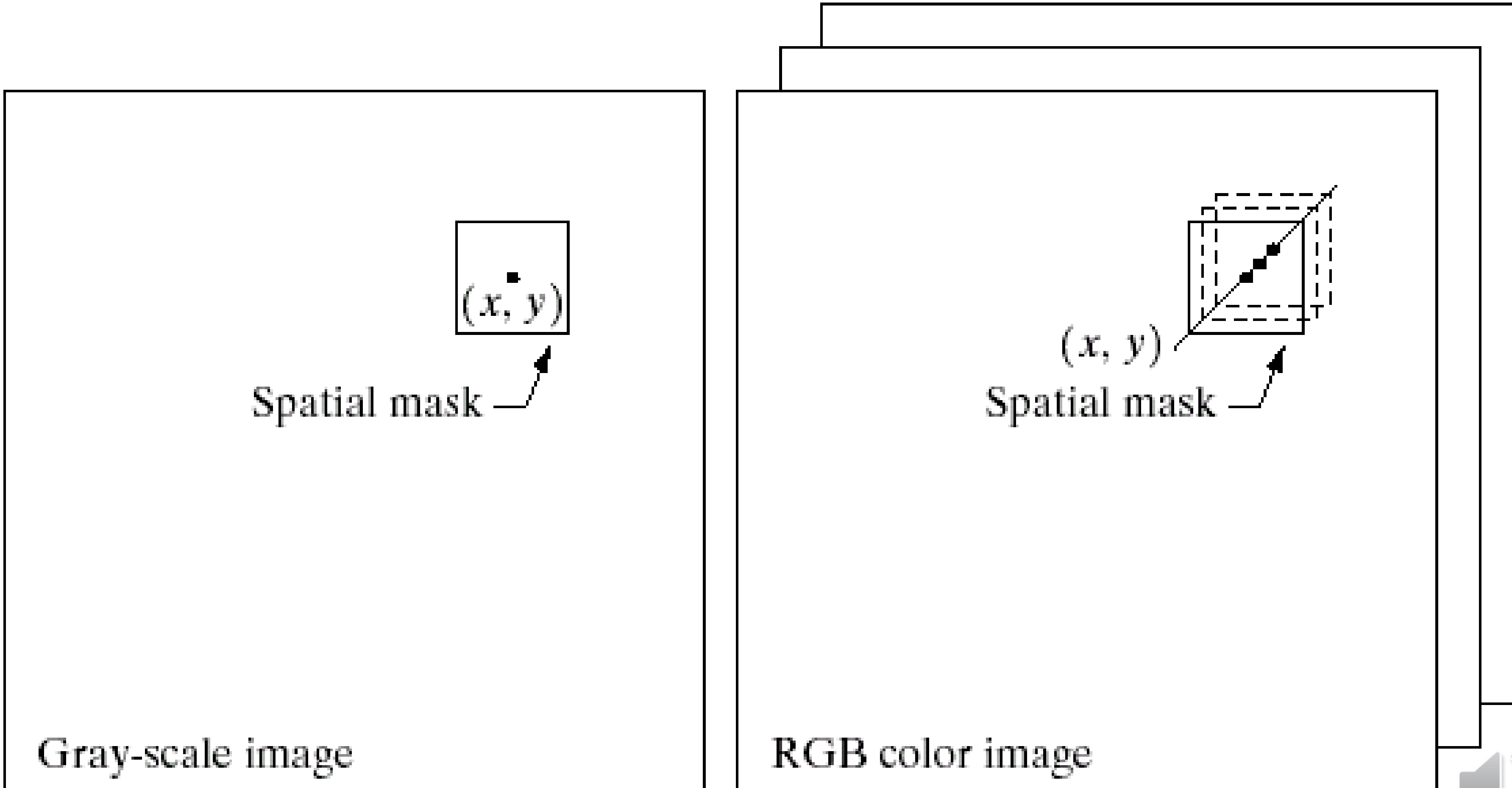
- A pixel at (x,y) is a vector in the color space
  - RGB color space

$$\mathbf{c}(x, y) = \begin{bmatrix} R(x, y) \\ G(x, y) \\ B(x, y) \end{bmatrix}$$

c.f. gray-scale image

f(x,y) = I(x,y)

Example: spatial mask



How to deal with color vector?

- Per-color-component processing
  - Process each color component
- Vector-based processing
  - Process the color vector of each pixel
- When can the above methods be equivalent?
  - Process can be applied to both scalars and vectors
  - Operation on each component of a vector must be independent of the other component

Two spatial processing categories

- Similar to gray scale processing studied before, we have to major categories
- Pixel-wise processing
- Neighborhood processing

COLOR IMAGE - SMOOTHING

- Smoothing can be viewed as a spatial filtering operation in which the coefficients of the filtering mask are all 1's
- This concept can be easily extended to the processing of full-color images
- Simply smooth each of the RGB color planes and then combine the processed planes to form a smoothed full-color result

$$\hat{\mathbf{c}}(x,y) = \frac{1}{MN}\begin{bmatrix} \sum_{(x,y)\in S_{xy}} R(x,y) \\ \sum_{(x,y)\in S_{xy}} G(x,y) \\ \sum_{(x,y)\in S_{xy}} B(x,y) \end{bmatrix}$$

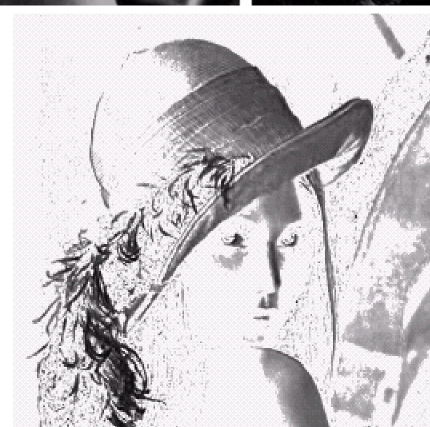
original

R

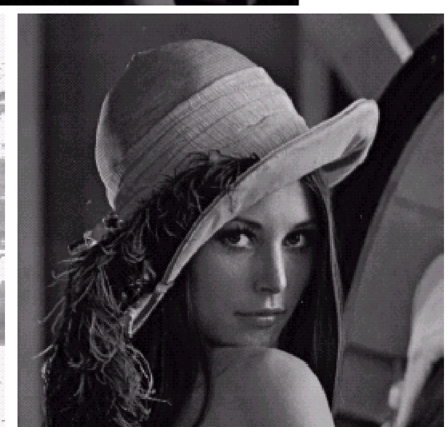
G

G

H



S



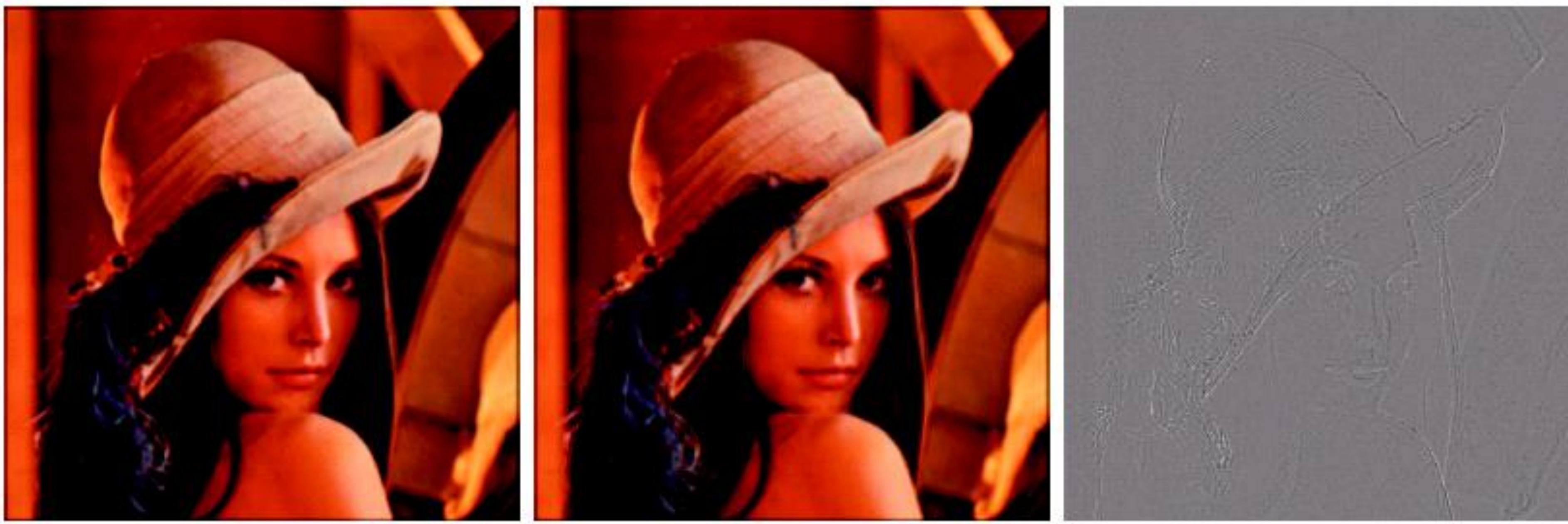
I

Example: 5x5 smoothing mask

RGB model

Smooth I
in HSI model

difference



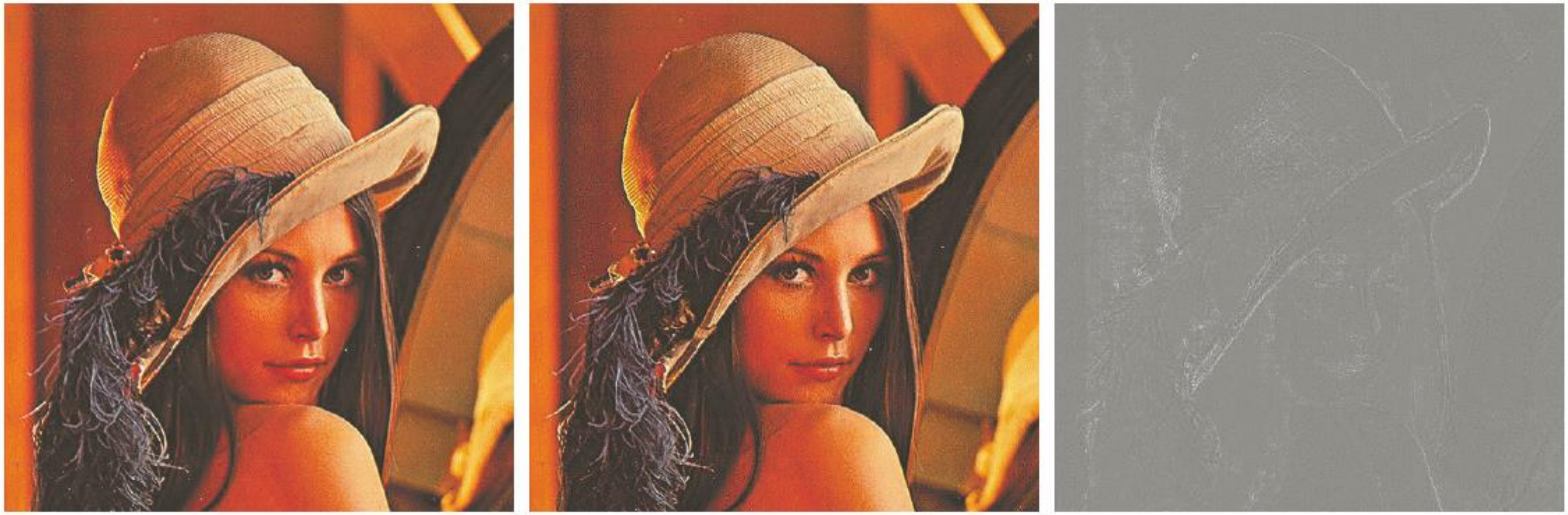
a b c

FIGURE 6.40 Image smoothing with a 5 × 5 averaging mask. (a) Result of processing each RGB component image. (b) Result of processing the intensity component of the HSI image and converting to RGB. (c) Difference between the two results.

Color Image Sharpening

The Laplacian of vector c is

$$\nabla^2 \left[c(x,y) \right] = \begin{bmatrix} \nabla^2 R(x,y) \\ \nabla^2 G(x,y) \\ \nabla^2 B(x,y) \end{bmatrix}$$



a b c

FIGURE 6.41 Image sharpening with the Laplacian. (a) Result of processing each RGB channel. (b) Result of processing the HSI intensity component and converting to RGB. (c) Difference between the two results.

Color Edge Detection (1)

Let r, g, and b be unit vectors along the R, G, and B axis of RGB color space, and define vectors

$$\mathrm{u} = \frac{\partial R}{\partial x}\mathrm{r} + \frac{\partial G}{\partial x}\mathrm{g} + \frac{\partial B}{\partial x}\mathrm{b}$$

and

$$\mathrm{v} = \frac{\partial R}{\partial y}\mathrm{r} + \frac{\partial G}{\partial y}\mathrm{g} + \frac{\partial B}{\partial y}\mathrm{b}$$

Color Edge Detection (2)

$$g_{xx} = \mathbf{u} \cdot \mathbf{u} = \left|\frac{\partial R}{\partial x}\right|^2 + \left|\frac{\partial G}{\partial x}\right|^2 + \left|\frac{\partial B}{\partial x}\right|^2$$

$$g_{yy} = \mathbf{v} \cdot \mathbf{v} = \left|\frac{\partial R}{\partial y}\right|^2 + \left|\frac{\partial G}{\partial y}\right|^2 + \left|\frac{\partial B}{\partial y}\right|^2$$

and

$$g_{xy} = \mathbf{u} \cdot \mathbf{v} = \frac{\partial R}{\partial x}\frac{\partial R}{\partial y} + \frac{\partial G}{\partial x}\frac{\partial G}{\partial y} + \frac{\partial B}{\partial x}\frac{\partial B}{\partial y}$$

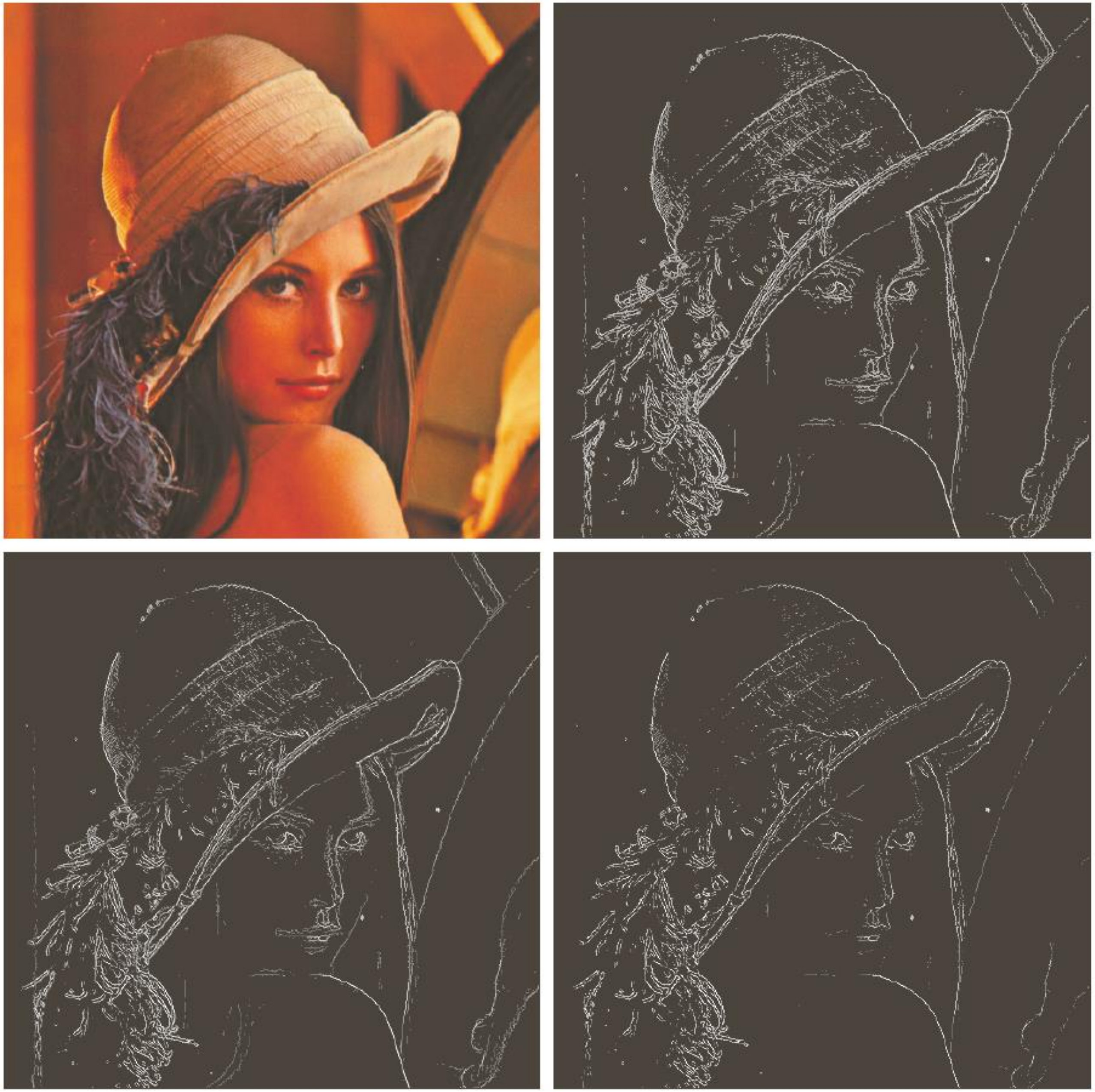


FIGURE 6.46
(a) RGB image.
(b) Gradient computed in RGB color vector space.
(c) Gradients computed on a per-image basis and then added.
(d) Difference between (b) and (c).

Image Enhancement in Frequency Domain

Joseph Fourier (1768 – 1830)

– Most famous for his work *"La Théorie Analitique de la Chaleur"* published in 1822
– Translated into English in 1878: *"The Analytic Theory of Heat"*

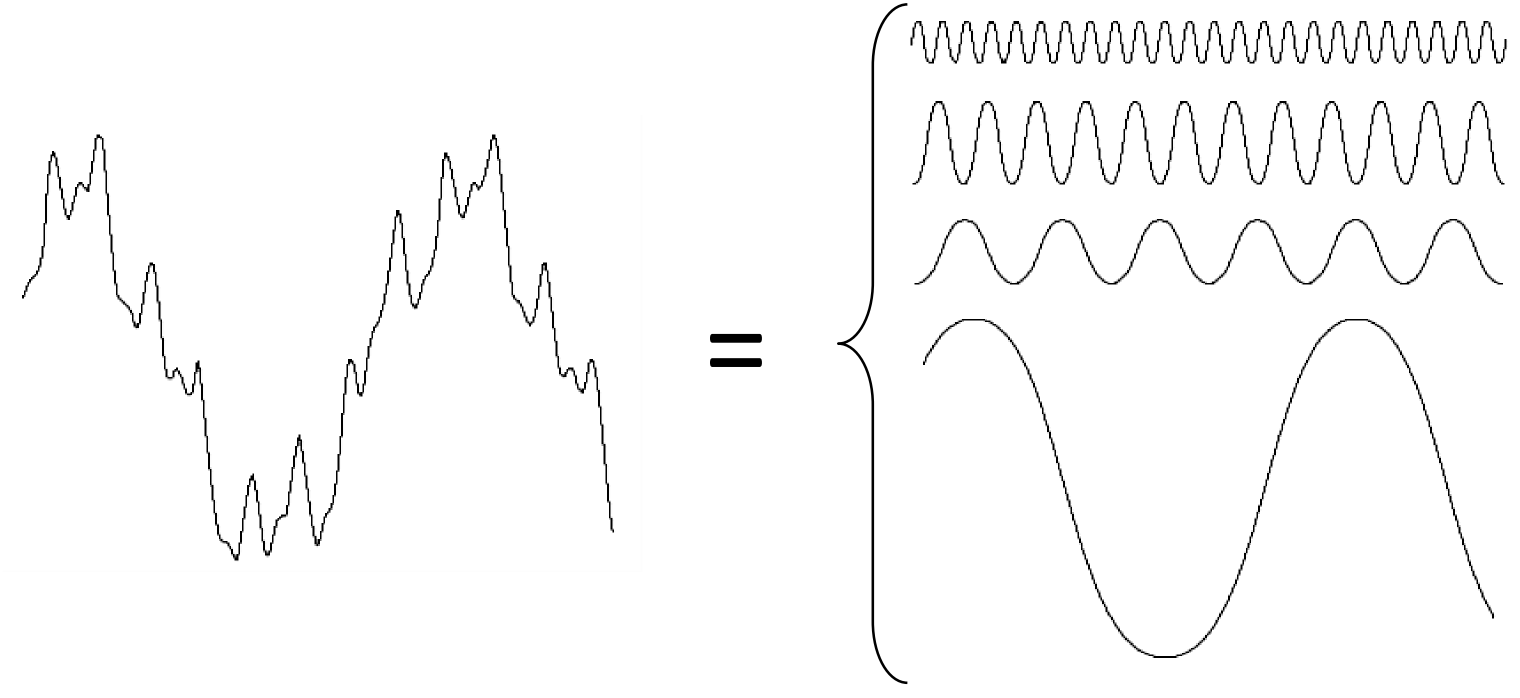
Nobody paid much attention when the work was first published
One of the most important mathematical theories in modern engineering

Background

- Any function that periodically repeats itself can be expressed as the sum of sines and/or cosines of different frequencies, each multiplied by a different coefficient (Fourier series).
- Even functions that are not periodic (but whose area under the curve is finite) can be expressed as the integral of sines and/or cosines multiplied by a weighting function (Fourier transform).

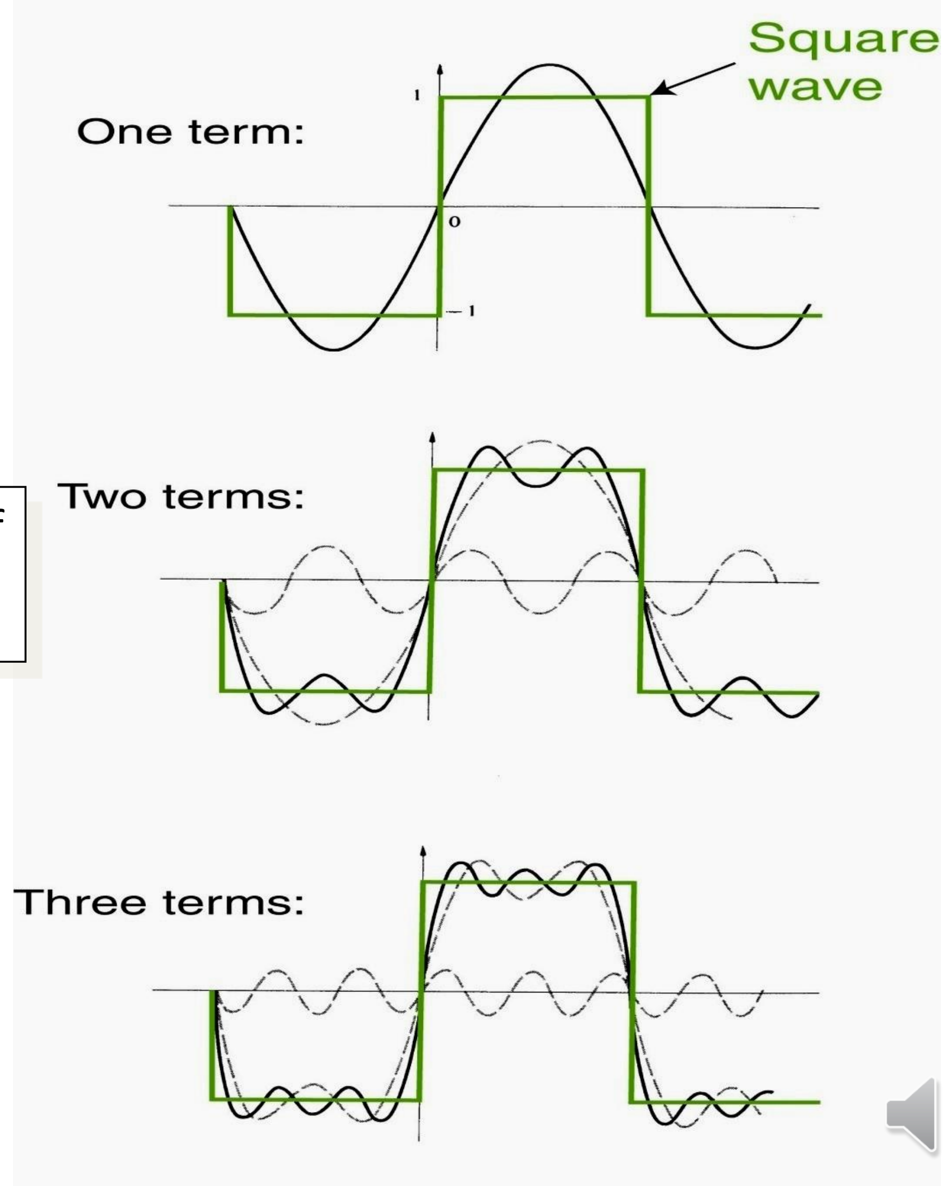
The big idea …

Any function that periodically repeats itself can be expressed as a sum of sines and cosines of different frequencies each multiplied by a different coefficient – a *Fourier series*

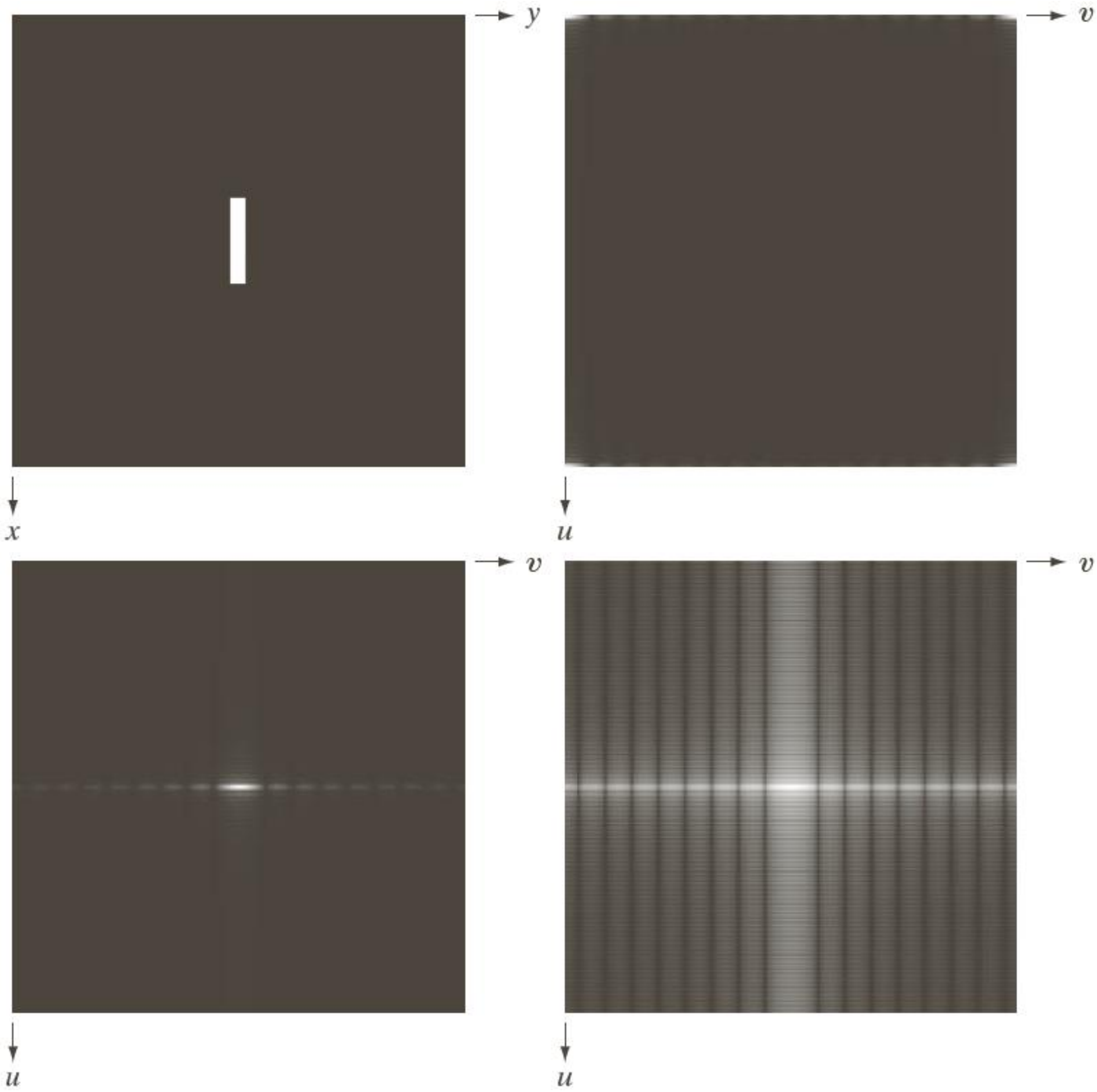


The big idea…

Approximating a square wave as the sum of sine waves

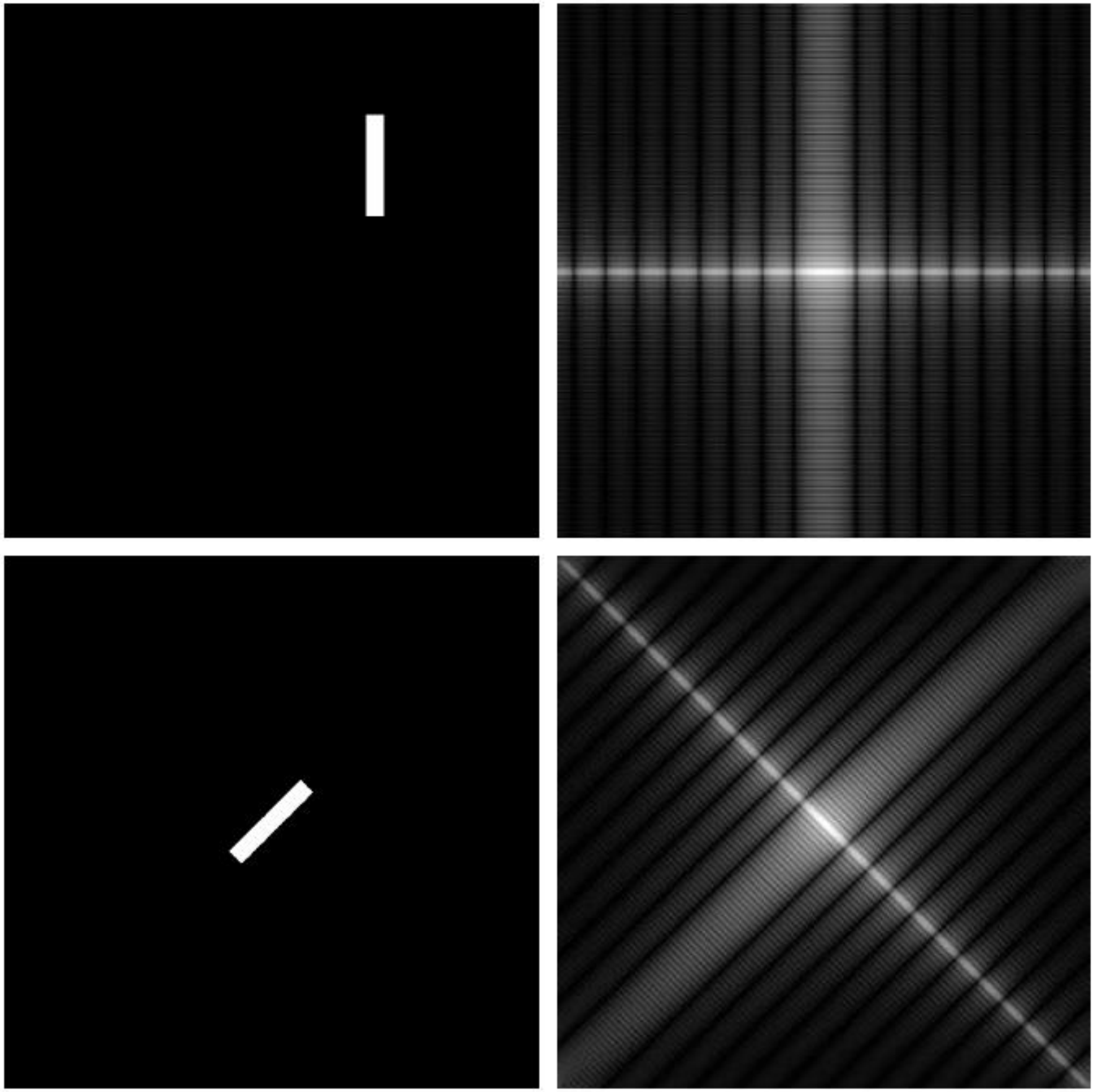


Frequencies in Images



a b
c d

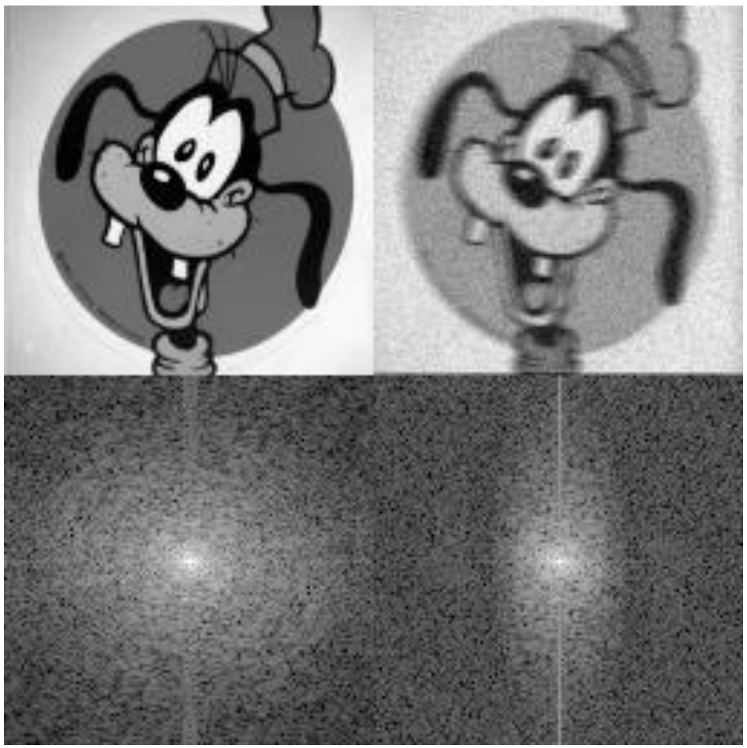
FIGURE 4.24
(a) Image.
(b) Spectrum showing bright spots in the four corners.
(c) Centered spectrum. (d) Result showing increased detail after a log transformation. The zero crossings of the spectrum are closer in the vertical direction because the rectangle in (a) is longer in that direction. The coordinate convention used throughout the book places the origin of the spatial and frequency domains at the top left.

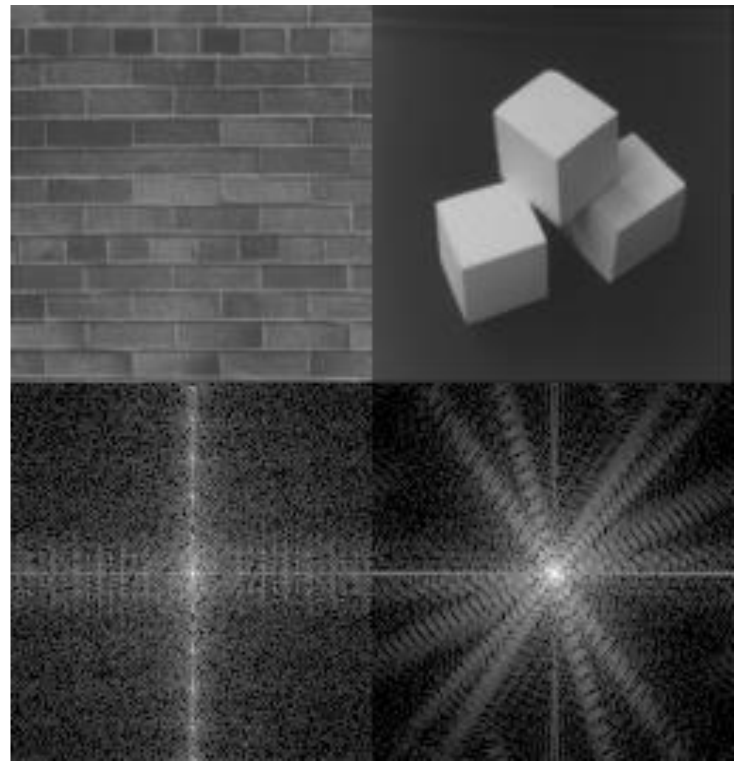


a b
c d

FIGURE 4.25
(a) The rectangle in Fig. 4.24(a) translated, and (b) the corresponding spectrum. (c) Rotated rectangle, and (d) the corresponding spectrum. The spectrum corresponding to the translated rectangle is identical to the spectrum corresponding to the original image in Fig. 4.24(a).

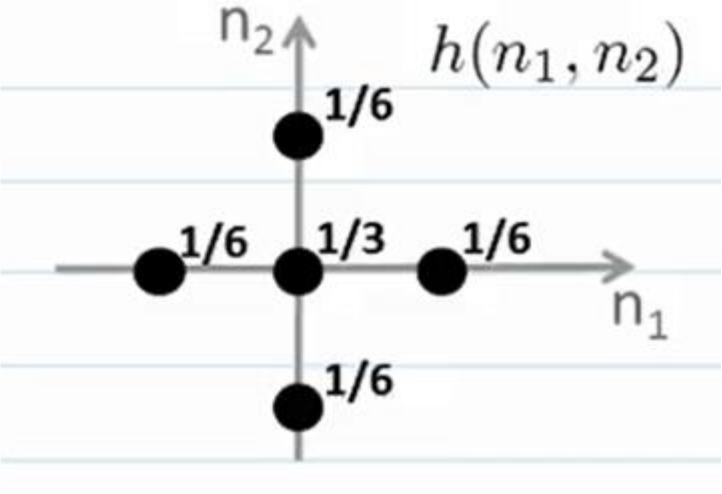
Frequencies in Images



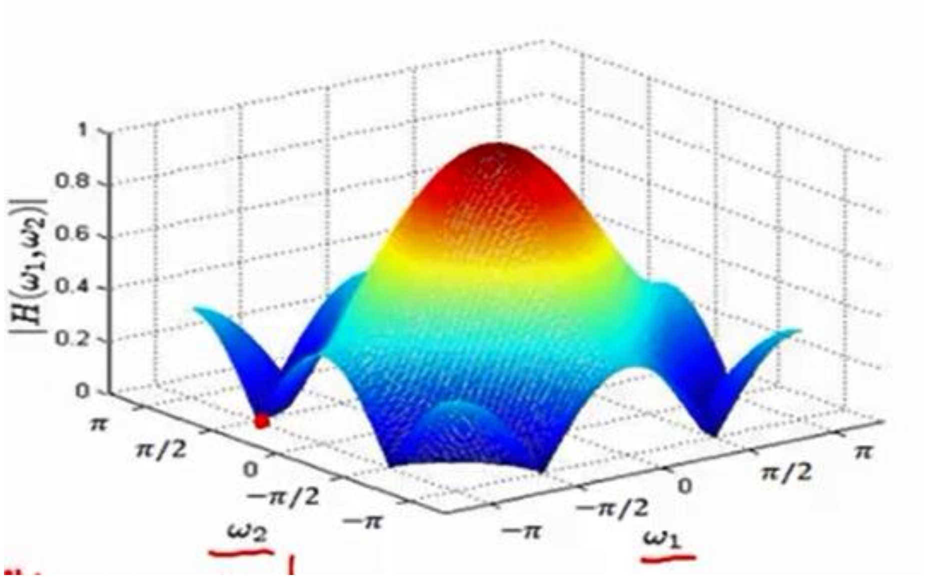


Basic 2D FT

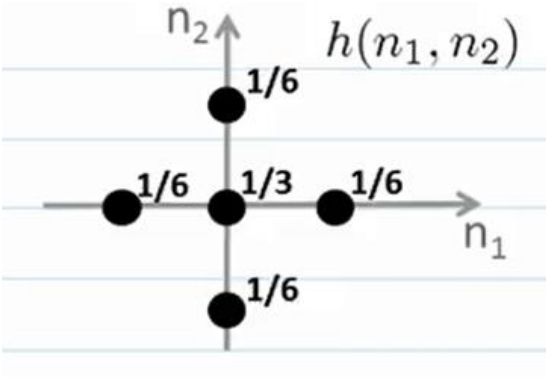
Example



n_2
$h(n_1, n_2)$
1/6
1/6
1/3
1/6
n_1
1/6

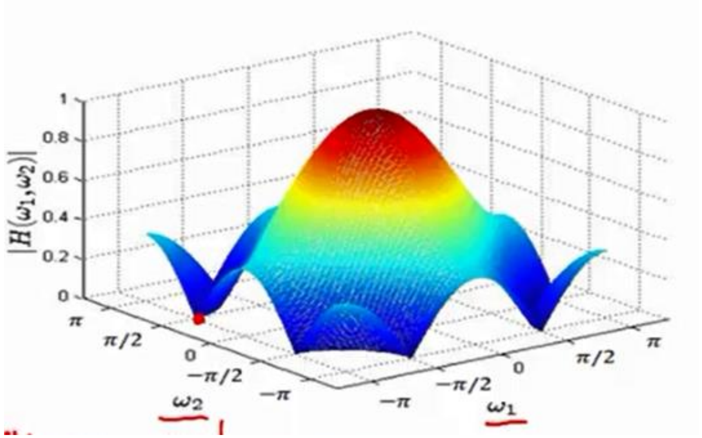


$|H(\omega_1, \omega_2)|$
1
0.8
0.6
0.4
0.2
0
π
$\pi/2$
0
$-\pi/2$
$-\pi$
ω_2
$-\pi$
$-\pi/2$
0
$\pi/2$
π
ω_1

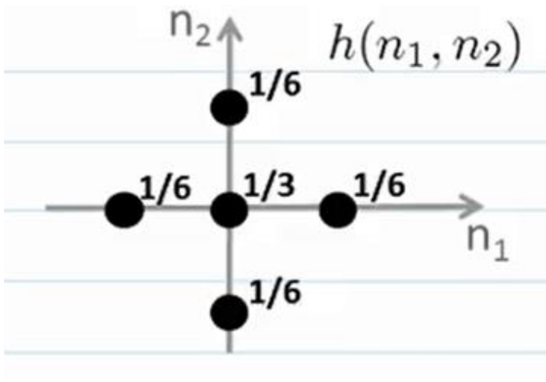


n_2
$h(n_1, n_2)$
1/6
1/6
1/3
1/6
n_1
1/6

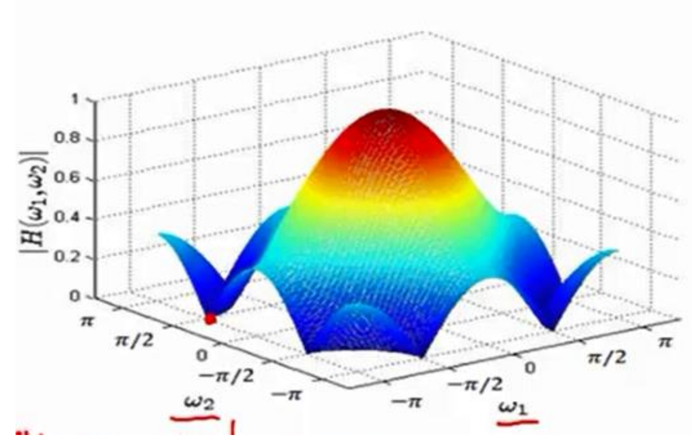
Example



$|H(\omega_1, \omega_2)|$
1
0.8
0.6
0.4
0.2
0
π
$\pi/2$
0
$-\pi/2$
$-\pi$
ω_2
$-\pi$
$-\pi/2$
0
$\pi/2$
π
ω_1



Example

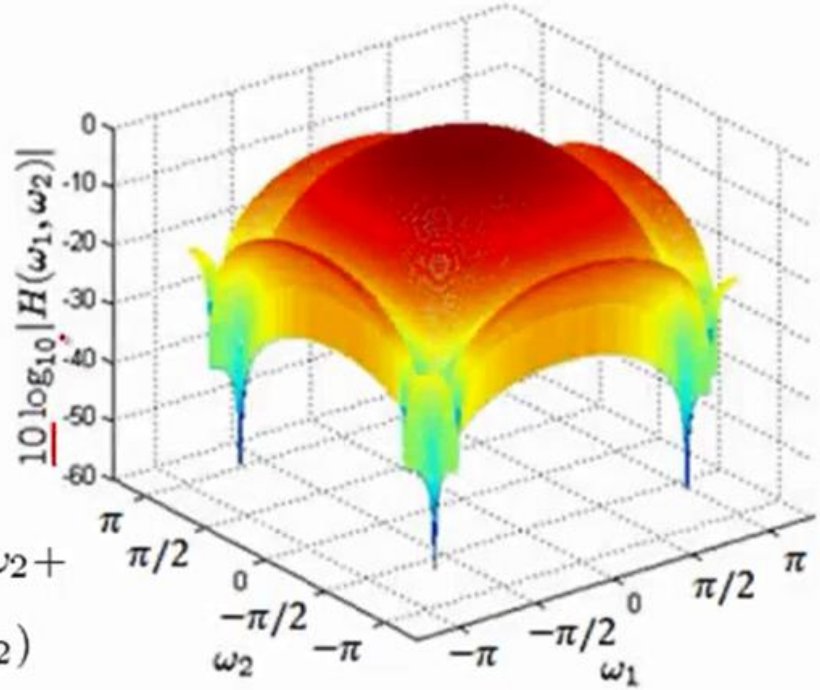


$$= \frac{1}{6} e^{j\omega_2} + \frac{1}{3} + \frac{1}{6} e^{-j\omega_2} + \frac{1}{6} e^{j\omega_1} + \frac{1}{6} e^{-j\omega_1}$$

Example

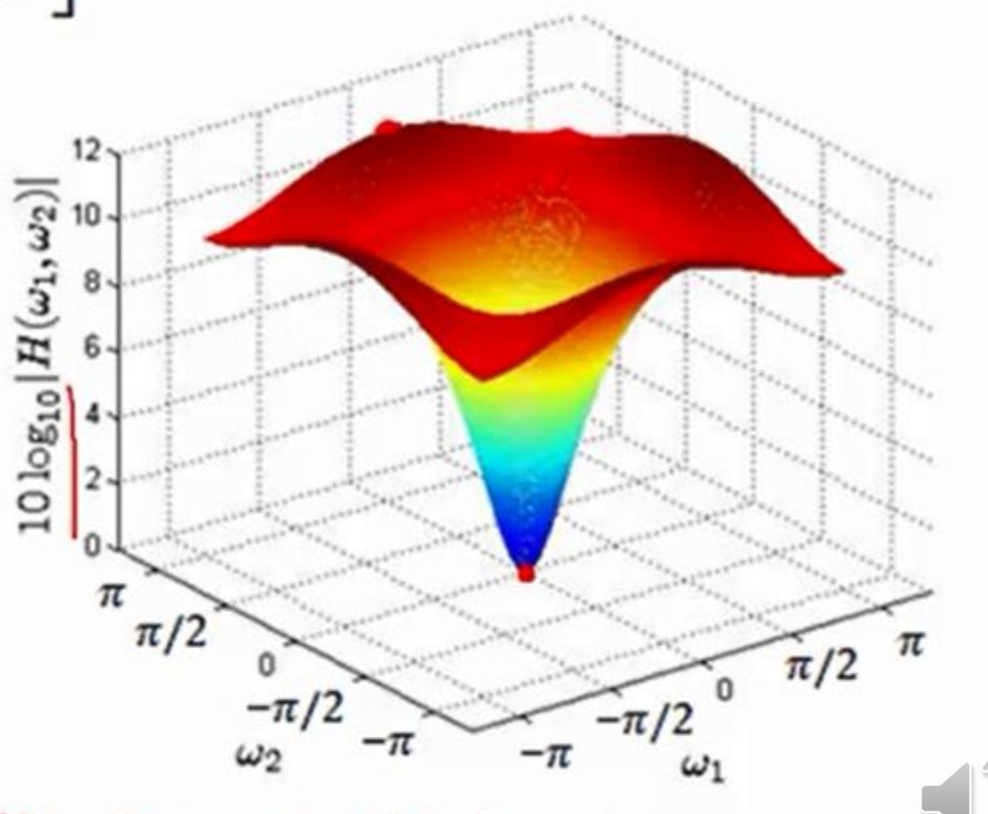
$$h(n_1, n_2) = \begin{bmatrix} 0.075 & 0.124 & 0.075 \\ 0.124 & 0.204 & 0.124 \\ 0.075 & 0.124 & 0.075 \end{bmatrix}$$

$$H(\omega_1, \omega_2) = 0.204 + 0.124 \cdot 2 \cdot cos\omega_1 + 0.124 \cdot 2 \cdot cos\omega_2 + 0.075 \cdot 2 \cdot cos(\omega_1 + \omega_2) + 0.075 \cdot 2 \cdot cos(\omega_1 - \omega_2)$$



Example

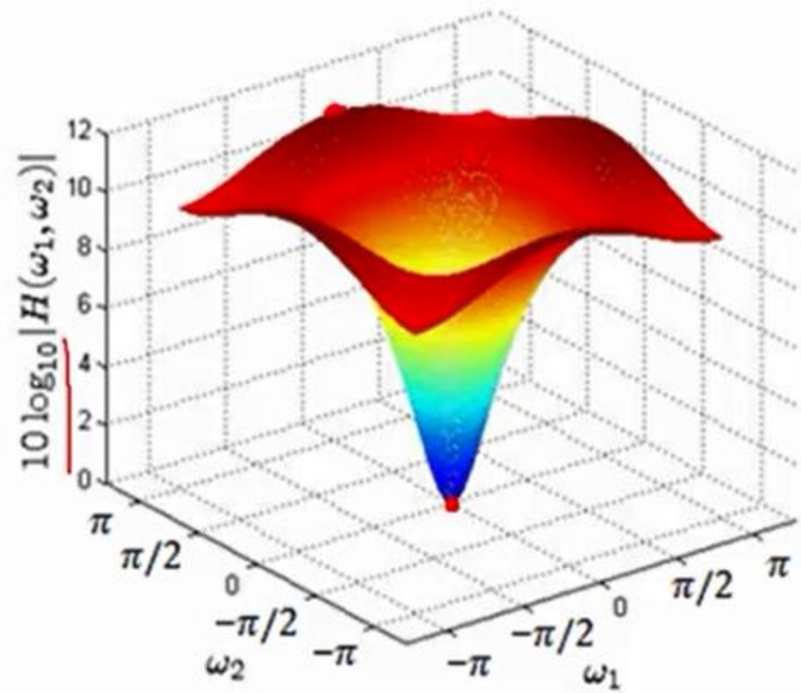
$$h(n_1, n_2) = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



$$h(n_1, n_2) = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

h(0,0)

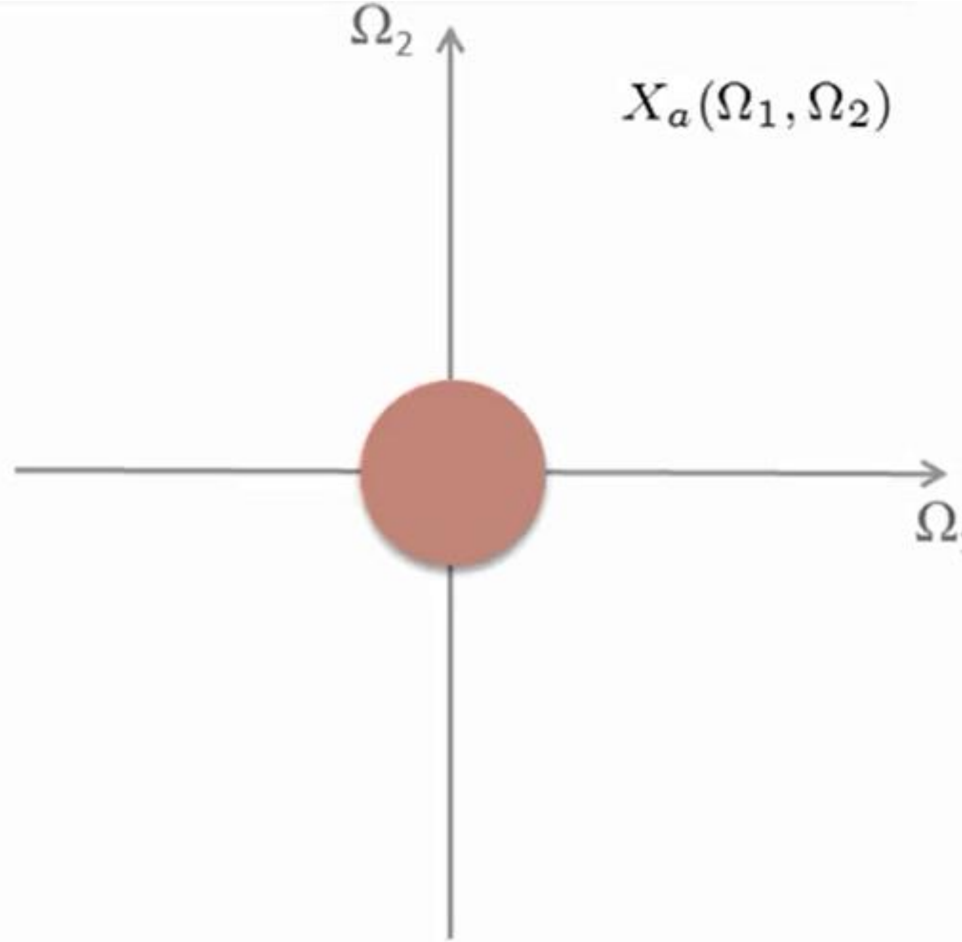
$$H(\omega_1, \omega_2) = 9 - 2 \cdot cos\omega_1 - 2 \cdot cos\omega_2 - 2 \cdot cos(\omega_1 + \omega_2) - 2 \cdot cos(\omega_1 - \omega_2)$$



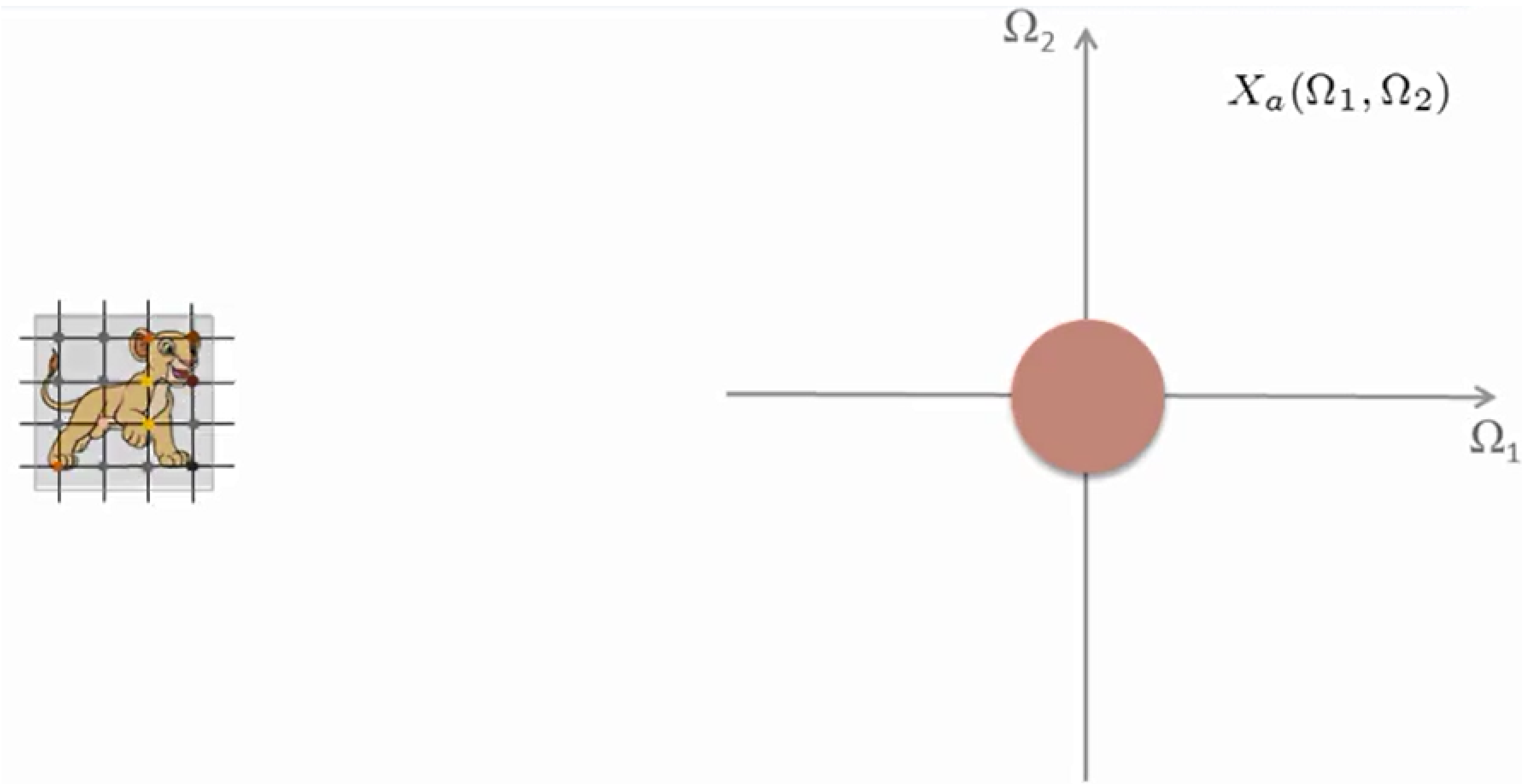
$H(0,0)=1 \rightarrow \log H(0,0)=0$

$H(0,\pi)=13, \ H(\pi,\pi)=9$

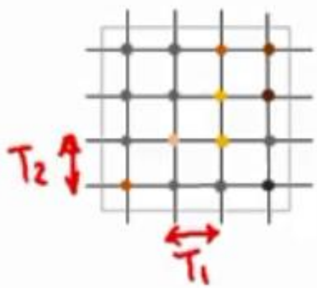
2D Sampling

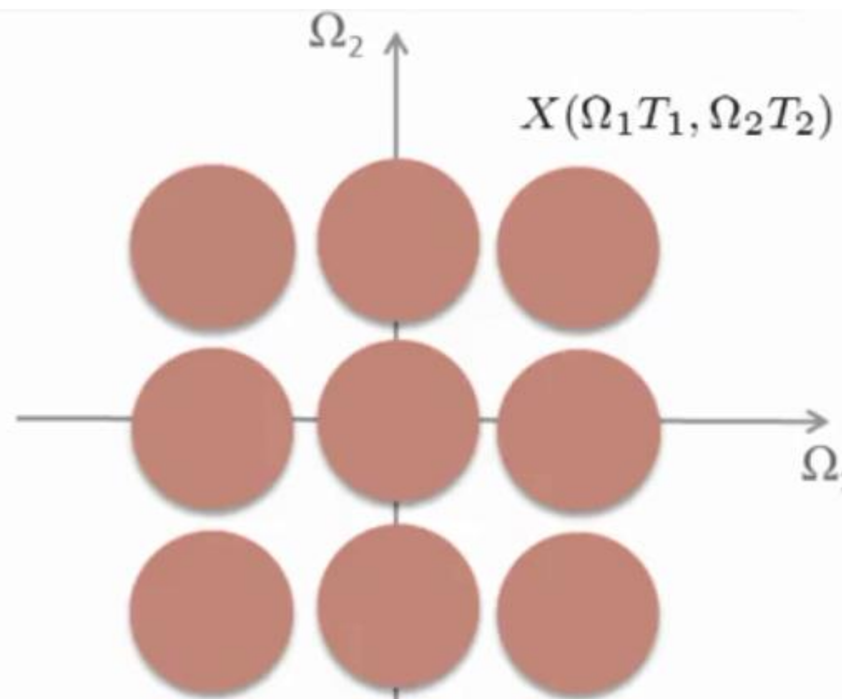


2D Sampling



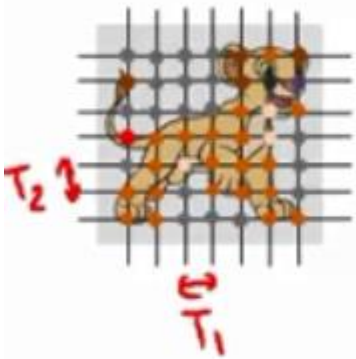
2D Sampling

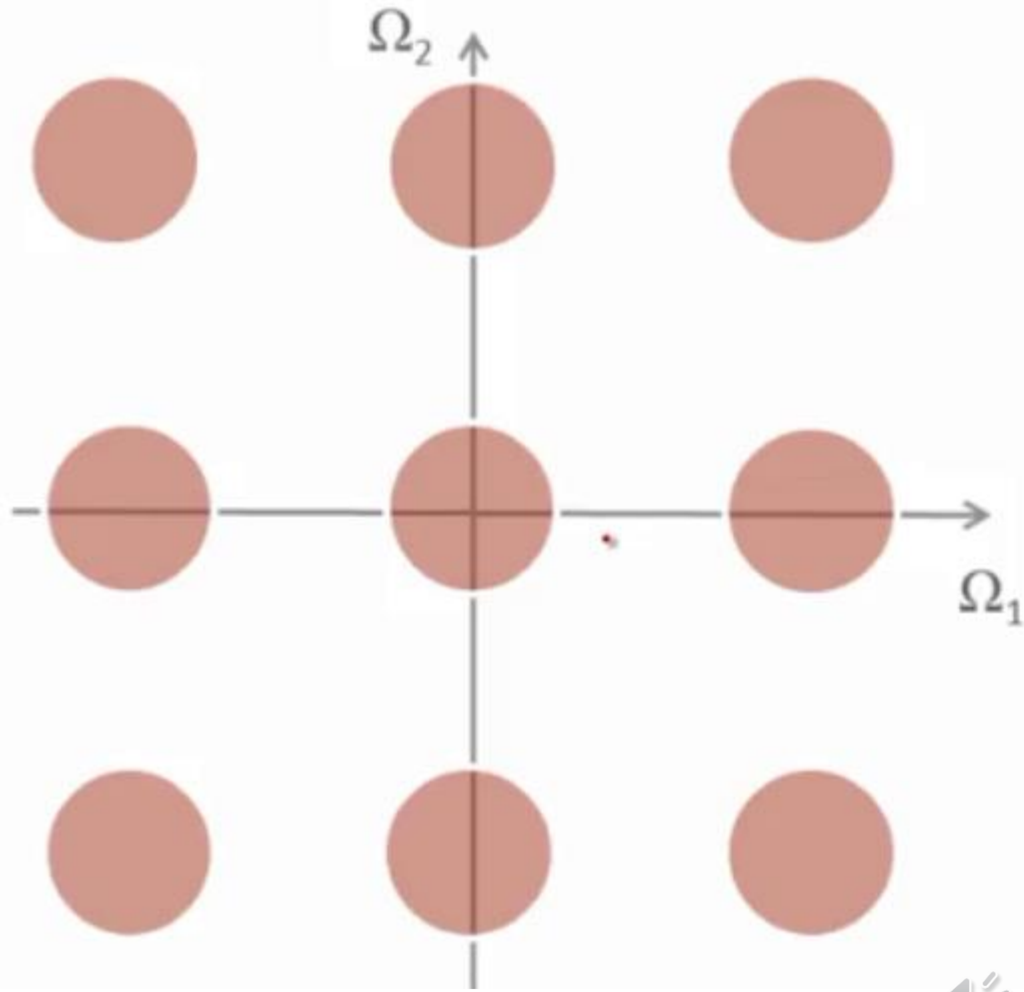




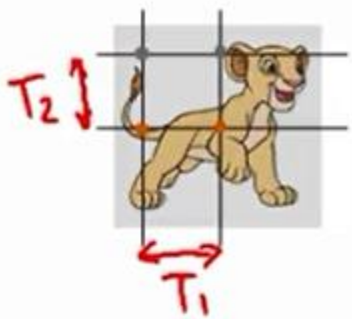
$$X(\Omega_1 T_1, \Omega_2 T_2) = \frac{1}{T_1 T_2} \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} X_a(\Omega_1 - \frac{2\pi}{T_1} k_1, \Omega_2 - \frac{2\pi}{T_2} k_2)$$

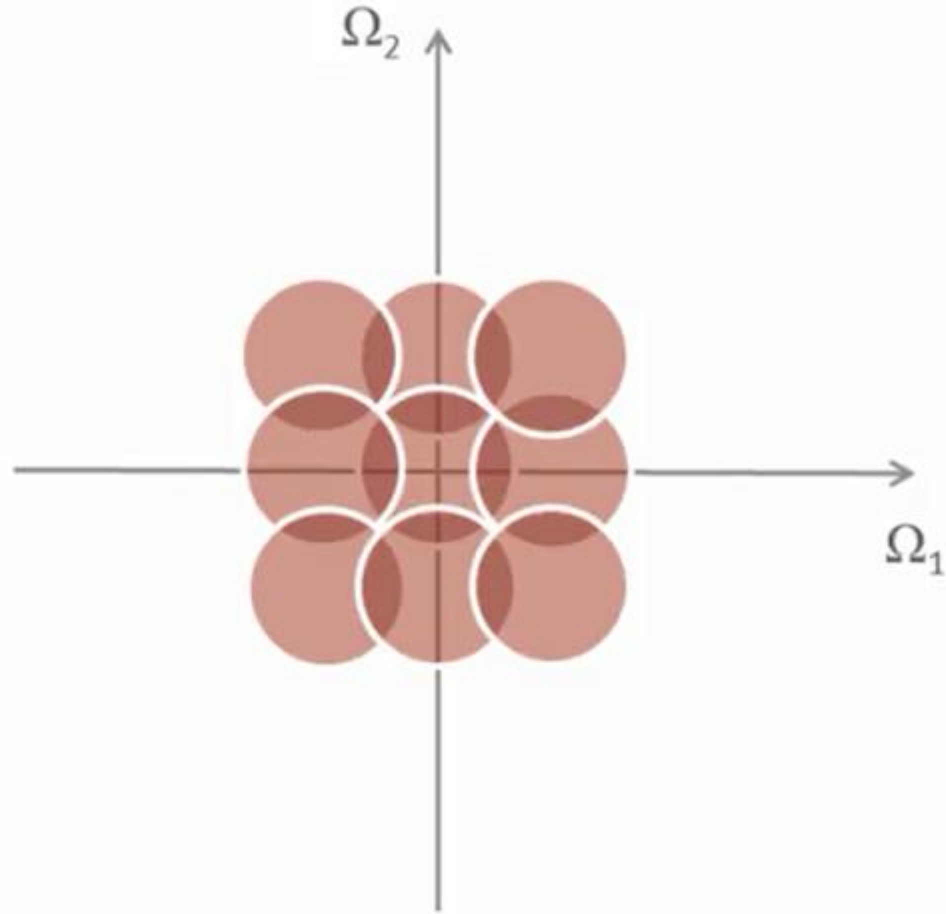
Over Sampling



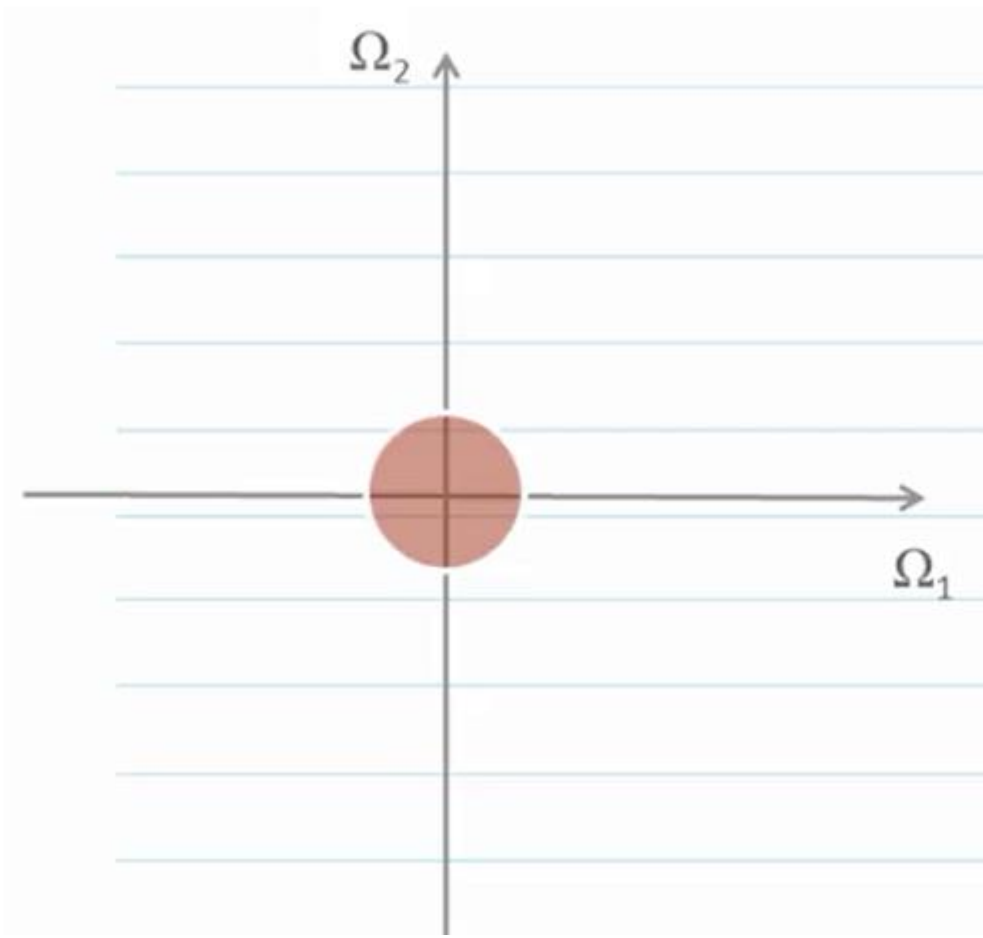


Under Sampling

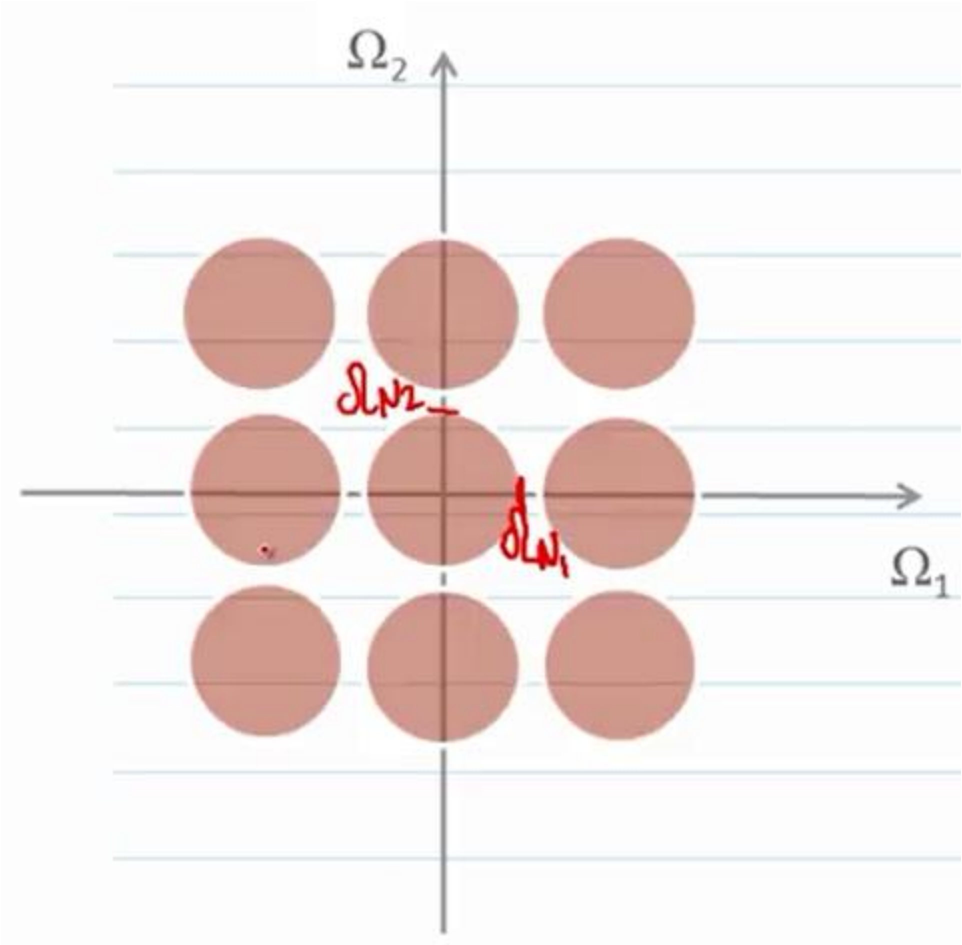




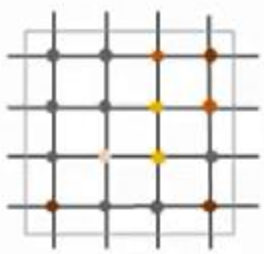
2D Nyquist Theorem

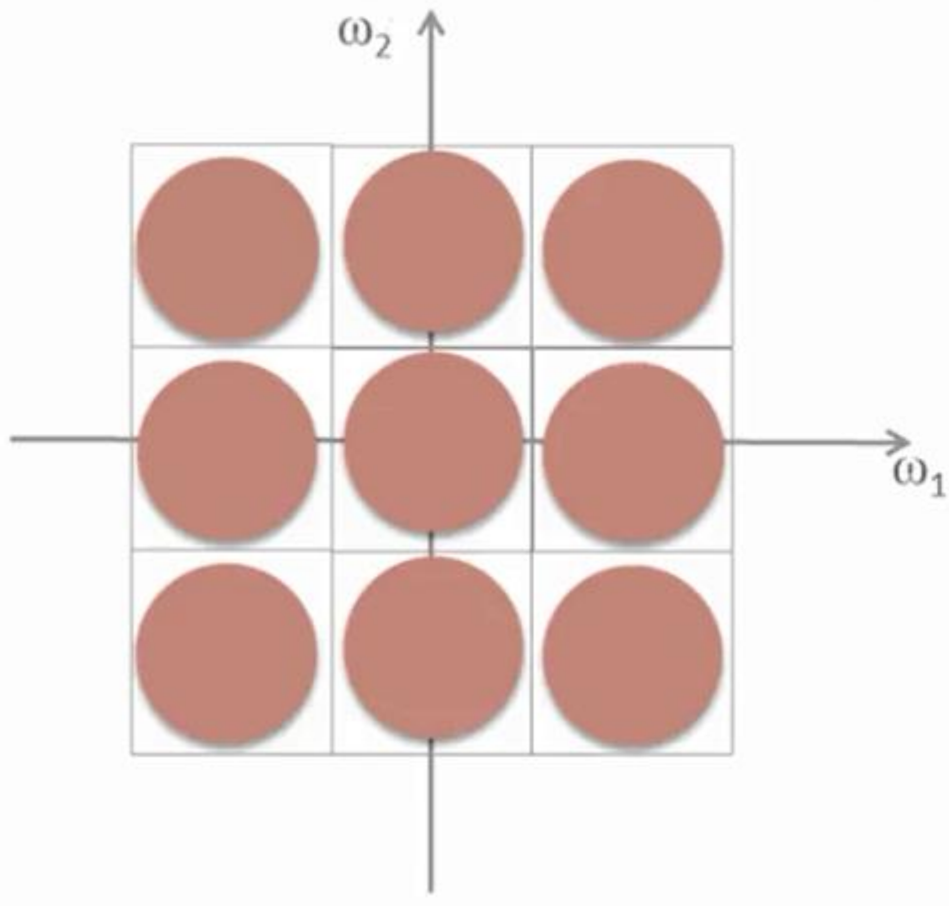


2D Nyquist Theorem

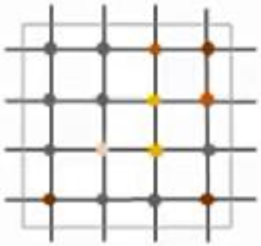


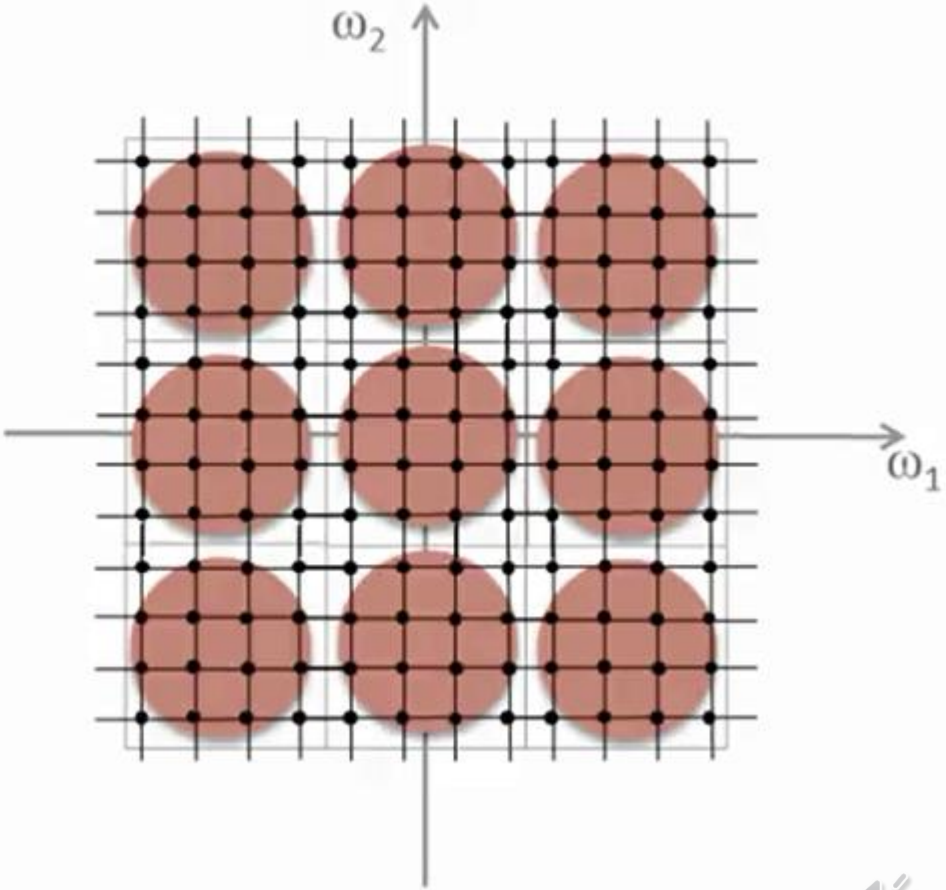
Sampling in the Frequency



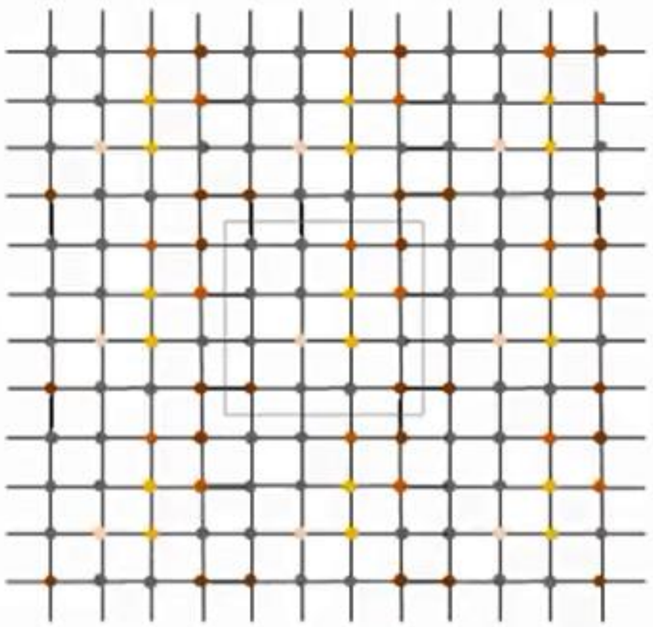


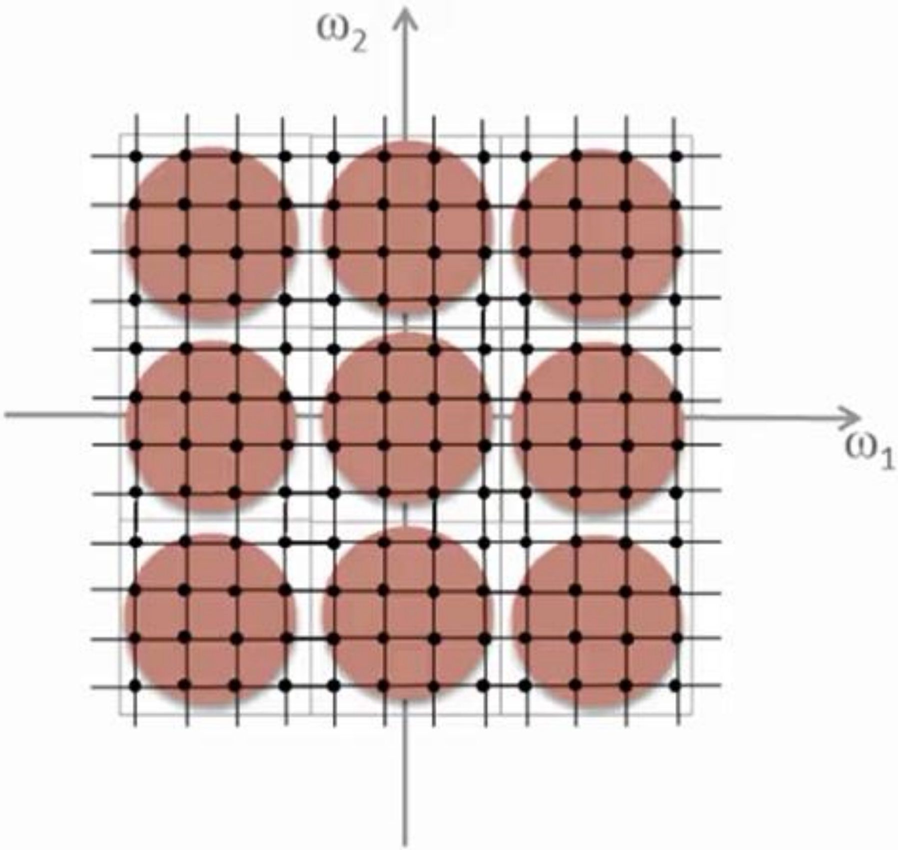
Sampling in the Frequency



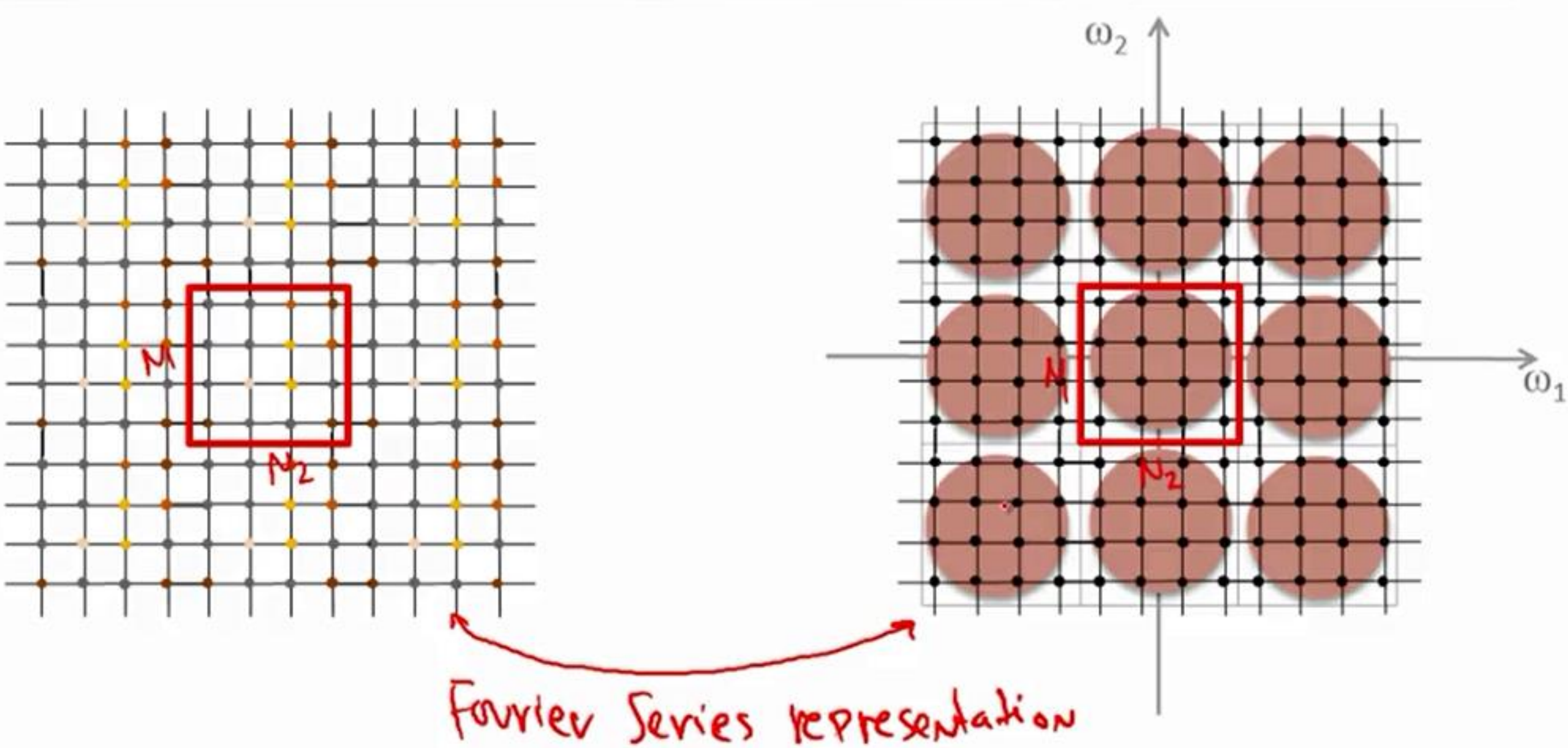


Sampling in the Frequency





Sampling in the Frequency



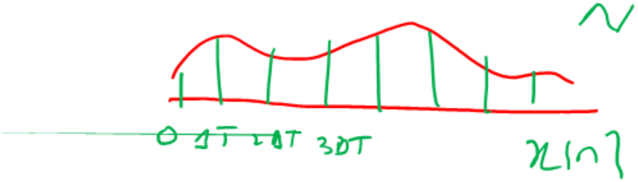
2D FT to 2D DFT

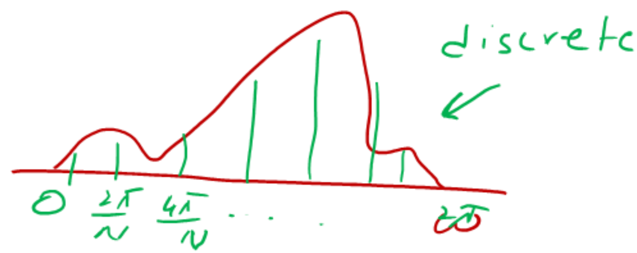
2D FT to 2D DFT

DTFT ≠ DFT

$$X(e^{j\omega}) = \sum_n x[n]\, e^{j\omega n}$$

$$\omega = \frac{2\pi}{N} k$$



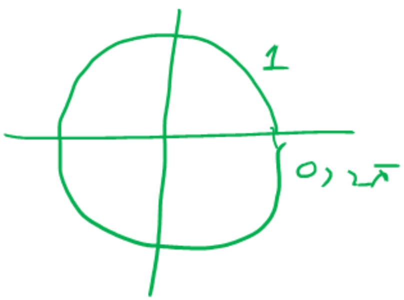


$k = 0, 1, \ldots N-1$

$k = 0 \rightarrow \omega = 0$

$k = 1 \rightarrow \frac{2\pi}{N}$

$k = 2 \rightarrow \frac{2\pi}{N}(2)$



2D FT to 2D DFT

2D FT to 2D DFT

$$X(\omega_1, \omega_2) = \sum_{n_1} \sum_{n_2} x(n_1, n_2)\, e^{-j\omega_1 n_1} e^{-j\omega_2 n_2}$$

$$M \times N \longrightarrow f(x,y) \overset{2D\ FT}{\longleftrightarrow} F(\omega_1, \omega_2)$$

$$F(\omega_1, \omega_2) = \sum_{x} \sum_{y} f(x,y)\, e^{-j\omega_1 x} e^{-j\omega_2 y}$$

$$\omega_1 = \frac{2\pi}{M} u \qquad \omega_2 = \frac{2\pi}{N} v$$

$$u = 0, 1, \ldots M-1 \qquad v = 0, 1, \ldots N-1$$

$$F(u,v) = \sum_{x} \sum_{y} f(x,y)\, e^{-j\frac{2\pi}{M} ux} e^{-j\frac{2\pi}{N} vy}$$

$$= \sum_{x} \sum_{y} f(x,y)\, e^{-j2\pi}$$

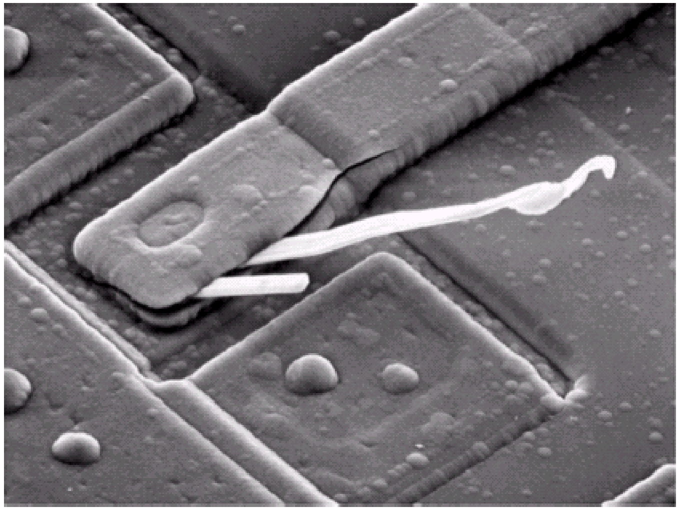
The Discrete Fourier Transform (DFT)

The *Discrete Fourier Transform* of *f(x, y)*, for x = 0, 1, 2…M-1 and y = 0,1,2…N-1, denoted by *F(u, v)*, is given by the equation:

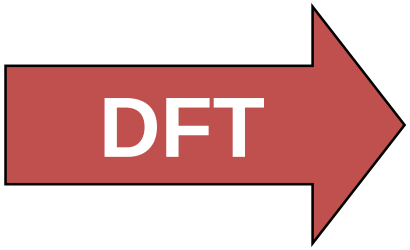
$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(ux/M + vy/N)}$$

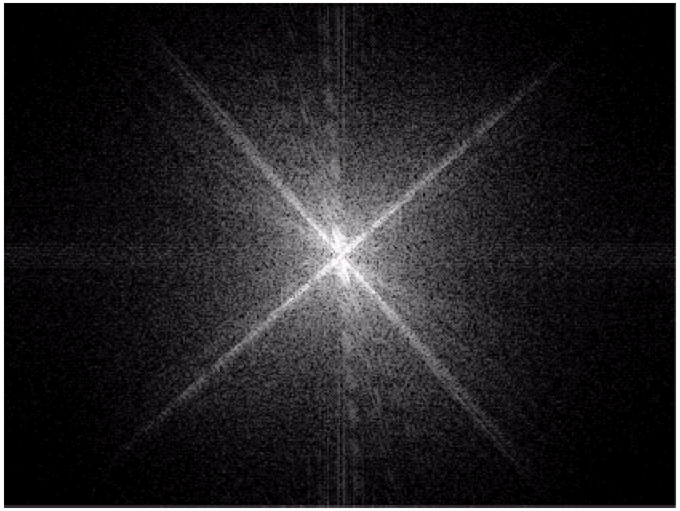
for u = 0, 1, 2…M-1 and v = 0, 1, 2…N-1.

DFT & Images



Scanning electron microscope image of an integrated circuit magnified ~2500 times





Fourier spectrum of the image

The Inverse DFT

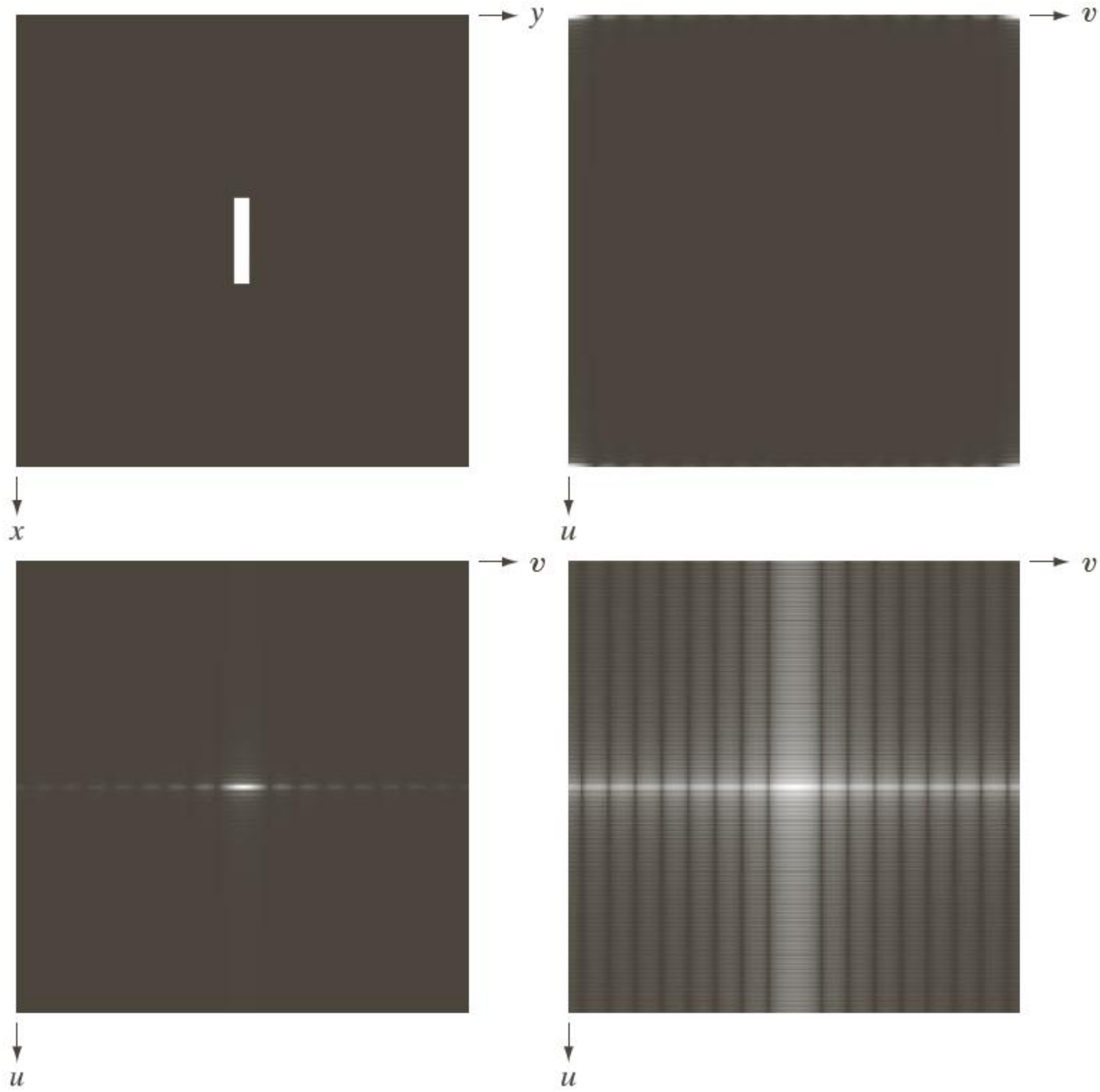
It is really important to note that the Fourier transform is completely **reversible**

The inverse DFT is given by:

$$f(x,y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi(ux/M + vy/N)}$$

for x = 0, 1, 2…M-1 and y = 0, 1, 2…N-1

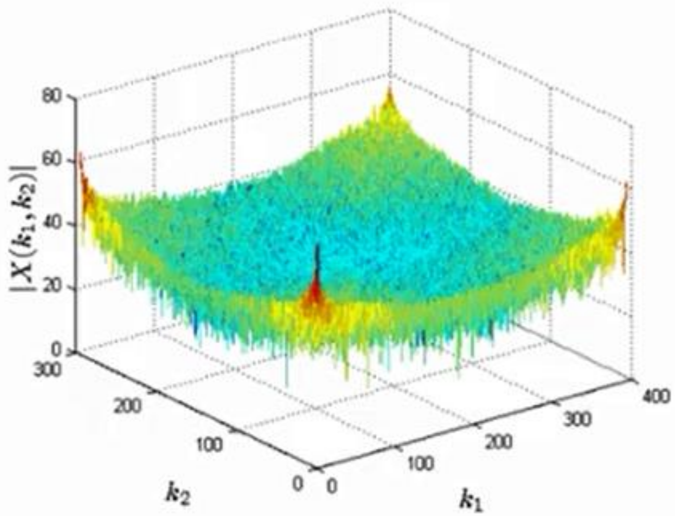
Frequencies in Images

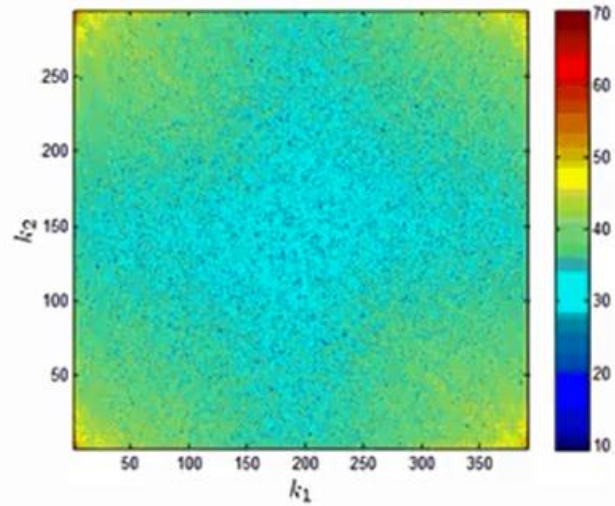


a b
c d

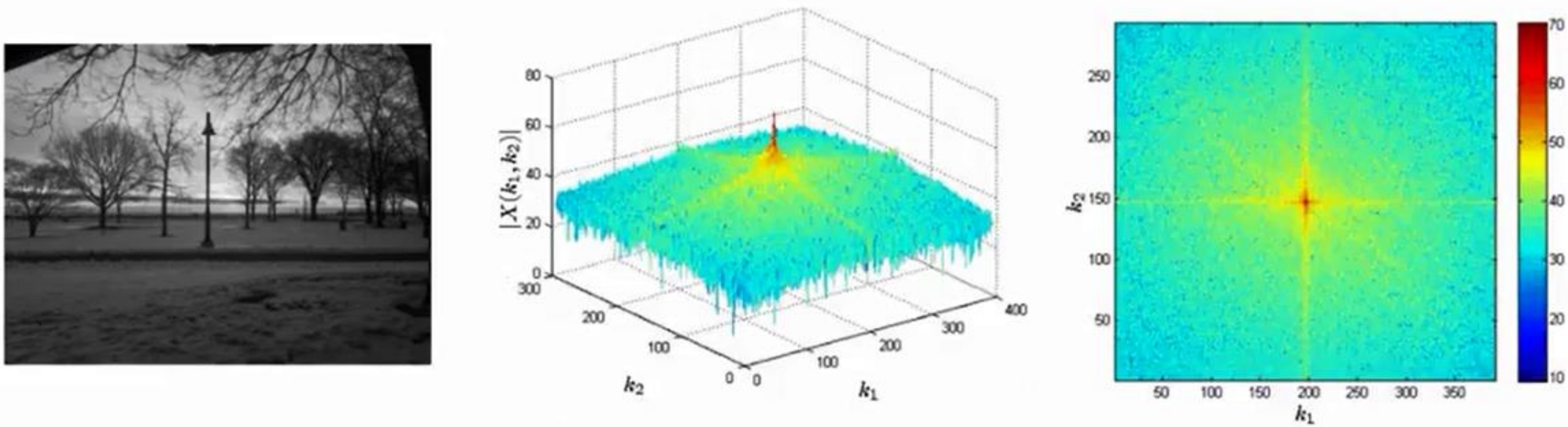
FIGURE 4.24
(a) Image. (b) Spectrum showing bright spots in the four corners. (c) Centered spectrum. (d) Result showing increased detail after a log transformation. The zero crossings of the spectrum are closer in the vertical direction because the rectangle in (a) is longer in that direction. The coordinate convention used throughout the book places the origin of the spatial and frequency domains at the top left.

DFT


$|X(k_1,k_2)|$
0
20
40
60
80
300
200
100
0
k_2
0
100
200
300
400
k_1


250
200
150
100
50
k_2
50
100
150
200
250
300
350
k_1
70
60
50
40
30
20
10

Centered DFT



Readings from Book (3rd Edn.)

- Color Processing Chapter-6
- Fourier Transform Chapter-4

Acknowledgements

- Digital Image Processing", Rafael C. Gonzalez & Richard E. Woods, Addison-Wesley, 2002
- Computer Vision: Algorithms and Applications Richard Szeliski

Material in these slides has been taken from, the following resources