Texture Mapping

Reading

Required

- Hearn & Baker 14.8-14.9
- Angel, pages 373-386

Optional

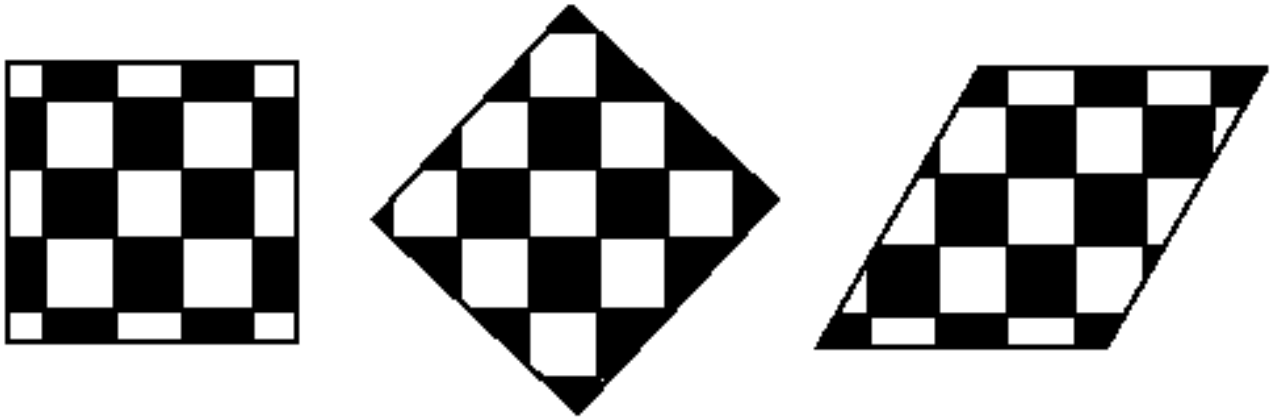
- Paul S. Heckbert. Survey of texture mapping. Computer Graphics and Applications 6(11): 56-67, November 1986

  http://www.cs.cmu.edu/afs/cs/user/ph/www/texsurv.ps.gz

Texture Mapping

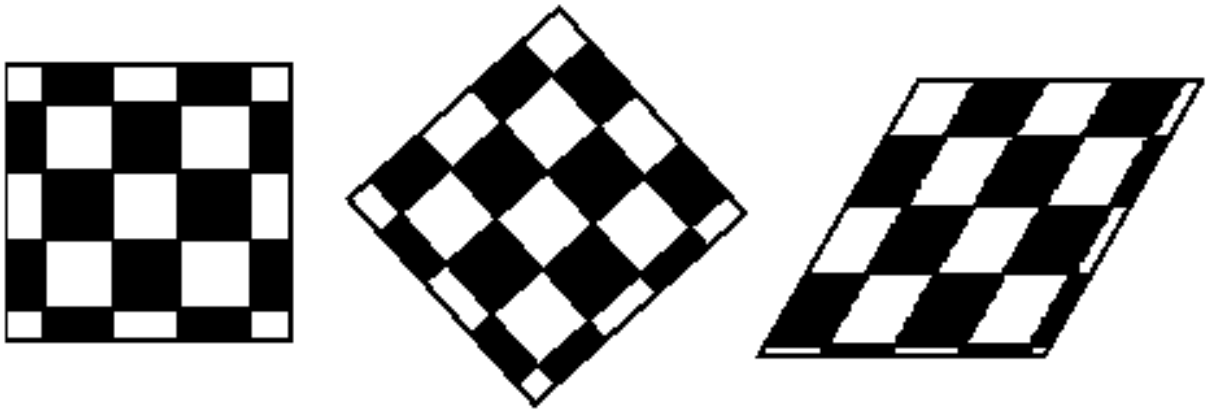
- Texture mapping allows you to take a simple polygon and give it the appearance of something much more complex
  - Due to Ed Catmull, PhD thesis, 1974
  - ensures that "all the right things" happen as a texture polygon is transformed and rendered

Non-Parametric Texture Mapping



- With non parametric texture mapping:
  - Texture size and orientation are fixed
  - Unrelated to size and orientation of polygon
  - Gives a cookie-cutter effect

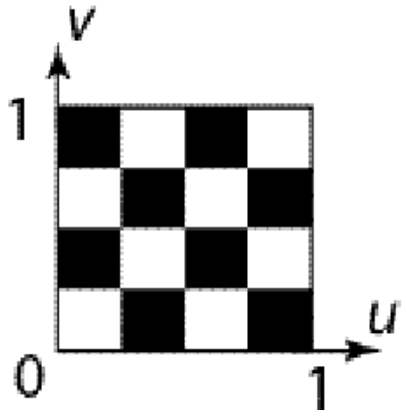
Parametric Texture Mapping



- With parametric texture mapping, texture size and orientation are tied to the polygon:
  - Separate texture space and screen space
  - Texture the polygon as before but in texture space
  - Deform (render) the textured polygon into screen space

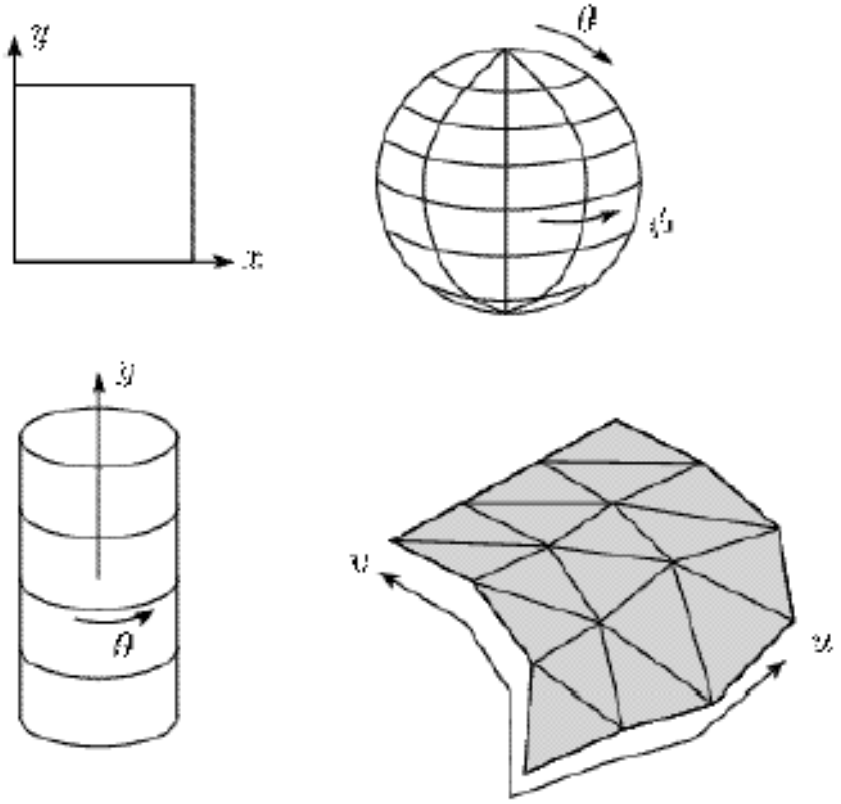
Implementing Texture Mapping

- A texture lives in it own image coordinates paramaterized by (*u*,*v*):



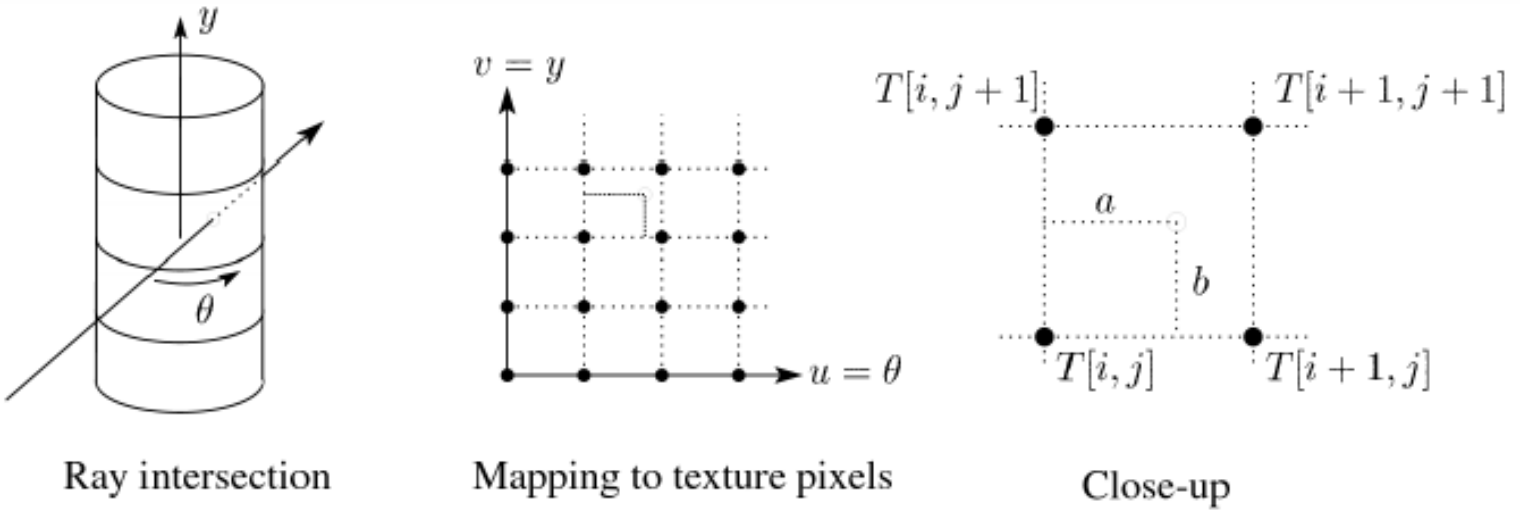
Implementing Texture Mapping

- It can be wrapped around many different surfaces:



Texture Resampling

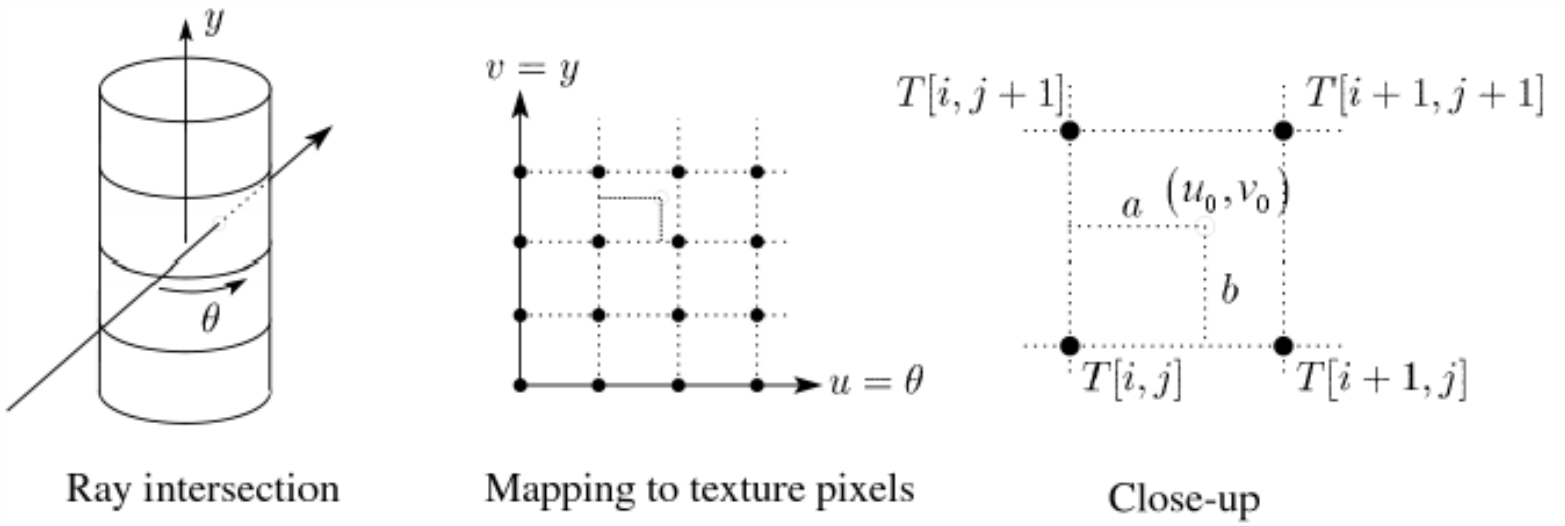
- What do we do when the texture sample lands between texture pixels?



Ray intersection Mapping to texture pixels Close-up

- We resample. Common choice is **bilinear resampling**.

Bilinear Resampling



Ray intersection

Mapping to texture pixels

Close-up

$T(u_0, v_0) =$

$T(i\Delta + a, j\Delta + b) =$ _____ $T[i,j]$ +

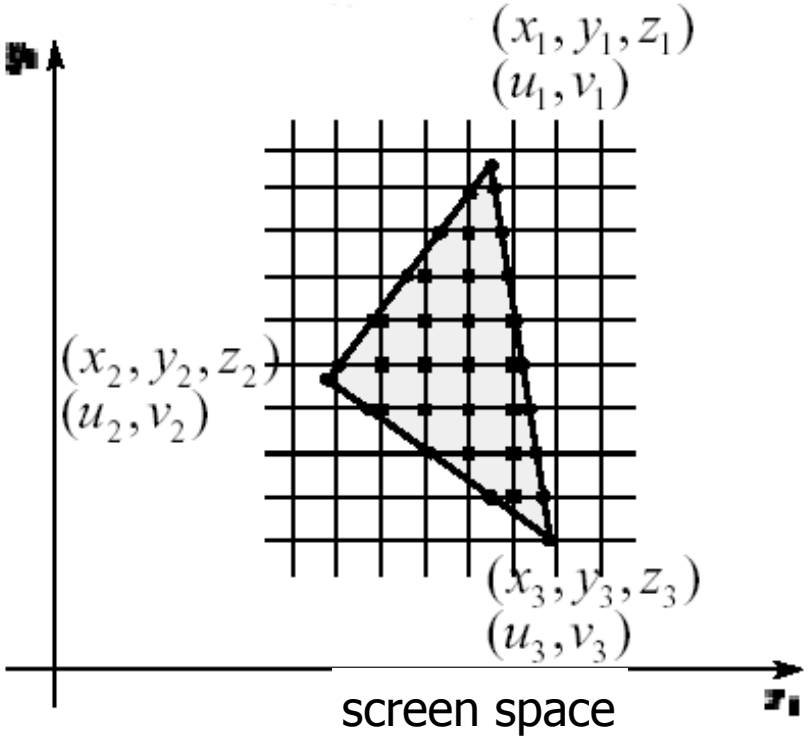
_____ $T[i+1,j]$ +

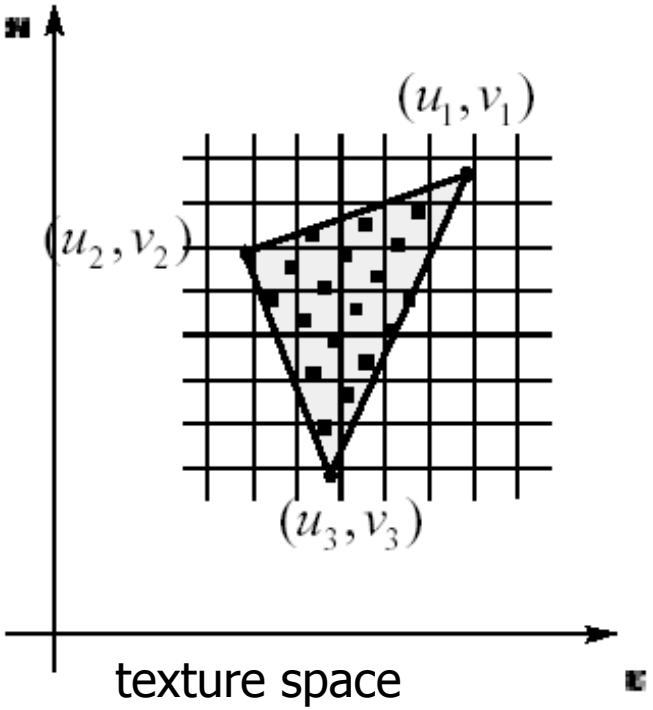
_____ $T[i,j+1]$ +

_____ $T[i+1,j+1]$

Implementing, cont'd

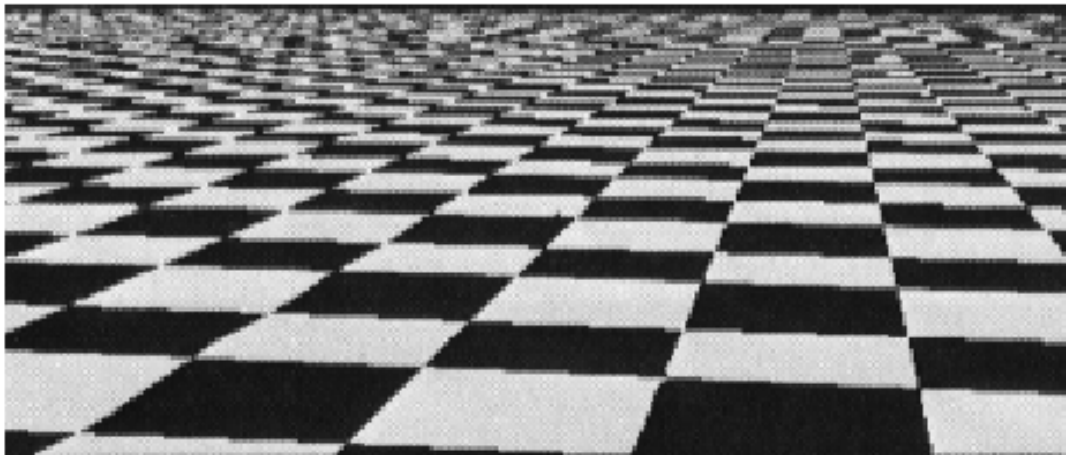
- Texture mapping can also be handled in z-buffer algorithms
  - Scan conversion is done in screen space, as usual
  - Each pixel is colored according to the texture
  - Texture coordinates are found by Gouraud-style interpolation





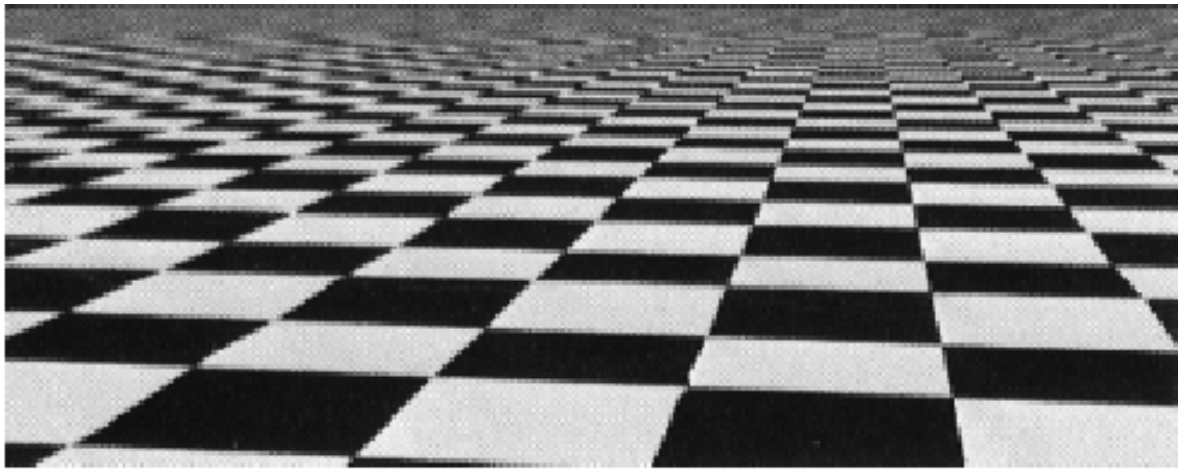
Antialiasing

- If you point-sample the texture map, you get aliasing (stair-casing effect):



Antialiasing

- Proper antialiasing requires area averaging in the texture:

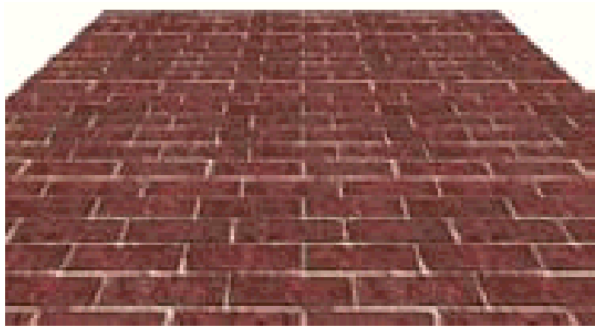


What is Aliasing?

- This is the phenomenon that occurs when we undersample a signal. It causes certain high frequency feature to appear as low frequencies.
- To eliminate aliasing we have to either band limit (low pass filter) our input signal or sample it at a higher rate.
- The first approach (i.e. prefiltering) is more efficient.

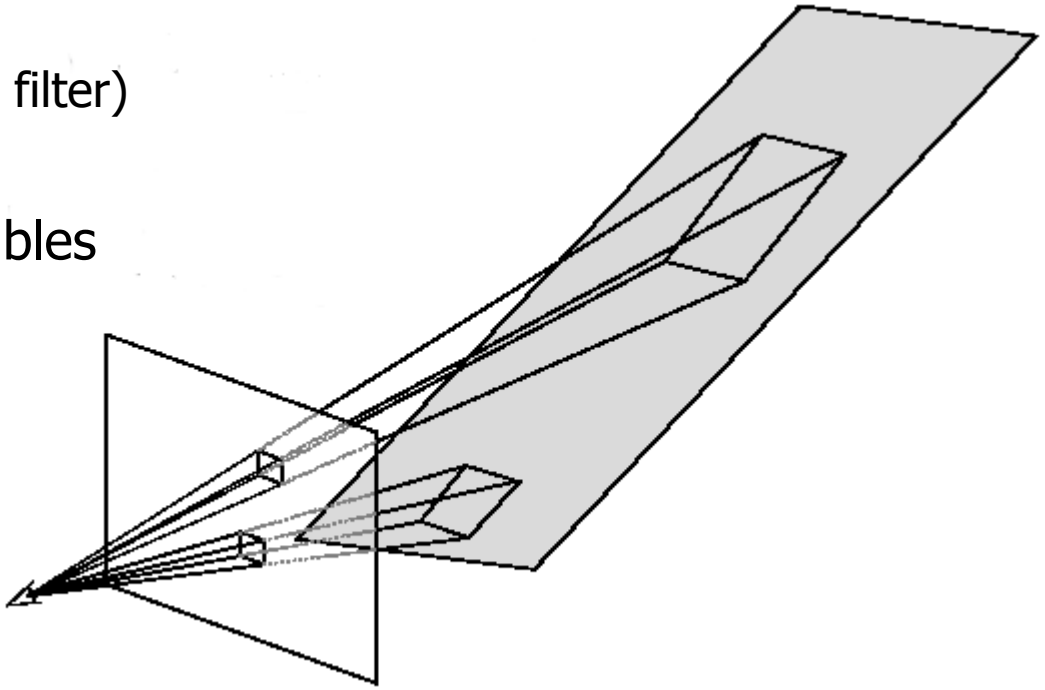
Spatial filtering

- We prefilter the texture to remove the high frequencies that show up as artifacts in the final rendered image.
- The prefiltering required in the undersampling case is basically a spatial integration (summing) over the extent of the sample.



Computing Average Colors

- Computationally difficult part is prefiltering (summing) over the covered pixels:
- Several methods have been used:
  1. Brute force
     Just sum (mean filter)
  2. Mip maps
  3. Summed Area Tables

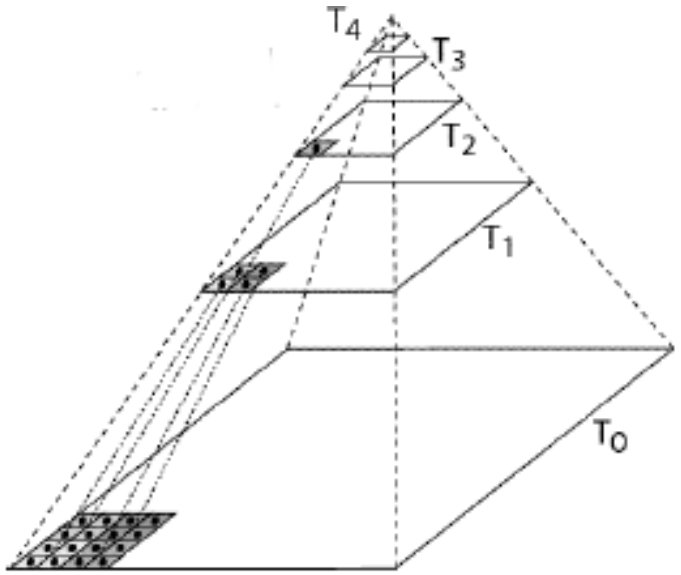


Mip Maps

- Lance Williams, 1983
  - "multum in parvo" – many things in a small place"
  - Keep textures prefiltered at multiple resolutions
  - Figure out two closest levels
  - Linear interpolate between the two

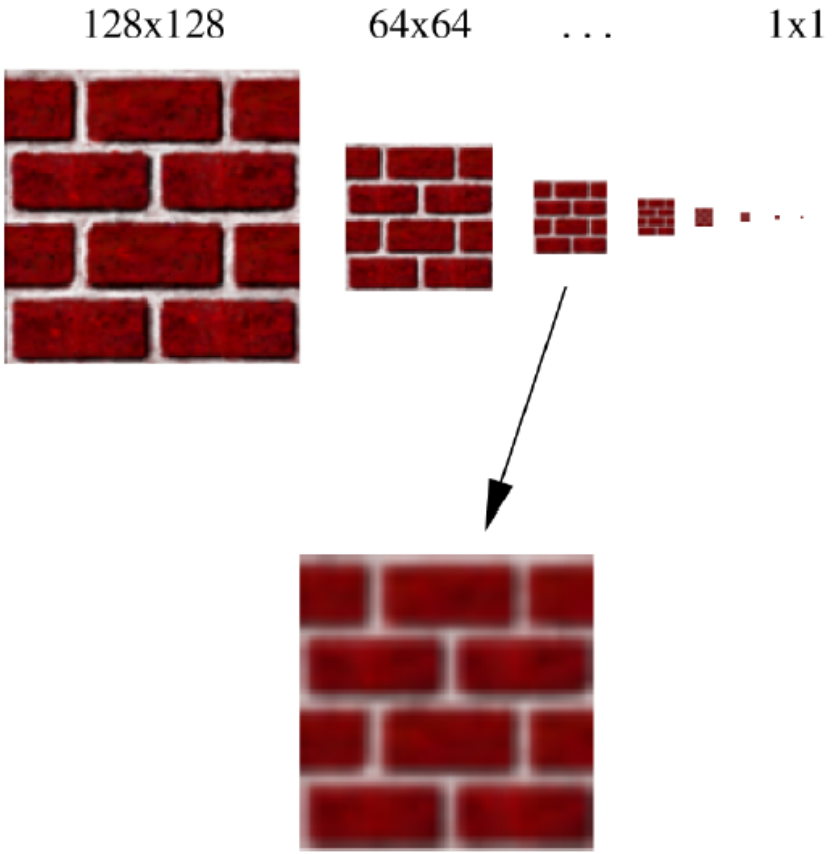
Mip Map Pyramid

- The mip map hierarchy can be thought of as an image pyramid:
  - Level 0 (T0[i,j]) is the original image.
  - Level 1 (T1[i,j]) averages over 2x2 neighborhoods of original.
  - Level 2 (T2[i,j]) averages over 4x4 neighborhoods of original
  - Level 3 (T3[i,j]) averages over 8x8 neighborhoods of original



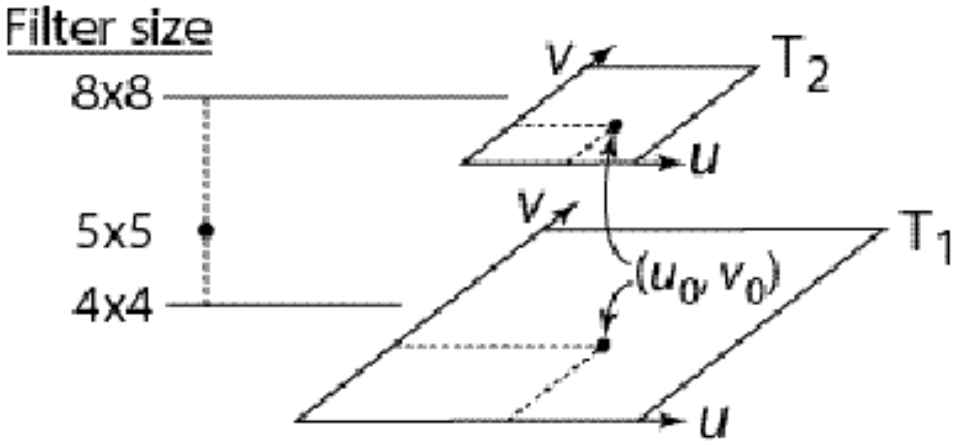
Mip Maps

1. Figure out two closest levels
2. Linear interpolate between the two



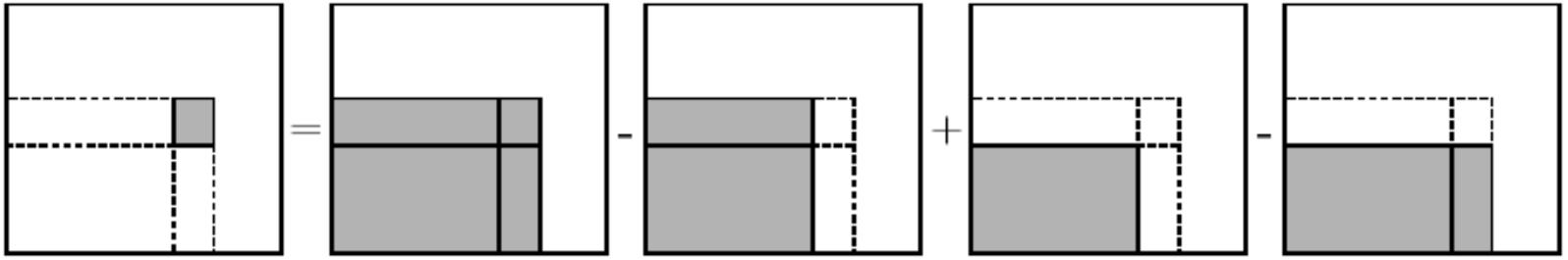
Mip Map Resampling

- What would the mip-map return for an average over a 5x5 neighborhood at location (u_0, v_0)?



- How do we measure the fractional distance between levels?
- What if you need to average over a non-square region?

Summed Area Table (SAT)

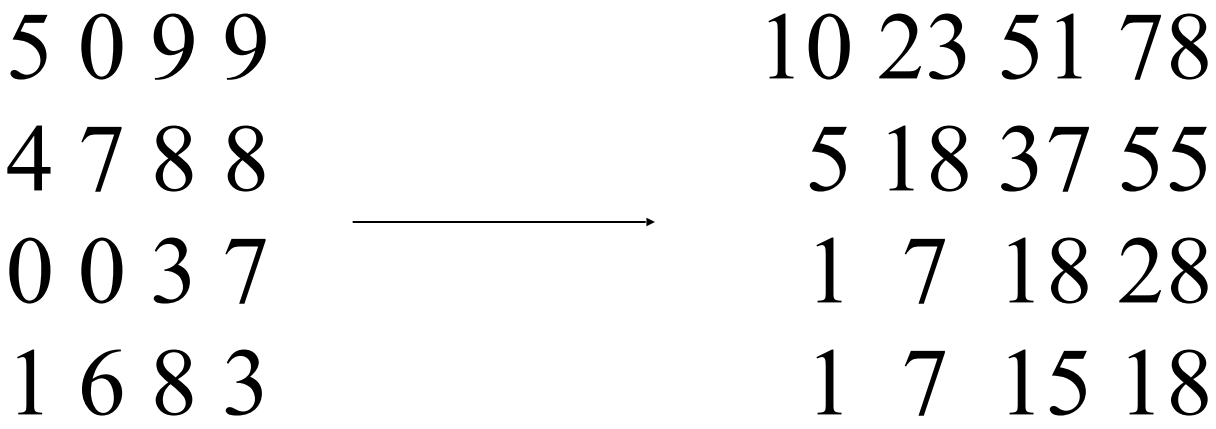


- Recall in Calculus (in continuous or discrete form):

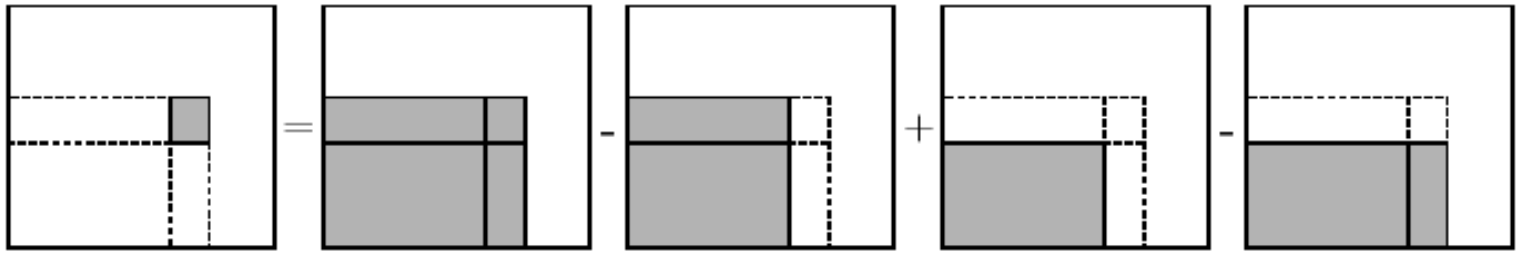
$$\int_a^b f(x)dx = \int_{-\infty}^{b} f(x)dx - \int_{-\infty}^{a} f(x)dx$$

$$\sum_{i=k}^{m} f[i] = \sum_{i=0}^{m} f[i] - \sum_{i=0}^{k} f[i]$$

What is a SAT?



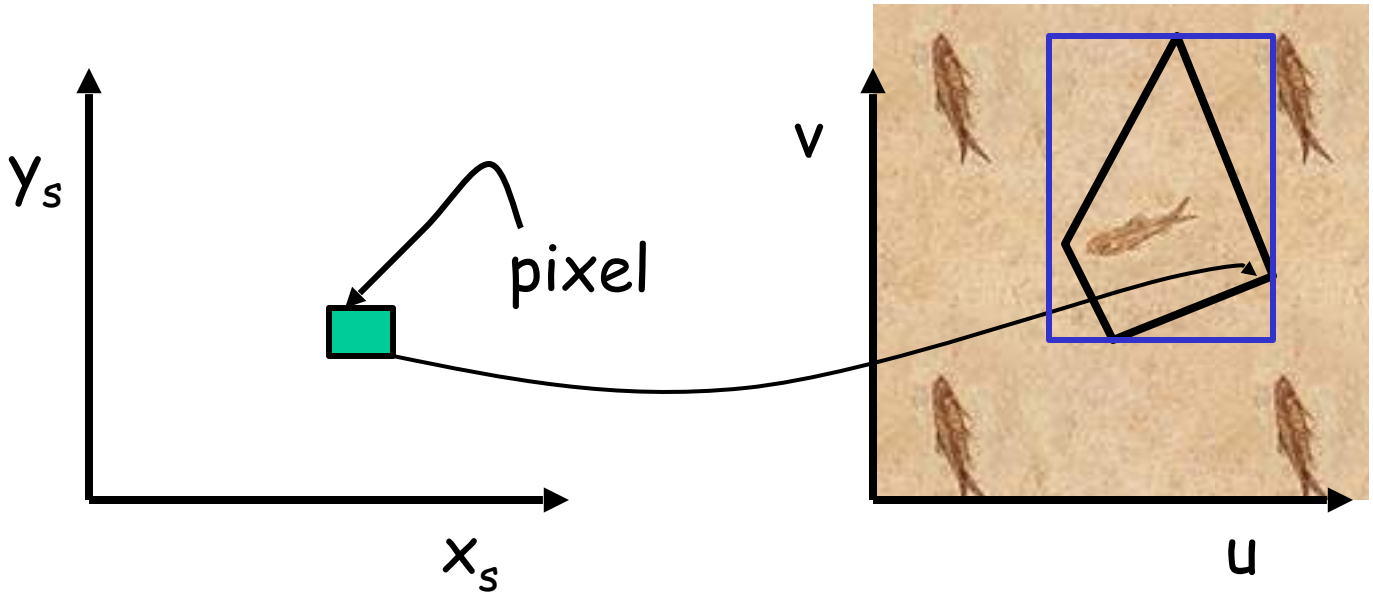
Summed Area Table



- Due to Frank Crow (1984).
- Keep sum of everything below and to the left.
- Use four table lookup.
- Requires more memory (2-4 times the original)
- Gives less blurry results.

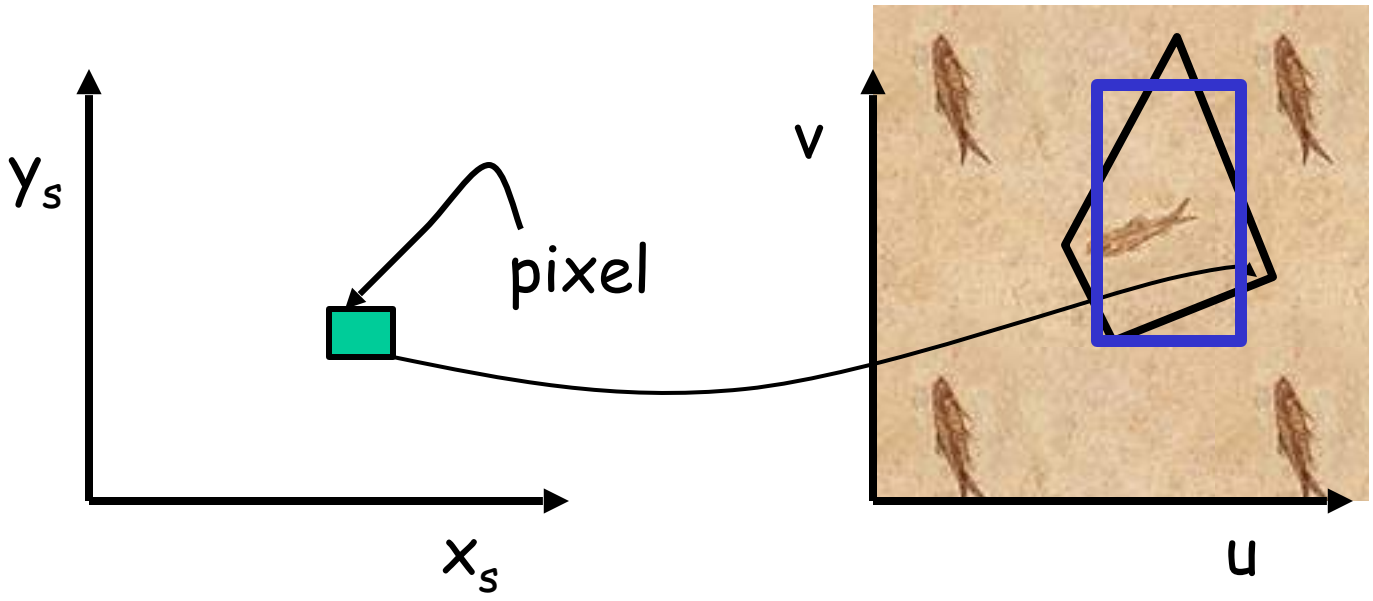
Summed Area Table (SAT)

- Determining the rectangle:
  - Find bounding box and calculate its aspect ratio



Summed Area Table (SAT)

- Determine the rectangle with the same aspect ratio as the bounding box and the same area as the pixel mapping.

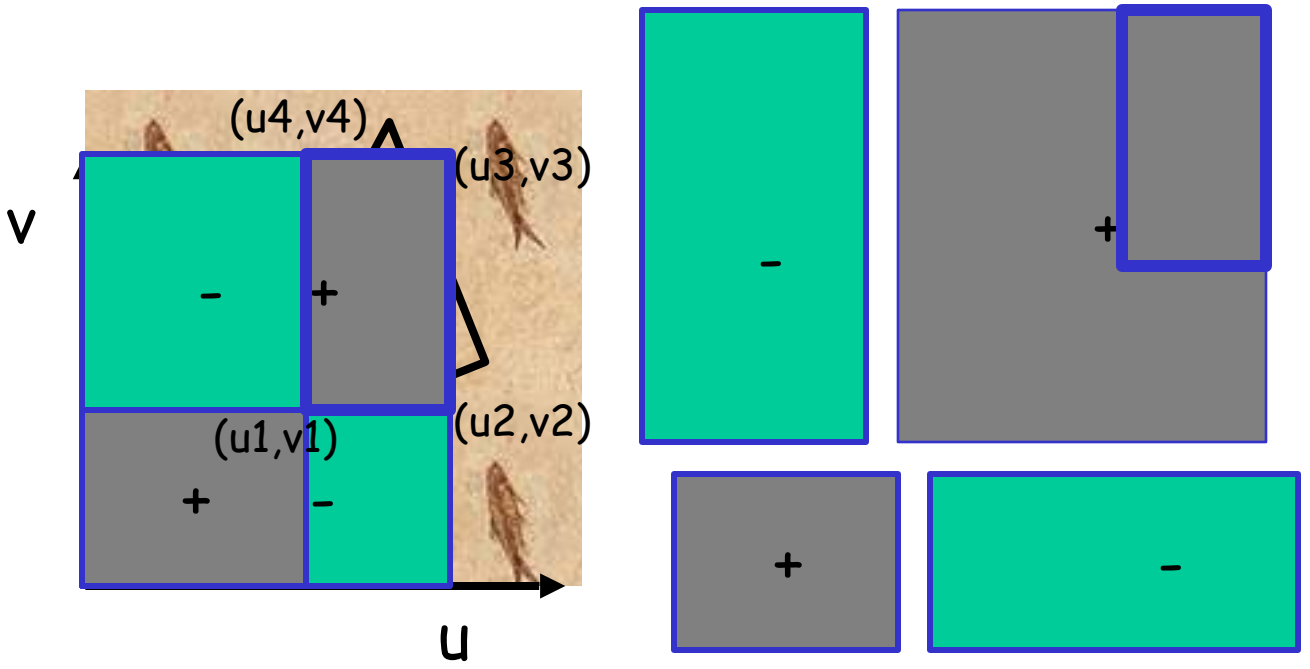


Summed Area Table (SAT)

- Center this rectangle around the bounding box center.
- Formula:
    - Area = aspect_ratio*x*x
    - Solve for x – the width of the rectangle
- Other derivations are also possible using the aspects of the diagonals, …

Summed Area Table (SAT)

- Calculating the color
  - We want the average of the texture element colors within this rectangle

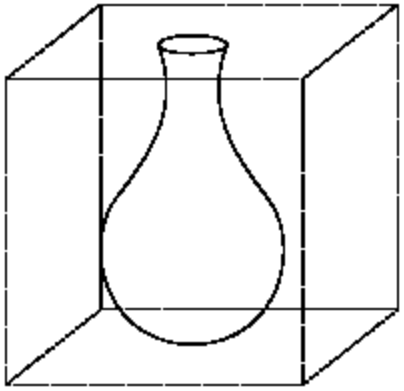


Summed Area Table (SAT)

- To get the average, we need to divide by the number of texels falling in the rectangle.
  - Color = SAT(u3,v3)-SAT(u4,v4)-SAT(u2,v2)+SAT(u1,v1)
  - Color = Color / ((u3-u1)*(v3-v1))

Solid Textures

- Q: What kinds of artifacts might you see from using a marble wallpaper instead of a real marble?

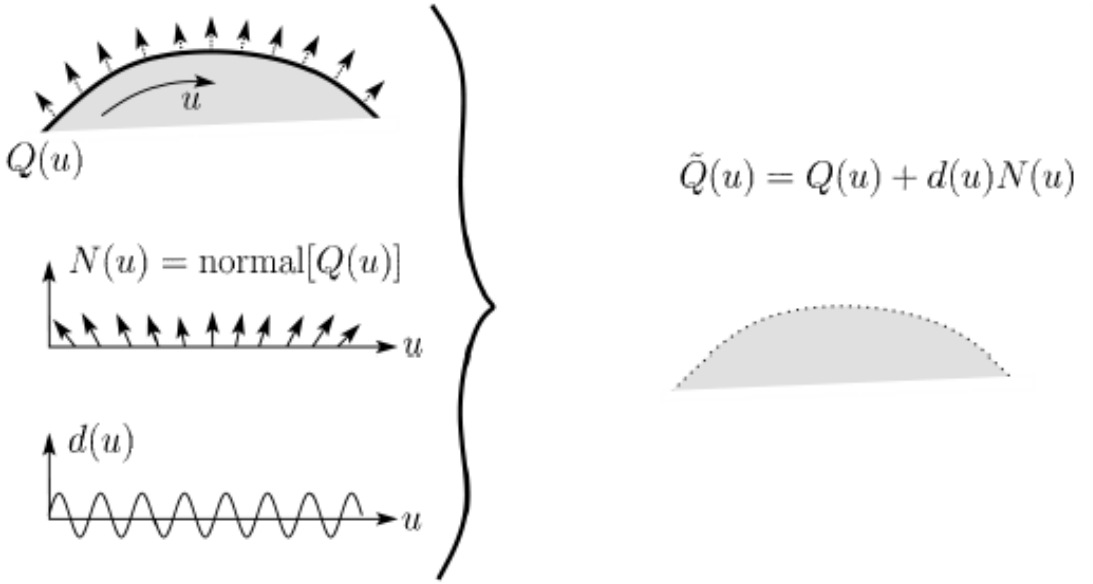


  - One solution is to u
  - Use model-space coordinates to index into a 3D texture.
  - Like "carving" the object from the material
- The mathematics is easy; but coming up with the solid texture is difficult…

Solid Textures

Displacement Mapping

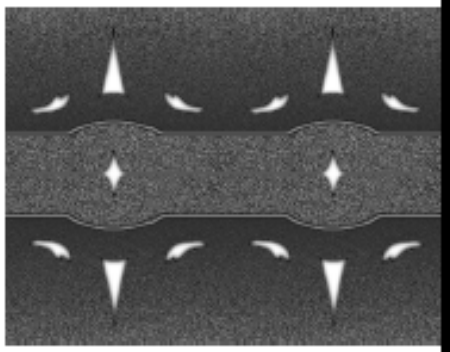
- In displacement mapping, a texture is used to perturb the surface geometry itself:



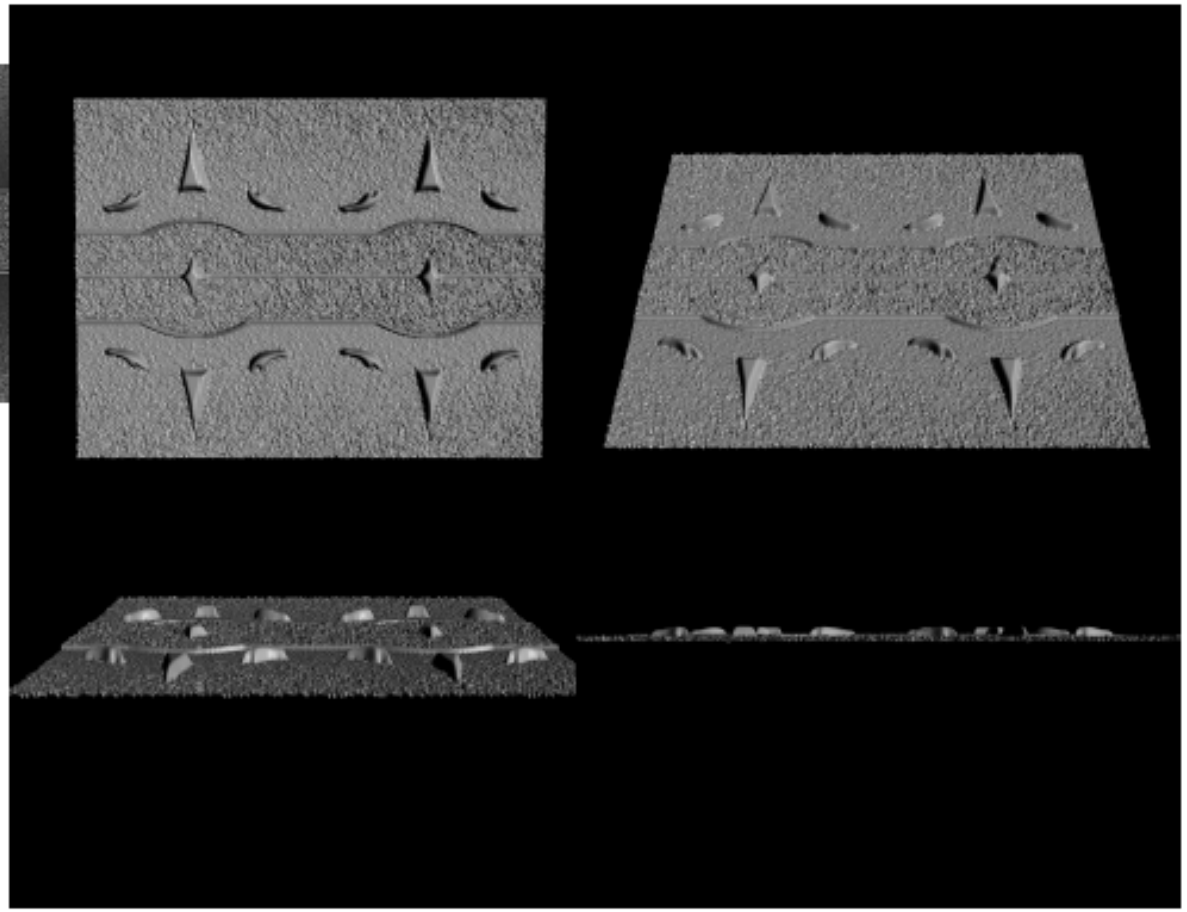
- Silhouettes are correct.
- Requires doing additional hidden surface calculations.

Displacement Mapping

Input texture:



Displacement map over rectangular surface:



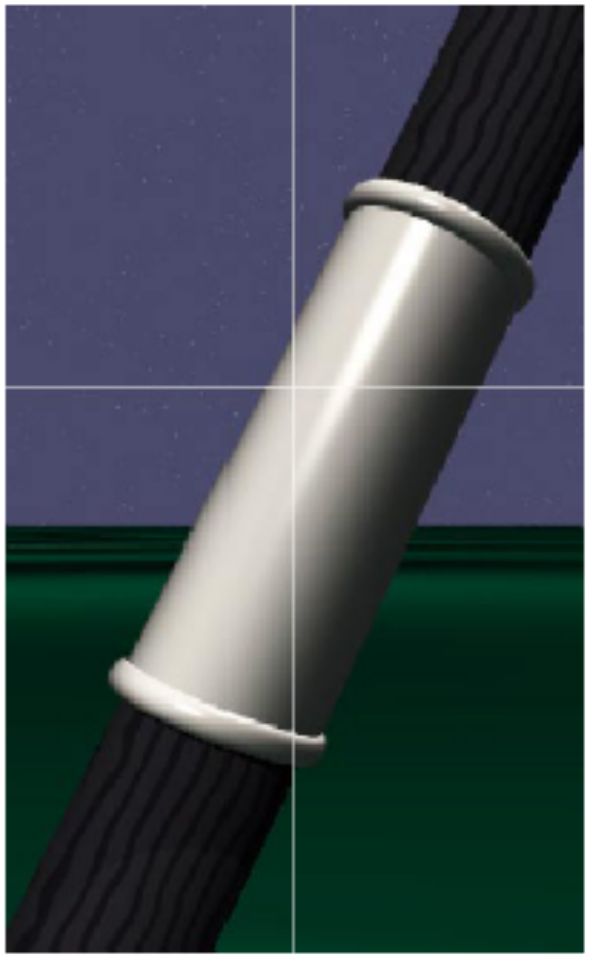
Bump Mapping

- Textures can be used for more than just color

$$I = k_a I_a + \sum_i f(d_i) I_{li} \left(k_d (\mathbf{N} \cdot \mathbf{L}_i)_+ + k_s (\mathbf{V} \cdot \mathbf{R})_+^{n_s} \right)$$

- In bump mapping, a texture is used to perturb the normal:
  - The normal is perturbed the normal according to the partial derivatives of the texture.
  - These bumps 'animate

Bump Mapping Example

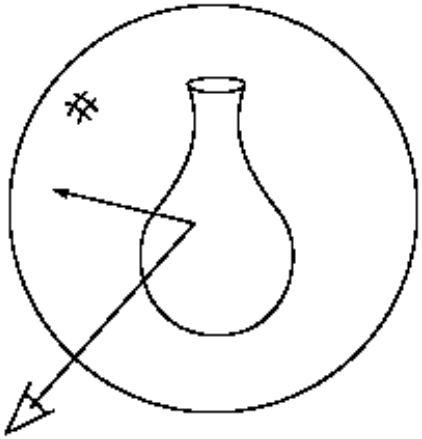


Original rendering

Rendering with bump map wrapped around a cylinder

Environment Mapping

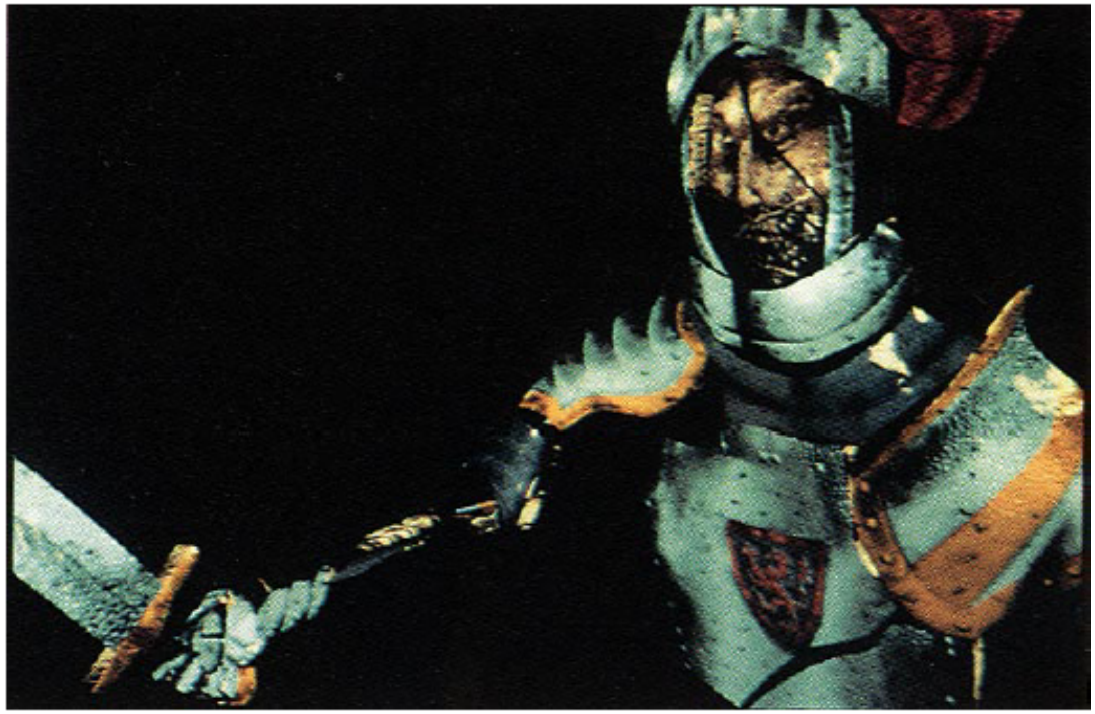
Environment Mapping



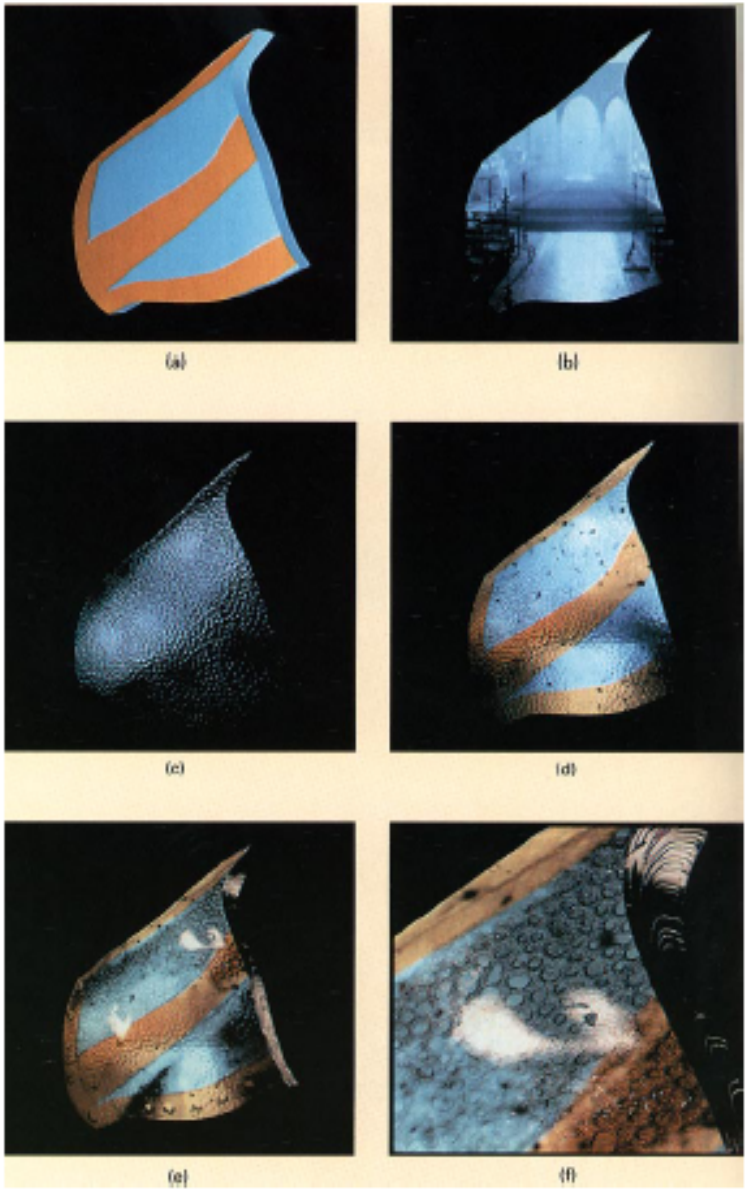
- A.k.a. reflection mapping
- Use texture to model object's environment
- Rays are bounced off objects into environment to determine color of illumination
- Works well when there is just a single object
- With some simplifications can be implemented in hardware
- Ray tracer can be extended to handle refractions as well

Combining Texture Maps

- Using texture maps in combination to achieve better effects.



Combining Texture Maps



Summary

What to take from this lecture:

- What texture mapping is and what is it good for
- Understanding the various approaches to antialiased textured mapping
  - Brute force
  - Mipmaps
  - Summed area tables
- Additional effect with texture mapping techniques
  - Bump mapping
  - Displacement mapping
  - Environment mapping