CS559 – Lecture 4
Sampling, Reconstruction, Resampling

These are course notes (not used as slides)

Written by Mike Gleicher, Sept. 2005

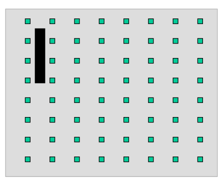
With slides adapted from notes from Stephen Chenney

Last time

- Representing images as samples
- Today:
  - Sampling Theory
  - Why bad sampling is bad
  - How to understand what is lost (sampling theory)
  - How to do as well as possible
  - Reconstruction from sampled representations
  - Resampling in practice

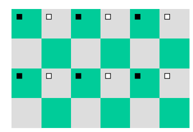
Bad sampling is bad

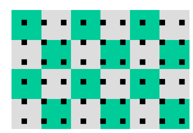
- Miss small things between samples



Get really weird results

- Sample a checkerboard
- Too few samples
  - Get all black
  - Get all white
  - Get weird patterns
    - Aliasing
    - Moire'
  - Arbitrary algorithm decision gives very different answers!
- Imagine resampling

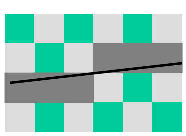


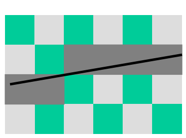


Demonstration ratios: 4/6 (here) = 2/3

Ugly

- Imagine line drawing
- Jaggies
- Crawlies
  - Small change causes jump
  - Smooth motion becomes jumpy



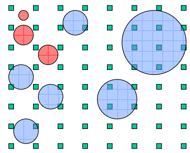


Dealing with discretization

- Sampling
  - Understand what information we are throwing away
- Reconstruction
  - Recreate as well as possible from the samples
- Re-Sampling
  - Transform the image
- Signal Processing / Image Processing
- Consider the 1D case first since its easier

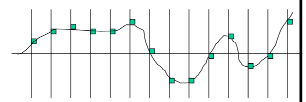
Intuition

- Too few samples = BAD
- Sampling rate depends on the thing you're sampling
- Need to sample close enough to get smallest object
- Need to limit small objects to be big enough that they aren't missed

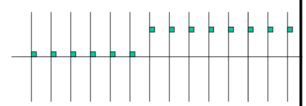


Point sampling in 1D

- Only record samples
- Don't know what happens in between samples

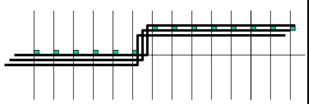


- Given the samples, don't know what really happened!

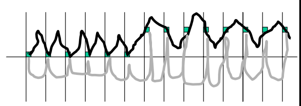


Reconstruction from Sampling

- Can't localize events
  - Bigger problems than that
- No idea! Signal could be anything



- Without additional information, we're guessing as to what the signal is



- But what additional info?

Sampling Intuitions

- Reconstruct the "smoothest" signal that makes sense from samples
- If signal is "smooth enough", sampling will give something we can reconstruct
- If signal is not "smooth", sampling will give something that will reconstruct to something else
  - Aliasing
- But how do we define "smooth"

Signal processing

- Need better "language" for talking about signals
- Idea: represent signals in a different way
- Up till now: time domain (graph against time)
  - Good for asking "what does signal do at time X"
- New idea: frequency domain
  - Good for talking about how smooth signals are
- Different view of the same thing

Frequency Domain

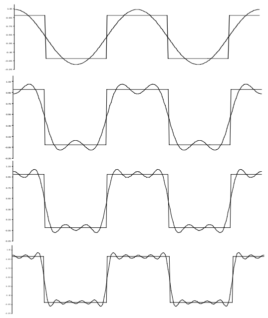
- Fourier Theorem:
  - Any periodic signal can be represented as a sum of sine and cosine waves with harmonic frequencies
  - If one function has frequency *f*, then its harmonics are function with frequency *nf* for integer *n*
  - Extensions to non-periodic signals later
  - Also works in any dimension (e.g. 2 for images, 3, …)
- Example: box

Example: Box (Square Wave)

- 1 cosine – bad
- More cosines, better approx

$$f(x) = \begin{cases} 1 & |x| \le \frac{1}{2} \\ 0 & |x| > \frac{1}{2} \end{cases}$$

$$S_{boxes}(x) = \frac{1}{2} + \frac{2}{\pi}\sum_{k=1}^{\infty}(-1)^{k-1}\frac{\cos(2k-1)\omega x}{2k-1}$$
$$= \frac{1}{2} + \frac{2}{\pi}\left(\cos\omega x - \frac{1}{3}\cos 3\omega x + \frac{1}{5}\cos 5\omega x - \cdots\right)$$



Intuitions

- Low frequencies are smooth
  - High frequencies change fast, are not smooth
- If a signal can be made of only low frequencies, it is smooth
- If a signal has sharp changes, it will require high frequencies to represent

General Functions

- A non-periodic function can be represented as a sum of sin's and cos's of (possibly) all frequencies:

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

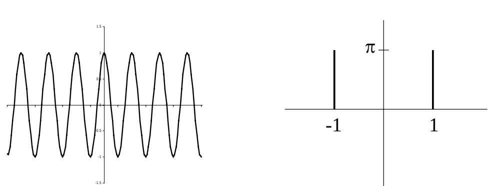
- *F(ω)* is the *spectrum* of the function *f(x)*
  - The spectrum is how much of each frequency is present in the function
  - We're talking about functions, not colors, but the idea is the same

Fourier Transform

- *F(ω)* is the Fourier Transform of f(t)
  - A different representation of the same signal
- To get f(t) back you use the Inverse Fourier Transform
- You don't need to know how to compute them

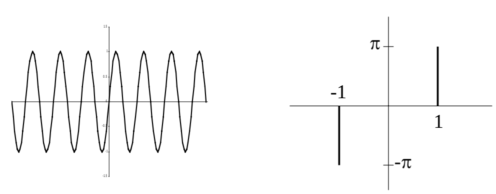
$$F(\omega) = \int_{-\infty}^{\infty} f(x)e^{-i\omega x}dx$$

Cosine and Its Transform



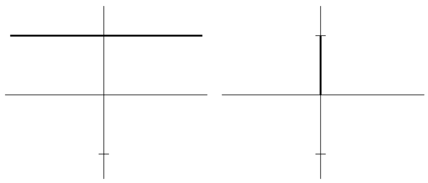
If *f(x)* is even, so is *F(ω)*

Sine and Its Transform



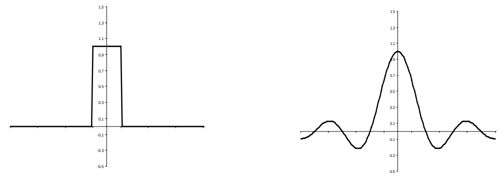
If *f(x)* is odd, so is *F(ω)*

Constant Function and Its Transform

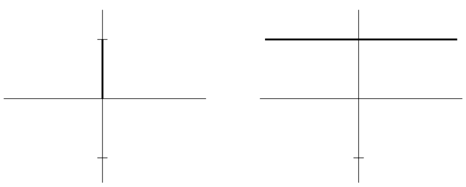


The constant function only contains the 0^{th} frequency – it has no wiggles

Box Function and Its Transform

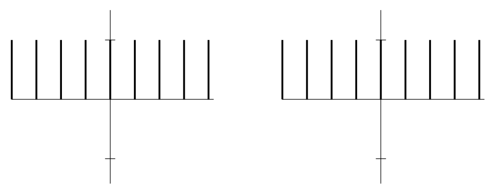


Delta Function and Its Transform

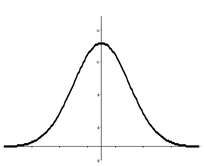


Fourier transform and inverse Fourier transform are qualitatively the same, so **knowing one direction gives you the other**

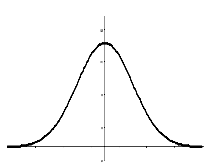
Shah (Spikes) and Its Transform



Gaussian and Its Transform



$$\frac{1}{2\pi} e^{-\frac{x^2}{2}}$$



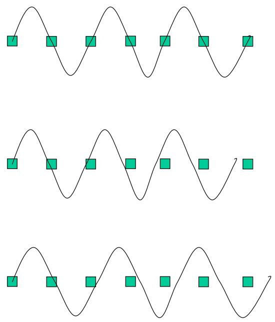
They are the same

Qualitative Properties

- The spectrum of a functions tells us the relative amounts of high and low frequencies
  - **Sharp edges give high frequencies**
  - **Smooth variations give low frequencies**
- A function is *bandlimited* if its spectrum has no frequencies above a maximum limit
  - sin, cos are band limited
  - Box, Gaussian, etc are not
- To band-limit a signal we *low-pass filter* it

Sampling Theorem (intuition)

- High frequencies get lost
  - Can only sample band limited signals
- Sampling rate must be 2 times higher than signal
- Signal must be half frequency of sample rate
  - Otherwise, signal can “turn around” between samples
- Nyquist rate
  - 2x highest frequency in signal



Sampling Theorem

- If your signal is bandlimited
- And you know what the band limit is
- And you sample at (at least) twice that frequency
  - Above the Nyquist rate
- Then – you can reconstruct your signal EXACTLY!

- Caveat
  - Ideal reconstruction requires perfect band limiting in both sampling and reconstruction