Sampling and Pyramids

15-463: Rendering and Image Processing
Alexei Efros

…with lots of slides from Steve Seitz

Today
Sampling
Nyquist Rate
Antialiasing
Gaussian and Laplacian Pyramids
Fourier transform pairs

Spatial domain

\[ f(x) \]

Sampling pattern

\[ g(x; \sigma) \]

Sampled signal

\[ f(x) \]

Frequency domain

\[ F(s) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi sf} \, dx \]

\[ \text{sinc}(s) \]

\[ g(x; 1/\sigma) \]

\[ \text{s} \]

\[ 1/w \]

\[ 1/w \]

Sampling

\[ F(s) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi sf} \, dx \]

\[ f(x) \]

\[ f/\text{s} \]

Spatial domain

Frequency domain
Reconstruction

$f(x) = \int_{-\infty}^{\infty} f(x)e^{-2\pi i s x} \, dx$

Spatial domain

Frequency domain

What happens when the sampling rate is too low?
Nyquist Rate

What’s the minimum Sampling Rate $1/w$ to get rid of overlaps?

\[ f(x) \quad \Rightarrow \quad \text{sinc function} \quad \Rightarrow \quad F(s) = \frac{\sin(\pi s)}{\pi s} \]

Spatial domain \quad \rightarrow \quad Frequency domain

\[ F(s) = \frac{\sin(\pi s)}{\pi s} \]

Sampling Rate $\geq 2 \times$ max frequency in the image

- this is known as the Nyquist Rate

Antialiasing

What can be done?

- Sampling rate $\geq 2 \times$ max frequency in the image

1. Raise sampling rate by oversampling
   - Sample at $k$ times the resolution
   - Continuous signal: easy
   - Discrete signal: need to interpolate

2. Lower the max frequency by prefiltering
   - Smooth the signal enough
   - Works on discrete signals

3. Improve sampling quality with better sampling
   - Nyquist is best case!
   - Stratified sampling (jittering)
   - Importance sampling (salaries in Seattle)
   - Relies on domain knowledge
Good sampling:
• Sample often or,
• Sample wisely

Bad sampling:
• See aliasing in action!

Gaussian pre-filtering

Solution: filter the image, then subsample
• Filter size should double for each ½ size reduction. Why?
Subsampling with Gaussian pre-filtering

Solution: filter the image, \textit{then} subsample
- Filter size should double for each \(\frac{1}{2}\) size reduction. Why?
- How can we speed this up?

Compare with...

Why does this look so crufty?
Image resampling (interpolation)

So far, we considered only power-of-two subsampling

- What about arbitrary scale reduction?
- How can we increase the size of the image?

Recall how a digital image is formed

\[ F[x, y] = \text{quantize}\{f(xd, yd)\} \]

- It is a discrete point-sampling of a continuous function
- If we could somehow reconstruct the original function, any new image could be generated, at any resolution and scale
Image resampling

So what to do if we don’t know $f$

- Answer: guess an approximation $\tilde{f}$
- Can be done in a principled way: filtering

$$F[x]$$

$d = 1$ in this example

Image reconstruction

- Convert $F$ to a continuous function
  $$f_\ell(x) = F(\frac{x}{\ell})$$ when $\frac{x}{\ell}$ is an integer, 0 otherwise
- Reconstruct by cross-correlation:
  $$\tilde{f} = h \otimes f_F$$

Resampling filters

What does the 2D version of this hat function look like?

- $h(x)$ performs linear interpolation
- (tent function) performs bilinear interpolation

Better filters give better resampled images

- Bicubic is common choice

Why not use a Gaussian?

What if we don’t want whole $f$, but just one sample?
Bilinear interpolation

Sampling at $f(x,y)$:

$$
\begin{align*}
  f(x, y) &= (1 - a)(1 - b) f[i, j] \\
          &+ a(1 - b) f[i + 1, j] \\
          &+ ab f[i + 1, j + 1] \\
          &+ (1 - a)b f[i, j + 1]
\end{align*}
$$

Image Pyramids

Idea: Represent $N \times N$ image as a "pyramid" of $1 \times 1$, $2 \times 2$, $4 \times 4$, ..., $2^n \times 2^n$ images (assuming $N = 2^n$)

Known as a Gaussian Pyramid [Burt and Adelson, 1983]
- In computer graphics, a mip map [Williams, 1983]
- A precursor to wavelet transform
A bar in the big images is a hair on the zebra’s nose; in smaller images, a stripe; in the smallest, the animal’s nose.

Figure from David Forsyth

Gaussian pyramid construction

- Filter
- Subsample

Repeat

Until minimum resolution reached

- can specify desired number of levels (e.g., 3-level pyramid)

The whole pyramid is only 4/3 the size of the original image!
Laplacian Pyramid

What are they good for?

Improve Search
- Search over translations
  - Like homework
  - Classic coarse-to-fine strategy
- Search over scale
  - Template matching
  - E.g. find a face at different scales

Precomputation
- Need to access image at different blur levels
- Useful for texture mapping at different resolutions (called mip-mapping)

Image Processing
- Editing frequency bands separately
- E.g. image blending... next time!