CS443: Digital Imaging and Multimedia
Introduction to Spectral Techniques

Spring 2008
Ahmed Elgammal
Dept. of Computer Science
Rutgers University

Outlines

- Fourier Series and Fourier integral
- Fourier Transform (FT)
- Discrete Fourier Transform (DFT)
- Aliasing and Nyquest Theorem
- 2D FT and 2D DFT
- Application of 2D-DFT in imaging
- Inverse Convolution
- Discrete Cosine Transform (DCT)

Sources:

- Burger and Burge “Digital Image Processing” Chapter 13, 14, 15
- Fourier transform images from Prof. John M. Brayer @ UNM

http://www.cs.unm.edu/~brayer/vision/fourier.html

- Representation and Analysis of Signals in the frequency domain
  - Audio: 1D temporal signal
  - Images: 2D spatial signal
  - Video: 2D spatial signal + 1D temporal signal
- How to decompose a signal into sine and cosine function. Also known as harmonic functions.
- Fourier Transform, Discrete Fourier Transform, Discrete Cosine Transform

Basics

- Sine and Cosine functions are periodic

  $$f(x) = \cos(x)$$

  $$\cos(x) = \cos(x + 2\pi) = \cos(x + 4\pi) = \cdots = \cos(x + k2\pi)$$

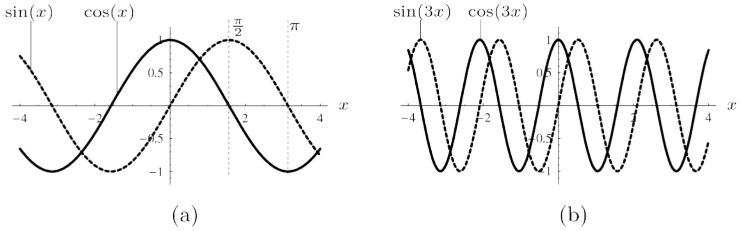
- Angular Frequency: number of oscillations over the distance 2π

  T: the time for a complete cycle

  $$T = \frac{2\pi}{\omega}$$

$$f(x) = \cos(x)$$

$$\omega = \frac{2\pi}{T} = 1$$



(a)

(b)

Basics

- Angular Frequency (ω) and Amplitude (a)

$$a \cdot \cos(\omega x) \qquad \text{and} \qquad a \cdot \sin(\omega x)$$

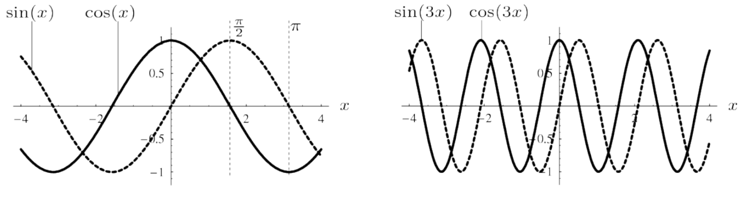
- Angular Frequency: number of oscillations over the distance 2π

  T: the time for a complete cycle

$$T = \frac{2\pi}{\omega}$$

- Common Frequency f: number of oscillation in a unit time

$$f = \frac{1}{T} = \frac{\omega}{2\pi} \quad \text{or} \quad \omega = 2\pi f$$



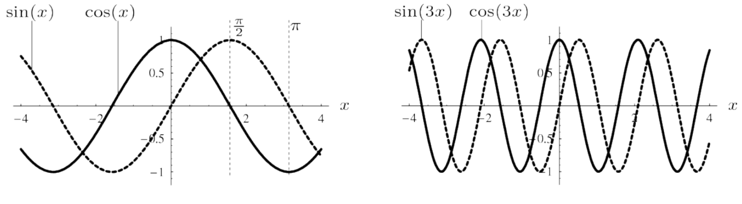
(a)

(b)

Basics

- Phase: Shifting a cosine function along the x axis by a distance φ change the phase of the cosine wave. φ denotes the phase angle

$$\cos(x) \rightarrow \cos(x - \varphi)$$



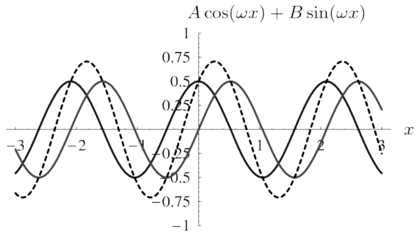
(a)

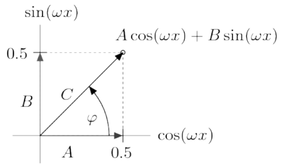
(b)

- Adding cosines and sines with the same frequency results in another sinusoid

$$A \cdot \cos(\omega x) + B \cdot \sin(\omega x) = C \cdot \cos(\omega x - \varphi)$$

$C = \sqrt{A^2 + B^2}$ and $\varphi = \tan^{-1}\left(\frac{B}{A}\right)$





Fourier Series and Fourier integral

- We can represent any periodic function as sum of pairs of sinusoidal functions- using a basic (fundamental) frequency

$$g(x) = \sum_{k=0}^{\infty} [A_k \cos(k\omega_0 x) + B_k \sin(k\omega_0 x)]$$

- Fourier Integral: any function can be represented as combination of sinusoidal functions with many frequencies

$$g(x) = \int_0^{\infty} A_\omega \cos(\omega x) + B_\omega \sin(\omega x) \,\mathrm{d}\omega$$

- Fourier Integral

$$g(x) = \int_0^\infty A_\omega \cos(\omega x) + B_\omega \sin(\omega x)\ \mathrm{d}\omega$$

- How much of each frequency contributes to a given function

$$A_\omega = A(\omega) = \frac{1}{\pi}\int_{-\infty}^{\infty} g(x)\cdot\cos(\omega x)\ \mathrm{d}x$$

$$B_\omega = B(\omega) = \frac{1}{\pi}\int_{-\infty}^{\infty} g(x)\cdot\sin(\omega x)\ \mathrm{d}x$$

Fourier Transform

$$A_\omega = A(\omega) = \frac{1}{\pi}\int_{-\infty}^{\infty} g(x)\cdot\cos(\omega x)\ \mathrm{d}x$$

$$B_\omega = B(\omega) = \frac{1}{\pi}\int_{-\infty}^{\infty} g(x)\cdot\sin(\omega x)\ \mathrm{d}x$$

$$\begin{aligned}
G(\omega) &= \sqrt{\tfrac{\pi}{2}}\Big[A(\omega) - \mathrm{i}\cdot B(\omega)\Big] \\
&= \sqrt{\tfrac{\pi}{2}}\left[\frac{1}{\pi}\int_{-\infty}^{\infty} g(x)\cdot\cos(\omega x)\ \mathrm{d}x \ - \ \mathrm{i}\cdot\frac{1}{\pi}\int_{-\infty}^{\infty} g(x)\cdot\sin(\omega x)\ \mathrm{d}x\right] \\
&= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} g(x)\cdot\Big[\cos(\omega x) - \mathrm{i}\cdot\sin(\omega x)\Big]\ \mathrm{d}x,
\end{aligned}$$

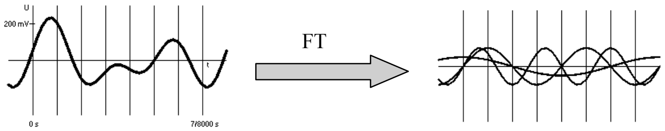
$$\boxed{\begin{aligned}
G(\omega) &= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} g(x)\cdot\Big[\cos(\omega x) - \mathrm{i}\cdot\sin(\omega x)\Big]\ \mathrm{d}x \\
&= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} g(x)\cdot e^{-\mathrm{i}\omega x}\ \mathrm{d}x.
\end{aligned}}$$

- Fourier transform

$$\begin{aligned} G(\omega) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} g(x) \cdot \Big[\cos(\omega x) - \mathrm{i} \cdot \sin(\omega x)\Big] \, \mathrm{d}x \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} g(x) \cdot e^{-\mathrm{i}\omega x} \, \mathrm{d}x. \end{aligned}$$

- Inverse Fourier transform

$$\begin{aligned} g(x) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} G(\omega) \cdot \Big[\cos(\omega x) + \mathrm{i} \cdot \sin(\omega x)\Big] \, \mathrm{d}\omega \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} G(\omega) \cdot e^{\mathrm{i}\omega x} \, \mathrm{d}\omega. \end{aligned}$$



Frequency domain

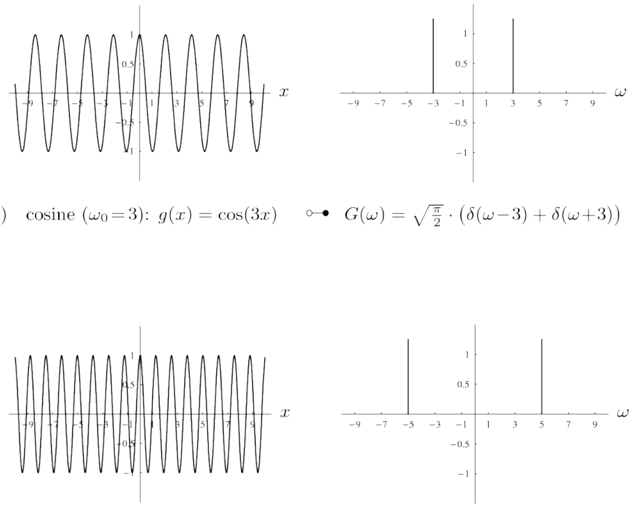
Temporal or spatial domain

Fourier Transform

- The forward and inverse transformation are almost similar (only the sign in the exponent is different)
- any signal is represented in the frequency space by its frequency “spectrum”
- The Fourier spectrum is uniquely defined for a given function. The opposite is also true.
- Fourier transform pairs

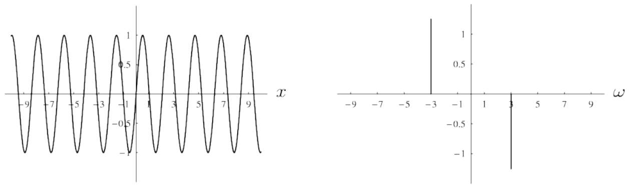
$$g(x) \;\circ\!\!-\!\!\bullet\; G(\omega)$$

Function	*Transform Pair* $g(x) \circ\!\!-\!\!\bullet\, G(\omega)$	*Figure*
Cosine function with frequency ω_0	$g(x) = \cos(\omega_0 x)$ $G(\omega) = \sqrt{\frac{\pi}{2}} \cdot \big(\delta(\omega{-}\omega_0) + \delta(\omega{+}\omega_0)\big)$	13.3 (a, c)
Sine function with frequency ω_0	$g(x) = \sin(\omega_0 x)$ $G(\omega) = \mathrm{i}\sqrt{\frac{\pi}{2}} \cdot \big(\delta(\omega{-}\omega_0) - \delta(\omega{+}\omega_0)\big)$	13.3 (b, d)
Gaussian function of width σ	$g(x) = \frac{1}{\sigma} \cdot e^{-\frac{x^2}{2\sigma^2}}$ $G(\omega) = e^{-\frac{\sigma^2\omega^2}{2}}$	13.4 (a, b)
Rectangular pulse of width $2b$	$g(x) = \Pi_b(x) = \begin{cases} 1 & \text{for } \lvert x\rvert \le b \\ 0 & \text{otherwise} \end{cases}$ $G(\omega) = \frac{2b\sin(b\omega)}{\sqrt{2\pi}\omega}$	13.4 (c, d)

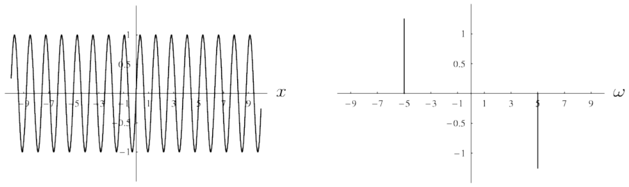


(a) cosine ($\omega_0 = 3$): $g(x) = \cos(3x)$ $\circ\!\!-\!\!\bullet$ $G(\omega) = \sqrt{\frac{\pi}{2}} \cdot \big(\delta(\omega{-}3) + \delta(\omega{+}3)\big)$

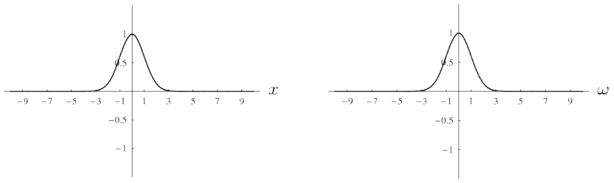
(c) cosine ($\omega_0 = 5$): $g(x) = \cos(5x)$ $\circ\!\!-\!\!\bullet$ $G(\omega) = \sqrt{\frac{\pi}{2}} \cdot \big(\delta(\omega{-}5) + \delta(\omega{+}5)\big)$



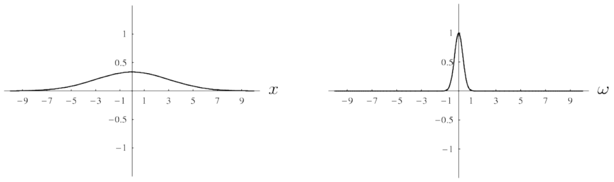
(b) sine ($\omega_0=3$): $g(x)=\sin(3x)$ $\circ\!\!-\!\!\bullet$ $G(\omega)=\mathrm{i}\sqrt{\frac{\pi}{2}}\cdot\big(\delta(\omega-3)-\delta(\omega+3)\big)$



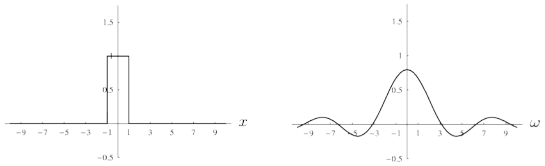
(d) sine ($\omega_0=5$): $g(x)=\sin(5x)$ $\circ\!\!-\!\!\bullet$ $G(\omega)=\mathrm{i}\sqrt{\frac{\pi}{2}}\cdot\big(\delta(\omega-5)-\delta(\omega+5)\big)$



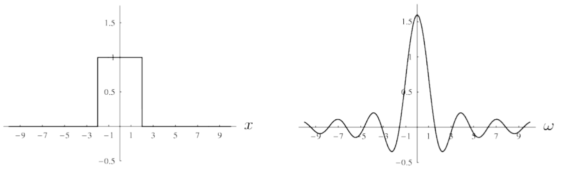
(a) Gauss ($\sigma=1$): $g(x)=e^{-\frac{x^2}{2}}$ $\circ\!\!-\!\!\bullet$ $G(\omega)=e^{-\frac{\omega^2}{2}}$



(b) Gauss ($\sigma=3$): $g(x)=\frac{1}{3}\cdot e^{-\frac{x^2}{2\cdot 9}}$ $\circ\!\!-\!\!\bullet$ $G(\omega)=e^{-\frac{9\omega^2}{2}}$



(c) rect. pulse ($b=1$): $g(x) = \Pi_1(x)$ ○—● $G(\omega) = \frac{2\sin(\omega)}{\sqrt{2\pi}\omega}$

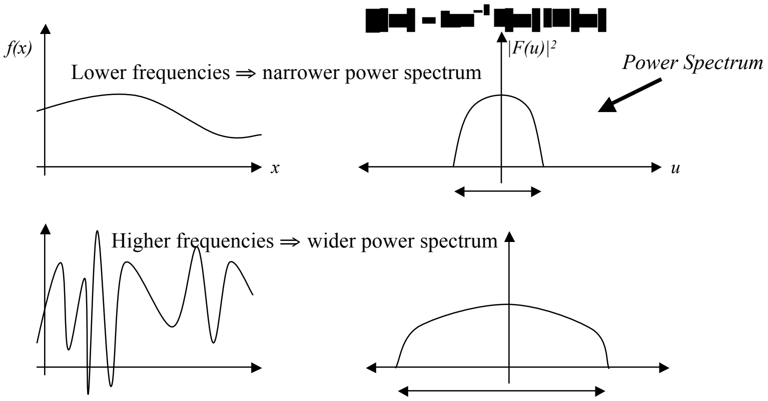


(d) rect. pulse ($b=2$): $g(x) = \Pi_2(x)$ ○—● $G(\omega) = \frac{4\sin(2\omega)}{\sqrt{2\pi}\omega}$

- Since of FT of a real function is generally complex, we use magnitude and phase

[illegible] ⇒ [illegible]

[illegible]


f(x)
Lower frequencies ⇒ narrower power spectrum
|F(u)|²
Power Spectrum
x
u
Higher frequencies ⇒ wider power spectrum

Properties

- Symmetry: for real-valued functions

$$G(\omega) = G^*(-\omega)$$

- Linearity

$$c \cdot g(x) \circ\!\!-\!\!\bullet\; c \cdot G(\omega)$$

$$g_1(x) + g_2(x) \circ\!\!-\!\!\bullet\; G_1(\omega) + G_2(\omega)$$

- Similarity

$$g(sx) \circ\!\!-\!\!\bullet\; \frac{1}{|s|} \cdot G\left(\frac{\omega}{s}\right)$$

- Shift Property

$$g(x-d) \circ\!\!-\!\!\bullet\; e^{-\mathrm{i}\omega d} \cdot G(\omega)$$

Important Properties:

- FT and Convolution
- Convolving two signals is equivalent to multiplying their Fourier spectra

$$g(x) * h(x) \circ\!\!-\!\!\bullet\ G(\omega) \cdot H(\omega)$$

- Multiplying two signals is equivalent to convolving their Fourier spectra

$$g(x) \cdot h(x) \circ\!\!-\!\!\bullet\ G(\omega) * H(\omega)$$

- FT of a Gaussian is a Gaussian

Discrete Fourier Transform

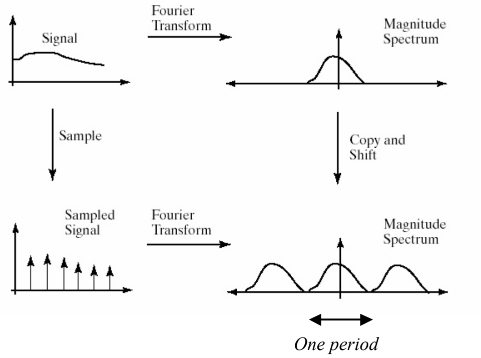
- If we discretize *f(x)* using uniformly spaced samples *f(0), f(1),…,f(N-1)*, we can obtain FT of the sampled function

[illegible]

[illegible]

- Important Property:

Periodicity *F(m)=F(m+N)*



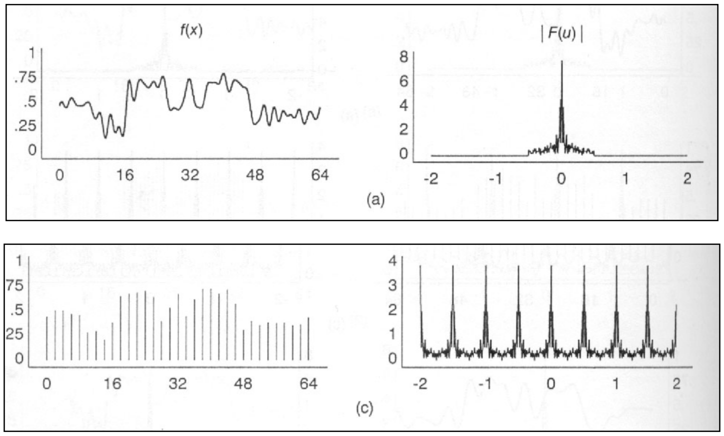


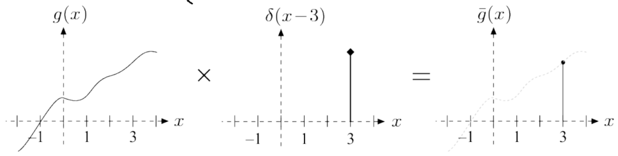
Image from *Computer Graphics: Principles and Practice*
by Foley, van Dam, Feiner, and Hughes

Impulse function

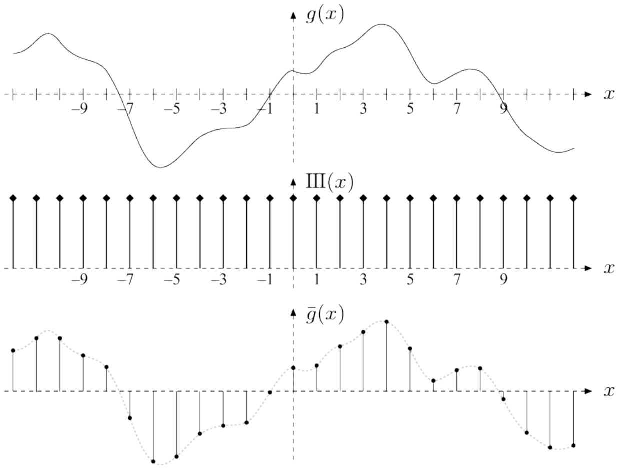
$$\delta(x) = 0 \text{ for } x \neq 0 \quad \text{and} \quad \int_{-\infty}^{\infty} \delta(x)\, \mathrm{d}x = 1$$

$$\delta(sx) = \frac{1}{|s|} \cdot \delta(x) \quad \text{for } s \neq 0$$

$$\bar{g}(x) = g(x) \cdot \delta(x - x_0) = \begin{cases} g(x_0) & \text{for } x = x_0 \\ 0 & \text{otherwise} \end{cases}$$

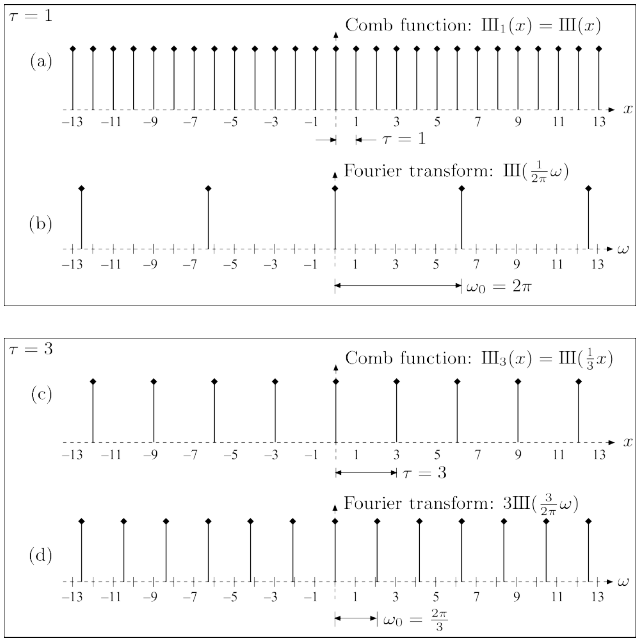


$$f(x) * \delta(x - d) = f(x - d)$$



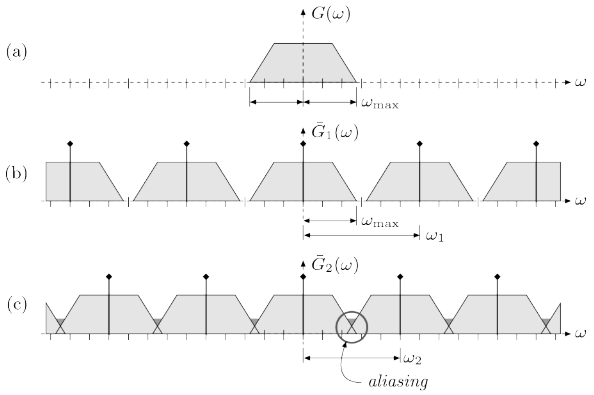
$$\bar{g}(x) = g(x) \cdot \text{III}(x)$$

$$\text{III}(x) = \sum_{i=-\infty}^{\infty} \delta(x - i)$$



$$\text{III}(x) \;\circ\!\!-\!\!\bullet\; \text{III}\left(\tfrac{1}{2\pi}\omega\right)$$

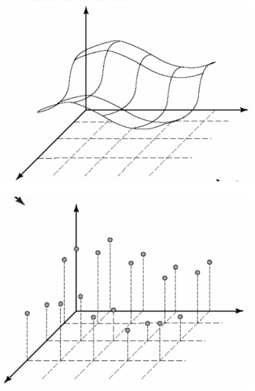
$$\text{III}\left(\tfrac{x}{\tau}\right) \;\circ\!\!-\!\!\bullet\; \tau\text{III}\left(\tfrac{\tau}{2\pi}\omega\right)$$



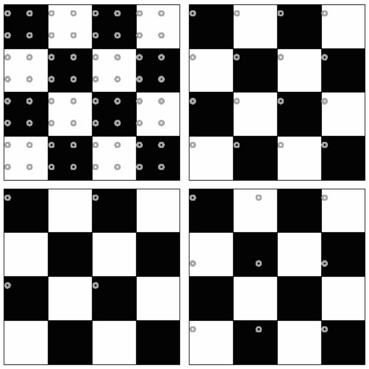
$$g(x) \cdot \mathrm{III}\left(\tfrac{x}{\tau}\right) \circ\!\!-\!\!\bullet\; G(\omega) * \tau\mathrm{III}\left(\tfrac{\tau}{2\pi}\omega\right)$$

Sampling and Aliasing

- Differences between continuous and discrete images
- Images are sampled version of a continuous brightness function.



successful sampling

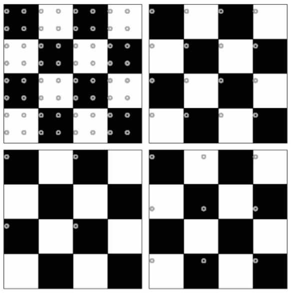


unsuccessful sampling

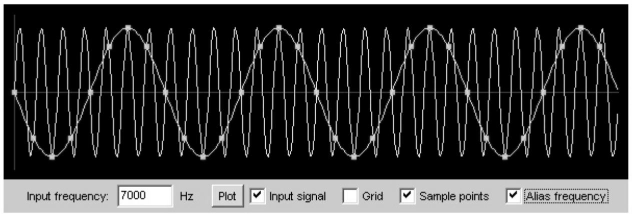
Sampling and Aliasing

- Sampling involves loss of information
- Aliasing: high spatial frequency components appear as low spatial frequency components in the sampled signal

successful sampling



unsuccessful sampling

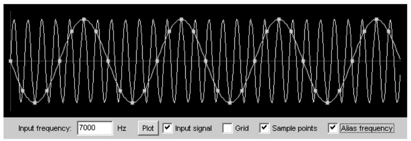


Input frequency: 7000 Hz Plot ☑ Input signal □ Grid ☑ Sample points ☑ Alias frequency

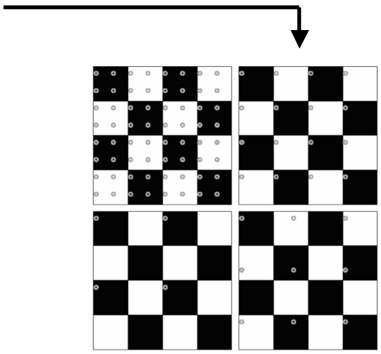
Java applet from: http://www.dsptutor.freeuk.com/aliasing/AD102.html

Aliasing

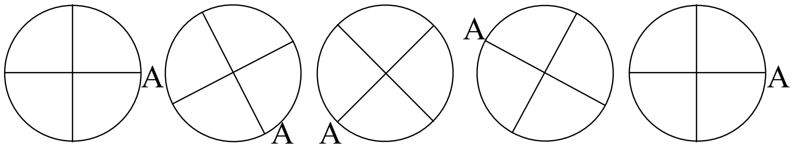
- *Nyquist theorem*: The sampling frequency must be at least twice the highest frequency present for a signal to be reconstructed from a sampled version. *(Nyquist frequency)*



Input frequency: 7000 Hz Plot ☑ Input signal □ Grid ☑ Sample points ☑ Alias frequency



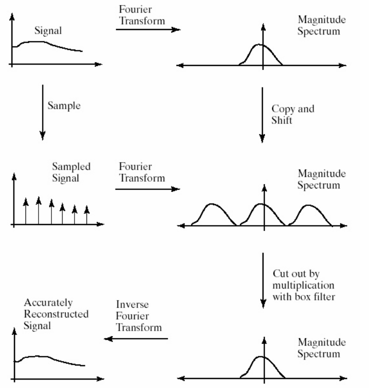
Temporal Aliasing

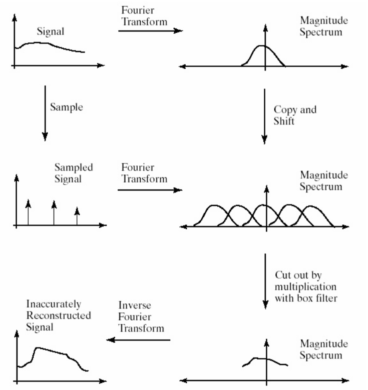


The wheel appears to be moving backwards at about ¼ angular frequency

Sampling, aliasing, and DFT

- DFT consists of a sum of copies of the FT of the original signal shifted by the sampling frequency:
  - If shifted copies do not intersect: reconstruction is possible.
  - If shifted copies do intersect: incorrect reconstruction, high frequencies are lost (Aliasing)





2-dimension

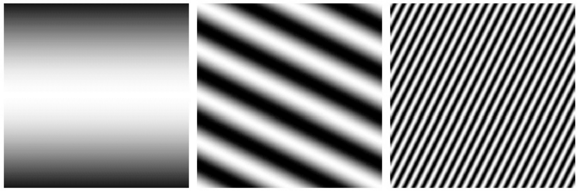
In two dimension

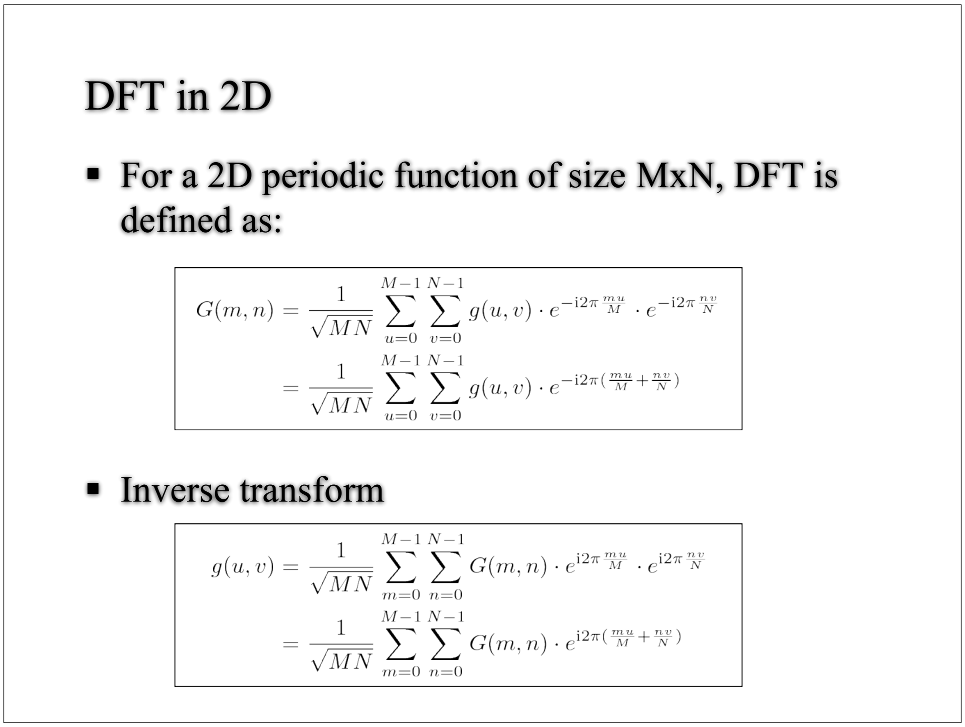
[illegible]

[illegible]

[illegible]

- These terms are sinusoids on the x,y plane whose orientation and frequency are defined by u,v





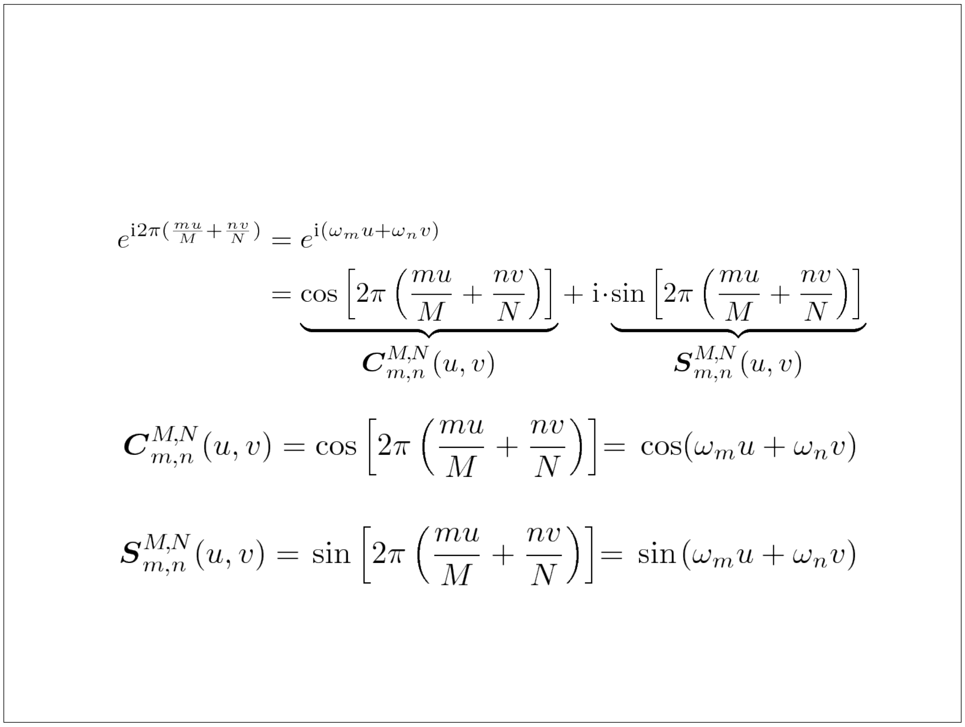
DFT in 2D

- For a 2D periodic function of size MxN, DFT is defined as:

$$\boxed{\begin{aligned} G(m,n) &= \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} g(u,v) \cdot e^{-\mathrm{i}2\pi \frac{mu}{M}} \cdot e^{-\mathrm{i}2\pi \frac{nv}{N}} \\ &= \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} g(u,v) \cdot e^{-\mathrm{i}2\pi \left(\frac{mu}{M} + \frac{nv}{N}\right)} \end{aligned}}$$

- Inverse transform

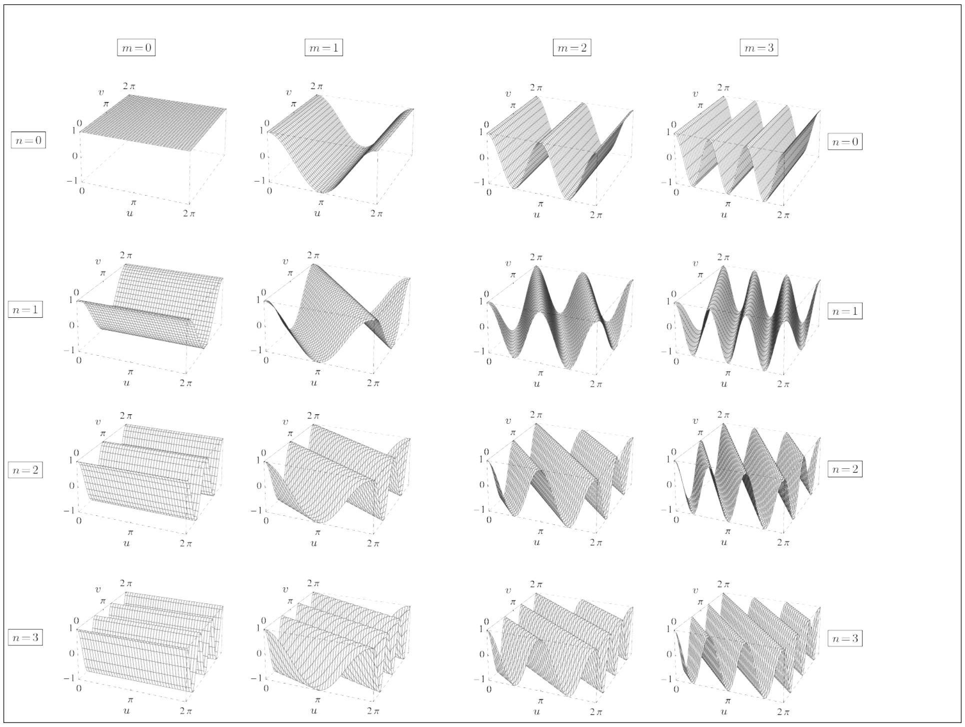
$$\boxed{\begin{aligned} g(u,v) &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} G(m,n) \cdot e^{\mathrm{i}2\pi \frac{mu}{M}} \cdot e^{\mathrm{i}2\pi \frac{nv}{N}} \\ &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} G(m,n) \cdot e^{\mathrm{i}2\pi \left(\frac{mu}{M} + \frac{nv}{N}\right)} \end{aligned}}$$

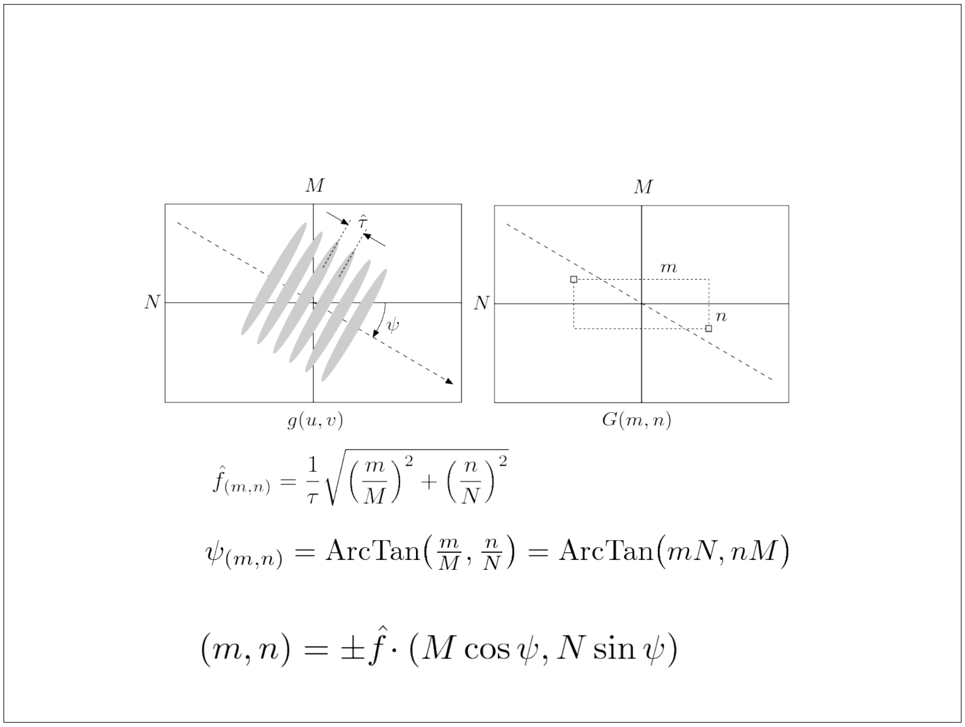


$$\begin{aligned} e^{\mathrm{i}2\pi\left(\frac{mu}{M}+\frac{nv}{N}\right)} &= e^{\mathrm{i}(\omega_m u + \omega_n v)} \\ &= \underbrace{\cos\left[2\pi\left(\frac{mu}{M}+\frac{nv}{N}\right)\right]}_{\boldsymbol{C}_{m,n}^{M,N}(u,v)} + \mathrm{i}\cdot \underbrace{\sin\left[2\pi\left(\frac{mu}{M}+\frac{nv}{N}\right)\right]}_{\boldsymbol{S}_{m,n}^{M,N}(u,v)} \end{aligned}$$

$$\boldsymbol{C}_{m,n}^{M,N}(u,v) = \cos\left[2\pi\left(\frac{mu}{M}+\frac{nv}{N}\right)\right] = \cos(\omega_m u + \omega_n v)$$

$$\boldsymbol{S}_{m,n}^{M,N}(u,v) = \sin\left[2\pi\left(\frac{mu}{M}+\frac{nv}{N}\right)\right] = \sin(\omega_m u + \omega_n v)$$

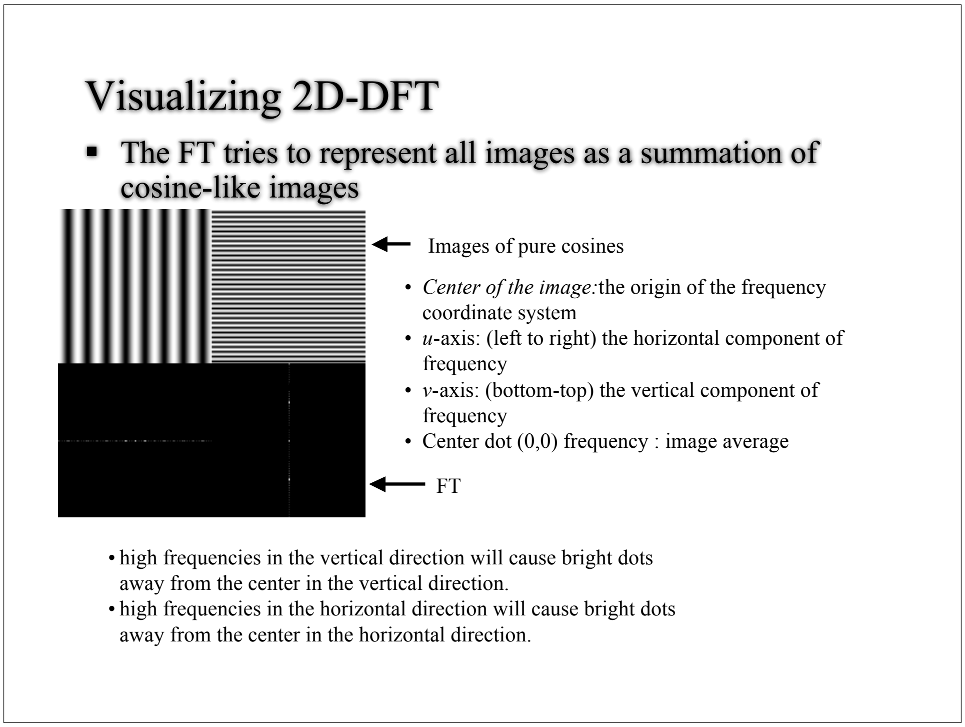




$$\hat{f}_{(m,n)} = \frac{1}{\tau}\sqrt{\left(\frac{m}{M}\right)^2 + \left(\frac{n}{N}\right)^2}$$

$$\psi_{(m,n)} = \mathrm{ArcTan}\big(\tfrac{m}{M}, \tfrac{n}{N}\big) = \mathrm{ArcTan}\big(mN, nM\big)$$

$$(m,n) = \pm\hat{f}\cdot(M\cos\psi, N\sin\psi)$$



Visualizing 2D-DFT

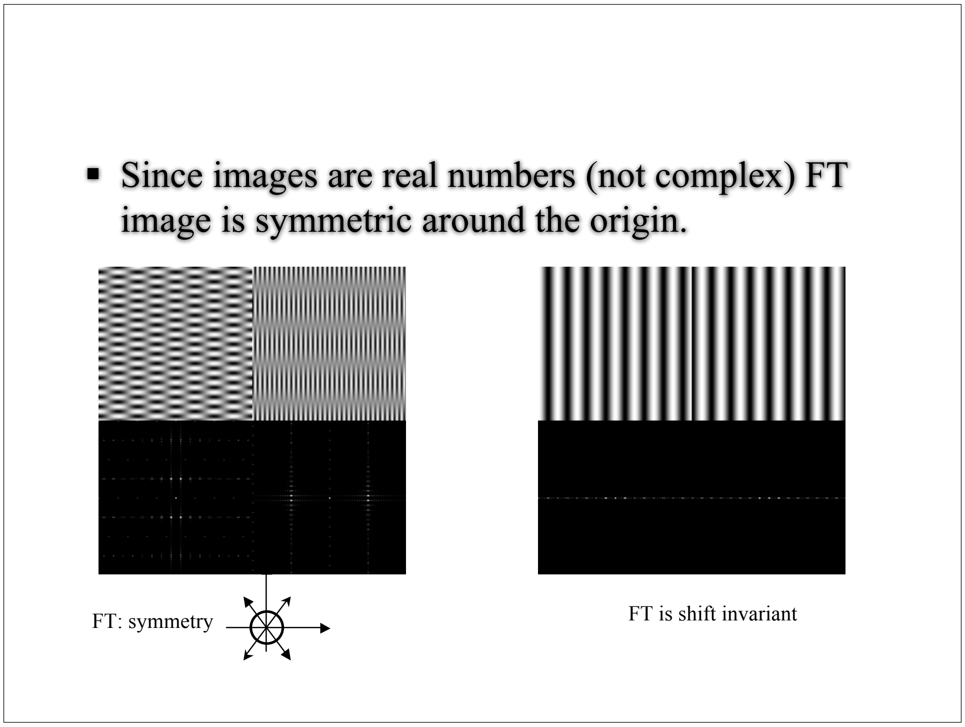
- The FT tries to represent all images as a summation of cosine-like images

Images of pure cosines

- *Center of the image:*the origin of the frequency coordinate system
- *u*-axis: (left to right) the horizontal component of frequency
- *v*-axis: (bottom-top) the vertical component of frequency
- Center dot (0,0) frequency : image average

FT

- high frequencies in the vertical direction will cause bright dots away from the center in the vertical direction.
- high frequencies in the horizontal direction will cause bright dots away from the center in the horizontal direction.

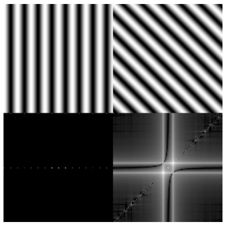


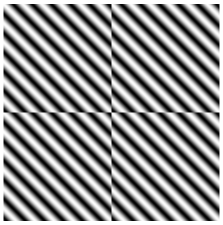
- Since images are real numbers (not complex) FT image is symmetric around the origin.

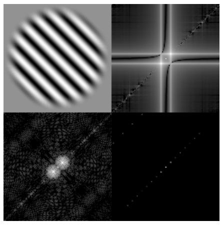
FT: symmetry

FT is shift invariant

- In general, rotation of the image results in equivalent rotation of its FT



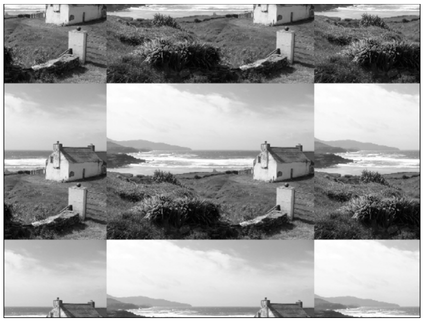


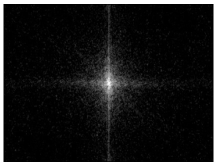


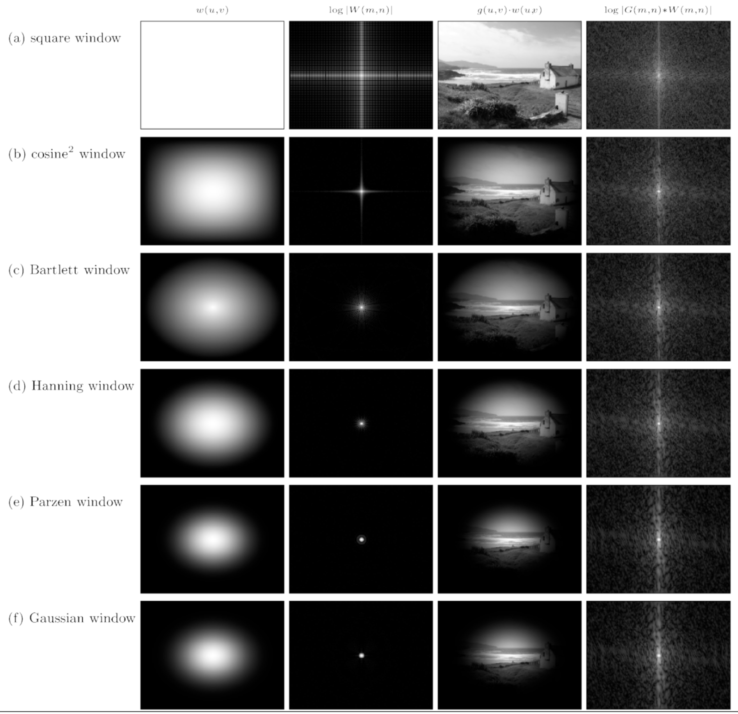
Why it is not the case ?

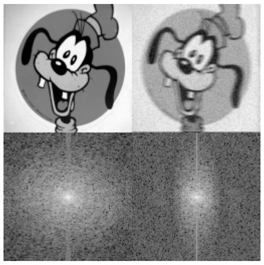
- Edge effect !
- FT always treats an image as if it were part of a periodically replicated array of identical images extending horizontally and vertically to infinity
- Solution: “windowing” the image

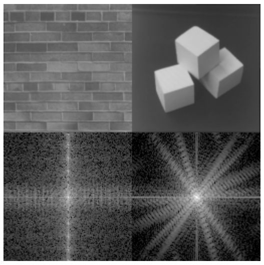
Edge effect



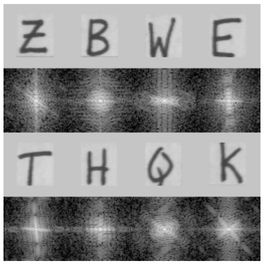


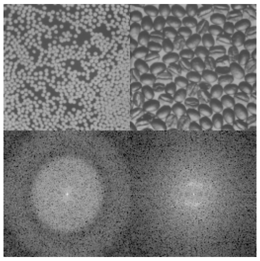


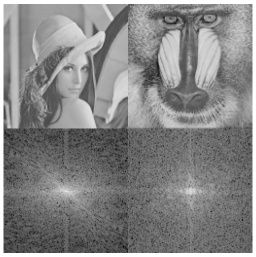




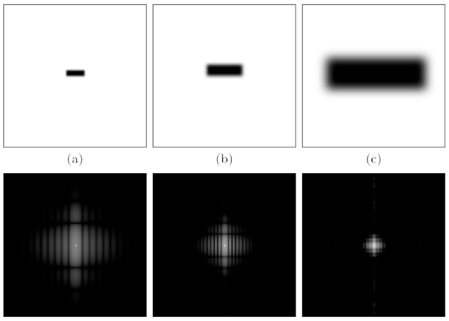
- notice a bright band going to high frequencies perpendicular to the strong edges in the image
- Anytime an image has a strong-contrast, sharp edge the gray values must change very rapidly. It takes lots of high frequency power to follow such an edge so there is usually such a line in its magnitude spectrum.





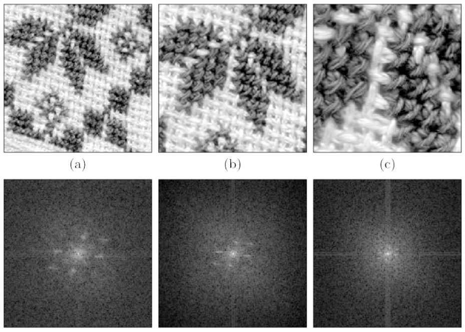


Scaling


(a) (b) (c)

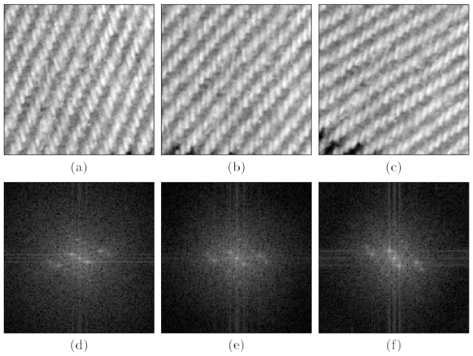
(d) (e) (f)

Periodic image patterns


(a) (b) (c)

(d) (e) (f)

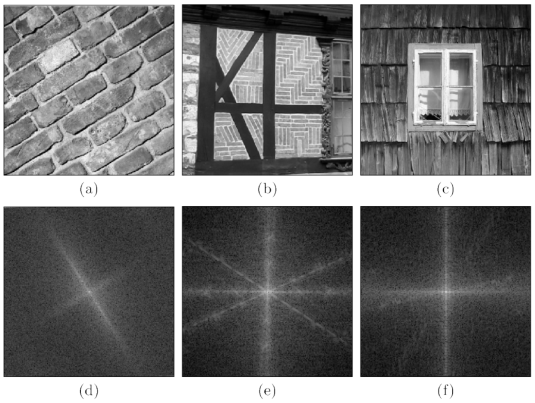
Rotation



(a) (b) (c)

(d) (e) (f)

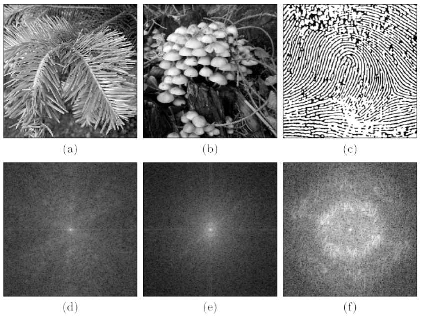
Oriented, elongated structures



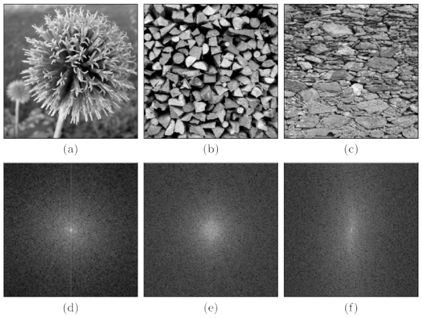
(a) (b) (c)

(d) (e) (f)

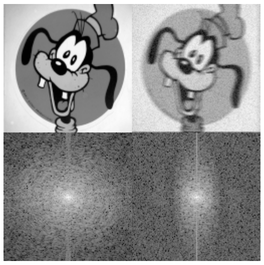
Natural Images

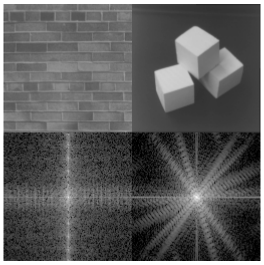

(a) (b) (c)

(d) (e) (f)

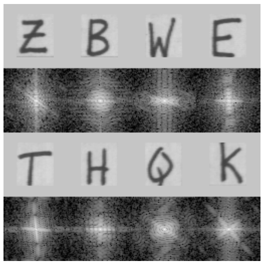

(a) (b) (c)

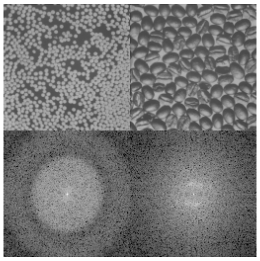
(d) (e) (f)

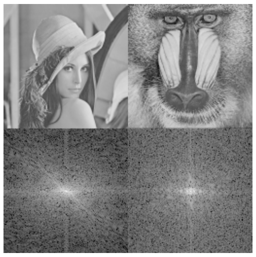




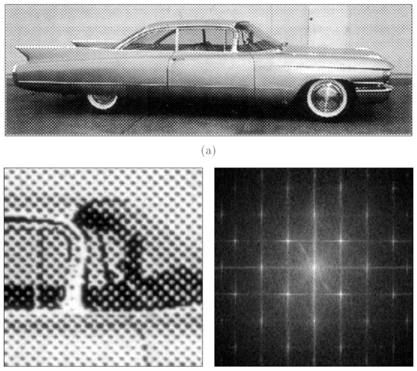
- notice a bright band going to high frequencies perpendicular to the strong edges in the image
- Anytime an image has a strong-contrast, sharp edge the gray values must change very rapidly. It takes lots of high frequency power to follow such an edge so there is usually such a line in its magnitude spectrum.







Print patterns



(a)

(b)

(c)

Linear Filters in Frequency space

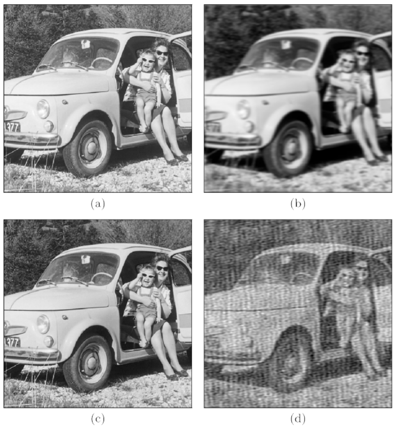
$$\begin{tikzcd}
\text{Image space:} & g(u,v) \arrow[d] & * & h(u,v) \arrow[d] & = & g'(u,v) \\
& \mathsf{DFT} \arrow[d] & & \mathsf{DFT} \arrow[d] & & \mathsf{DFT}^{-1} \arrow[u] \\
\text{Frequency space:} & G(m,n) & \cdot & H(m,n) \arrow[rr] & & G'(m,n) \arrow[u]
\end{tikzcd}$$

Inverse Filters - De-convolution

- How can we remove the effect of a filter ?

$$g_{\text{blur}} = g_{\text{orig}} * h_{\text{blur}}$$

$$G_{\text{blur}} = G_{\text{orig}} \cdot H_{\text{blur}}$$



(a) (b) (c) (d)

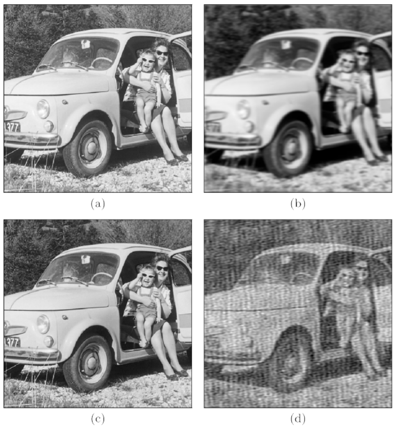
Inverse Filters - De-convolution

- How can we remove the effect of a filter ?

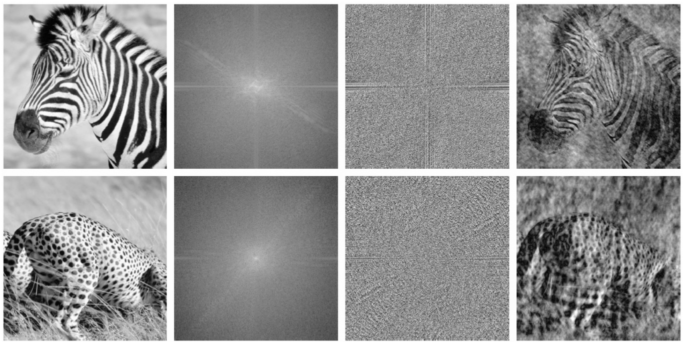
$$g_{\text{blur}} = g_{\text{orig}} * h_{\text{blur}}$$

$$G_{\text{blur}} = G_{\text{orig}} \cdot H_{\text{blur}}$$

$$G_{\text{orig}}(m,n) = \frac{G_{\text{blur}}(m,n)}{H_{\text{blur}}(m,n)}$$



(a) (b) (c) (d)



- What happens if we swap the magnitude spectra ?
- Phase spectrum holds the spatial information (where things are),
- Phase spectrum is more important for perception than magnitude spectrum.

The Discrete Cosine Transform (DCT)

- FT and DFT are designed for processing complex-valued signal and always produce a complex-valued spectrum.
- For a real-valued signal, the Fourier spectrum is symmetric
- Discrete Cosine Transform (DCT): similar to DFT but does not work with complex signals.
- DCT uses cosine functions only, with various wave numbers as the basis functions and operates on real-valued signals

One dimensional DCT

- DCT:

$$G(m) = \sqrt{\tfrac{2}{M}} \sum_{u=0}^{M-1} g(u) \cdot c_m \cos\left(\pi \tfrac{m(2u+1)}{2M}\right)$$

$$c_m = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } m = 0 \\ 1 & \text{otherwise} \end{cases}$$

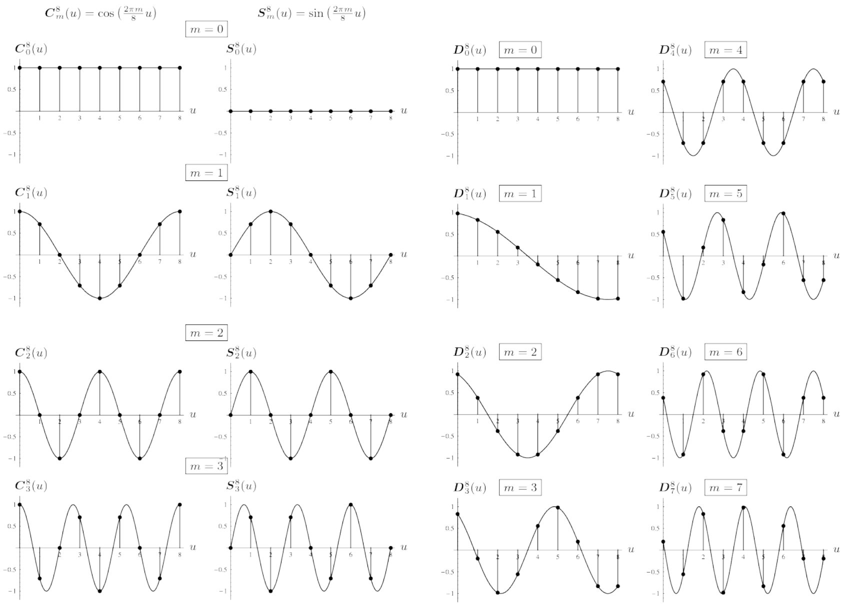
- Inverse DCT

$$g(u) = \sqrt{\tfrac{2}{M}} \sum_{m=0}^{M-1} G(m) \cdot c_m \cos\left(\pi \tfrac{m(2u+1)}{2M}\right)$$

DCT functions has half the period and shifted by 0.5

$$\text{DFT: } \boldsymbol{C}_m^M(u) = \cos\bigl(2\pi \tfrac{mu}{M}\bigr),$$
$$\text{DCT: } \boldsymbol{D}_m^M(u) = \cos\bigl(\pi \tfrac{m(2u+1)}{2M}\bigr) = \cos\bigl(2\pi \tfrac{m(u+0.5)}{2M}\bigr)$$

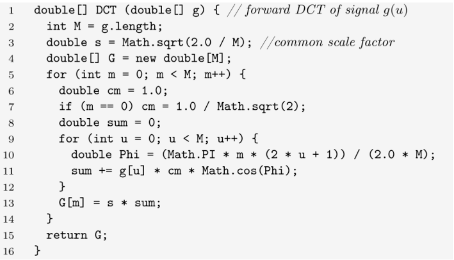
- DC component: the coefficient of the D_0
- AC components: the rest of the coefficients



$$\text{DFT: } \boldsymbol{C}_m^M(u) = \cos\bigl(2\pi \tfrac{mu}{M}\bigr),$$
$$\text{DCT: } \boldsymbol{D}_m^M(u) = \cos\bigl(\pi \tfrac{m(2u+1)}{2M}\bigr) = \cos\bigl(2\pi \tfrac{m(u+0.5)}{2M}\bigr)$$

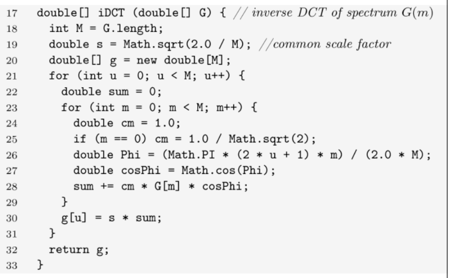
Implementing DCT

$$G(m) = \sqrt{\tfrac{2}{M}} \sum_{u=0}^{M-1} g(u) \cdot c_m \cos\left(\pi \tfrac{m(2u+1)}{2M}\right)$$



```
double[] DCT (double[] g) { // forward DCT of signal g(u)
  int M = g.length;
  double s = Math.sqrt(2.0 / M); //common scale factor
  double[] G = new double[M];
  for (int m = 0; m < M; m++) {
    double cm = 1.0;
    if (m == 0) cm = 1.0 / Math.sqrt(2);
    double sum = 0;
    for (int u = 0; u < M; u++) {
      double Phi = (Math.PI * m * (2 * u + 1)) / (2.0 * M);
      sum += g[u] * cm * Math.cos(Phi);
    }
    G[m] = s * sum;
  }
  return G;
}
```

$$g(u) = \sqrt{\tfrac{2}{M}} \sum_{m=0}^{M-1} G(m) \cdot c_m \cos\left(\pi \tfrac{m(2u+1)}{2M}\right)$$



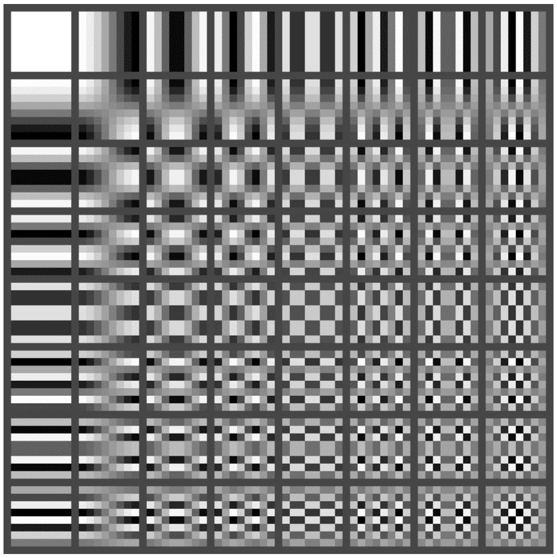
```
double[] iDCT (double[] G) { // inverse DCT of spectrum G(m)
  int M = G.length;
  double s = Math.sqrt(2.0 / M); //common scale factor
  double[] g = new double[M];
  for (int u = 0; u < M; u++) {
    double sum = 0;
    for (int m = 0; m < M; m++) {
      double cm = 1.0;
      if (m == 0) cm = 1.0 / Math.sqrt(2);
      double Phi = (Math.PI * (2 * u + 1) * m) / (2.0 * M);
      double cosPhi = Math.cos(Phi);
      sum += cm * G[m] * cosPhi;
    }
    g[u] = s * sum;
  }
  return g;
}
```

2D DCT

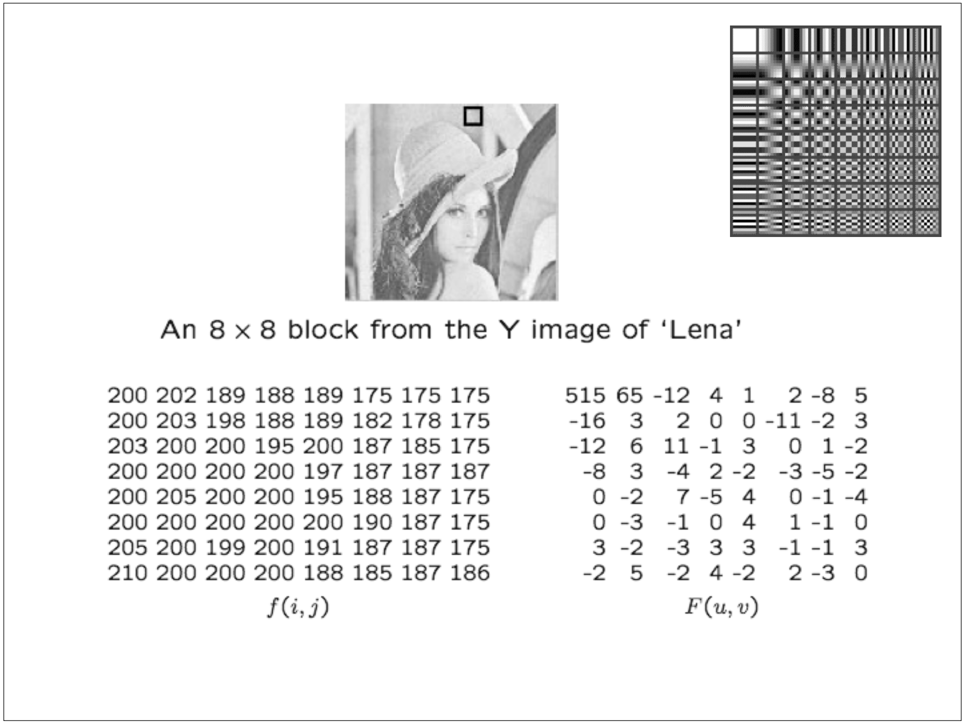
$$\begin{aligned} G(m,n) &= \frac{2}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} g(u,v) \cdot c_m \cos\left(\tfrac{\pi(2u+1)m}{2M}\right) \cdot c_n \cos\left(\tfrac{\pi(2v+1)n}{2N}\right) \\ &= \frac{2c_m c_n}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} g(u,v) \cdot \boldsymbol{D}_m^M(u) \cdot \boldsymbol{D}_n^N(v) \end{aligned}$$

$$\begin{aligned} g(u,v) &= \frac{2}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} G(m,n) \cdot c_m \cos\left(\tfrac{\pi(2u+1)m}{2M}\right) \cdot c_n \cos\left(\tfrac{\pi(2v+1)n}{2N}\right) \\ &= \frac{2}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} G(m,n) \cdot c_m \boldsymbol{D}_m^M(u) \cdot c_n \boldsymbol{D}_n^N(v) \end{aligned}$$

$$0 \le u < M,\ 0 \le v < N$$



- DC component: the coefficient of the $D_{0,0}$
- AC components: the rest of the coefficients



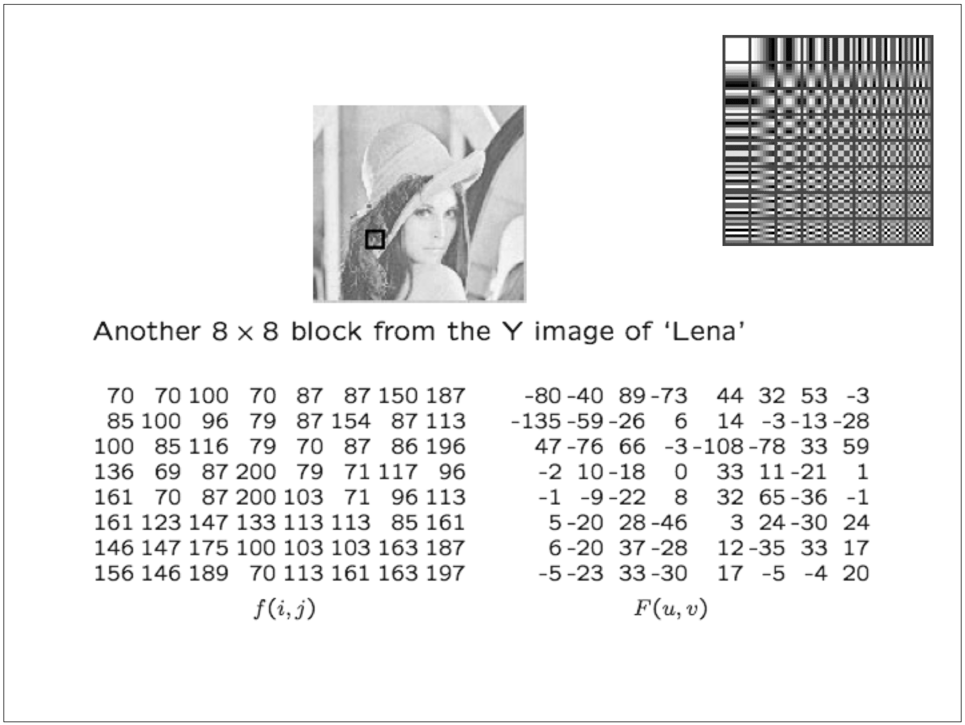
An 8 × 8 block from the Y image of 'Lena'

200	202	189	188	189	175	175	175
200	203	198	188	189	182	178	175
203	200	200	195	200	187	185	175
200	200	200	200	197	187	187	187
200	205	200	200	195	188	187	175
200	200	200	200	200	190	187	175
205	200	199	200	191	187	187	175
210	200	200	200	188	185	187	186

$f(i,j)$

515	65	-12	4	1	2	-8	5
-16	3	2	0	0	-11	-2	3
-12	6	11	-1	3	0	1	-2
-8	3	-4	2	-2	-3	-5	-2
0	-2	7	-5	4	0	-1	-4
0	-3	-1	0	4	1	-1	0
3	-2	-3	3	3	-1	-1	3
-2	5	-2	4	-2	2	-3	0

$F(u,v)$



Another 8 × 8 block from the Y image of 'Lena'

70	70	100	70	87	87	150	187
85	100	96	79	87	154	87	113
100	85	116	79	70	87	86	196
136	69	87	200	79	71	117	96
161	70	87	200	103	71	96	113
161	123	147	133	113	113	85	161
146	147	175	100	103	103	163	187
156	146	189	70	113	161	163	197

$f(i,j)$

-80	-40	89	-73	44	32	53	-3
-135	-59	-26	6	14	-3	-13	-28
47	-76	66	-3	-108	-78	33	59
-2	10	-18	0	33	11	-21	1
-1	-9	-22	8	32	65	-36	-1
5	-20	28	-46	3	24	-30	24
6	-20	37	-28	12	-35	33	17
-5	-23	33	-30	17	-5	-4	20

$F(u,v)$

Separability

- 2D DCT can be implemented as two 1D DCTs

$$G(m,n) = \sqrt{\tfrac{2}{N}} \sum_{v=0}^{N-1} \Bigg[\underbrace{\sqrt{\tfrac{2}{M}} \sum_{u=0}^{M-1} g(u,v) \cdot c_m \boldsymbol{D}_m^M(u)}_{\text{one-dimensional } \mathsf{DCT}[g(\cdot,v)]} \Bigg] \cdot c_n \boldsymbol{D}_n^N(v)$$