Fourier Transform and Its Applications

Instructor

LE Thanh Sach, Ph.D.

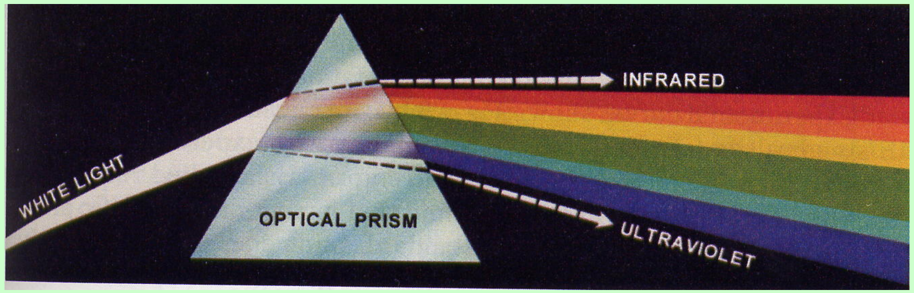
Ltsach@cse.hcmut.edu.vn

Outline

- Fourier Transform
- Its Applications

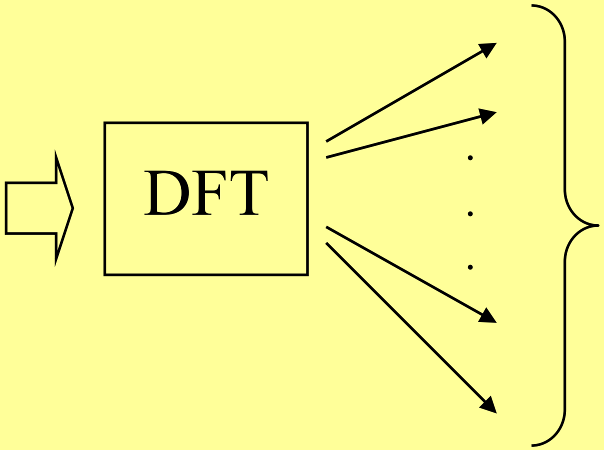
Fourier Analysis

White Light



Color lights

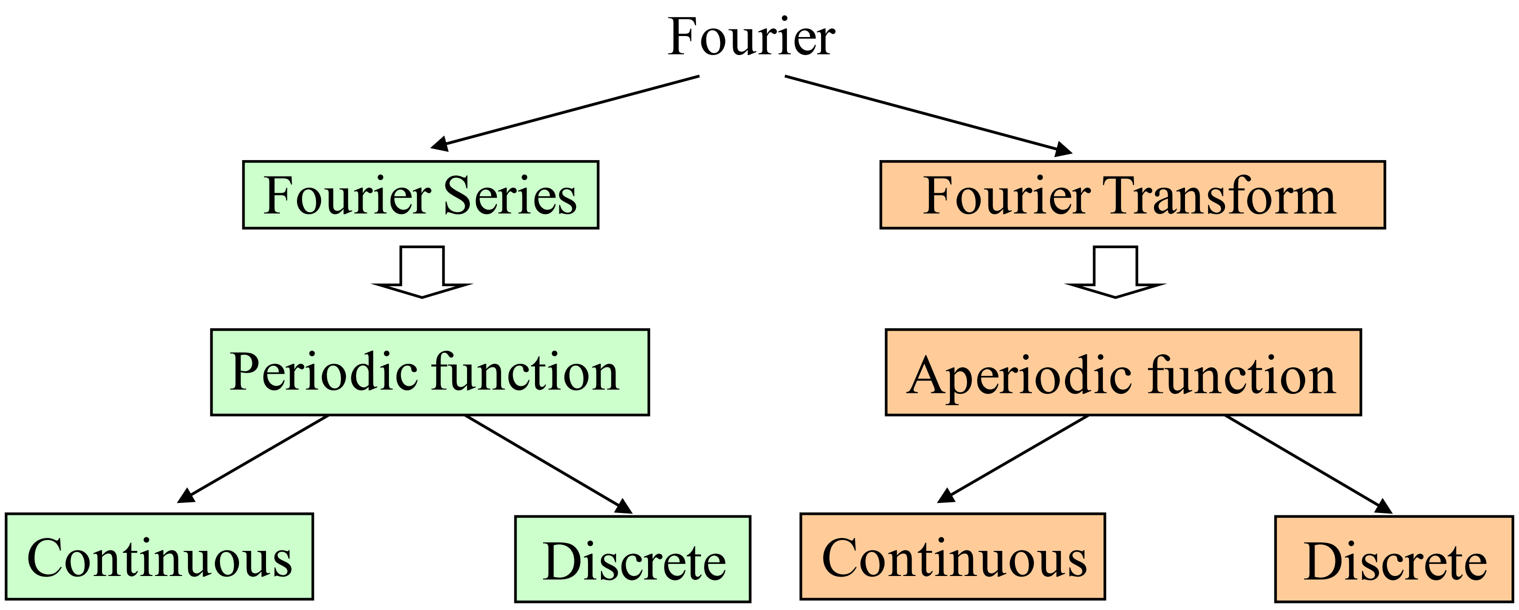
Signal



Signals with different frequencies

Fourier Theory

$$\text{Any functions or signals} = \sum_{u,v} A\sin(\theta) + B\cos(\theta)$$



1D Discrete Fourier Transform
Formula

Forward Transform:

from space domain to frequency domain (Fourier Domain)

$$F(u) = \sum_{x=0}^{M-1} f(x) e^{-j\frac{2\pi}{M}ux} \qquad u = 0,1,.., M$$

Backward Transform (Inverse Transform):

from space domain to frequency domain (Fourier Domain)

$$f(x) = \frac{1}{M} \sum_{u=0}^{M-1} F(u) e^{j\frac{2\pi}{M}ux} \qquad x = 0,1,.., M$$

1D Discrete Fourier Transform

Interpretation

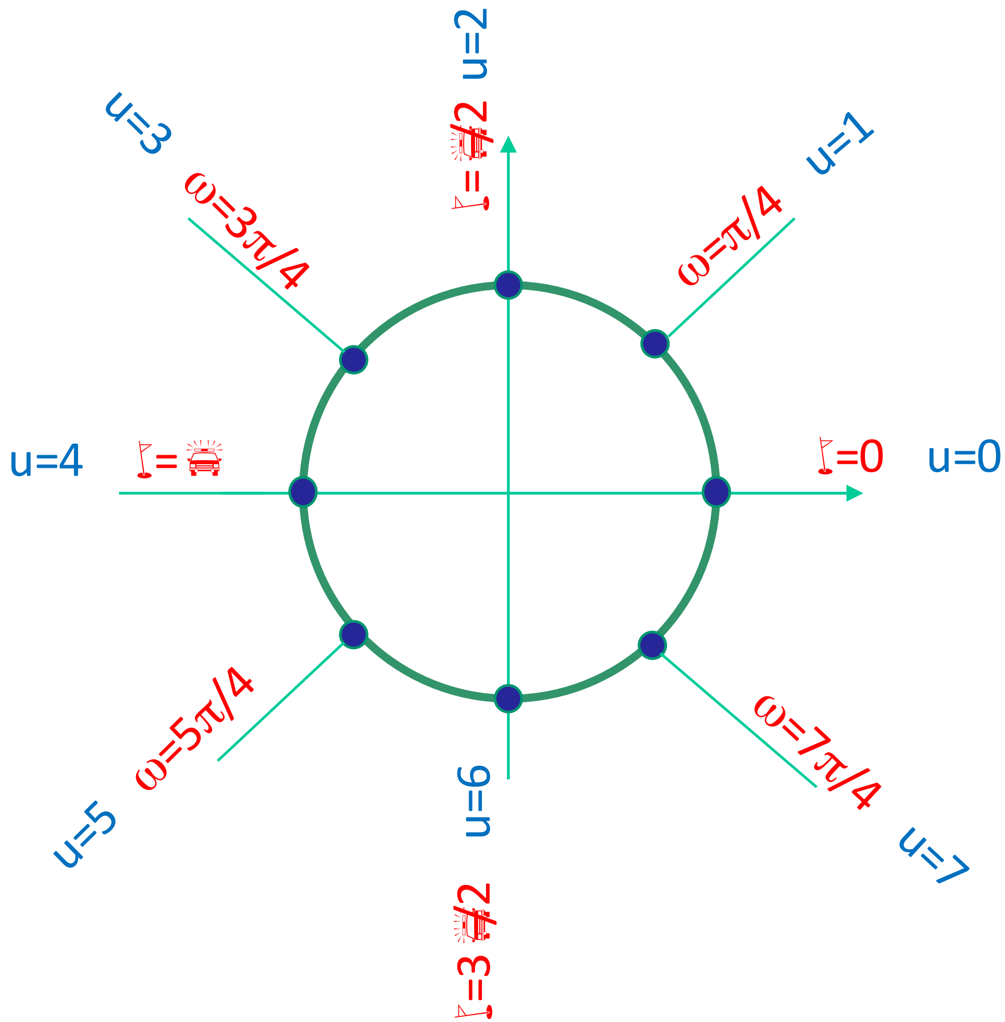
Forward Transform:

$$F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x) e^{-j2\pi\left(\frac{ux}{M}\right)}$$

- M: Number of frequencies that we want to decompose $f(x)$ into them
- u: index of decomposed frequencies, $u = 1..M\text{-}1$
- For example, $M = 8$, i.e., to decompose into 8 components that their frequencies are $\frac{2\pi}{8}u$, $u = 0..7$

1D Discrete Fourier Transform Interpretation

M frequencies
in a trigonometric circle,
For $M = 8$:



1D Discrete Fourier Transform Interpretation

Fourier transform's meaning:

$$F(u) = \sum_{x=0}^{M-1} f(x) e^{-j\frac{2\pi}{M}ux} \qquad u = 0,1,.., M$$

F(u) tells that how much frequency component $\frac{2\pi}{M}u$ contribute to signal *f(x)*. In the case that *F(u)* = 0 then there is no frequency $\frac{2\pi}{M}u$ the input signal. Otherwise, if the magnitude of *F(u)* is significantly larger than other frequencies' then frequency $\frac{2\pi}{M}u$ contribute much to the signal, and the shape of the signal tends to be similar with the shape of frequency $\frac{2\pi}{M}u$.

1D Discrete Fourier Transform Interpretation

Fourier transform meaning:

$$F(u) = \sum_{x=0}^{M-1} f(x) e^{-j\frac{2\pi}{M}ux} \quad u = 0,1,.., M$$

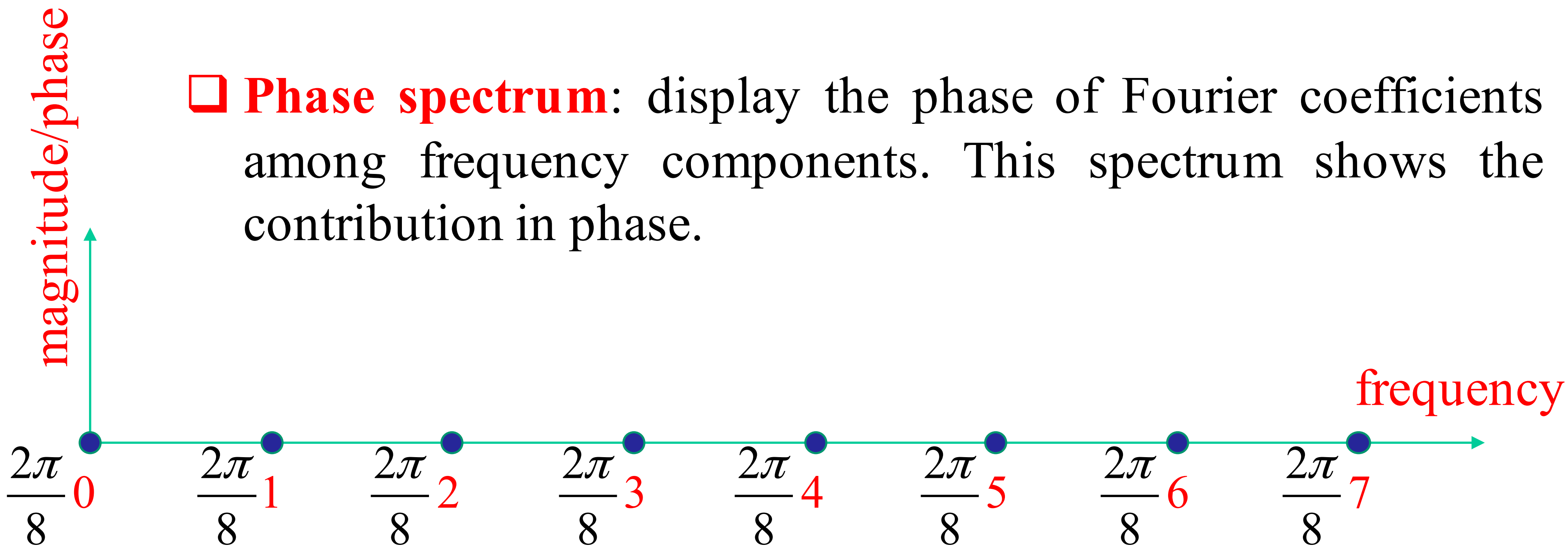
In dot product/ correlation

1D Discrete Fourier Transform Interpretation

Fourier transform's meaning:

- **Magnitude spectrum**: display the magnitude of Fourier coefficients among frequency components. This spectrum can tell how much a frequency contribute to the input signal.

- **Phase spectrum**: display the phase of Fourier coefficients among frequency components. This spectrum shows the contribution in phase.



1D Discrete Fourier Transform Interpretation

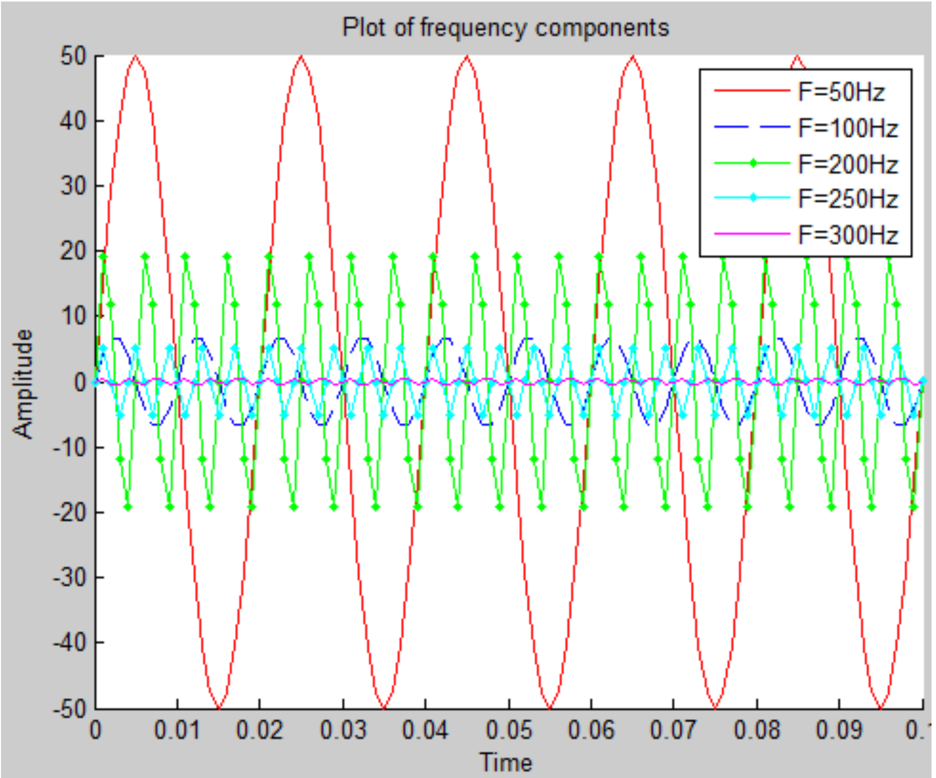
Fourier transform's meaning:

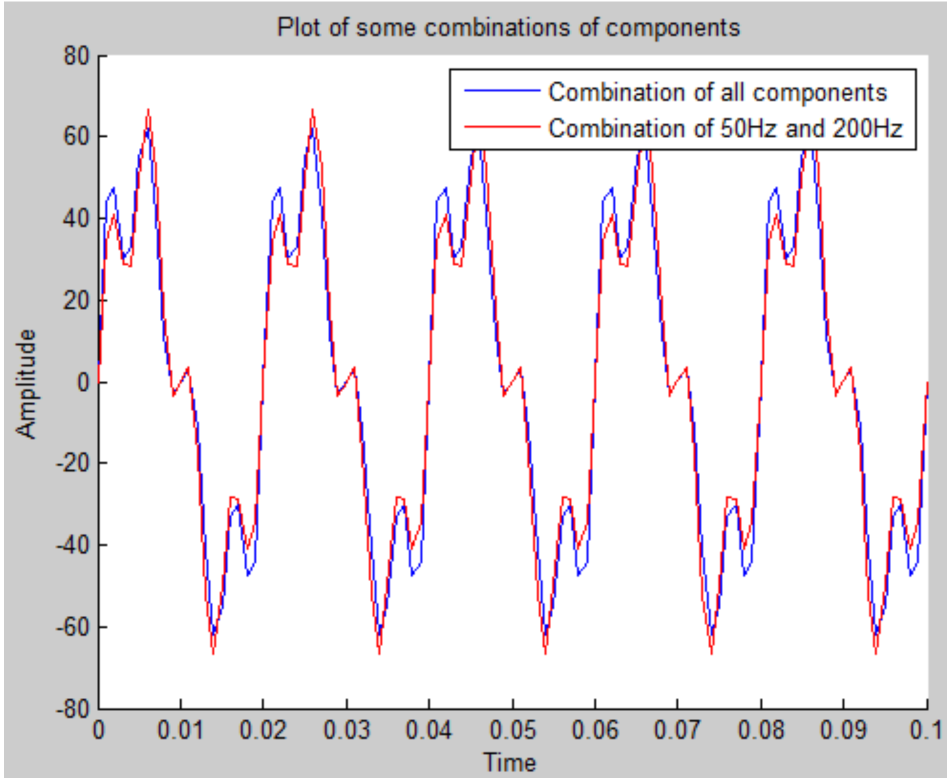
Consider the following signals $y_k(t) = C_k * \sin(2\pi F_k t)$; where, C_k and F_k are given in the following table.

k	F_k (Hertz)	C_k
1	50	50
2	100	7
3	200	20
4	250	5
5	300	0.5

1D Discrete Fourier Transform Interpretation

Fourier transform's meaning:

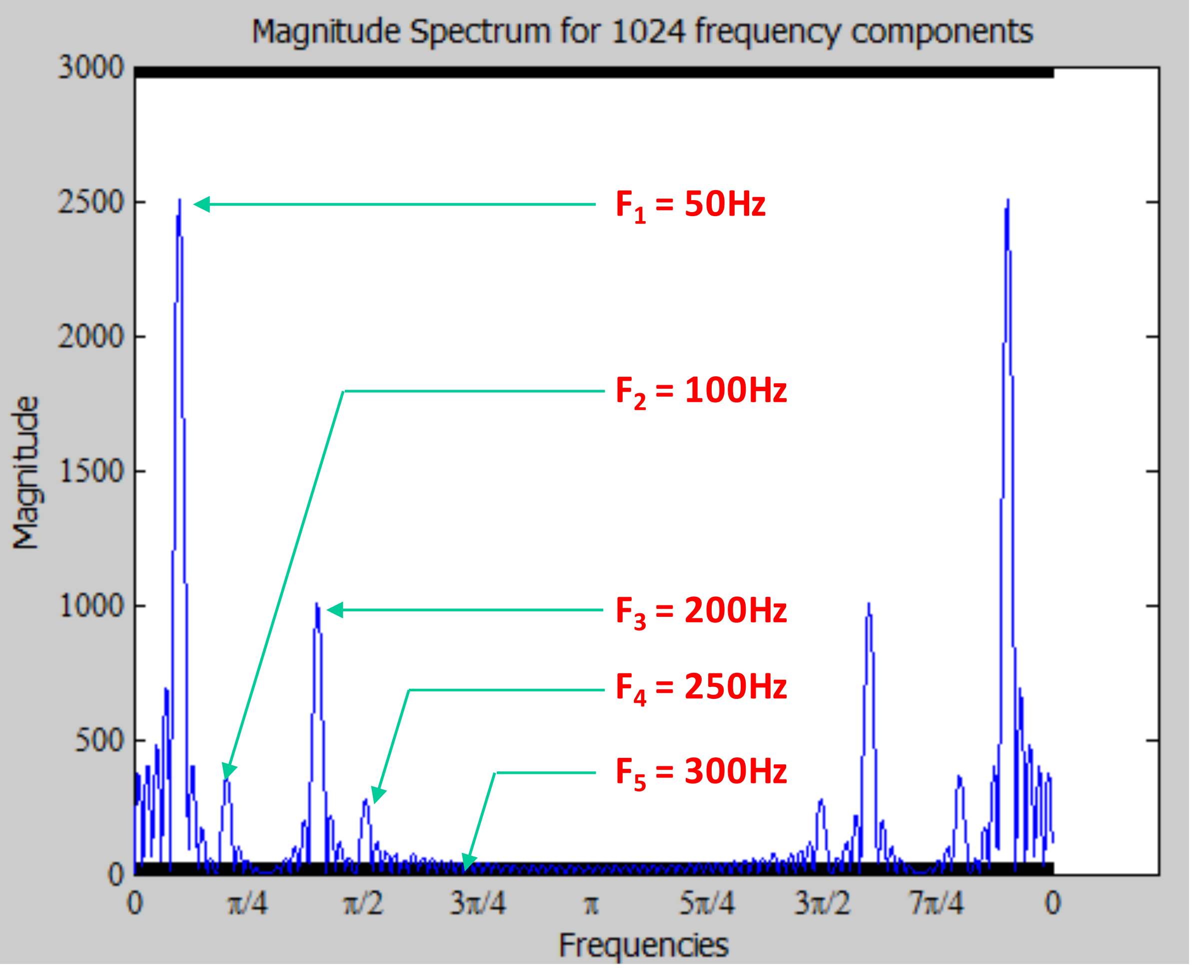




A combination of only significantly contributed components can approximate the signal that contains all the components.

1D Discrete Fourier Transform Interpretation

Fourier transform's meaning:



1D Discrete Fourier Transform

Matrix form

$$F(u) = \sum_{x=0}^{M-1} f(x) \boxed{e^{-j\frac{2\pi}{M}ux}}$$

Example for $M = 4$

Kernel Matrix:

x=0 x=1 x=3 x=4

$$W_4 = \begin{bmatrix} e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}0} \\ e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}1} & e^{-j\frac{2\pi}{4}2} & e^{-j\frac{2\pi}{4}3} \\ e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}2} & e^{-j\frac{2\pi}{4}4} & e^{-j\frac{2\pi}{4}6} \\ e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}3} & e^{-j\frac{2\pi}{4}6} & e^{-j\frac{2\pi}{4}9} \end{bmatrix} \begin{matrix} u=0 \\ u=1 \\ u=2 \\ u=3 \end{matrix}$$

1D Discrete Fourier Transform

Matrix form

$$W_4 = \begin{bmatrix} e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}0} \\ e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}1} & e^{-j\frac{2\pi}{4}2} & e^{-j\frac{2\pi}{4}3} \\ e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}2} & e^{-j\frac{2\pi}{4}4} & e^{-j\frac{2\pi}{4}6} \\ e^{-j\frac{2\pi}{4}0} & e^{-j\frac{2\pi}{4}3} & e^{-j\frac{2\pi}{4}6} & e^{-j\frac{2\pi}{4}9} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & e^{-j\frac{\pi}{2}} & e^{-j\pi} & e^{-j3\frac{\pi}{2}} \\ 1 & e^{-j\pi} & e^{-j2\pi} & e^{-j\pi} \\ 1 & e^{-j3\frac{\pi}{2}} & e^{-j\pi} & e^{-j\frac{\pi}{2}} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

1D Discrete Fourier Transform

Matrix form

$$F(u) = \sum_{x=0}^{M-1} f(x) e^{-j\frac{2\pi}{M}ux}$$

Example for ***M* = 4**

Vector signal:

$$f = \begin{bmatrix} 2 \\ 0 \\ 3 \\ 0 \end{bmatrix}$$

1D Discrete Fourier Transform

Matrix form

$$F(u) = \sum_{x=0}^{M-1} f(x)\, e^{-j\frac{2\pi}{M}ux}$$

$$= \underbrace{W_M}_{\text{Matrix}} * \underbrace{f}_{\text{Vector}}$$

1D Discrete Fourier Transform
Example

Example:

- ***f(x)*** **= {2 0 3 0}**
- ***M*** **= 4: to decompose into 4 frequency components**
- ***x*** **= 0, 1, 2, 3**
- ***u*** **= 0, 1, 2, 3**

- **So,** $e^{-j\frac{2\pi}{M}ux}$ **forms a matrix of** W_4**, size of 4×4**

1D Discrete Fourier Transform

Example

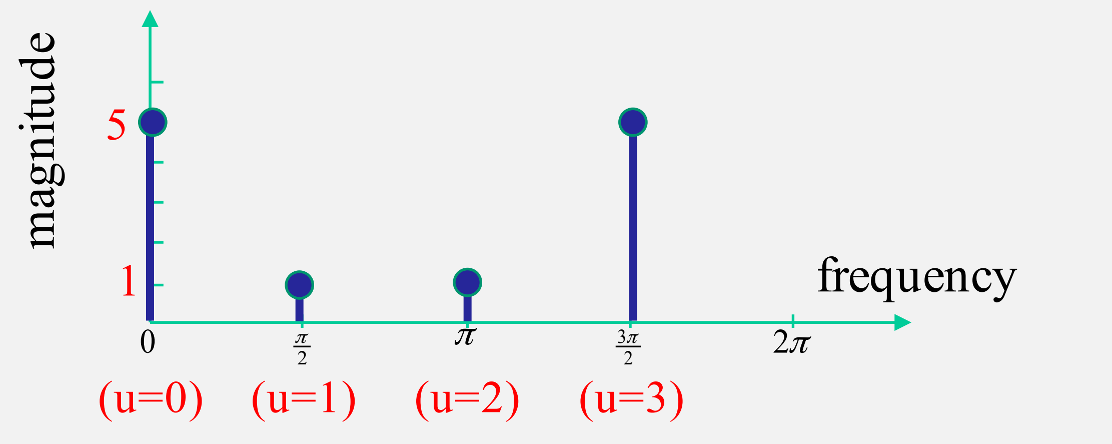
$$f = [2 \quad 0 \quad 3 \quad 0]^T$$

$$F(u) = W_4 f$$

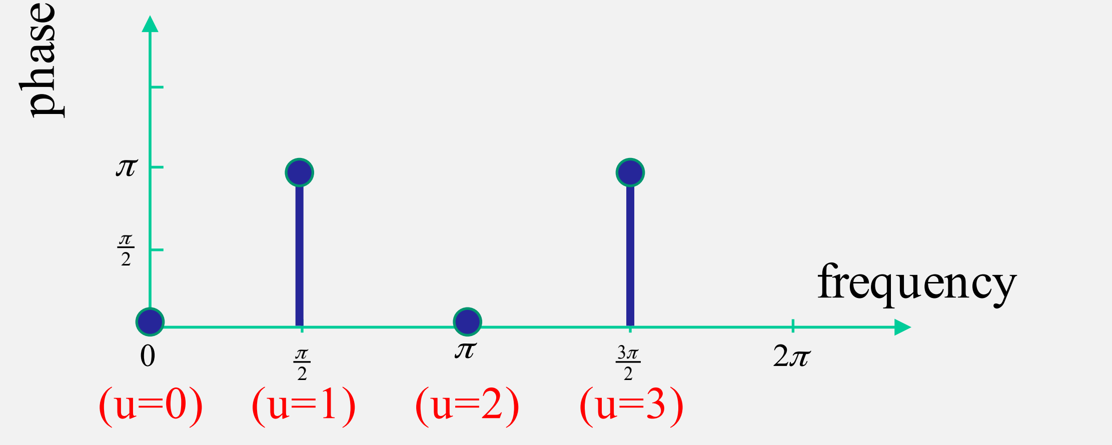
$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ 3 \\ 0 \end{bmatrix} = \begin{bmatrix} 5 \\ -1 \\ 5 \\ -1 \end{bmatrix}$$

1D Discrete Fourier Transform Example

$$F(u) = \begin{bmatrix} 5 \\ -1 \\ 5 \\ -1 \end{bmatrix}$$

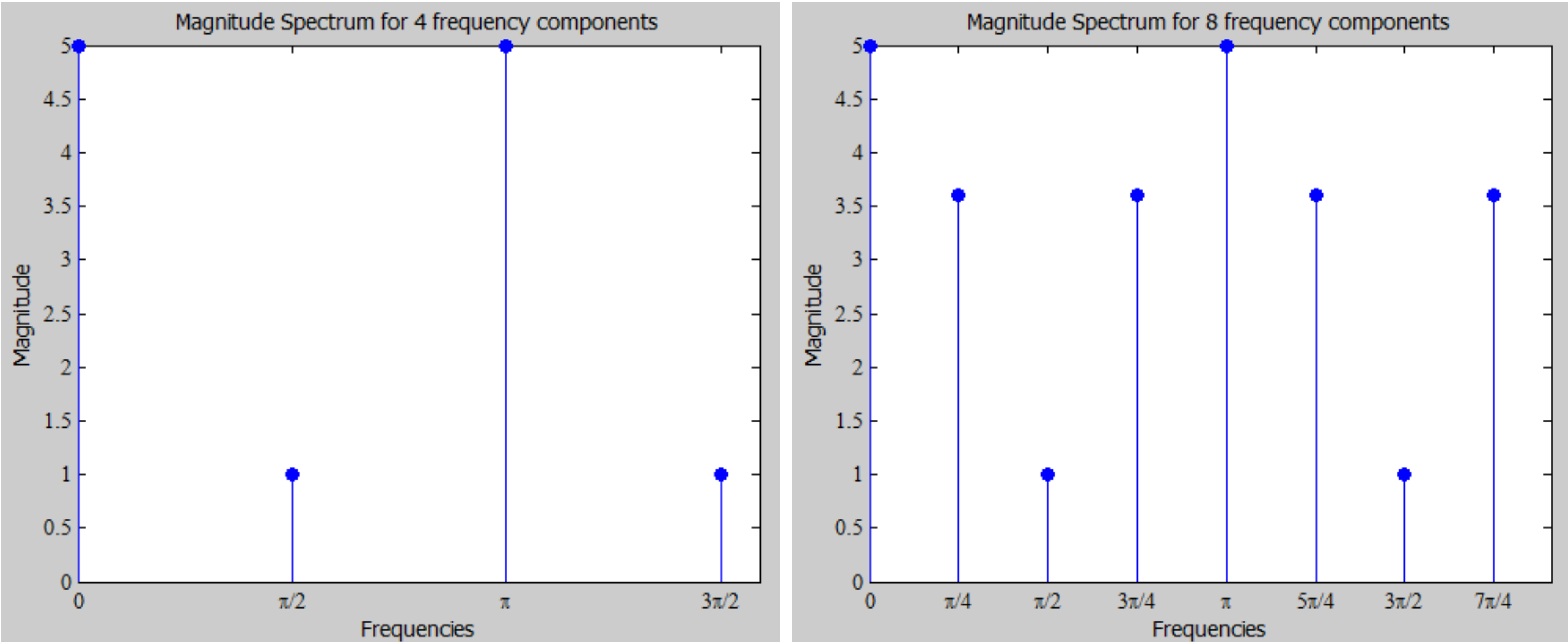


Magnitude spectrum



Phase spectrum

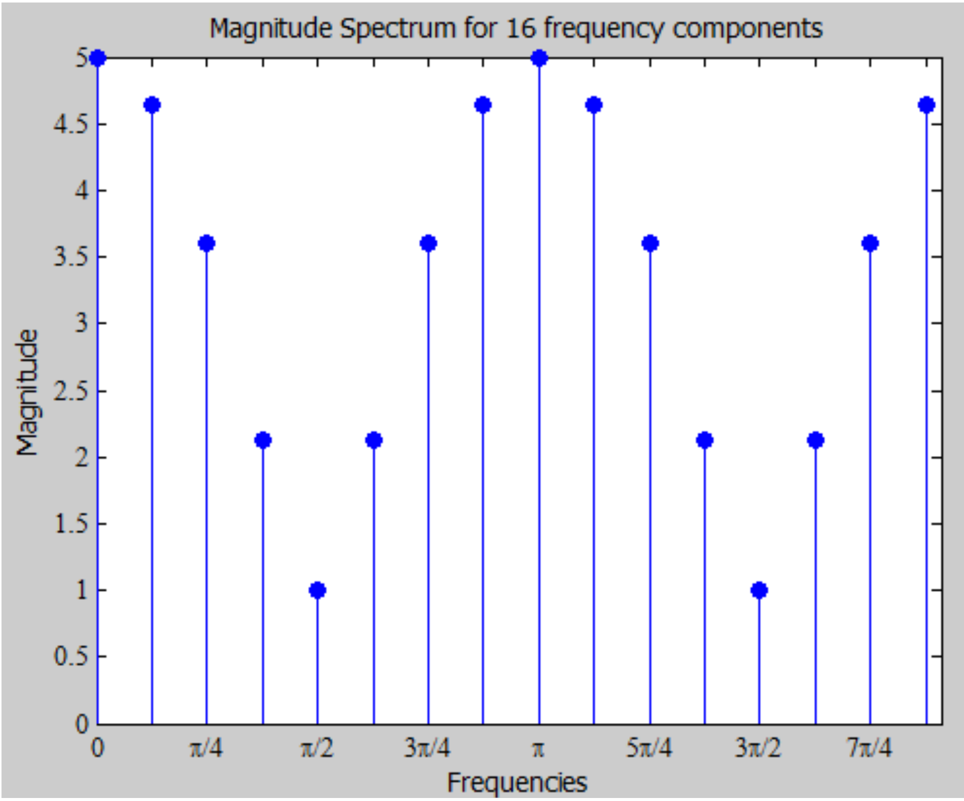
1D Discrete Fourier Transform Example

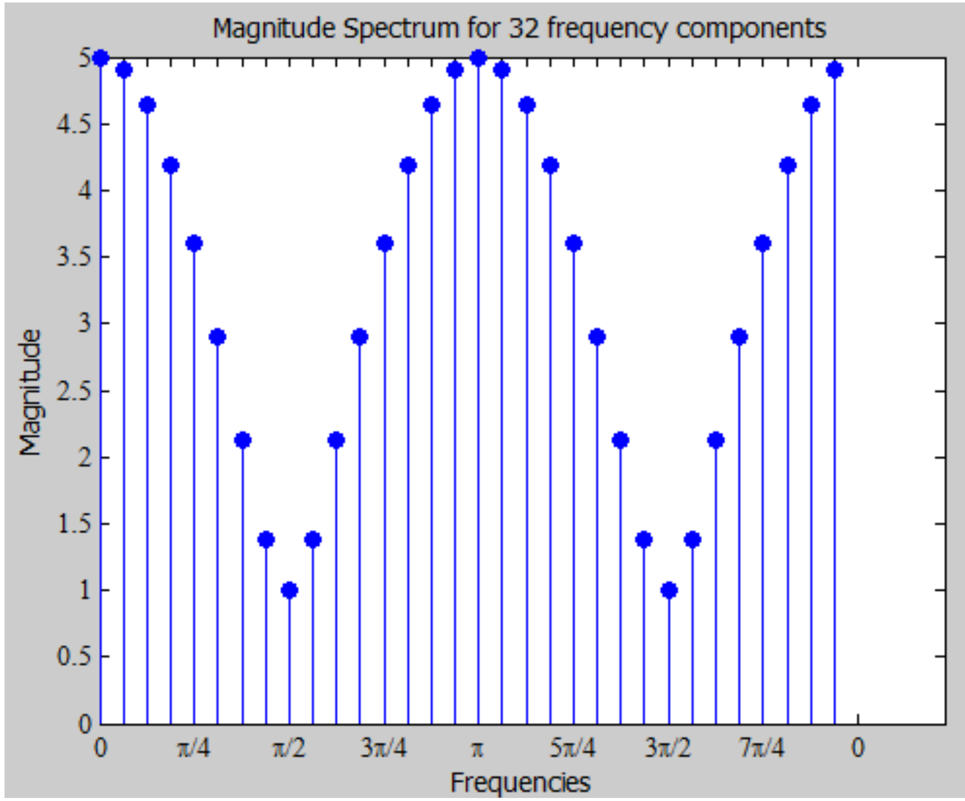


$F(\omega = 0) = F(\omega = \pi) = 5$

$F(\omega = \pi/2) = F(\omega = 3\pi/2) = 1$

1D Discrete Fourier Transform Example

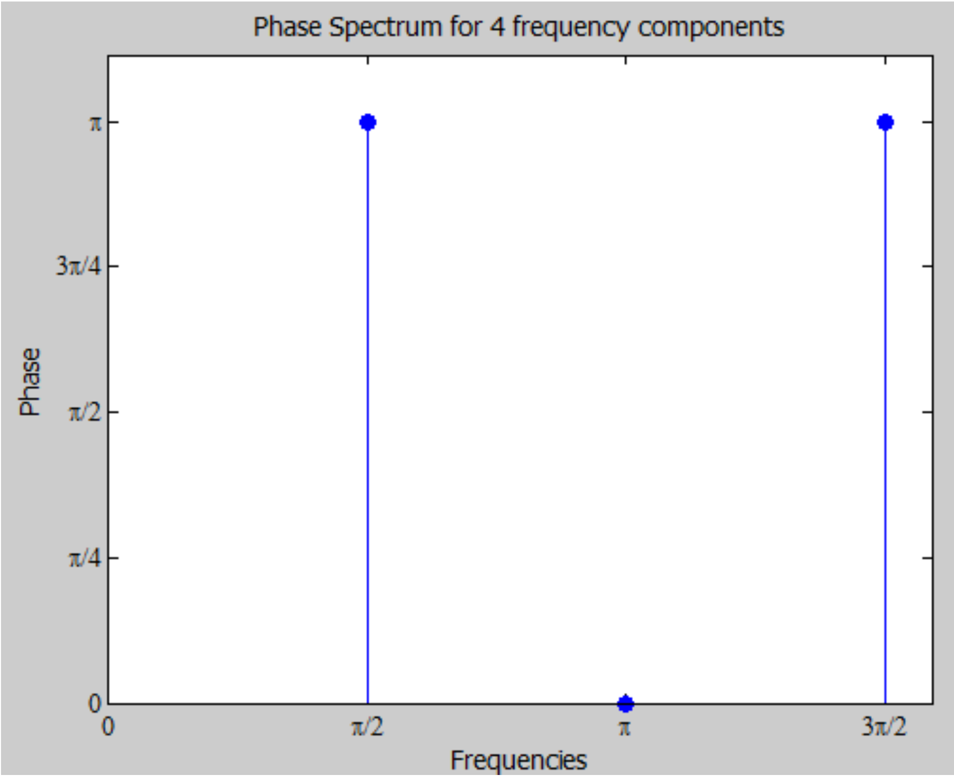


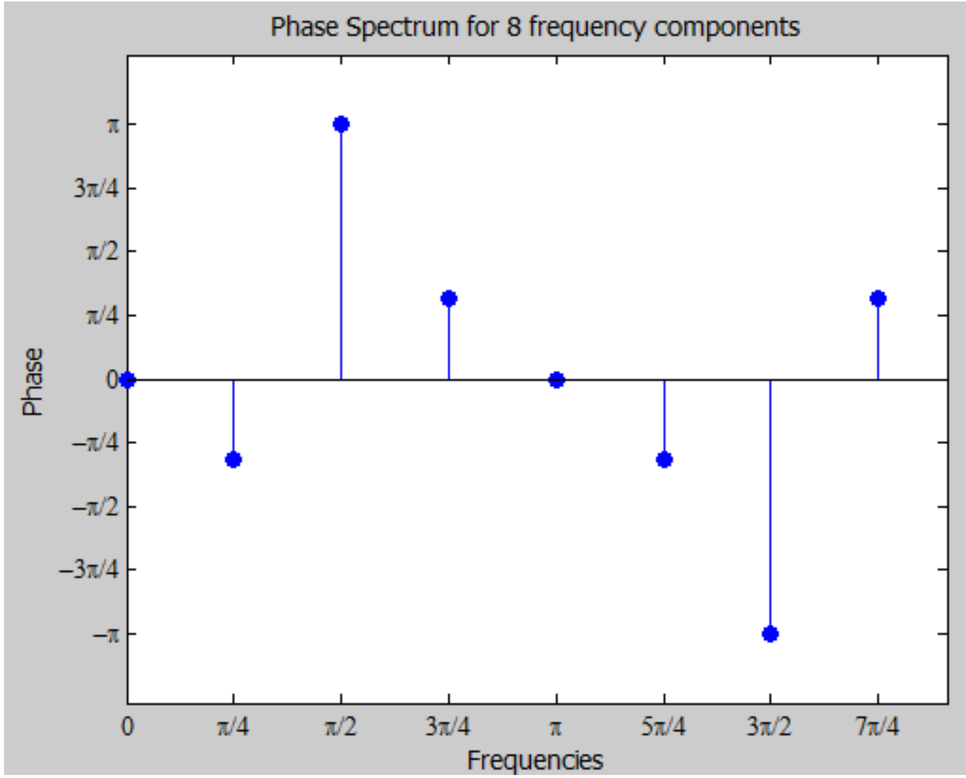


$F(\omega = 0) = F(\omega = \pi) = 5$

$F(\omega = \pi/2) = F(\omega = 3\pi/2) = 1$

1D Discrete Fourier Transform Example

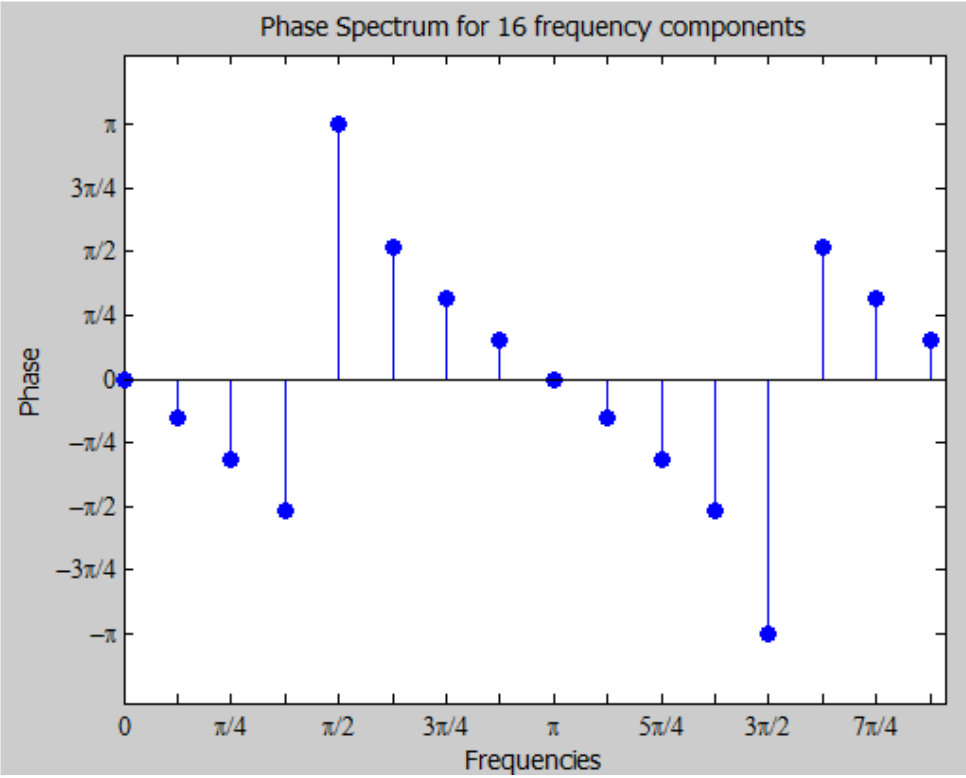


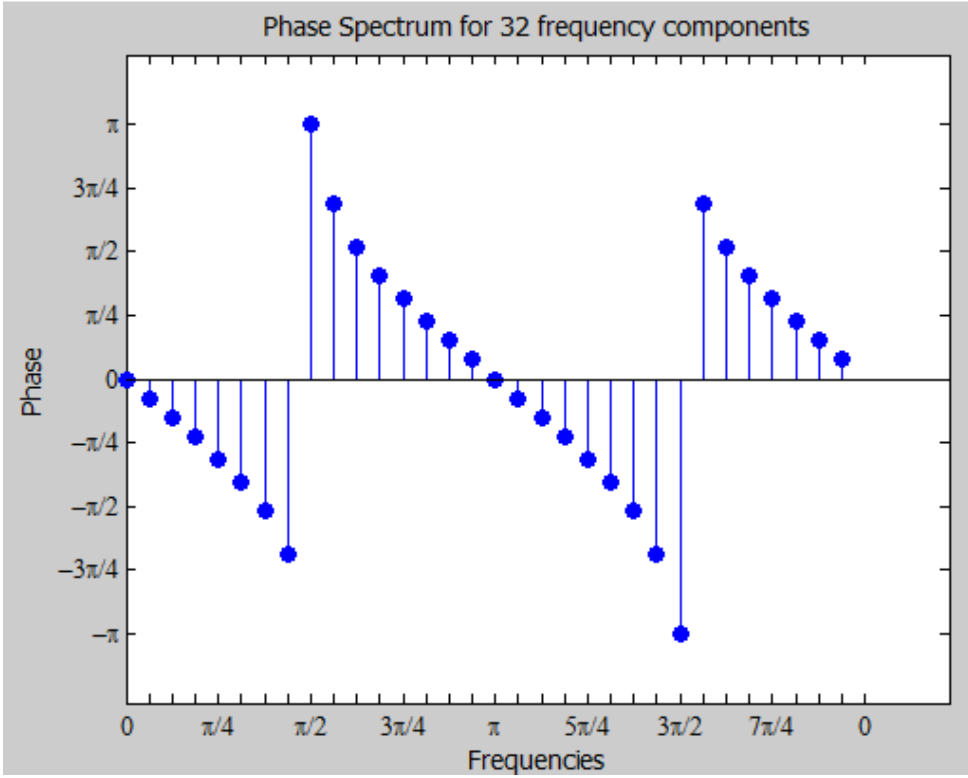


$F(\omega = 0) = F(\omega = \pi) = 0$ (radian)

$F(\omega = \pi/2) = F(\omega = 3\pi/2) = \pi$ (or $-\pi$ the same value)

1D Discrete Fourier Transform Example





$F(\omega = 0) = F(\omega = \pi) = 0$ (radian)

$F(\omega = \pi/2) = F(\omega = 3\pi/2) = \pi$ (or $-\pi$ the same value)

1D Discrete Fourier Transform

Matrix form

Kernel Matrix:

x=0 x=1 x=3 x=4 …

$$W_8 = \begin{bmatrix}
e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}0} \\
e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}1} & e^{-j\frac{\pi}{4}2} & e^{-j\frac{\pi}{4}3} & e^{-j\frac{\pi}{4}4} & e^{-j\frac{\pi}{4}5} & e^{-j\frac{\pi}{4}6} & e^{-j\frac{\pi}{4}7} \\
e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}2} & e^{-j\frac{\pi}{4}4} & e^{-j\frac{\pi}{4}6} & e^{-j\frac{\pi}{4}8} & e^{-j\frac{\pi}{4}10} & e^{-j\frac{\pi}{4}12} & e^{-j\frac{\pi}{4}14} \\
e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}3} & e^{-j\frac{\pi}{4}6} & e^{-j\frac{\pi}{4}9} & e^{-j\frac{\pi}{4}12} & e^{-j\frac{\pi}{4}15} & e^{-j\frac{\pi}{4}18} & e^{-j\frac{\pi}{4}21} \\
e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}4} & e^{-j\frac{\pi}{4}8} & e^{-j\frac{\pi}{4}12} & e^{-j\frac{\pi}{4}16} & e^{-j\frac{\pi}{4}20} & e^{-j\frac{\pi}{4}24} & e^{-j\frac{\pi}{4}28} \\
e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}5} & e^{-j\frac{\pi}{4}10} & e^{-j\frac{\pi}{4}15} & e^{-j\frac{\pi}{4}20} & e^{-j\frac{\pi}{4}25} & e^{-j\frac{\pi}{4}30} & e^{-j\frac{\pi}{4}35} \\
e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}6} & e^{-j\frac{\pi}{4}12} & e^{-j\frac{\pi}{4}18} & e^{-j\frac{\pi}{4}24} & e^{-j\frac{\pi}{4}30} & e^{-j\frac{\pi}{4}36} & e^{-j\frac{\pi}{4}42} \\
e^{-j\frac{\pi}{4}0} & e^{-j\frac{\pi}{4}7} & e^{-j\frac{\pi}{4}14} & e^{-j\frac{\pi}{4}21} & e^{-j\frac{\pi}{4}28} & e^{-j\frac{\pi}{4}35} & e^{-j\frac{\pi}{4}42} & e^{-j\frac{\pi}{4}49}
\end{bmatrix}$$

u=0
u=1
u=2
u=3
⋮

1D Discrete Fourier Transform

Matrix form

Kernel Matrix:

x=0 x=1 x=3 x=4 ...

$$W_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & e^{-j\frac{\pi}{4}} & e^{-j\frac{\pi}{2}} & e^{-j\frac{3\pi}{4}} & e^{-j\pi} & e^{-j\frac{5\pi}{4}} & e^{-j\frac{3\pi}{2}} & e^{-j\frac{7\pi}{4}} \\ 1 & e^{-j\frac{\pi}{2}} & e^{-j\pi} & e^{-j\frac{3\pi}{2}} & e^{-j2\pi} & e^{-j\frac{\pi}{2}} & e^{-j\pi} & e^{-j\frac{3\pi}{2}} \\ 1 & e^{-j\frac{3\pi}{4}} & e^{-j\frac{3\pi}{2}} & e^{-j\frac{\pi}{4}} & e^{-j\pi} & e^{-j\frac{7\pi}{4}} & e^{-j\frac{\pi}{2}} & e^{-j\frac{5\pi}{4}} \\ 1 & e^{-j\pi} & e^{-j2\pi} & e^{-j\pi} & e^{-j2\pi} & e^{-j\pi} & e^{-j2\pi} & e^{-j\pi} \\ 1 & e^{-j\frac{5\pi}{4}} & e^{-j\frac{\pi}{2}} & e^{-j\frac{7\pi}{4}} & e^{-j\pi} & e^{-j\frac{\pi}{4}} & e^{-j\frac{3\pi}{2}} & e^{-j\frac{3\pi}{4}} \\ 1 & e^{-j\frac{3\pi}{2}} & e^{-j\pi} & e^{-j\frac{\pi}{2}} & e^{-j2\pi} & e^{-j\frac{3\pi}{2}} & e^{-j\pi} & e^{-j\frac{\pi}{2}} \\ 1 & e^{-j\frac{7\pi}{4}} & e^{-j\frac{3\pi}{2}} & e^{-j\frac{5\pi}{4}} & e^{-j\pi} & e^{-j\frac{3\pi}{4}} & e^{-j\frac{\pi}{2}} & e^{-j\frac{\pi}{4}} \end{bmatrix}$$

u=0
u=1
u=2
u=3
⋮

1D Discrete Fourier Transform

Matrix form

Kernel Matrix:

$$W_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & \frac{\sqrt{2}}{2}(1-j) & -j & -\frac{\sqrt{2}}{2}(1+j) & -1 & -\frac{\sqrt{2}}{2}(1-j) & j & \frac{\sqrt{2}}{2}(1+j) \\ 1 & -j & -1 & j & 1 & -j & -1 & j \\ 1 & -\frac{\sqrt{2}}{2}(1+j) & j & \frac{\sqrt{2}}{2}(1-j) & -1 & \frac{\sqrt{2}}{2}(1+j) & -j & -\frac{\sqrt{2}}{2}(1-j) \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -\frac{\sqrt{2}}{2}(1-j) & -j & \frac{\sqrt{2}}{2}(1+j) & -1 & \frac{\sqrt{2}}{2}(1-j) & j & -\frac{\sqrt{2}}{2}(1+j) \\ 1 & j & -1 & -j & 1 & -j & -1 & -j \\ 1 & \frac{\sqrt{2}}{2}(1+j) & j & -\frac{\sqrt{2}}{2}(1+j) & -1 & -\frac{\sqrt{2}}{2}(1+j) & -j & \frac{\sqrt{2}}{2}(1-j) \end{bmatrix}$$

1D Discrete Fourier Transform

Matrix form

<u>Kernel Matrix's Properties:</u>

- ❑ Symmetric on the main diagonal

- ❑ Let $c_M^u = e^{-j\frac{2\pi}{M}u}$

  Then

$$c_M^{u+\frac{M}{2}} = -c_M^u$$

$$c_M^{u+M} = c_M^u$$

1D Discrete Fourier Transform

Question

Signal: $f = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{bmatrix}$ ⇒ $F(u) = W_8 f$

$= ?$

1. Draw magnitude and phase spectrum for $F(u)$
2. How much does frequency $\omega = 3\pi/4$ contribute to the signal (in magnitude and in phase)?
3. Write Matlab/C/CPP function program to compute DFT/FFT.
4. Let $M_1 = 256$ and $M_2 = 8$ be number of frequency into which we decompose, and $F_{256}(u)$ and $F_8(u)$ are Fourier coefficients for those two transforms. Assume that we already have $F_{256}(u)$, how can obtain $F_8(u)$ from $F_{256}(u)$?

2D Discrete Fourier Transform

Formula

Forward transform:

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

u=0..M-1 v=0..N-1

Backward transform:

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

x=0..M-1 y=0..N-1

2D Discrete Fourier Transform

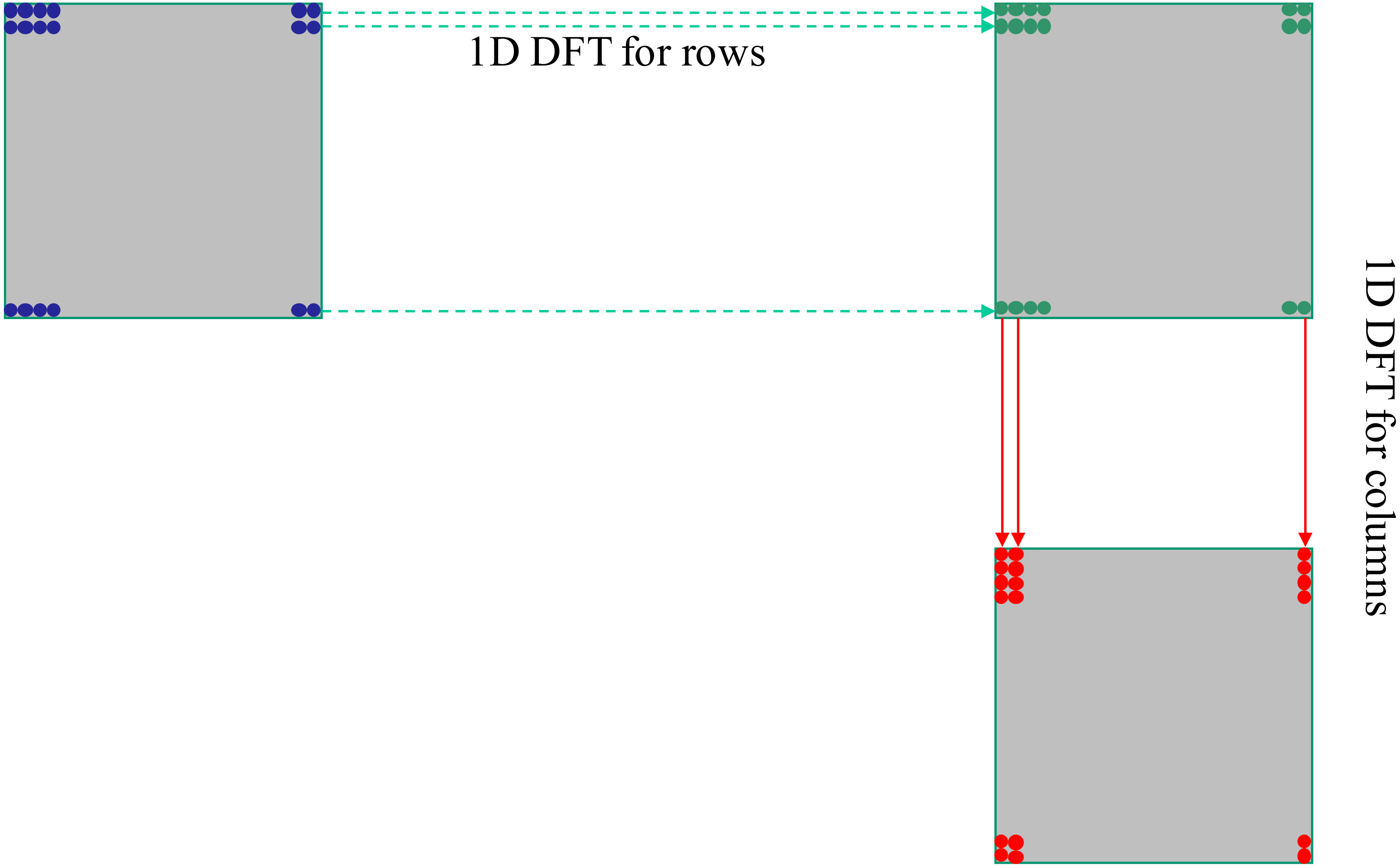
Computation

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

$$= \sum_{y=0}^{N-1}\underbrace{\sum_{x=0}^{M-1} f(x,y)e^{-j\frac{2\pi}{M}ux}}_{\text{1D DFT for rows}} e^{-j\frac{2\pi}{N}vx}$$

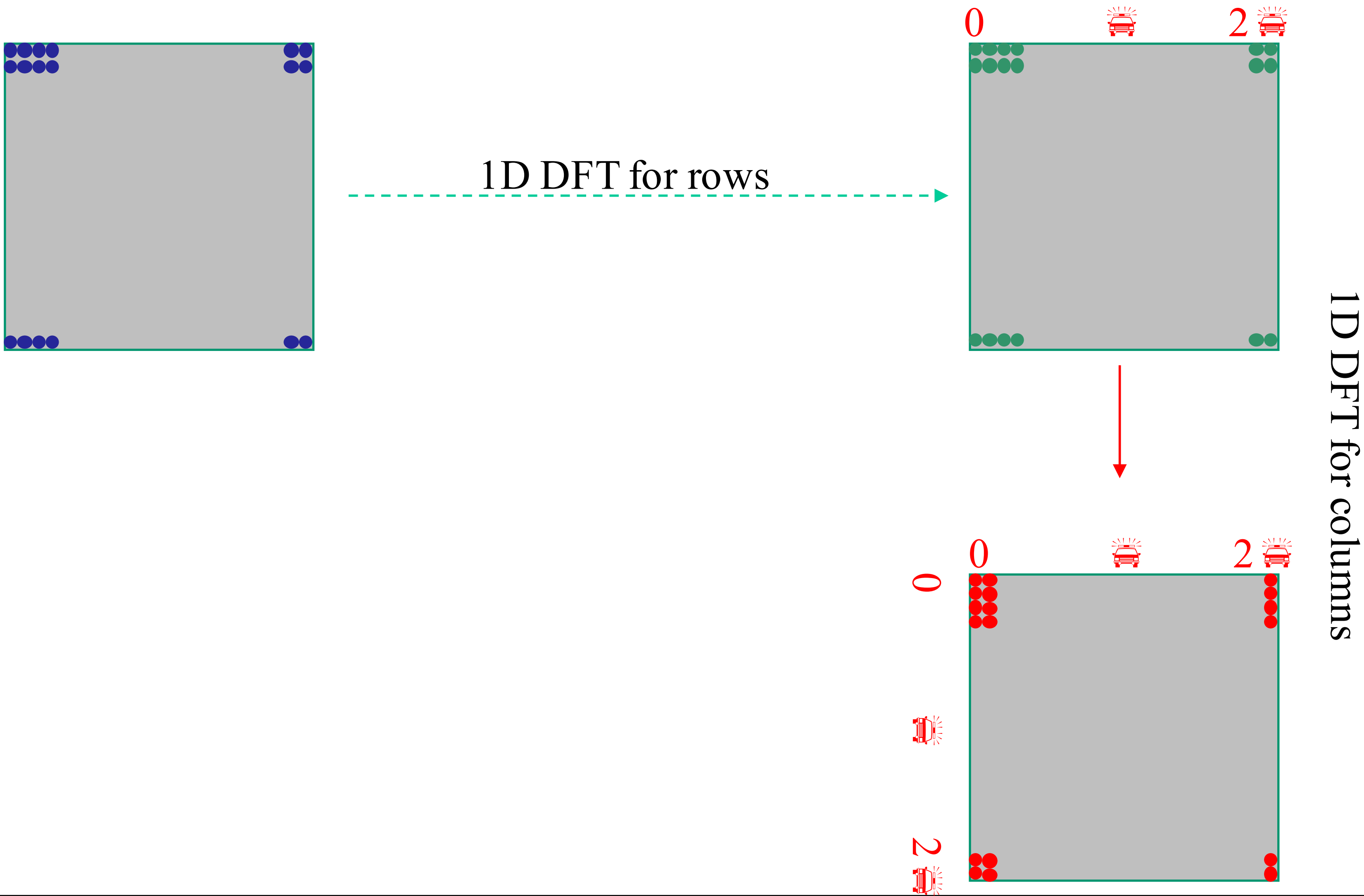
$$= \underbrace{\sum_{y=0}^{N-1} F(u,y)e^{-j\frac{2\pi}{N}vx}}_{\text{1D DFT for columns}}$$

2D Discrete Fourier Transform
Computation

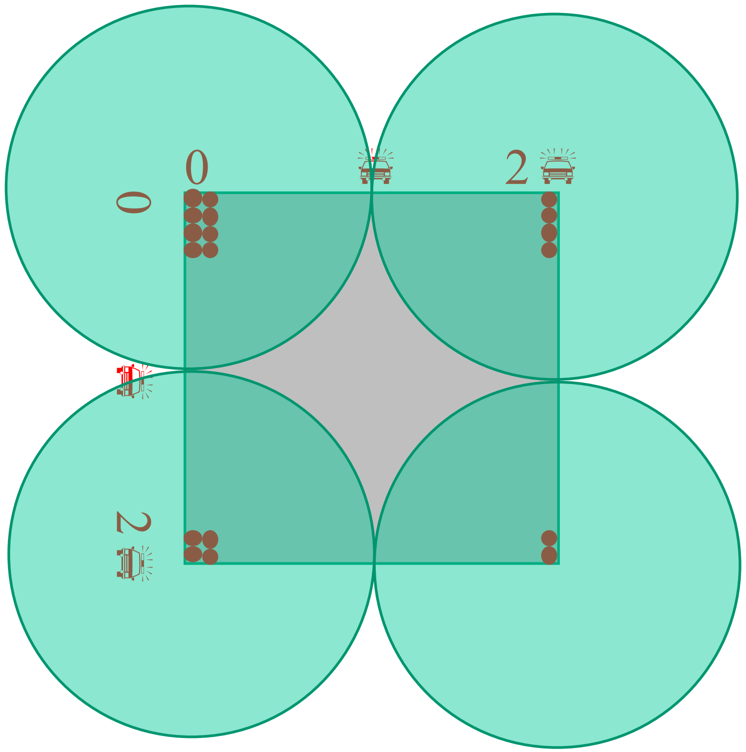


2D Discrete Fourier Transform

Computation



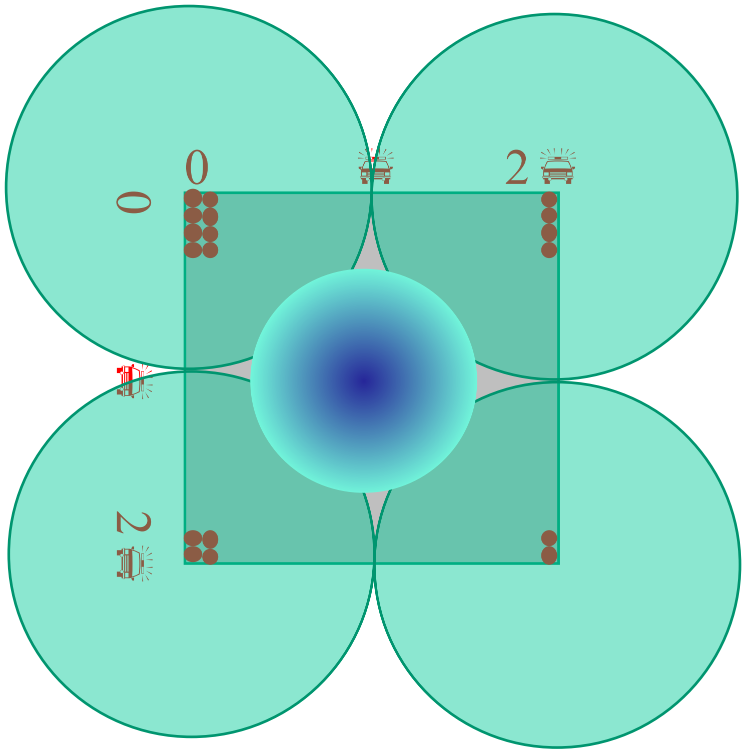
2D Discrete Fourier Transform
Computation



- Low frequencies are at four corners
- High frequencies are at the center
- Low frequencies mean intensities inside the input images have small variation. For example, perfectly smooth image has only one frequency, 0 radian.

2D Discrete Fourier Transform

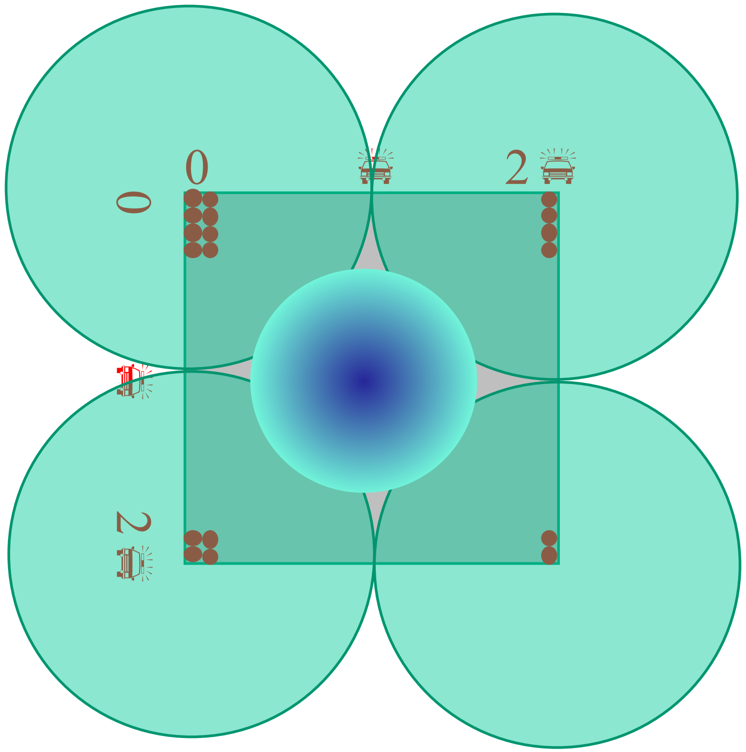
Computation



- Low frequencies are at four corners
- High frequencies are at the center
- High frequencies mean the image has a large variation. For example, images have high variation around their edges.

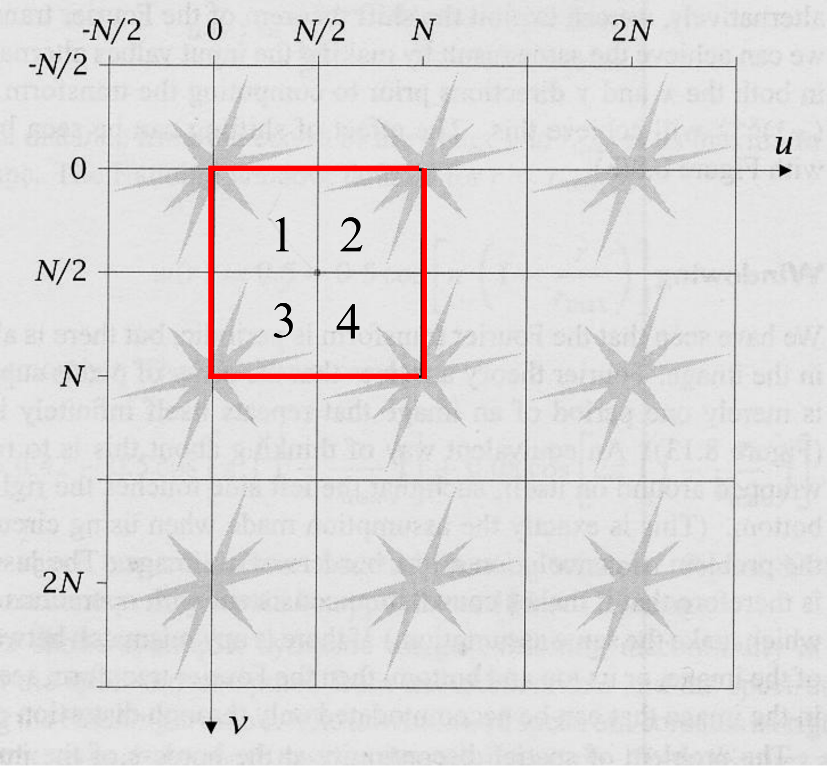
2D Discrete Fourier Transform

Computation



- DFT are periodic on both of horizontal and vertical directions, as shown in the next slide
- So, SHIFT (or swap) the diagonal direction to place low frequencies at the center of spectrum.
- Why shift? Coefficients (or contribution) of low frequencies are usually significant larger than high frequencies'

Fourier Transform property (1)



(*: conjugate)

$$1 = 4^*$$

$$2 = 3^*$$

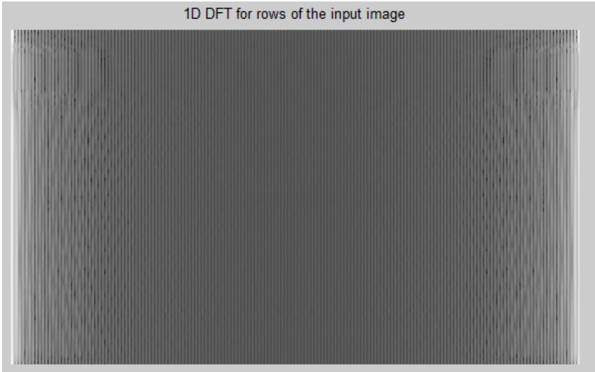
$$F(u,v) = F(-u,-v)^*$$

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

2D Discrete Fourier Transform

Computation

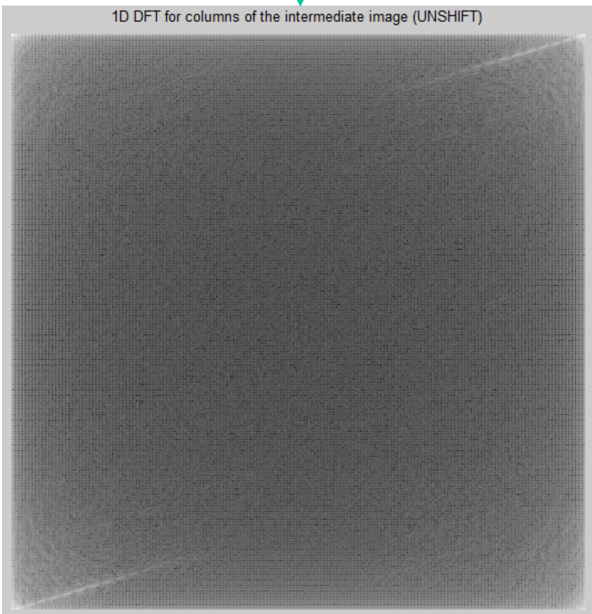
DFT for rows



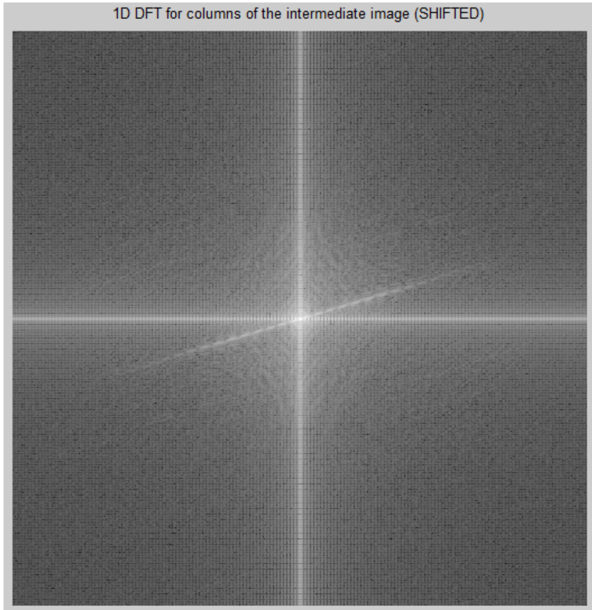
(Enhanced by Gamma, 0.1)

DFT for columns

Enhanced by (1+ Log)



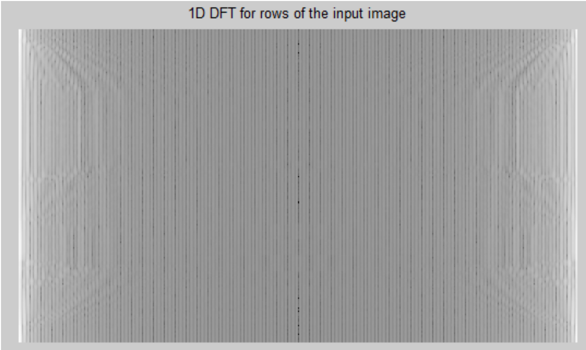
Shift



2D Discrete Fourier Transform

Computation

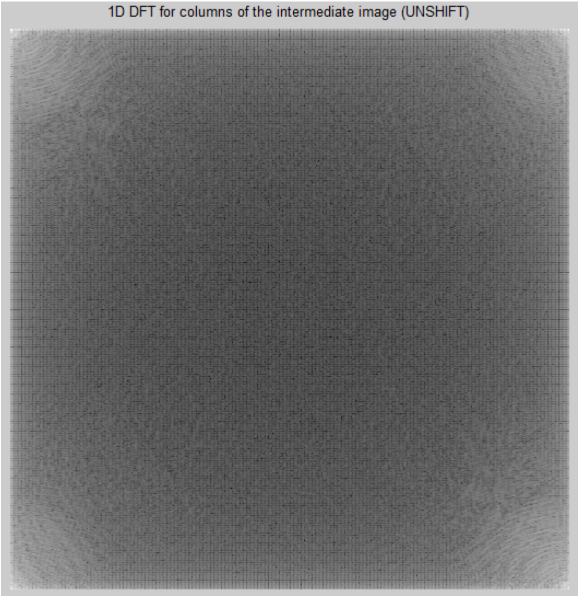
DFT for rows



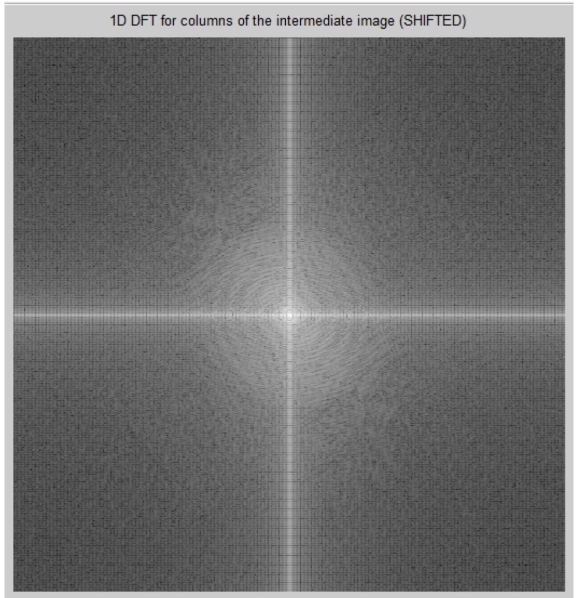
(Enhanced by Gamma, 0.1)

DFT for columns

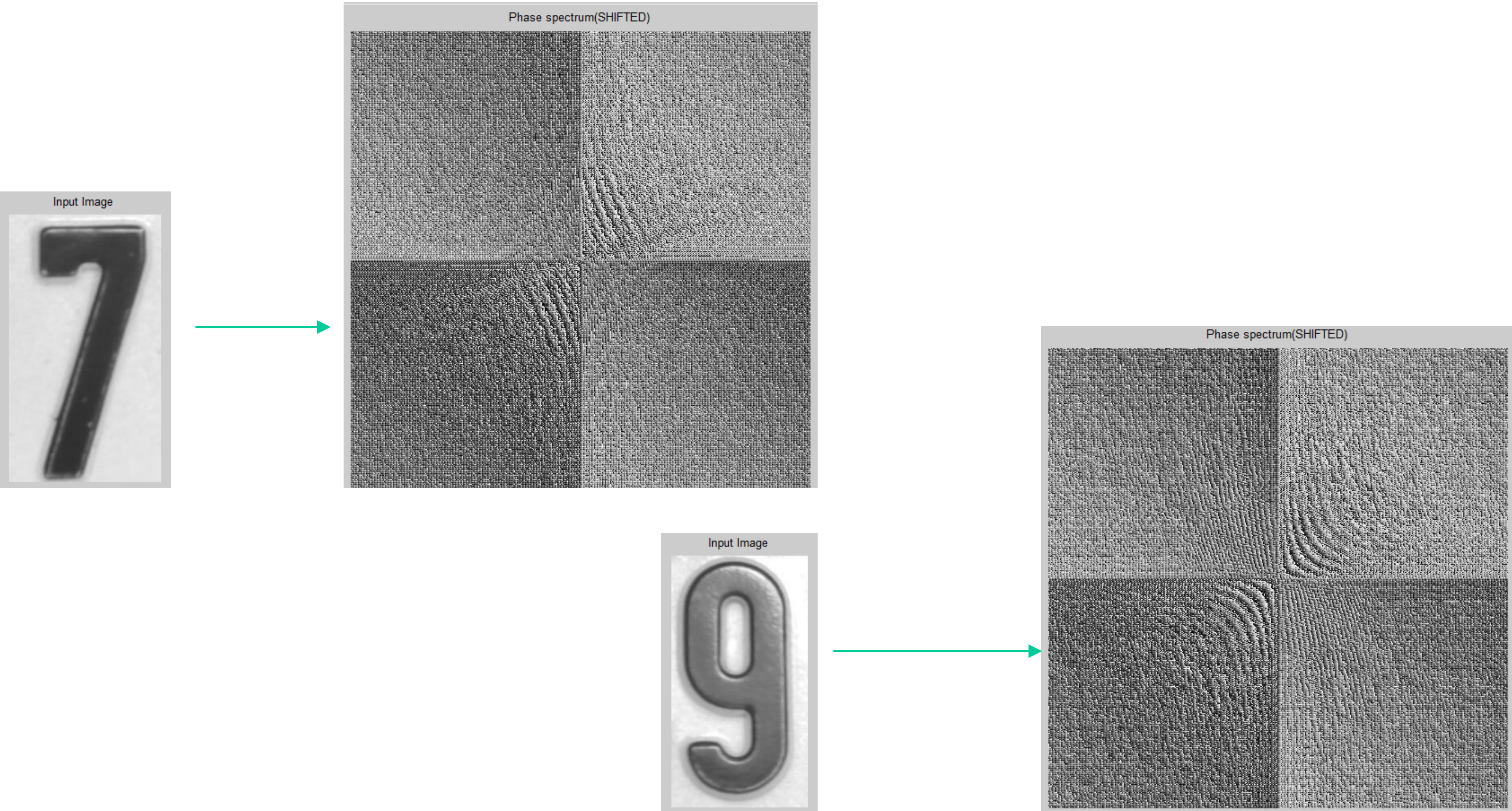
Enhanced by (1+ Log)



Shift



2D Discrete Fourier Transform
Computation



Fast Fourier Transform

Computation

Let $w = e^{-j\frac{2\pi}{M}}$.

If M = 2N (even, actually $M = 2^k$), then:

$$F(u) = \sum_{x=0}^{M-1} f(x) w_M^{ux}$$

$$= \sum_{x=0}^{N-1} f(2x) w_M^{u2x} + \sum_{x=0}^{N-1} f(2x+1) w_M^{u(2x+1)}$$

Fast Fourier Transform

Computation

$$W_M^{u2x} = e^{-j\frac{2\pi}{2N}2ux}$$

$$= e^{-j\frac{2\pi}{N}ux}$$

$$= W_N^{ux}$$

$$W_M^{u(2x+1)} = e^{-j\frac{2\pi}{2N}u(2x+1)}$$

$$= e^{-j\frac{\pi}{N}(2ux+u)}$$

$$= e^{-j\frac{2\pi}{N}ux} e^{-j\frac{2\pi}{2N}u}$$

$$= e^{-j\frac{2\pi}{M}u} e^{-j\frac{2\pi}{N}ux}$$

$$= W_M^u W_N^{ux}$$

Fast Fourier Transform

Computation

$$F(u) = \sum_{x=0}^{M-1} f(x) w_M^{ux}$$

$$= \sum_{x=0}^{N-1} f(2x) w_M^{u2x} + \sum_{x=0}^{N-1} f(2x+1) w_M^{u(2x+1)}$$

$$= \underbrace{\sum_{x=0}^{N-1} f(2x) w_N^{ux}}_{F_{even}(u)} + w_M^{u} \underbrace{\sum_{x=0}^{N-1} f(2x+1) w_N^{ux}}_{F_{odd}(u)}$$

For $F(u)$: **u = 0..M-1 = 0..2N-1**

For $F_{even}(u)$: **u = 0..N-1**

For $F_{odd}(u)$: **u = 0..N-1**

Fast Fourier Transform

Computation

$$
\begin{aligned}
w_M^{u+N} &= e^{-j\frac{2\pi}{2N}(u+N)} \\
&= e^{-j\frac{\pi}{N}(u+N)} \\
&= e^{-j\pi} e^{-j\frac{2\pi}{2N}u} \\
&= -w_M^{u}
\end{aligned}
$$

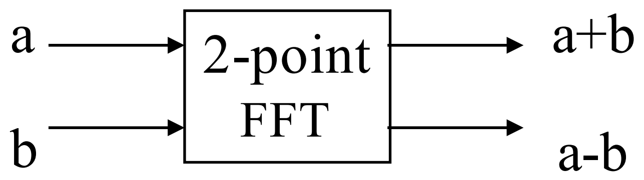
Fast Fourier Transform

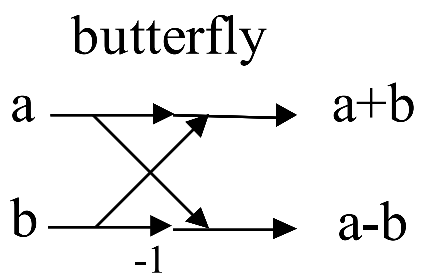
Computation

$$F(u) = \begin{cases} F_{even}(u) + w_M^u F_{odd}(u) & | u = 0..(N-1) \\ F_{even}(u) - w_M^u F_{odd}(u) & | u = N..(2M-1) \end{cases}$$

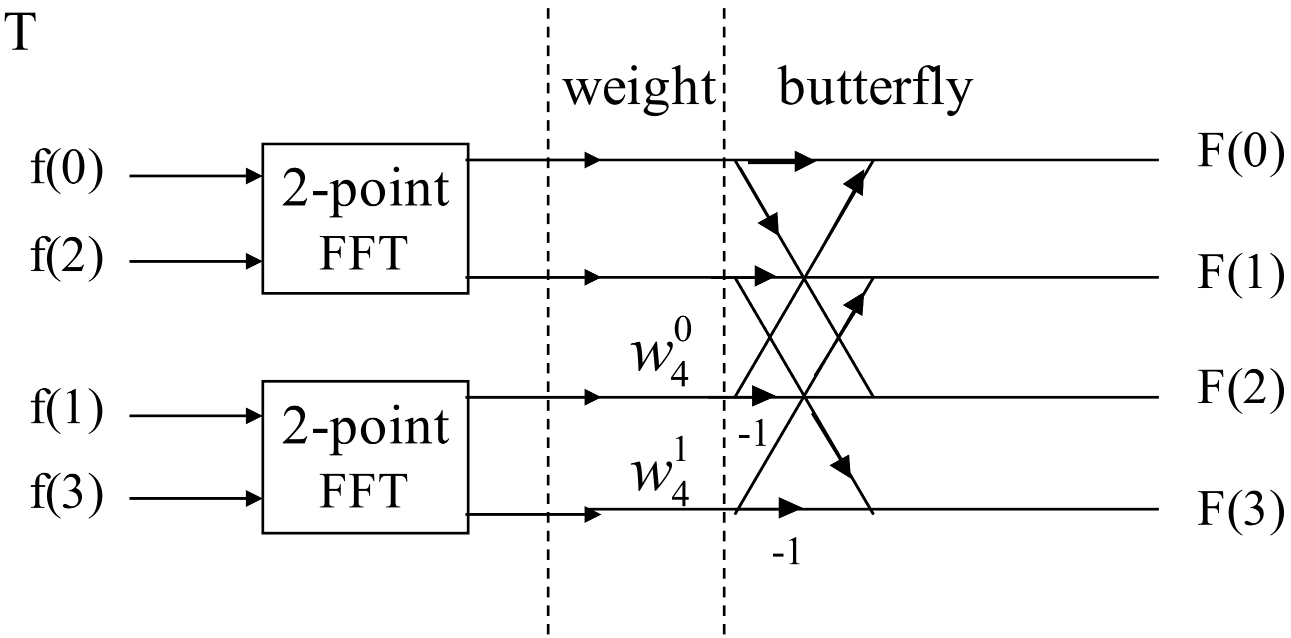
DIT-FFT (Scalable) (1)

2-point FFT



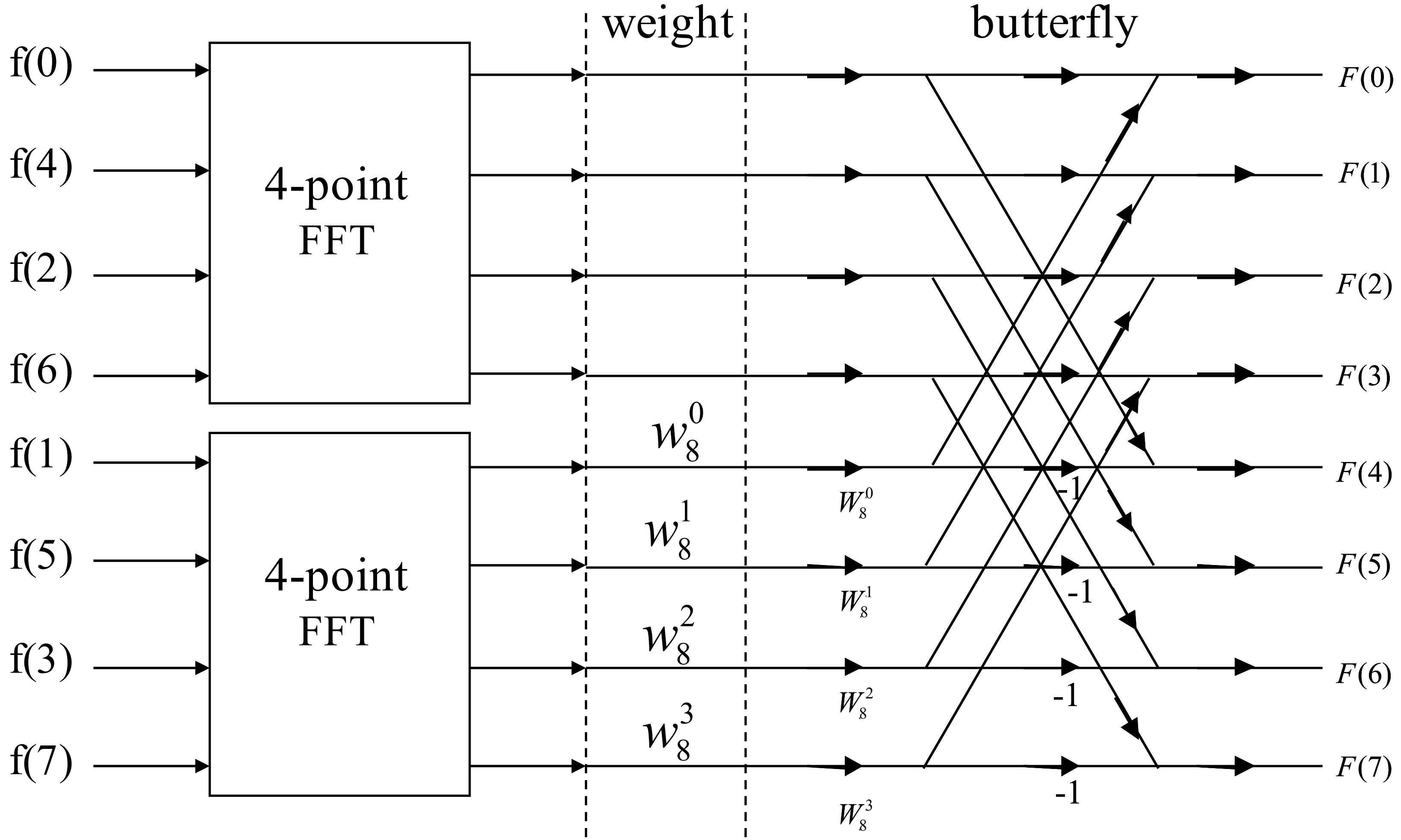


4-point FFT



DIT-FFT (Scalable) (2)

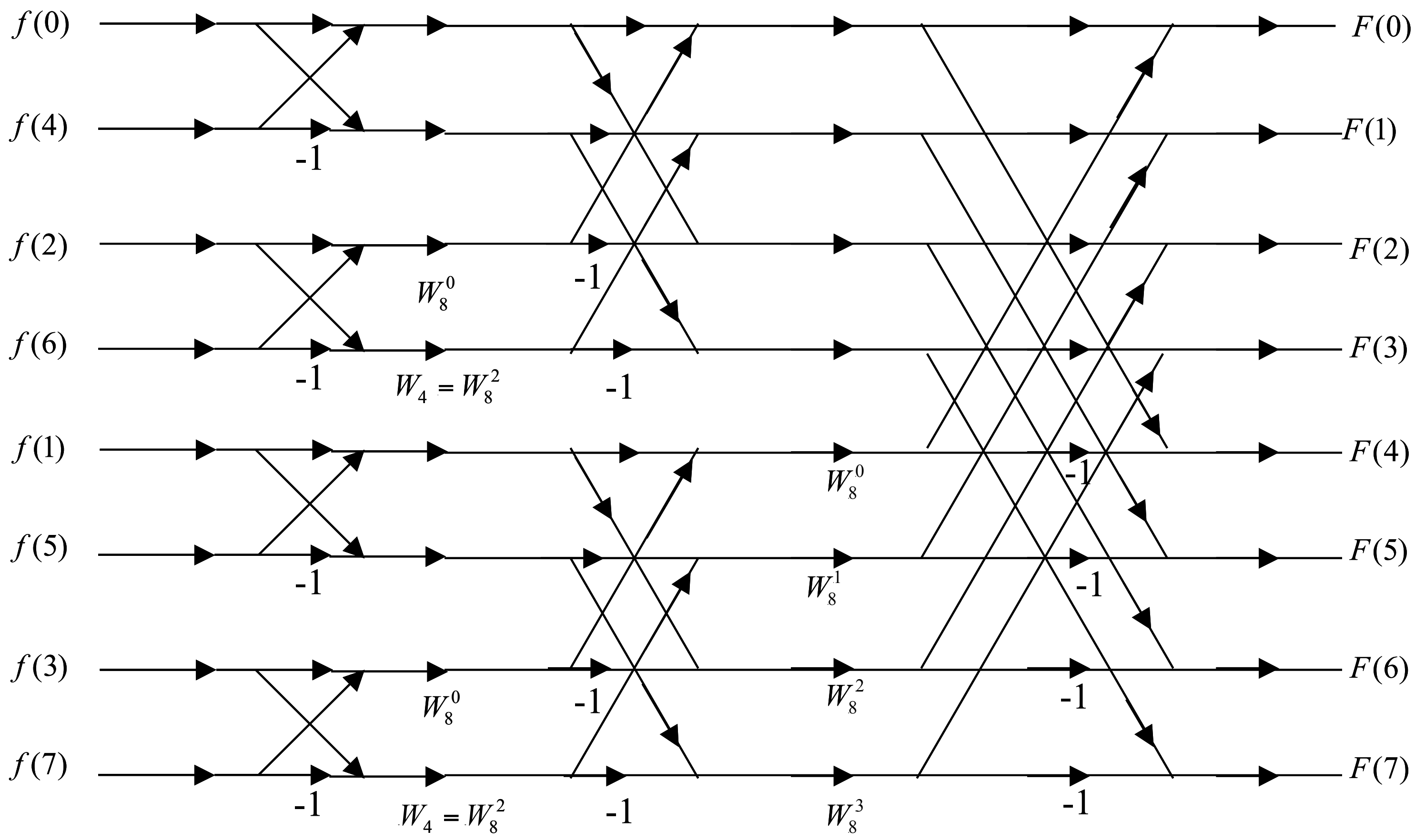
8-point FFT



Fast Fourier Transform (DIT-FFT)

Direct calculation = N^2 **FFT = $N\log_2 N$**

8-point FFT



Fast Fourier Transform

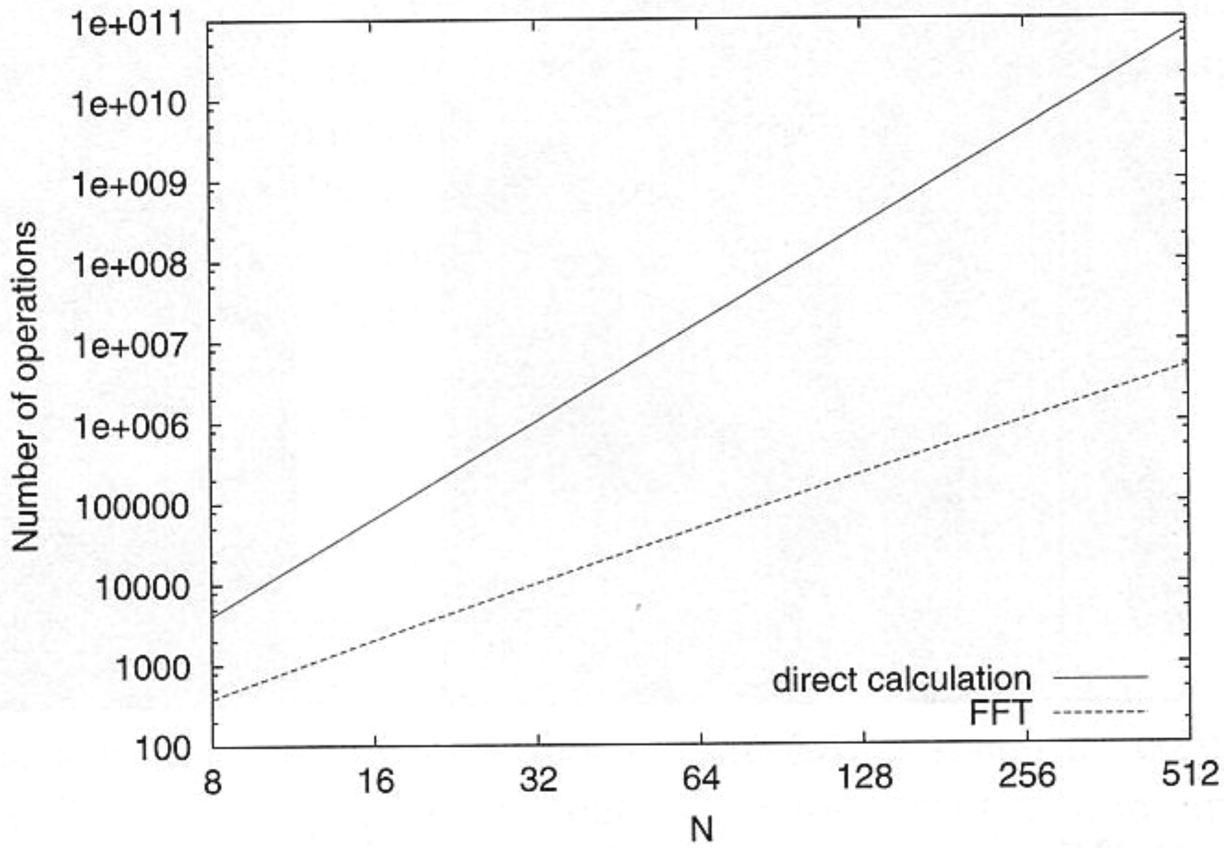
Computational Complexity:

Discrete Fourier Transform → $O(N^2)$

Fast Fourier Transform → $O(N \log N)$

Remember: The Fast Fourier Transform is just a faster **algorithm** for computing the Discrete Fourier Transform — it does *not* produce a different result.

Processing Time (DFT vs FFT)



Relationship between Convolution and DFT

$$f(x,y)*h(x,y) \Leftrightarrow F(u,v)H(u,v)$$

$$g(x,y) = f(x,y)*h(x,y) = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)h(x-m,y-n)$$

$$DFT[g(x,y)] = G(u,v) = \sum_{y=0}^{M-1}\sum_{x=0}^{N-1} g(x,y)W_M^{ux}W_N^{vy} \quad ; \qquad W_M^{ux} = e^{-j\frac{2\pi ux}{M}} \; ; \; W_M^{vy} = e^{-j\frac{2\pi vy}{N}}$$

$$= \sum_{y=0}^{M-1}\sum_{x=0}^{N-1}\left[\frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)h(x-m,y-n)\right]W_M^{ux}W_N^{vy}$$

$$= \frac{1}{MN}\sum_{y=0}^{M-1}\sum_{x=0}^{N-1}\left[\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)h(x-m,y-n)\right]W_M^{mu}W_N^{nv}W_M^{(x-m)u}W_N^{(y-n)v}$$

$$= \frac{1}{MN}\left[\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)W_M^{mu}W_N^{nv}\right]\sum_{y=0}^{M-1}\sum_{x=0}^{N-1} h(x-m,y-n)W_M^{(x-m)u}W_N^{(y-n)v}$$

$$= \frac{1}{MN}F(u,v)H(u,v)$$

$$IDFT\left[\frac{1}{MN}F(u,v)H(u,v)\right] = g(x,y) = f(x,y)*h(x,y)$$

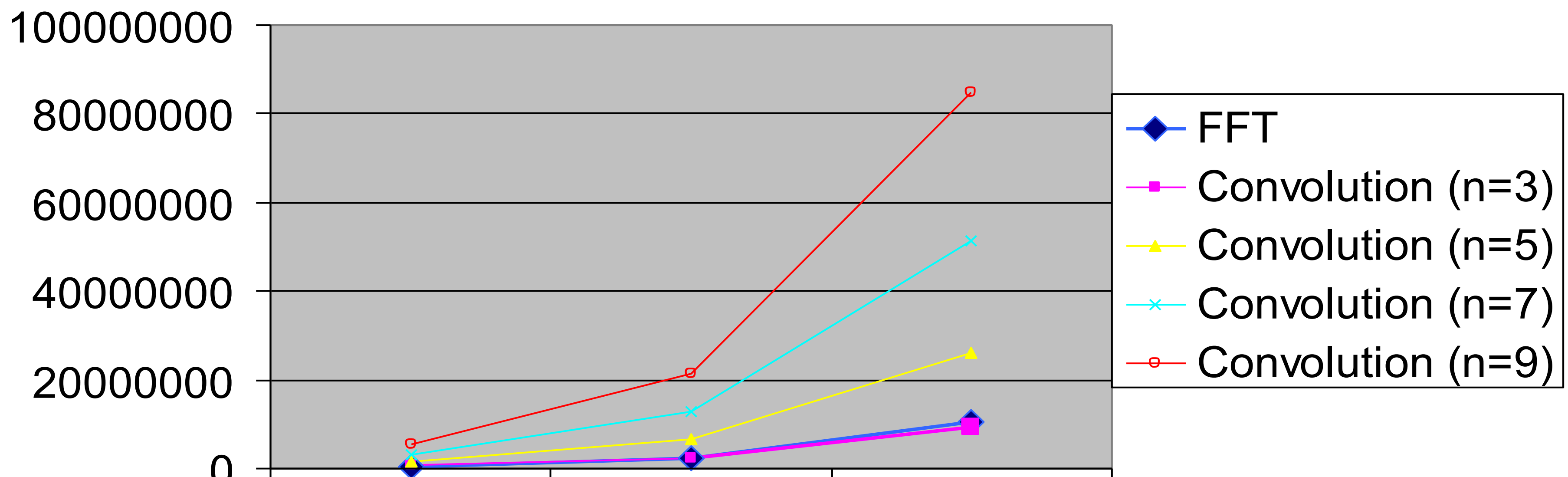
Comparison of Convolution and FFT

	Convolution	FFT
Computational Process	Simple	Complex with many steps
Number of Multiplications	$N^2 n^2$	$< N^2(\log_2 N)$

N x N -> image size

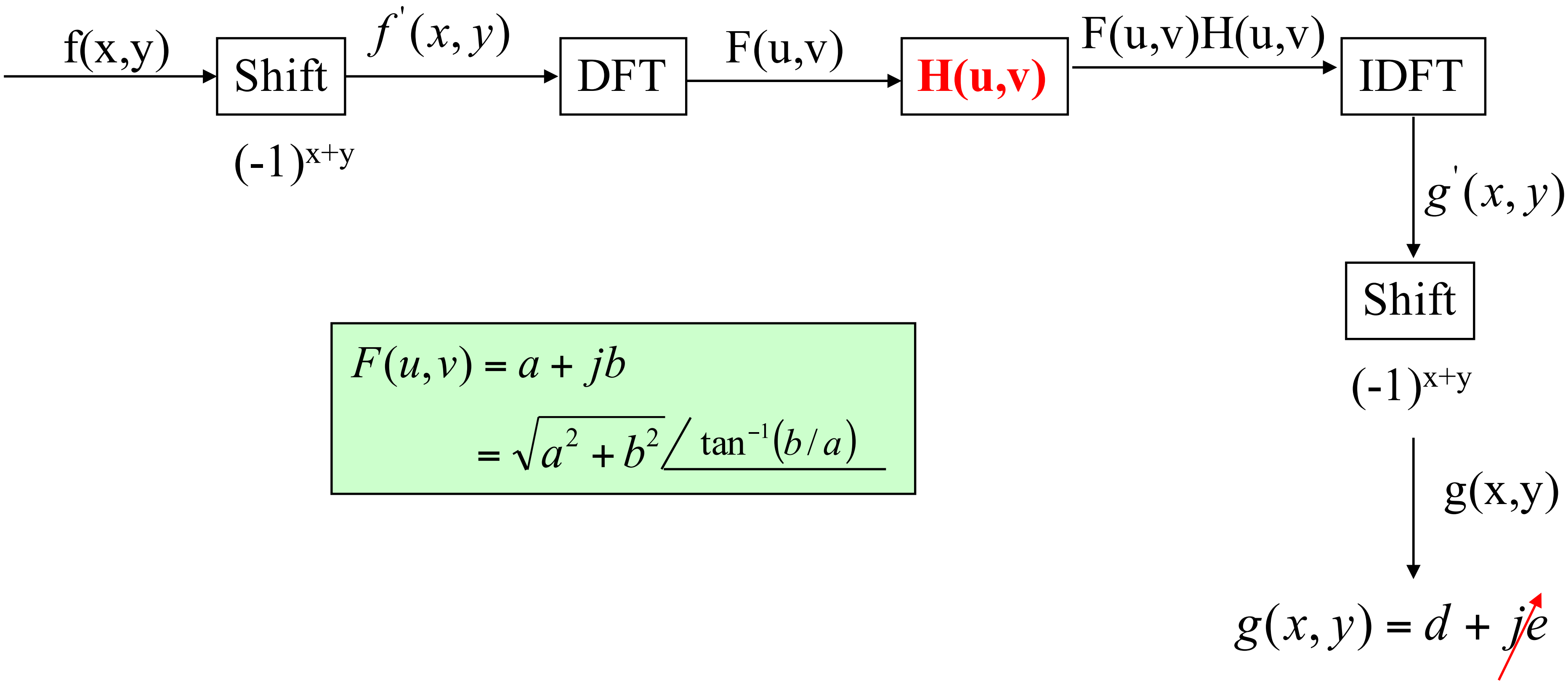
n x n -> filter window size

Number of Multiplication (FFT vs Convolution)

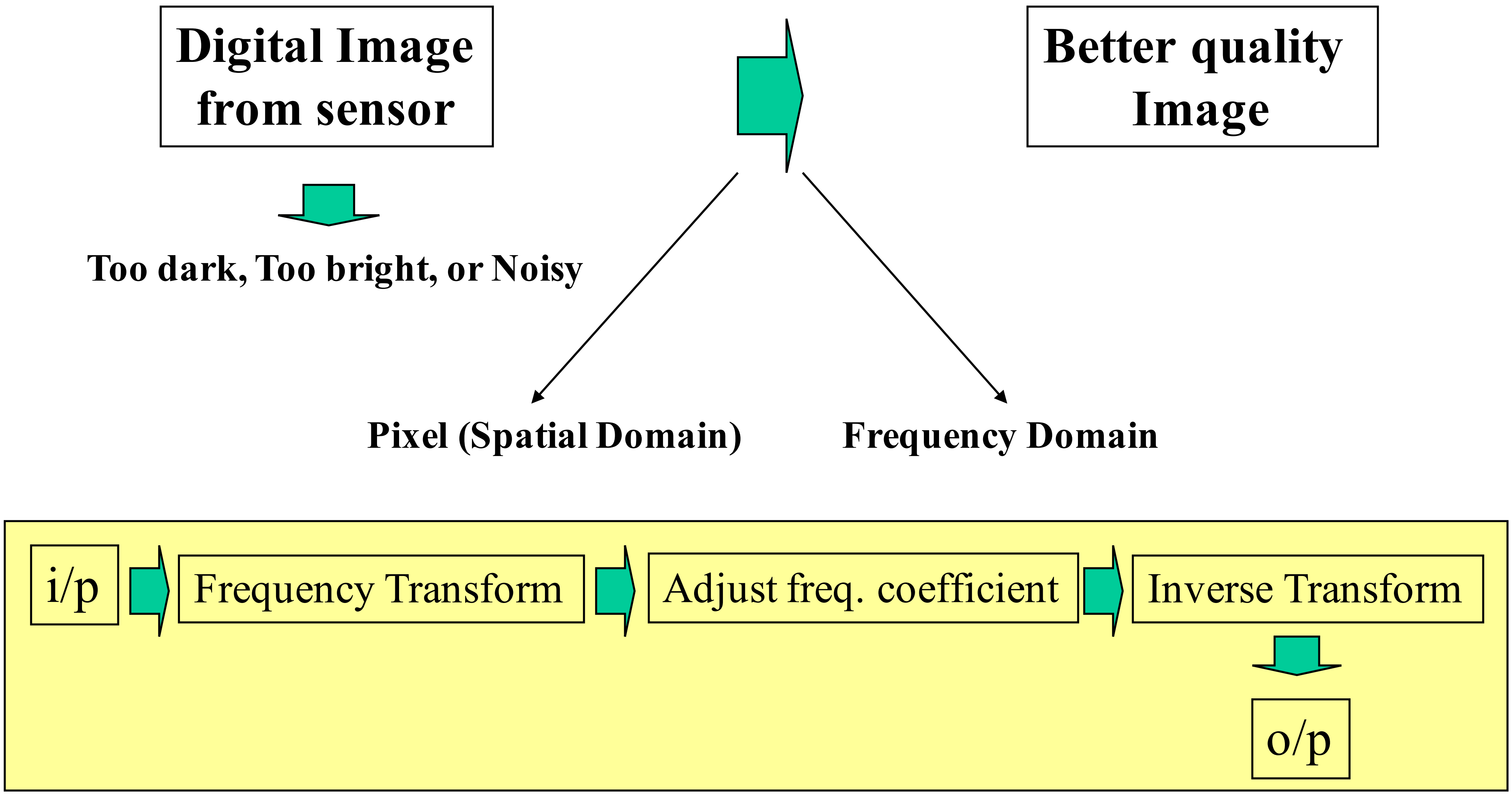


	256	512	1024
FFT	524288	2359296	10485760
Convolution (n=3)	589824	2359296	9437184
Convolution (n=5)	1638400	6553600	26214400
Convolution	3211264	12845056	51380224

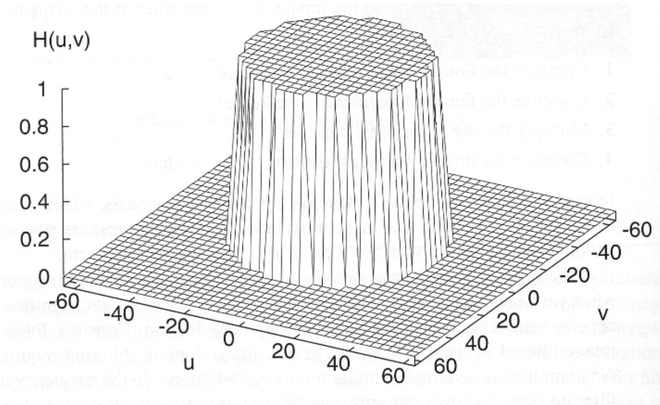
Filtering in Frequency Domain

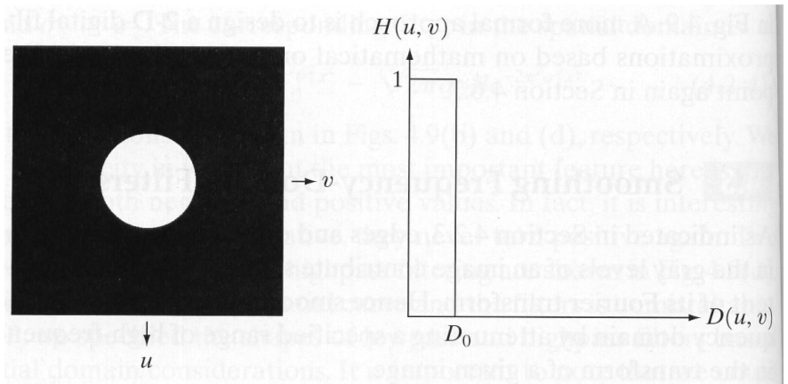


Fourier Transform Applications: Image Enhancement and Restoration



Ideal Low Pass Filter (ILPF)

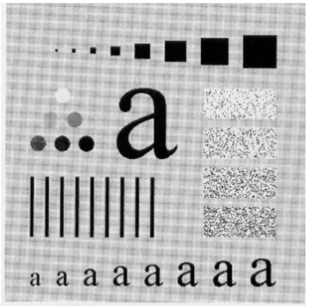




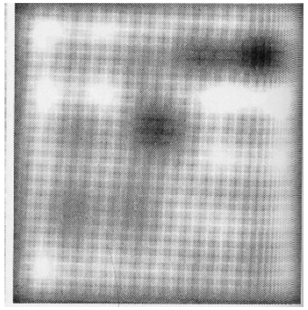
$$H(u,v)=\begin{cases}1 & r(u,v)\le r_0\\ 0 & r(u,v)>r_0\end{cases}$$

$$r(u,v)=\sqrt{(u-M/2)^2+(v-N/2)^2}$$

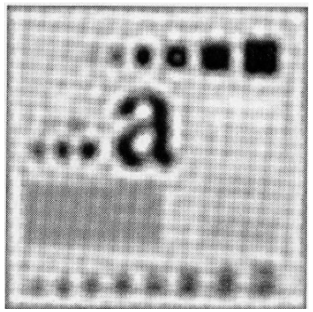
ILPF results



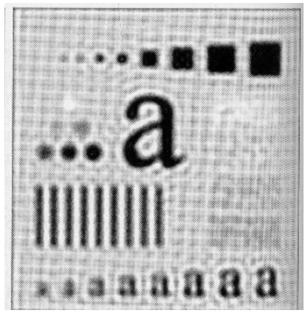
Original image



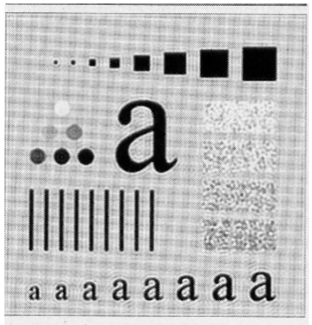
$r_0 = 5$



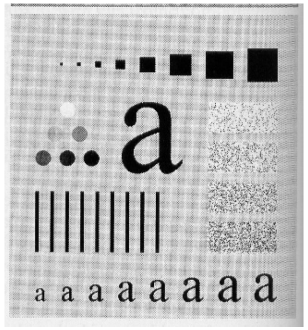
$r_0 = 15$



$r_0 = 30$

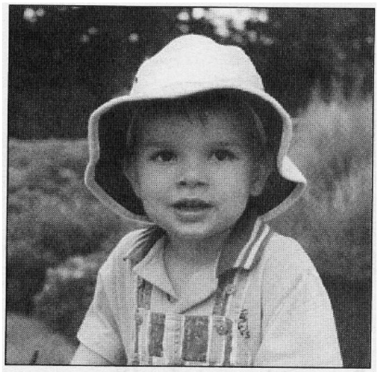


$r_0 = 80$

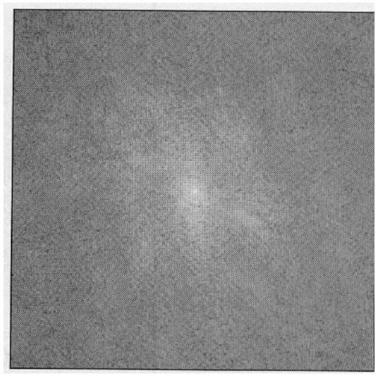


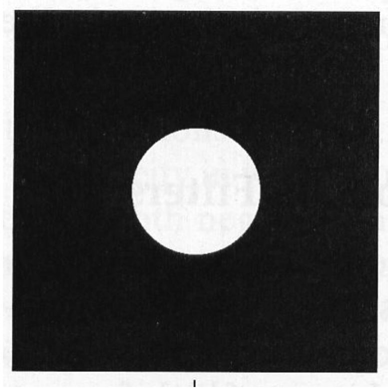
$r_0 = 230$

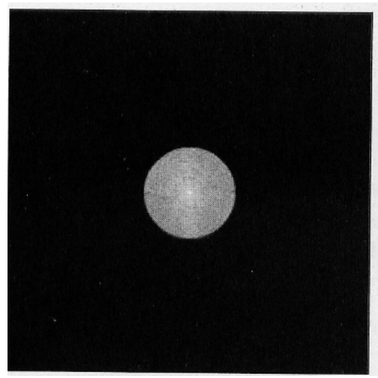
ILPF ripple effects



DFT







Original image [f(x,y)]

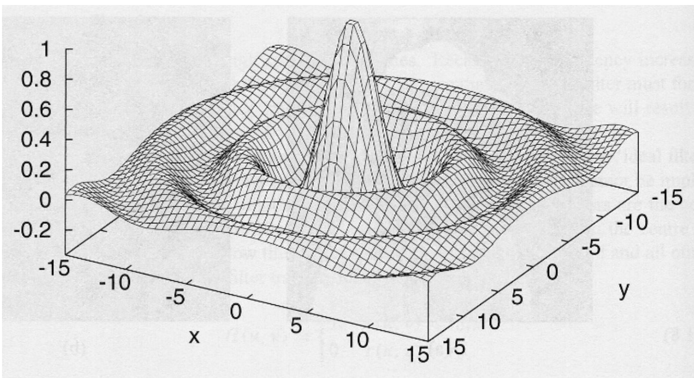
F(u,v)

H(u,v)

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

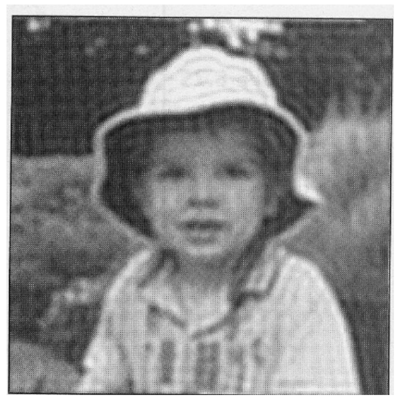
$f * h$

Inverse DFT



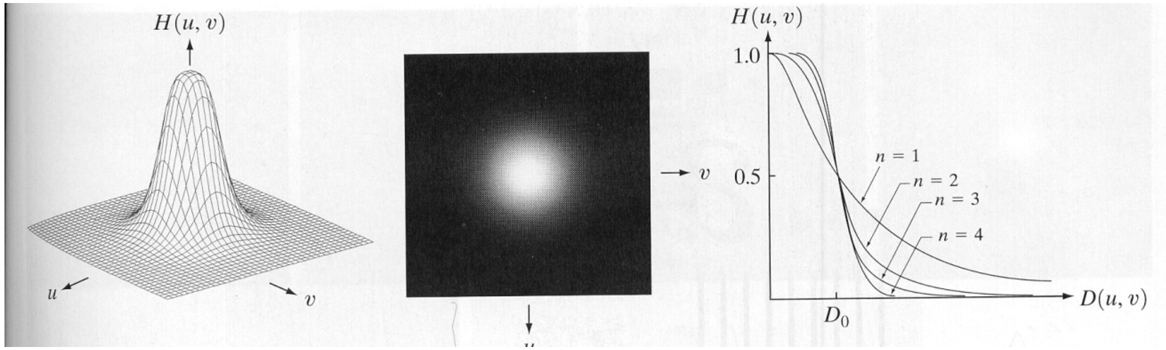
h(x,y)

$$f * h = inverse\ DFT[FH]$$



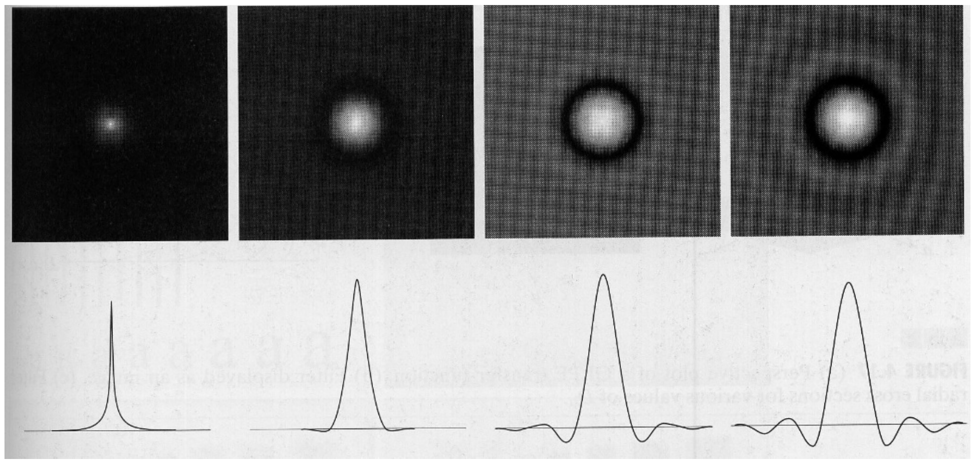
g(x,y)

Butterworth Low Pass Filter (BLPF)



$$H(u,v) = \frac{1}{1 + [r(u,v)/r_0]^{2n}}$$

$$r(u,v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$

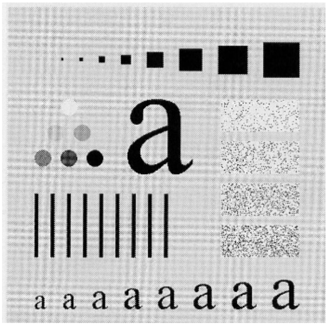


h(x,y)

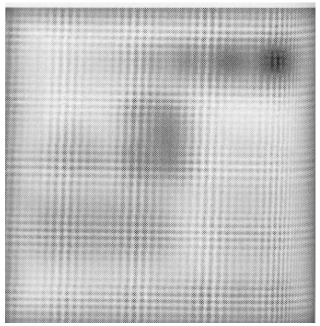
n = 1 n = 2 n = 3 n = 4

BLPF results

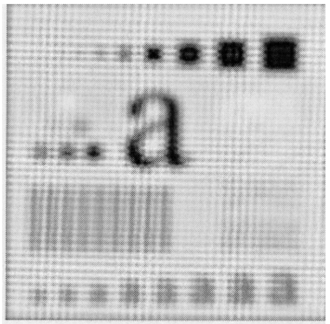
n = 2



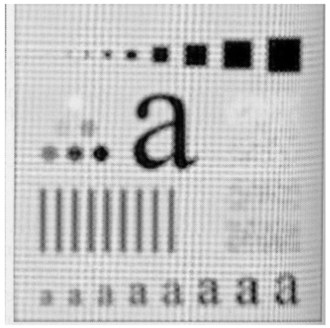
Original image



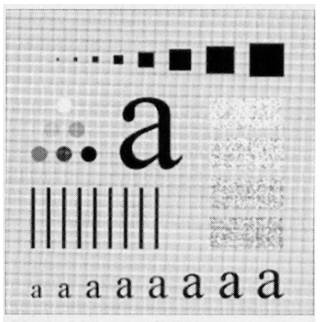
$r_0 = 5$



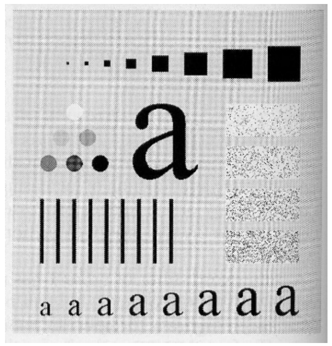
$r_0 = 15$



$r_0 = 30$

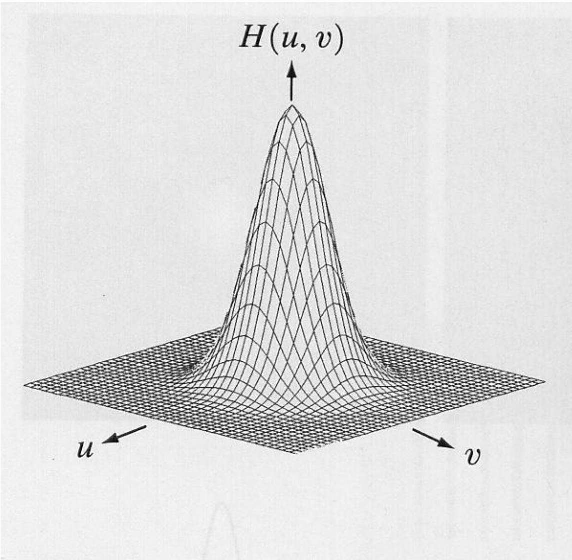


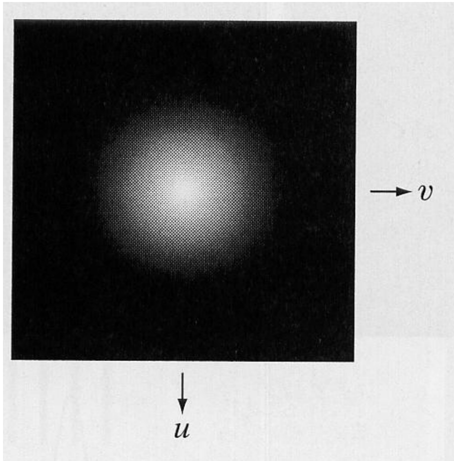
$r_0 = 80$

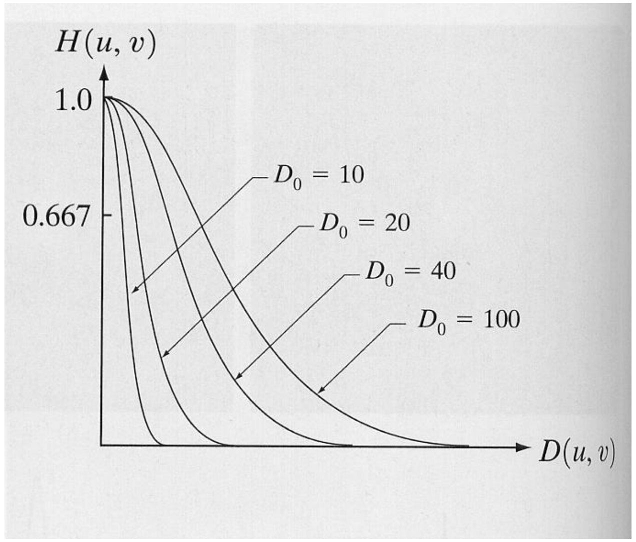


$r_0 = 230$

Gaussian Low Pass Filter (GLPF)







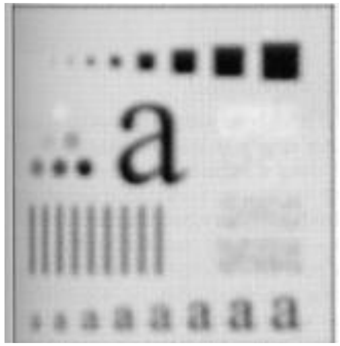
$$H(u,v) = e^{-r^2(u,v)/2r_0^2}$$

$$r(u,v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$

GLPF results

Original image

$r_0 = 5$



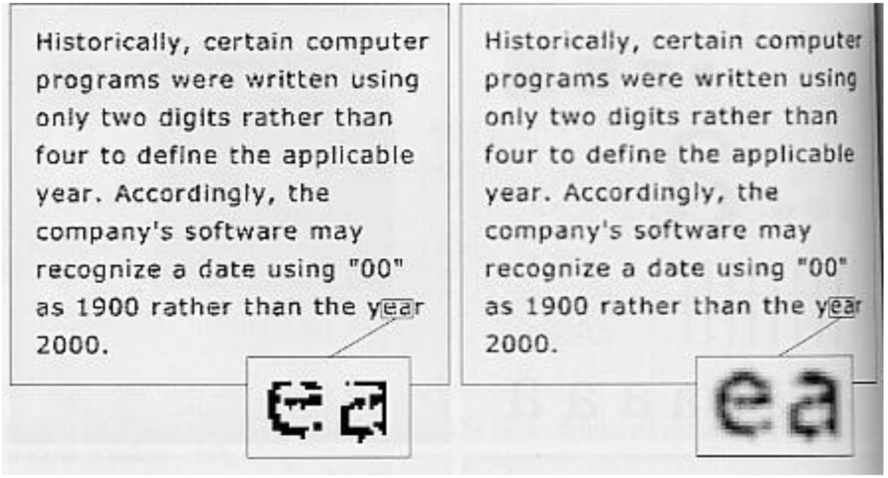
$r_0 = 15$

$r_0 = 30$

$r_0 = 80$

$r_0 = 230$

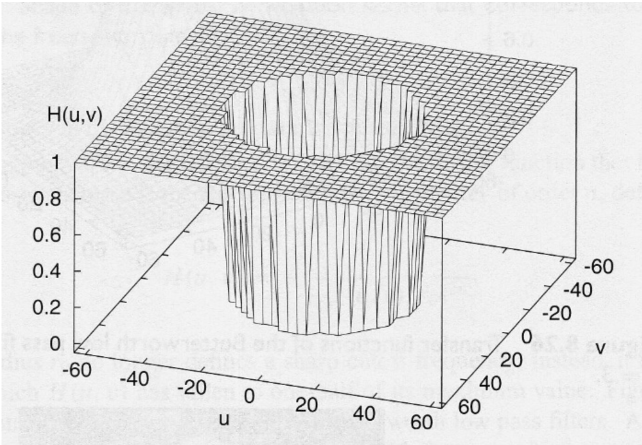
Applications of Low Pass Filter



Character recognition

Picture Studio Decoration

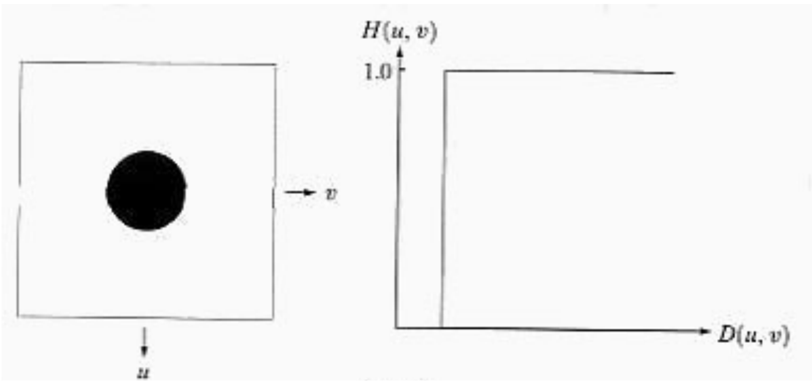
Ideal High Pass Filter (IHPF)



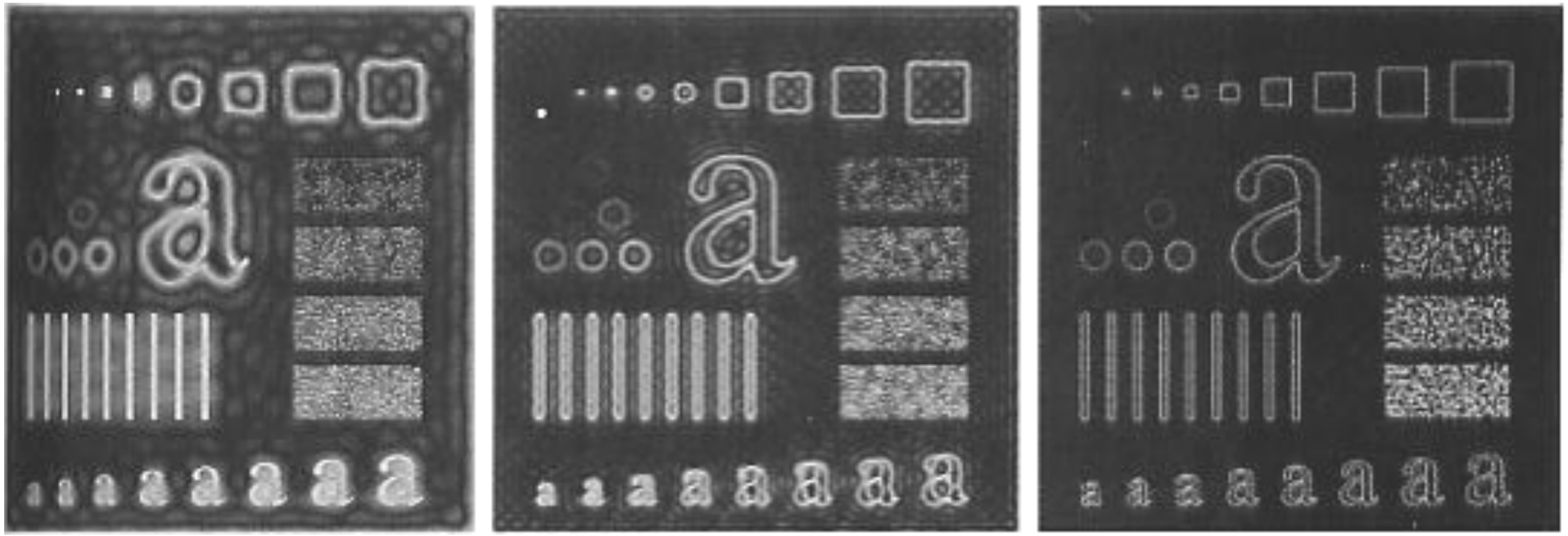
$H(u,v) =$

$$H(u,v) = \begin{cases} 0 & r(u,v) \le r_0 \\ 1 & r(u,v) > r_0 \end{cases}$$

$$r(u,v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$

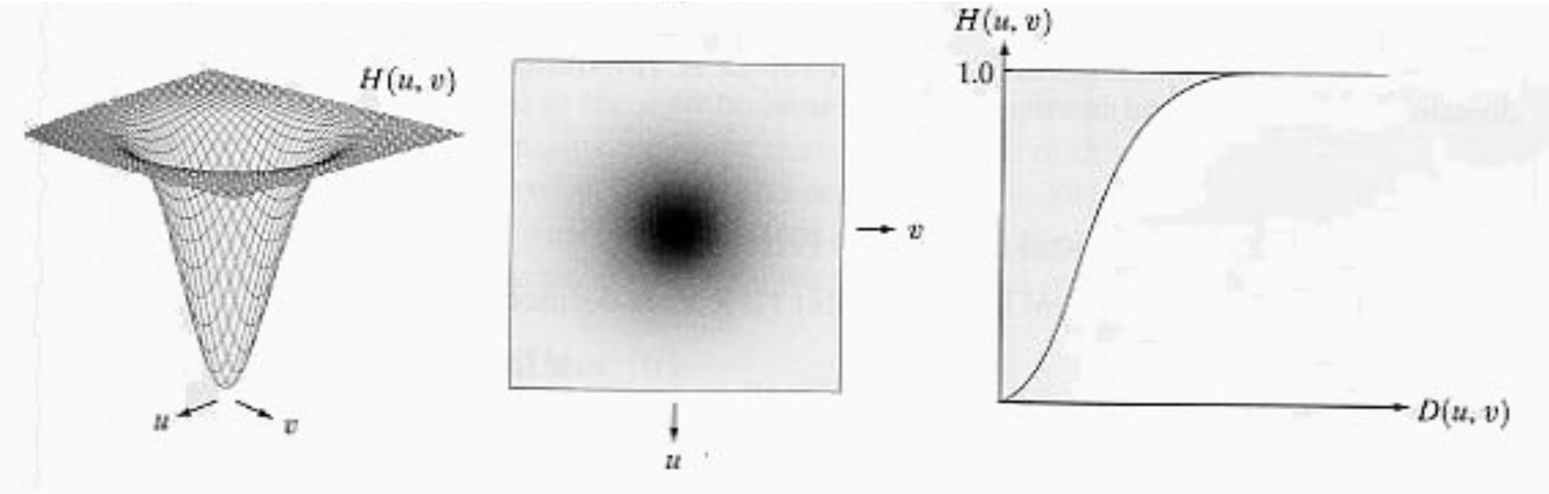


IHPF results



$r_0 = 15$ $r_0 = 30$ $r_0 = 80$

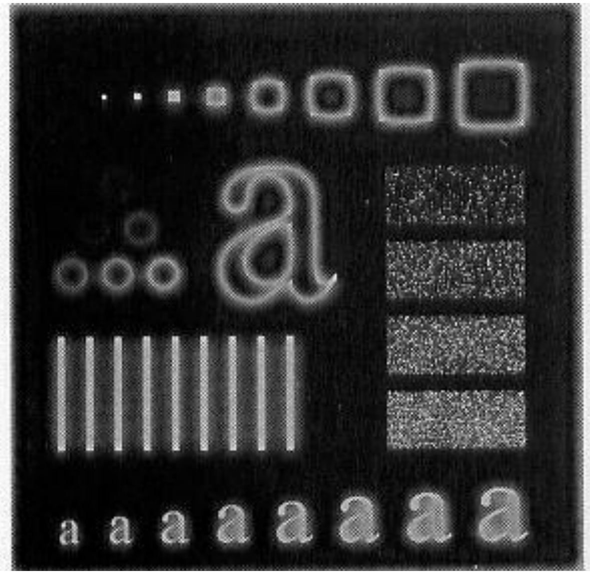
Butterworth High Pass Filter (BHPF)



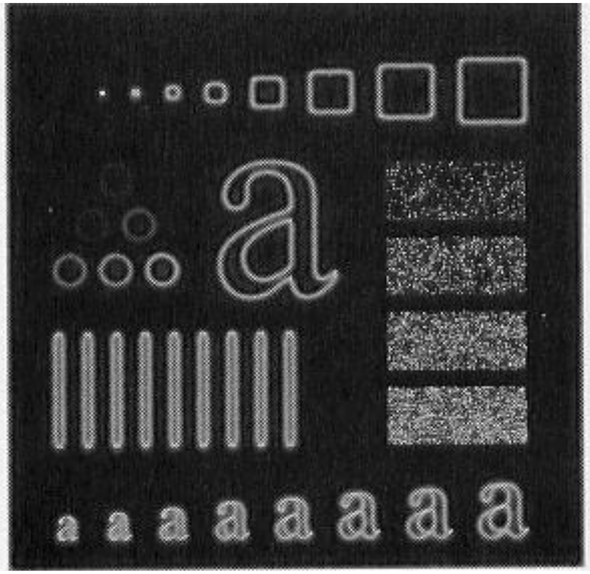
$$H(u,v) = \frac{1}{1 + [r_0 / r(u,v)]^{2n}}$$

$$r(u,v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$

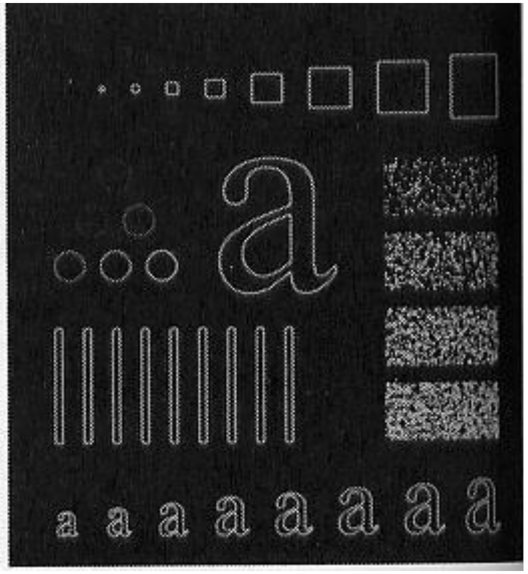
BHPF results



$r_0 = 15$

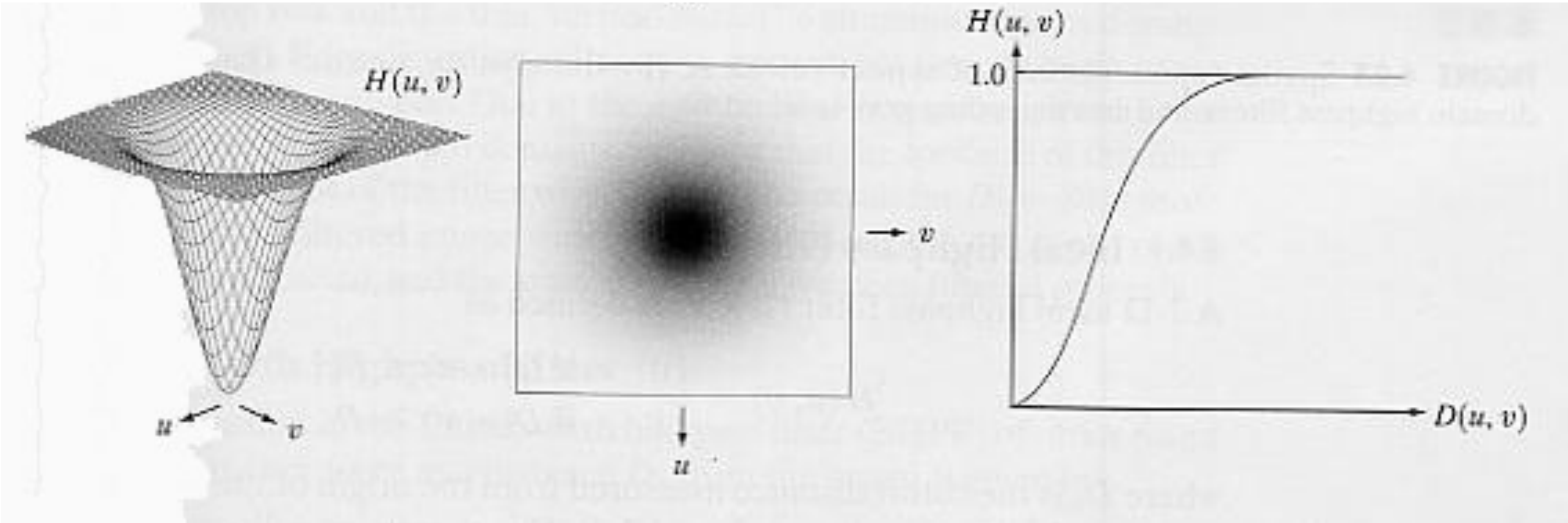


$r_0 = 30$



$r_0 = 80$

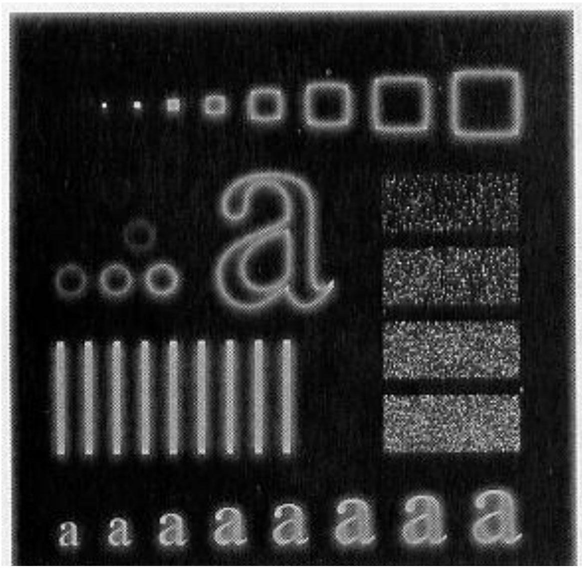
Gaussian High Pass Filter (GHPF)



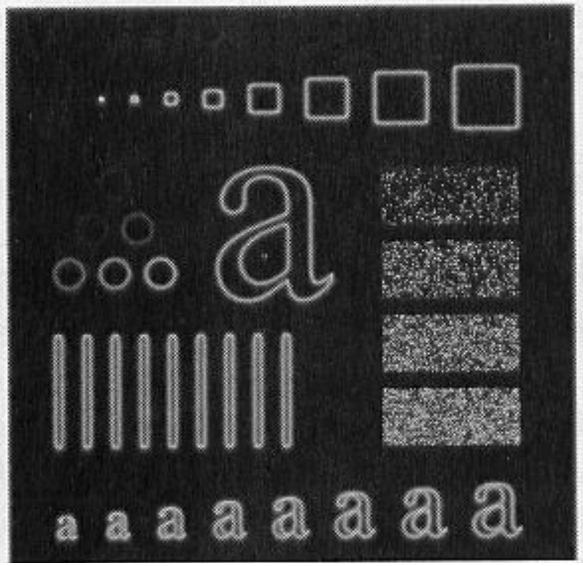
$$H(u,v) = 1 - e^{-r^2(u,v)/2r_0^2}$$

$$r(u,v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$

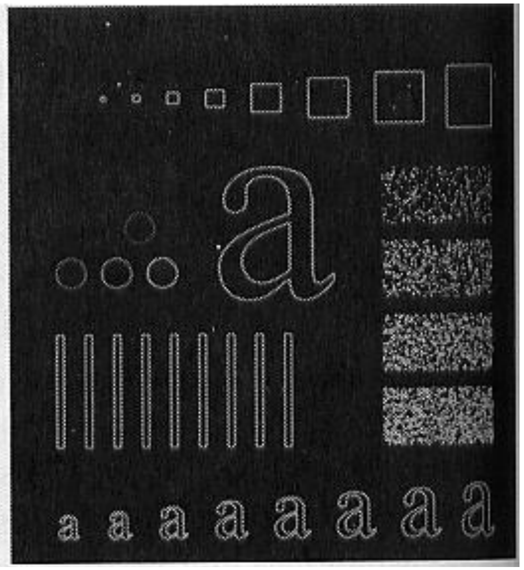
GHPF results



$r_0 = 15$

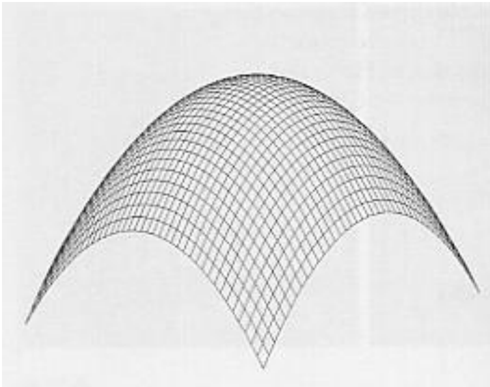


$r_0 = 30$

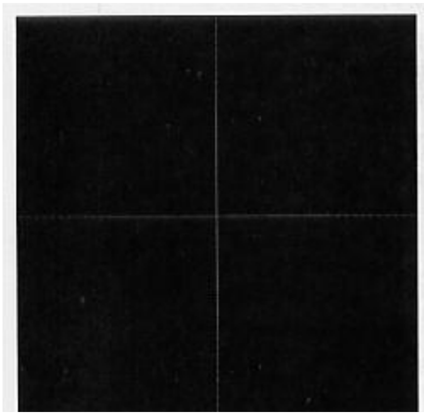


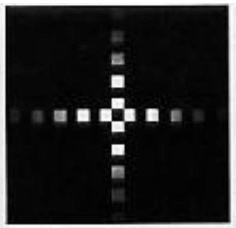
$r_0 = 80$

Laplacian Filter (Second-order Filter)

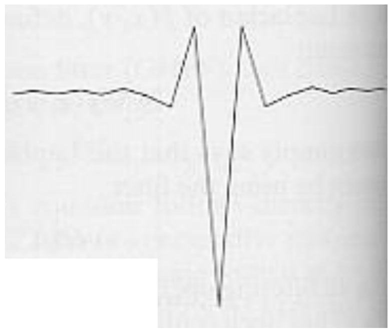


$$H(u,v) = -[(u - M/2)^2 + (v - N/2)^2]$$





h(x,y)



Laplacian Filter results

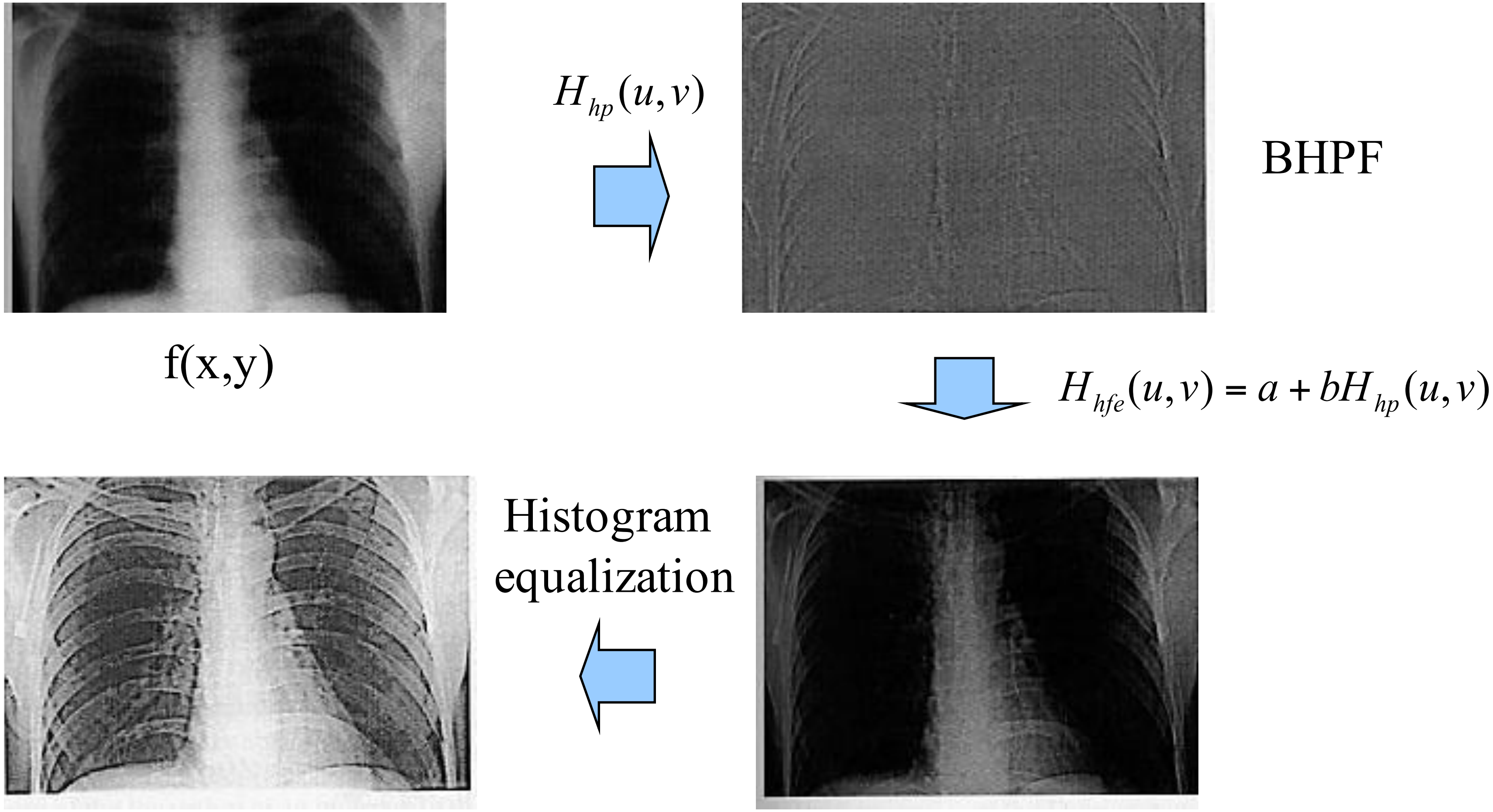
Original image

Laplacian filter

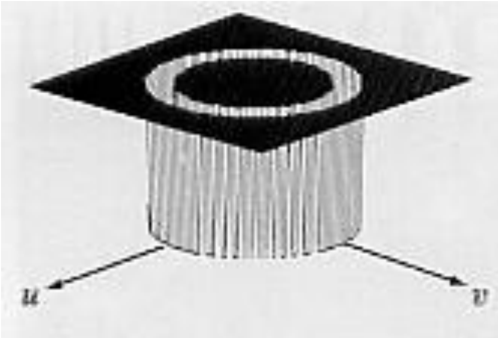
Laplacian filtered image

$$g(x,y) = f(x,y) - \nabla^2 f(x,y)$$

High Frequency Emphasis Filter

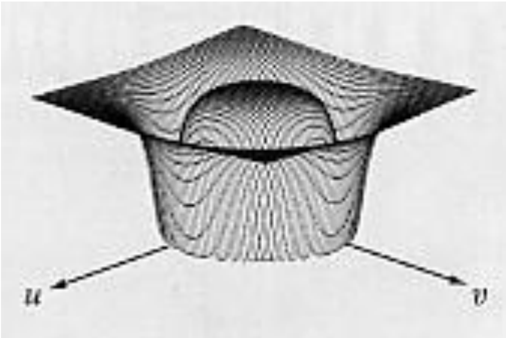


Band Reject Filter (BRF)



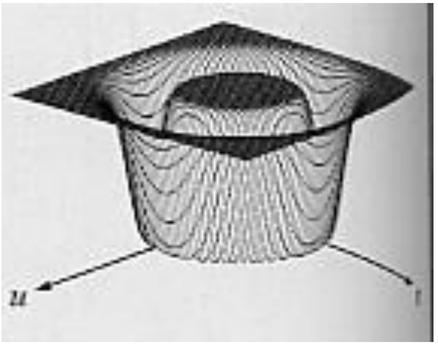
$$H(u,v)=\begin{cases}1 & ; \quad r(u,v) < r_0 - \frac{BW}{2} \\ 0 & ; \quad r_0 - \frac{BW}{2} \le r(u,v) \le r_0 + \frac{BW}{2} \\ 1 & ; \quad r(u,v) > r_0 + \frac{BW}{2}\end{cases}$$

Ideal BRF



Butterworth BRF

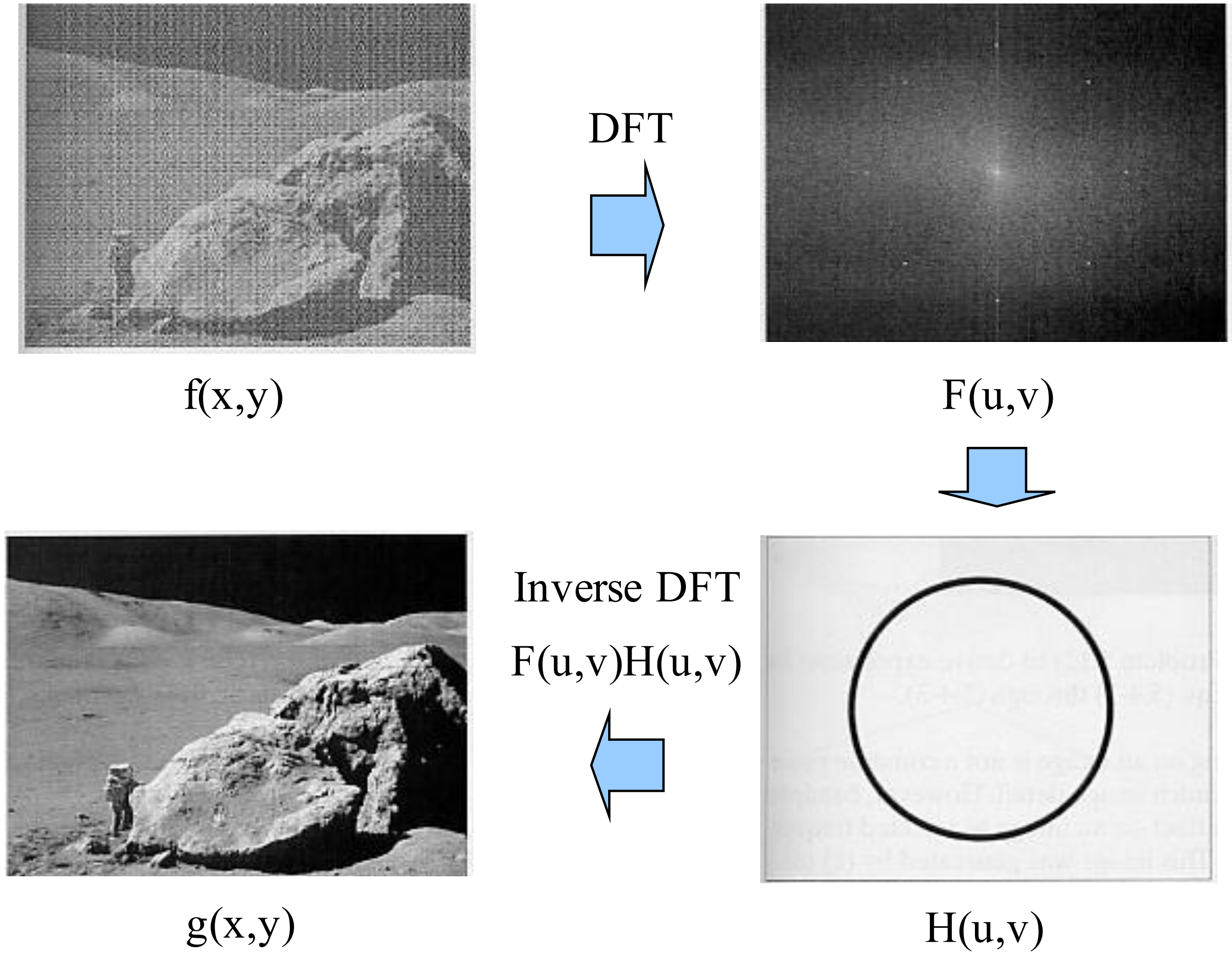
$$H(u,v)=\frac{1}{1+\left[\frac{r(u,v).BW}{r^2(u,v)-r_0^2}\right]}$$



Gaussian BRF

$$H(u,v)=1-e^{-\frac{1}{2}\left[\frac{r^2(u,v)-r_0^2}{r(u,v).BW}\right]}$$

BRF results



Band Pass Filter (BPF)

$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$