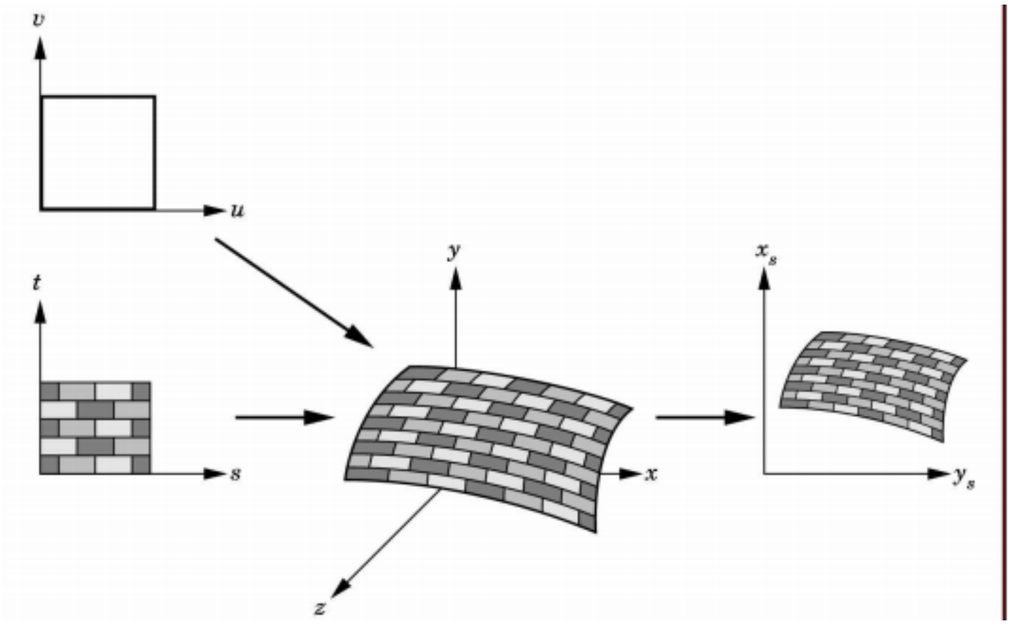
Ray Tracing and Texturing (2)

Texture mapping & Code skeleton

Yuehao Wang; Apr 15, 2021

2D Texture Mapping

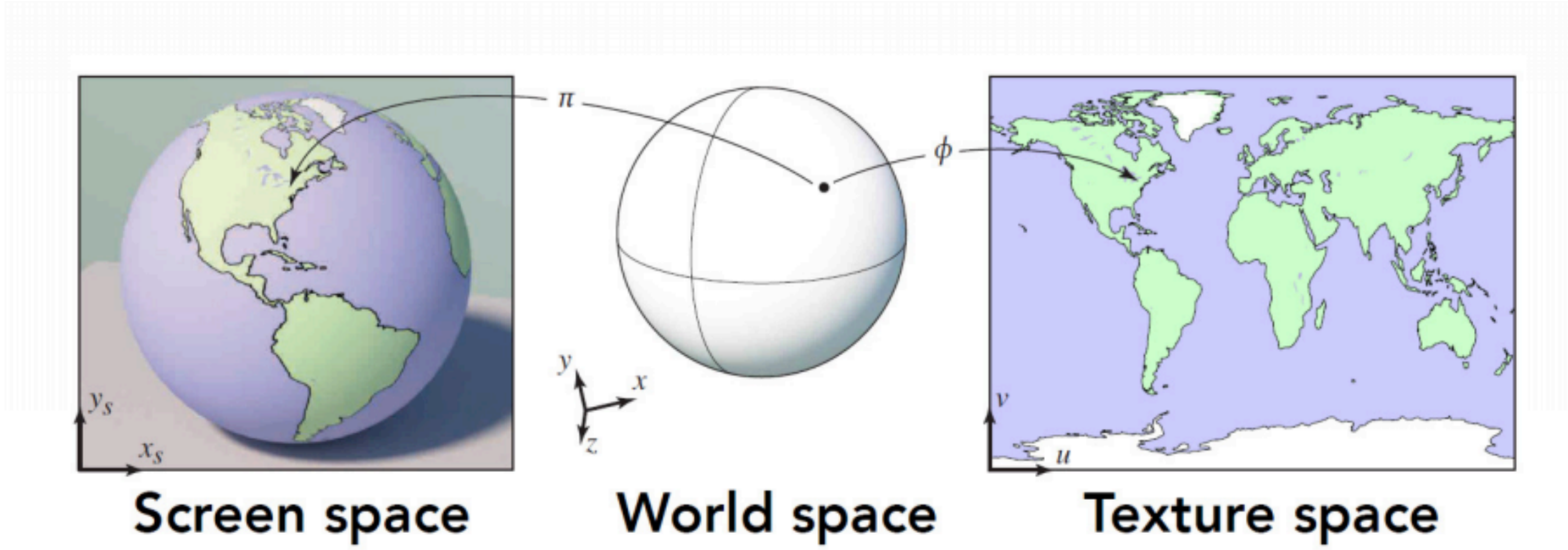
- Make our surfaces more life-like
  - Retrieve Texture: scanned from the world, generate
  - Texture stored in a 2D image
- Mapping
  - For each texture image, we have horizontal and vertical axes (u, v), where u and v vary from 0 to 1.
  - We need a mapping from points on the surface to (u, v) on the image.



Review Texturing

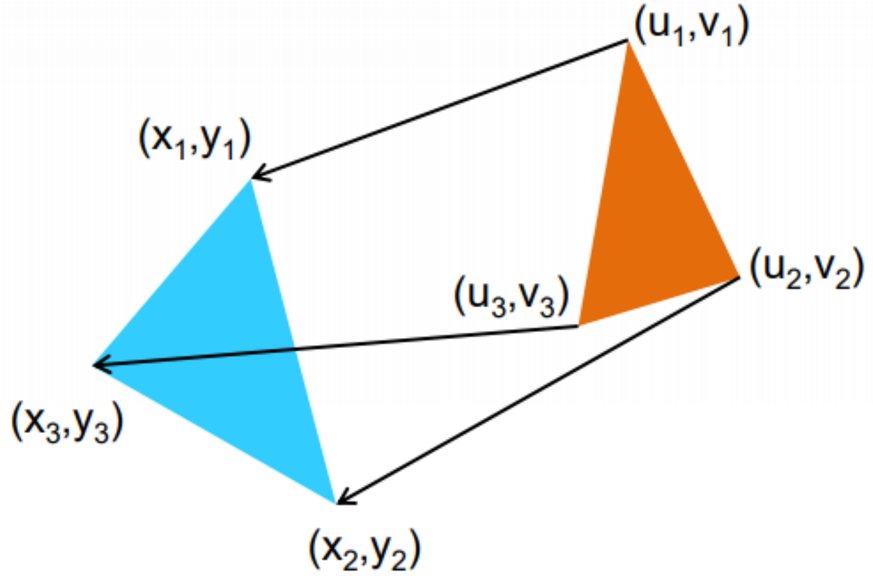
How to Find the Mapping

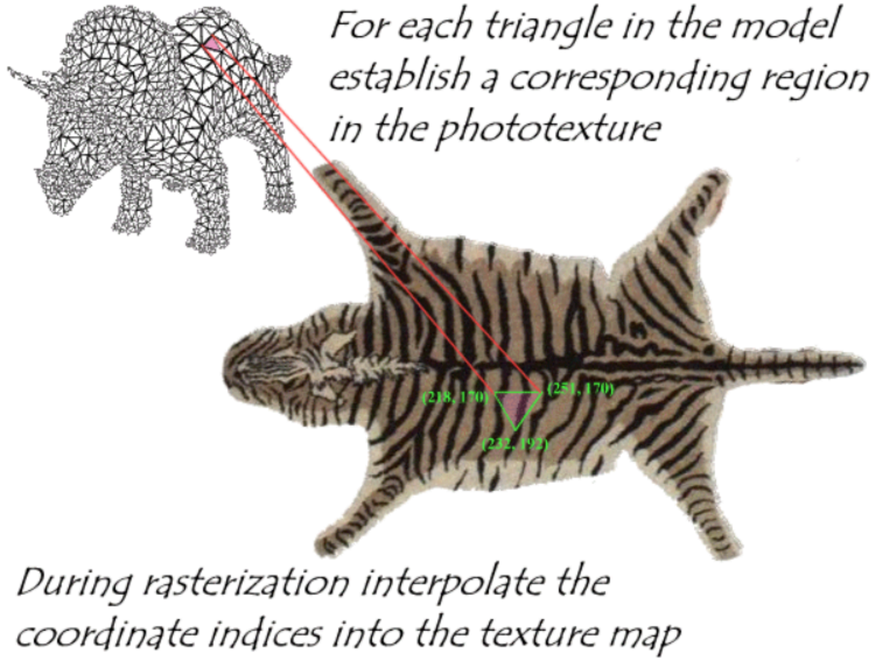
- Parameterized geometries
  - Sphere: $(\phi, \theta) = (2\pi u, \pi v)$, ϕ is azimuth, θ is zenith
  - Cylinder: $(h, \theta) = (u, 2\pi v)$, h is height, θ is azimuth



How to Find the Mapping

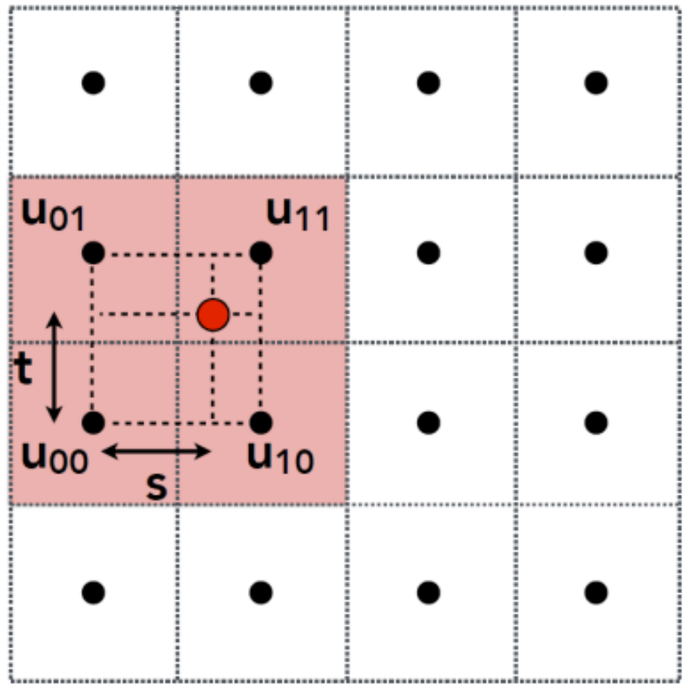
- Meshes
  - Specify uv coordinates for each triangle vertex.
  - For each point inside the triangle, compute uv as the interpolation of vertices' uvs according to its barycentric coordinate.





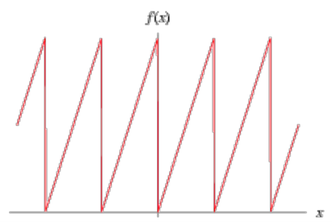
Retrieve Values of UV Mapping

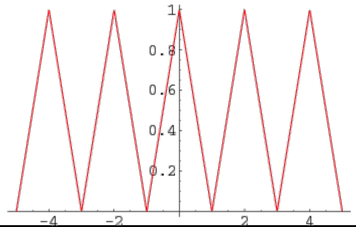
- The uv space is a continuous manifold. However, textures are discretized pixel arrays.
- Retrieval strategies (Texture filtering)
  - Nearest neighbor, Bilinear, Bicubic
- Mipmap/Ripmap
  - Generate multi-scale textures using image filters
  - Handle texturing for delicate objects
- Please refer to Lecture 08.

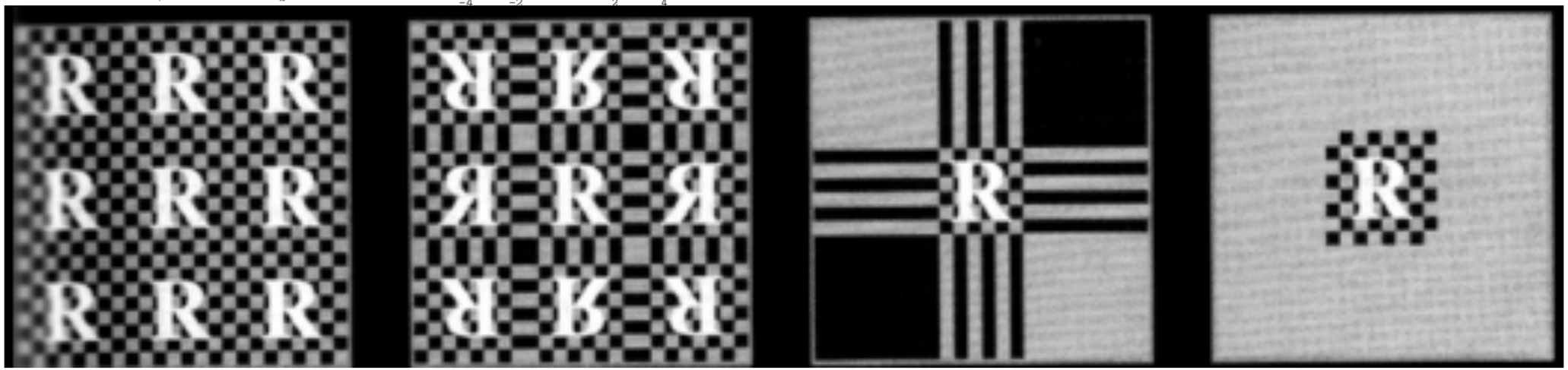


Texture Wrapping

- Strategies for handling boundaries of textures
  - Repeat: Sawtooth wave
  - Mirror: Triangle wave
  - Clamp to edge / Clamp to a fixed color







HW3 Code Skeleton