Lecture 7:

Discrete Fourier Transform:

$$e^{j\theta} = \cos\theta + j\sin\theta$$

Definition:

The 2D DFT of a $M \times N$ image $g = (g(k,l))_{k,l}$, where $0 \leq k \leq M-1$, $0 \leq l \leq N-1$ is defined as:

$$\hat{g}(m,n) = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} g(k,l)\, e^{-j2\pi\left(\frac{km}{M} + \frac{ln}{N}\right)}$$

$$\hat{g} = UgU$$

Remark: The inverse of DFT is given by:

$$g(p,q) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \hat{g}(m,n)\, e^{j2\pi\left(\frac{pm}{M} + \frac{qn}{N}\right)}$$

(no $\frac{1}{Mn}$!)   DFT of g (no -ve sign)

Why is DFT useful in imaging:

1. DFT of convolution:

Recall: $$g * w(n,m) = \sum_{n'=0}^{N-1} \sum_{m'=0}^{N-1} g(n-n', m-m')\, w(n', m')$$

$\left(g, m \in M_{N \times M}(\mathbb{R}) \right)$

Then, the DFT of $g * w$ = $MN\, DFT(g)\, DFT(w)$

$\therefore$ DFT of convolution can be reduced to simple multiplication!

Recall: Shift-invariant image transformation = 2D convolution.

$\therefore$ Easy computation/manipulation of shift-invariant transf. after DFT!!

2. Average value of image

Average value of $g = \bar{g} = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} g(k,l) = \underbrace{\frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} g(k,l)\, e^{-j2\pi(0)}}_{\hat{g}(0,0)}$

3. DFT of a rotated image

Consider a $N \times N$ image g.

Then: $\hat{g}(m,n) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} g(k,l)\, e^{-j2\pi\left(\frac{km+ln}{N}\right)}$

Write k and l in polar coordinates:

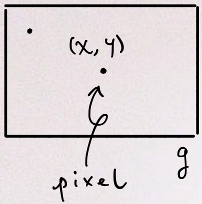
$k \equiv r\cos\theta$; $l \equiv r\sin\theta$

Similarly, write $m \equiv \omega\cos\phi$; $n = \omega\sin\phi$.

Note that: $km + ln = r\omega(\cos\theta\cos\phi + \sin\theta\sin\phi) = r\omega\cos(\theta - \phi)$.

Denote $\mathcal{P}(g) = \{(r,\theta) : (r\cos\theta, r\sin\theta) \text{ is a pixel of } g\}$

(Polar coordinate set of g)



If $\begin{cases} x = r\cos\theta \\ y = r\sin\theta \end{cases}$, then $(r, \theta) \in \mathcal{D}(g)$.

Then: $$\underbrace{\hat{g}(m,n) = \hat{g}(\omega, \phi)}_{\text{Identify } \hat{g}(m,n) \text{ with } \hat{g}(\omega,\phi)} = \frac{1}{N^2}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} \underbrace{g(r,\theta)}_{\text{Identify } g(k,l) \text{ with } g(r,\theta)} e^{-j2\pi\left(\frac{r\omega\cos(\theta-\phi)}{N}\right)}$$

Consider a rotated image $\tilde{g}(r,\theta) = g(r, \theta+\theta_0)$ where θ is defined between $-\theta_0$ to $\pi/2 - \theta_0$.

$\therefore$ image g is rotated clockwisely by θ_0.

DFT of $\tilde{g}$ is:

$$\hat{\tilde{g}}(\omega, \phi) = \frac{1}{N^2}\sum_{(r,\theta)\in\mathcal{D}(\tilde{g})} \underset{g(r,\, \underbrace{\theta+\theta_0}_{\tilde{\theta}})}{\underset{\shortparallel}{\tilde{g}(r,\theta)}}\, e^{-j2\pi\left(\frac{r\omega\cos(\theta-\phi)}{N}\right)} = \frac{1}{N^2}\sum_{(r,\tilde{\theta})\in\mathcal{D}(g)} g(r,\tilde{\theta})\, e^{-j2\pi\left(\frac{r\omega\cos(\tilde{\theta}-\theta_0-\phi)}{N}\right)}$$

$\therefore$ $\hat{\tilde{g}}(\omega, \phi) = \hat{g}(\omega, \phi+\theta_0)$. (ϕ is also defined between $-\theta_0$ to $\pi/2 - \theta_0$)

4. DFT of a shifted image

Let $g = (g(k', l'))$ be a $N \times N$ image, where the indices are taken as:

$-k_0 \le k' \le N-1-k_0$ and $-l_0 \le l' \le N-1-l_0$

Let $\tilde{g}$ be shifted image of g defined as:

$\tilde{g}(k, l) = g(k-k_0, l-l_0)$ where $0 \le k \le N-1$, $0 \le l \le N-1$

Then:

$$\hat{\tilde{g}}(m,n) = \frac{1}{N^2}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} g(k-k_0, l-l_0)\, e^{-j2\pi\left(\frac{km+ln}{N}\right)}$$

$$= \underbrace{\frac{1}{N^2}\sum_{k'=-k_0}^{N-1-k_0}\sum_{l'=-l_0}^{N-1-l_0} g(k', l')\, e^{-j2\pi\left(\frac{k'm+l'n}{N}\right)}}_{\hat{g}(m,n)}\, e^{-j2\pi\left(\frac{k_0 m + l_0 n}{N}\right)}$$

$$\therefore \quad \hat{\tilde{g}}(m,n) = \hat{g}(m,n)\, e^{-j2\pi\left(\frac{k_0 m + l_0 n}{N}\right)}$$

<u>Remark:</u> $\hat{g}(m-m_0, n-n_0) = \mathrm{DFT}\left(g \times e^{j2\pi\left(\frac{m_0 k + n_0 l}{N}\right)}\right)$ with carefully chosen indices!

Mathematics of JPEG

Consider a $N \times N$ image f. Extend f to a $2M \times 2N$ image $\tilde{f}$, whose indices are taken from $[-M, M-1]$ and $[-N, N-1]$.

Define $f(k,l)$ for $-M \leq k \leq M-1$ and $-N \leq l \leq N-1$ such that

$$f(-k-1,-l-1) = f(k,l) \quad \Big\} \text{ Reflection about } (-1/2,-1/2)$$

$$\left.\begin{aligned} f(-k-1,l) &= f(k,l) \\ f(k,l-1) &= f(k,l) \end{aligned}\right\} \text{ Reflection about the axis } k=-1/2 \text{ and } l=-1/2$$

Example:

$l = -1/2$

$$\begin{array}{ccc|ccc|l}
9 & 8 & 7 & 7 & 8 & 9 & k=-3 \\
6 & 5 & 4 & 4 & 5 & 6 & k=-2 \\
3 & 2 & 1 & 1 & 2 & 3 & k=-1 \\
\hline
3 & 2 & 1 & 1 & 2 & 3 & k=0 \\
6 & 5 & 4 & 4 & 5 & 6 & k=1 \\
9 & 8 & 7 & 7 & 8 & 9 & k=2 \\
l=-3 & l=-2 & l=-1 & l=0 & l=1 & l=2 &
\end{array}$$

$k = -1/2$

$f(-2,-2)$

$=$

$f(1,1)$

Reflection about $(-1/2,-1/2)$.

$f(-1,1)$

$=$

$f(0,1)$

Reflection about the axis $k = -1/2$.

Make the extension as a reflection about $(0,0)$, the axis $k=0$ and the axis $l=0$.

Done by shifting the image by $(1/2, 1/2)$

After shifting

						k
9	8	7	7	8	9	$\frac{1}{2}+(-3)$
6	5	4	4	5	6	$\frac{1}{2}+(-2)$
3	2	1	1	2	3	$\frac{1}{2}+(-1)$
3	2	1	1	2	3	$\frac{1}{2}+0$
6	5	4	4	5	6	$\frac{1}{2}+1$
9	8	7	7	8	9	$\frac{1}{2}+2$
$\frac{1}{2}+-3$	$\frac{1}{2}+-2$	$\frac{1}{2}+-1$	$\frac{1}{2}+0$	$\frac{1}{2}+1$	$\frac{1}{2}+2$	
l						

Now, we compute the DFT of (shifted) $\tilde{f}$:

$$F(m,n) = \frac{1}{(2M)(2N)} \sum_{k=-M}^{M-1} \sum_{l=-N}^{N-1} f(k,l) e^{-j\frac{2\pi}{2M}m(k+\frac{1}{2})} e^{-j\frac{2\pi}{2N}n(l+\frac{1}{2})}$$

$$= \frac{1}{4MN} \sum_{k=-M}^{M-1} \sum_{l=-N}^{N-1} f(k,l) e^{-j(\frac{\pi}{M}m(k+\frac{1}{2})+\frac{\pi}{N}n(l+\frac{1}{2}))}$$

$$= \frac{1}{4MN} \Big(\underbrace{\sum_{k=-M}^{-1} \sum_{l=-N}^{-1}}_{A_1} + \underbrace{\sum_{k=-M}^{-1} \sum_{l=0}^{N-1}}_{A_2} + \underbrace{\sum_{k=0}^{M-1} \sum_{l=-N}^{-1}}_{A_3} + \underbrace{\sum_{k=0}^{M-1} \sum_{l=0}^{N-1}}_{A_4} \Big)$$

$$f(k,l) e^{-j(\frac{\pi}{M}m(k+\frac{1}{2})+\frac{\pi}{N}n(l+\frac{1}{2}))}$$

After some messy simplication, we can get:

$$A_1 + A_2 + A_3 + A_4 = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} f(k,l) \cos\left[\frac{m\pi}{M}\left(k+\frac{1}{2}\right)\right] \cos\left[\frac{n\pi}{N}\left(l+\frac{1}{2}\right)\right]$$

Definition: (Even symmetric discrete cosine transform [EDCT])

Let f be a $M \times N$ image, whose indices are taken as $0 \leq k \leq M-1$ and $0 \leq l \leq N-1$. The **even symmetric discrete cosine transform (EDCT)** of f is given by:

$$\hat{f}_{ec}(m,n) = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} f(k,l) \cos\left[\frac{m\pi}{M}\left(k+\frac{1}{2}\right)\right] \cos\left[\frac{n\pi}{N}\left(l+\frac{1}{2}\right)\right]$$

with $0 \leq m \leq M-1, 0 \leq n \leq N-1$

Remark:
- Smart idea to get a decomposition consisting only of cosine function (by reflection and shifting!)
- Can be formulated in matrix form
- Again, it is a separable image transformation.

- The inverse of EDCT can be explicitly computed. More specifically, the **inverse EDCT** is defined as:

$$f(k,l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} C(m)C(n)\hat{f}_{ec}(m,n) \cos\frac{\pi m(2k+1)}{2M} \cos\frac{\pi n(2l+1)}{2N} \qquad (**)$$

where $C(0) = 1, C(m) = C(n) = 2$ for $m, n \neq 0$

Also involving cosine functions only!

- Formula (**) can be expressed as matrix multiplication:

$$f = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \hat{f}_{ec}(m,n) \vec{T}_m \vec{T'}_n^T$$

elementary images under EDCT!

where: $\vec{T}_m = \begin{pmatrix} T_m(0) \\ T_m(1) \\ \vdots \\ T_m(M-1) \end{pmatrix}, \vec{T'}_n = \begin{pmatrix} T'_n(0) \\ T'_n(1) \\ \vdots \\ T'_n(N-1) \end{pmatrix}$ with $T_m(k) = C(m) \cos\frac{\pi m(2k+1)}{2M}$

This is what JPEG does!!

and $T'_n(k) = C(n) \cos\frac{\pi n(2k+1)}{2N}$.

Note:

(Spatial domain) $I * g$ (Linear filtering: Linear combination of neighborhood pixel values)

$$\Big\downarrow \text{DFT}$$

(Frequency domain) $MN\,\hat{I} \underbrace{\odot}_{\text{pixel-wise multiplication}} \hat{g}$ (Modifying the Fourier coefficients by multiplication)

Image enhancement in the frequency domain:

Goal: 1. Remove $\underbrace{\text{high-frequency components}}_{\text{noise}}$ (low-pass filter) for image denoising.

2. Remove $\underbrace{\text{low-frequency components}}_{\text{non-edge}}$ (high-pass filter) for the extraction of image details.

High/Low frequency components of $\hat{F}$

Let F be a $N \times N$ image, N = even. Let $\hat{F}$ = DFT of F.

$$\therefore \quad \hat{F}(k, l) = \frac{1}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} F(m,n) \, e^{-j 2\pi \cdot \left(\frac{mk + nl}{N}\right)}$$

Fourier coefficients of F at (k, l)

Observe that : for $0 \le k, l \le \frac{N}{2} - 1$

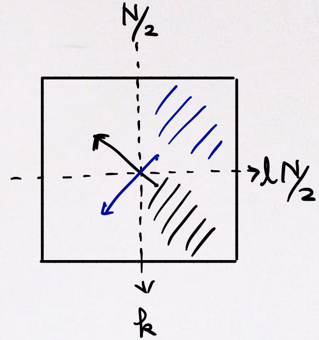
$$\hat{F}\left(\frac{N}{2} + k, \frac{N}{2} + l\right) = \frac{1}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} F(m,n) \, e^{-j \frac{2\pi}{N}\left(m\left(\frac{N}{2} + k\right) + n\left(\frac{N}{2} + l\right)\right)}$$

$$= \frac{1}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} F(m,n) (-1)^{m+n} \, \overline{e^{-j \frac{2\pi}{N}(m(-k) + n(-l))}}$$

$$= \frac{1}{N^2} \overline{\sum_{m=0}^{N-1} \sum_{n=0}^{N-1} F(m,n)\, e^{-j\frac{2\pi}{N}\left(m\left(\frac{N}{2}-k\right)+n\left(\frac{N}{2}-l\right)\right)}}$$

$$= \overline{\hat{F}\left(\frac{N}{2}-k,\ \frac{N}{2}-l\right)}$$

$\therefore$ Computing part of $\hat{F}$ can determine the rest !!



Observation:

1. When k and l are close to $N/2$, $\hat{F}\left(\frac{N}{2}+k, \frac{N}{2}+l\right)$ is associated to $e^{+j\frac{2\pi}{N}\left(\left(\frac{N}{2}+k\right)m+\left(\frac{N}{2}+l\right)n\right)}$

   $\frac{N}{2}+k \approx N$, $\frac{N}{2}+l \approx N$

   $\approx e^{j\frac{2\pi}{N}(k'm+l'n)}$ where $(k',l') \approx (0,0)$

   $= \cos\left(\frac{2\pi}{N}(k'm+l'n)\right) + i\sin\left(\frac{2\pi}{N}(k'm+l'n)\right)$

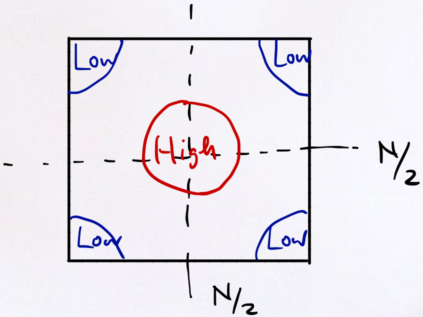
   Low-frequency if $(k,l) \approx (0,0)$

   $\therefore$ Fourier coefficients at the bottom right are associated to low frequency components!

2. Similarly, we can check that Fourier coefficients at the 4 corners are associated to low frequency components.

3. Fourier coefficients in the middle are associated to high-frequency components

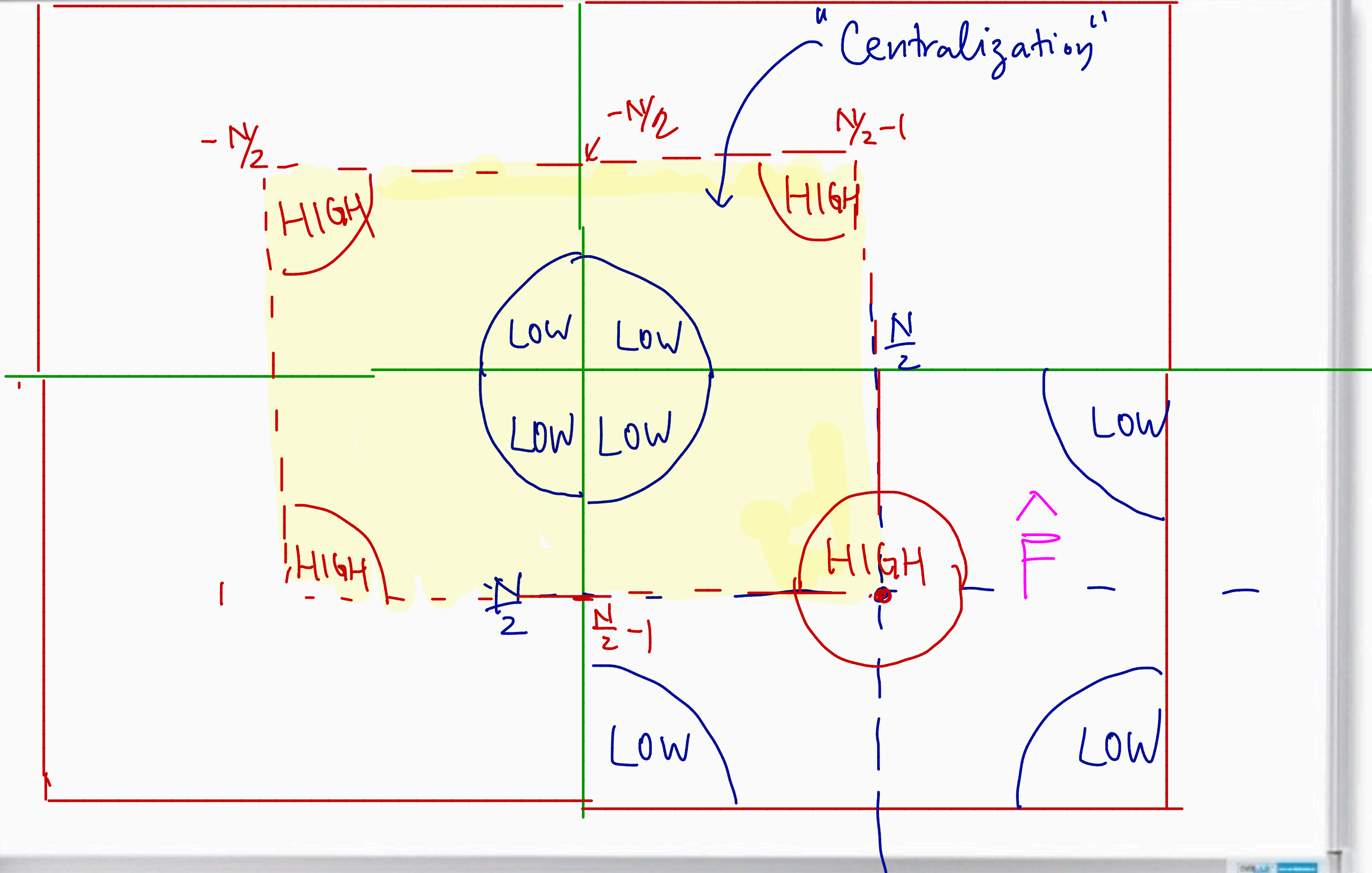
$$e^{j\frac{2\pi}{N}\left(\frac{N}{2}m+\frac{N}{2}n\right)} = e^{j\pi(m+n)} = (-1)^{m+n}$$



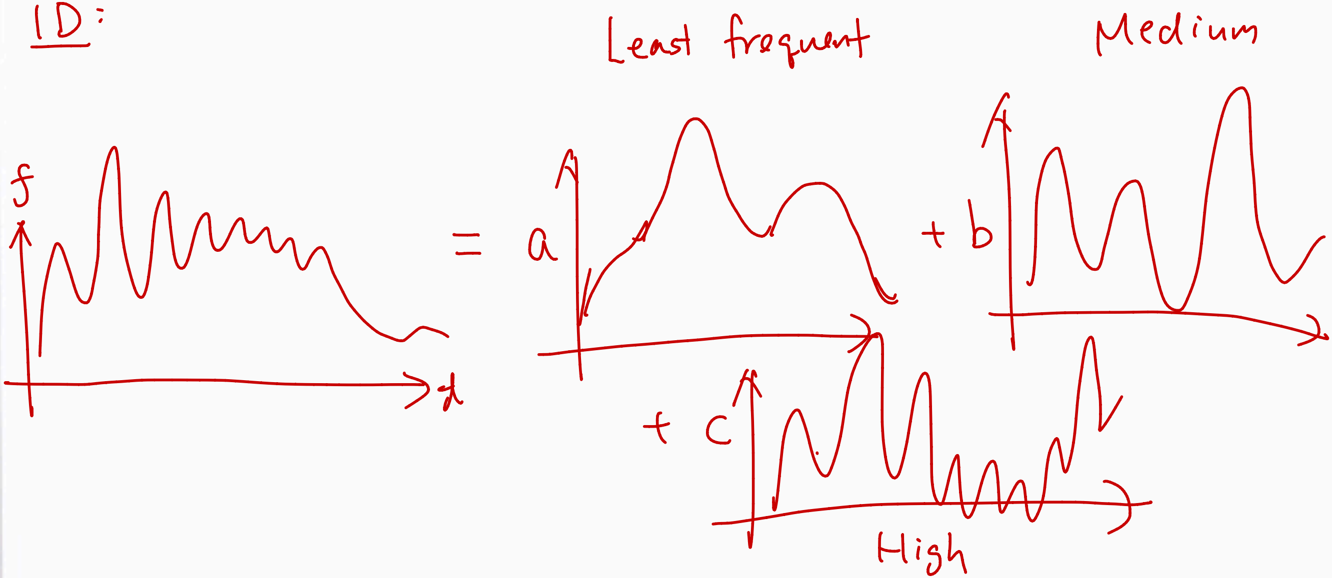
!!

$\therefore$ High-pass filtering $=$ Remove coefficients at 4 corners

Low-pass filtering $=$ Remove coefficients at the center


"Centralization"
-N/2
-N/2
N/2-1
HIGH
HIGH
LOW
LOW
LOW
LOW
N/2
HIGH
HIGH
LOW
F̂
N/2
N/2-1
LOW
LOW

ID:



To remove noise, truncate c (let c = 0)