Lecture 5 – Sampling Theory in Practice

Michael Gleicher

Sept 2007

Notes not for Display

Overview

Last Time

- Why we care
- Aliasing
- Sampling Theory Intuitions
- Fourier Transforms
- Filtering
- Nyquist Theorem

This Time

- Using Sampling Theory
- Ideal Reconstruction
- Real filtering
- Convolutions
- Implementing
  - Reconstruction
  - Re-Sampling

Sampling Theory

- Given a set of samples (at a sampling rate):
  - There is exactly one band-passed signal that goes through those samples
  - Where the band-pass is less than half the sampling rate
- Ideal reconstruction
  - View samples as spike chain, low-pass filter
  - Need an ideal low-pass filter
  - Approximate ideal low-pass filter

Sampling Theory (2)

- If we sample a band-passed signal
  AND the sampling rate is > 2*highest freq
  THEN we can do ideal reconstruction
- If you know the highest frequencies you care about, you know how fast you need to sample!
  - CD Audio Example: human hearing isn't so great after 22Khz, so sample at 44.1Khz

Sampling Theory (3)

- If your signal is not bandpassed
  (i.e. has HF >= 2*sampling rate)
  THEN you will get aliasing when you sample
- Once you've aliased – you can't go back!
- You have no idea what the original was!
- Need to PREFILTER the signal before sampling to make it bandpassed

What's a filter?

- Generic – an operation that maps a signal to another signal
- Specifically: a LOW-PASS filter
  - Attenuates high frequencies
  - Easy to describe in frequency domain (give frequency response)
  - Multiply certain values

Convolution

- Multiplication in frequency is convolution in time (space)
- Convolution is the generalization of averaging
- Continuous convolution
  Discrete convolution

Convolution

- Operator on 2 signals
  - f(t) * g(t)   (f and g are both signals)
- Specifically
  - One signal is "our signal"
  - The other is the filter (called a kernel)

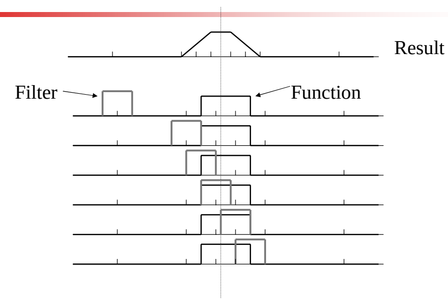
Filtering in the Spatial Domain

- Filtering the spatial domain is achieved by *convolution*

$$h(x) = f \otimes g = \int_{-\infty}^{\infty} f(u) g(x-u) du$$

- Qualitatively: Slide the filter to each position, *x*, then sum up the function multiplied by the filter at that position

Convolution Example



Discrete Convolution

- h(t) = (f*g)(t) = SUM f(i) g(t-i)
  - Notice that we flip g backwards as we slide it
  - Often g is symmetric, so this is easy to forget
- g = [1 2] f = [1 3 1 2 0] (outside range is 0)
- Zero centering of g ([1/3 1/3 1/3])
  - Weighted average

Dealing with boundaries

- Pretend data outside boundaries is 0
  - Dims edges
- Reflect about ends
- Keep constant values at edges
- Renormalize kernel

Convolution in 2D

- Show box moving around
- Seperable filters
  - Can do as 1D convolution in both directions
  - Not all filters can do this
  - Useful to find ones that can

Reconstruction in Practice

- Sample a sample – no problem!
- Issue is samples between samples
- Theory: LPF a spike chain
  - Convolve “resonstruction kernel” with samples
  - Only really need to evaluate at places where you’ll sample
- Another view: interpolation
  - Different interpolations are different filters

Some reconstruction kernels
Crude approximations to LPF

Constant

Triangle (Bartlet)

Interpolating Cubic (Catmull-Rom)

Spacing (1 unit = sample distance)

Scaling issues

Interpolating (non-interpolating kernels exist as well)

Approx to Ideal LPF

Reconstruction Example

- Sample at sample
- Sample between samples
- Bartlett filter
  - Width correct for sample spacing
- See how we get linear interpolation
- Could do this as linear interpolation
  - Generalizes nicely this way
- Need to evaluate filter for various values
- Convolve reconstruction kernel with sampling kernel (LPF for frequency limit)
- Easier ways to implement nearest neighbor