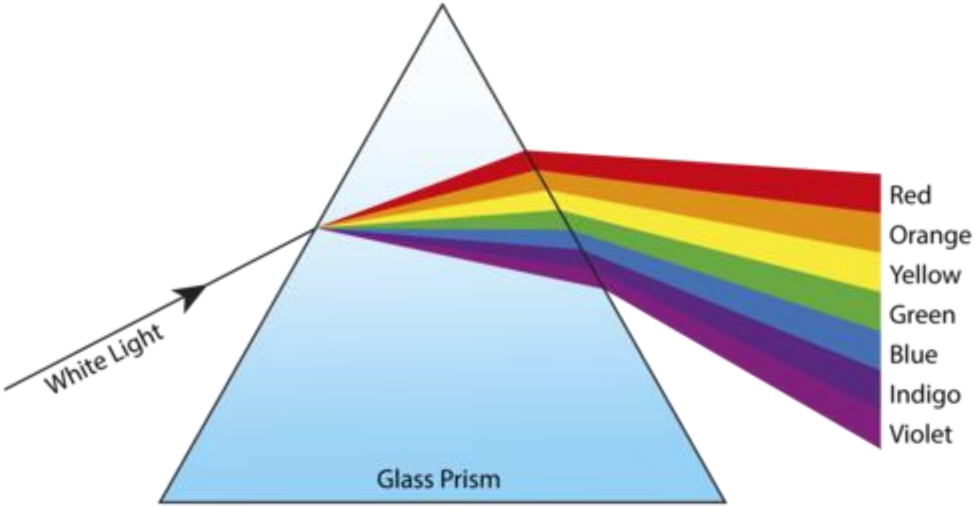
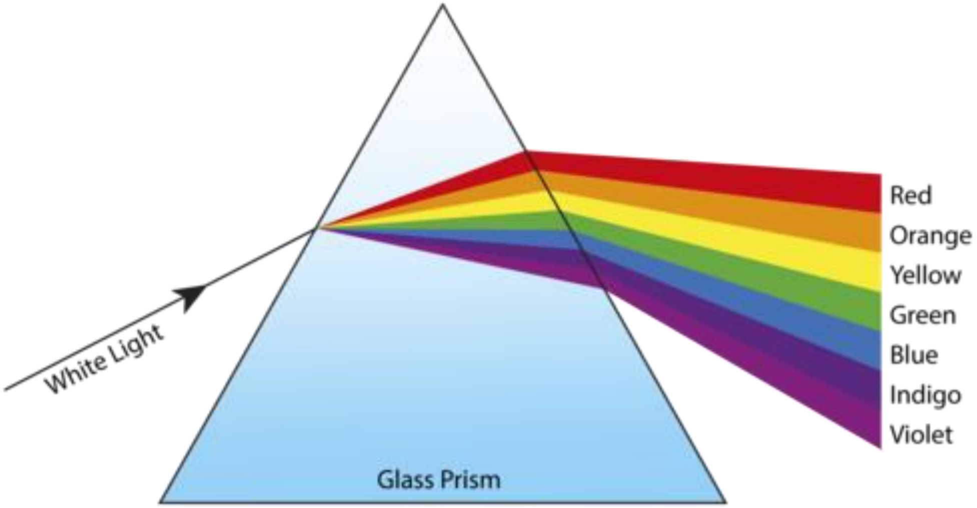
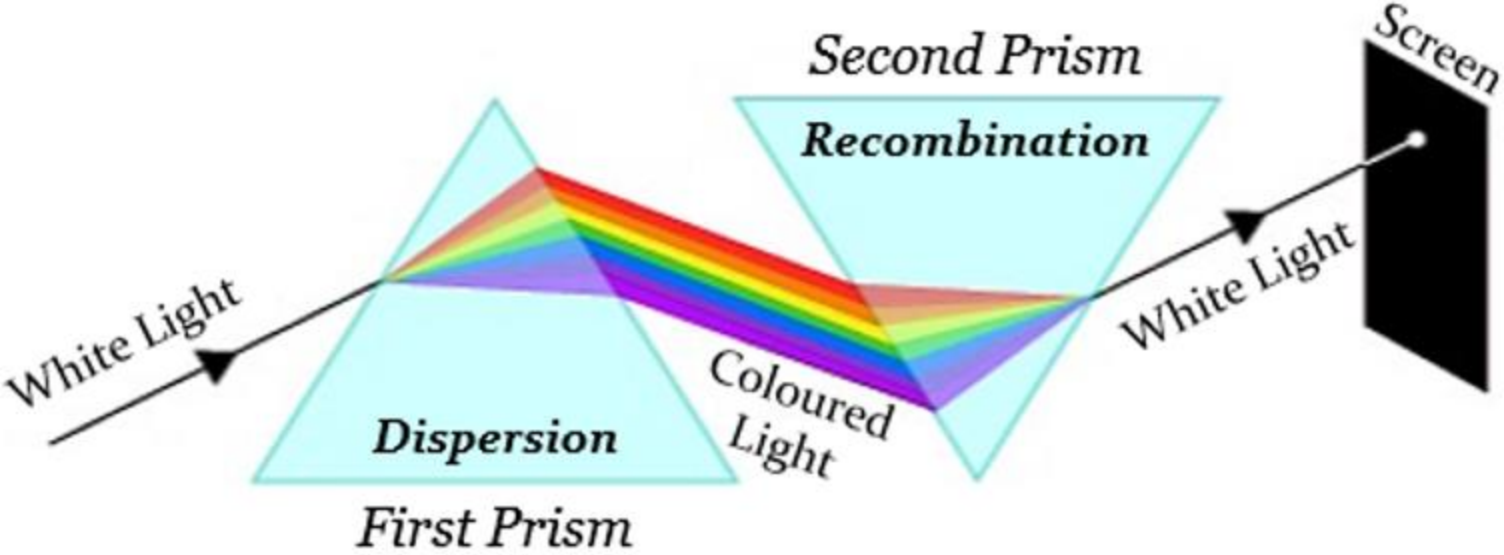
Image Enhancement in Frequency Domain

- All about working in the Fourier domain
- By end of this topic, you will understand the relationship between the spatial and frequency domain



- The analysis of light into color is actually a form of frequency analysis
- The prism splits light into separate colors (wavelengths)
- What prism does to light, the Fourier transform does to signals


White Light
Red
Orange
Yellow
Green
Blue
Indigo
Violet
Glass Prism


Second Prism
Screen
Recombination
White Light
White Light
Coloured Light
Dispersion
First Prism

1D Fourier Transform

- is a mathematical technique that decomposes a function (typically a signal) into its constituent frequencies.
- It transforms a function from the time domain (or spatial domain) into the frequency domain.
- FT states that any function that periodically repeats itself can be expressed as the sum of sines and cosines of different frequencies and different amplitudes.

- Image enhancement in the frequency domain is needed because certain image characteristics, such as noise, edges, and textures, are more easily manipulated in the frequency domain than in the spatial domain.
- The frequency domain represents an image in terms of its frequency components, which can be selectively modified to achieve various enhancement effects.

Key Reasons for Image Enhancement in the Frequency Domain

- Better Noise Reduction
- Edge and Contrast Enhancement
- Sharpening and Smoothing
- Efficient Computation with Fast Fourier Transform (FFT)

Common Techniques Used in Frequency-Domain Image Enhancement

- **Fourier Transform (FT)**: Converts an image from the spatial domain to the frequency domain.
- **Low-Pass Filtering (LPF)**: Reduces high-frequency noise and smoothens the image.
- **High-Pass Filtering (HPF)**: Enhances edges and details by emphasizing high-frequency components.

Many applications, such as medical imaging, remote sensing, and industrial quality control, benefit from frequency-based techniques.

- The fourier transform of f(x) is

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

$$\mathrm{F}\ (\mathrm{u}) = \int_{-\infty}^{\infty} f(x)[\cos(2\pi ux) - jsin(2\pi ux)]\mathrm{dx}$$

- Since F(u) is complex,

F(u) = R(u) + j I(u)

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

Phase = $\tan^{-1}\frac{I(u)}{R(u)}$

- The Inverse Fourier Transform is

$$\mathrm{f(x)} = \int_{-\infty}^{\infty} F(u)e^{j\,2\pi u x}\,\mathrm{dx}$$

- For numerical applications, Discrete Fourier Transform (DFT) is used

$$\mathrm{F(u)} = \sum_{x=0}^{N-1} f(x) e^{-j2\pi ux/N} dx$$

Where N = total number of samples in input signal

2D Fourier Transform

- It decomposes a 2D function (e.g., an image) into its frequency components in both horizontal and vertical directions.

For a continuous function $f(x, y)$, the **2D Fourier Transform** is defined as:

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-i2\pi(ux+vy)} dx\, dy$$

where:

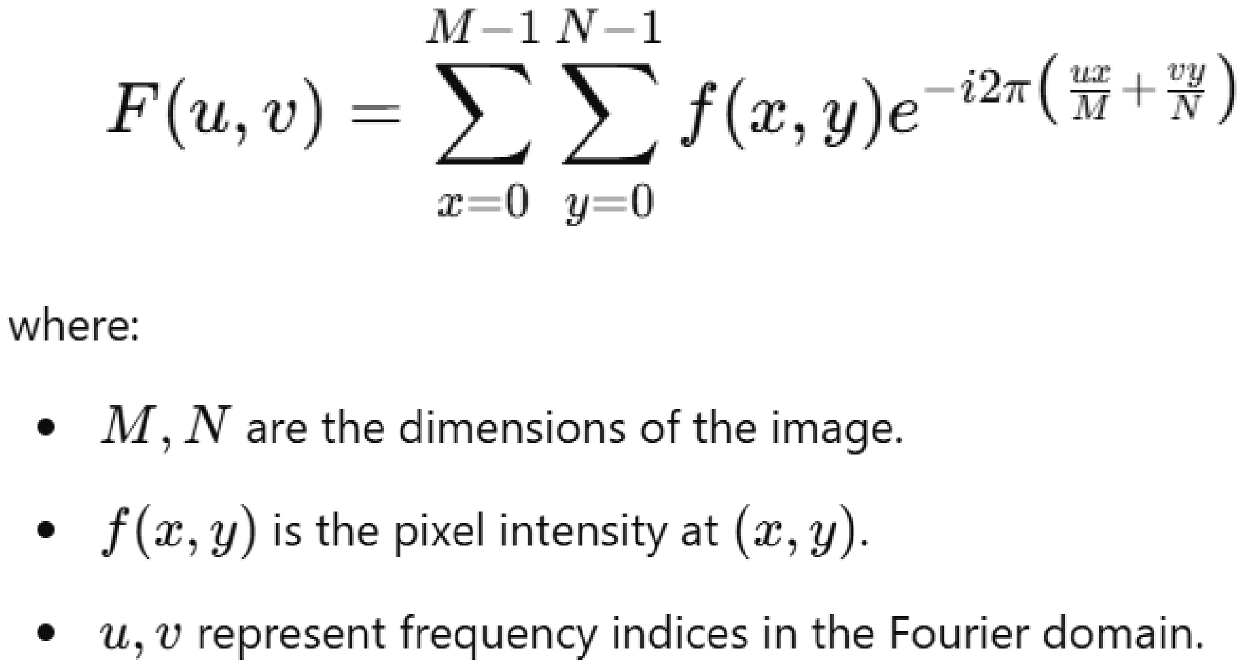
- $f(x, y)$ is the function in the **spatial domain** (e.g., an image or a surface function).
- $F(u, v)$ is the function in the **frequency domain.**
- u, v represent spatial frequencies along the **x** and **y** axes.
- $e^{-i2\pi(ux+vy)}$ is the complex exponential basis function that decomposes $f(x, y)$ into frequency components.

Inverse 2D Fourier Transform

$$f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v) e^{i2\pi(ux+vy)} \, du \, dv$$

Discrete 2D fourier transform

- For digital images and discrete signals, we use the **2D Discrete Fourier Transform (2D-DFT)**:



$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-i2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

where:

- M, N are the dimensions of the image.
- $f(x,y)$ is the pixel intensity at (x,y).
- u, v represent frequency indices in the Fourier domain.

Inverse 2D Discrete Fourier Transform

$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{i2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)}$$

Find the DFT of f(x) = {0,1,2,1}

$$F(u) = \sum_{x=0}^{N-1} f(x) e^{-j2\pi ux/N} dx$$

F(u) =

Compute F(0), F(1), F(2), F(3)

DFT using the Matrix Method

- Is an NxN matrix based on the Twiddle Factor

$$X[k] = \sum_{n=0}^{N-1} x(n) e^{-j2\pi nk/N}$$

Twiddle Factor = $W_N = e^{-i2\pi/N}$ = $\cos\left(\frac{2\pi}{N}\right) - jsin(\frac{2\pi}{N})$

The DFT equation reduces to :

$$F(u) = \sum_{x=0}^{N-1} f(x) {W_N}^{nk}$$

- For N=4, the NxN i.e., 4x4 matrix of W_N is

$$W_4 = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix}$$

$$W_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

Working for the W values

For $N = 4$:

$$W_N = e^{-i2\pi/N} = \cos\left(\frac{2\pi}{N}\right) - jsin(\frac{2\pi}{N})$$

$$W_4 = e^{-j\frac{2\pi}{4}} = e^{-j\frac{\pi}{2}}$$

$$W_N^k = \cos\left(\frac{2\pi k}{N}\right) - jsin(\frac{2\pi k}{N})$$

$$W_4^0 = e^0 = 1$$

$$W_4^1 = e^{-j\pi/2} = \cos\left(\frac{\pi}{2}\right) - jsin(\frac{\pi}{2}) = \text{-j}$$

$$W_4^2 = e^{-j\pi} = \cos(\pi) - jsin(\pi) = \text{-1}$$

$$W_4^3 = e^{-j3\pi/2} = \cos\left(\frac{3\pi}{2}\right) - jsin(\frac{3\pi}{2}) = \text{j}$$

Find the DFT of the sequence f(x) = {0,1,2,1} using the matrix method

N =

F(k) = W4 * f(x)

$$W_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

Find the DFT for the following image

1	3	6	8
9	8	8	2
5	4	2	3
6	6	3	3

DFT = W_N * X * WN

Properties of Discrete Fourier Transform

1. The Separability Property

- states that a **2D DFT** can be computed as a sequence of **1D DFTs** along each dimension separately.
- For an **N×N square image** f(x,y), the **2D DFT** is given by:

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

1. The Separability Property

Step 1: Apply 1D DFT Along Each Row

- For each **row x** in the image, compute a **1D DFT along the y-axis**:

$$F_r(x,v) = \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi \frac{vy}{N}}$$

- This gives an **intermediate transformed image** $F_r(x,v)$, where each row is now in the frequency domain.

1. The Separability Property

Step 2: Apply 1D DFT Along Each Column

- Now, compute the **1D DFT along the x-axis** for each frequency component v:

$$F(u,v) = \sum_{x=0}^{N-1} F_r(x,v) e^{-j2\pi \frac{ux}{N}}$$

- This gives the **final 2D DFT** F(u,v).

Example: Find the DFT of the image using the separability property

0	1	2	1
1	2	3	2
2	3	4	3
1	2	3	2

STEP 1: DFT along rows

- Since, N=4, we need a 4*4 DFT matrix
- For Row 1:

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

- Intermediate image obtained

$$\begin{bmatrix} 4 & -2 & 0 & -2 \\ 8 & -2 & 0 & -2 \\ 12 & -2 & 0 & -2 \\ 8 & -2 & 0 & -2 \end{bmatrix}$$

STEP 2: DFT along columns

- For column 1:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 4 \\ 8 \\ 12 \\ 8 \end{bmatrix} =$$

- Final DFT of the entire image

2. The Translation Property (Shifting Property)

If f(x,y) is multiplied by an exponential, the original Fourier transform F(u,v) get shifted by F(u-u_0, v-v_0)

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

Multiplying f(x,y) by $e^{j2\pi\frac{(u_0x+v_0y)}{N}}$

F(u,v) = $\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi\frac{(ux+vy)}{N}}\, e^{j2\pi\frac{(u_0x+v_0y)}{N}}$

3. Periodicity Property

The **DFT is periodic** in both the **spatial domain** and the **frequency domain**.

This periodicity means that the transformed frequency spectrum repeats itself periodically.

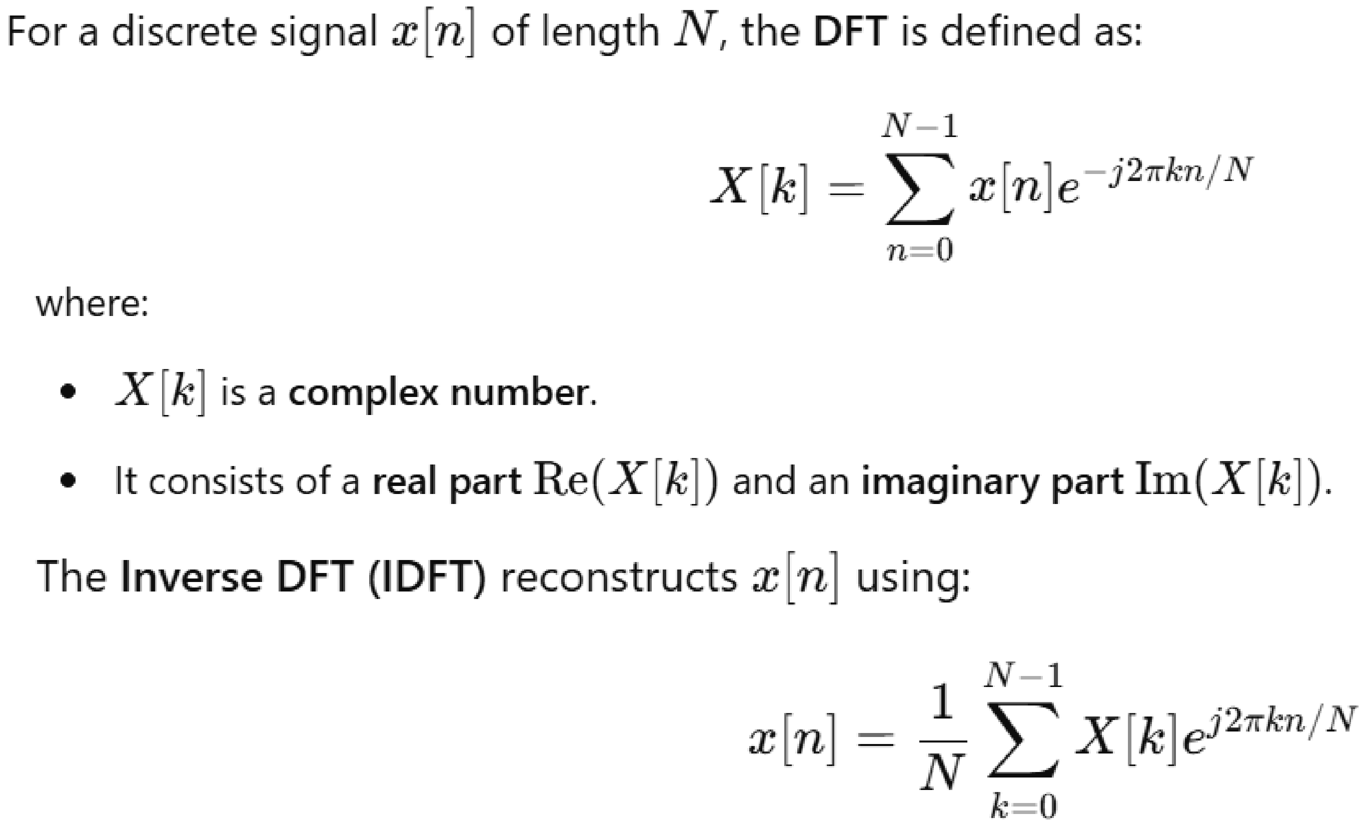
$F(u) = F(u+N)$

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N}$$

Replace k with k+N

4. Conjugate Symmetry Property

- for a **real-valued signal or image**, the Fourier transform exhibits **symmetry in its complex values**



For a discrete signal $x[n]$ of length N, the **DFT** is defined as:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N}$$

where:

- $X[k]$ is a **complex number**.
- It consists of a **real part** $\mathrm{Re}(X[k])$ and an **imaginary part** $\mathrm{Im}(X[k])$.

The **Inverse DFT (IDFT)** reconstructs $x[n]$ using:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j2\pi kn/N}$$

4. Conjugate Symmetry Property

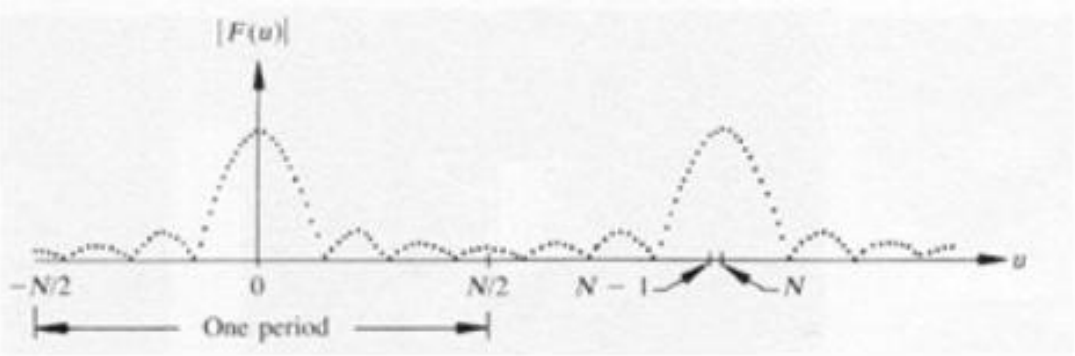
For a **real-valued signal** $x[n]$, the DFT exhibits **conjugate symmetry**:

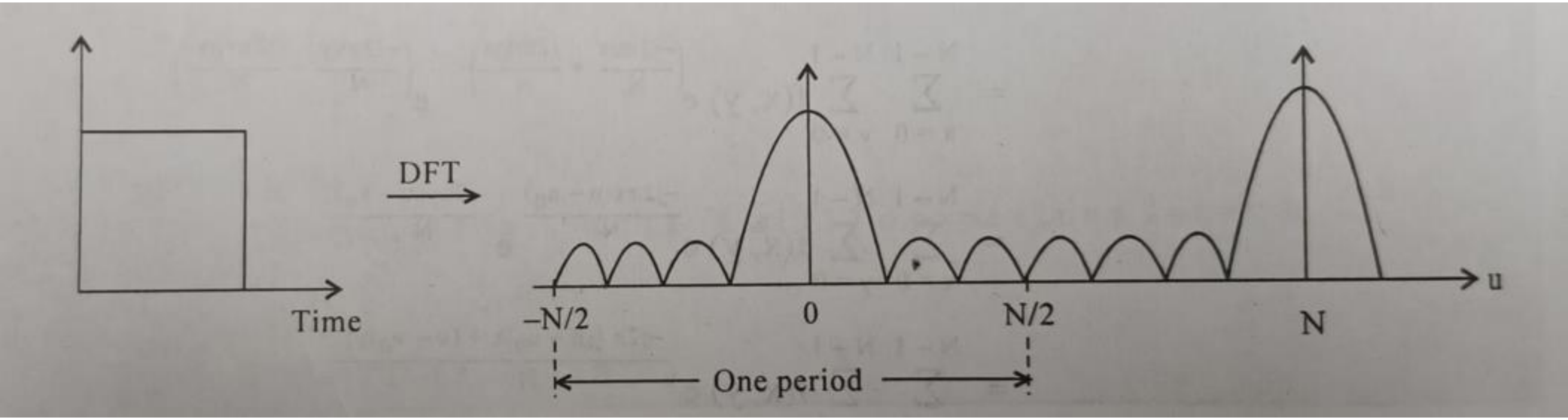
$$X[N-k] = X^*[k]$$

where $X^*[k]$ is the **complex conjugate** of $X[k]$, meaning:

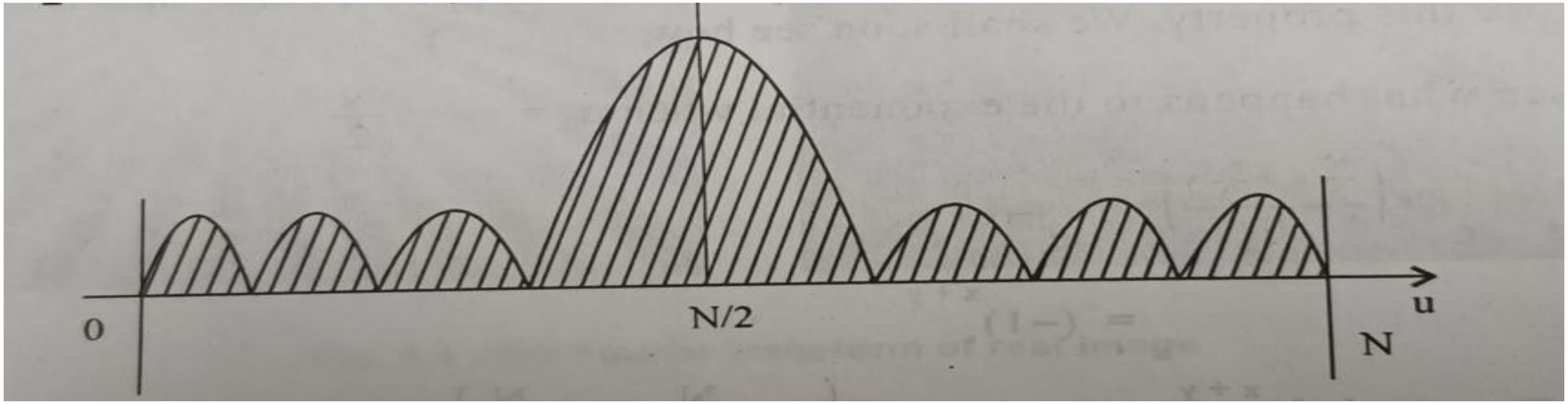
$$\mathrm{Re}(X[N-k]) = \mathrm{Re}(X[k])$$

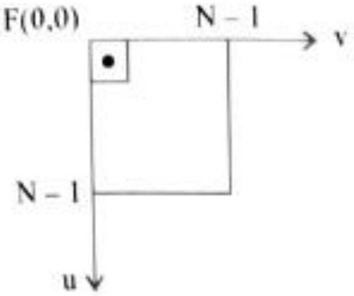
$$\mathrm{Im}(X[N-k]) = -\mathrm{Im}(X[k])$$





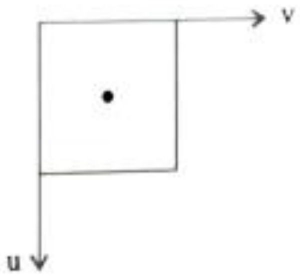
- The FT is symmetric about 0. i.e. –N/2 to 0 is a reflection of 0 to N/2
- To display one full period, we need to move the origin to a point u=N/2





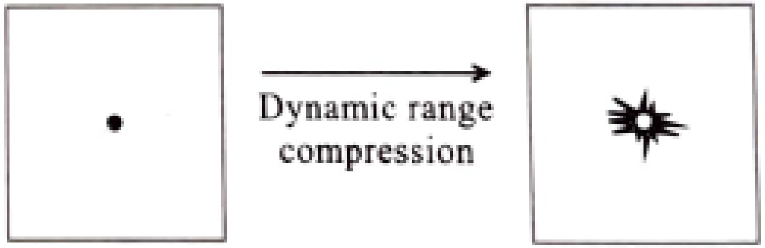
(**Note** : F(0,0) is the d.c which is the maximum value. Value at the centre will be very small because the higher frequencies lie there)

After using the shifting property, we get



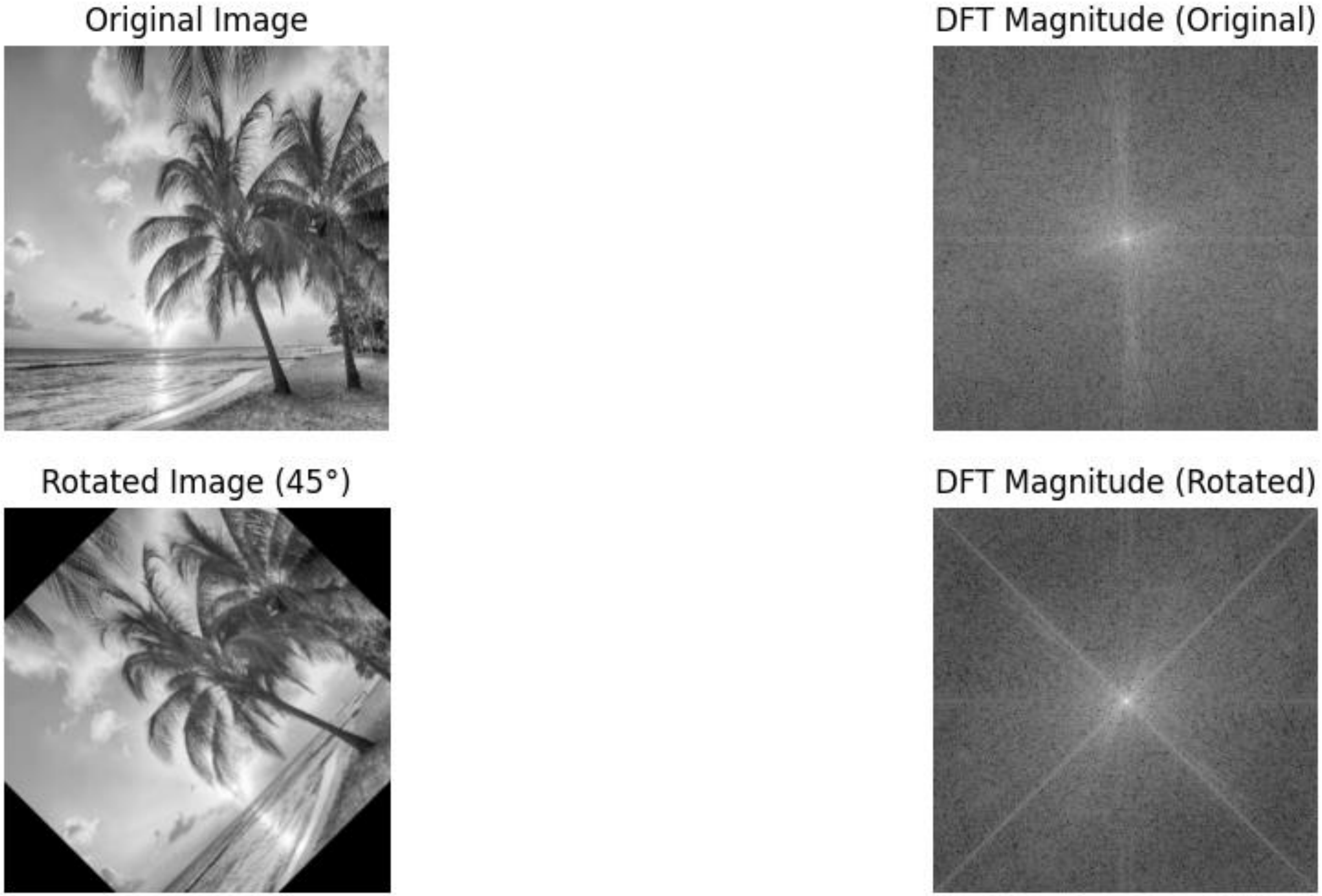
Now the d.c. value will lie at the centre i.e. F(N/2 , N/2) and the high frequencies will lie at exterior ends.

- The dynamic range of the d.c. component is very large compared to the higher frequencies
- The dynamic range of FT can be as large as [0, $2.5\text{x}10^6$]
- A dynamic range compressor is used
  $G(u,v) = c.\log(1+F(u,v))$
- Now the spectrum looks like



5. Rotation Property

- if an image is rotated by a certain angle in the spatial domain, its corresponding **DFT magnitude spectrum also rotates by the same angle** in the frequency domain.



Original Image

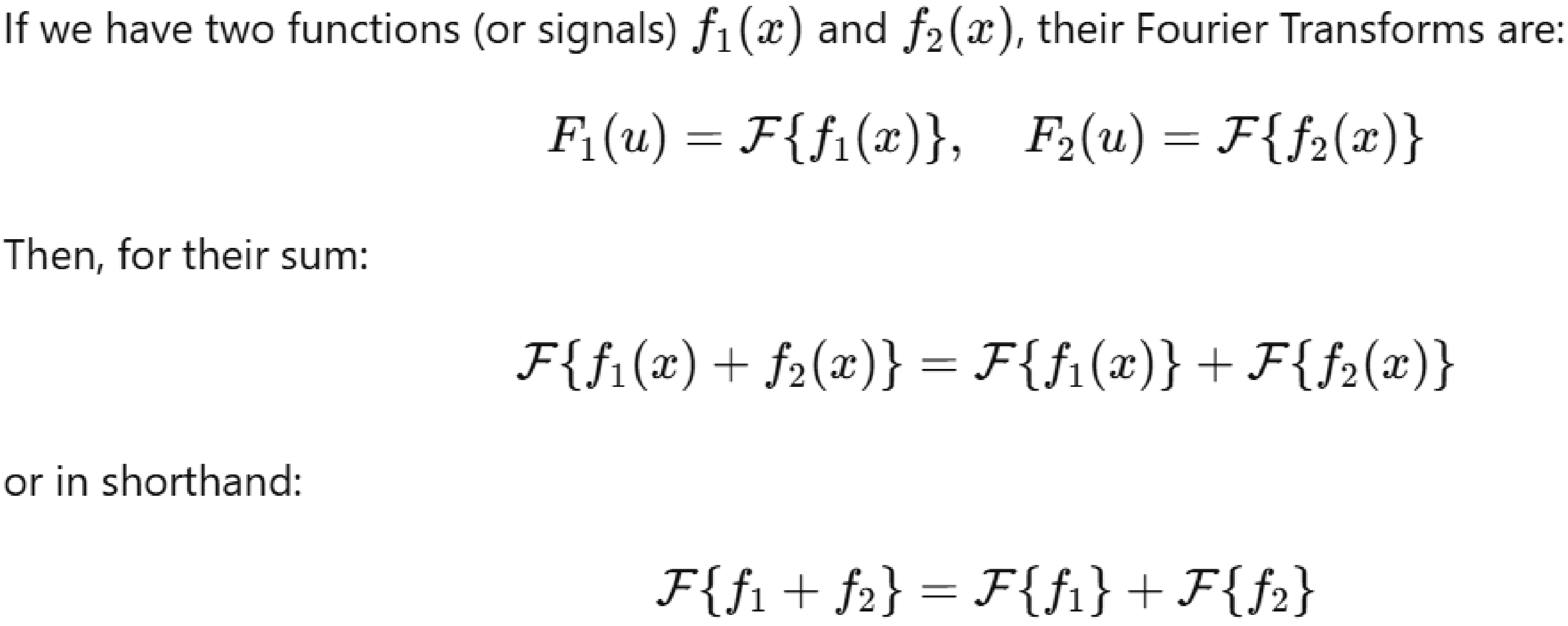
DFT Magnitude (Original)

Rotated Image (45°)

DFT Magnitude (Rotated)

6. Distributivity and Scaling Property

Fourier transform and its inverse is distributive over addition but not over multiplication.



If we have two functions (or signals) $f_1(x)$ and $f_2(x)$, their Fourier Transforms are:

$$F_1(u) = \mathcal{F}\{f_1(x)\}, \quad F_2(u) = \mathcal{F}\{f_2(x)\}$$

Then, for their sum:

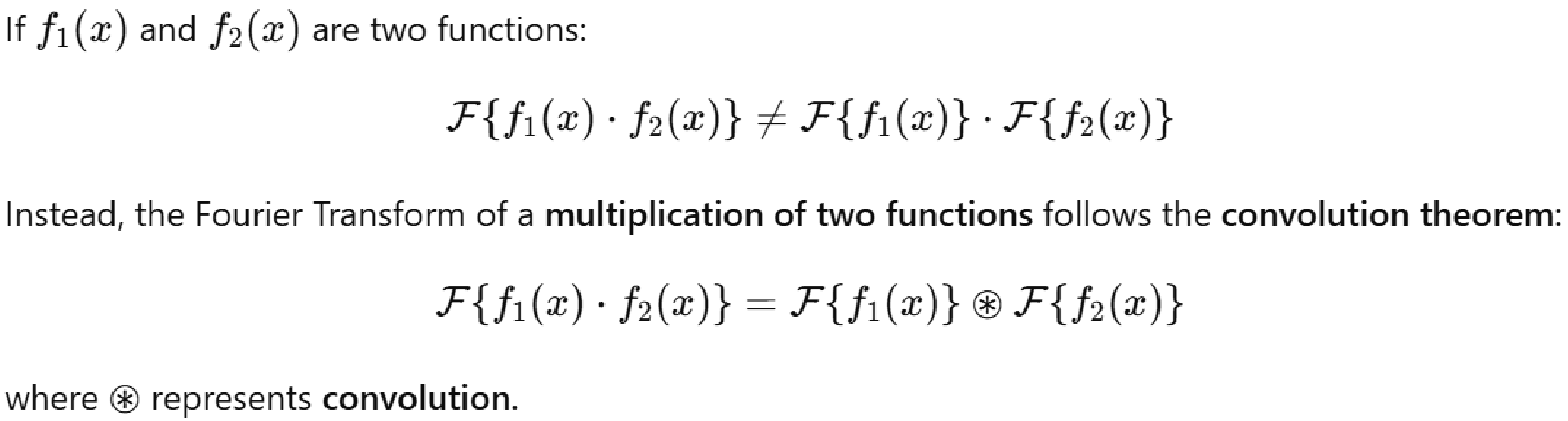
$$\mathcal{F}\{f_1(x) + f_2(x)\} = \mathcal{F}\{f_1(x)\} + \mathcal{F}\{f_2(x)\}$$

or in shorthand:

$$\mathcal{F}\{f_1 + f_2\} = \mathcal{F}\{f_1\} + \mathcal{F}\{f_2\}$$

6. Distributivity and Scaling Property

The **Fourier Transform is NOT distributive over multiplication** in the spatial domain.



If $f_1(x)$ and $f_2(x)$ are two functions:

$$\mathcal{F}\{f_1(x) \cdot f_2(x)\} \neq \mathcal{F}\{f_1(x)\} \cdot \mathcal{F}\{f_2(x)\}$$

Instead, the Fourier Transform of a **multiplication of two functions** follows the **convolution theorem**:

$$\mathcal{F}\{f_1(x) \cdot f_2(x)\} = \mathcal{F}\{f_1(x)\} \circledast \mathcal{F}\{f_2(x)\}$$

where $\circledast$ represents **convolution**.

7. Average Value Property

- states that the **DC (zero-frequency) component of the DFT represents the average (mean) value of the image** in the spatial domain.

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

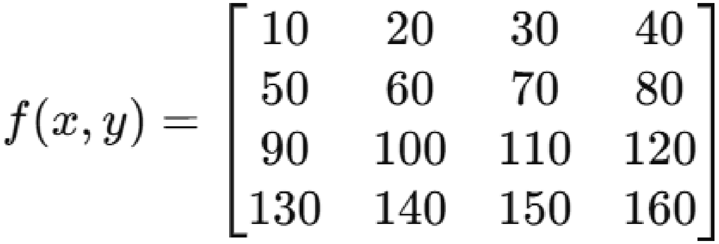
- When u=0 and v=0

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

The **average value (mean intensity) of the image** is: $\frac{F(0,0)}{MN}$

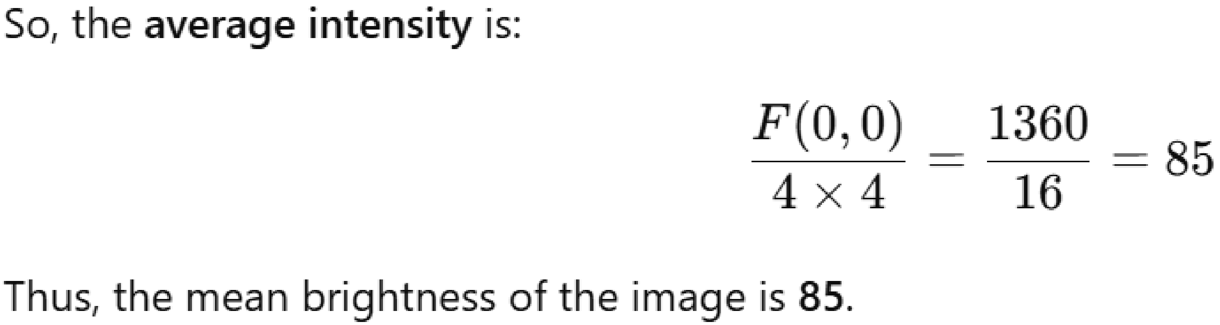
7. Average Value Property

- Example:



$$f(x,y) = \begin{bmatrix} 10 & 20 & 30 & 40 \\ 50 & 60 & 70 & 80 \\ 90 & 100 & 110 & 120 \\ 130 & 140 & 150 & 160 \end{bmatrix}$$

The **DC component** is:
F(0,0)=10+20+30+40+50+60+70+80+90+100+110+120+130+140+150+160
=1360



So, the **average intensity** is:

$$\frac{F(0,0)}{4 \times 4} = \frac{1360}{16} = 85$$

Thus, the mean brightness of the image is **85**.

8. Laplacian Property (Second Derivative)

- states that the **DFT of the Laplacian (∇^2) of an image in the spatial domain corresponds to multiplication by the frequency squared in the frequency domain**.
- if **f(x,y)** is an image and **F{f(x,y)}** represents its **DFT**, then the Laplacian in the **Fourier domain** follows:

$$\mathcal{F}\{\nabla^2 f(x,y)\} = -(u^2 + v^2) \cdot F(u,v)$$

9. Convolution Property

- states that **convolution in the spatial domain corresponds to multiplication in the frequency domain**, and vice versa.
- If f(x,y) and g(x,y) are two images, their convolution in the **spatial domain** is:

$$h(x, y) = f(x, y) * g(x, y)$$

9. Convolution Property

- Taking DFT of both sides

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

- Thus,

$$\mathcal{F}\{f(x,y) * g(x,y)\} = \mathcal{F}\{f(x,y)\} \cdot \mathcal{F}\{g(x,y)\}$$

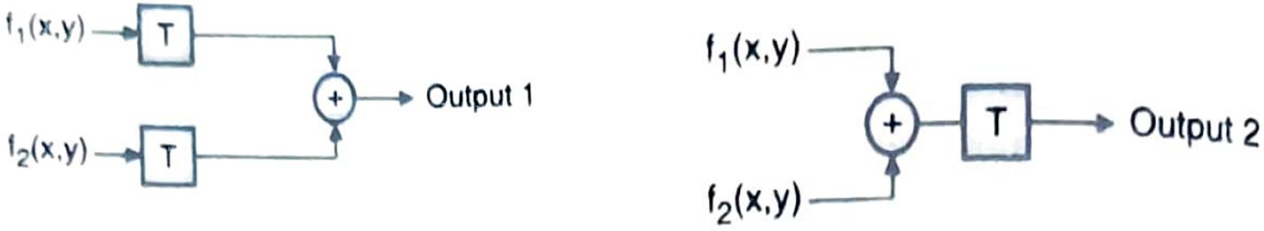
- This means that **convolution in the spatial domain transforms into simple multiplication in the frequency domain**, making filtering operations computationally efficient.

10. Linearity Property

- States that the DFT of a linear weighted combination of two or more signals is equal to similar linear weighted combination of the DFT of individual signals

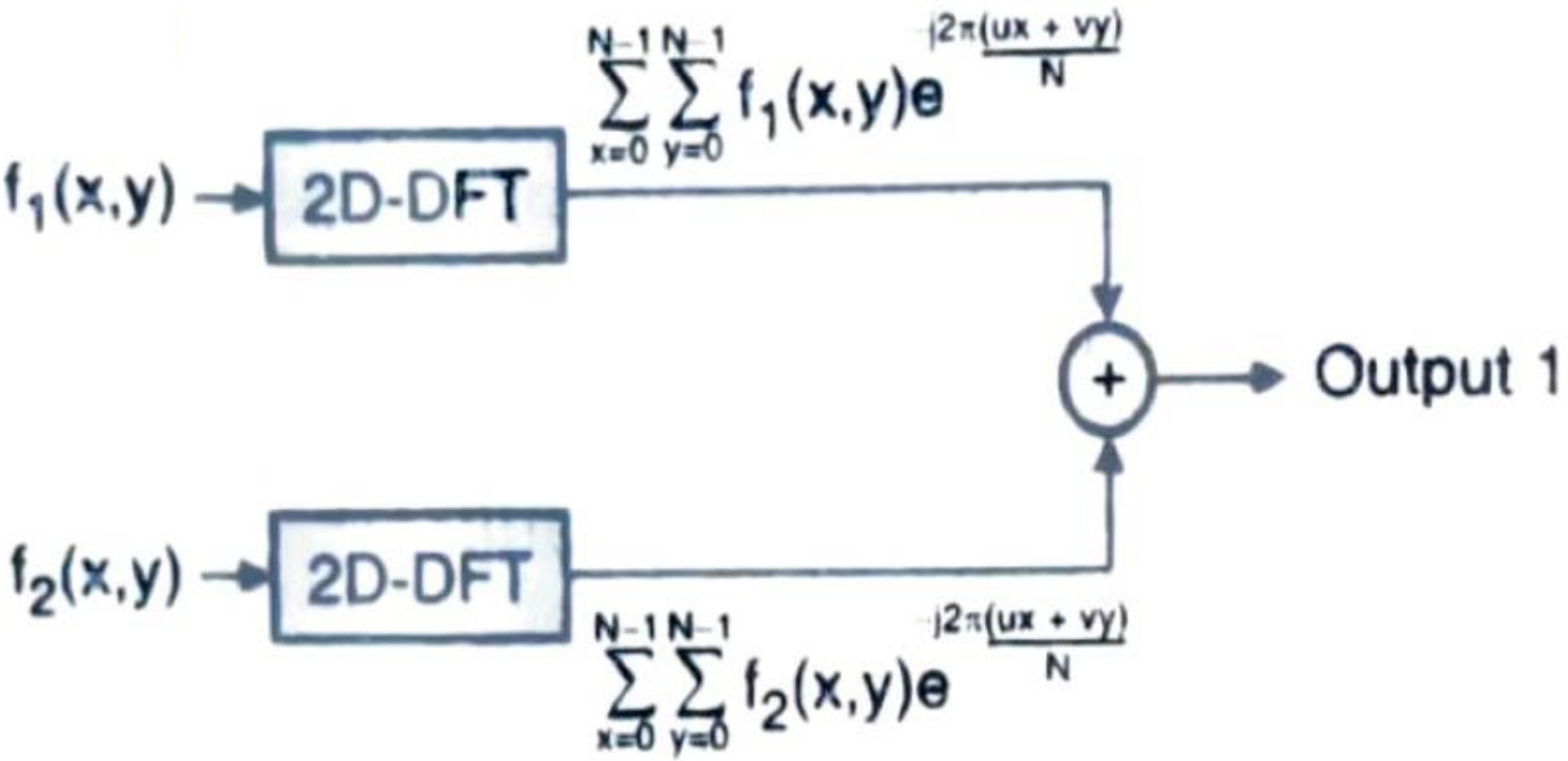
10. Linearity Property

- Consider 2 images f1(x,y) and f2(x,y) of the same size (NxN). T is the transformation (2D-DFT).
- A system is said to be linear if Output1=Output2



10. Linearity Property

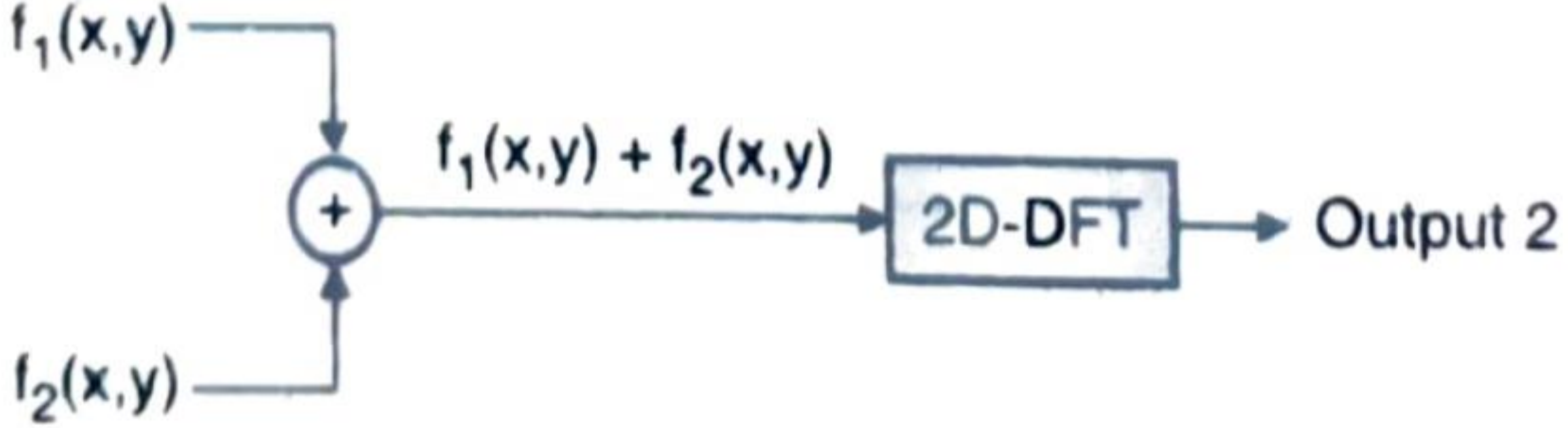
- For Output 1:


$\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f_1(x,y)e^{-j2\pi\frac{(ux+vy)}{N}}$
$f_1(x,y)$
2D-DFT
Output 1
$f_2(x,y)$
2D-DFT
$\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f_2(x,y)e^{-j2\pi\frac{(ux+vy)}{N}}$

$$\text{Output 1} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f_1(x,y)\, e^{-j\,2\pi\left(\frac{ux+uy}{N}\right)} + \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f_2(x,y)\, e^{-j\,2\pi\left(\frac{ux+uy}{N}\right)}$$

10. Linearity Property

- For Output 2:

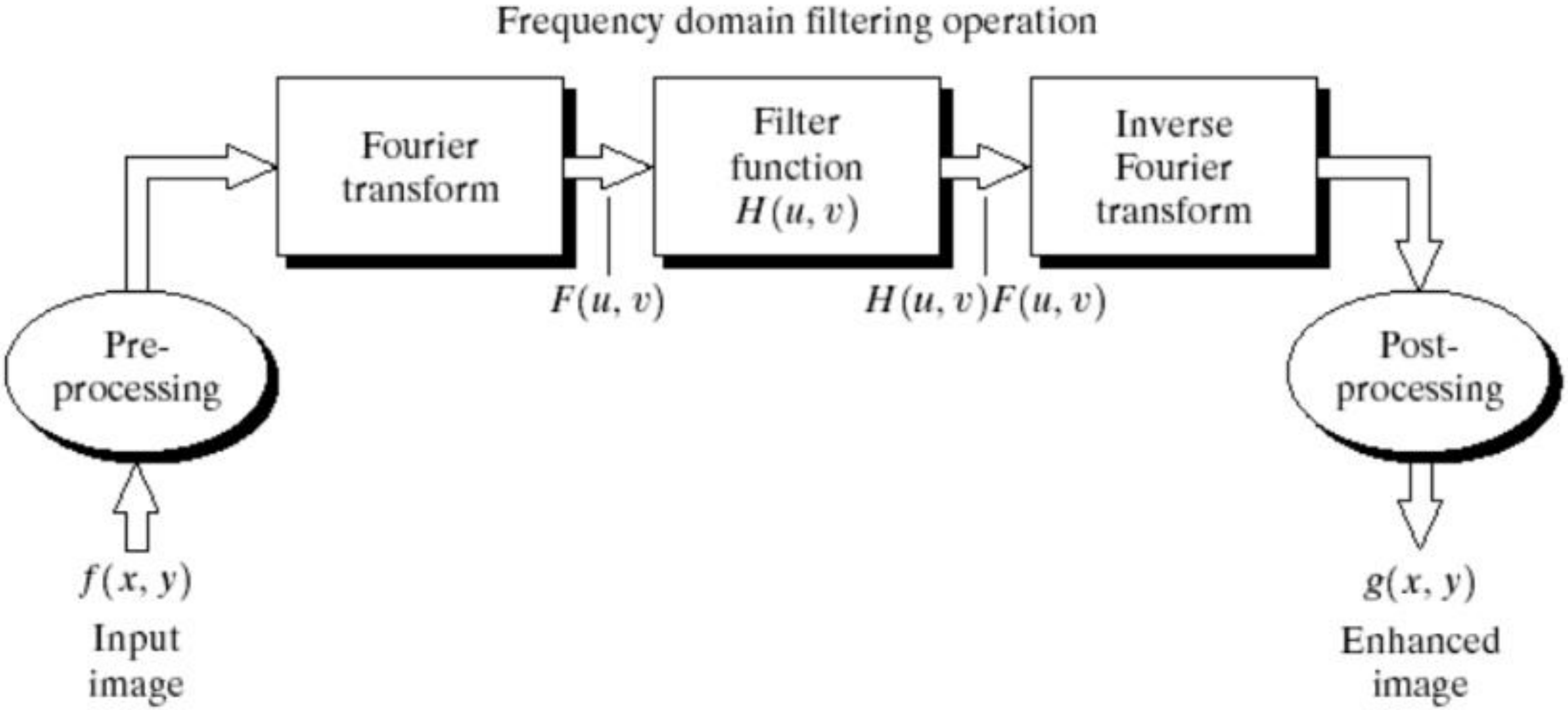


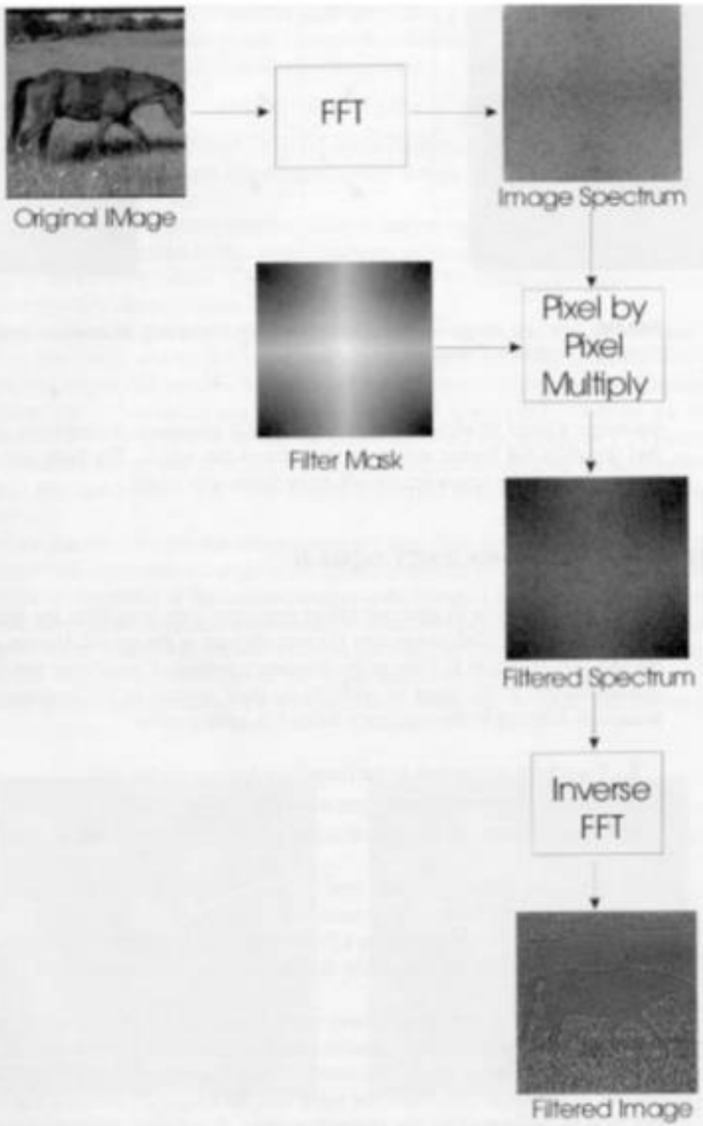
$$\text{Output } 2 = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} [f_1(x, y) + f_2(x, y)]\, e^{-j 2\pi \left(\frac{ux + uy}{N}\right)}$$

$$= \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f_1(x, y)\, e^{-j 2\pi \left(\frac{ux + uy}{N}\right)} + \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f_2(x, y)\, e^{-j 2\pi \left(\frac{ux + uy}{N}\right)}$$

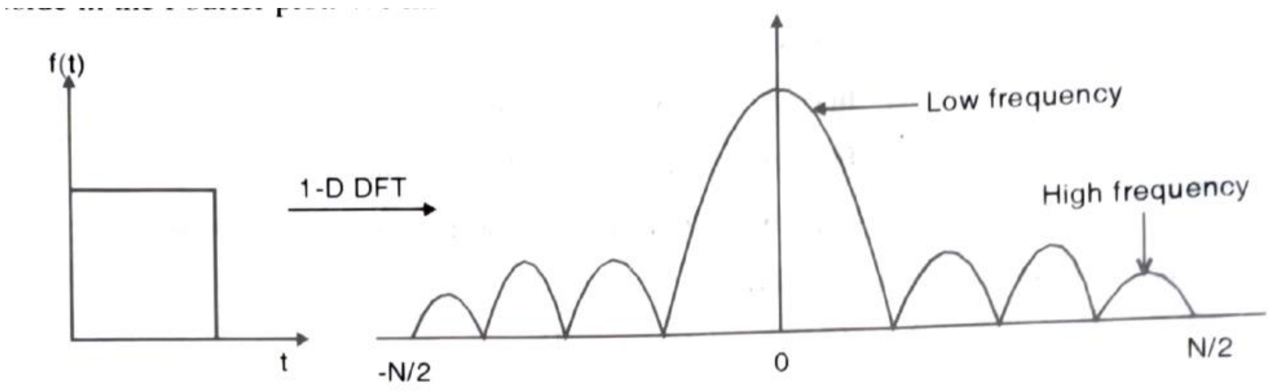
Image Enhancement in Frequency Domain

Steps in Frequency Domain



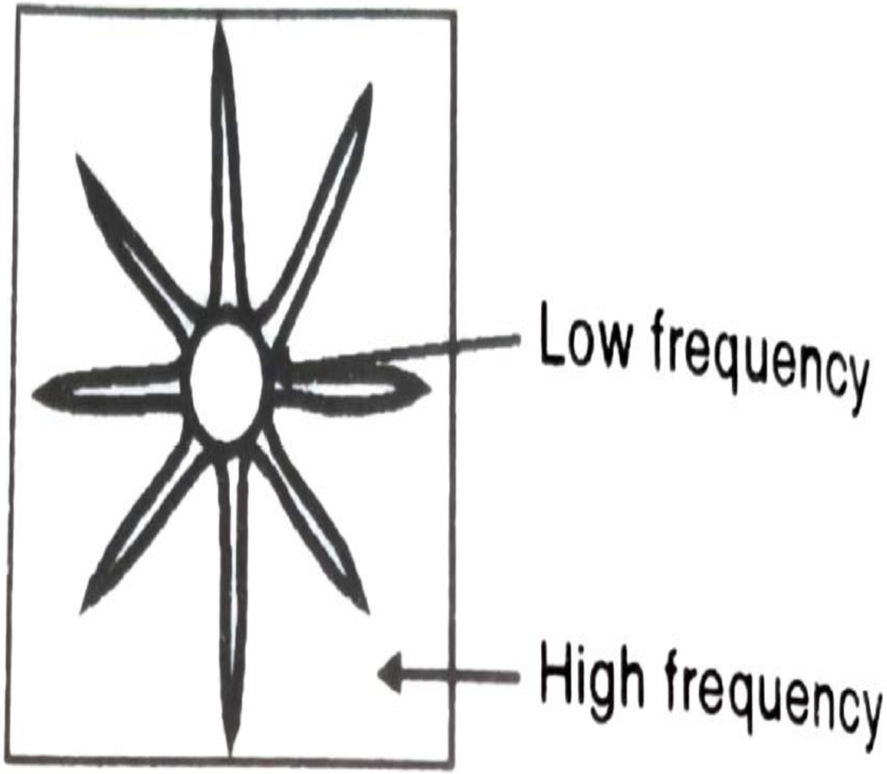

Original IMage
FFT
Image Spectrum
Pixel by Pixel Multiply
Filter Mask
Filtered Spectrum
Inverse FFT
Filtered Image

Frequencies in Fourier Plot for 1D signal



- Here, 0 represents the dc term, which is the average value of the signal over a period of time.
- The maximum frequency is N/2.

Frequencies in Fourier Plot for 2D domain



- In Fourier spectrum, the centre is where the low frequencies reside.
- When we go away from the centre, we encounter the high frequencies.

Low Pass Frequency Domain Filters

- Ideal Low Pass Filter (ILPF)
- Butterworth Low Pass Filter (BLPF)
- Gaussian Low Pass Filter

Smoothing Frequency Domain Filters

Smoothing is achieved in the frequency domain by dropping out the high frequency components

The basic model for filtering is:

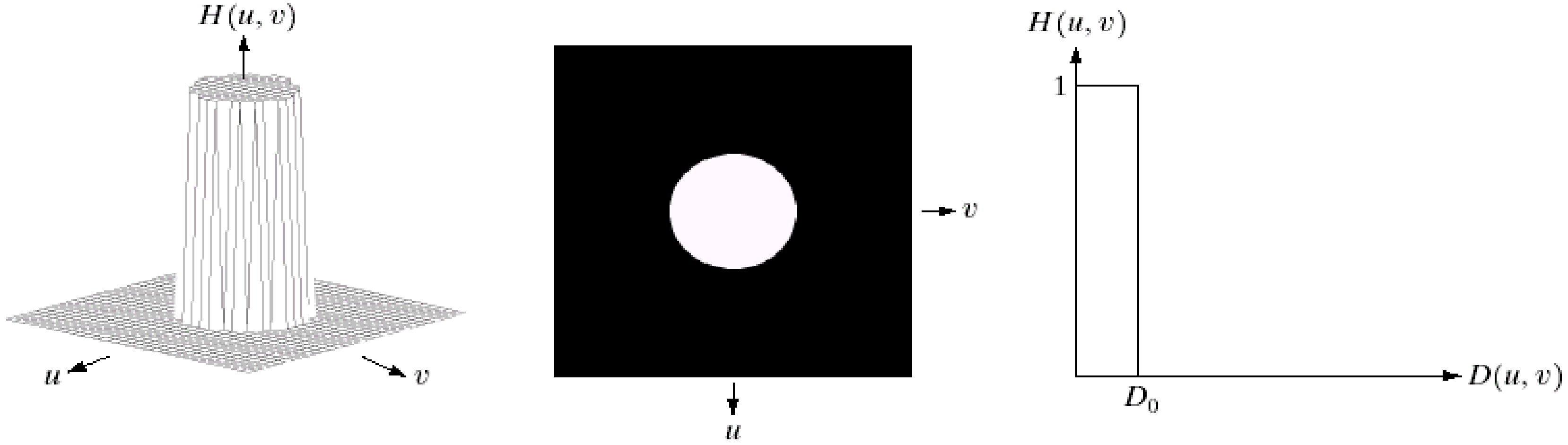
$$G(u,v) = H(u,v)F(u,v)$$

where $F(u,v)$ is the Fourier transform of the image being filtered and $H(u,v)$ is the filter transform function

Low pass filters – only pass the low frequencies, drop the high ones

Ideal Low Pass Filter

Simply cut off all high frequency components that are a specified distance D_0 from the origin of the transform



changing the distance changes the behaviour of the filter

Ideal Low Pass Filter

This filter cuts off all high frequency components of the fourier transform that are at the distance greater than a specified distance D_0

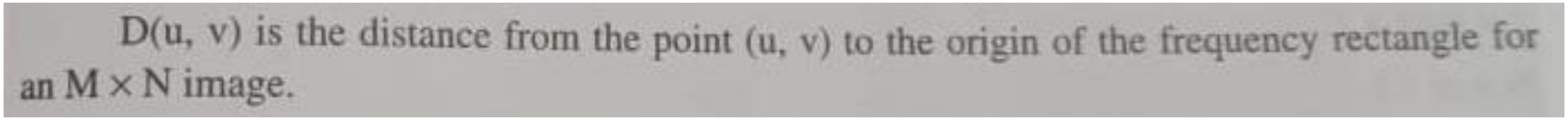
Ideal Low Pass Filter (cont…)

The transfer function for the ideal low pass filter can be given as:

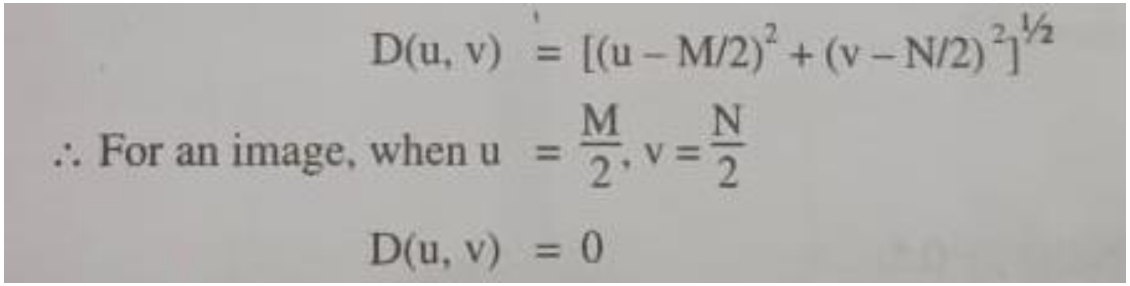
$$H(u,v) = \begin{cases} 1 \text{ if } D(u,v) \leq D_0 \\ 0 \text{ if } D(u,v) > D_0 \end{cases}$$

where *D(u,v)* is given as:

$$D(u,v) = [(u - M/2)^2 + (v - N/2)^2]^{1/2}$$



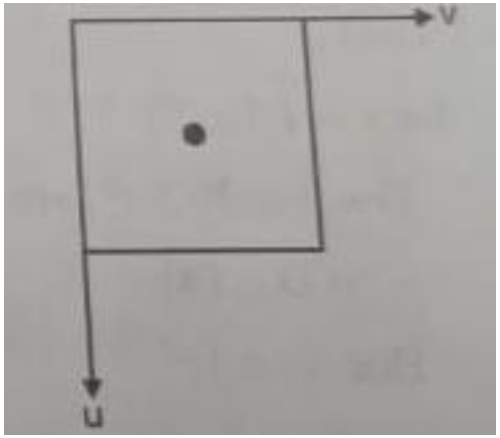
D(u, v) is the distance from the point (u, v) to the origin of the frequency rectangle for an M × N image.



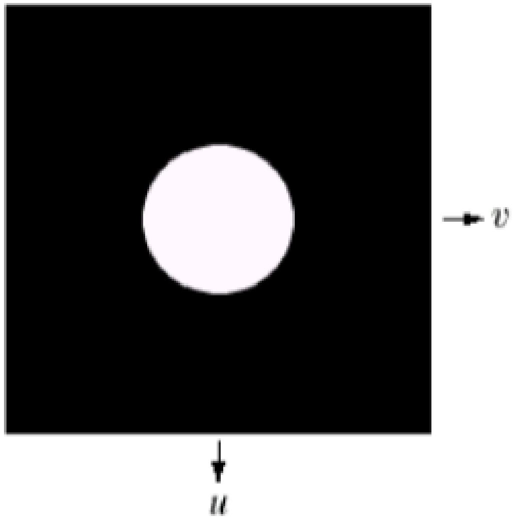
$$D(u, v) = [(u - M/2)^2 + (v - N/2)^2]^{1/2}$$

$$\therefore \text{ For an image, when } u = \frac{M}{2}, v = \frac{N}{2}$$

$$D(u, v) = 0$$

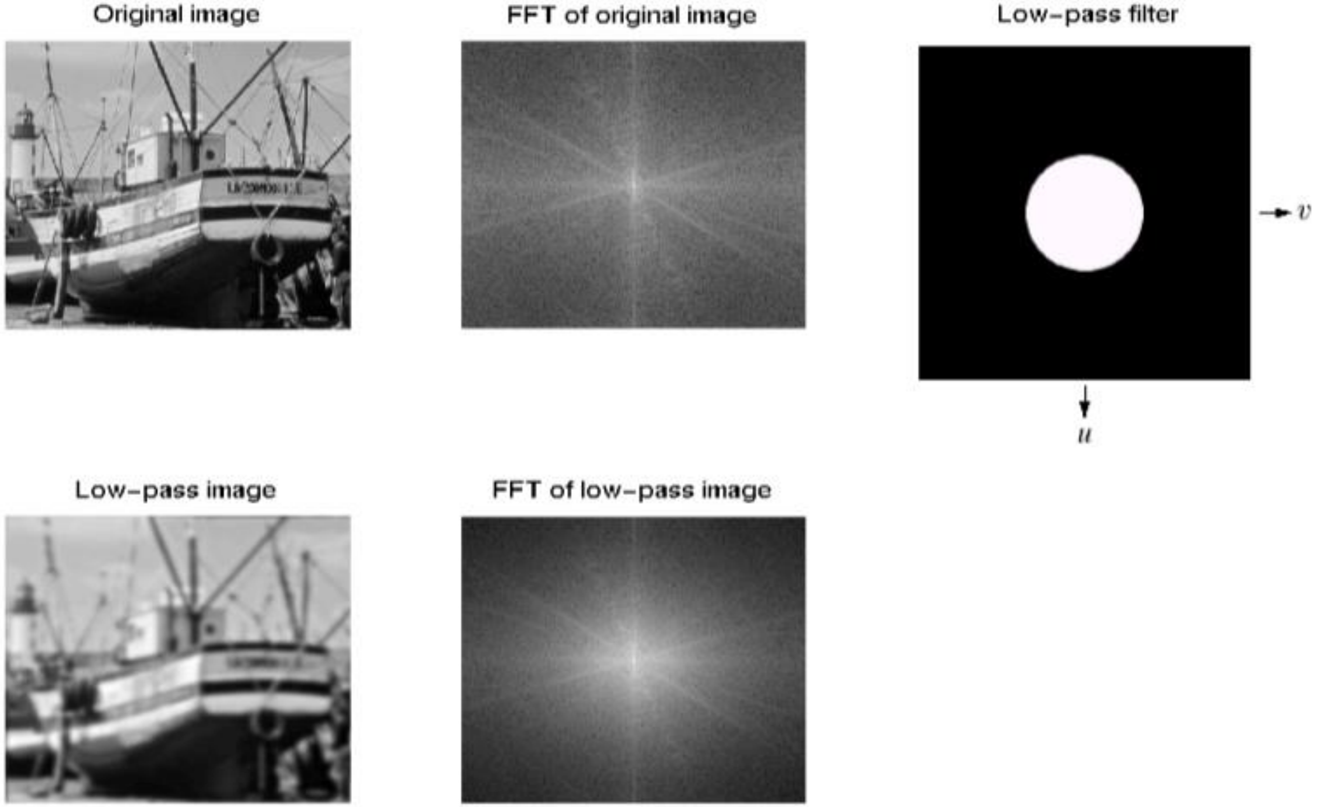


A 2D Low Pass Filter



- all frequencies inside a circle of radius D_0 are passed with no attenuation
- all frequencies outside this circle are completely attenuated.

Example



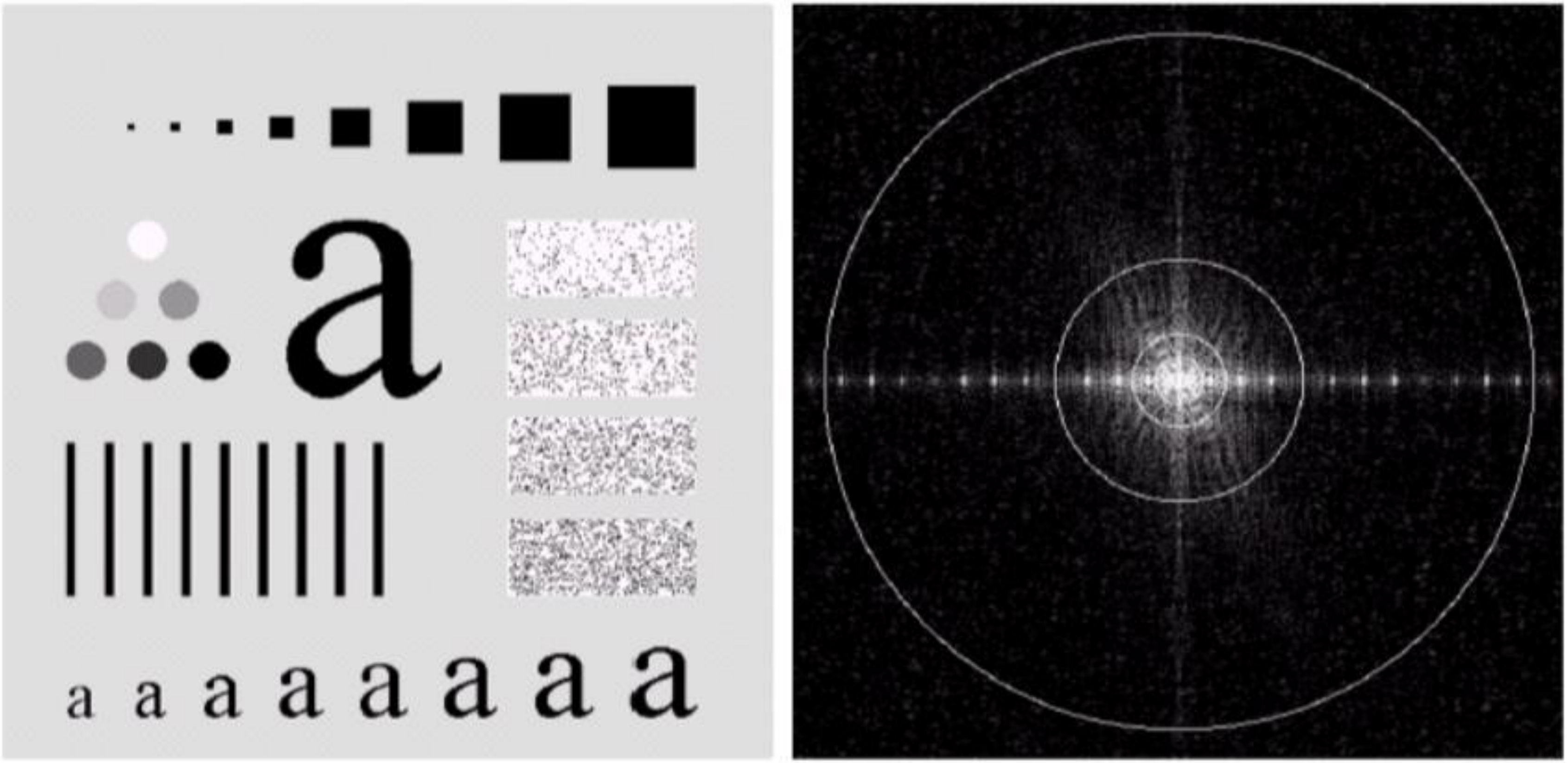
Original image

FFT of original image

Low-pass filter

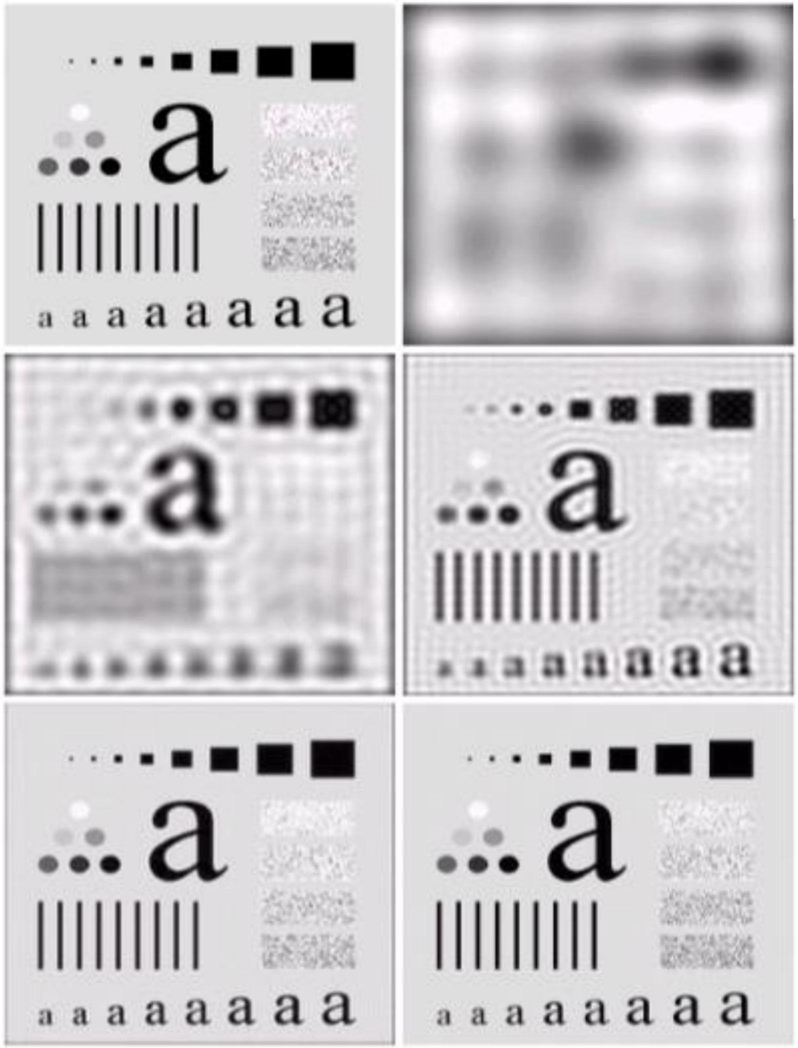
Low-pass image

FFT of low-pass image



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.



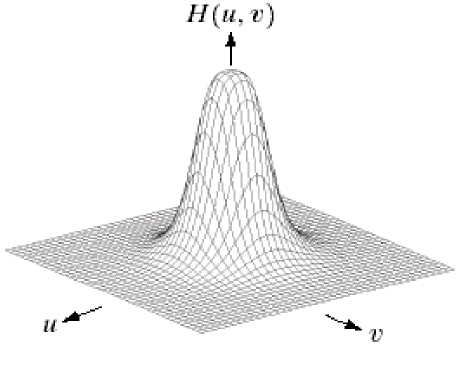
a b
c d
e f

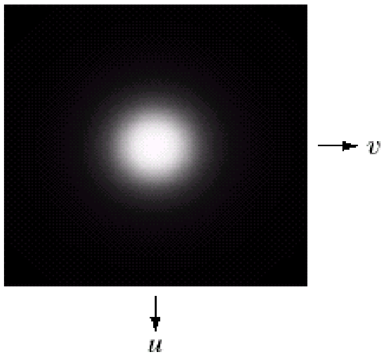
FIGURE 4.12 (a) Original image. (b)–(f) Results of ideal lowpass filtering with cutoff frequencies set at radii values of 5, 15, 30, 80, and 230, as shown in Fig. 4.11(b). The power removed by these filters was 8, 5.4, 3.6, 2, and 0.5% of the total, respectively.

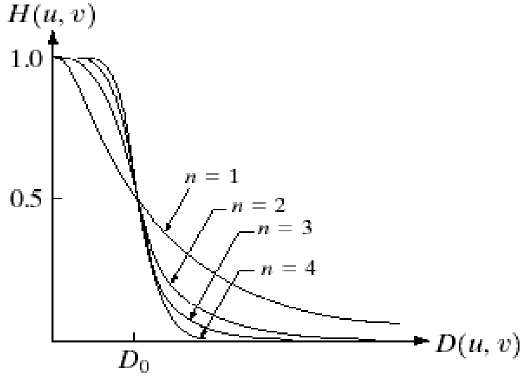
Butterworth Low Pass Filter

- The transfer function of a Butterworth lowpass filter of order n with cutoff frequency at distance D_0 from the origin is defined as:

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

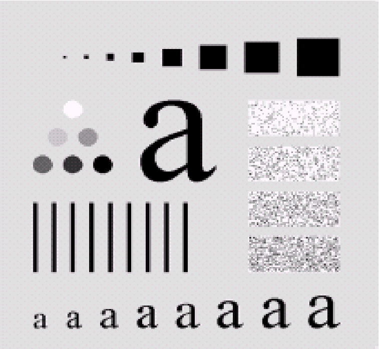


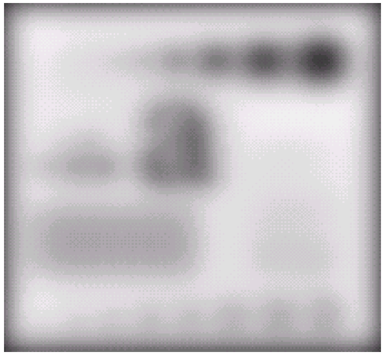




Butterworth Lowpass Filter (cont…)

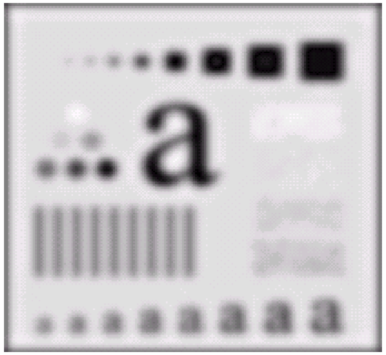
Original image

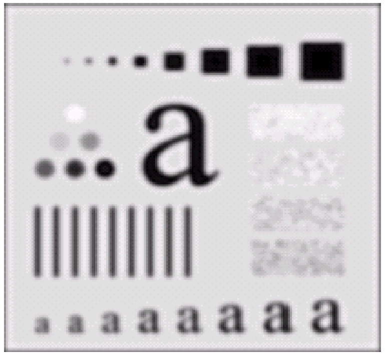




Result of filtering with Butterworth filter of order 2 and cutoff radius 5

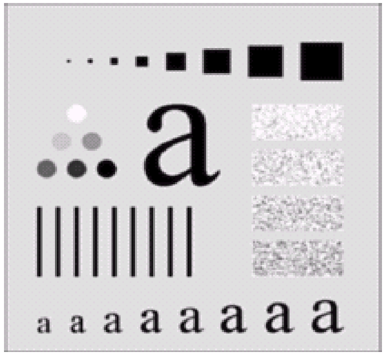
Result of filtering with Butterworth filter of order 2 and cutoff radius 15

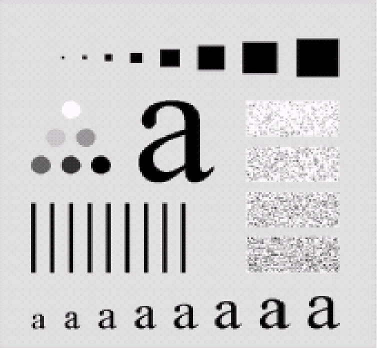




Result of filtering with Butterworth filter of order 2 and cutoff radius 30

Result of filtering with Butterworth filter of order 2 and cutoff radius 80





Result of filtering with Butterworth filter of order 2 and cutoff radius 230

Butterworth Lowpass Filter (cont...)

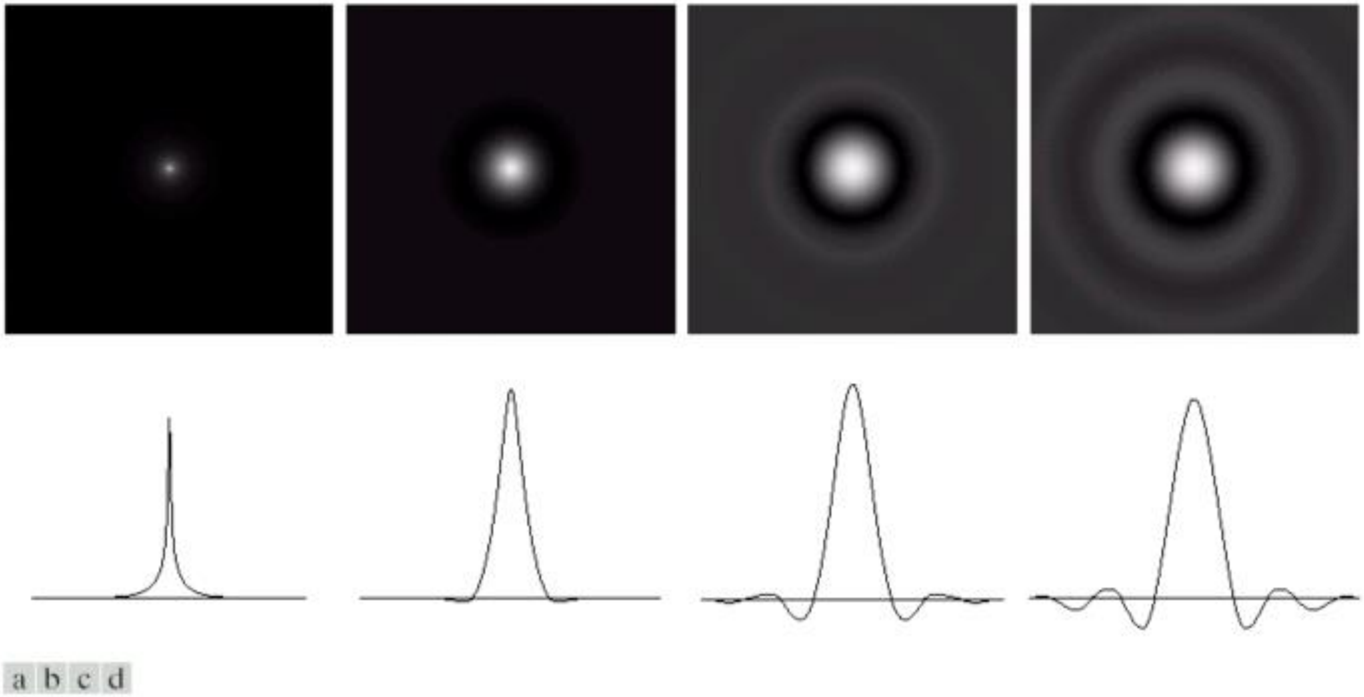
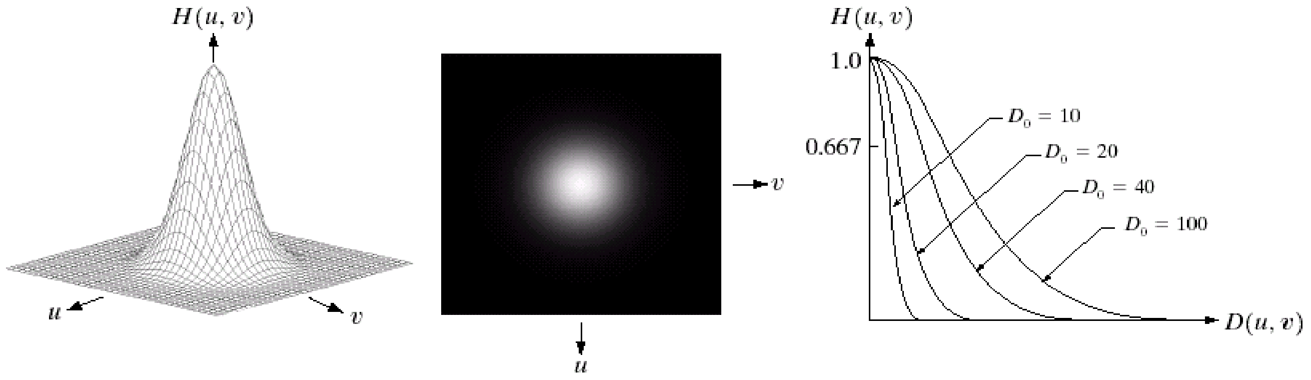


FIGURE 4.16 (a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding gray-level profiles through the center of the filters (all filters have a cutoff frequency of 5). Note that ringing increases as a function of filter order.

Gaussian Lowpass Filters

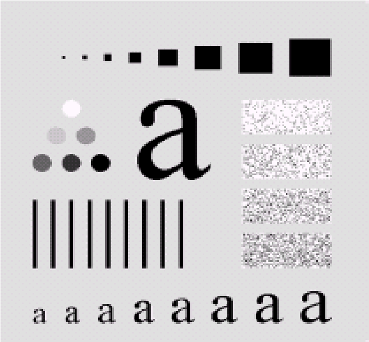
The transfer function of a Gaussian lowpass filter is defined as:

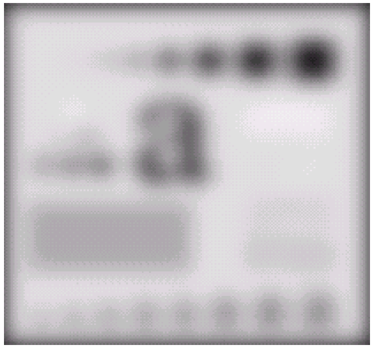
$$H(u,v) = e^{-D^2(u,v)/2{D_0}^2}$$



Gaussian Lowpass Filters (cont…)

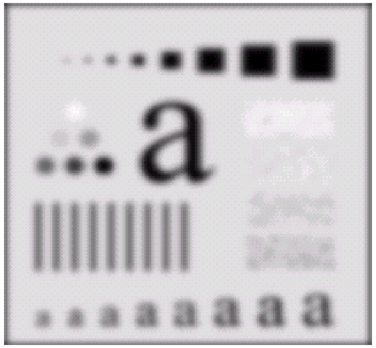
Original image

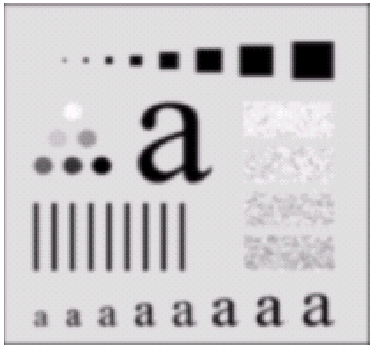




Result of filtering with Gaussian filter with cutoff radius 5

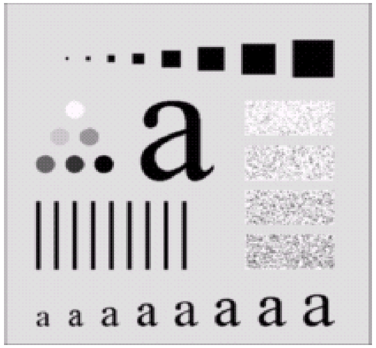
Result of filtering with Gaussian filter with cutoff radius 15

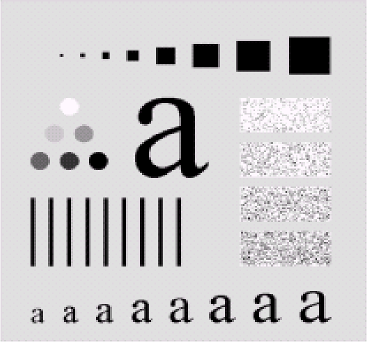




Result of filtering with Gaussian filter with cutoff radius 30

Result of filtering with Gaussian filter with cutoff radius 85

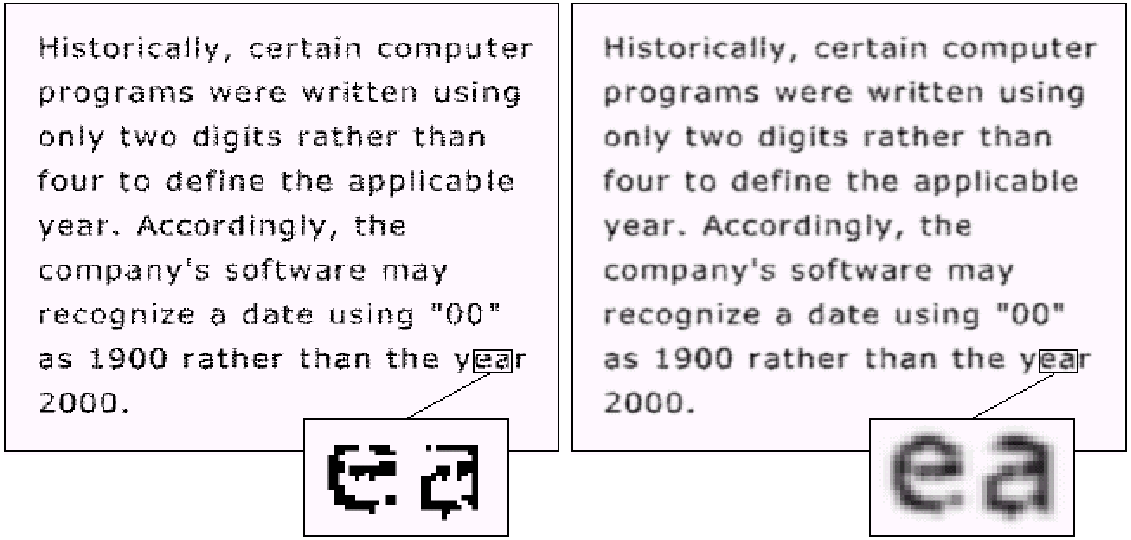




Result of filtering with Gaussian filter with cutoff radius 230

Lowpass Filtering Examples

A low pass Gaussian filter is used to connect broken text

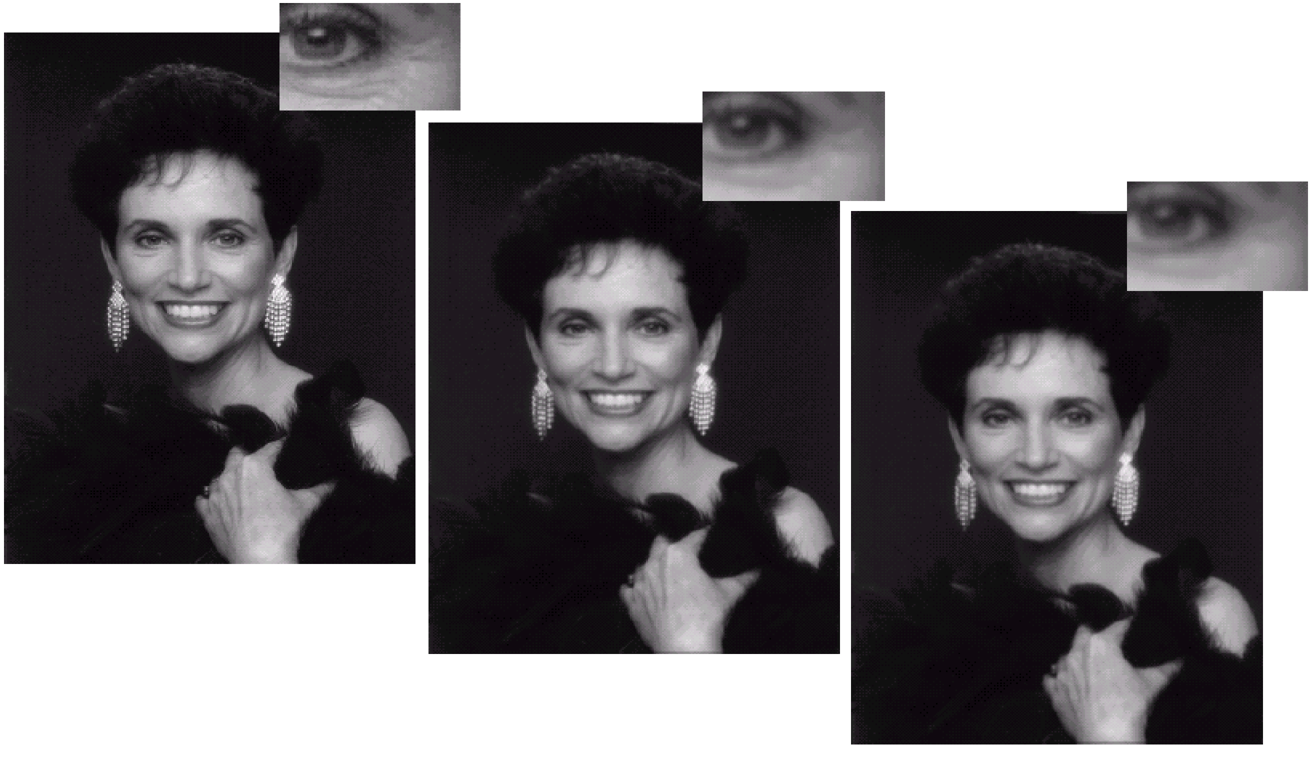


Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

Lowpass Filtering Examples (cont…)

Different lowpass Gaussian filters used to remove blemishes in a photograph



Sharpening in the Frequency Domain - High Pass Filters

Edges and fine detail in images are associated with high frequency components

High pass filters – only pass the high frequencies, drop the low ones

High pass frequencies are precisely the reverse of low pass filters, so:

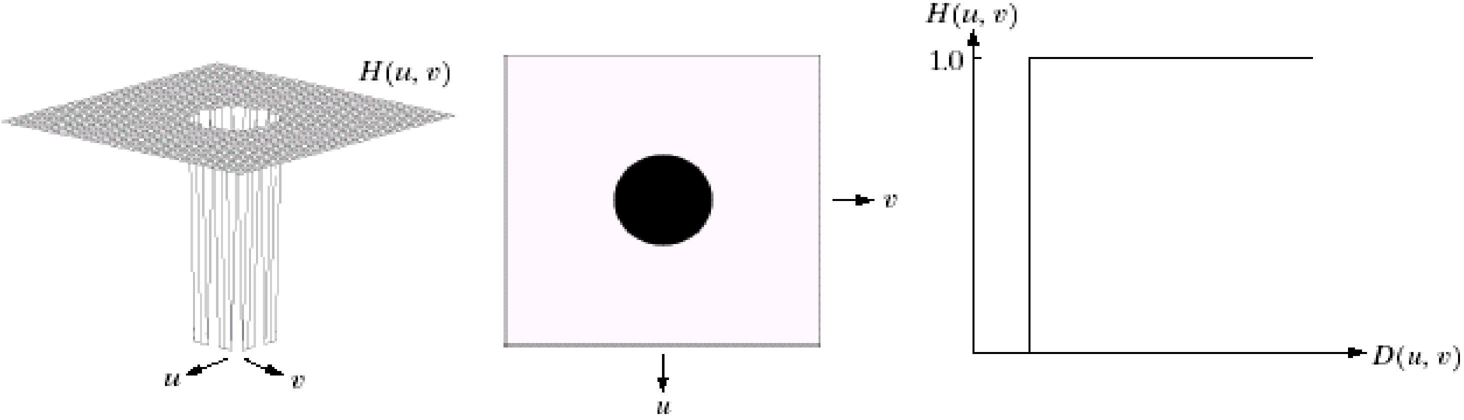
$$H_{hp}(u, v) = 1 - H_{lp}(u, v)$$

Ideal High Pass Filters

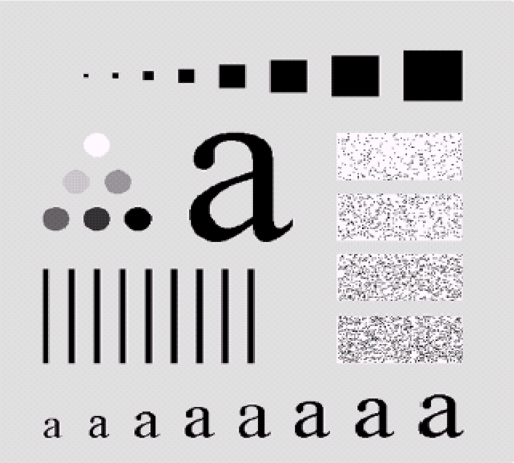
The ideal high pass filter is given as:

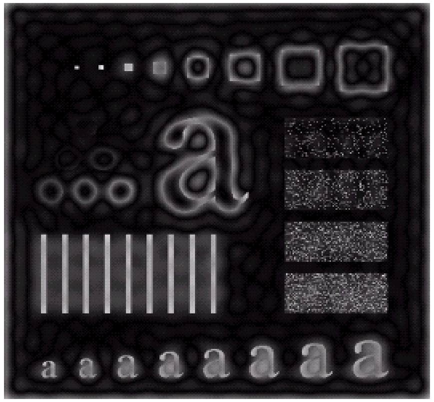
$$H(u,v) = \begin{cases} 0 \text{ if } D(u,v) \le D_0 \\ 1 \text{ if } D(u,v) > D_0 \end{cases}$$

where D_0 is the cut off distance as before

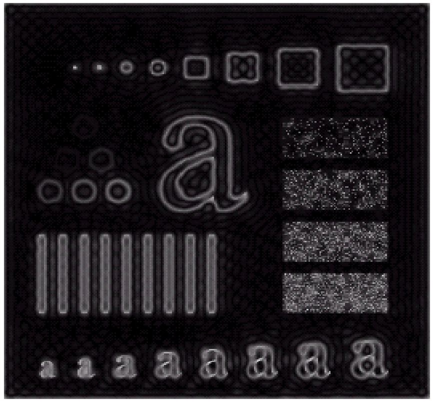


Ideal High Pass Filters (cont…)

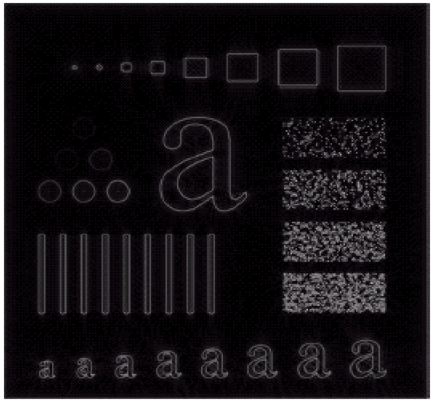




Results of ideal high pass filtering with D_0 = 15



Results of ideal high pass filtering with D_0 = 30



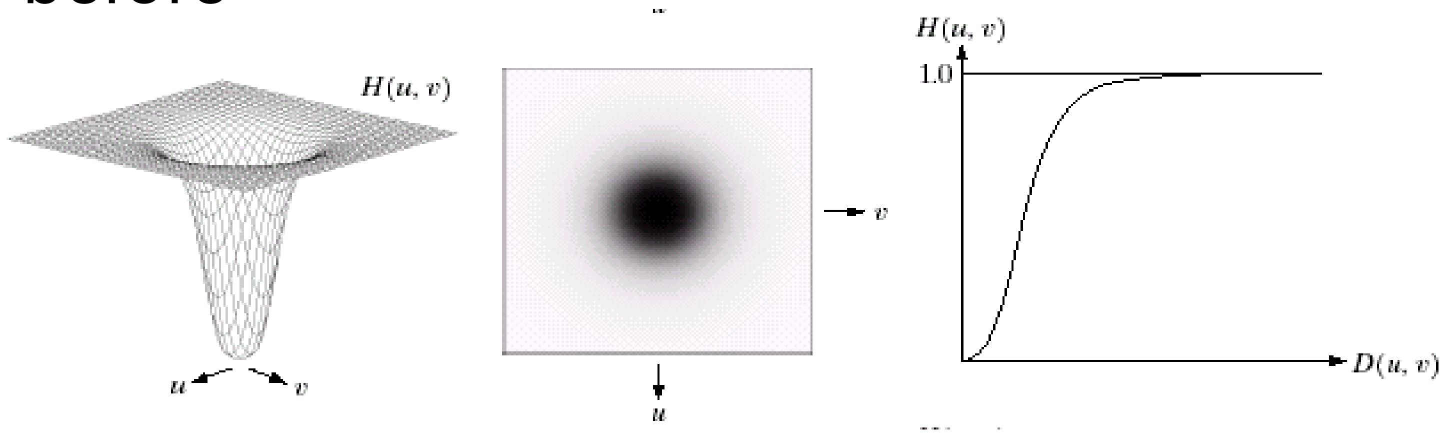
Results of ideal high pass filtering with D_0 = 80

Butterworth High Pass Filters

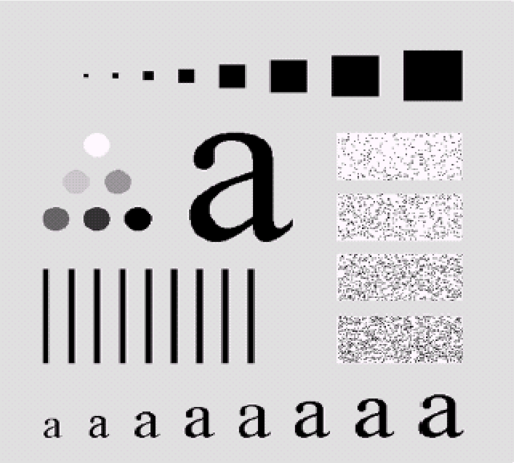
The Butterworth high pass filter is given as:

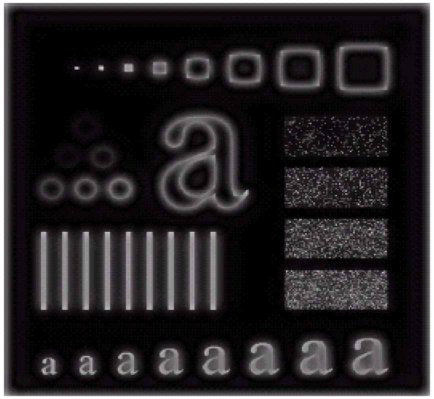
$$H(u,v) = \frac{1}{1+[D_0 / D(u,v)]^{2n}}$$

where n is the order and D_0 is the cut off distance as before

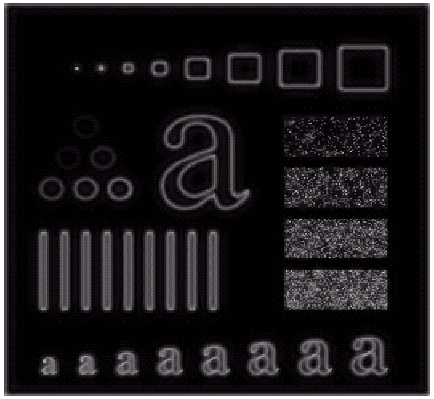


Butterworth High Pass Filters (cont…)

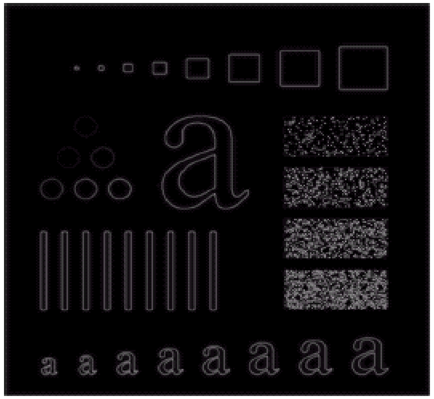




Results of Butterworth high pass filtering of order 2 with D_0 = 15



Results of Butterworth high pass filtering of order 2 with D_0 = 30



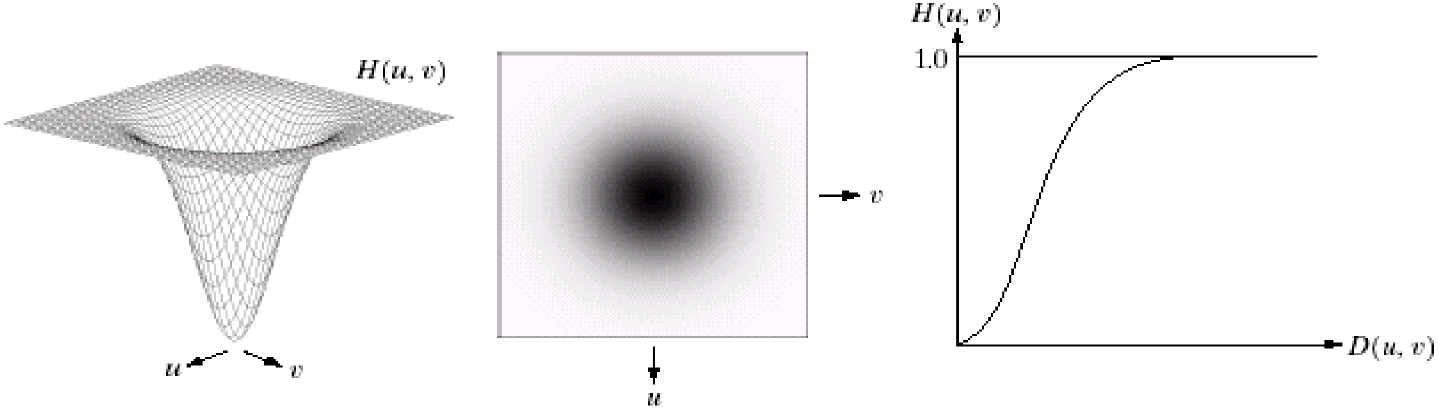
Results of Butterworth high pass filtering of order 2 with D_0 = 80

Gaussian High Pass Filters

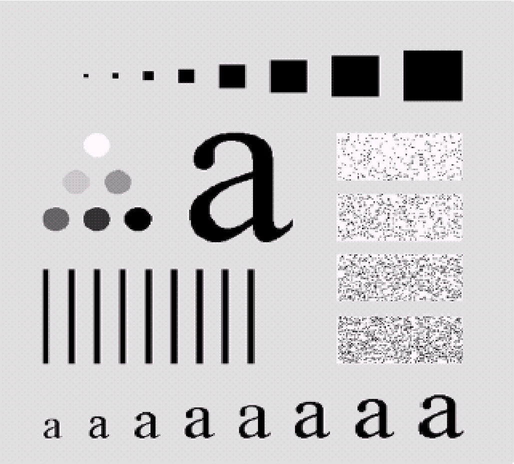
The Gaussian high pass filter is given as:

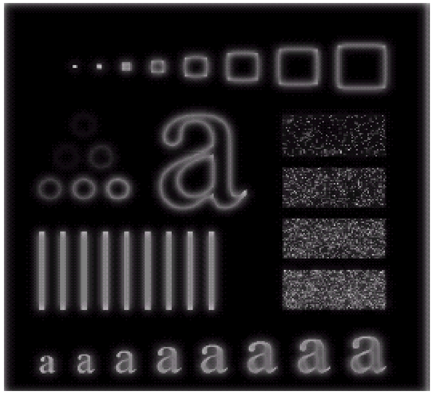
$$H(u,v) = 1 - e^{-D^2(u,v)/2{D_0}^2}$$

where D_0 is the cut off distance as before

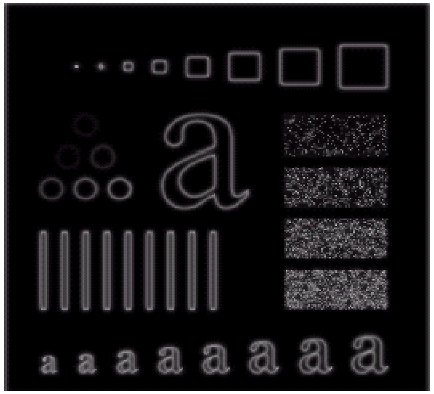


Gaussian High Pass Filters (cont…)

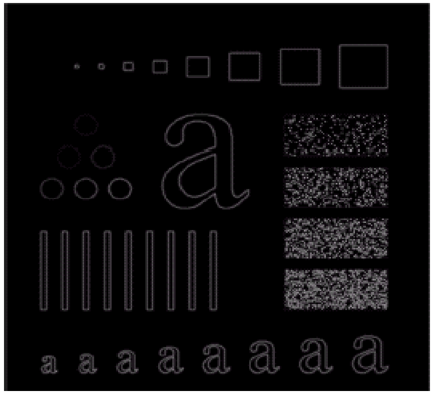




Results of Gaussian high pass filtering with D_0 = 15

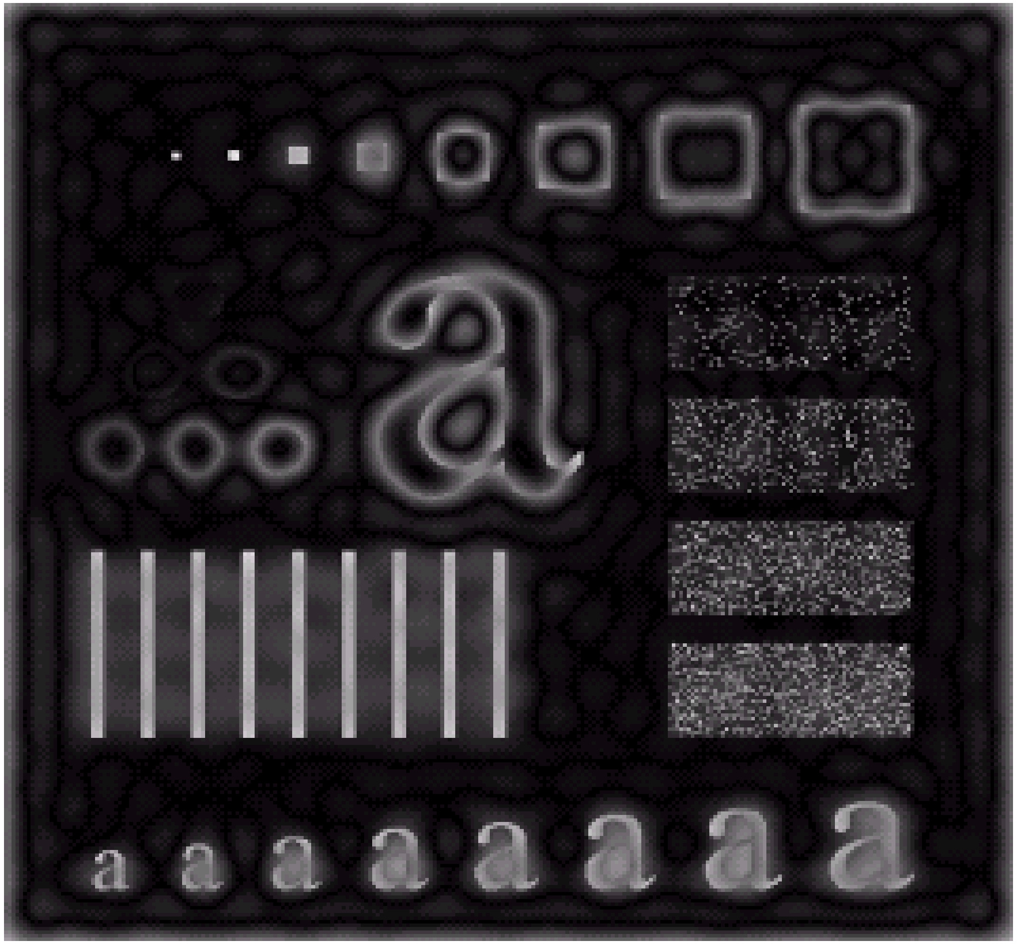


Results of Gaussian high pass filtering with D_0 = 30



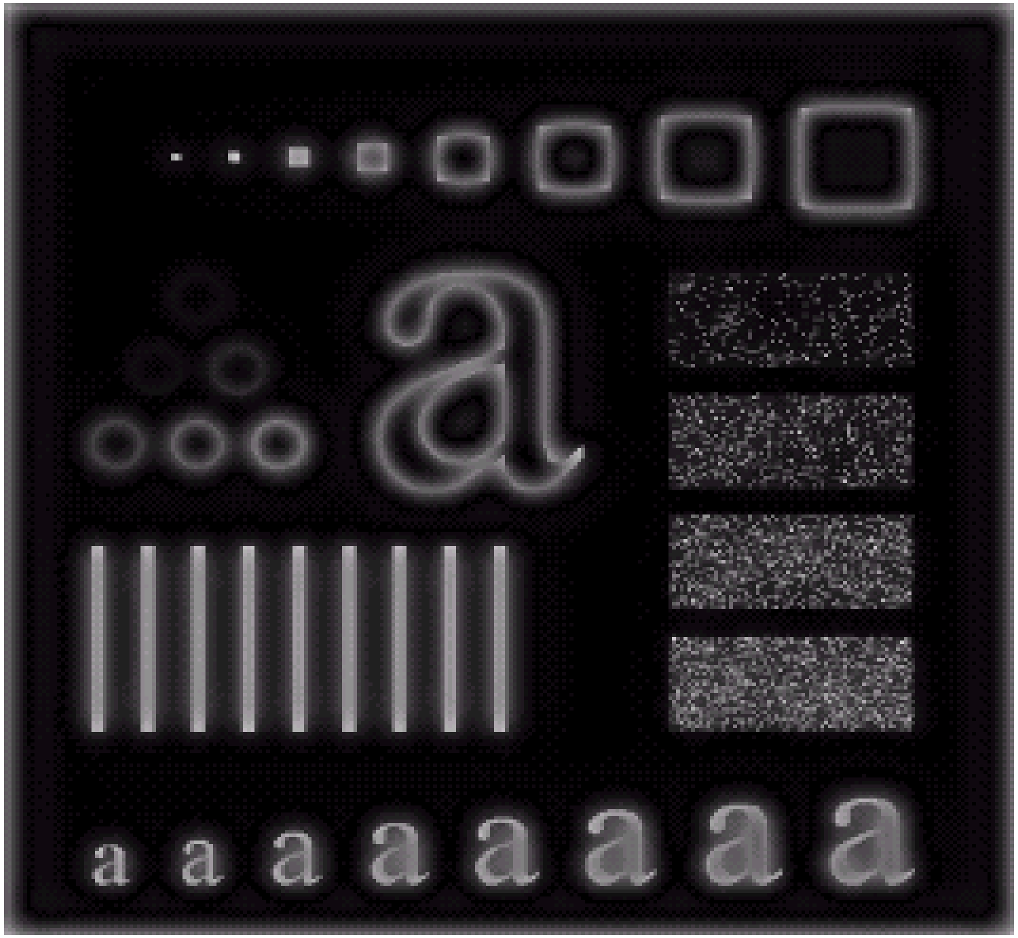
Results of Gaussian high pass filtering with D_0 = 80

Highpass Filter Comparison



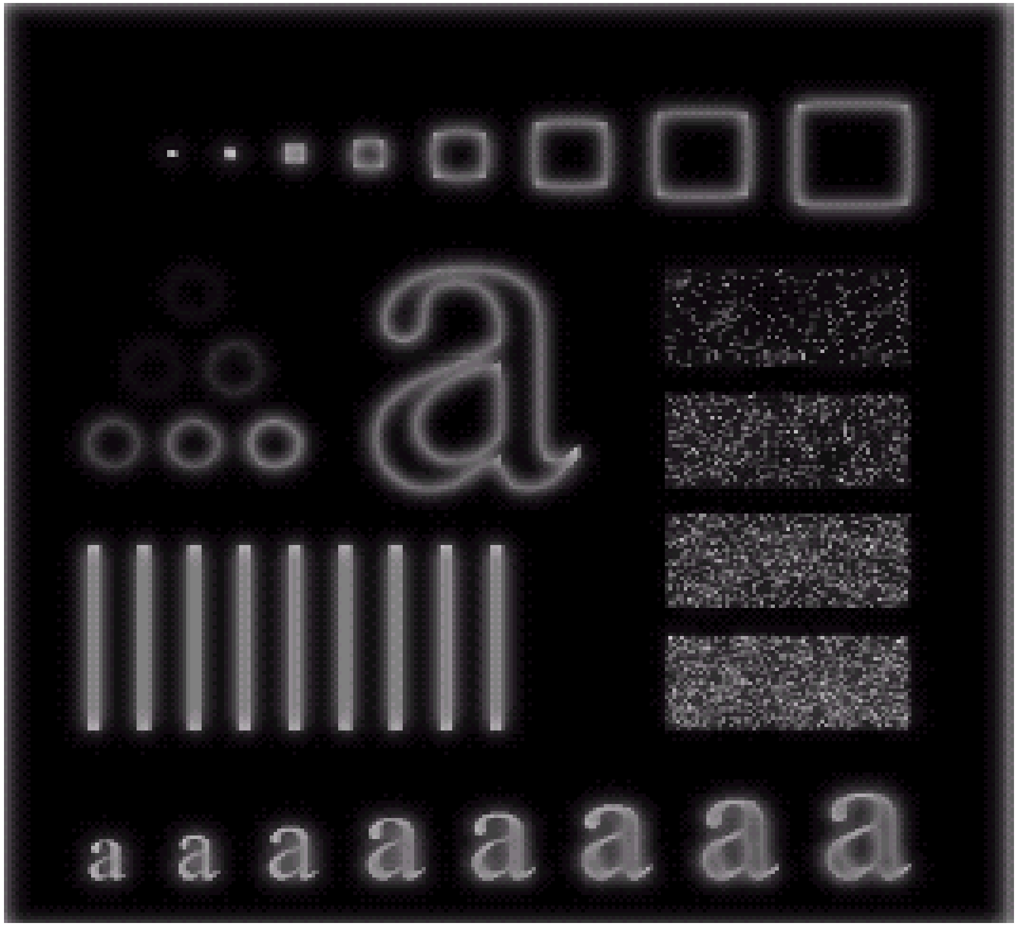
Results of ideal high pass filtering with D_0 = 15

Highpass Filter Comparison



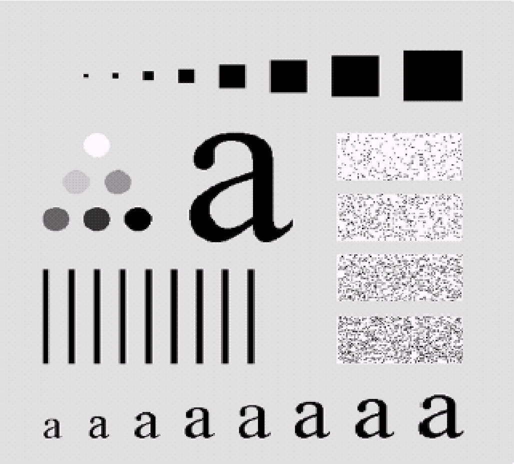
Results of Butterworth high pass filtering of order 2 with D_0 = 15

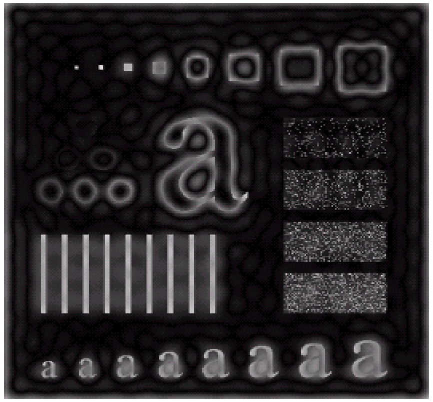
Highpass Filter Comparison



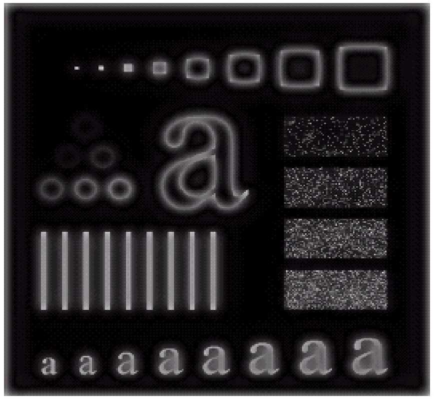
Results of Gaussian high pass filtering with D_0 = 15

Highpass Filter Comparison

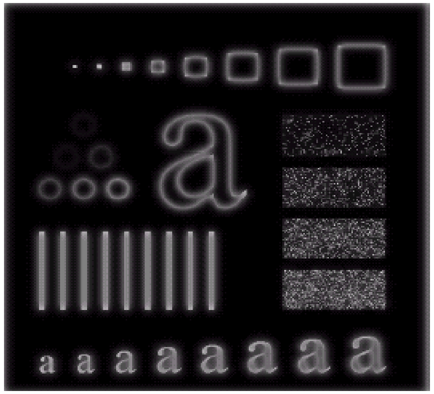




Results of ideal high pass filtering with D_0 = 15

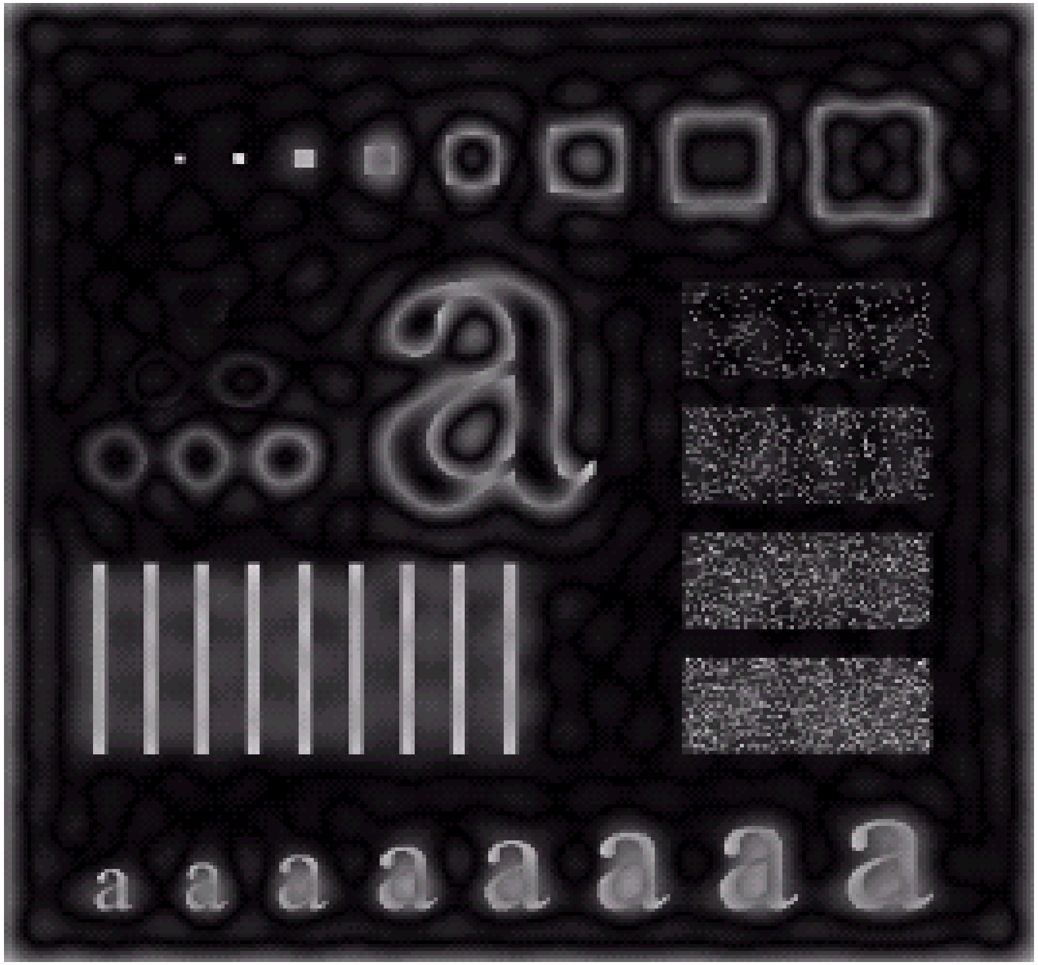


Results of Butterworth high pass filtering of order 2 with D_0 = 15



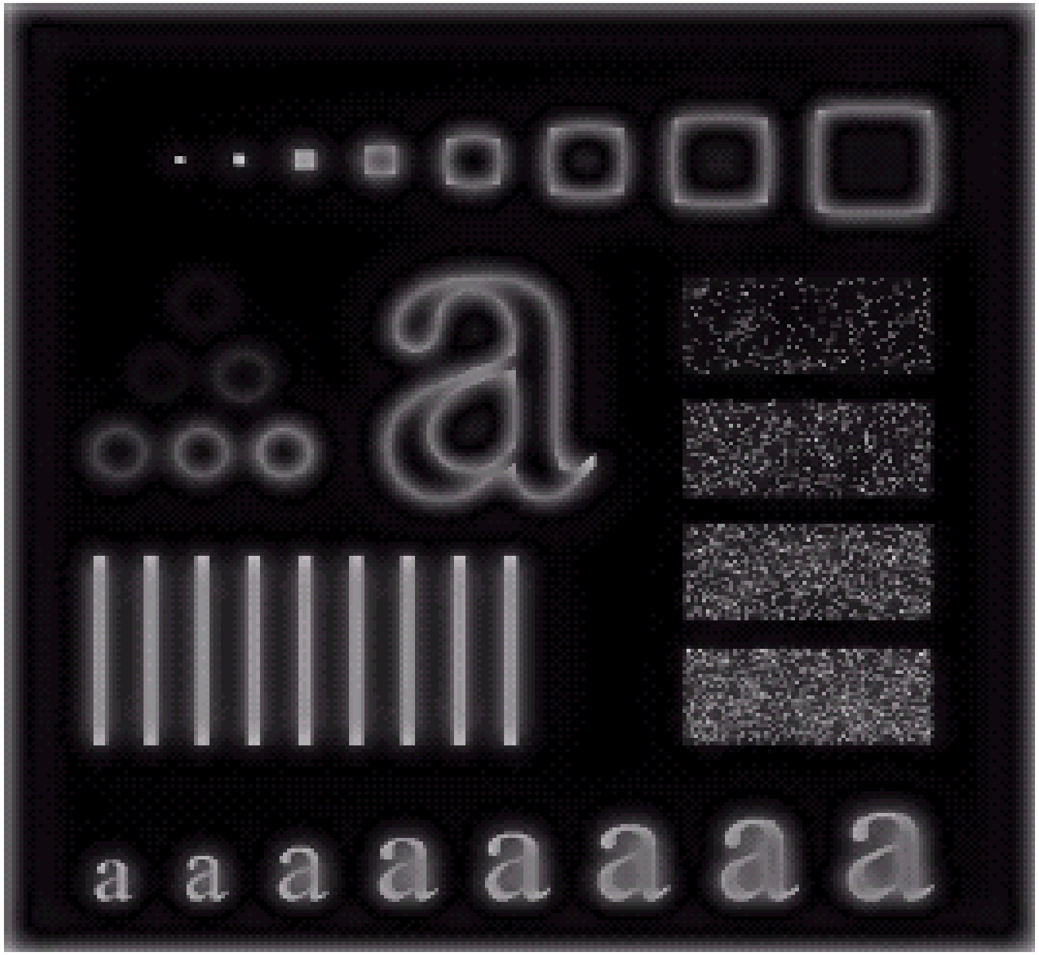
Results of Gaussian high pass filtering with D_0 = 15

Highpass Filter Comparison



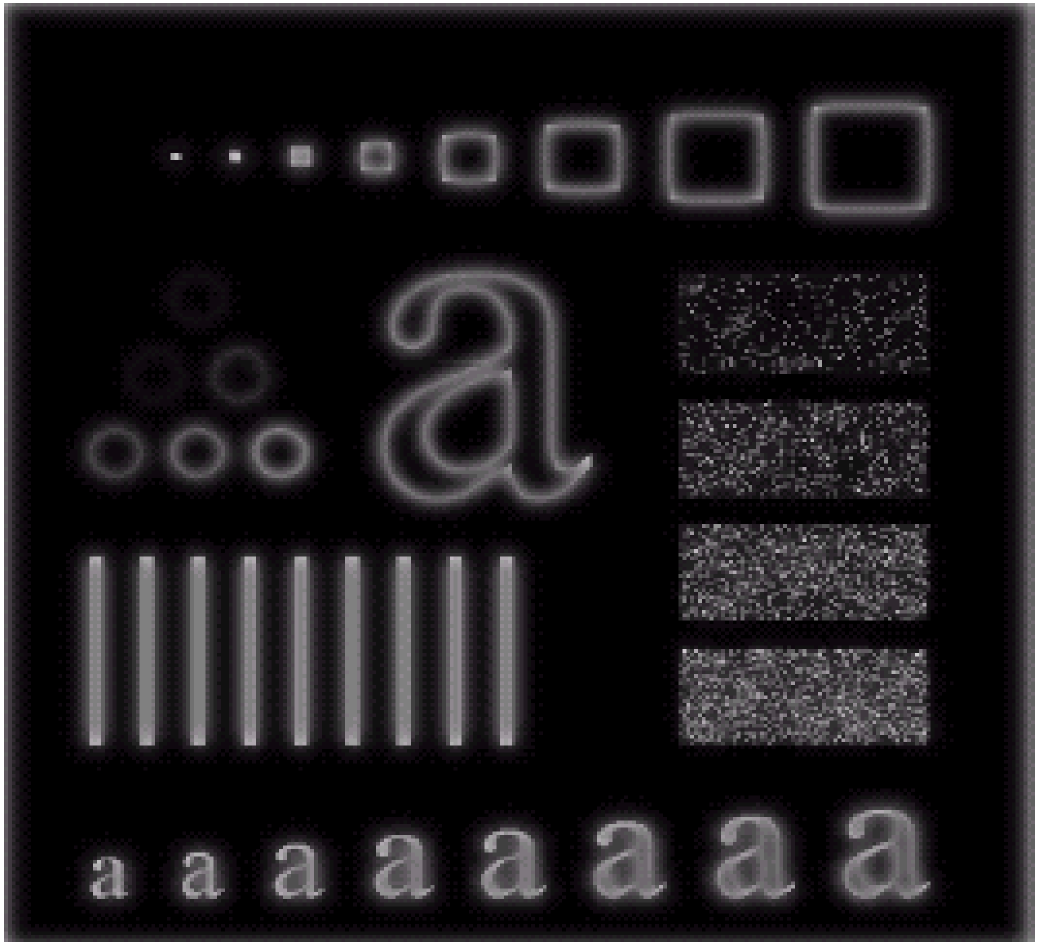
Results of ideal high pass filtering with D_0 = 15

Highpass Filter Comparison



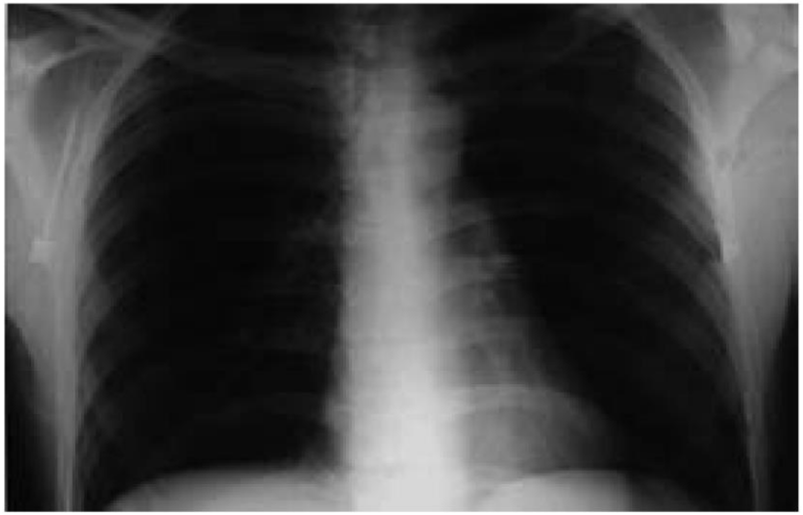
Results of Butterworth high pass filtering of order 2 with D_0 = 15

Highpass Filter Comparison

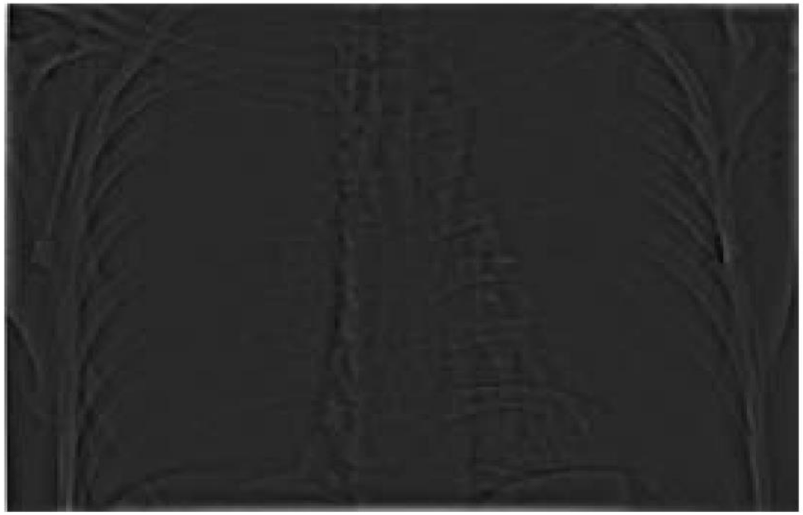


Results of Gaussian high pass filtering with D_0 = 15

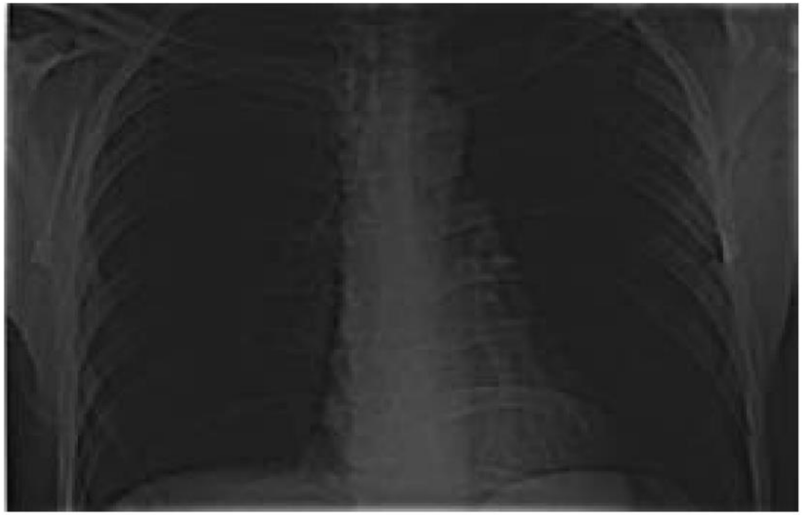
Highpass Filtering Example



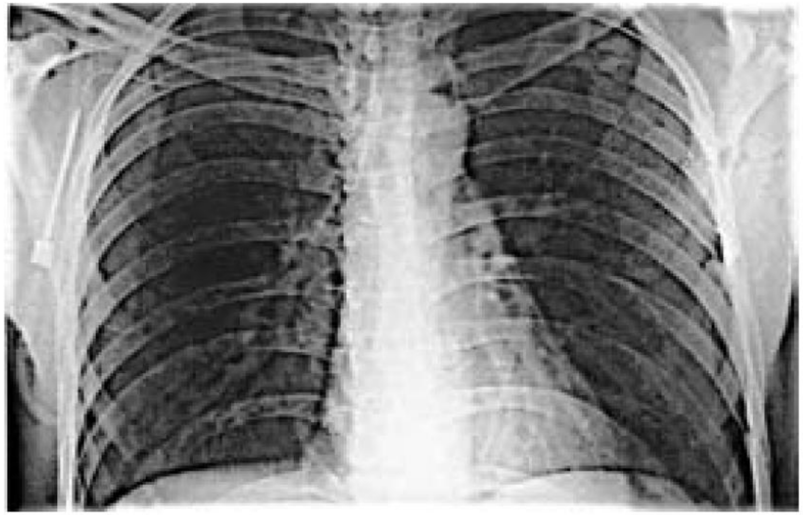
Original image



Highpass filtering result



High frequency emphasis result



After histogram equalisation

DFT EXAMPLE

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

4x4 Image

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

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

lowest frequency component

highest frequency component

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

DISCRETE COSINE TRANSFORM (DCT)

Discrete Cosine Transform (DCT)

- FT uses sines and cosines waves to represent a signal
- DCT uses only cosine waves
- Hence, DCT is purely real

1-D DCT

The 1-D DCT of a sequence f(x), 0<=x<=N-1 is:

$$C(u) = a(u)\sum_{x=0}^{N-1} f(x)\cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

$$u = 0, 1, \mathrm{K}, N-1$$

$$a(u) = \begin{cases} \sqrt{\dfrac{1}{N}} & u = 0 \\ \sqrt{\dfrac{2}{N}} & u = 1, \mathrm{K}, N-1 \end{cases}$$

The inverse transformation is:

$$f(x) = \sum_{u=0}^{N-1} a(u)\ C(u)\ \cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

2-D DCT

The 2-D DCT is:

$$C(u,v) = a(u)a(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

For u, v = 0,1,2,…,N-1

The inverse transformation is:

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} a(u)a(v)C(u,v)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

For x, y = 0,1,2,…,N-1

Matrix Method

- The NxN Cosine Transform Matrix is defined as:

$$C\,(u, v) = \begin{cases} \sqrt{\dfrac{1}{N}} & u = 0,\ 0 \le v \le N-1 \\ \sqrt{\dfrac{2}{N}}\cos\left[\dfrac{\pi\,(2v+1)\,u}{2N}\right] & 1 \le u \le N-1,\ 0 \le v \le N-1 \end{cases}$$

	v			
u	0.5	0.5	0.5	0.5
	0.653	0.2705	-0.2705	-0.653
	0.5	-0.5	-0.5	0.5
	0.2705	-0.653	0.653	-0.2705

- The cosine transform matrix is real and orthogonal but not symmetric.
- The 2D DCT of an image is generated using

  F= C.f.C’

  Where C.C’=I

  C’ = transpose of C

Find the DCT of the following sequence f(x)={1,2,4,7}

- F = C*f

=

0.5	0.5	0.5	0.5
0.653	0.2705	-0.2705	-0.653
0.5	-0.5	-0.5	0.5
0.2705	-0.653	0.653	-0.2705

$$\begin{bmatrix} 1 \\ 2 \\ 4 \\ 7 \end{bmatrix}$$

$$= \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}$$

Find the DCT of the given 4x4 image

2	4	4	2
4	6	8	3
2	8	10	4
3	8	6	2

F = C.f.C’

Non-Sinusoidal Transforms

- means not sine-shaped, or not sinusoidal (sine-like). Non-sinusoidal waveforms are not periodic and don't have a constant amplitude.
- TYPES
  - Hadamard
  - Walsh
  - Haar

Introduction to Hadamard Transform

- **Discrete Hadamard Transform –** Transform Matrix (Kernel Matrix)

$$H_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$H_4 = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

$$H_8 = \frac{1}{2^{\frac{3}{2}}} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{pmatrix}$$

Introduction to Hadamard Transform

$$X[n] = \frac{1}{N}[H(N).\ x(n)]$$

$$X[n] = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ 2 \\ 0 \\ 3 \end{bmatrix}$$

$$X[n] = \frac{1}{4}\begin{bmatrix} 6 \\ -4 \\ 0 \\ 2 \end{bmatrix}$$

$$X[n] = \begin{bmatrix} 1.5 \\ -1 \\ 0 \\ 0.5 \end{bmatrix}$$

Inverse Hadamard Transform:

$$x(n) = H(N)\ X[n]$$

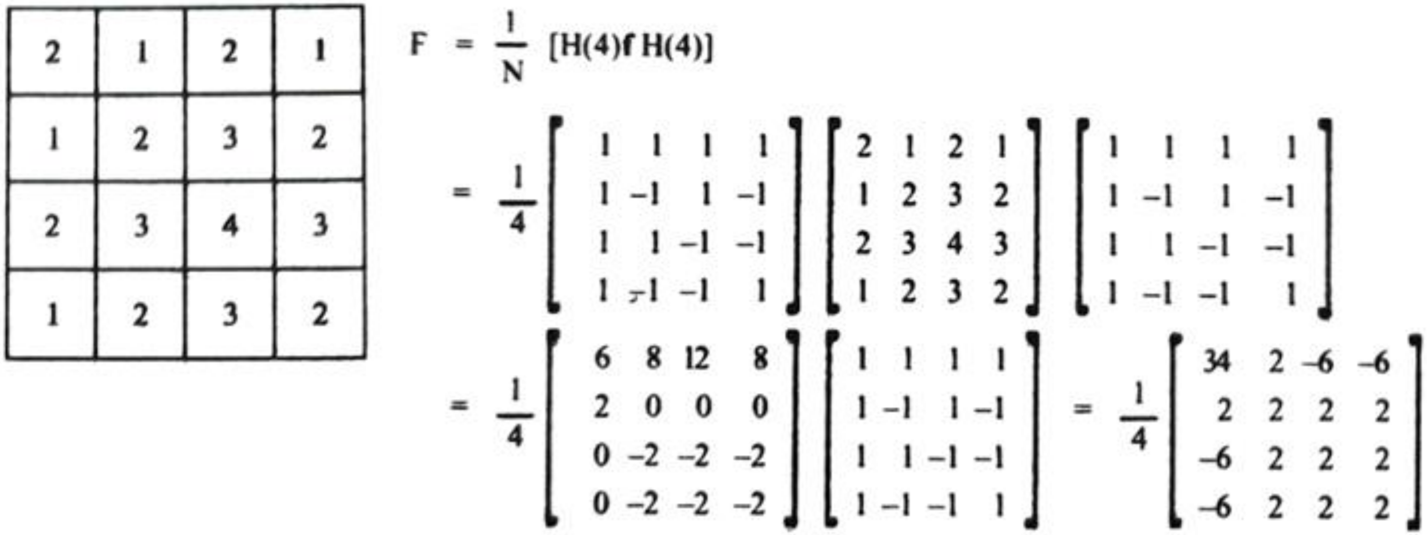
$$x(n) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}\begin{bmatrix} 1.5 \\ -1 \\ 0 \\ 0.5 \end{bmatrix}$$

$$x(n) = \begin{bmatrix} 1 \\ 2 \\ 0 \\ 3 \end{bmatrix}$$

Introduction to Hadamard Transform

- Example
- 1D Transform = $H_4 * f(x)$
- 2D Transform = $H_4 * f(x, y) * H_4^T$

$$H_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad H_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$



2	1	2	1
1	2	3	2
2	3	4	3
1	2	3	2

$$F = \frac{1}{N}\,[H(4) f\, H(4)]$$

$$= \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}\begin{bmatrix} 2 & 1 & 2 & 1 \\ 1 & 2 & 3 & 2 \\ 2 & 3 & 4 & 3 \\ 1 & 2 & 3 & 2 \end{bmatrix}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

$$= \frac{1}{4}\begin{bmatrix} 6 & 8 & 12 & 8 \\ 2 & 0 & 0 & 0 \\ 0 & -2 & -2 & -2 \\ 0 & -2 & -2 & -2 \end{bmatrix}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 34 & 2 & -6 & -6 \\ 2 & 2 & 2 & 2 \\ -6 & 2 & 2 & 2 \\ -6 & 2 & 2 & 2 \end{bmatrix}$$

Introduction to Walsh Transform

- **Discrete Walsh Transform -** What and Why
  - Mathematical Model

$$W(u)=\frac{1}{N}\sum_{x=0}^{N-1} f(x)\left[\prod_{i=0}^{n-1}(-1)^{b_i(x)b_{n-1-i}(u)}\right]$$

  - or

$$W(u)=\frac{1}{N}\sum_{x=0}^{N-1} f(x)(-1)^{\sum_{i=1}^{n-1} b_i(x)b_{n-1-i}(u)}$$

Introduction to Walsh-Hadamard Transform

$$W_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad W_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

$$\begin{array}{cc} & \text{Sign change} \\ \mathrm{W}(8) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} & \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \end{array}$$

Compute the discrete Walsh Transform of the data sequence {1, 2, 0, 3}

N=4, we form a Walsh matrix of size 4x4

$$W(4) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

Generate the Walsh Transform of the given 4x4 image

2	1	2	1
1	2	3	2
2	3	4	3
1	2	3	2

$F = W(4).f.\ W(4)'$

Introduction to Slant Transform

$$S_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad S_4 = \frac{1}{\sqrt{4}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 3/\sqrt{5} & 1/\sqrt{5} & -1/\sqrt{5} & -3/\sqrt{5} \\ 1 & -1 & -1 & 1 \\ 1/\sqrt{5} & -3/\sqrt{5} & 3/\sqrt{5} & -1/\sqrt{5} \end{bmatrix}$$

- 1D Transform = S_4 * f(x)
- 2D Transform = S_4 * f(x, y) * S_4^T

Introduction to Slant Transform

- **1D Discrete Slant Transform**
  - ○ Find DST of the given x(n) = [1, 2, 2, 1]
  - ○ DST, X(u) = S(4) . x(n)'

$$\text{DST} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 3/\sqrt{5} & 1/\sqrt{5} & -1/\sqrt{5} & -3/\sqrt{5} \\ 1 & -1 & -1 & 1 \\ 1/\sqrt{5} & -3/\sqrt{5} & 3/\sqrt{5} & -1/\sqrt{5} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix}$$

Introduction to Slant Transform

2	2	2	1
2	4	4	2
2	4	4	2
2	2	2	2

4x4 Image

$$\mathrm{DST} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 3/\sqrt{5} & 1/\sqrt{5} & -1/\sqrt{5} & -3/\sqrt{5} \\ 1 & -1 & -1 & 1 \\ 1/\sqrt{5} & -3/\sqrt{5} & 3/\sqrt{5} & -1/\sqrt{5} \end{bmatrix} \begin{bmatrix} 2 & 2 & 2 & 1 \\ 2 & 4 & 4 & 2 \\ 2 & 4 & 4 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 3/\sqrt{5} & 1/\sqrt{5} & -1/\sqrt{5} & -3/\sqrt{5} \\ 1 & -1 & -1 & 1 \\ 1/\sqrt{5} & -3/\sqrt{5} & 3/\sqrt{5} & -1/\sqrt{5} \end{bmatrix}$$

Introduction to Haar Transform

$$Haar_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad Haar_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

Introduction to Haar Transform

- **Example:** 1D Example
  - f(x) = {1, 2, 2, 1}
  - Find Haar Transform Coefficients

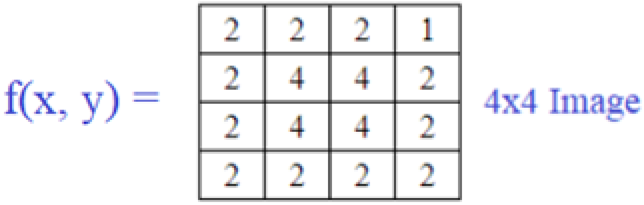
1D Transform = $H_4 * f(x)$

2D Transform = $H_4 * f(x, y) * H_4^T$

$$= \frac{1}{\sqrt{4}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{bmatrix}$$

Introduction to Haar Transform

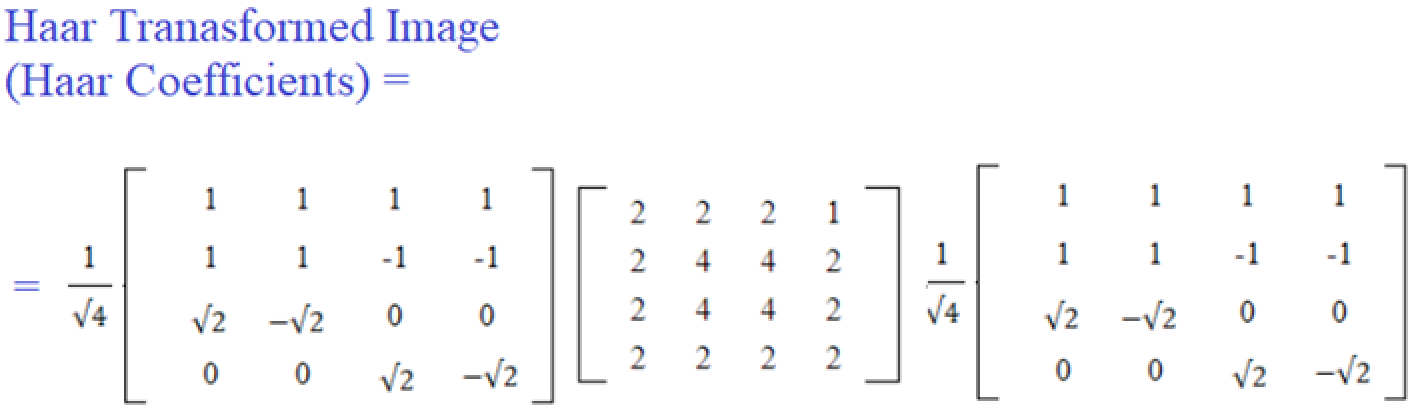
- **Example:** 2D Example
  - Apply Haar Transform on given 4x4 image



f(x, y) =

2	2	2	1
2	4	4	2
2	4	4	2
2	2	2	2

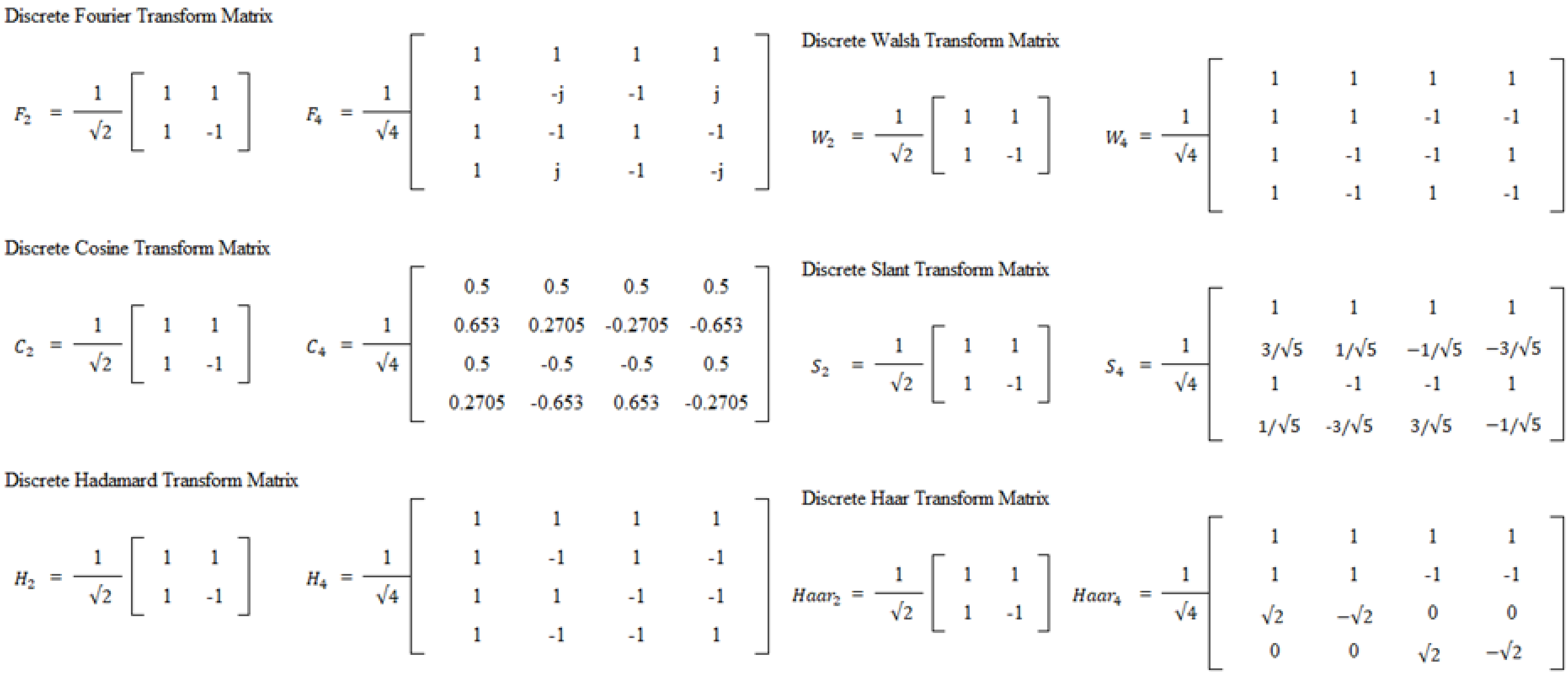
4x4 Image



Haar Tranasformed Image
(Haar Coefficients) =

$$= \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}\begin{bmatrix} 2 & 2 & 2 & 1 \\ 2 & 4 & 4 & 2 \\ 2 & 4 & 4 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix}\frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

Introduction to Discrete Image Transform: Summary



Discrete Fourier Transform Matrix

$$F_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad F_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

Discrete Walsh Transform Matrix

$$W_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad W_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

Discrete Cosine Transform Matrix

$$C_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad C_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.653 & 0.2705 & -0.2705 & -0.653 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.2705 & -0.653 & 0.653 & -0.2705 \end{bmatrix}$$

Discrete Slant Transform Matrix

$$S_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad S_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 3/\sqrt{5} & 1/\sqrt{5} & -1/\sqrt{5} & -3/\sqrt{5} \\ 1 & -1 & -1 & 1 \\ 1/\sqrt{5} & -3/\sqrt{5} & 3/\sqrt{5} & -1/\sqrt{5} \end{bmatrix}$$

Discrete Hadamard Transform Matrix

$$H_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad H_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

Discrete Haar Transform Matrix

$$Haar_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad Haar_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$