EC861 - Image Processing and Computer Vision
Assignement - 03.

1) Various properties of 2D Discrete Fourier Transform are :-

a) **Separability**

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

Here, kernel is separable.

$$\therefore e^{-j2\pi\left(\frac{ux+vy}{N}\right)} = e^{-j2\pi\left(\frac{ux}{N}\right)} \cdot e^{-j2\pi\left(\frac{vy}{N}\right)}$$

F(u,v) can be written as :

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

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

$$\therefore F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} e^{-j2\pi\left(\frac{ux}{N}\right)} \cdot F(x,v).$$

This implies that 2D DFT F(u,v) can be obtained by:

i) Taking 1D DFT of every row of image $f(x,y)$

ii) Taking 1D DFT of every column of $F(x,v)$

b) **Periodicity**.

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

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

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

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

$$\Rightarrow F(u+N, v+N) = F(u,v).$$

$\therefore$ DFT and IDFT are periodic with period N.

c) Translation

We know: $\mathcal{F} f(x,y) = \cancel{\frac{1}{N}} \cdot \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cdot e^{-j2\pi\left(\frac{ux+vy}{N}\right)}$

Multiply both sides by $e^{j2\pi\left(\frac{u_0x+v_0y}{N}\right)}$, we get,

$$\mathcal{F} f(x,y) \cdot e^{j2\pi\left(\frac{u_0x+v_0y}{N}\right)} = \cancel{\frac{1}{N}} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cdot e^{-j2\pi\left(\frac{ux+vy}{N}\right)} \cdot e^{j2\pi\left(\frac{u_0x+v_0y}{N}\right)}$$

$$= \cancel{\frac{1}{N}} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cdot e^{\left(-\frac{j2\pi ux}{N} + \frac{j2\pi u_0 x}{N}\right)} \cdot e^{\left(-\frac{j2\pi vx}{N} + \frac{j2\pi v_0 y}{N}\right)}$$

$$= F(u-u_0, v-v_0)$$

~~$\therefore F(u,v)$~~ $\therefore$ If $f(x,y)$ is multiplied by an exponential, the original fourier transform $F(u,v)$ gets shifted in frequency by $F(u-u_0, v-v_0)$

d) Rotation

Let $x = r\cos\theta$ and $y = \sin\theta$

$u = w\cos\phi$ and $v = \sin\phi$

Then we have,

$f(x,y) = f(r,\theta)$ — in spatial domain

$F(u,v) = F(w,\phi)$ — in frequency domain.

Now rotated image is $f(r, \theta + \theta_0)$ and

$$f(r, \theta + \theta_0) \longleftrightarrow F(\omega, \phi + \theta_0)$$

$\therefore$ If $f(x,y)$ is rotated by angle θ_0, then the $F(u,v)$ also rotates the same angle.

e) Scaling

we know : $f(x,y) \xrightarrow{DFT.} F(u,v)$

$\therefore$ DFT of function $f(ax, by)$ is given by:

$$DFT\{f(ax, by)\} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(ax, by) \cdot e^{-j2\pi\left(\frac{ux+vy}{N}\right)}$$

By multiplying and dividing the power of exponential term $-j2\pi\left(\frac{ux}{N}\right)$ by a and $-j2\pi\left(\frac{vy}{N}\right)$ by b, we get,

$$DFT\{f(ax, by)\} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(ax, by) \cdot e^{-j\frac{2\pi}{N} \cdot ux\left(\frac{a}{a}\right)} \cdot e^{-j\frac{2\pi}{N} \cdot vy\left(\frac{b}{b}\right)}$$

$$= \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(ax, by) \cdot e^{-j\frac{2\pi}{N} x \cdot a\left(\frac{u}{a}\right)} \cdot e^{-j\frac{2\pi}{N} \cdot yb \cdot \left(\frac{v}{b}\right)}$$

$$DFT\{f(ax, by)\} = \frac{1}{|ab|} \cdot F(u/a, y/b)$$

~~4~~ ~~2 a) $x[m,n] = $ np.array([[1,0],[2,1]])~~

3. a) input = np.array([[1,2,3], [4,5,6], [7,8,9]])

filter = np.array([[1,2,1], [0,0,0], [-1,-2,-1]])

Here, $M_1 = 3$ and $N_1 = 3$

$M_2 = 3$ and $N_2 = 3$.

Length of 2D circular convolution :

$= \max(M_1, M_2) \times \max(N_1, N_2)$

$= 3 \times 3$

Here, $h[m,n] = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$

$\therefore H_0 = \begin{bmatrix} -1 & -1 & -2 \\ -2 & -1 & -1 \\ -1 & -2 & -1 \end{bmatrix}, \quad H_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad H_2 = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 1 & 1 \\ 1 & 2 & 1 \end{bmatrix}$

Matrix $H = \begin{bmatrix} H_0 & H_2 & H_1 \\ H_1 & H_0 & H_2 \\ H_2 & H_1 & H_0 \end{bmatrix}$

$$g = Hf = \begin{bmatrix} H_0 & H_2 & H_1 \\ H_1 & H_0 & H_2 \\ H_2 & H_1 & H_0 \end{bmatrix} \begin{bmatrix} (1 \ 2 \ 3)^T \\ (4 \ 5 \ 6)^T \\ (7 \ 8 \ 9)^T \end{bmatrix}$$

$$= \begin{bmatrix} -1 & -1 & -2 & 1 & 1 & 2 & 0 & 0 & 0 \\ -2 & -1 & -1 & 2 & 1 & 1 & 0 & 0 & 0 \\ -1 & -2 & -1 & 1 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & -2 & 1 & 1 & 2 \\ 0 & 0 & 0 & -2 & -1 & -1 & 2 & 1 & 1 \\ 0 & 0 & 0 & -1 & -2 & -1 & 1 & 2 & 1 \\ 1 & 1 & 2 & 0 & 0 & 0 & -1 & -1 & -2 \\ 2 & 1 & 1 & 0 & 0 & 0 & -2 & -1 & -1 \\ 1 & 2 & 1 & 0 & 0 & 0 & -1 & -2 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{bmatrix}$$

$$= \begin{bmatrix} -12 & -12 & -12 \\ -12 & -12 & -12 \\ 24 & 24 & 24 \end{bmatrix}$$

2. b) i) $x(m,n)$ = np.array([[1,0],[2,1]])

$$W_N = e^{-j\frac{2\pi}{N}}, \text{ here, } N = 2.$$

$$\therefore W_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$X[k,l] = W_2 \cdot x[m,n] \cdot W_2^T$$

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

ii) $x(m,n)$ = np.array([[1,2,3,4],[5,6,7,8],[9,10,11,12],[13,14,15,16]])

Here, $N = 4$.

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

$$X[k,l] = W_4 \cdot x[m,n] \cdot W_4^T$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

$$= \begin{bmatrix} 136 & -8+8j & -8 & -8-8j \\ -32+32j & 0 & 0 & 0 \\ -32 & 0 & 0 & 0 \\ -32+32j & 0 & 0 & 0 \end{bmatrix}$$