Textures

Fragment shader:
Textures, bump maps, normal maps, parallax mapping

Vertex shader: Displacement maps

http://www.opengl-tutorial.org/intermediate-tutorials/tutorial-13-normal-mapping/

Textures

bump maps imitate $\mathbf{p}^{\text{new}} = \mathbf{p} + d\mathbf{n}.$ for lighting

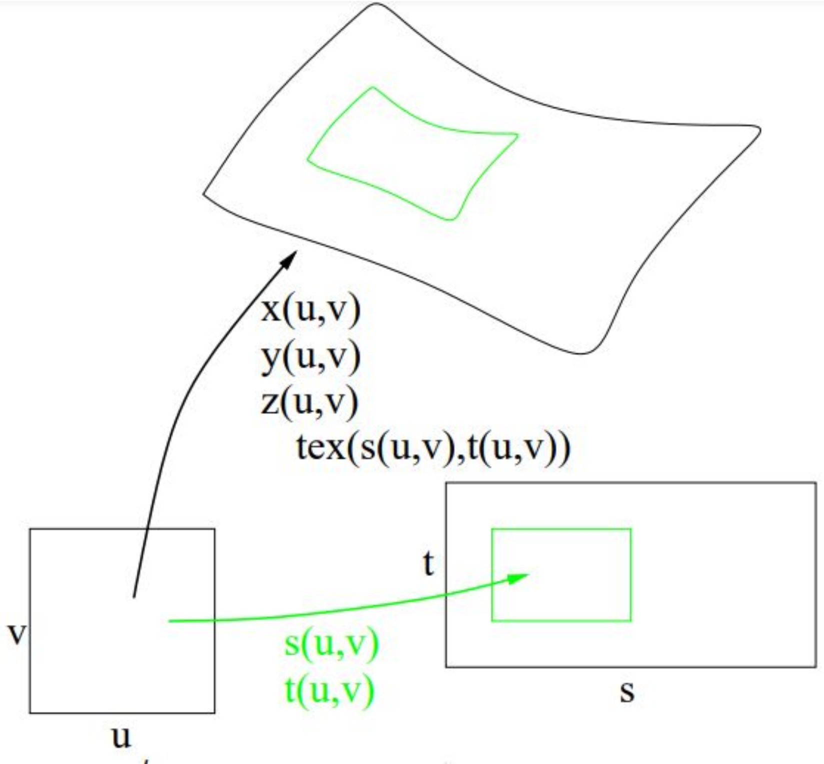
$$\frac{\partial \mathbf{p}^{\text{new}}}{\partial u} \times \frac{\partial \mathbf{p}^{\text{new}}}{\partial v} = (\frac{\partial \mathbf{p}}{\partial u} + \frac{\partial d}{\partial u}\mathbf{n} + d\frac{\partial \mathbf{n}}{\partial u}) \times (\frac{\partial \mathbf{p}}{\partial v} + \frac{\partial d}{\partial v}\mathbf{n} + d\frac{\partial \mathbf{n}}{\partial v})$$

$$= \mathbf{n} + \underbrace{\frac{\partial d}{\partial u}\mathbf{n} \times \frac{\partial \mathbf{p}}{\partial v} - \frac{\partial d}{\partial v}\mathbf{n} \times \frac{\partial \mathbf{p}}{\partial u}}_{\partial \mathbf{n}} + O(\frac{\partial \mathbf{n}}{\partial u}, \frac{\partial \mathbf{n}}{\partial v}).$$

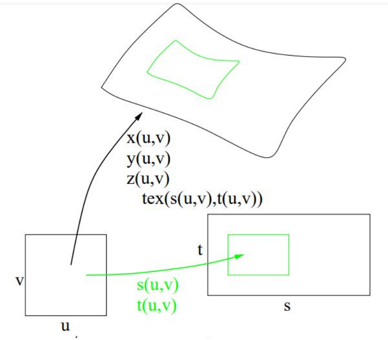
Texture Map

Fragment shader: Texture mapping

2D texture: pasting an image onto a surface (challenges: distortion and aliasing)



2D Texture



texels (texture pixels)

Many bit patterns (formats) (gimp exports C-arrays!)

- fill unit square
  access outside square → texture wrapping

- Choice of: GL LINEAR, GL NEAREST

- Mipmapping

- **Transfer texture** from an intermediate object (sphere or cylinder) for better parametrization
- **Video texture**

Textures

Fragment shader:

Environment Map, **cube map**:
place viewer at object center. Transfer resulting image as texture (possibly via intermediate)

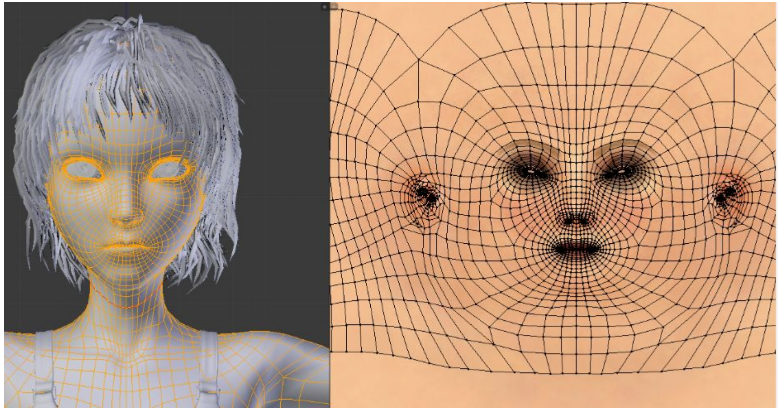
3D texture: generated (random), x,y,z direct, discrete grid

Texture Mapping: Projections

- ➢ Distortion (flat → sphere): fundamental!
- ➢ “uv-unwrapping” in Blender

Peters projection

Other projections



Afine Connection (by Cartan) local coordinates similar to the Frenet frame for curves!
connects nearby tangent spaces for vector fields (Riemann)

Exponential (polar) map for texture flattening
(exponential maps in Lie theory)

Texture Mapping Challenges

- Distortion
- Want to color pixel
  - map screen coordinates ⟷ texture coordinates

http://www.opengl-tutorial.org/beginners-tutorials/tutorial-5-a-textured-cube/

Texture Mapping Challenges

- Distortion
- screen / texture coordinates
- Areas, not points should be mapped →
  bilinear interpolation

- Aliasing (Moire pattern)
  - pointwise: might miss, average, smears out