24.09.2021

Digital Image Processing (CSE/ECE 478)

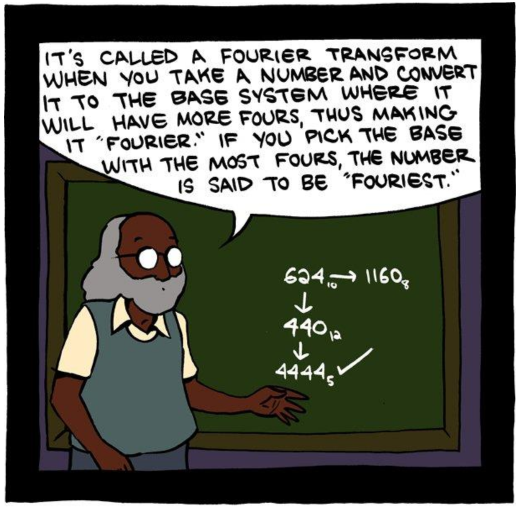
Lecture-10: Frequency Domain Processing

Ravi Kiran and Sudipta Banerjee

Center for Visual Information Technology (CVIT), IIIT Hyderabad

Fourier Transform

Approximate **non-periodic signals** with complex sinusoids



Teaching math was way more fun after tenure.

Announcements

- Mini Quiz 3 TODAY
- Quiz 1 TOMORROW

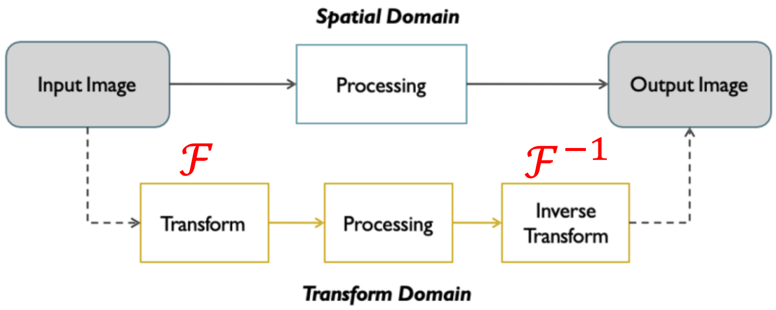
Fourier Transform and Inverse Fourier Transform

- Fourier Transform

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt \qquad F(\omega) = \mathcal{F}[f(t)]$$

- Inverse Fourier Transform

$$f(t) = \int_{\omega=-\infty}^{\omega=\infty} \frac{1}{2\pi} F(\omega)e^{i\omega t}d\omega \qquad f(t) = \mathcal{F}^{-1}[F(\omega)]$$



Intuition for FT

- *Fourier Transform* essentially measures the strength of presence of a particular frequency within a signal
- Sweep over a frequency range, and quantify how dominant is each particular frequency component in original signal

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$$

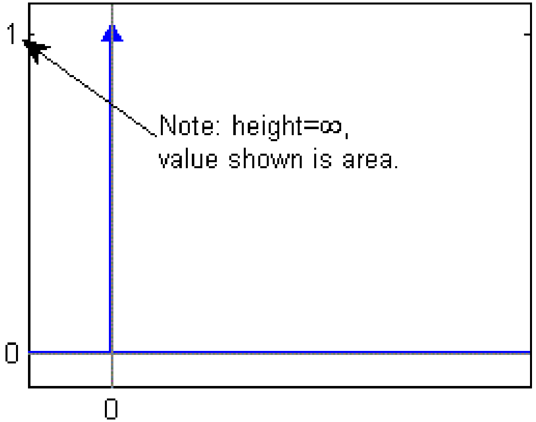
$$F(\omega) = \mathcal{F}[f(t)]$$

Unit Impulse Function – Some properties

$\delta(t) = 0, \; for \; t \neq 0.$

$\delta(0) = +\infty$

$$\int_{-\infty}^{\infty} \delta(t)dt = 1$$



$$\int_a^b \delta(t)dt = \begin{cases} 1, & a < 0 < b \\ 0, & \text{otherwise} \end{cases}$$

Integral property

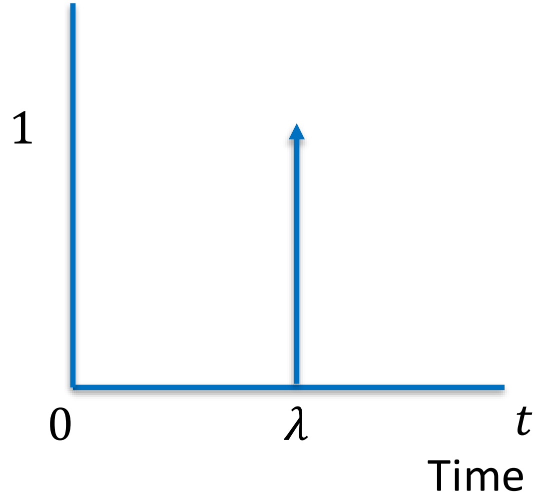
$$\begin{aligned} \int_a^b \delta(t) \cdot f(t)dt &= \int_a^b \delta(t) \cdot f(0)dt \\ &= f(0) \cdot \int_a^b \delta(t)dt \\ &= \begin{cases} f(0), & a < 0 < b \\ 0, & \text{otherwise} \end{cases} \end{aligned}$$

Sifting property

http://lpsa.swarthmore.edu/BackGround/ImpulseFunc/ImpFunc.html

Shifted Unit Impulse Function – Some properties

$$\delta(t-\lambda) = +\infty \; at \; t = \lambda$$
$$\delta(t-\lambda) = 0 \; at \; t \neq \lambda$$



$$\int_a^b \delta(t-\lambda)dt = \begin{cases} 1, & a < \lambda < b \\ 0, & otherwise \end{cases}$$

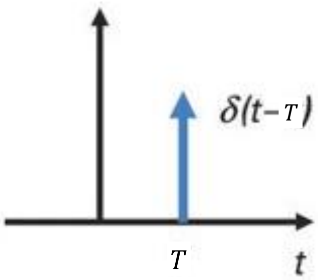
Integral property

$$\int_a^b \delta(t-\lambda) \cdot f(t)dt = \begin{cases} f(\lambda), & a < \lambda < b \\ 0, & otherwise \end{cases}$$

Sifting property

http://lpsa.swarthmore.edu/BackGround/ImpulseFunc/ImpFunc.html

FT of time-shifted impulse



$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt$$

$$= e^{-i\omega T}$$

$$\int_{a}^{b} \delta(t-T)\, x(t)dt = x(T), \qquad a < T < b$$

$$= 0 \; otherwise$$

Fourier Transform of a time-shifted impulse is a complex exponential

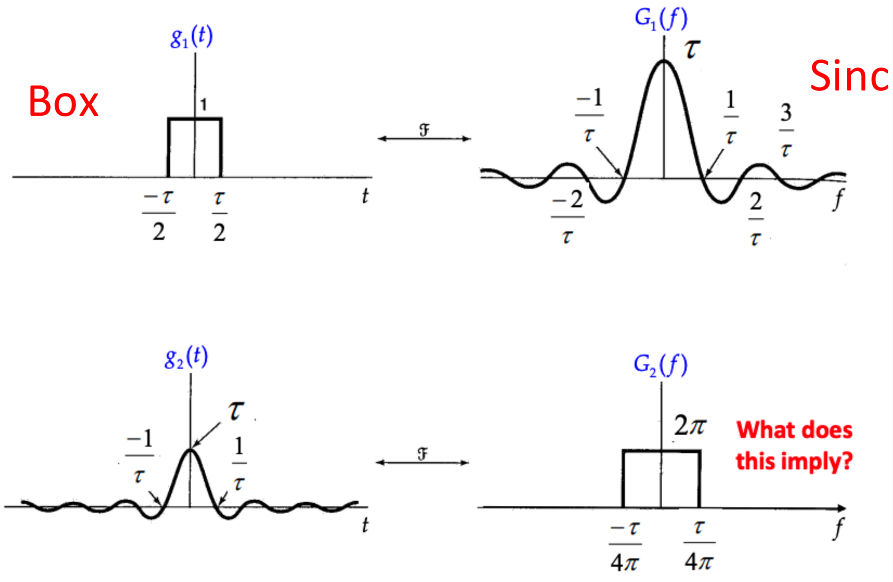
Duality property of FT

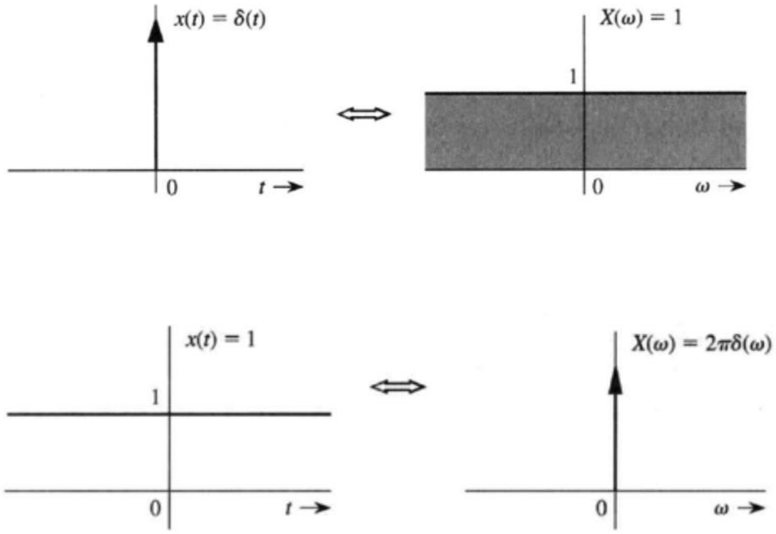
$$\mathcal{F}[f(t)] = F(\omega)$$

$$\Longrightarrow \mathcal{F}[F(t)] = 2\pi\, f(-\omega)$$

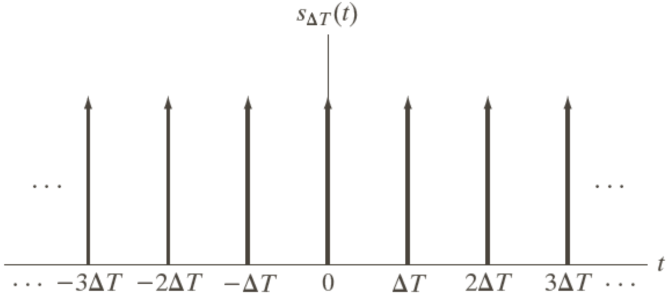
$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$$

$$f(t) = \frac{1}{2\pi} \int_{\omega=-\infty}^{\omega=\infty} F(\omega) e^{i\omega t} d\omega$$





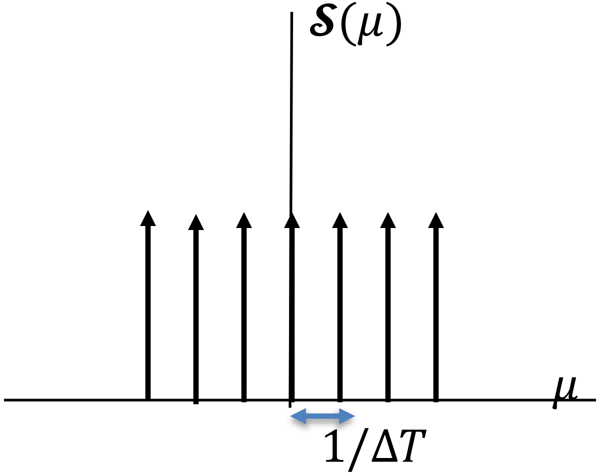
FT of impulse train(G&W, 4.2.4)



$$s_{\Delta T}(t) = \sum_{n=-\infty}^{\infty} \delta\,(t - n\Delta T)$$

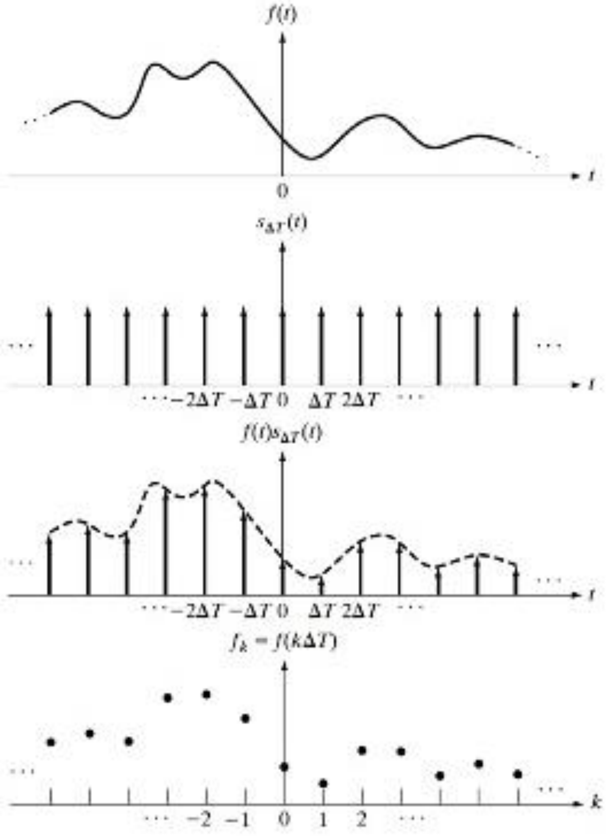
$$\mathcal{F}(s_{\Delta T}(t)) \;= \sum_{n=-\infty}^{\infty} \delta\,(\mu - n/\Delta T)$$

$\mathcal{S}(\mu)$



Sampling = f(t) x Impulse Train

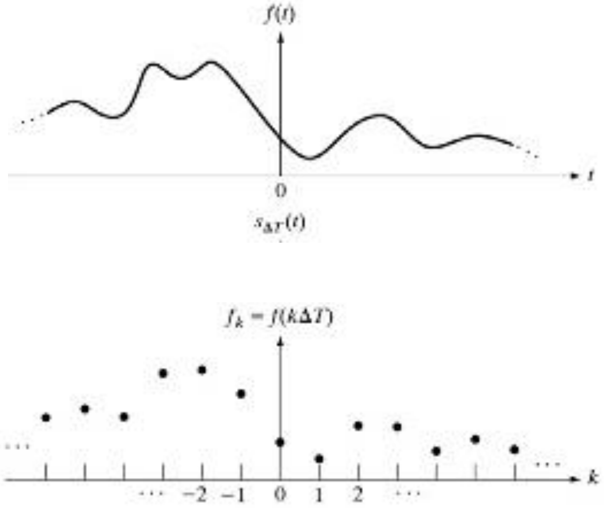
Sampled signal: Scaled version of time shifted impulses

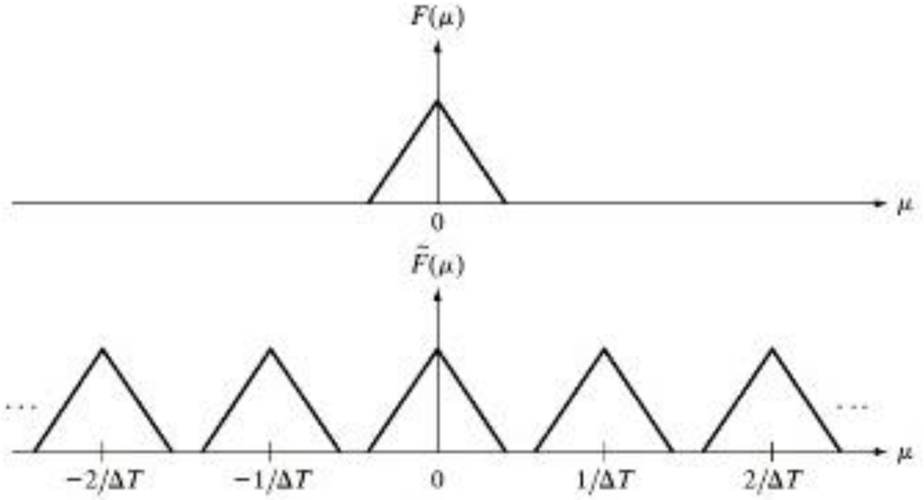


$$s_{\Delta T}(t) = \sum_{n=-\infty}^{n=\infty} \delta(t - n\Delta T)$$

$$\tilde{f}(t) = \sum_{n=-\infty}^{n=\infty} f_n \, \delta(t - n\Delta T)$$

FT of sampled function (G&W 4.2.4)



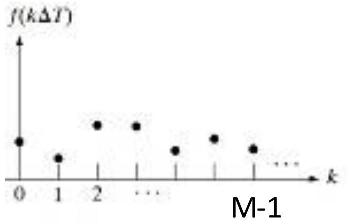


- Continuous
- Periodic (copies of f(t)'s FT)

$$\tilde{f}(t) = \sum_{n=-\infty}^{n=\infty} f_n\, \delta(t - n\Delta T)$$

$$\tilde{F}(\mu) = \boxed{\frac{1}{\Delta T}} \sum_{n=-\infty}^{n=\infty} F\left(\mu - \frac{n}{\Delta T}\right)$$

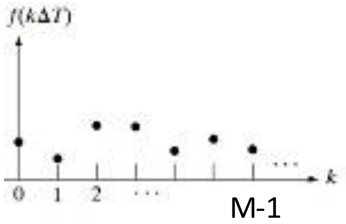
FT of sampled function

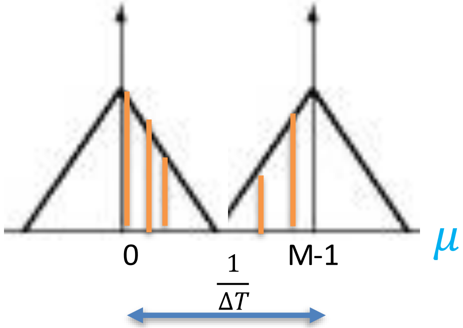


$$\tilde{f}(t) = \sum_{n=0}^{n=(M-1)\Delta T} f_n\, \delta(t - n\Delta T)$$

$$\tilde{F}(\mu) = \sum_{n=0}^{n=(M-1)} f_n e^{-j2\pi\mu n\Delta T}$$

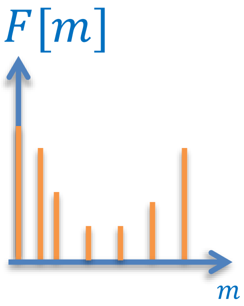
FT of sampled function (G&W 4.4.1)





$$\tilde{F}(\mu) = \sum_{n=0}^{n=(M-1)} f_n e^{-j2\pi\mu n\Delta T}, \mu \in R$$

$$\tilde{f}(t) = \sum_{n=0}^{n=(M-1)\Delta T} f_n\, \delta(t - n\Delta T)$$



- Substituting

$$\mu = \frac{m}{M\Delta T} \qquad m = 0, 1, 2, \ldots, M-1$$

$$F[m] = \sum_{n=0}^{n=(M-1)} f_n\, e^{\frac{-j2\pi mn}{M}}, \mathrm{m} = 0,1, \ldots (\mathrm{M}-1)$$

NOTE: No direct dependence on ΔT

DFT and IDFT

$$F[m] = \sum_{n=0}^{n=(M-1)} f_n \, e^{\frac{-j2\pi nm}{M}} , \mathrm{m} = 0,1, \ldots (\mathrm{M}-1)$$

$$f_n = \frac{1}{M} \sum_{m=0}^{m=(M-1)} F_m \, e^{\frac{j2\pi nm}{M}} , \mathrm{n} = 0,1, \ldots (\mathrm{M}-1)$$

$F[m]$

- A complex value
- Represents amplitude, phase of function f[.]'s content at angular frequency 2πm/M

DFT: Record of ‘energy’ portion at various frequency bands present in input function f[.]

$$Magnitude = \sqrt[2]{Re\{F[m]\}^2 + Im\{F[m]\}^2}$$

$$Phase = tan^{-1} \frac{Im\{F[m]\}}{Re\{F[m]\}}$$

DFT (in practice)

$$F[m] = \sum_{n=0}^{n=(M-1)} f[n]\, e^{\frac{-j2\pi mn}{M}}, \mathrm{m} = 0,1, \ldots (\mathrm{M}-1)$$

$$F = P.f$$

$M \times 1 \quad M \times M \quad M \times 1$

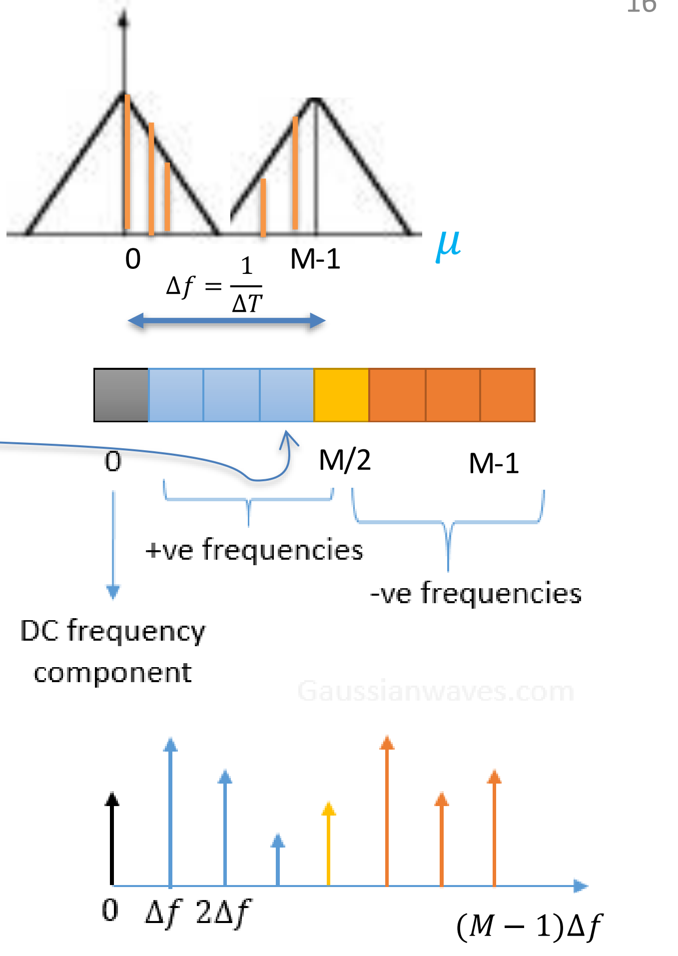
$$[P] = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & \rho & \rho^2 & \rho^3 & \rho^4 \\ 1 & \rho^2 & \rho^4 & \rho^6 & \rho^8 \\ 1 & \rho^3 & \rho^6 & \rho^9 & \rho^{12} \\ 1 & \rho^4 & \rho^8 & \rho^{12} & \rho^{16} \end{bmatrix}$$

$For\ M = 5$

$$\rho = e^{-j2\pi/5} = \cos\left(\frac{2\pi}{5}\right) - jsin\left(\frac{2\pi}{5}\right)$$

$$e^{-j4\pi/5} = \left(e^{-j2\pi/5}\right)^2 = \rho^2$$

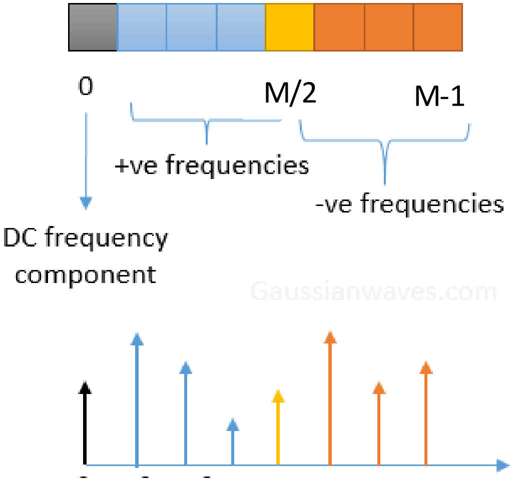
Dale Mugler, “The centered Discrete Fourier Transform and a parallel implementation of the FFT,” ICASSP 2011

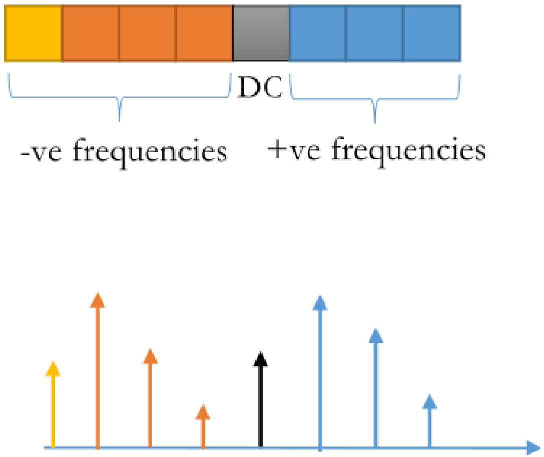


https://www.gaussianwaves.com/2015/11/interpreting-fft-results-complex-dft-frequency-bins-and-fftshift/

DFT – center shifted (for plotting)

$$F[m] = \sum_{n=0}^{n=(M-1)} f[n]\, e^{\frac{-j2\pi mn}{M}}, \mathrm{m} = 0,1, \ldots (\mathrm{M}-1)$$



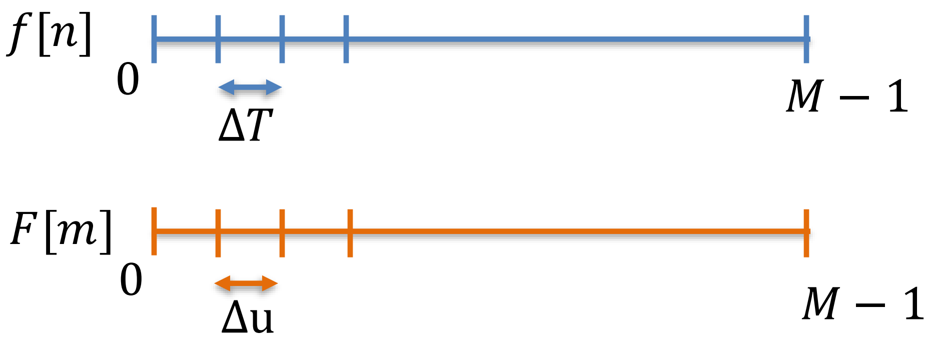


$$F[m] = \sum_{n=0}^{M-1} f[n] e^{\frac{-j2\pi(m-\frac{M-1}{2})(n-\frac{M-1}{2})}{M}}$$

https://www.gaussianwaves.com/2015/11/interpreting-fft-results-complex-dft-frequency-bins-and-fftshift/

Dale Mugler, “The centered Discrete Fourier Transform and a parallel implementation of the FFT,” ICASSP 2011

Relationship between Sampling and Frequency Intervals



- $f[n]$ contains M samples of a function taken ΔT units apart
- Δu (Frequency Resolution of DFT)$= \frac{1}{M\Delta T}$
- Entire frequency range spanned by M components, $R = M\Delta u = \left(\frac{1}{\Delta T}\right)$
- Δu and ΔT follow an inverse relationship

1-D DFT example

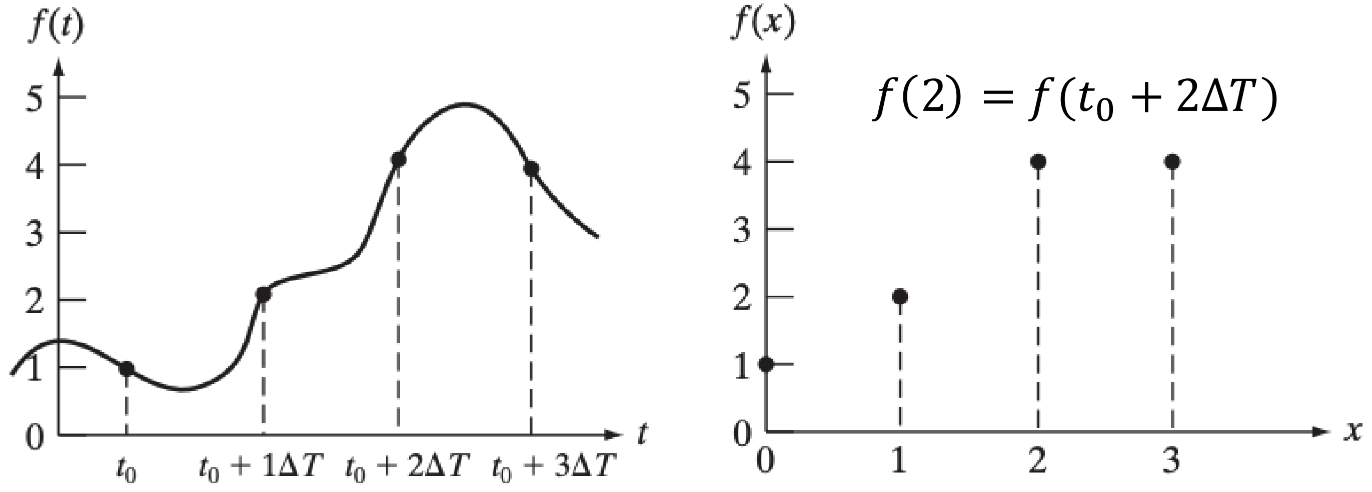


FIGURE 4.11 (a) A function, and (b) samples in the x-domain. In (a), t is a continuous variable; in (b), x represents integer values.

$f(2) = f(t_0 + 2\Delta T)$

$f(0) = 1, f(1) = 2, f(2) = 4, f(3) = 4$

$$F[m] = \sum_{n=0}^{n=(M-1)} f[n]\, e^{\frac{-j2\pi mn}{M}}, \mathrm{m} = 0,1, \ldots (\mathrm{M}-1)$$

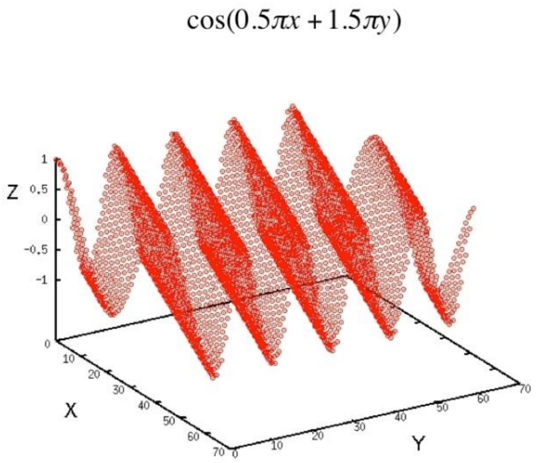
$$e^{j\theta} = cos\theta + jsin\theta;\ e^{-j\theta} = cos\theta - jsin\theta$$

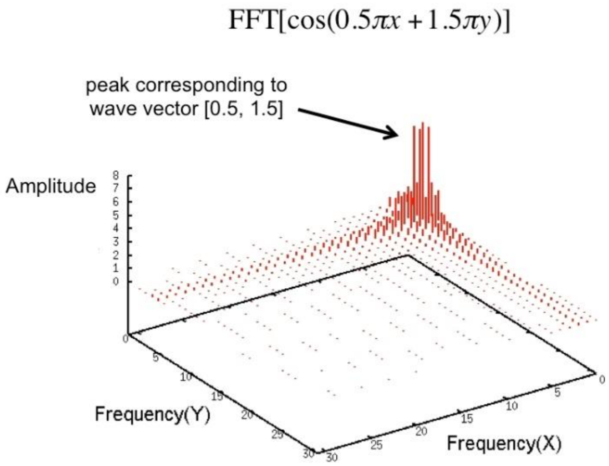
$$F[m]\ at\ m = 0: F[0] = \sum_{x=0}^{3} f(x) = [f(0) + f(1) + f(2) + f(3)] = 1 + 2 + 4 + 4 = 11$$

$$F[1] = \sum_{x=0}^{3} f(x) e^{-j2\pi(1)x/4} = \left[f(0).e^{0} + f(1).e^{-j\pi/2} + f(2).e^{-j\pi} + f(3).e^{-j3\pi/2}\right] = -3 + 2j$$

2D DFT and IDFT

$$F[m,n] = \sum_{y=0}^{y=(N-1)} \sum_{x=0}^{x=(M-1)} f[x,y]\, e^{-2\pi j\left(\frac{mx}{M} + \frac{ny}{N}\right)}$$

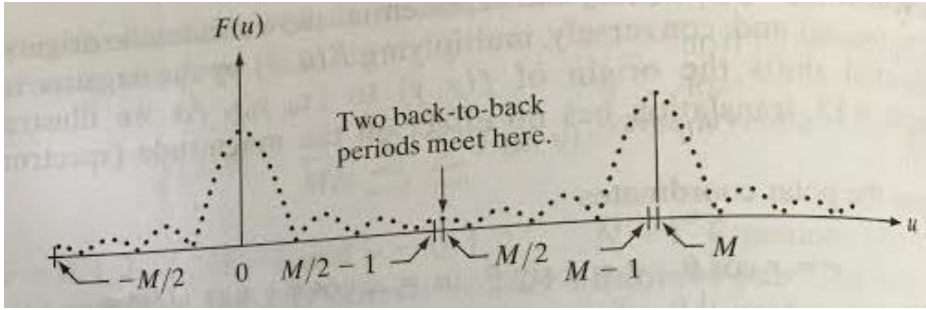


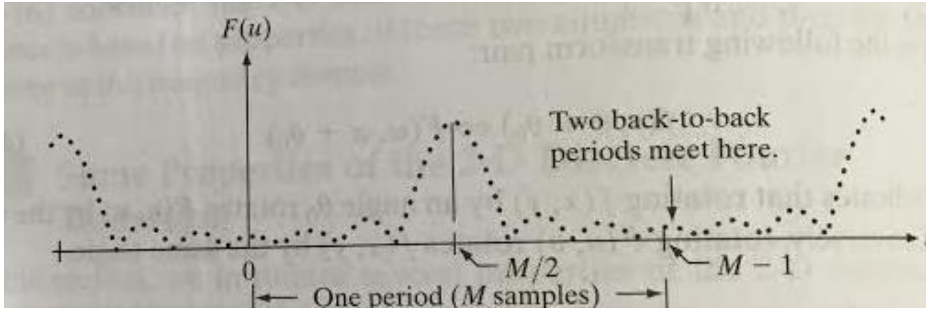


$$f[x,y] = \frac{1}{MN} \sum_{n=0}^{n=(N-1)} \sum_{m=0}^{m=(M-1)} F[m,n]\, e^{2\pi j\left(\frac{mx}{M} + \frac{ny}{N}\right)}$$

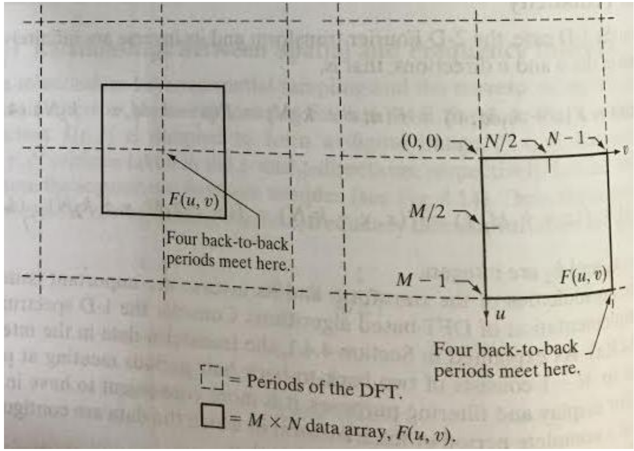
Shifting origin

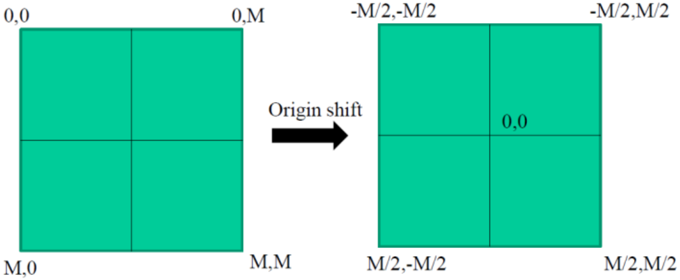
1-D





2-D

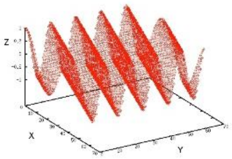




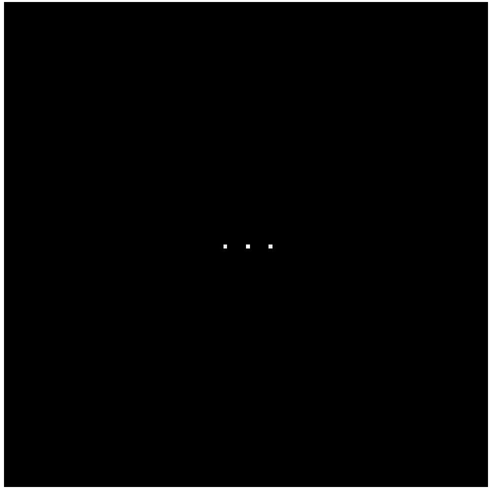
$$f[x,y]e^{j2\pi\left(\frac{u_o x}{M}+\frac{v_o y}{M}\right)} \leftrightarrow F(u-u_o, v-v_0)$$

DFT for simple spatial patterns

Brightness Image

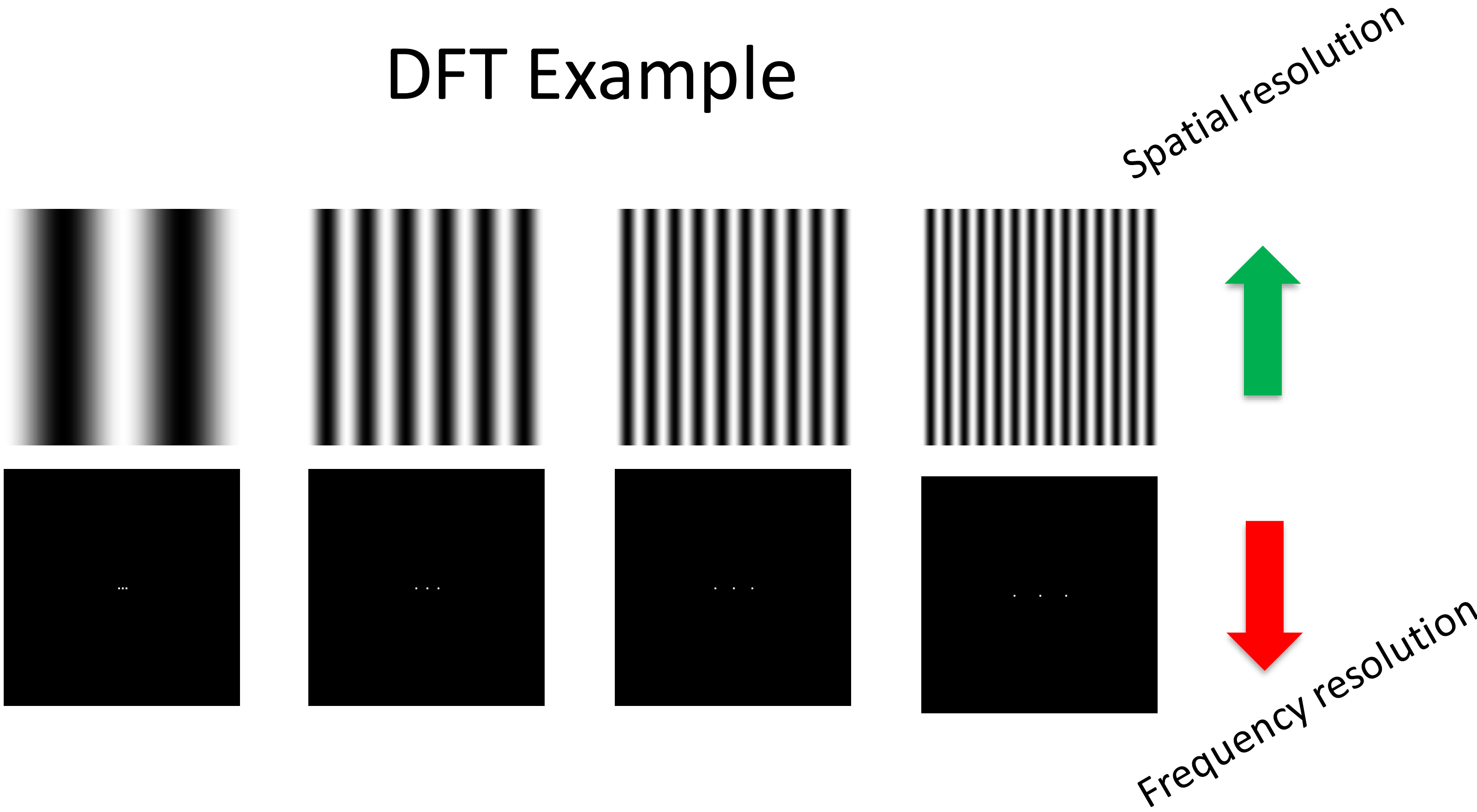


Fourier transform spectrum



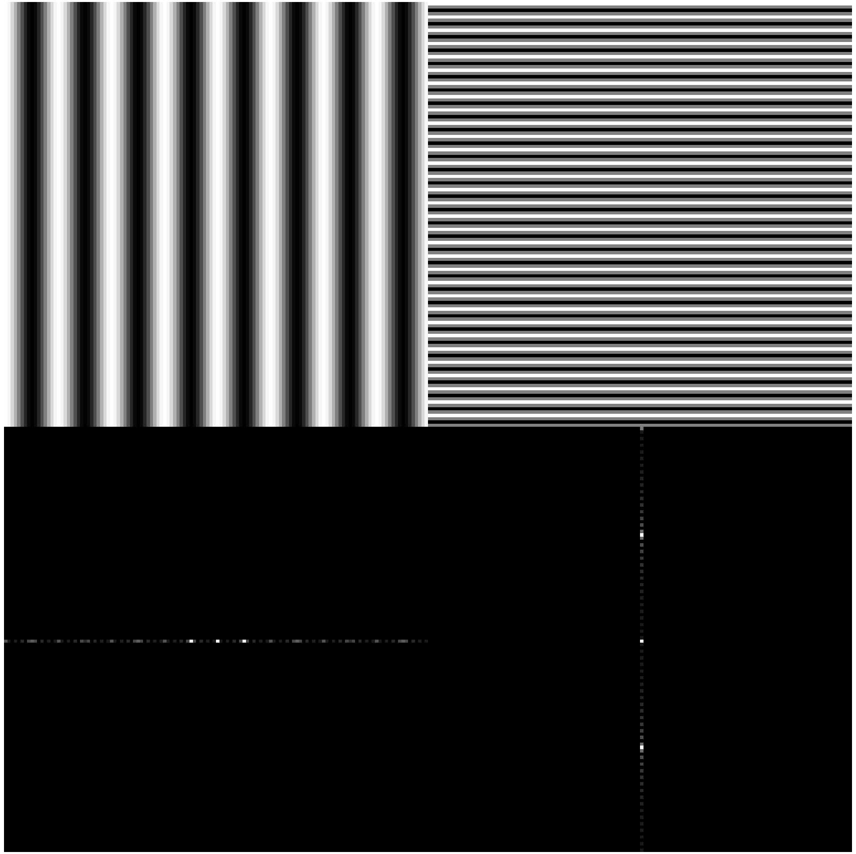
http://cns-alumni.bu.edu/~slehar/fourier/fourier.html

DFT Example

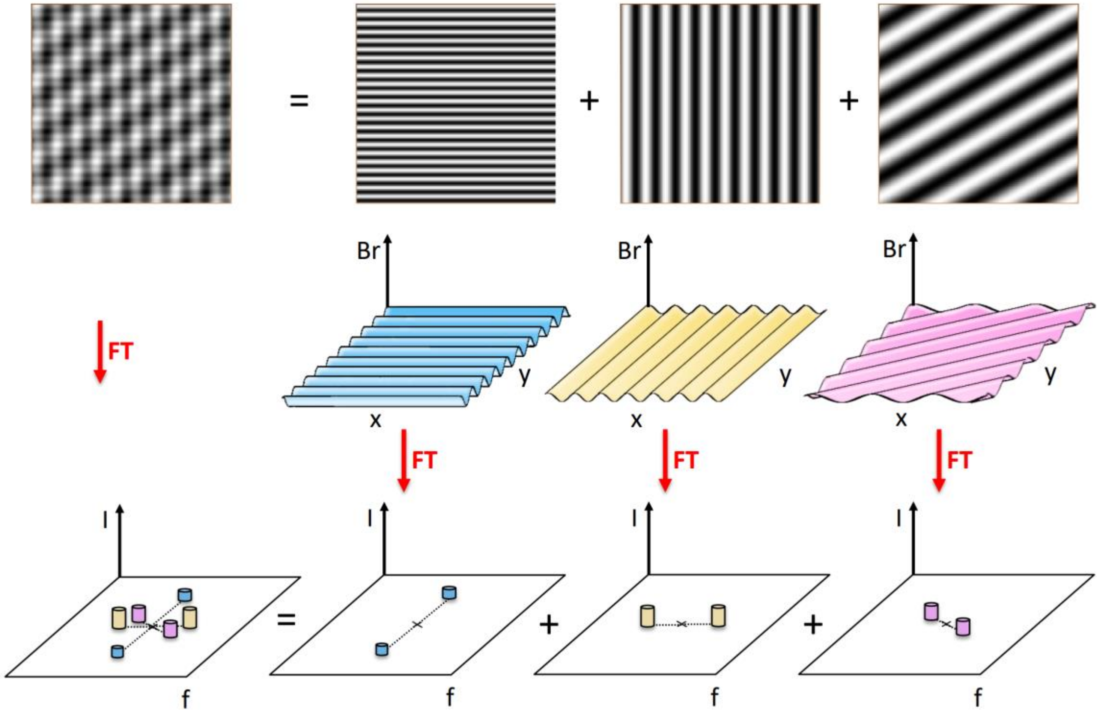


Courtesy: http://cns-alumni.bu.edu/~slehar/fourier/fourier.html

DFT for simple 'spatial' patterns

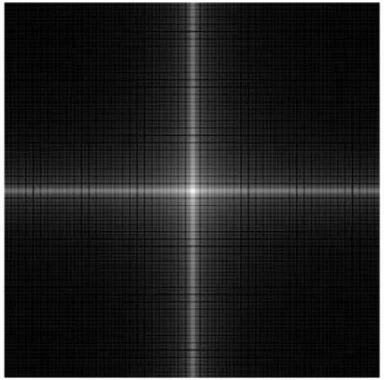


DFT for sum of signals

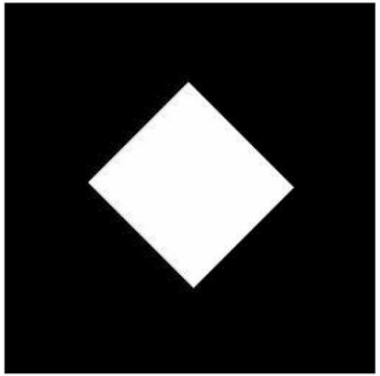


DFT for objects in images

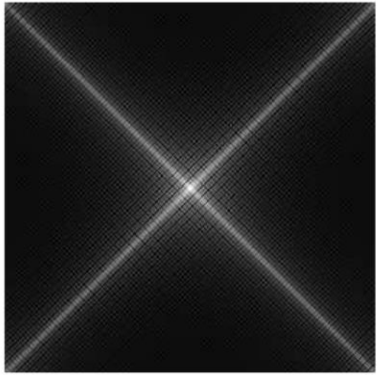
Box



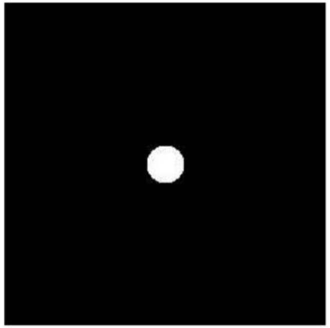
DFT



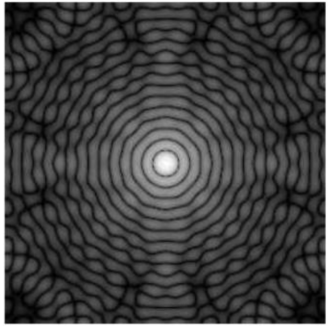
Rotated Box



DFT



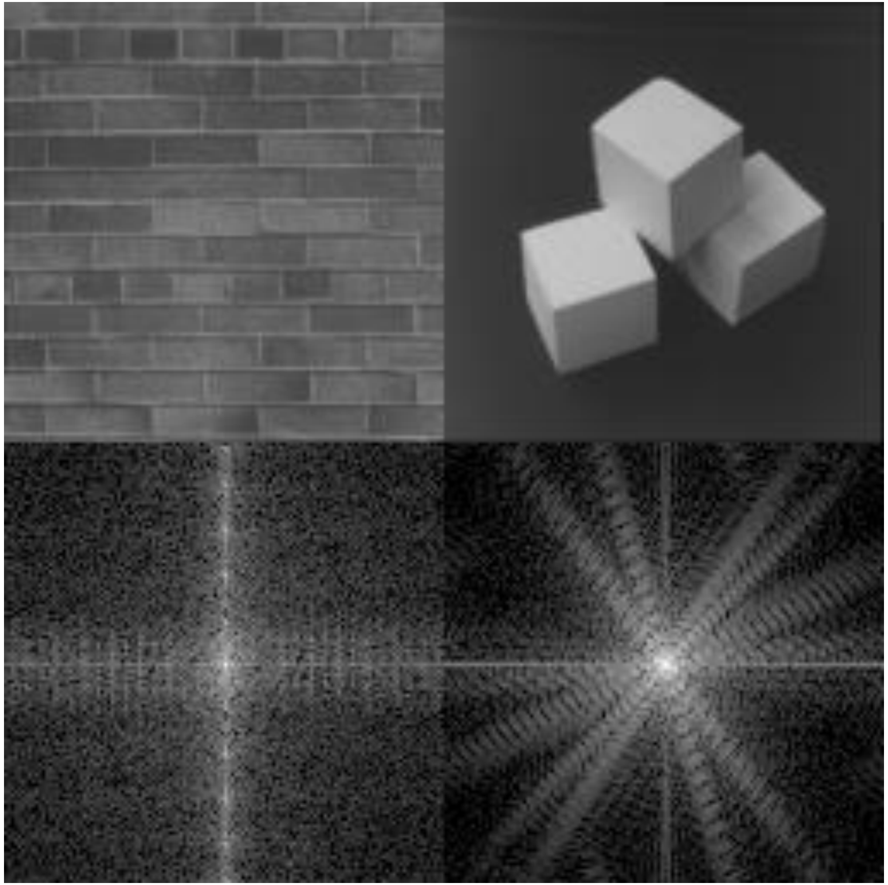
Circle



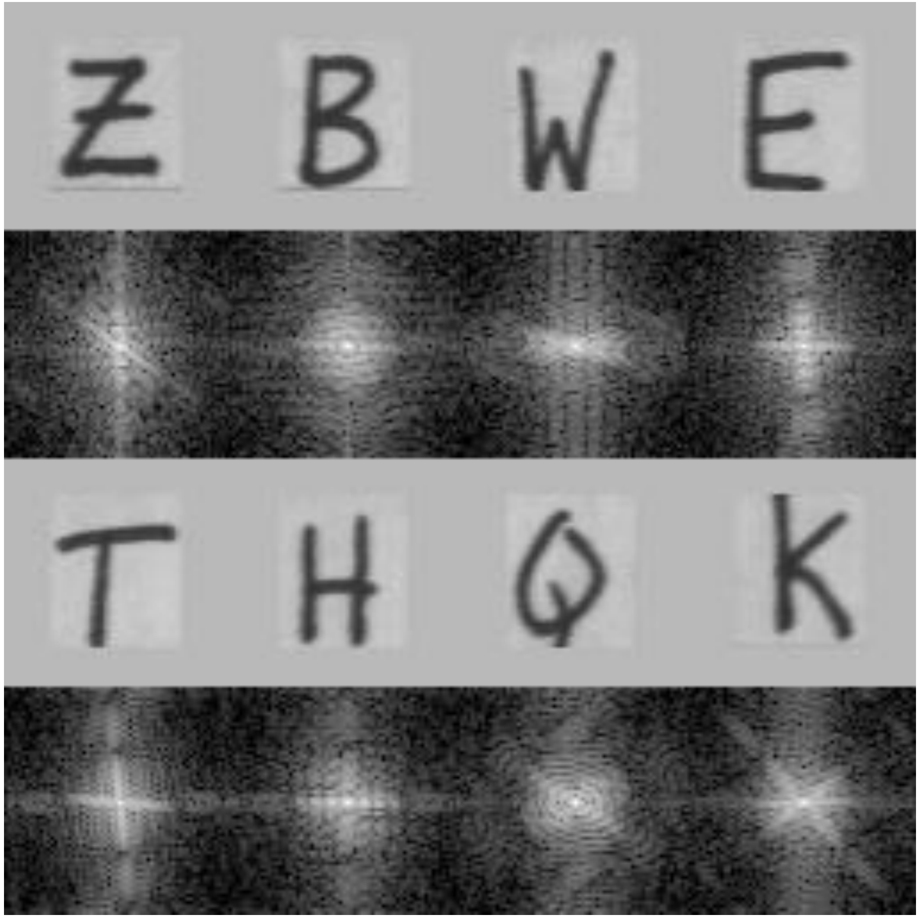
DFT

https://web.cs.wpi.edu/~emmanuel/courses/cs545/S14/slides/lecture10.pdf

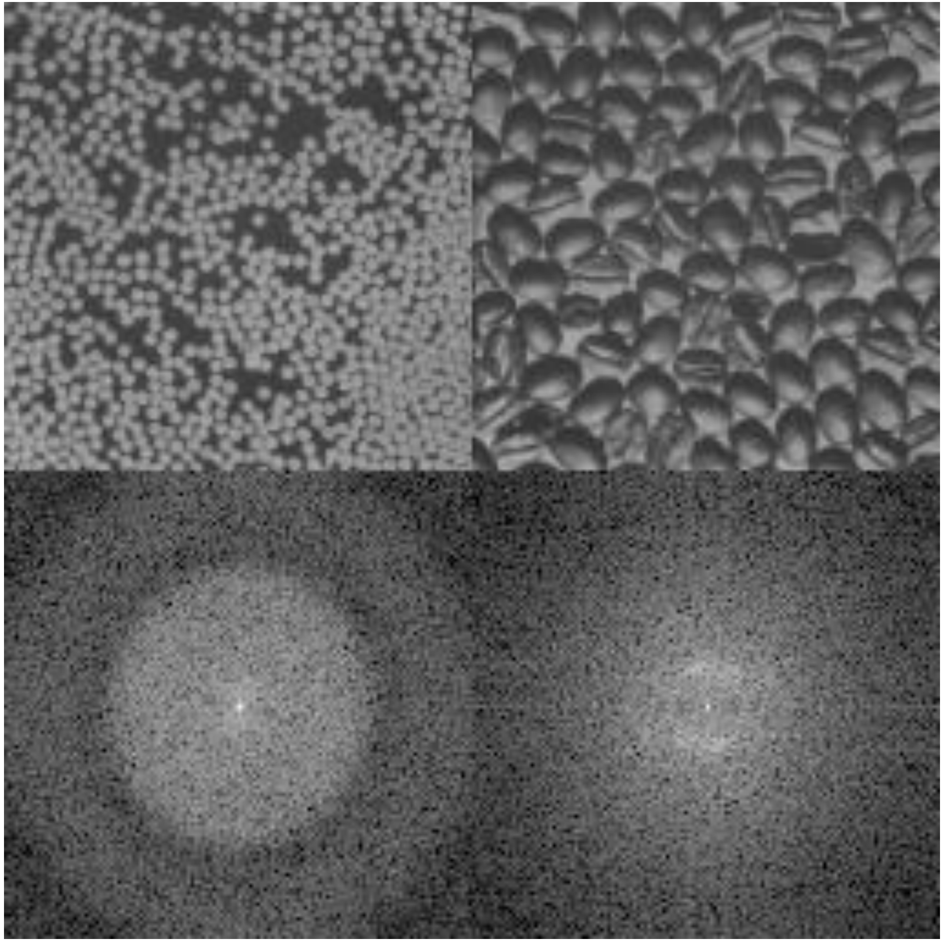
Some examples of images and spectra



Some examples of images and spectra

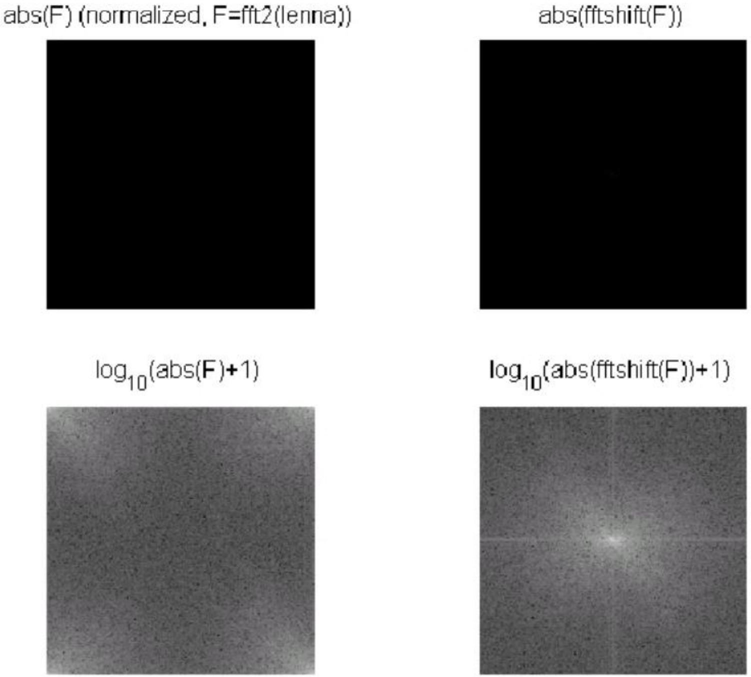


Some examples of images and spectra



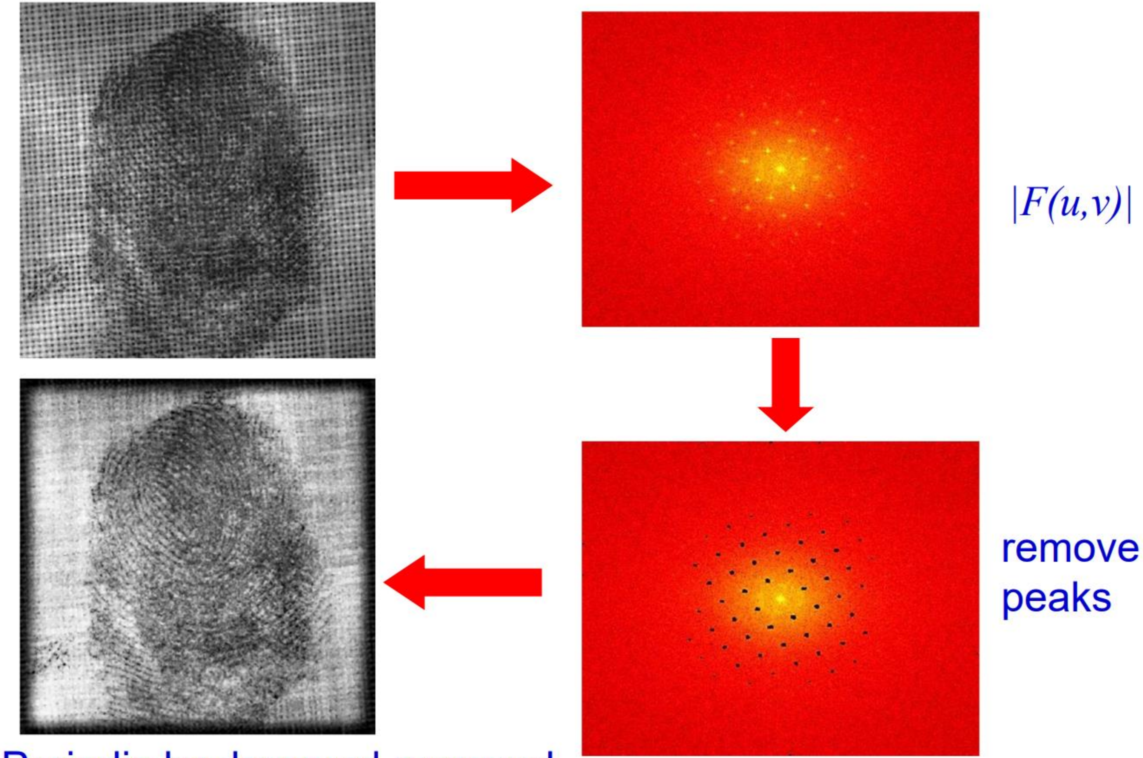
Better visualization

- Amplitude rescaling $G(k,l) = \log(1 + F(k,l))$



https://eeweb.engineering.nyu.edu/~yao/EL5123/lecture6_2D_DFT.pdf

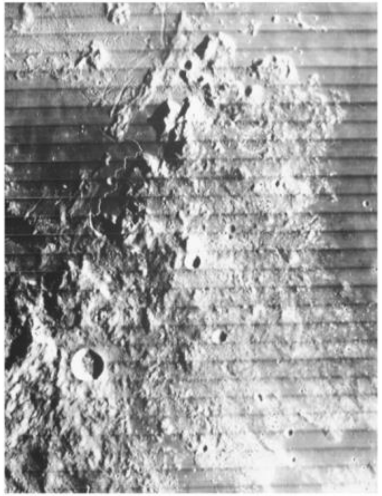
Example – Forensic application

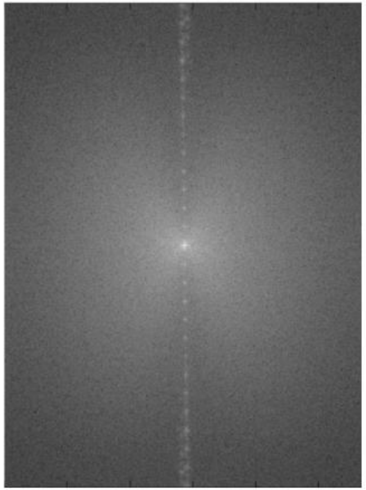


Periodic background removed

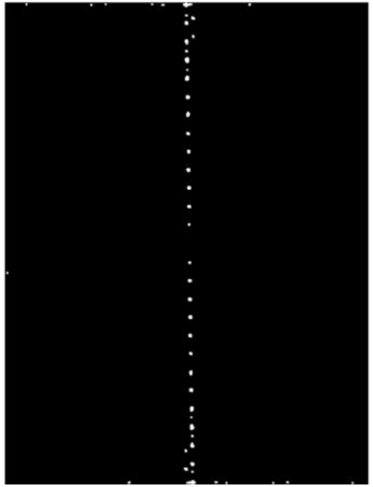
Example – Image processing

Lunar orbital image (1966)





$|F(u,v)|$



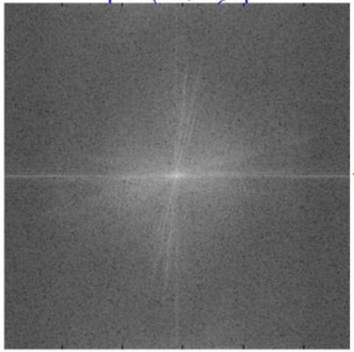
remove peaks

join lines removed

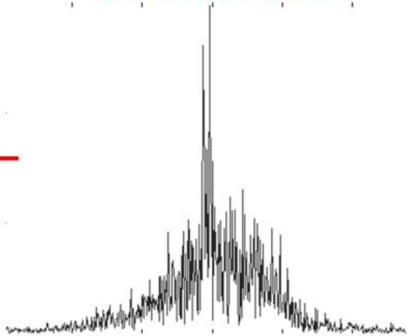
Magnitude vs Phase

$f(x,y)$

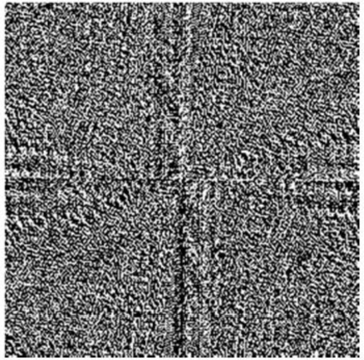
$|F(u,v)|$



cross-section



phase $F(u,v)$



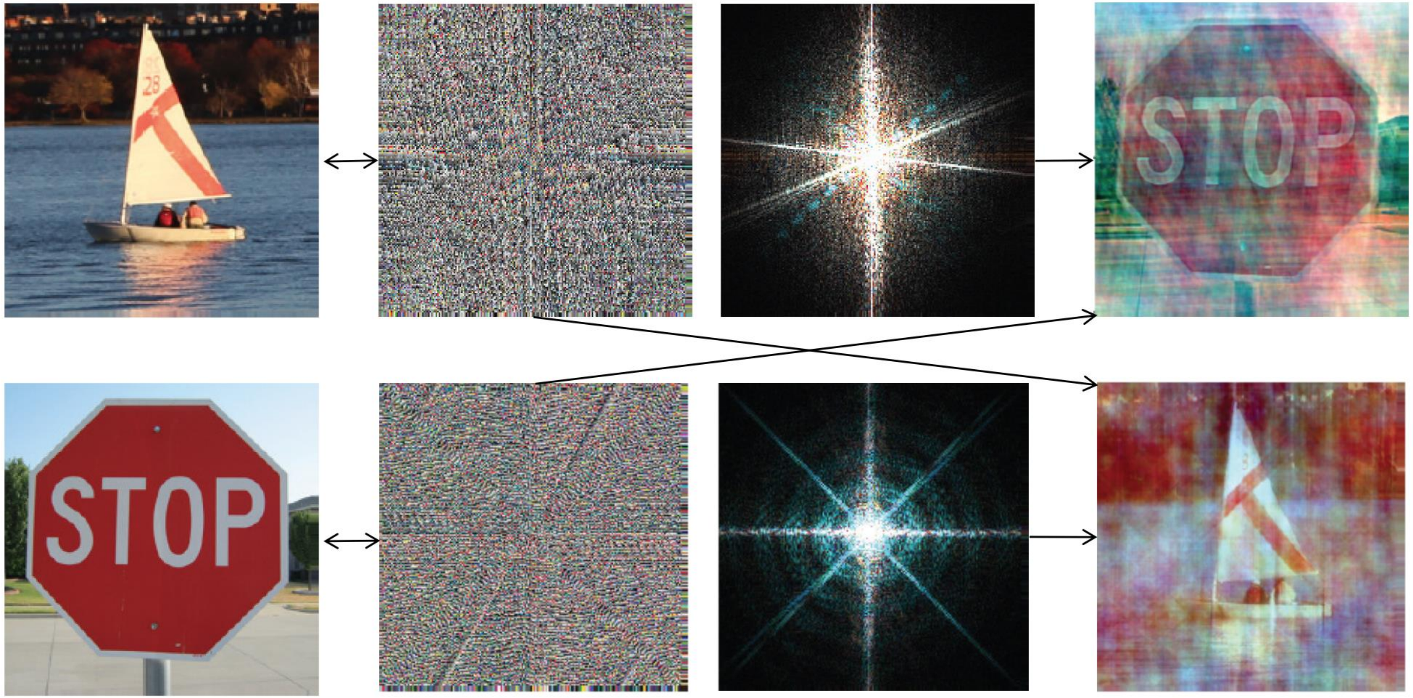
- |f(u,v)| generally decreases with higher spatial frequencies
- phase appears less informative

Phase and Magnitude

- Curious fact
  - all natural images have about the same magnitude transform
  - hence, phase seems to matter, but magnitude largely doesn't

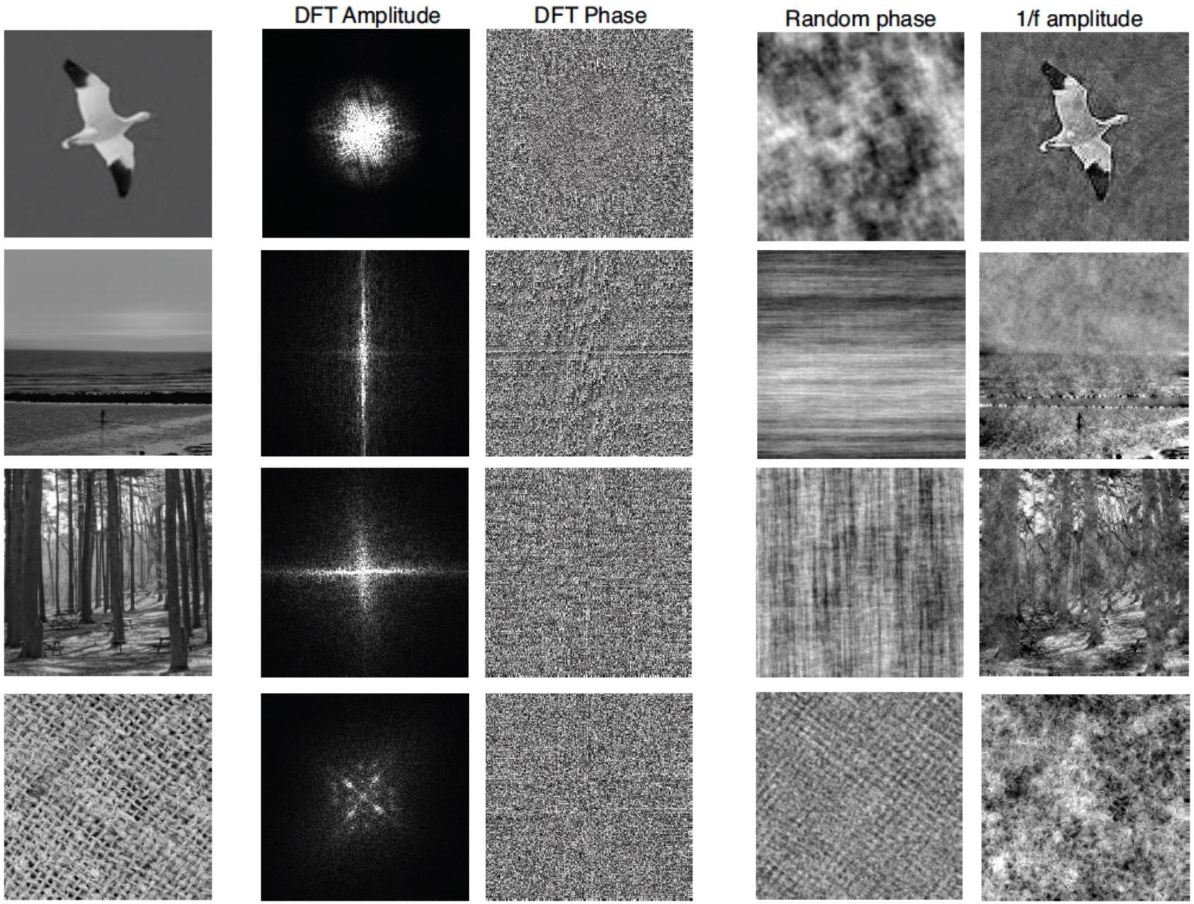
- Demonstration
  - Take two pictures, swap the phase transforms, compute the inverse - what does the result look like?

http://6.869.csail.mit.edu/fa16/lecture/lecture2linearfilters.pdf

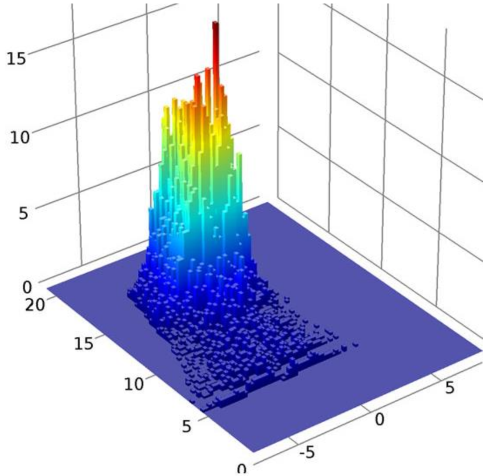


Each color channel is processed in the same way.

http://6.869.csail.mit.edu/fa16/lecture/lecture2linearfilters.pdf



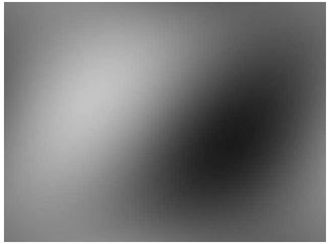
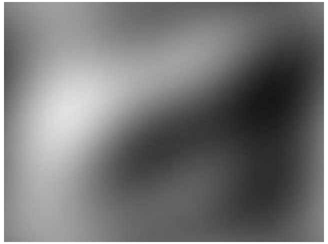
http://6.869.csail.mit.edu/fa16/lecture/lecture2linearfilters.pdf



$$F[m,n] = \sum_{y=0}^{y=(N-1)} \sum_{x=0}^{x=(M-1)} f[x,y]\, e^{-2\pi j\left(\frac{mx}{M}+\frac{ny}{N}\right)}$$

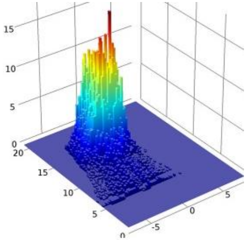
$$f[x,y] = \frac{1}{MN} \sum_{n=0}^{n=(N-1)} \sum_{m=0}^{m=(M-1)} F[m,n]\, e^{2\pi j\left(\frac{mx}{M}+\frac{ny}{N}\right)}$$

$$f[x,y] = \frac{1}{MN} \sum_{n=0}^{n=(N-1)} \sum_{m=0}^{m=(M-1)} F[m,n]\, e^{2\pi j\left(\frac{mx}{M} + \frac{ny}{N}\right)}$$

 +  +  = 

From 50% of the lowest frequencies

+ + = 



Adding up to 50% lowest frequencies

DFT vs FFT computation times

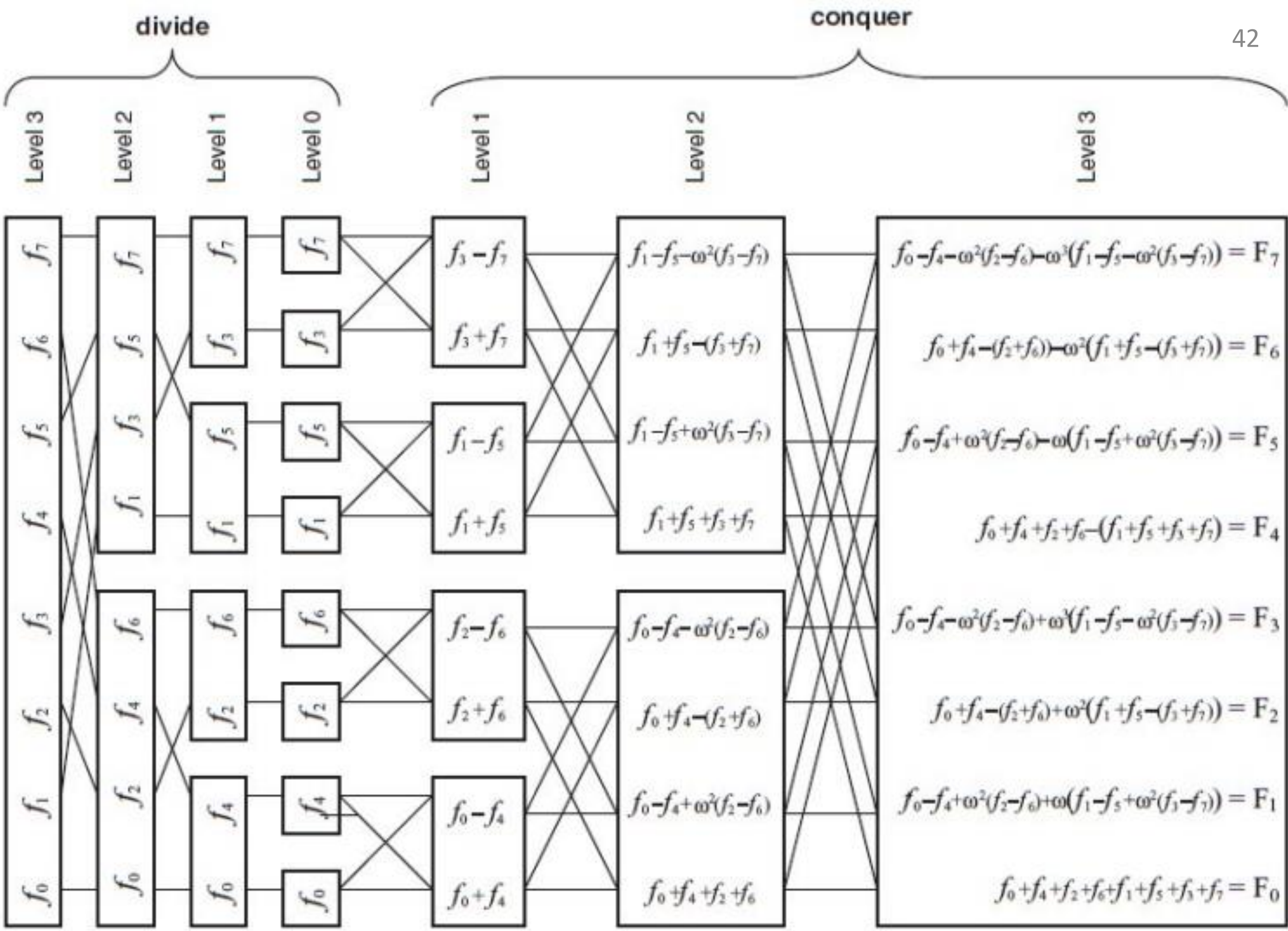
$$F[m] = \sum_{n=0}^{n=(M-1)} f[n]\, e^{\frac{-j2\pi mn}{M}}, \mathrm{m} = 0,1,\ldots(\mathrm{M}-1)$$

n	$N = 2^n$	N^2	$N \log N$
10	1 024	1 048 576	10 240
12	4 096	16 777 216	49 152
14	16 384	268 435 456	229 376
16	65 536	4 294 967 296	1 048 576

Direct computation DFT FFT

(Cooley and Tukey algorithm)

FFT(n, [a_0, a_1, …, a_{n-1}]):
  if n=1: return a_0
  F_{even} = FFT(n/2, [a_0, a_2, …, a_{n-2}])
  F_{odd} = FFT(n/2, [a_1, a_3, …, a_{n-1}])
  for k = 0 to n/2 – 1:
    $\omega^k = e^{2\pi ik/n}$
    $y^k = F_{even\ k} + \omega^k F_{odd\ k}$
    $y^{k+n/2} = F_{even\ k} - \omega^k F_{odd\ k}$
  return [y_0, y_1, …, y_{n-1}]



References

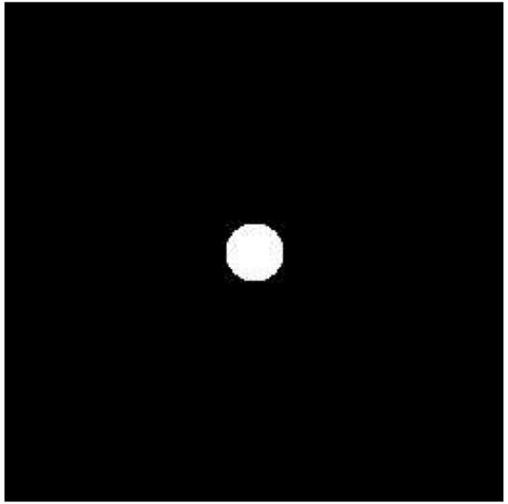
- http://cns-alumni.bu.edu/~slehar/fourier/fourier.html
- https://slideplayer.com/slide/5665338/
- https://2e.mindsmachine.com/asf07.02.html
- https://radiologykey.com/a-walk-through-the-spatial-frequency-domain/
- https://blogs.mathworks.com/steve/2009/12/04/fourier-transform-visualization-using-windowing/
- https://photo.stackexchange.com/questions/40401/what-does-frequency-mean-in-an-image
- http://homepages.inf.ed.ac.uk/rbf/HIPR2/fourier.htm
- http://paulbourke.net/miscellaneous/imagefilter
- https://www.cs.unm.edu/~brayer/vision/fourier.html
- http://6.869.csail.mit.edu/fa16/lecture/lecture2linearfilters.pdf
- https://eeweb.engineering.nyu.edu/~yao/EL5123/lecture6_2D_DFT.pdf

Image Enhancement and Filtering in Frequency Domain

Ideal Low Pass Filters

- Multiply Image Fourier Transform *F* by some filter matrix *m*

$$m(x,y) = \begin{cases} 1 & \text{if } (x,y) \text{ is closer to the center than some value } D, \\ 0 & \text{if } (x,y) \text{ is further from the center than } D. \end{cases}$$



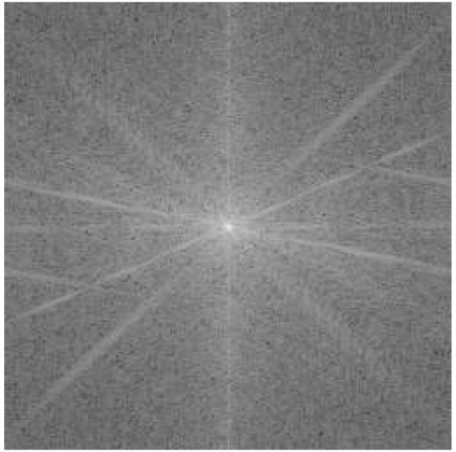
Ideal Low Pass Filters

- Low pass filtered image is inverse Fourier Transform of product of *F* and *m*

$$\mathcal{F}^{-1}(F \cdot m)$$

- Example: Consider the following image and its DFT

Image



DFT

Ideal Low Pass Filters

Applying
low pass filter
to DFT
Cutoff D = 15

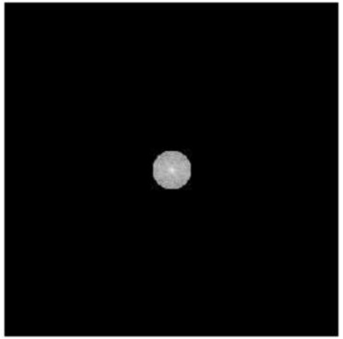
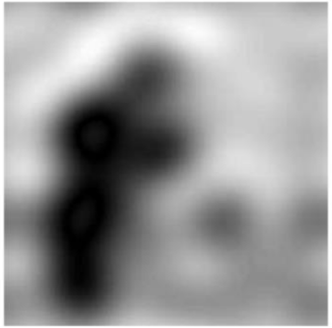


Image after
inversion

low pass filter
Cutoff D = 5

Note: Sharp filter
Cutoff causes
ringing

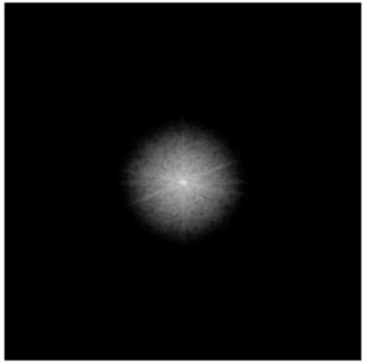


low pass filter
Cutoff D = 30

Gaussian Filtering

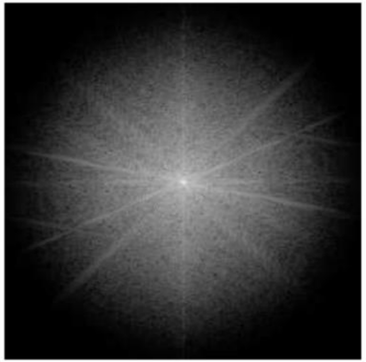
- Gaussian filters can be applied in frequency domain
- Same steps
  - Create gaussian filter
  - Multiply (**DFT of image**) by (**gaussian filter**)
  - Invert result
- **Note:** Fourier transform of gaussian is also a gaussian,
- Just apply gaussian multiply directly (no need to find Fourier transform)

Gaussian Filtering



(a) $\sigma = 10$

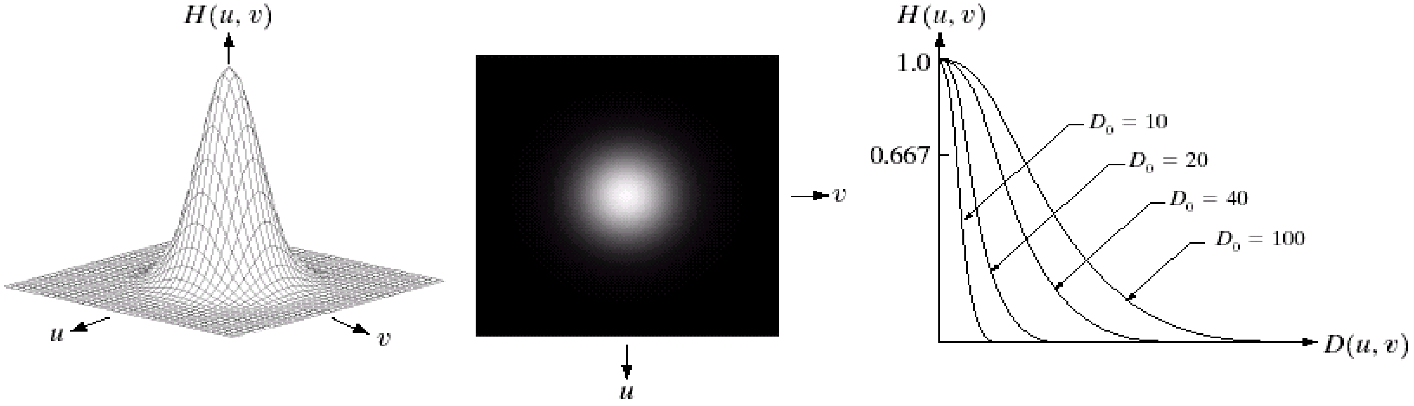
(b) Resulting image



(c) $\sigma = 30$

(d) Resulting image

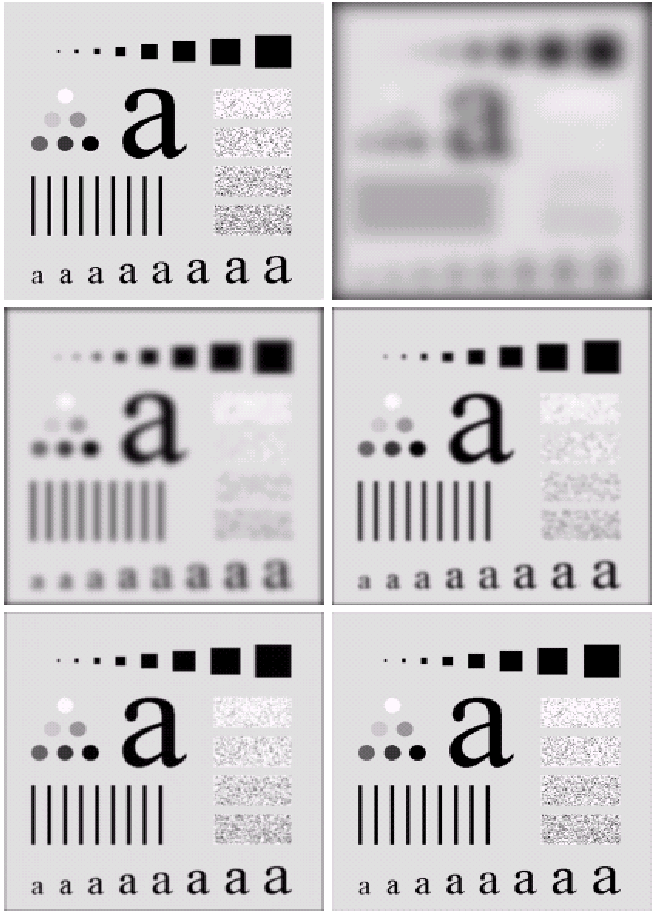
Gaussian Low Pass Filters



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

Image courtesy: Gonzalez and Woods

Gaussian Low Pass Filters (GLPF)



GLPF cut off
frequency 10

GLPF cut off
frequency 30

GLPF cut off
frequency 60

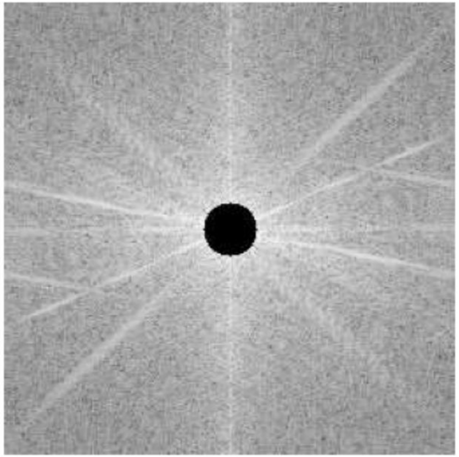
GLPF cut off
frequency 160

GLPF cut off
frequency 460

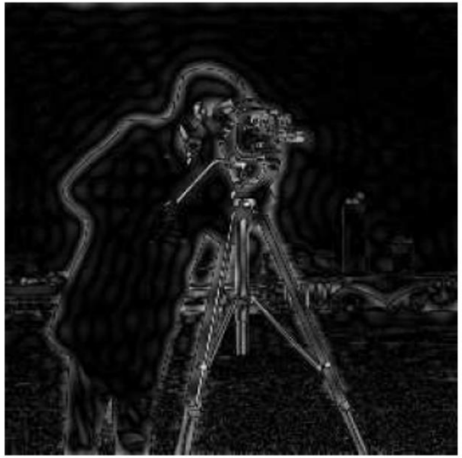
Image courtesy: Gonzalez and Woods

Ideal High Pass Filtering

- Opposite of low pass filtering: eliminate center (low frequency values), keeping others
- High pass filtering causes image **sharpening**
- If we use circle as cutoff again, size affects results
  - Large cutoff = More information removed



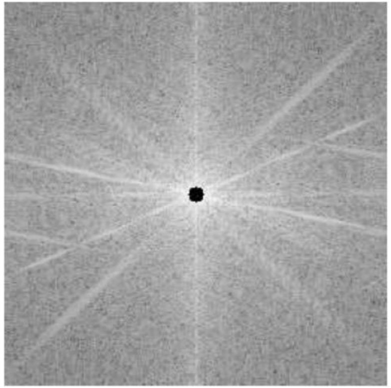
DFT of Image after high pass Filtering



Resulting image after inverse DFT

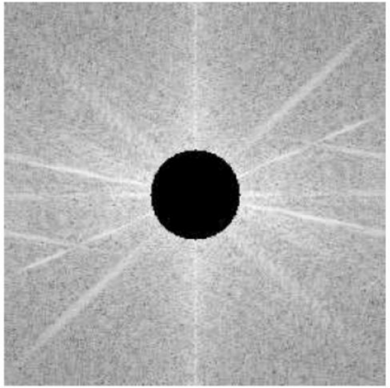
Ideal High Pass Filtering-Effects of cutoffs

High pass filtering of DFT with filter Cutoff D = 5



Low cutoff frequency removes Only few lowest frequencies

High pass filtering of DFT with filter Cutoff D = 30



High cutoff frequency removes many frequencies, leaving only edges

References

- G & W (4.5.1, 4.5.2, 4.5.5, 4.6 – 4.11)
- http://mstrzel.eletel.p.lodz.pl/mstrzel/pattern_rec/fft_ang.pdf