Sampling and Aliasing

- Sects. 4.17-18 in H&B
- Display is discrete and world is continuous
  (at least at the level we perceive)

- *Sampling*: Convert continuous to discrete
- *Reconstruction*: Converting from discrete to continuous
- *Aliasing*: Artifacts arising from sampling and consequent loss of information
- *Anti-aliasing*: Attempts to overcome aliasing

Sampling

Occurs when the sampling inherent in rendering does not contain enough information for an accurate image.

Original Scene  Sampling Pixel Centres  Rendered Image

Scanline Luminosity  Sampled Signal  Luminosity Signal

Images from SIGGRAPH 93 Educators' Slide Set
**Some Aliasing Artifacts**

- *Spatial:* Jaggies, Moire
- *Temporal:* Strobe lights, “Wrong” wheel rotations
- *Spatio-Temporal:* Small objects appearing and disappearing

**Effects of Aliasing**

Common aliasing errors (called artefacts) are: jagged profiles, disappearing or improper fine detail, and disintegrating textures. 

Images from SIGGRAPH 93 Educators' Slide Set
Jaggies

Moire Patterns
Moire Patterns

Image Courtesy:
Chen, Dachille, Kaufman

Moire Patterns

Image from http://www.daube.ch/docu/glossary/moiree.html
Disintegrating Texture

- The checkers on a plane should become smaller with distance.
- But aliasing causes them to become larger and/or irregular.
- Increasing resolution only moves the artefact closer to the horizon.

Temporal Aliasing

The wheel appears to be moving backwards at about ¼ angular frequency:
http://lite.bu.edu/lite1/perception/anamorphic/wade.html
Image as a 2D Signal

- Analytic geometry gives a coordinate system for describing geometric objects.
- Fourier series gives a coordinate system for functions.

Fourier Series
Basis

- \( P=(x,y) \) means \( P = x(1,0)+y(0,1) \)
- Similarly:

\[
f(\theta) = a_{00} + a_{11} \cos(\theta) + a_{12} \sin(\theta) + a_{21} \cos(2\theta) + a_{22} \sin(2\theta) + \ldots
\]

Note, I’m showing non-standard basis, these are from basis using complex functions.

Fourier Analysis

Represent a function as a weighted sum of sines

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

\[ \forall c, \exists a_1, a_2 \text{ such that:} \]
\[ \cos(\theta + c) = a_1 \cos \theta + a_2 \sin \theta \]

Orthogonal Basis

- \[ \|(1,0)\| = \|(0,1)\| = 1 \]
- \[ (1,0).(0,1) = 0 \]
- Similarly we use normal basis elements eg:
  \[ \frac{\cos(\theta)}{\|\cos(\theta)\|} \]
  \[ = \sqrt{\int_{0}^{2\pi} \cos^2 \theta \, d\theta} \]
- While, eg:
  \[ \int_{0}^{2\pi} \cos \theta \sin \theta \, d\theta = 0 \]
  \[ \int_{0}^{2\pi} \cos \theta \cos 2\theta \, d\theta = 0 \]
Coordinates with Inner Products

\[ x = (x, y) \cdot (1, 0) \quad y = (x, y) \cdot (0, 1) \]

\((x, y) = x(1, 0) + y(0, 1)\)

\[ a_{i,1} = \int_0^{2\pi} f \ast \frac{\cos i\theta}{\|\cos i\theta\|} d\theta \quad a_{i,2} = \int_0^{2\pi} f \ast \frac{\sin i\theta}{\|\sin i\theta\|} d\theta \]

\[ f = \sum a_{i,1} \frac{\cos i\theta}{\|\cos i\theta\|} + \sum a_{i,2} \frac{\sin i\theta}{\|\sin i\theta\|} + a_{0,0} \]

2D Example
The Fourier basis allows us to see which part of the image is causing trouble. The high frequency harmonics.

We’ll get rid of these.

• This is a very general concept
• But in this class, we’ll just talk about convolutions that perform smoothing.
• Averaging: replace each pixel by the average in its neighborhood.
  – This can be done for continuous images with an integral
• Can take a weighted average.
  – The most common uses a normal (Gaussian) function for weights.
Weighted Average

• We define a set of weights, \( w_{-k}, w_{-k+1}, \ldots, w_{-1}, w_0, w_1, \ldots, w_k \). Then, smooth \( g \) by:
  \[
  \sum_{i=-k}^{k} w_i g(x + i)
  \]
• This set of weights is a filter.
• For simple average, all \( w = 1/(2k+1) \)
• Other common alternative is to pick \( w \) from Normal (Gaussian) distribution.

Convolution Theorem

\[
\mathbf{f} \ast \mathbf{g} = \mathcal{T}^{-1} \mathbf{F} \ast \mathbf{G}
\]

• \( \mathbf{f} \) is a description of the weights in the weighted average (the filter), \( \mathbf{g} \) is the image.
• \( \mathbf{F}, \mathbf{G} \) are transform of \( \mathbf{f}, \mathbf{g} \)

That is, \( \mathbf{F} \) contains coefficients, when we write \( \mathbf{f} \) as linear combinations of harmonic basis.

This says convolution is equivalent to multiplication in the transform domain.
Examples

- Transform of box filter is sinc.
- Transform of Gaussian is Gaussian.

(Trucco and Verri)

Implications

- Smoothing a sine or cosine just scales it (reduces amplitude).
- Smoothing a high frequency sine or cosine reduces its amplitude more than smooth a low frequency one.
- Smoothing an image reduces high frequencies more than low ones.
- Sinc function explains artifacts.
- Need smoothing before subsampling to avoid aliasing.
Sampling

Every sample gives a linear equation in $a_{i,j}$.
Need two samples for every frequency.

Previous examples

- Disintegrating Texture
  - As squares get small, frequency grows.
  - At some point, sampling is slower than frequency of changes.
- Wagon wheel.
  - As wheel moves fast, we cannot sample world that fast.
Anti-aliasing: Fixing Aliasing

- **Nyquist Frequency**: Need at least twice the highest frequency in the signal to correctly reconstruct
- **Example**
  - Phone: 700 Hz at 8 bits per sample
  - CD Player: 44.1 kHz at 16 bits per sample
- **Solutions**:
  - **Prefiltering**: Lower the maximum frequency (filter out high frequencies)
  - **Super Sampling**: Raise the sampling frequency
  - **Stochastic Sampling**: Sample randomly instead of uniformly

Strategy for anti-aliasing

- **Remove high frequencies**
  - Intuitively, we do this by averaging.
  - High frequency = rapid change
  - Averaging slows down the changes
  - This can be made more rigorous
- **Then sample**

*Demo*
Example: Smoothing by Averaging

- Rotationally symmetric.
- Weights nearby pixels more than distant ones.
  - This makes sense as probabilistic inference.

A Gaussian gives a good model of a fuzzy blob.
Smoothing with a Gaussian

Lowering the Max Frequency

Image from http://www.lunaloca.com/tutorials/antialiasing/
Lowering the Max Frequency

Prefiltering

- Blur the images by some low-pass filters
- Easy for 2D text, images, and 2D graphics
- Difficult for 3D graphics (visibility is an issue since translucency is expensive)
Prefiltering

No Antialiasing

Prefiltering

Increasing Resolution

- Physically increase the display resolution
  - Not always possible
- Perform Postfiltering / Supersampling
  - Actual display is \( m \times n \), sample on a \( sm \times sn \) virtual display
  - Then “average” several (typically \( s \times s \)) virtual sub-pixels to yield one actual pixel
  - Averaging
    - unweighted (simple average)
    - weighted (more importance to virtual sub-pixels closer to the center)
Averaging with supersampling
Stochastic Sampling

- Sample at a random place within the pixel to determine its color (**Jittering**)
- Pick a random point and keep it if its nearest neighbor is > r units away (**Poisson Disk Sampling**)
- Often use stochastic sampling with supersampling (eg: 16 stochastic samples/pixel)
- Reduces aliasing at the cost of added noise.

Sampling: take $n \times n$ jittered, regular or random samples.

- **Jittered**
- **Regular**

Averaging: use an $n \times n$ filter to assign weights to each sample. Sum all the weighted samples to obtain a pixel.
Anti-aliasing for Geometric Objects

- Use gray value proportional to area of intersection
  - Overcomes jagged edges
  - Gives equal orientation lines same brightness
  - Equivalent to supersampling with averaging in the limit of infinite supersampling.
Antialiasing Examples

No antialiasing

Antialiasing Examples

3x3 supersampling
3x3 unweighted filter

Images from SIGGRAPH 93 Educators’ Slide Set
Antialiasing Examples

3x3 supersampling
5x5 weighted filter

Antialiasing Examples

3x3 jittered supersampling
5x5 weighted filter