CognitiveSystems

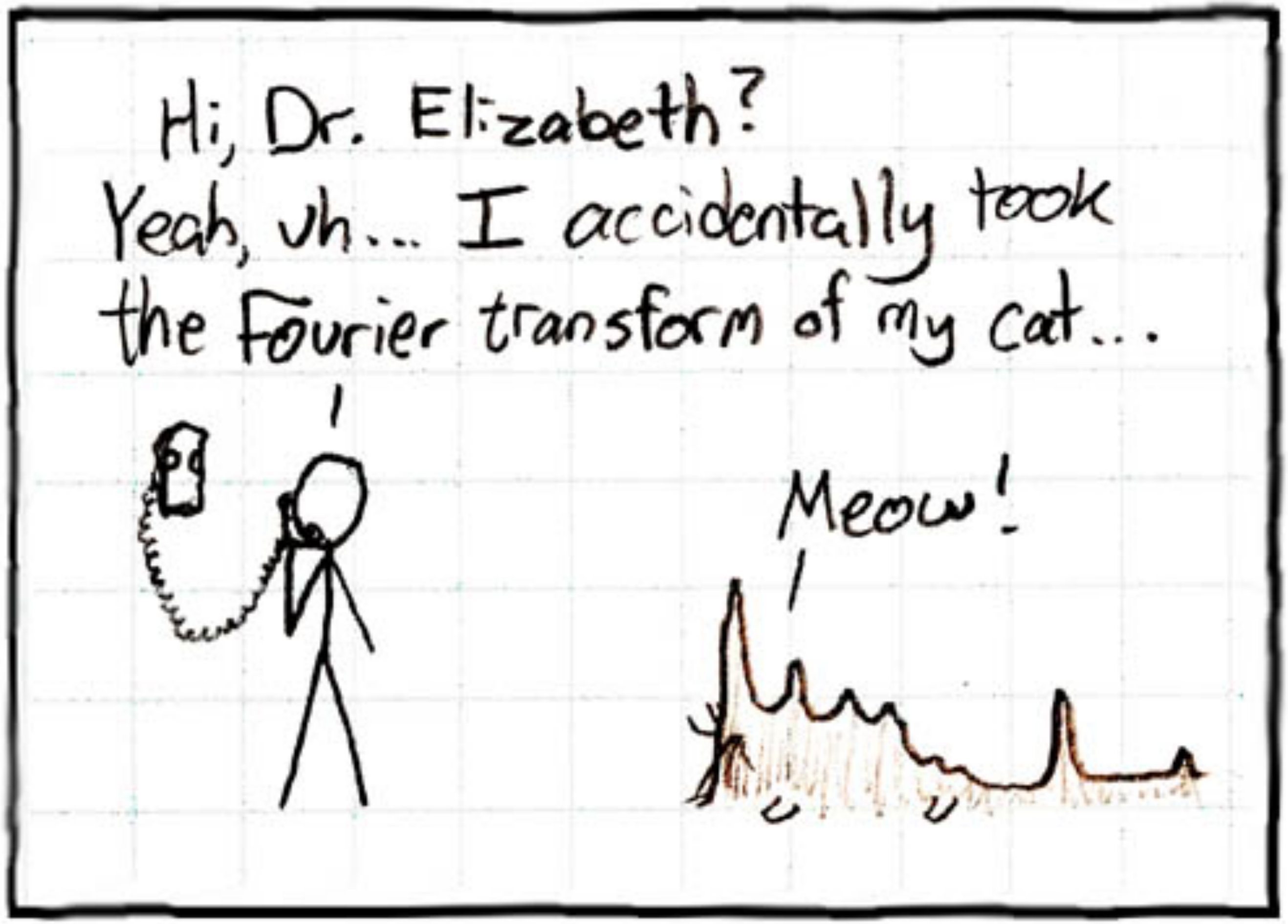
Fourier transform

Prof. Christian Wallraven
wallraven@korea.ac.kr

The message

- The Fourier transform is nothing magic
- First, it is a transform, that is, it changes the representation of the data
- Second, this transform is easily reversible, that is, one can go back and forth between the representations
- The standard representation of data is in space or time
  - You measure data as a function of position, or time
- The Fourier representation of data is in frequencies
  - How much is a certain frequency present in the data?
- It therefore is most useful for understanding periodic data, that is data that contains re-occurring patterns

xkcd


Hi, Dr. Elizabeth?
Yeah, uh... I accidentally took the Fourier transform of my cat...
Meow!

Mathematical Background: Complex Numbers

CognitiveSystems

- A complex number **x** has the form:

$$x = a + jb, \text{ where } j = \sqrt{-1}$$

a: **real part**, b: **imaginary** part

- Addition is done by adding real and imaginary parts separately

$$(a + jb) + (c + jd) = (a + c) + j(b + d)$$

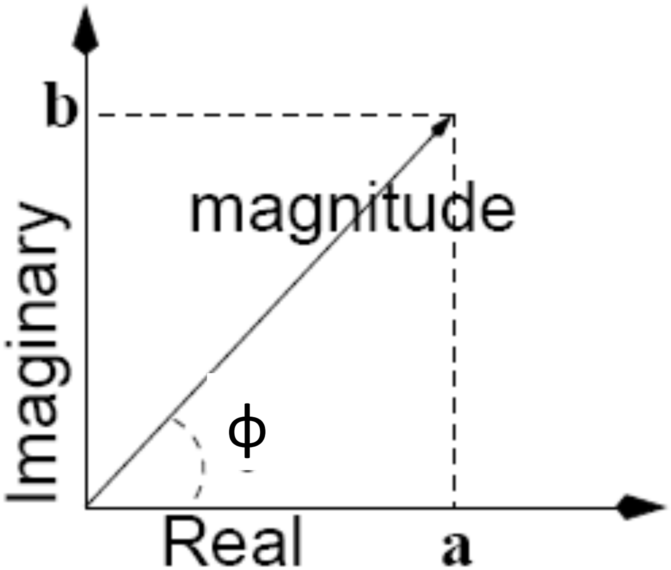
- Multiplication (taking into account the fact that j*j = -1)

$$(a + jb) \,.\, (c + jd) = (ac - bd) + j(ad + bc)$$

Mathematical Background: Complex Numbers

CognitiveSystems

- Magnitude-Phase (i.e., vector) representation of complex numbers



Magnitude: $|x| = r = \sqrt{a^2 + b^2}$

Phase: $\phi(x) = \tan^{-1}(b/a)$

Polar notations:

$$x = r(\cos\phi + j\sin\phi)$$

$$\boxed{x = |x|e^{j\phi(x)}}$$

Mathematical Background: Complex Numbers

CognitiveSystems

- Euler's formula

$$e^{\pm j\theta} = cos(\theta) \pm jsin(\theta)$$

- Properties

$$e^{j\pi} + 1 = 0$$

$$|e^{\pm j\theta}| = \sqrt{\cos^2(\theta) + \sin^2(\theta)} = 1$$

$$\phi(e^{\pm j\theta}) = \tan^{-1}(\pm \frac{sin(\theta)}{cos(\theta)}) = \tan^{-1}(\pm tan(\theta)) = \pm\theta$$

$$sin(\theta) = \frac{1}{2j}(e^{j\theta} - e^{-j\theta})$$

$$cos(\theta) = \frac{1}{2}(e^{j\theta} + e^{-j\theta})$$

Mathematical Background: Complex Numbers

- Multiplication using magnitude-phase representation

$$xy = |x|e^{j\phi(x)} \,.\, |y|e^{j\phi(y)} = |x|\,|y|\,e^{j(\phi(x)+\phi(y))}$$

- Complex conjugate

$$x^* = a - jb$$

- Properties

$$|x| = |x^*|$$
$$\phi(x) = -\phi(x^*)$$
$$xx^* = |x|^2$$

Mathematical Background: Sine and Cosine Functions

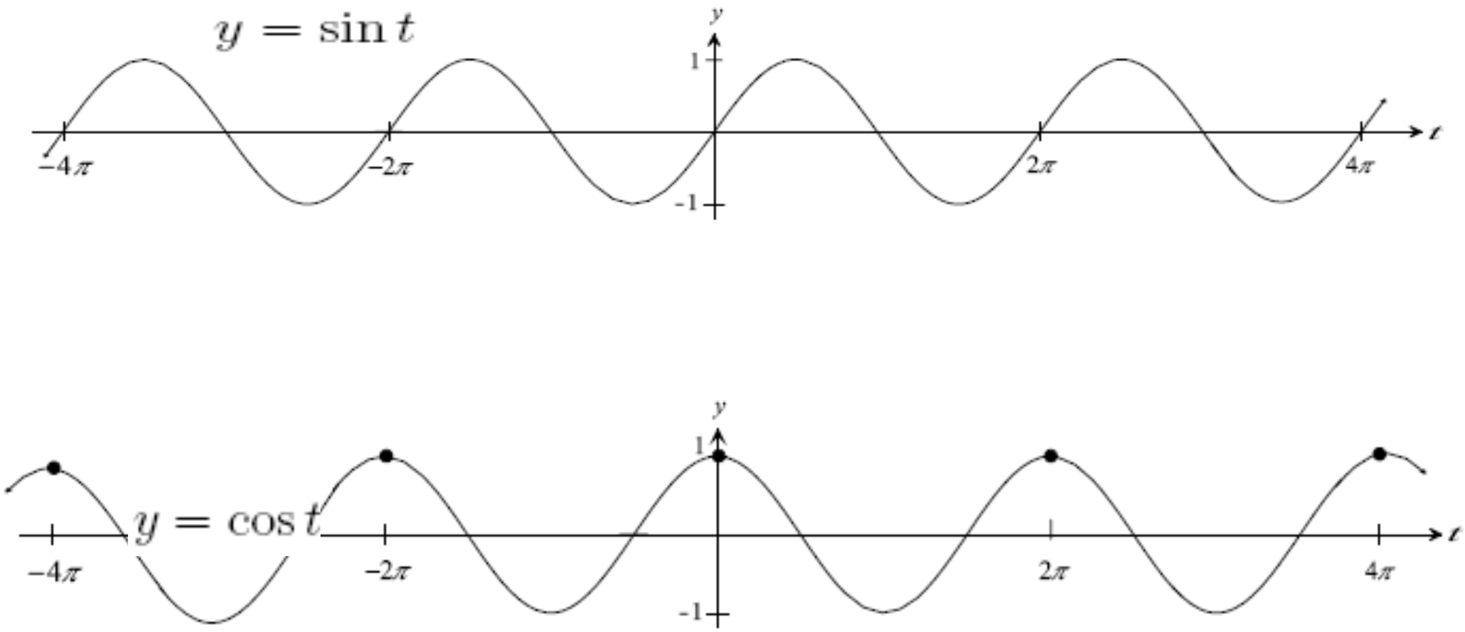
- Sine and Cosine functions are periodic
- As we have seen, the general form of sine and cosine functions are:

$$y\,(t) = A\sin[a(t+b)] \qquad y\,(t) = A\cos[a(t+b)]$$

$\lvert A\rvert$	amplitude
$\frac{2\pi}{\lvert a\rvert}$	period
b	phase shift

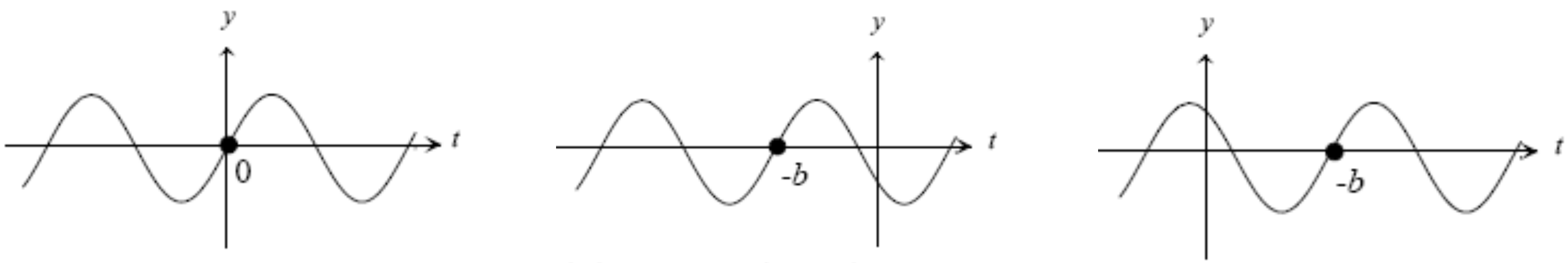
Mathematical Background:
Sine and Cosine Functions

Special case: unit amplitude, no phase shift, unit period
A=1, b=0, α=1



Mathematical Background: Sine and Cosine Functions

- Shifting or translating the sine function by a constant b (a phase shift)



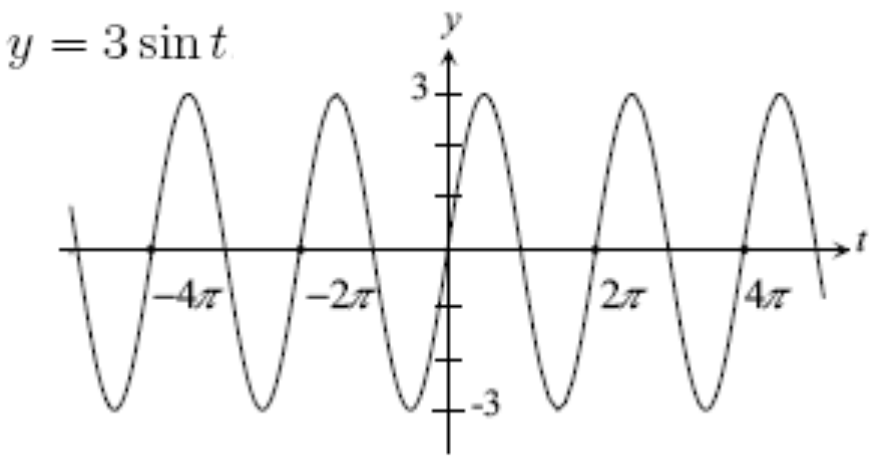
(a) $y = \sin t$ (b) $y = \sin(t+b)$, $b > 0$ (c) $y = \sin(t+b)$, $b < 0$

- We see, therefore, that the cosine is merely a shifted sine function:

$$\cos(t) = \sin(t + \frac{\pi}{2})$$

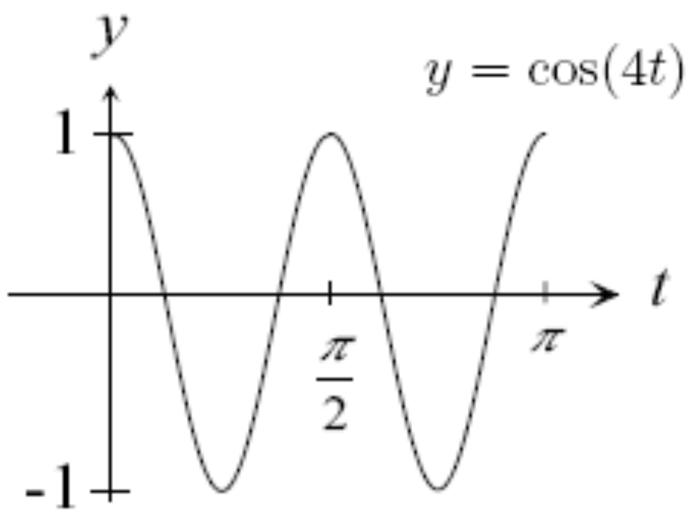
Mathematical Background: Sine and Cosine Functions

- Effect of changing the amplitude A



Mathematical Background: Sine and Cosine Functions

- Effect of changing the period $T=2\pi/|\alpha|$, for example, for $y=\cos(\alpha t)$



$\alpha = 4$

period $2\pi/4=\pi/2$

shorter period
higher frequency
(i.e., the function oscillates **faster**)

- Frequency is defined as $f=1/T$
- Frequency notation: $\sin(\alpha t)=\sin(2\pi t/T)=\sin(2\pi f t)$

Fourier Series Theorem

- Any periodic signal can be expressed as a weighted (infinite) sum of sine and cosine functions of varying frequency

$$\boxed{\frac{a_0}{2} + \sum_{n=1}^{\infty} [a_n \cos(nx) + b_n \sin(nx)]}$$

a_n and b_n, are the weights of the expansion

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(nx)\, dx, \quad n \geq 0$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(nx)\, dx, \quad n \geq 1$$

Fourier Series Theorem

- Any periodic signal can be expressed as a weighted (infinite) sum of sine and cosine functions of varying frequency (this is the compact notation):

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{inx}.$$

$$c_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) e^{-inx}\, dx.$$ are the weights of the expansion

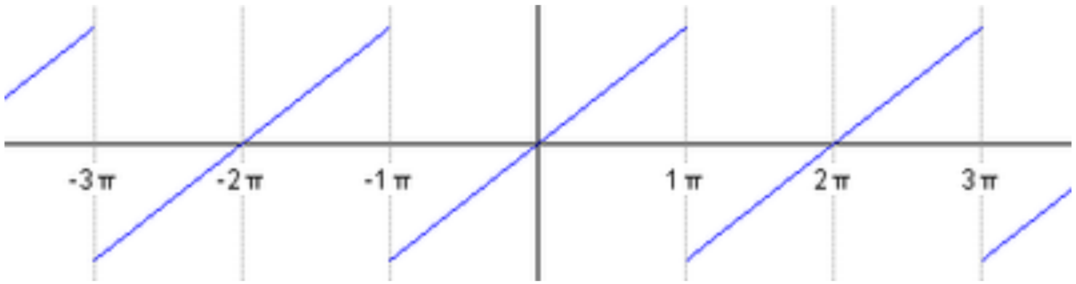
$$a_n = c_n + c_{-n} \text{ for } n = 0, 1, 2, \ldots,$$

$$b_n = i(c_n - c_{-n}) \text{ for } n = 1, 2, \ldots$$

Example – the sawtooth wave

- The sawtooth wave is defined as:

$$f(x) = x, \quad \text{for} -\pi < x < \pi,$$
$$f(x + 2\pi) = f(x), \quad \text{for} -\infty < x < \infty.$$



- The coefficients are therefore:

$$a_n = \frac{1}{\pi}\int_{-\pi}^{\pi} x\cos(nx)\,dx = 0, \quad n \geq 0.$$
$$b_n = \frac{1}{\pi}\int_{-\pi}^{\pi} x\sin(nx)\,dx = -\frac{2}{n}\cos(n\pi) + \frac{2}{n^2\pi}\sin(n\pi) = 2\,\frac{(-1)^{n+1}}{n}, \quad n \geq 1.$$

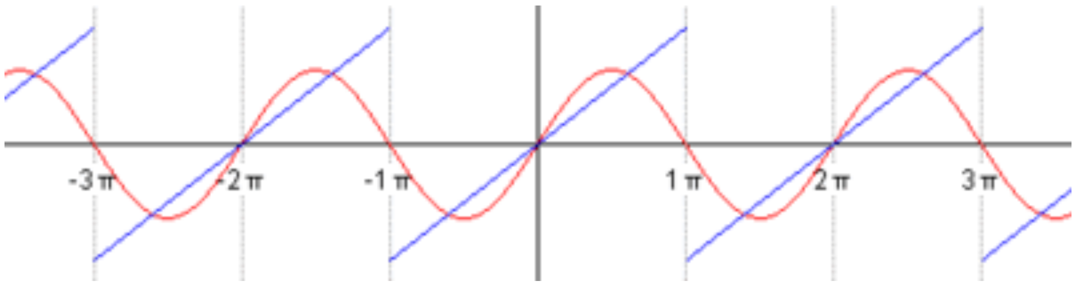
Example – the sawtooth wave

CognitiveSystems

- With that the Fourier series becomes:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[a_n \cos(nx) + b_n \sin(nx) \right]$$
$$= 2 \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{n} \sin(nx), \quad \text{for} \quad x - \pi \notin 2\pi \mathbf{Z}.$$

- This is an animated plot of the first five approximations of the Fourier series



Fourier series – applications

- The Fourier series is mainly used for solving partial differential equations
- This is also the origin of the work of Fourier who used the Fourier series to describe possible solutions to the heat equation

$$\frac{\partial u}{\partial t} - \alpha \Delta u = 0$$

where u is a function of x,y,z,t

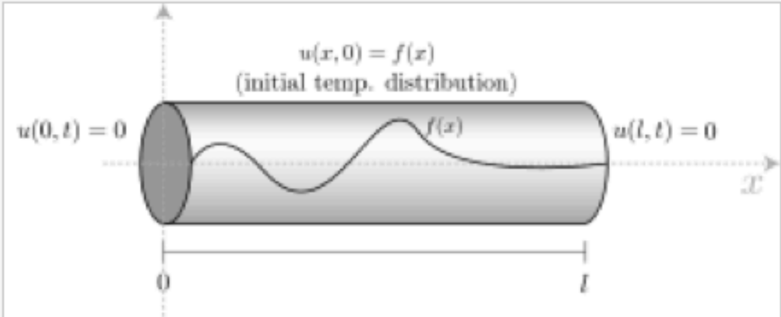
The heat equation

The following solution technique for the heat equation was proposed by Joseph Fourier in his treatise *Théorie analytique de la chaleur*, published in 1822. Let us consider the heat equation for one space variable. This could be used to model heat conduction in a rod. The equation is

$$u_t = \alpha u_{xx} \qquad (1)$$

where $u = u(x, t)$ is a function of two variables x and t. Here

- x is the space variable, so $x \in [0, L]$, where L is the length of the rod.
- t is the time variable, so $t \geq 0$.



Idealized physical setting for heat conduction in a rod with homogeneous boundary conditions.

We assume the initial condition

$$u(x,0) = f(x) \quad \forall x \in [0, L] \qquad (2)$$

where the function f is given, and the boundary conditions

$$u(0,t) = 0 = u(L,t) \quad \forall t > 0. \qquad (3)$$

The heat equation

Let us attempt to find a solution of (1) which is not identically zero satisfying the boundary conditions (3) but with the following property: *u* is a product in which the dependence of *u* on *x*, *t* is separated, that is:

$$u(x,t) = X(x)T(t). \qquad (4)$$

This solution technique is called separation of variables. Substituting *u* back into equation (1),

$$\frac{\dot{T}(t)}{\alpha T(t)} = \frac{X''(x)}{X(x)}.$$

Since the right hand side depends only on *x* and the left hand side only on *t*, both sides are equal to some constant value −λ. Thus:

$$\dot{T}(t) = -\lambda\alpha T(t) \qquad (5)$$

and

$$X''(x) = -\lambda X(x). \qquad (6)$$

The heat equation

CognitiveSystems

We will now show that nontrivial solutions for (6) for values of $\lambda \leq 0$ cannot occur:

1. Suppose that $\lambda < 0$. Then there exist real numbers B, C such that
$$X(x) = Be^{\sqrt{-\lambda}\,x} + Ce^{-\sqrt{-\lambda}\,x}.$$
From (3) we get $X(0) = 0 = X(L)$ and therefore $B = 0 = C$ which implies u is identically 0.
2. Suppose that $\lambda = 0$. Then there exist real numbers B, C such that $X(x) = Bx + C$. From equation (3) we conclude in the same manner as in 1 that u is identically 0.
3. Therefore, it must be the case that $\lambda > 0$. Then there exist real numbers A, B, C such that
$$T(t) = Ae^{-\lambda\alpha t}$$
and
$$X(x) = B\sin(\sqrt{\lambda}\,x) + C\cos(\sqrt{\lambda}\,x).$$
From (3) we get $C = 0$ and that for some positive integer n,
$$\sqrt{\lambda} = n\frac{\pi}{L}.$$

This solves the heat equation in the special case that the dependence of u has the special form (4).

The heat equation

This solves the heat equation in the special case that the dependence of *u* has the special form (4).

In general, the sum of solutions to (1) which satisfy the boundary conditions (3) also satisfies (1) and (3). We can show that the solution to (1), (2) and (3) is given by

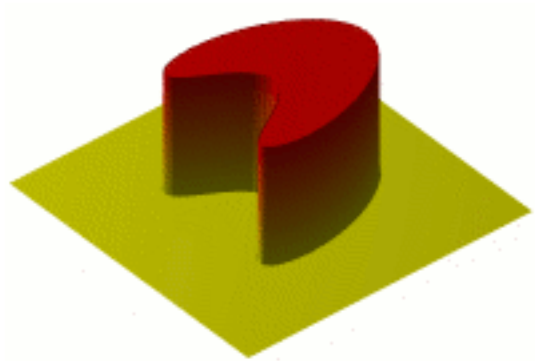
$$u(x,t) = \sum_{n=1}^{\infty} D_n \sin\left(\frac{n\pi x}{L}\right) e^{-\frac{n^2\pi^2\alpha t}{L^2}}$$

where

$$D_n = \frac{2}{L}\int_0^L f(x)\sin\left(\frac{n\pi x}{L}\right)\,dx.$$

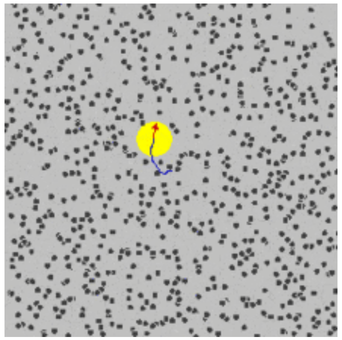
The heat equation

- Given the initial heating profile of a cool plate, the heat equation can be solved and predicts how temperature evolves over time
- Interestingly, the heat equation underlies also modelling of Brownian motion and of financial processes, such as the Black-Scholes model for option trading



Evolution of temperature on a heated plate

$$\frac{\partial \rho}{\partial t} = D \frac{\partial^2 \rho}{\partial x^2},$$



Diffusion equation for Brownian motion

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0$$

Black-Scholes equation

Fourier series to Fourier Transform

- Given a function that is zero outside a defined interval [-L/2,L/2], then for any T>=L, we can approximate the function by a Fourier series with coefficients

$$\hat{f}(n/T) = c_n = \int_{-T/2}^{T/2} e^{-2\pi i n x/T} f(x)\, dx$$

and the Fourier series is given by:

$$f(x) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \hat{f}(n/T) e^{2\pi i n x/T}.$$

- If we let ξ_n = n/T, and we let Δξ = (n + 1)/T − n/T = 1/T, then this last sum becomes

$$f(x) = \sum_{n=-\infty}^{\infty} \hat{f}(\xi_n) e^{2\pi i x \xi_n} \Delta\xi. \quad \xrightarrow{T \rightarrow \infty} \quad f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi)\, e^{2\pi i x \xi}\, d\xi,$$

Continuous Fourier Transform (FT)

- Transforms a signal (i.e., function) from the spatial domain (x) to the time/frequency domain (u or t).

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

Inverse FT: $$F^{-1}(F(u)) = f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du$$

with $e^{\pm j\theta} = cos(\theta) \pm jsin(\theta)$

Definitions

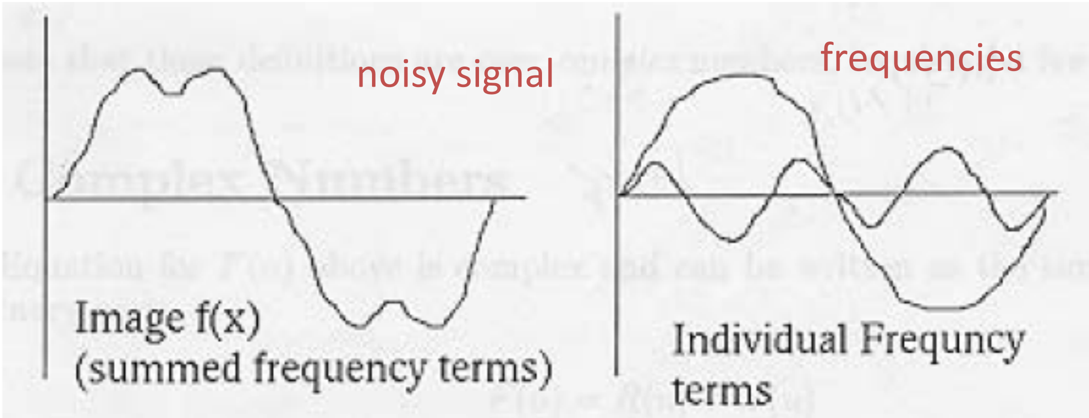
CognitiveSystems

- The Fourier Transform (FT) **F(u)** is a complex function: $$F(u) = R(u) + jI(u)$$
- Magnitude of FT (spectrum): $$|F(u)| = \sqrt{R^2(u) + I^2(u)}$$
- Phase of FT: $$\phi(F(u)) = \tan^{-1}\left(\frac{I(u)}{R(u)}\right)$$
- Magnitude-Phase representation: $$F(u) = |F(u)|e^{j\phi(u)}$$
- Power is the squared spectrum: **P(u)**=|F(u)|^2= $R^2(u) + I^2(u)$

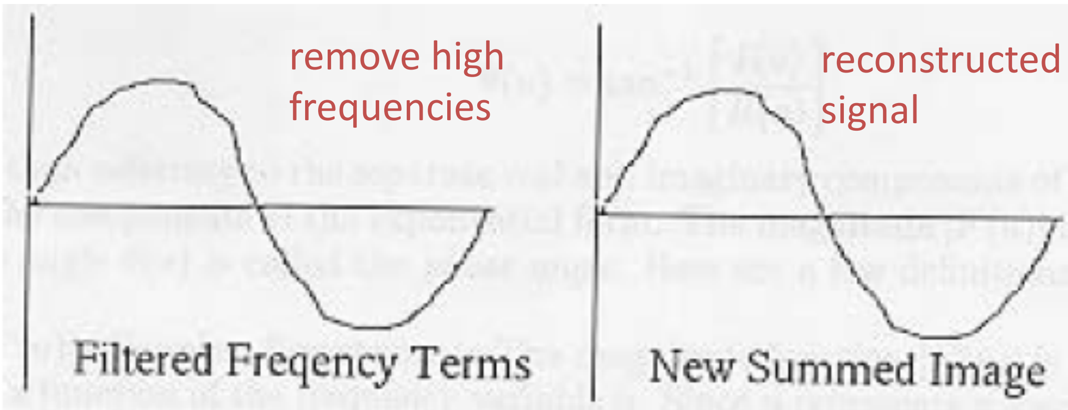
Typical applications of the FT

- Analyze frequency content of signals (EEG)
- Filtering: Remove frequencies from a signal
  - High-pass filtering
  - Low-pass filtering
- Easier and faster to perform certain operations in the **frequency** domain than in the **spatial** domain
  - Convolution
- Solution of certain numerical problems easier in the frequency domain

Example: Removing undesirable frequencies



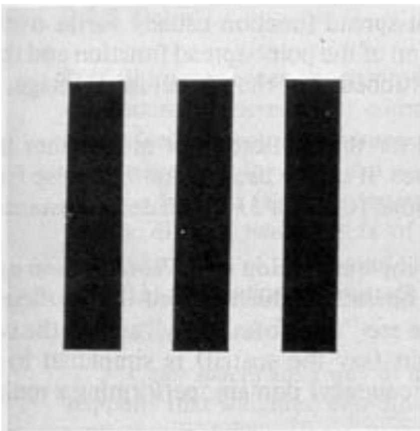
To remove certain frequencies, set their corresponding F(u) coefficients to zero!



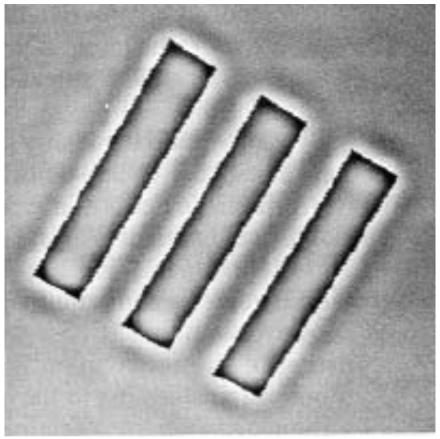
How do frequencies show up in an image?

CognitiveSystems

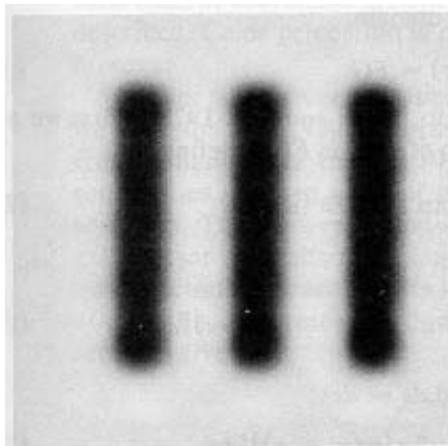
- High frequencies correspond to quickly varying information (e.g., edges)
- Low frequencies correspond to slowly varying information (e.g., continuous surface)



Original Image

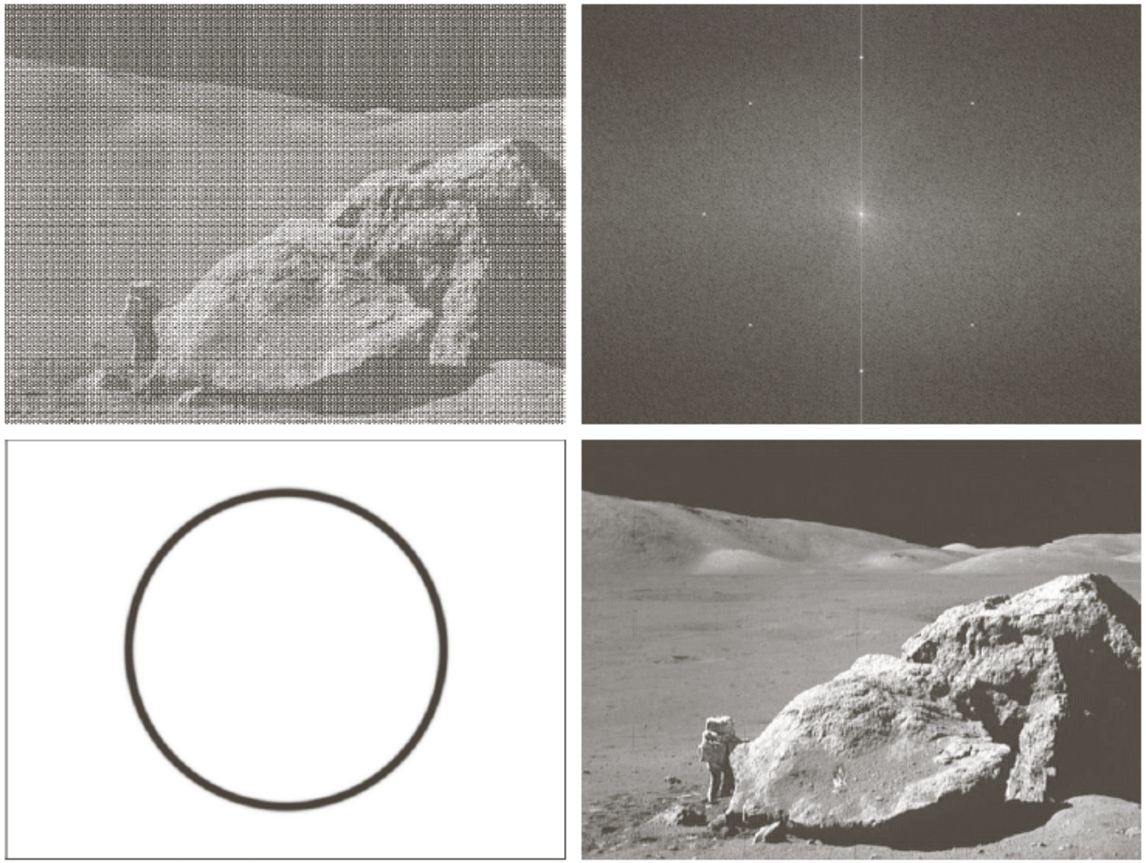


High-passed



Low passed

Example of noise reduction using FT



Frequency Filtering Steps

- 1. Take the FT of f(x):

$$F(f(x))$$

- 2. Remove undesired frequencies:

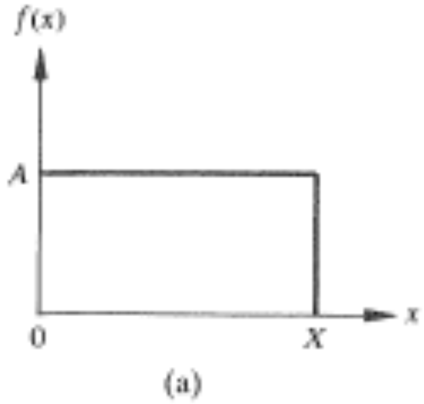
$$D(F(f(x)))$$

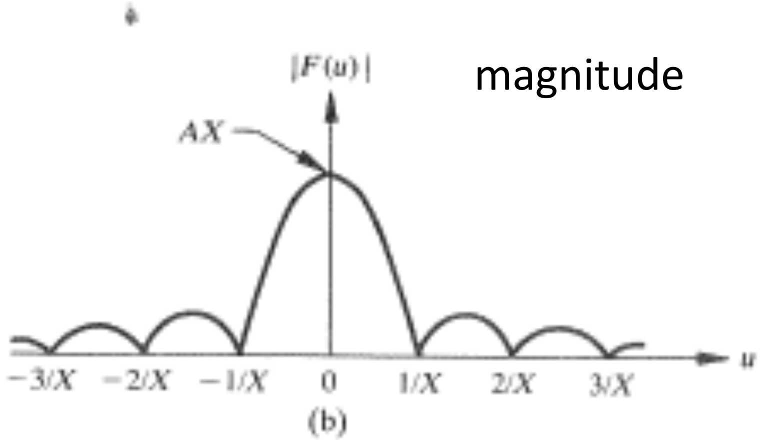
- 3. Convert back to a signal:

$$\hat{f}(x) = F^{-1}(D(F(f(x))))$$

We'll talk more about this later .....

Example: rectangular pulse





magnitude

rect(x) function

sinc(x)=sin(x)/x

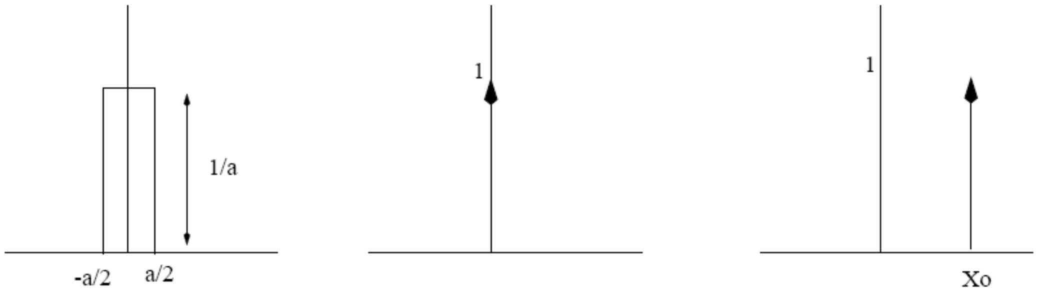
Example: impulse or “delta” function

- Definition of delta function:

$$\delta(x) = \lim_{a \to 0} \frac{1}{a} rect(x/a)$$

- Properties:

$$\int_{-\infty}^{\infty} \delta(x)dx = 1 \qquad \int_{-\infty}^{\infty} f(x)\delta(x - x_0)dx = f(x_0)$$

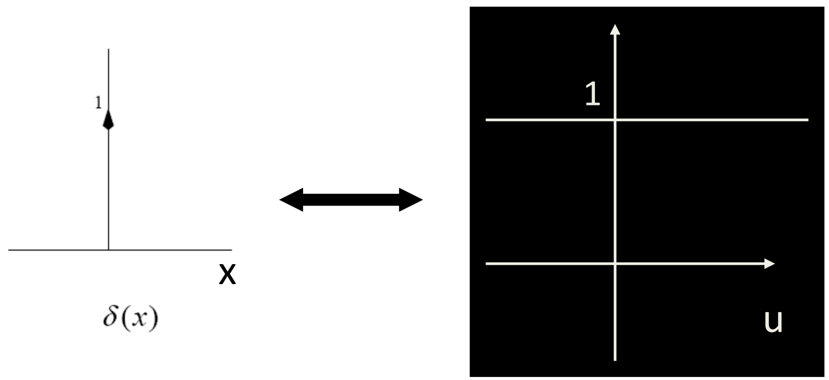


$\frac{1}{a} rect(x/a)$ $\qquad \delta(x)$ $\qquad \delta(x - x_0)$

Example: impulse or “delta” function

- FT of delta function

$$F(\delta(x)) = \int_{-\infty}^{\infty} \delta(x) e^{-j2\pi ux} dx = e^{0} = 1$$



Example: spatial/frequency shifts

CognitiveSystems

$$f(x) \Leftrightarrow F(u), \quad then$$

Special Cases

$$(1) \;\; f(x - x_0) \Leftrightarrow e^{-j2\pi u x_0} F(u)$$

$$\delta(x - x_0) \Leftrightarrow e^{-j2\pi u x_0}$$

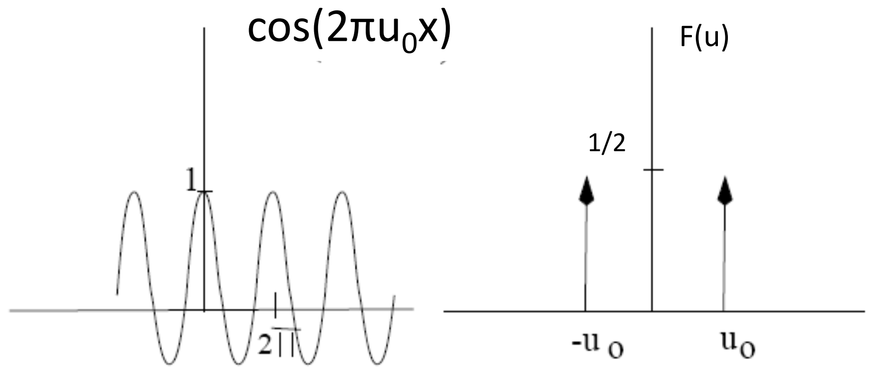
$$(2) \;\; f(x) e^{j2\pi u_0 x} \Leftrightarrow F(u - u_0)$$

$$e^{j2\pi u_0 x} \Leftrightarrow \delta(u - u_0)$$

Example: sine and cosine functions

- FT of the cosine function

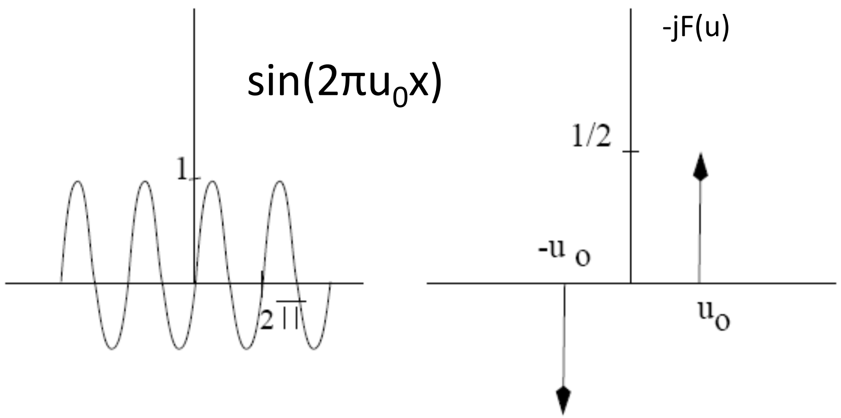
$$F(\cos(2\pi u_0 x)) = \frac{1}{2}\left[\delta(u-u_0)+\delta(u+u_0)\right]$$



Example: sine and cosine functions

- FT of the sine function

$$F(\sin(2\pi u_0 x)) = \frac{j}{2}[\delta(u+u_0) - \delta(u-u_0)]$$



Extending FT to two dimensions

CognitiveSystems

- Forward FT

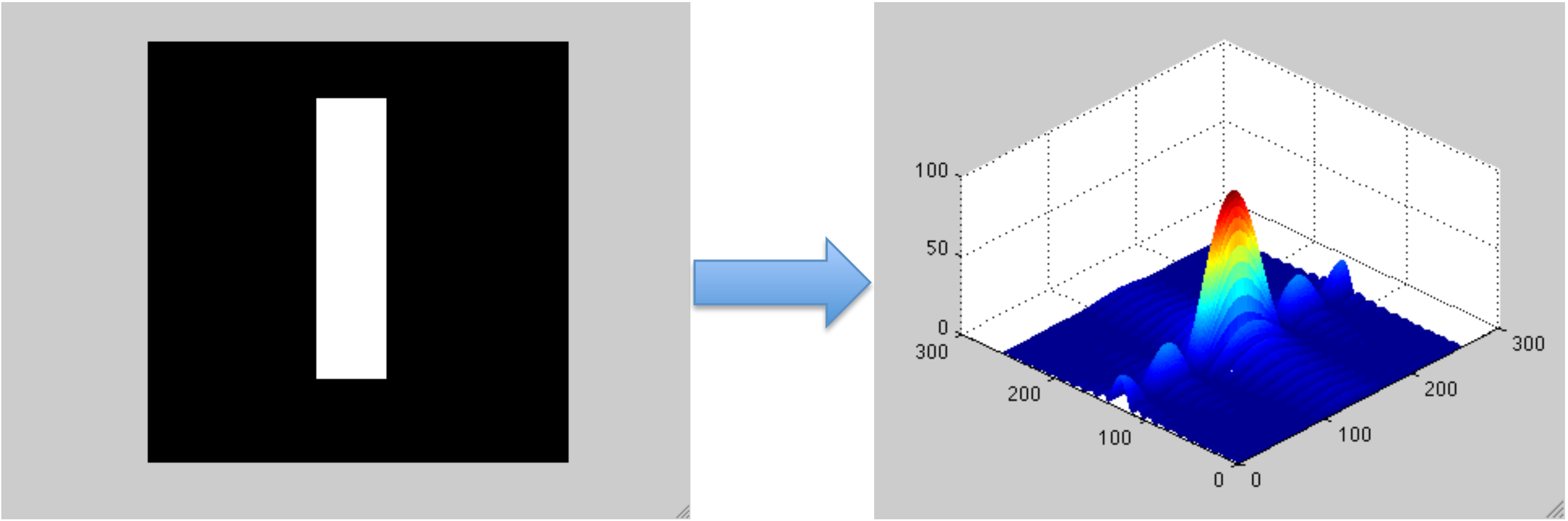
$$\mathcal{F}(f(x,y)) = F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}\,dxdy$$

- Inverse FT

$$\mathcal{F}^{-1}(F(u,v)) = f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(ux+uy)}\,dudv$$

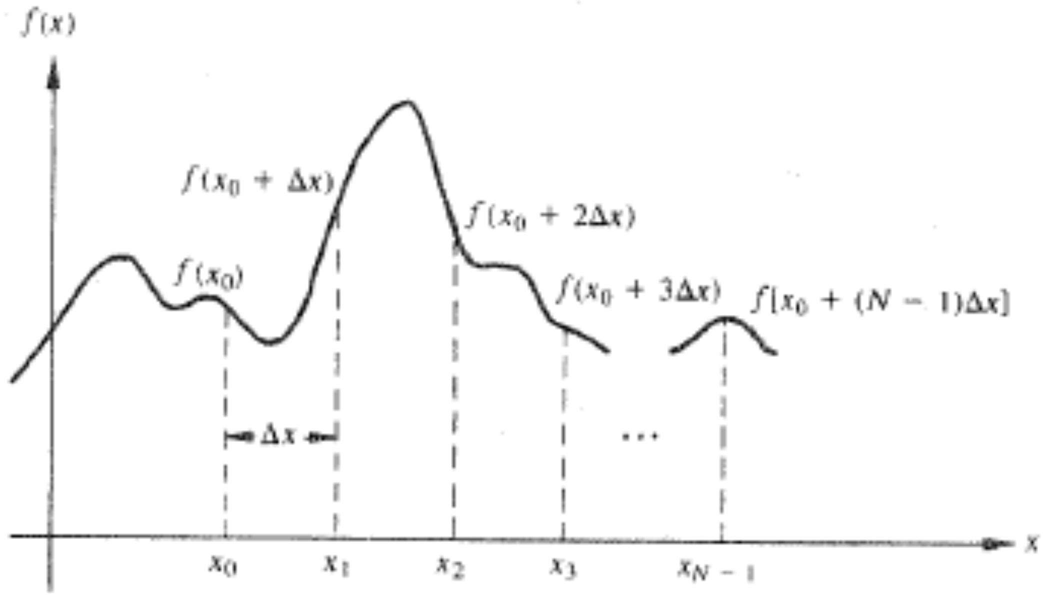
Example: 2D rectangle function

- FT of 2D rectangle function



Discrete Fourier Transform (DFT)

- In most applications, we have discrete data, that is, data that is sampled at discrete locations
  - Digitized time series (from an Analog-Digital-Converter)
  - Images



$$f(x) = f(x_0 + x\Delta x),\ x = 0, 1, \ldots, N-1$$

Discrete Fourier Transform (DFT)

CognitiveSystems

- Forward DFT

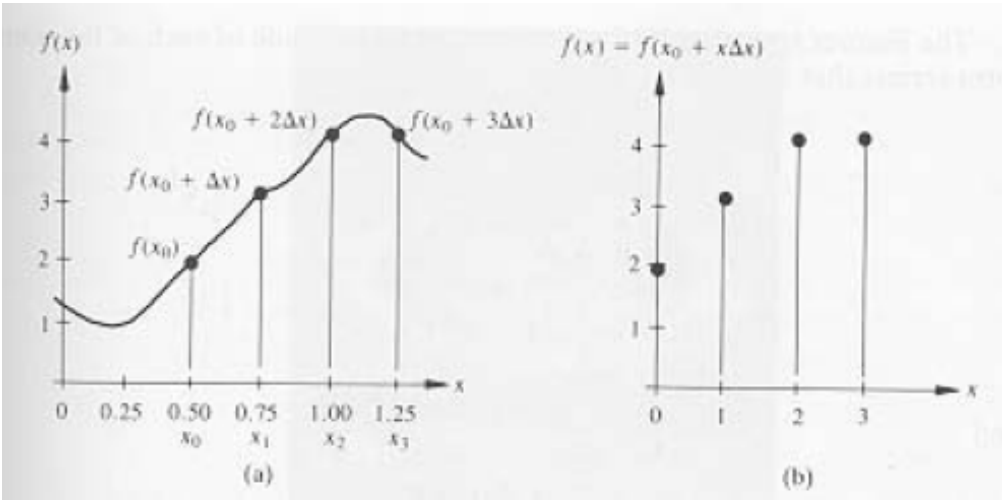
$$F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)e^{\frac{-j2\pi ux}{N}}, u = 0, 1, \ldots, N-1$$

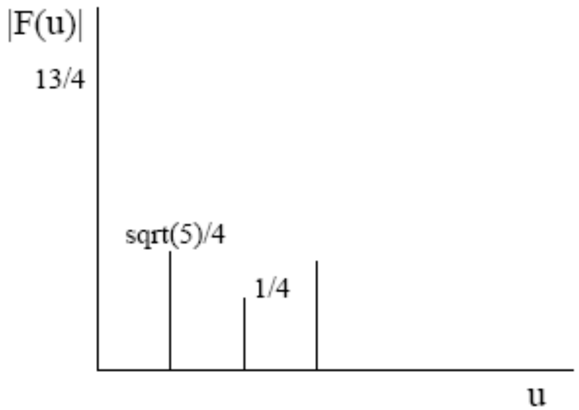
- Inverse DFT

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

$F(u)$ is discrete: $F(u) = F(u\Delta u)$, $u = 0, 1, \ldots, N-1$, $\Delta u =$ 1/NΔx

Example



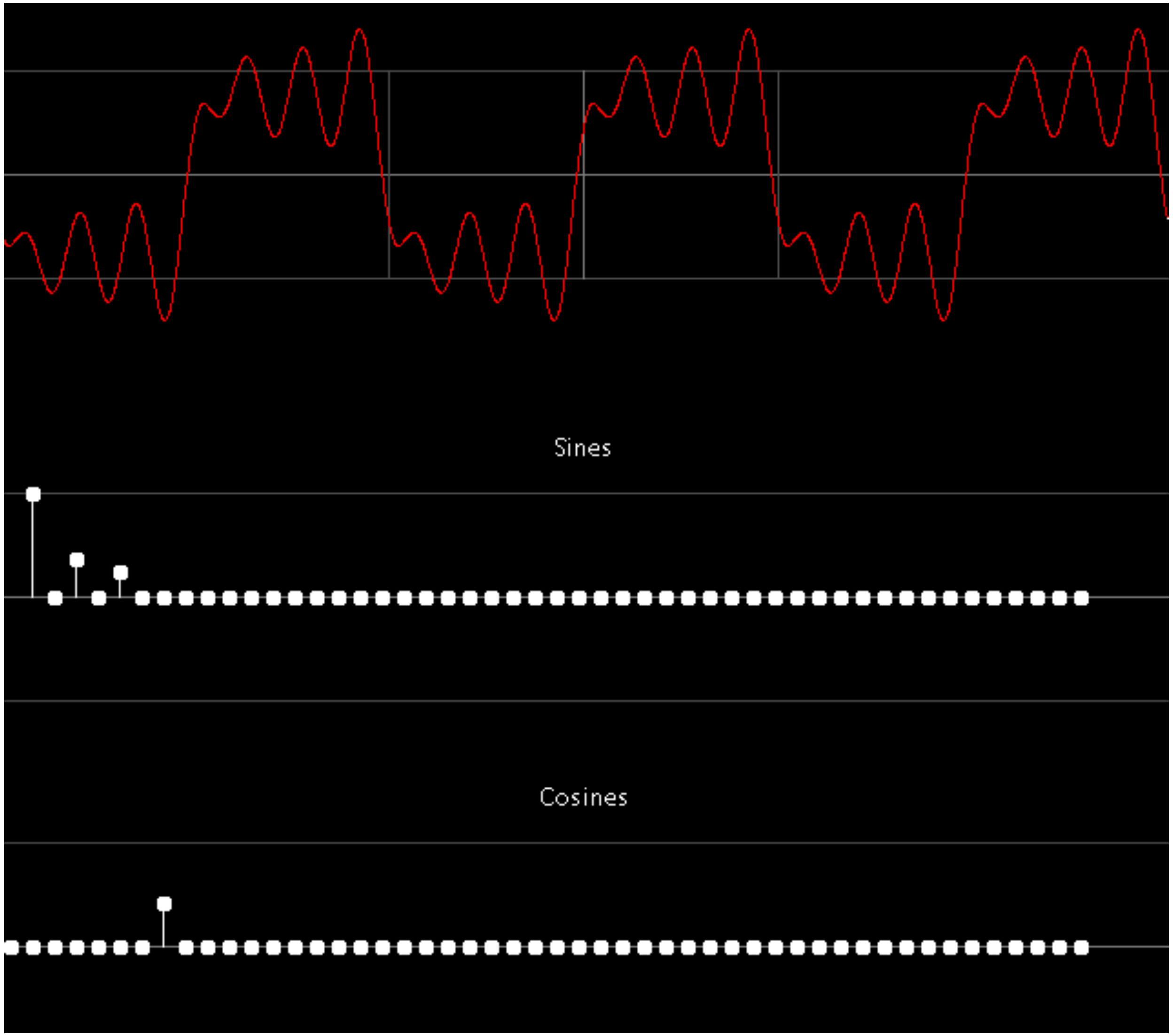


$N=4$

$\Delta x=0.25$

$$\Delta u=\frac{1}{N\Delta x}=1$$

Discrete Fourier Transform (DFT)



Extending DFT to 2D

CognitiveSystems

- Assume that f(x,y) is M x N image.

- Forward DFT

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

$$(u = 0, 1, \ldots, M-1, \; v = 0, 1, \ldots, N-1)$$

- Inverse DFT:

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

$$(x = 0, 1, \ldots, M-1, \; y = 0, 1, \ldots, N-1)$$

Extending DFT to 2D

- Special case: f(x,y) is N x N image.

- Forward DFT

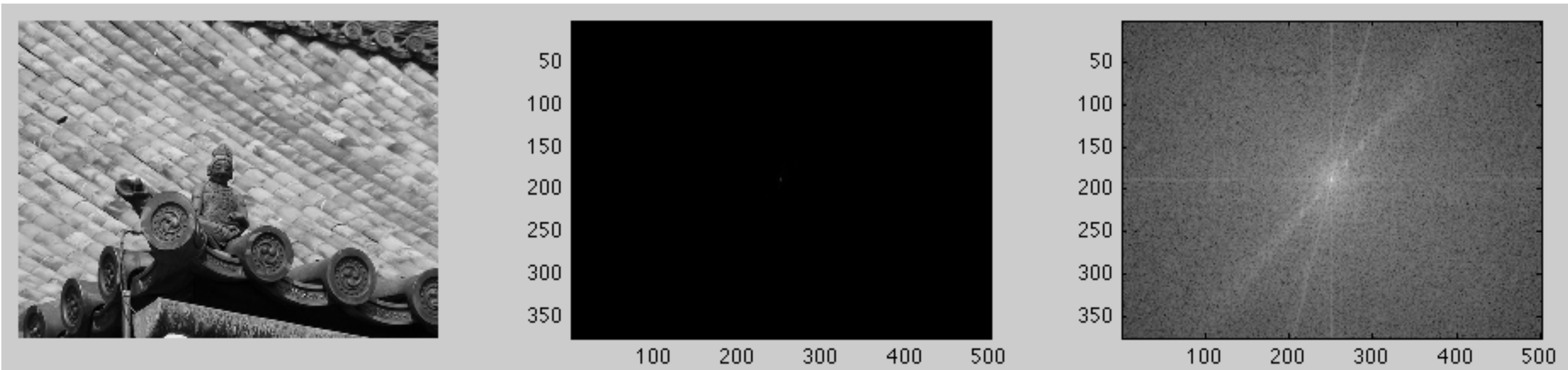
$$F(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(\frac{ux+vy}{N})},$$

- Inverse DFT

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

Visualizing DFT

- Typically, we look at the spectrum of F(u,v) = |F(u,v)|
  - In Matlab: plot(abs(F));
- The dynamic range of |F(u,v)| is typically very large
- Apply scaling: $D(u,v) = c\ log(1+|F(u,v)|)$ (c is a const)
  - In Matlab: plot(log(abs(F)));



original image before scaling after scaling

DFT Properties: (1) Separability

- The 2D DFT can be computed using 1D transforms only:
  - Forward DFT: $$F(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(\frac{ux+vy}{N})}$$
  - Inverse DFT: $$f(x,y) = \frac{1}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi(\frac{ux+vy}{N})}$$

DFT Properties: (1) Separability

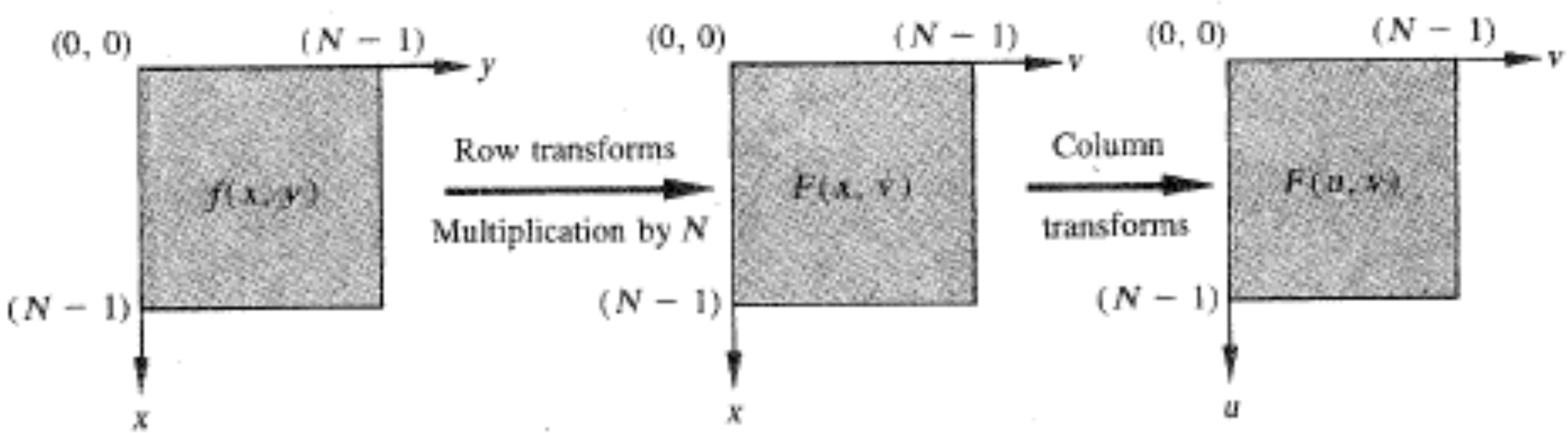
- Rewrite F(u,v) as follows:

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

- Let's set: $\sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(\frac{vy}{N})} = F(x,v) = N(\frac{1}{N}\sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(\frac{vy}{N})})$

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

DFT Properties: (1) Separability



DFT Properties: (2) Periodicity and Symmetry

CognitiveSystems

- The DFT and its inverse are periodic with period N

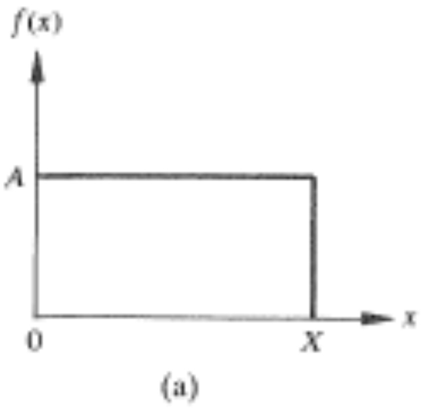
$$F(u,v) = F(u+N,v) = F(u,v+N) = F(u+N,v+N)$$

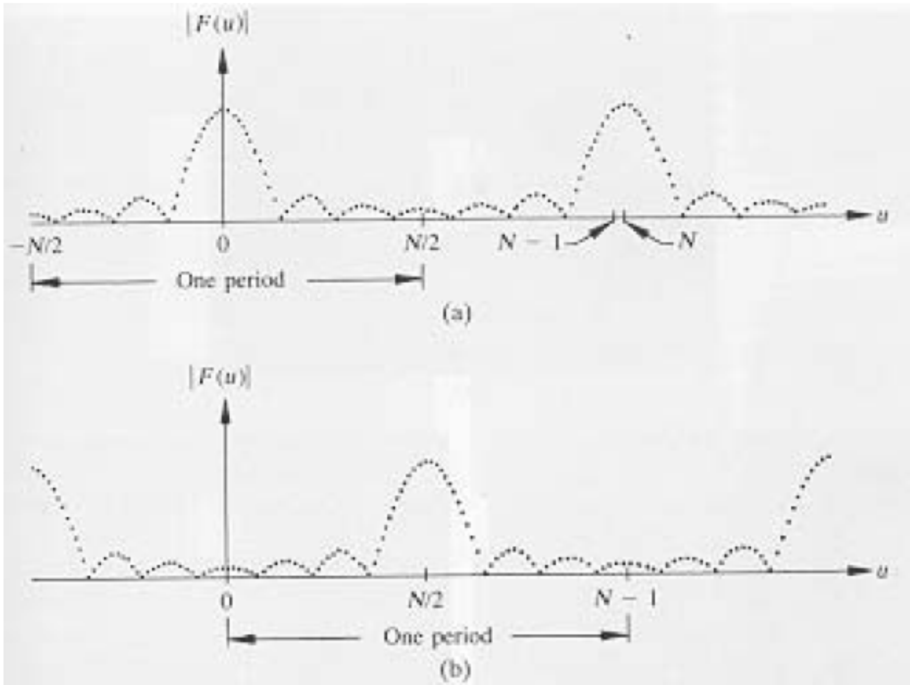
- If f(x,y) is real, then

$$F(u,v) = F^*(-u,-v) ==> |F(u,v)| = |F(-u,-v)|$$

DFT Properties: (2) Periodicity and Symmetry

- To display a full period, we need to translate the origin of the transform at **u=N/2** (or at **(N/2,N/2)** in 2D)
- In Matlab this is done with the command fftshift





DFT Properties: (3) Translation

f(x,y) $\longleftrightarrow$ **F(u,v)**

- Translation in the spatial domain

$$f(x - x_0, y - y_0) \longleftrightarrow F(u,v)e^{-j2\pi\left(\frac{ux_0+vy_0}{N}\right)} \qquad (1)$$

- Translation in the frequency domain

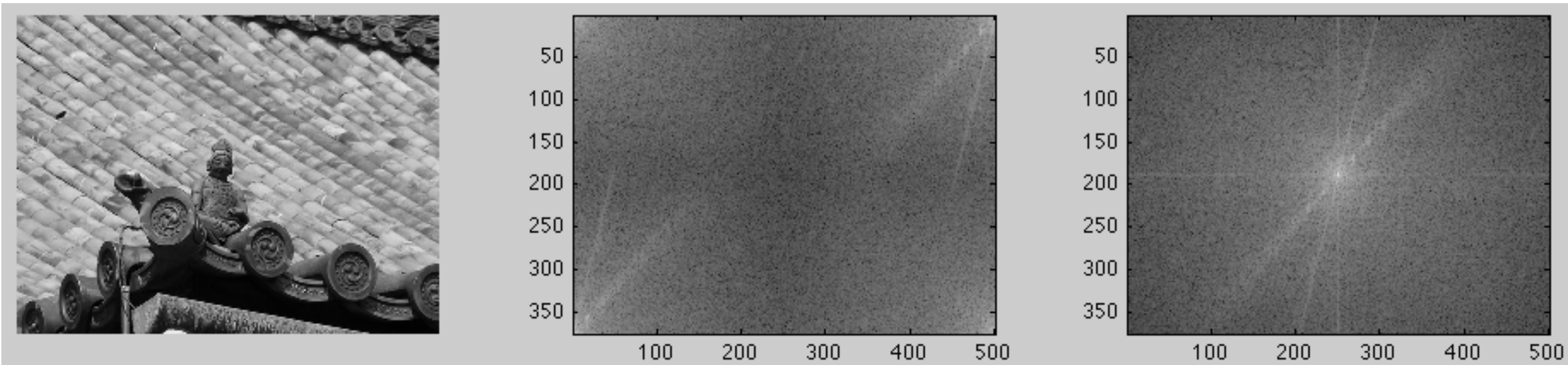
$$f(x,y)e^{j2\pi\left(\frac{u_0x+v_0y}{N}\right)} \longleftrightarrow F(u - u_0, v - v_0) \qquad (2)$$

DFT Properties: (3) Translation

- To move F(u,v) at (N/2, N/2), take $u_0 = v_0 = N/2$

from (2): $$e^{j2\pi\left(\frac{\frac{N}{2}x+\frac{N}{2}y}{N}\right)} = e^{j\pi(x+y)} = (-1)^{x+y}$$

$$f(x,y)(-1)^{x+y} \longleftrightarrow F(u-N/2, v-N/2)$$

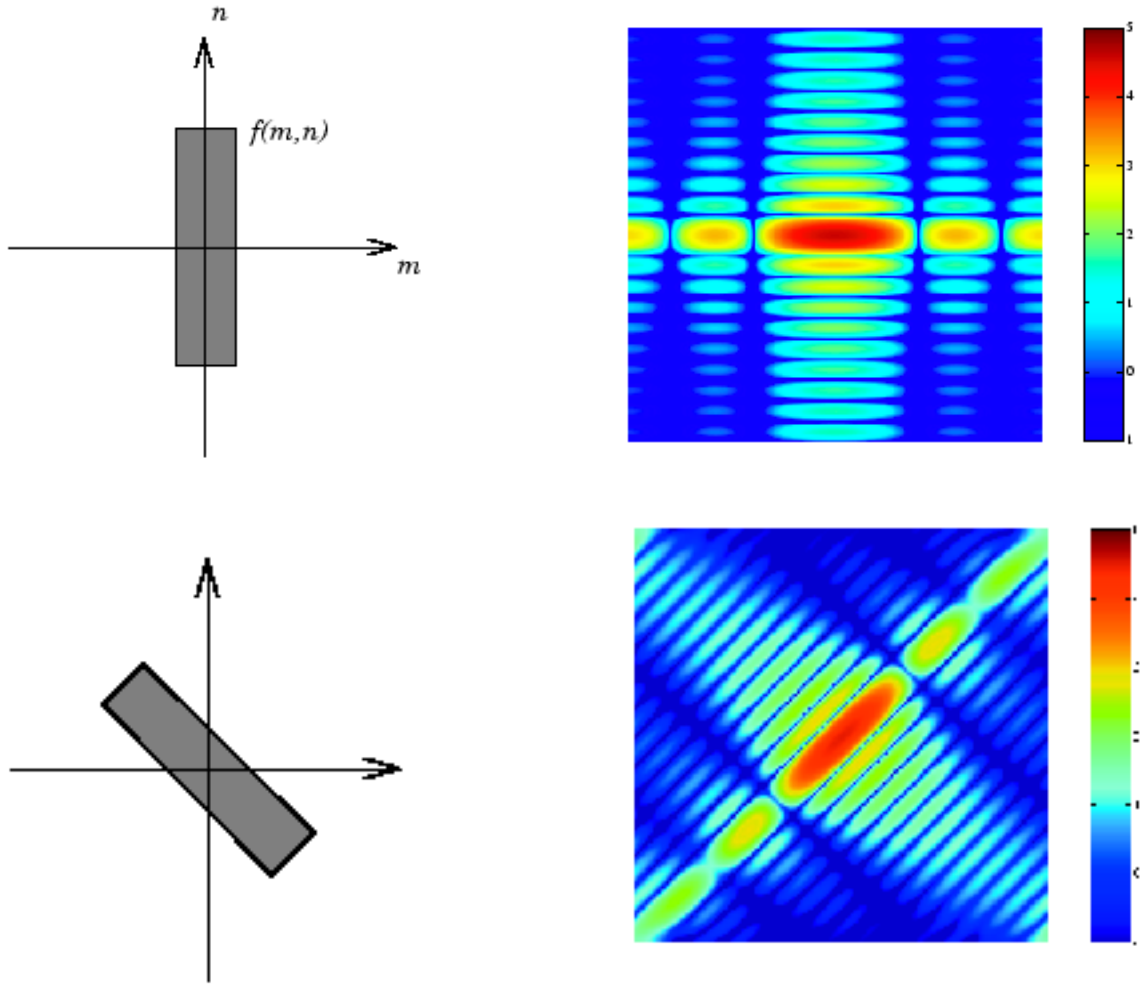


no translation

after translation

DFT Properties: (4) Rotation

- Rotating f(x,y) by θ rotates F(u,v) by θ



DFT Properties: (5) Distributive

- Fourier transforms are additive

$$F[f(x,y)+g(x,y)]=F[f(x,y)]+F[g(x,y)]$$

- In general, FTs are NOT multiplicative

$$F[f(x,y)g(x,y)]\neq F[f(x,y)]F[g(x,y)]$$

DFT Properties: (6) Scale

- FTs scale

$$af(x, y) \longleftrightarrow aF(u, v)$$

DFT Properties: (7) Average value

Average in the spatial domain:

$$\bar{f}(x,y) = \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)$$

From definition of F(u,v) it follows that F(u,v) at u=0, v=0:

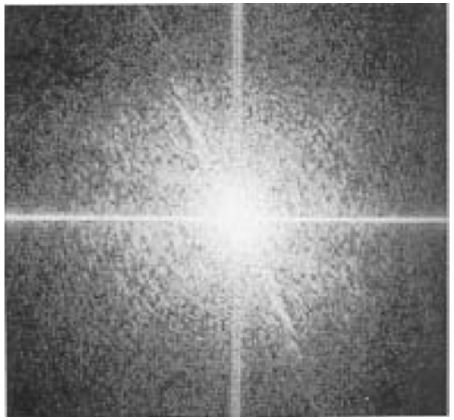
$$F(0,0) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)$$

So we have that:

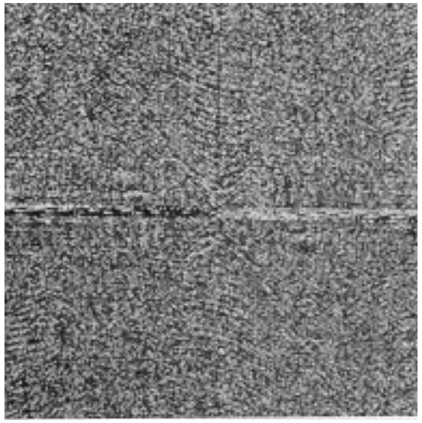
$$\bar{f}(x,y) = \frac{1}{N} F(0,0)$$

Magnitude and Phase of DFT

- What is more important?

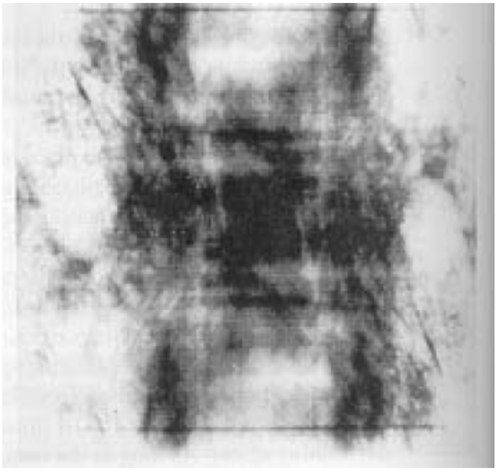


magnitude



phase

Magnitude and Phase of DFT

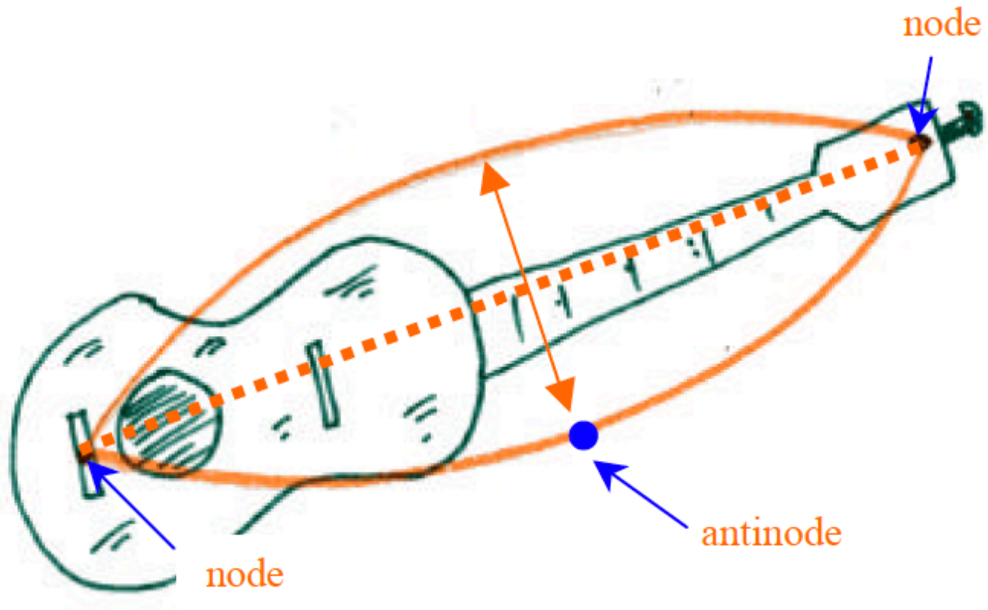


Reconstructed image using
 the magnitude information
→ magnitude determines the
contribution of each component

Reconstructed image using
the phase information
→ phase determines
where components are present

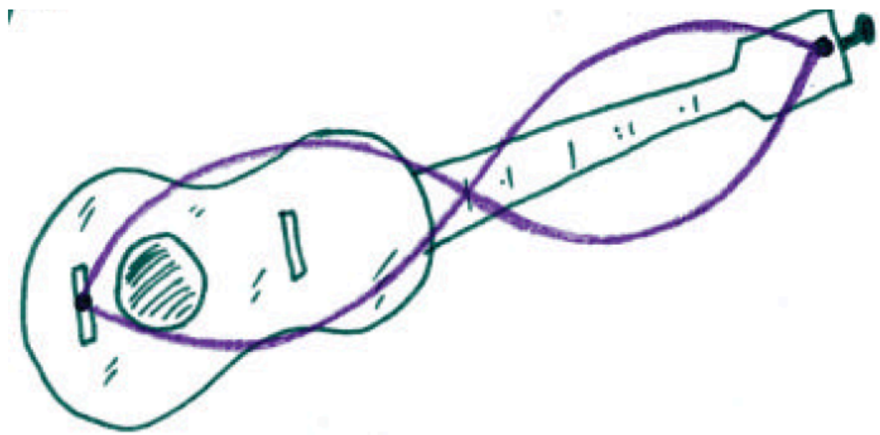
A foray into music

- When you pluck a string on a guitar, the string vibrates like this
- If the length of the guitar-string is L, then the wavelength of this vibration is?
- 2L with frequency f0



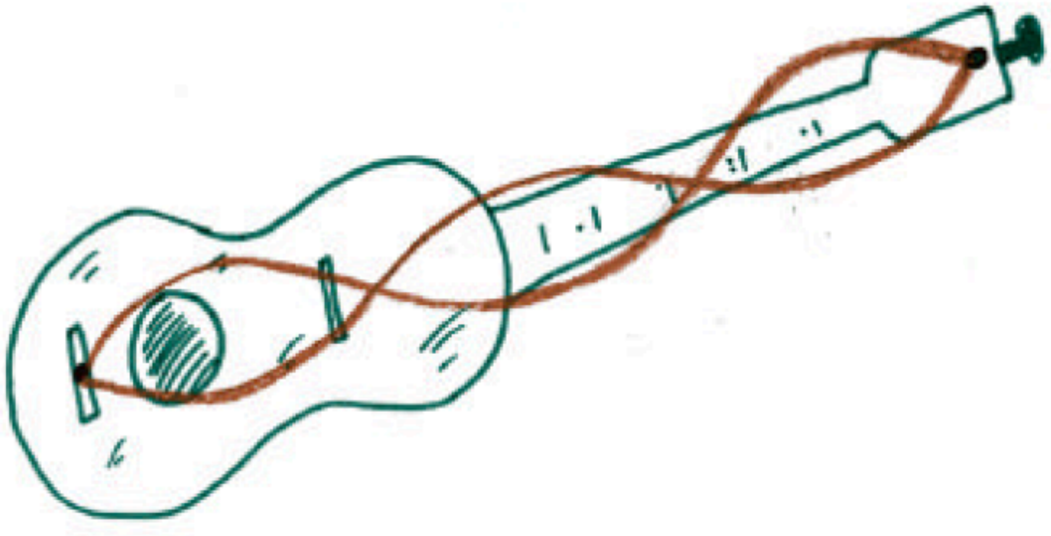
A foray into music

- But, not all parts of the string vibrate at the same frequency
- The next higher frequency that fits onto the fixed string is
- 2 * f0 with wavelength 1/2 * 2*L
- This is called the first overtone (harmonic)



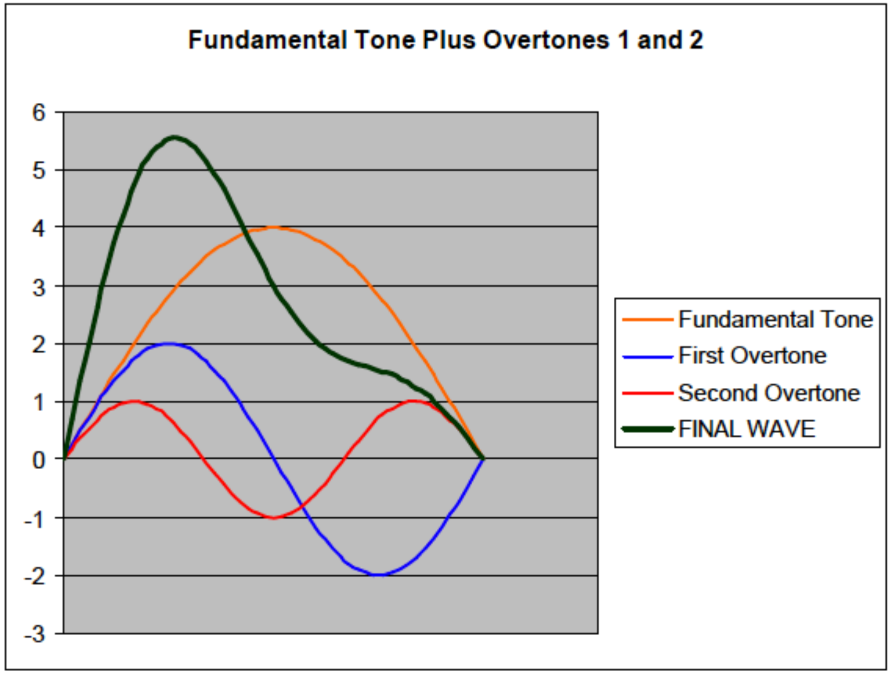
A foray into music

- The next overtone (harmonic) looks like this
- This has 3 * f0 with wavelength 1/3 * 2*L
- This is called the second overtone (harmonic)



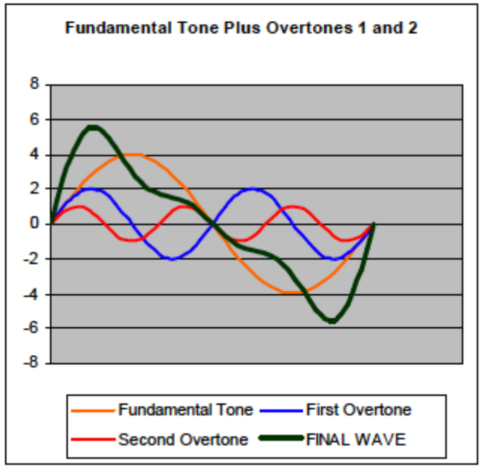
A foray into music

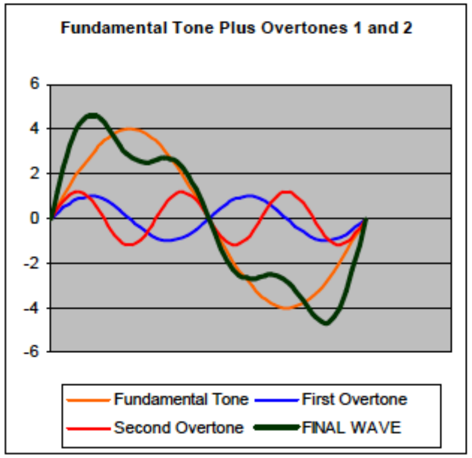
- How do we get music now?
- We add all the waves from all overtones

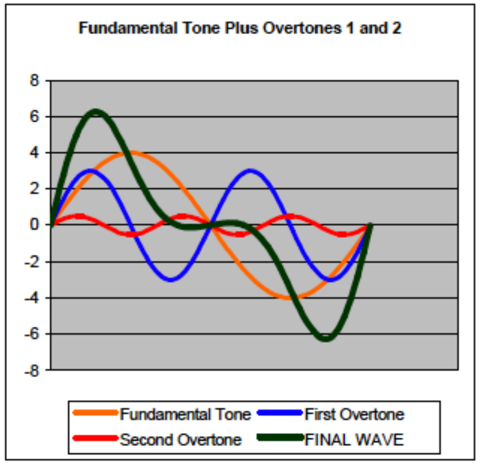


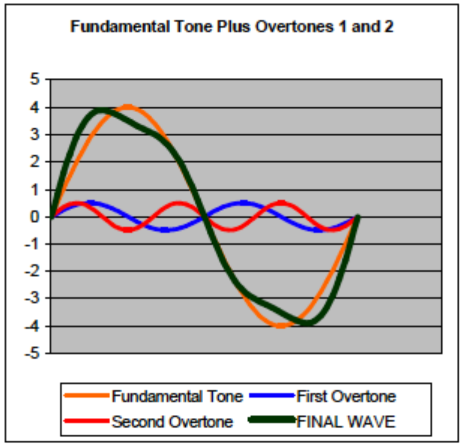
A foray into music

- And why do instruments sound all different?
- Because of the amount of how each overtone contributes to the whole



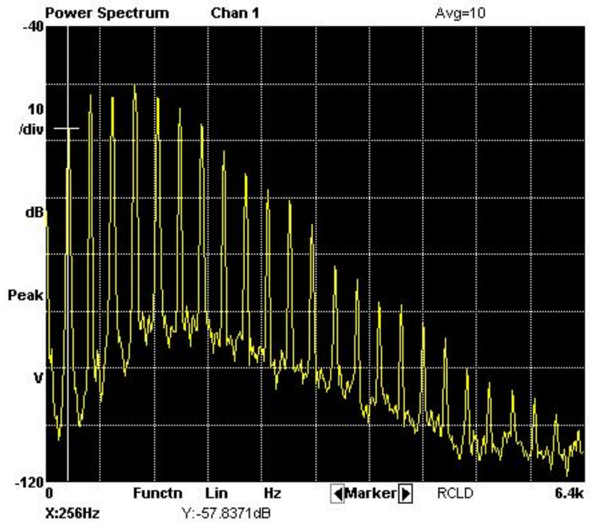




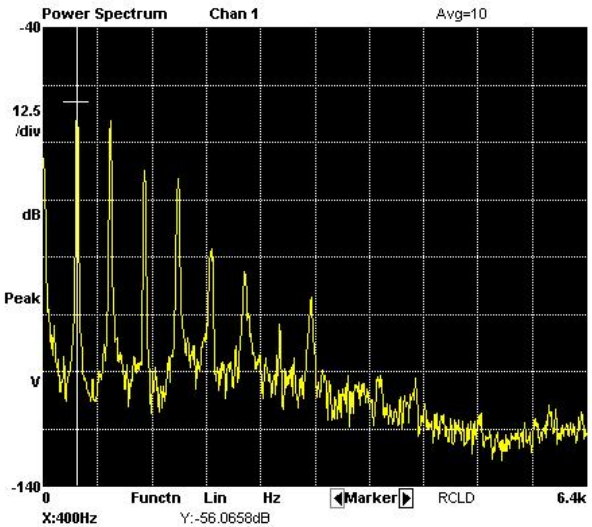


A foray into music

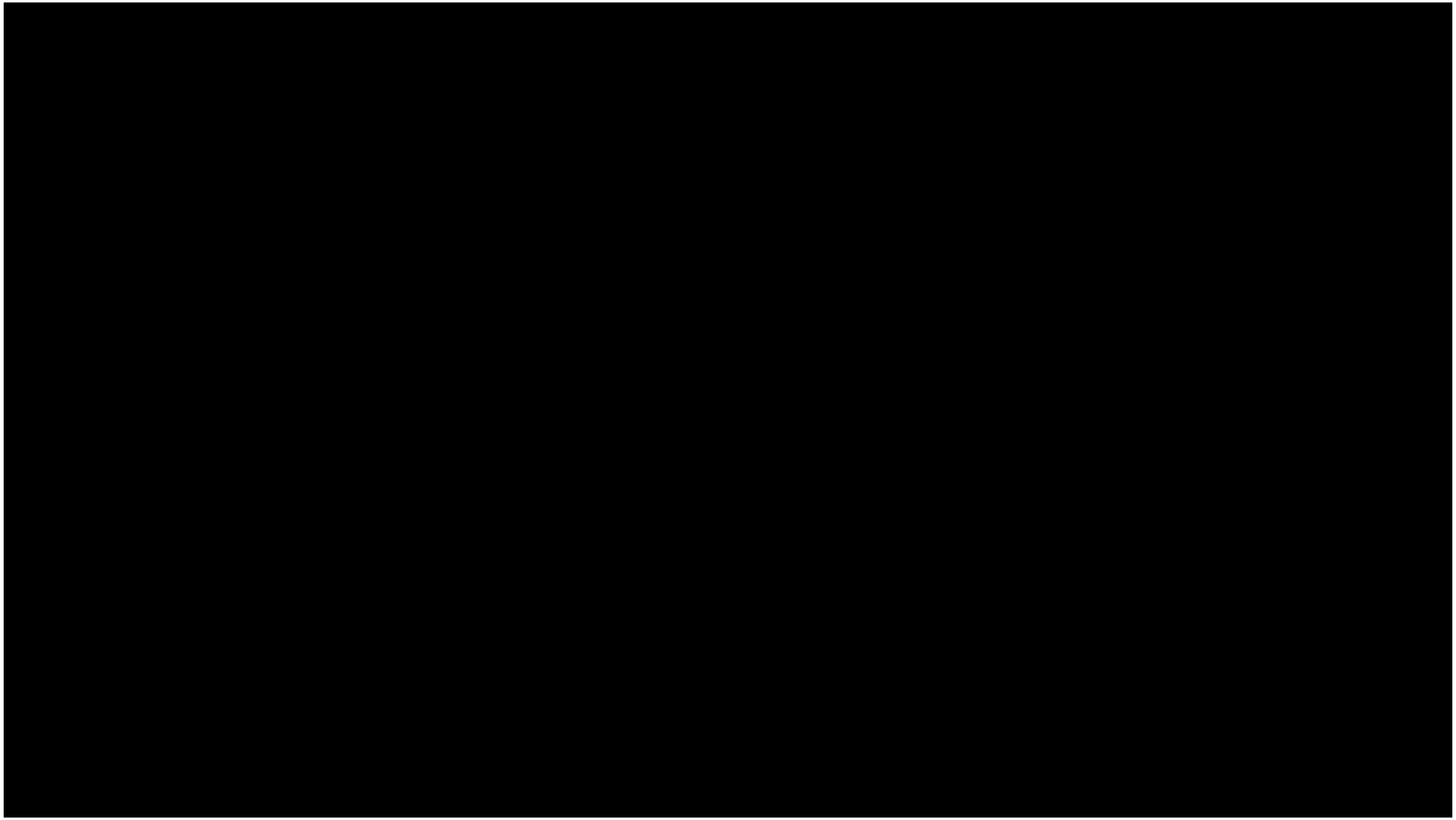
- Power spectrum of a trumpet



- Power spectrum of a clarinet

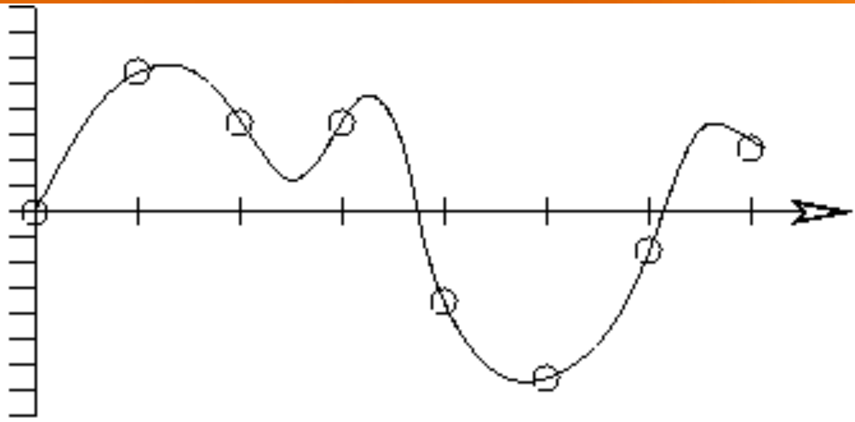


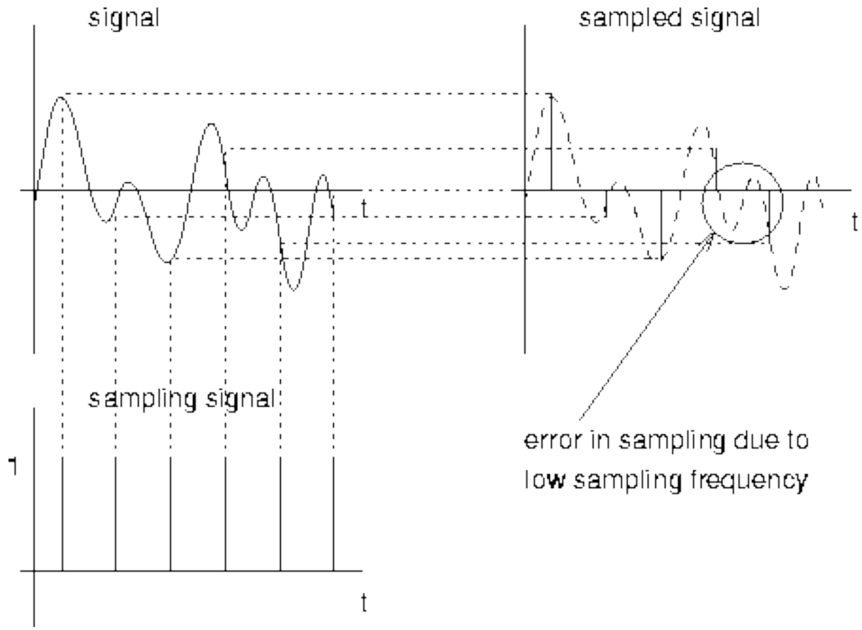
Sampling



Sampling: from analog to digital

- Real signals are continuous, but the computer can only handle discretized data.
  - We therefore have to convert analog to digital and vice versa (**ADC** and **DAC**)
- Sampling then means to measure the analog signal at different moments in time, recording its physical property
  - For EEG-recordings, for example, this is the voltage of the signal
  - This forms an approximation to the original signal



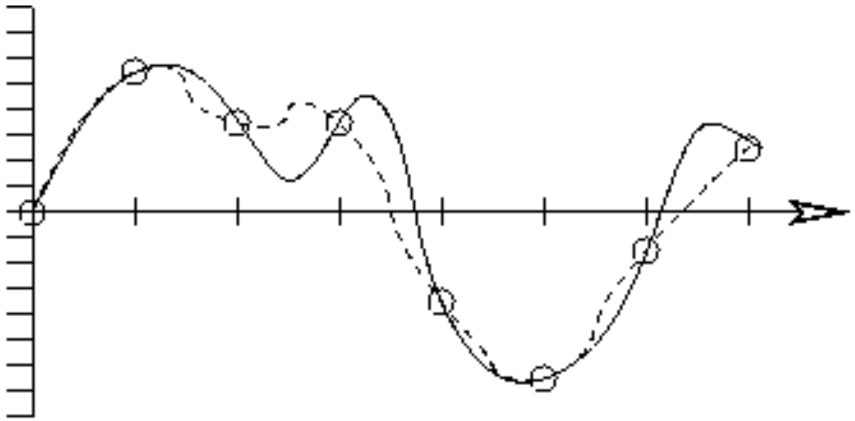


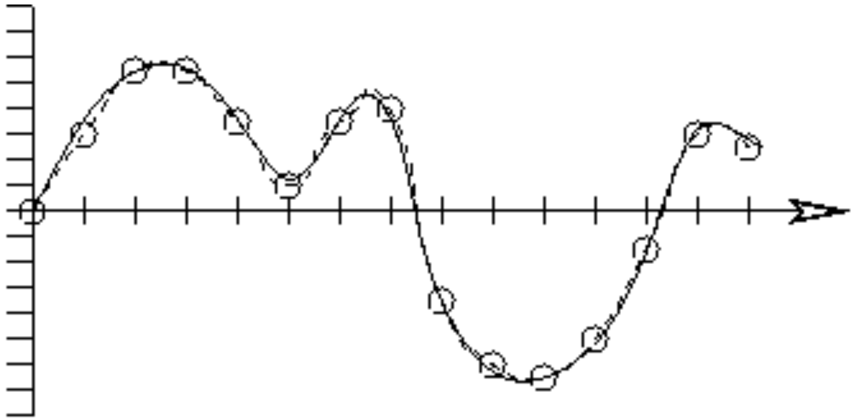
Courtesy of: http://puma.wellesley.edu/~cs110/lectures/M07-analog-and-digital/

Courtesy of: http://www.cs.ucl.ac.uk/staff/jon/mmbook/book/node96.html

Sampling: from digital to analog

- Going the opposite way means to reconstruct the analog signal from the digital signal
  - Interpolation problem!
  - Amounts to drawing a curve through the points
- What curve to choose?
- Multiple curves possible in (a)
  - First part reasonably ok but many errors in the latter part
- In (b), sampling has been doubled
  - Reconstructed curve much better
  - BUT: also needs more "bandwidth"!





Courtesy of: http://puma.wellesley.edu/~cs110/lectures/M07-analog-and-digital/

Nyquist Sampling Theorem

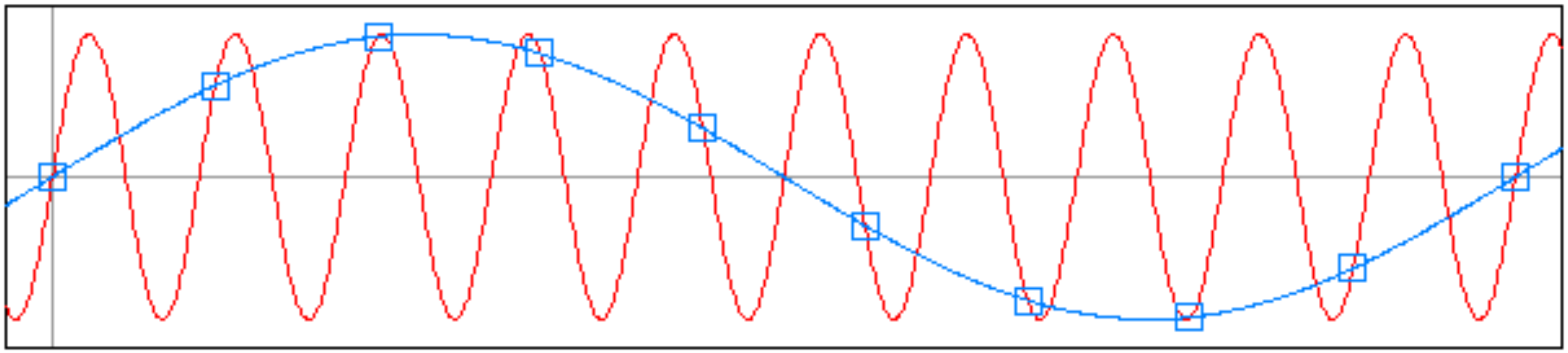
- This theorem answers the question of how often must we sample to faithfully represent the signal using discrete sampling?
  - First articulated by Harry Nyquist and later proved by Claude Shannon
- Solution: **Sample twice as often as the highest frequency that you need to represent!**
- $f_s >= 2 * f_H$ (Nyquist rate)
  - f_s is the sampling frequency and f_H is the highest frequency present in the signal
- For example, highest sound frequency that most people can hear is about 20 KHz (with some sharp ears able to hear up to 22 KHz), we can capture music by sampling at 44 KHz.
  - That's how fast music is sampled for CD-quality music

Courtesy of: http://puma.wellesley.edu/~cs110/lectures/M07-analog-and-digital/

Aliasing

CognitiveSystems

- If the sampling condition is **not** satisfied, then frequencies will overlap
- **Aliasing** is an effect that causes different continuous signals to become indistinguishable (or *aliases* of one another) when sampled.



Courtesy of http://en.wikipedia.org/wiki/Aliasing

Examples of aliasing

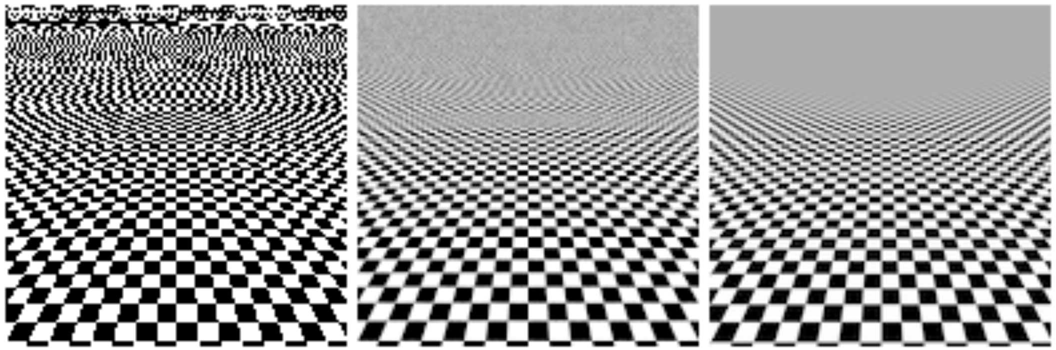
CognitiveSystems

- Sunrise – Temporal Aliasing
  - The sun moves east to west in the sky, with 24 hours between sunrises.
  - If one were to take a picture of the sky every 23 hours, the sun would appear to move west to east, with 24 × 23 = 552 hours between sunrises.
- Wagon Wheel effect – Temporal Aliasing
  - The same phenomenon causes spoked wheels to apparently turn at the wrong speed or in the wrong direction when filmed, or illuminated with a flashing light source.
- Moire pattern – Spatial Aliasing
  - Stripes captured on a digital camera would cause aliasing between the stripes and the camera sensor.
    - Distance between the stripes is smaller than what the sensor can capture
    - Solution to this would be to go closer or to use a higher resolution sensor

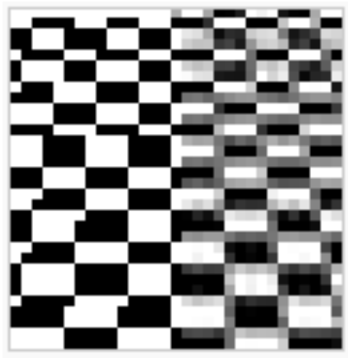
Courtesy of http://en.wikipedia.org/wiki/Aliasing

Aliasing

- To prevent aliasing, two things can be done
  - Increase the sampling rate
  - Introduce an anti-aliasing filter
- Anti-aliasing filter - restricts the bandwidth of the signal to satisfy the sampling condition.
  - This is not satisfiable in reality since a signal will have some energy outside of the bandwidth.
  - The energy can be small enough that the aliasing effects are negligible (not eliminated completely).
- Anti-aliasing filter: low pass filters, band pass filters, non-linear filters
- Always remember to apply an anti-aliasing filter prior to signal down-sampling

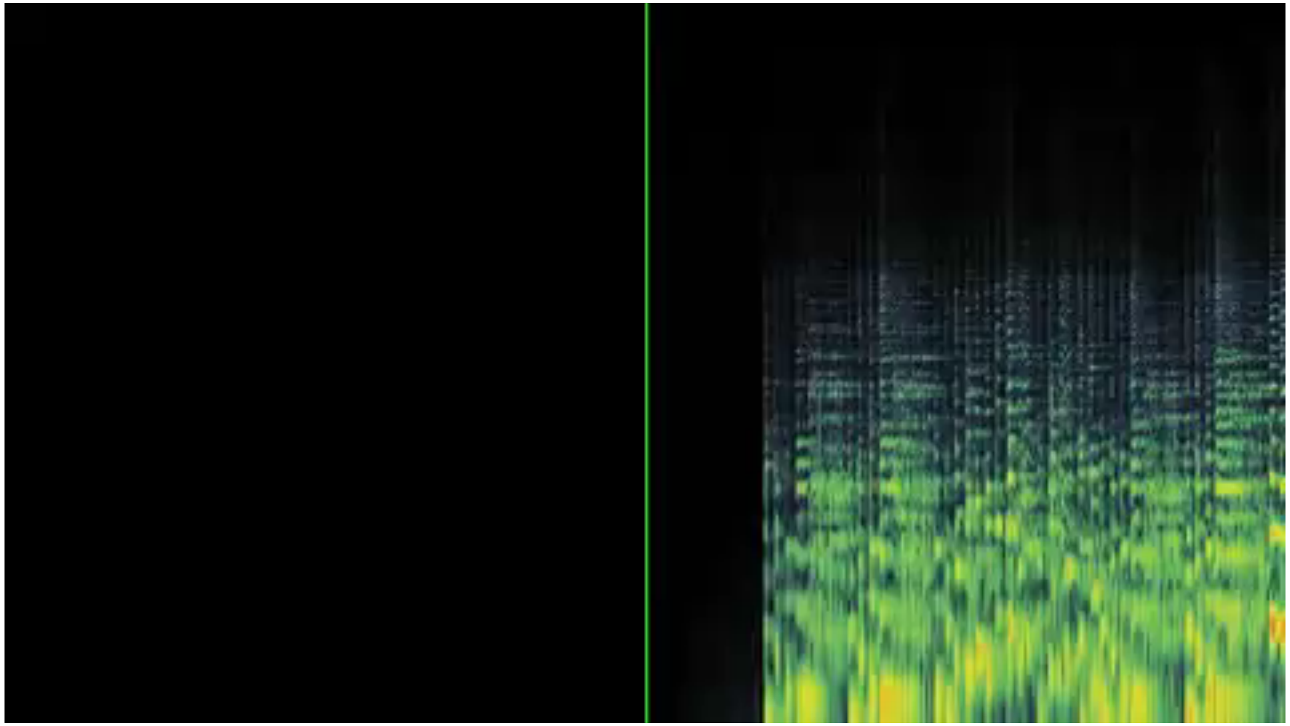


(a) (b) (c)



Spectrograms

- Spectrograms visualize the spectrum over time to show the temporal evolution of frequencies



EEG

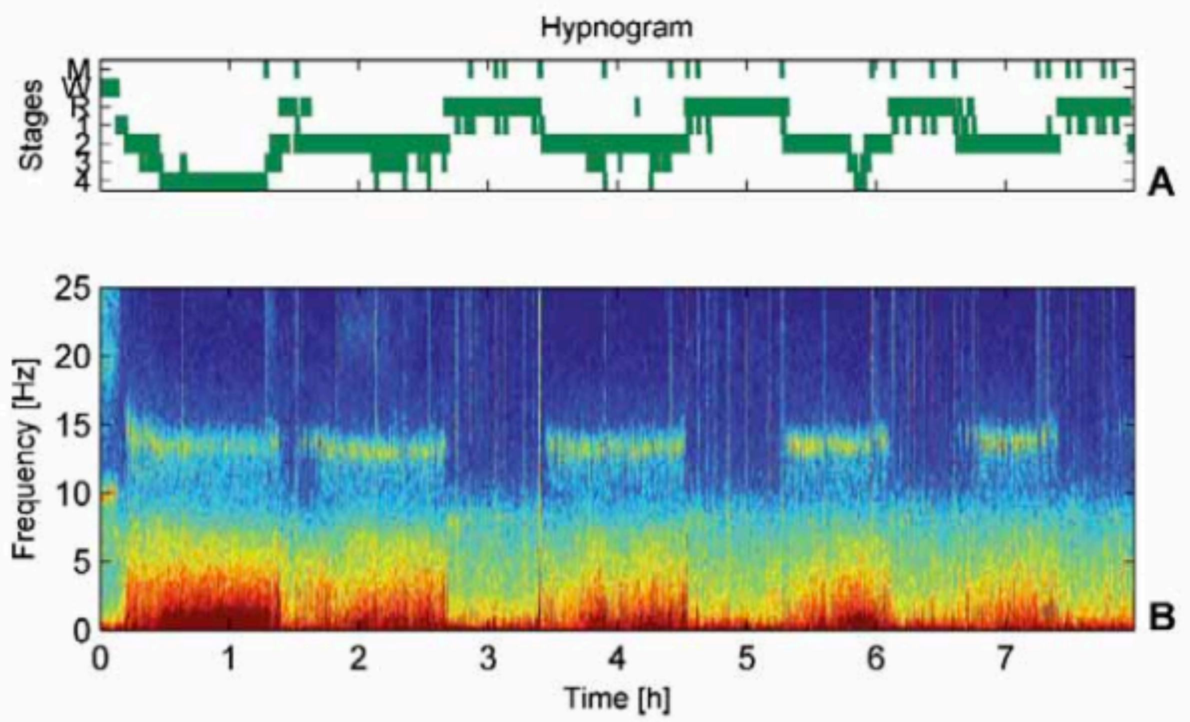


Figure 1: **A** Sleep profile (hypnogram) and **B** color-coded power spectra of consecutive 20-s epochs (average of 5 spectra calculated for 4-s epochs; Hanning window). Data were sampled with 256 Hz. Spectra (derivation C3A2) are color coded on a logarithmic scale (0 dB = 1 $\mu V^2/Hz$; -10 dB 20 dB). Sleep stages were visually scored for 20-s epochs (W: waking; M: movement time; R: REM sleep; 1 to 4: non-REM sleep stages 1 to 4).

EEG

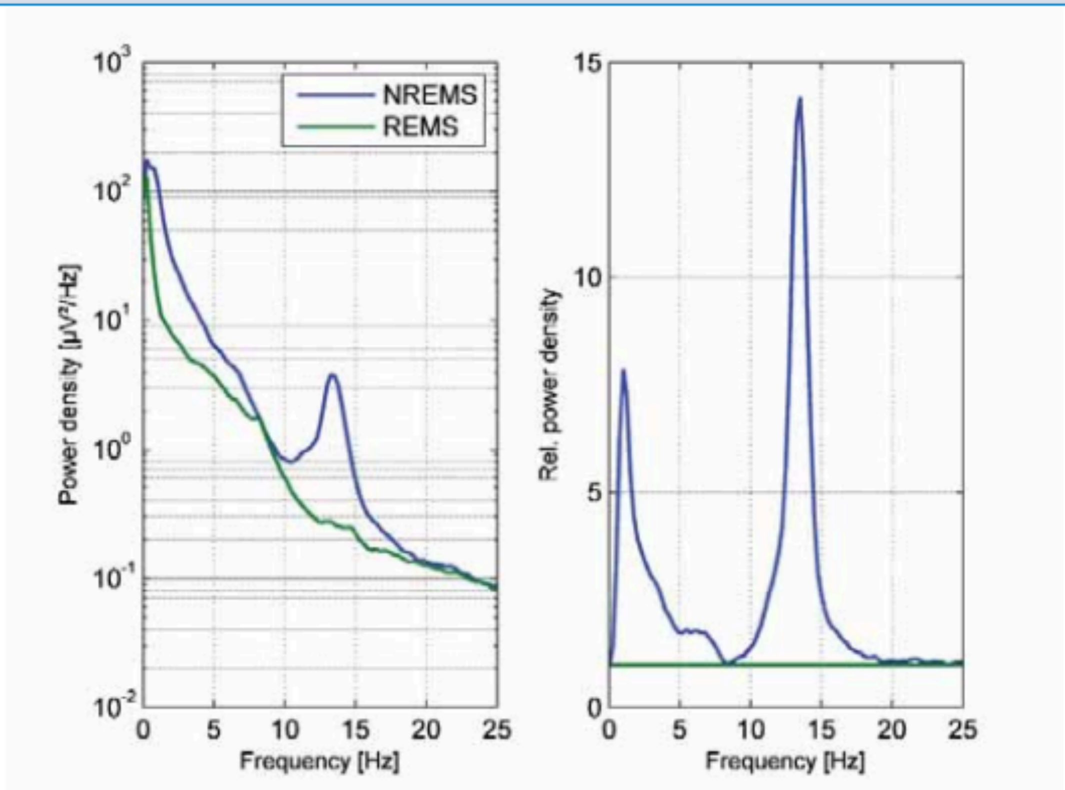
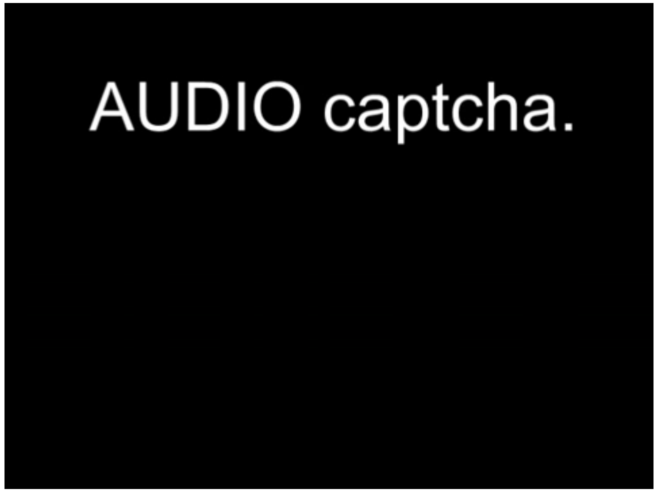
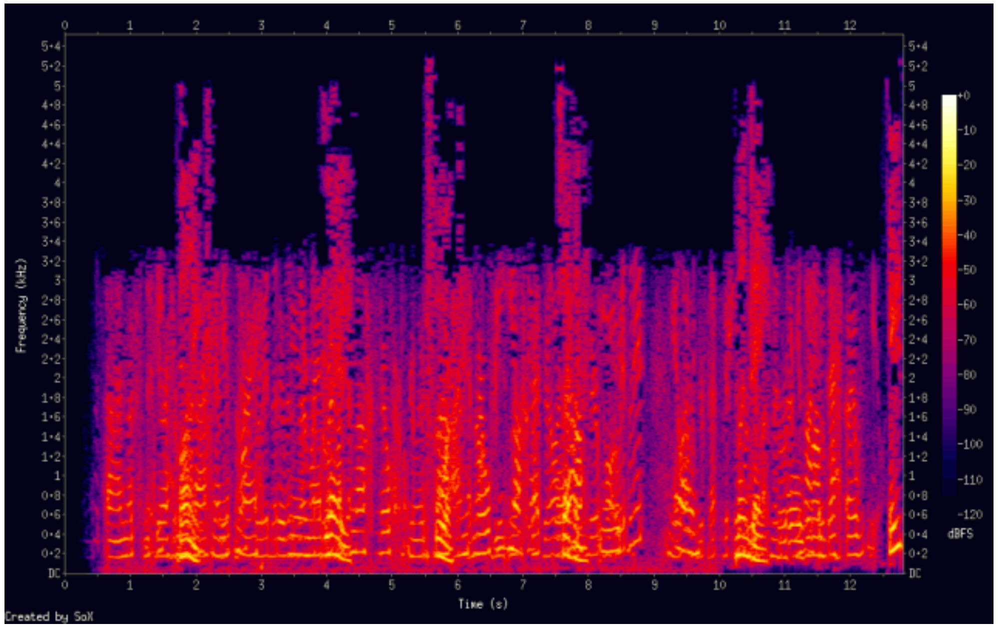


Figure 2: **Left:** Average spectra of non-REM sleep (stages 2, 3, and 4) and REM sleep. **Right:** Non-REM sleep spectrum plotted relative to the REM sleep spectrum. Only 20-s epochs without artifacts were included in the average spectra. Same recording as in Figure 1.

Spectrograms

- Google uses speech Captchas for disabled users to check for human versus web-bots inputs
- However, the system was vulnerable to attacks as this spectrogram shows:





Spectrograms

- Shortly before a team from CMU posted their findings, Google announced an upgrade to their system, shown below
- More at http://arstechnica.com/security/2012/05/google-recaptcha-brought-to-its-knees/