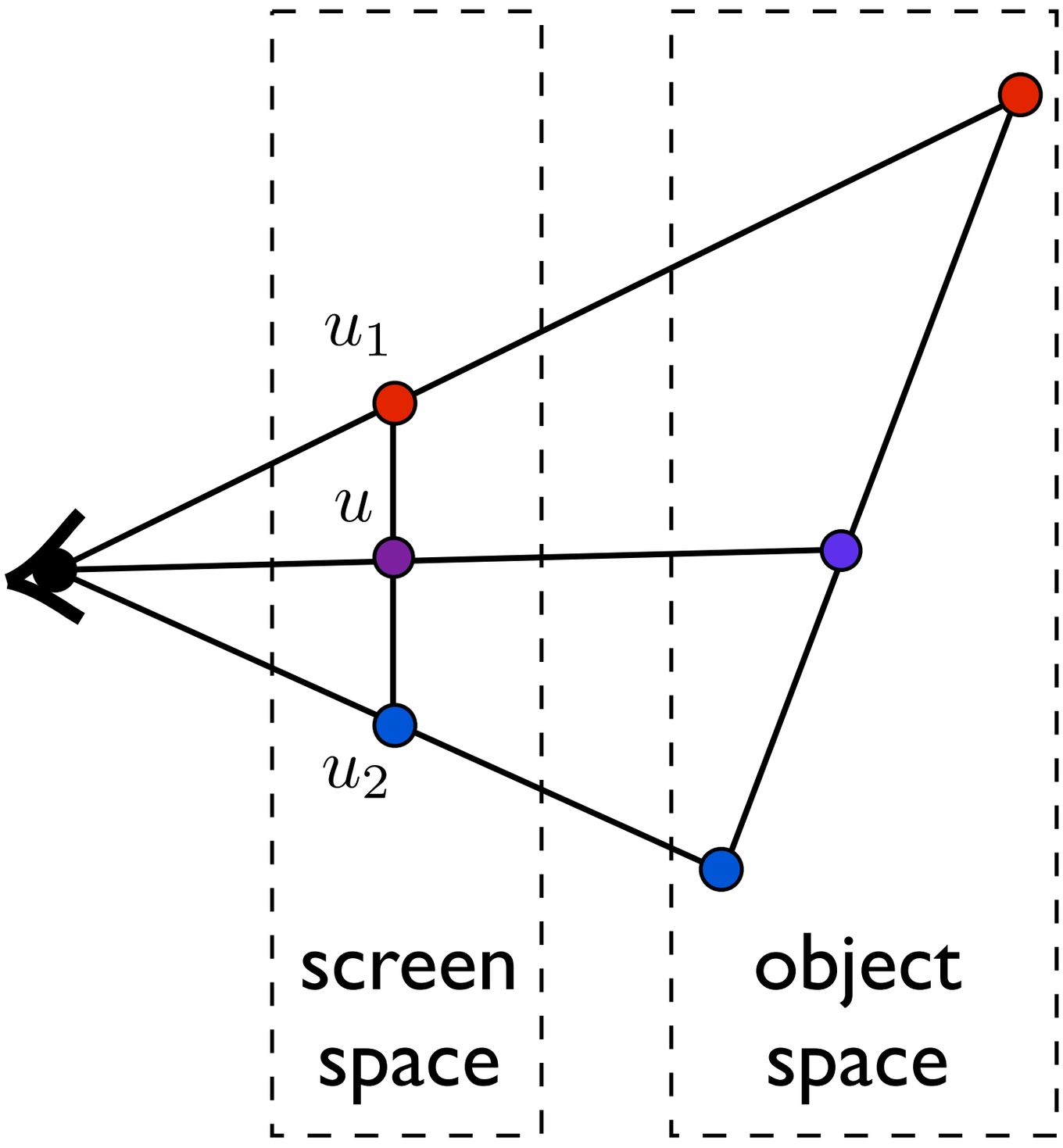
CS130 : Computer Graphics

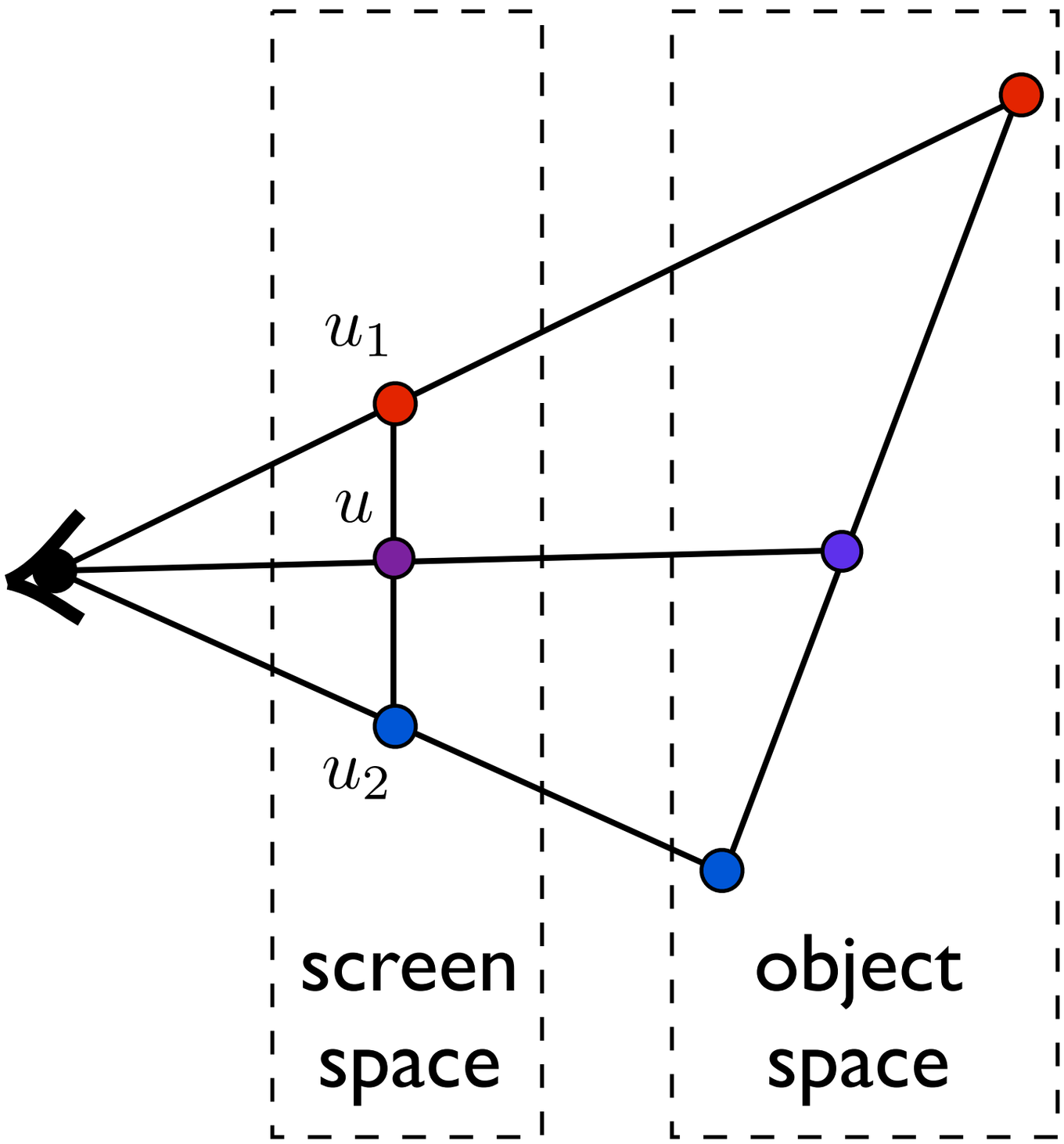
Lecture 17: Texture Mapping (cont.)

Tamar Shinar
Computer Science & Engineering
UC Riverside

Perspective correct interpolation



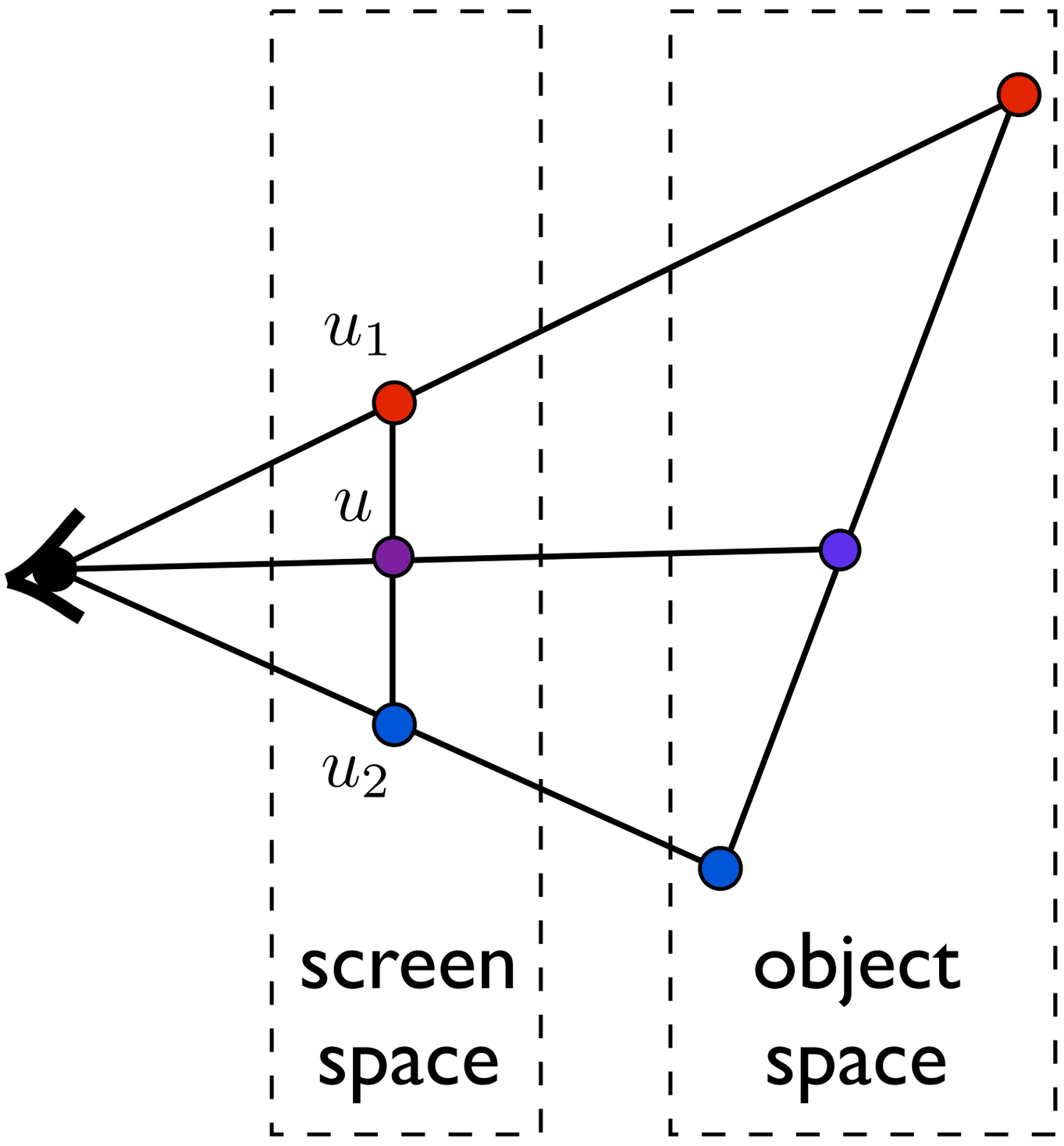
$$u = \frac{1}{2}u_1 + \frac{1}{2}u_2$$



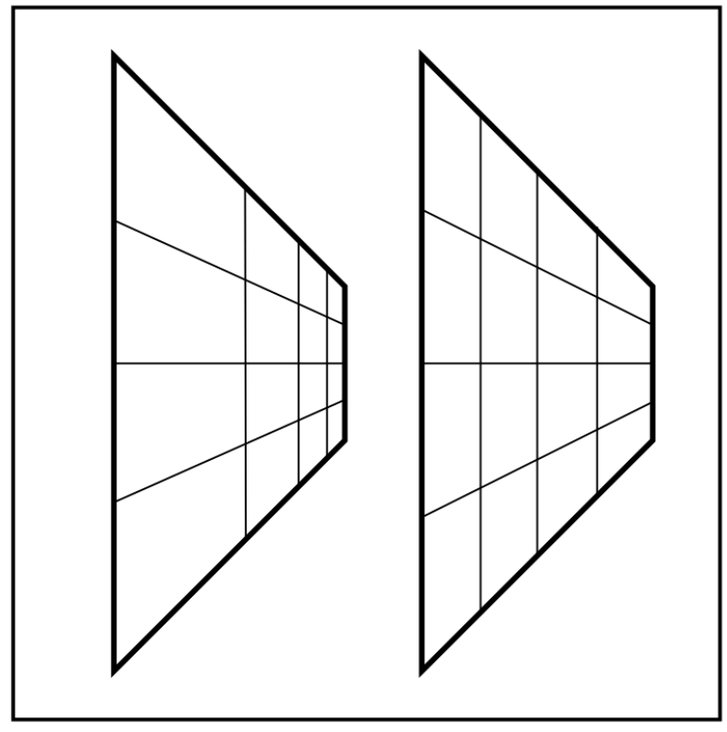
$$u = \frac{1}{2}u_1 + \frac{1}{2}u_2$$

Issue: to shade a fragment which is part of a textured triangle
we need the barycentric coordinates of the fragment
These will be the weights for the weighted average of the vertex texture coordinates.
However, after a perspective transformation, the relative distances inside the triangle have been distorted due to foreshortening.
I need to get my weights based on object or world space coordinates.

Interpolation with screen space weights is incorrect

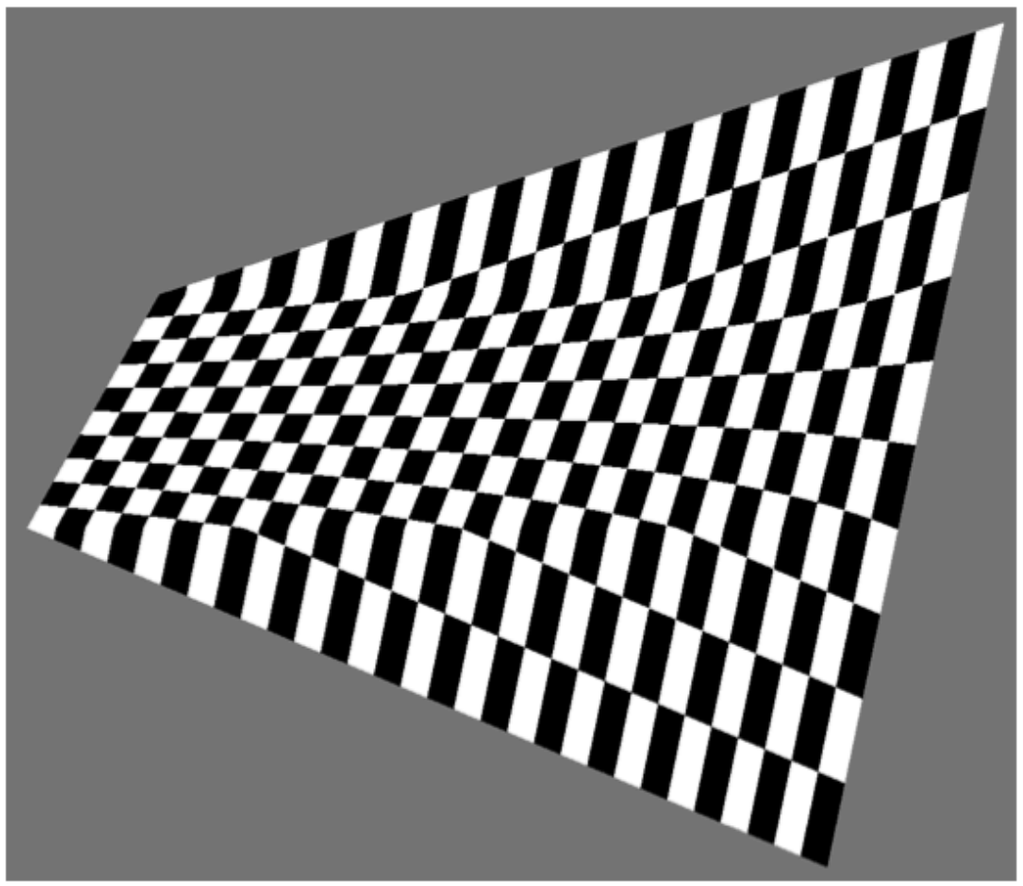


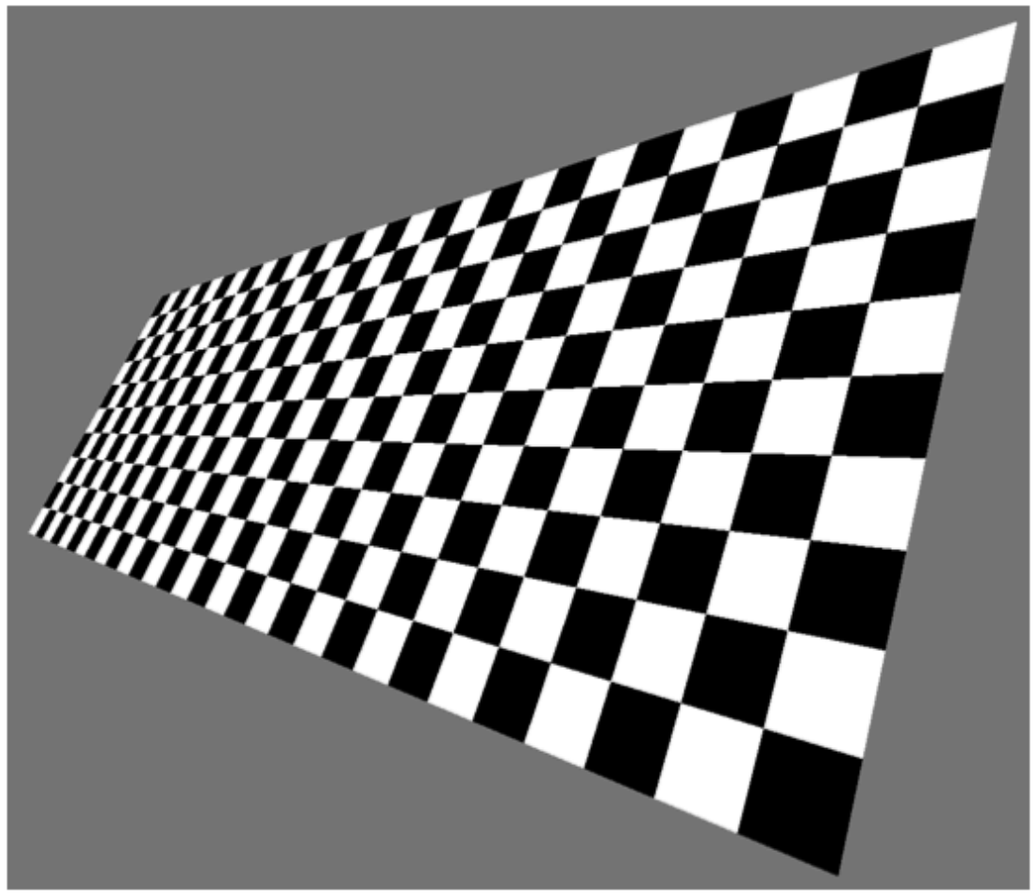
$$u = \frac{1}{2}u_1 + \frac{1}{2}u_2$$



Perspective correct interpolation

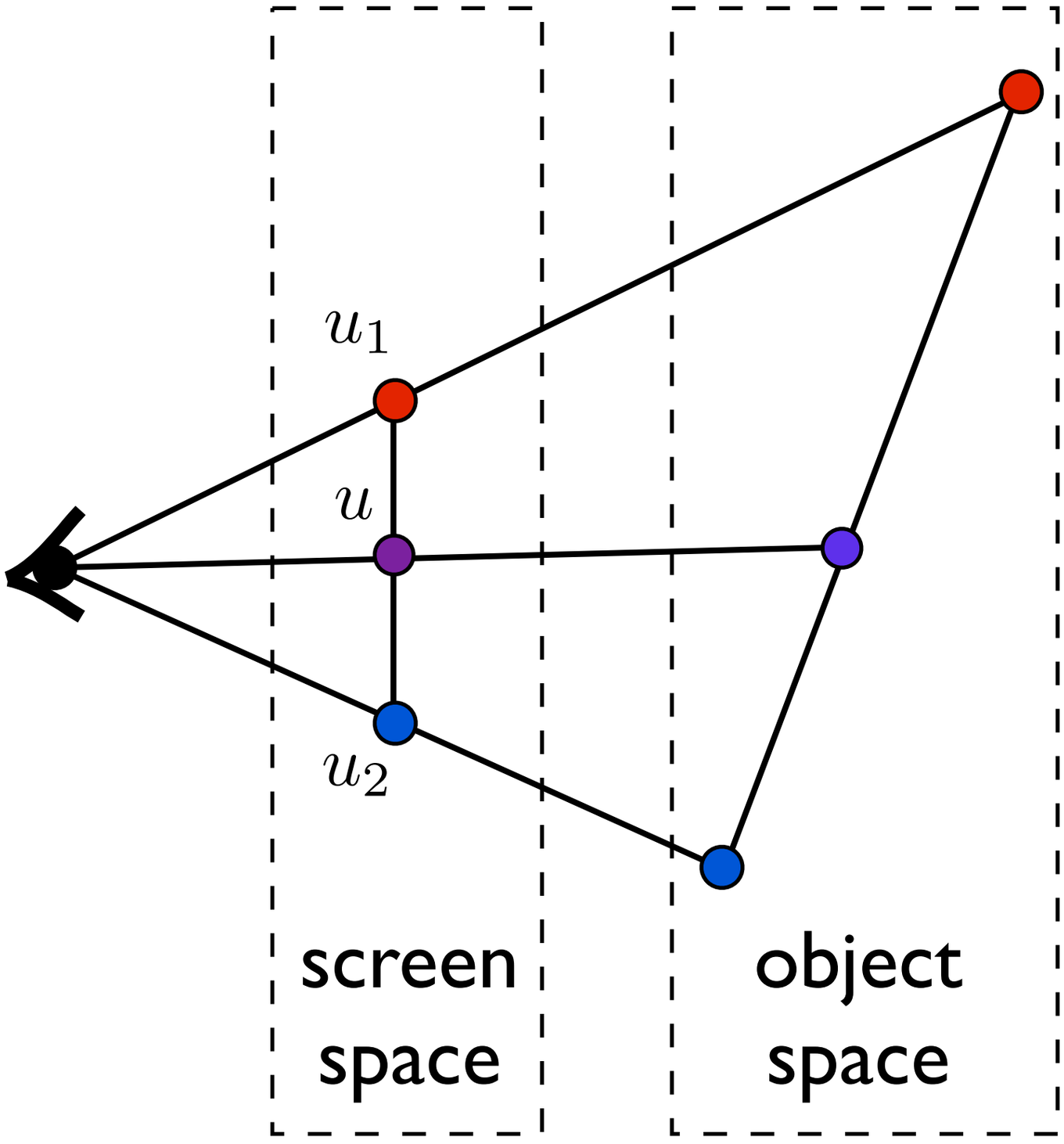
Using screen space weights looks wrong for textures





[Heckbert and Morton, 1990]

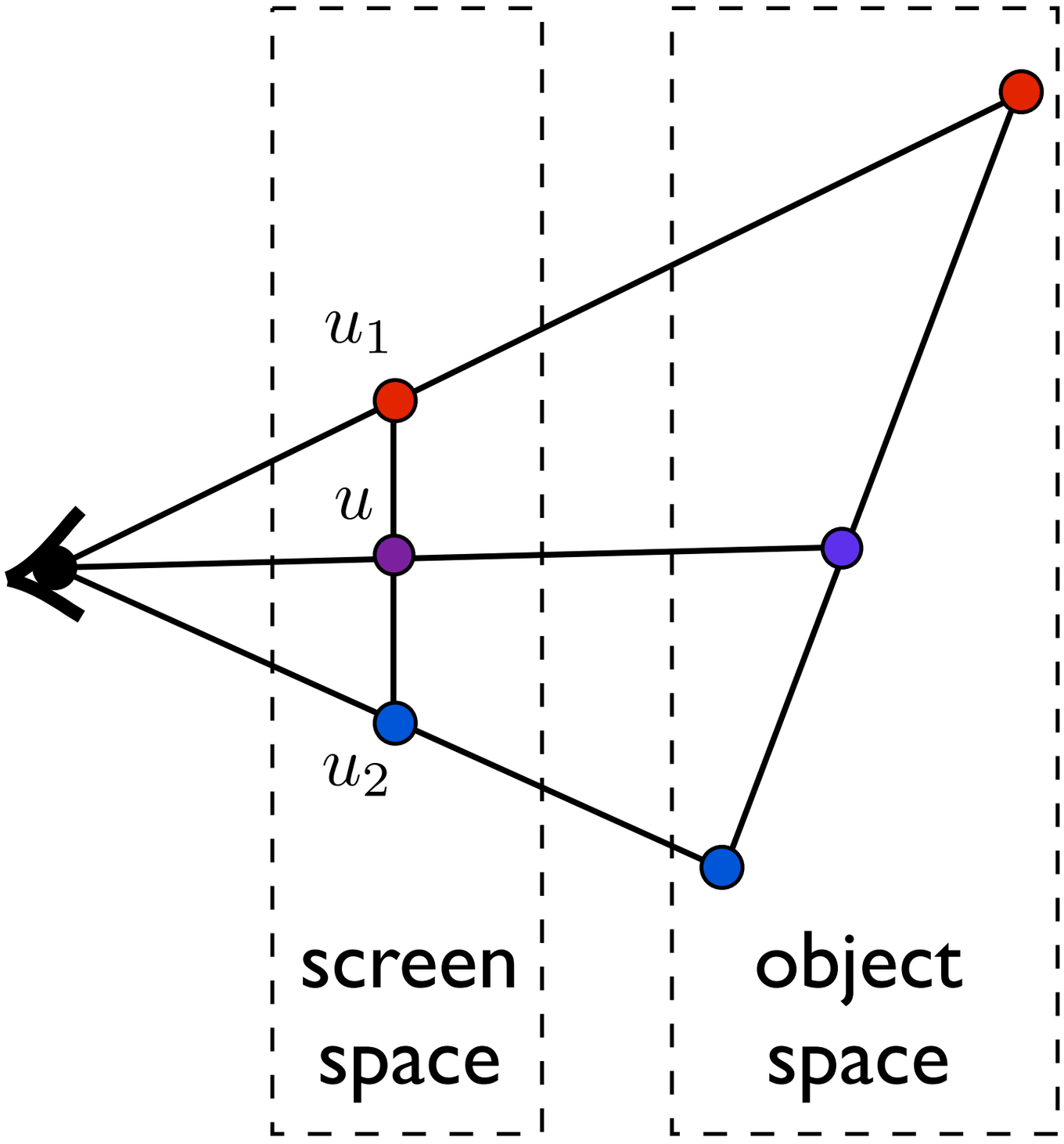
http://en.wikipedia.org/wiki/Texture_mapping#Perspective_correctness



$$u = \frac{1}{2}u_1 + \frac{1}{2}u_2$$

Do we need to transform back to object space?

$$\mathbf{v}_{\mathrm{sc}} = M_{\mathrm{vp}} M_{\mathrm{pers}} M_{\mathrm{cam}} \mathbf{v}$$



$$u = \frac{1}{2}u_1 + \frac{1}{2}u_2$$

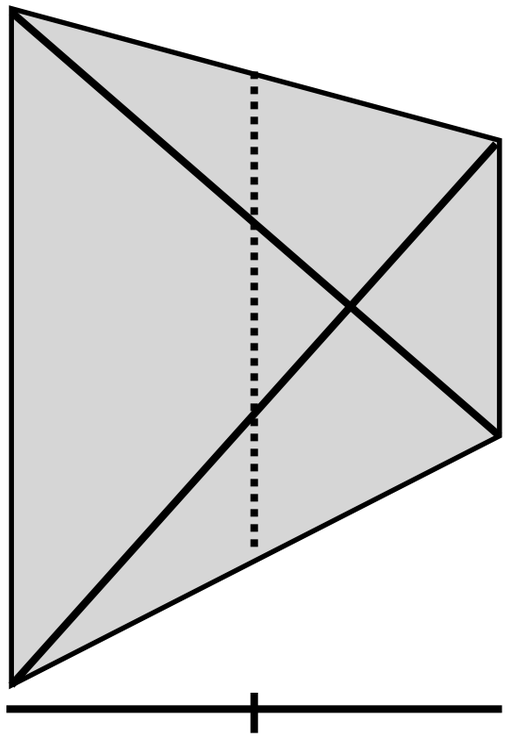
Do we need to transform back to object space?

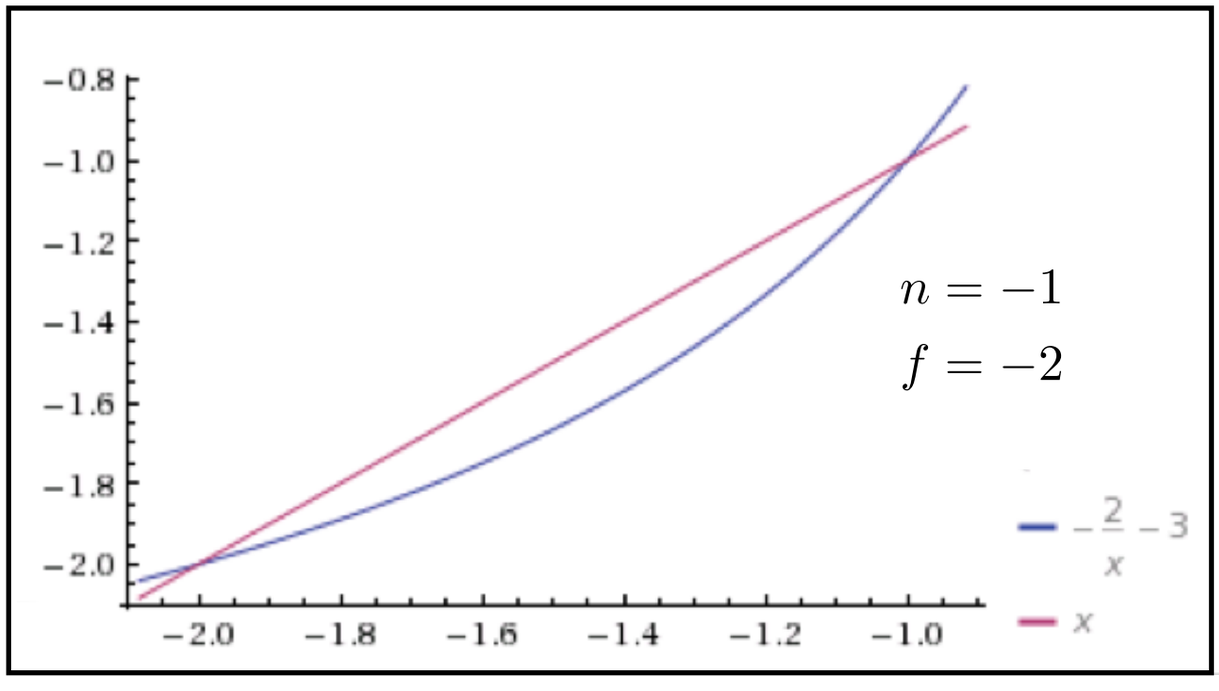
NO!

<whiteboard>

Perspective correct interpolation

- In assignment 1, we found barycentric coordinates in 2D screen space
  - but not the correct object space barycentric coords
  - these coordinates were okay for z-buffer test

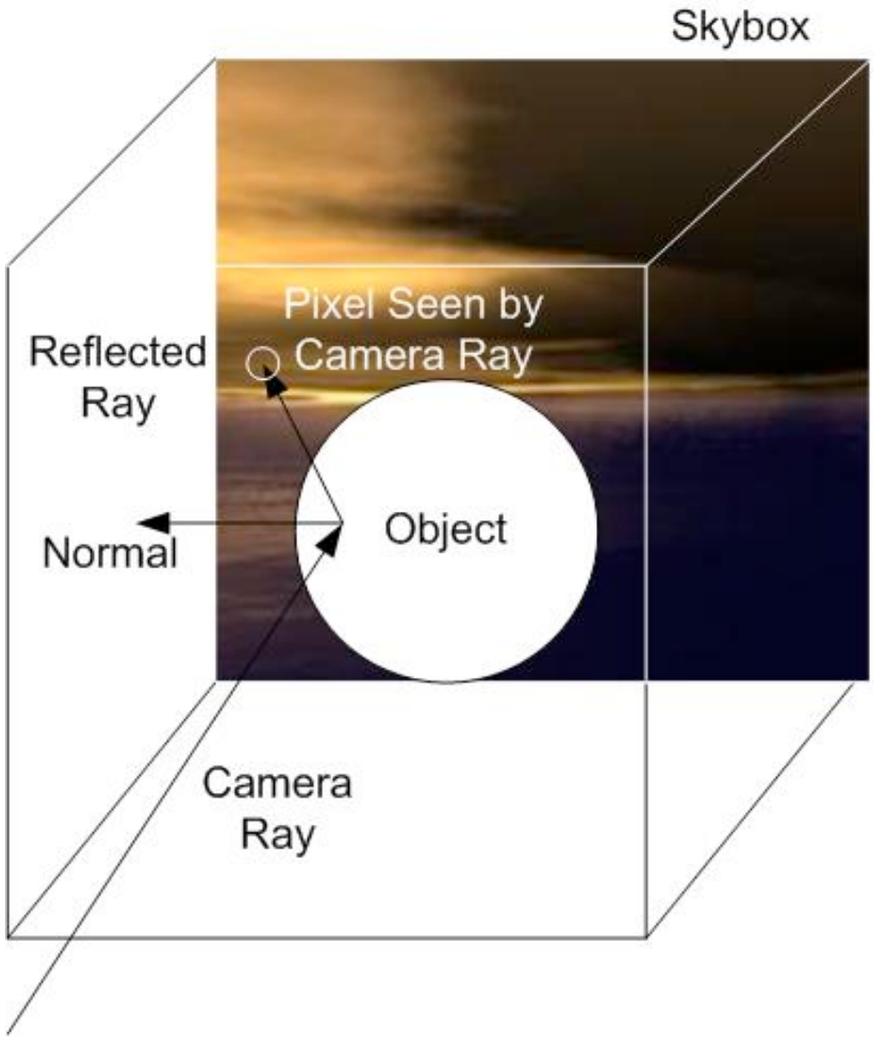


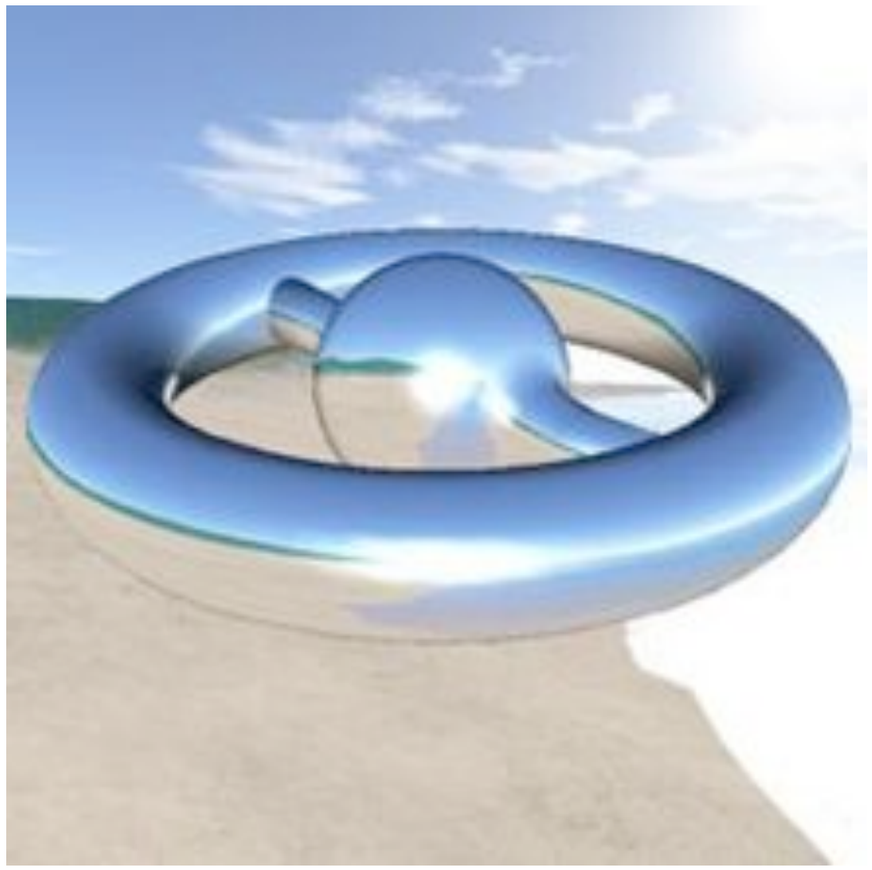


Environment mapping

Environment Mapping

Use a texture for the distant environment simulate the effect of ray tracing more cheaply

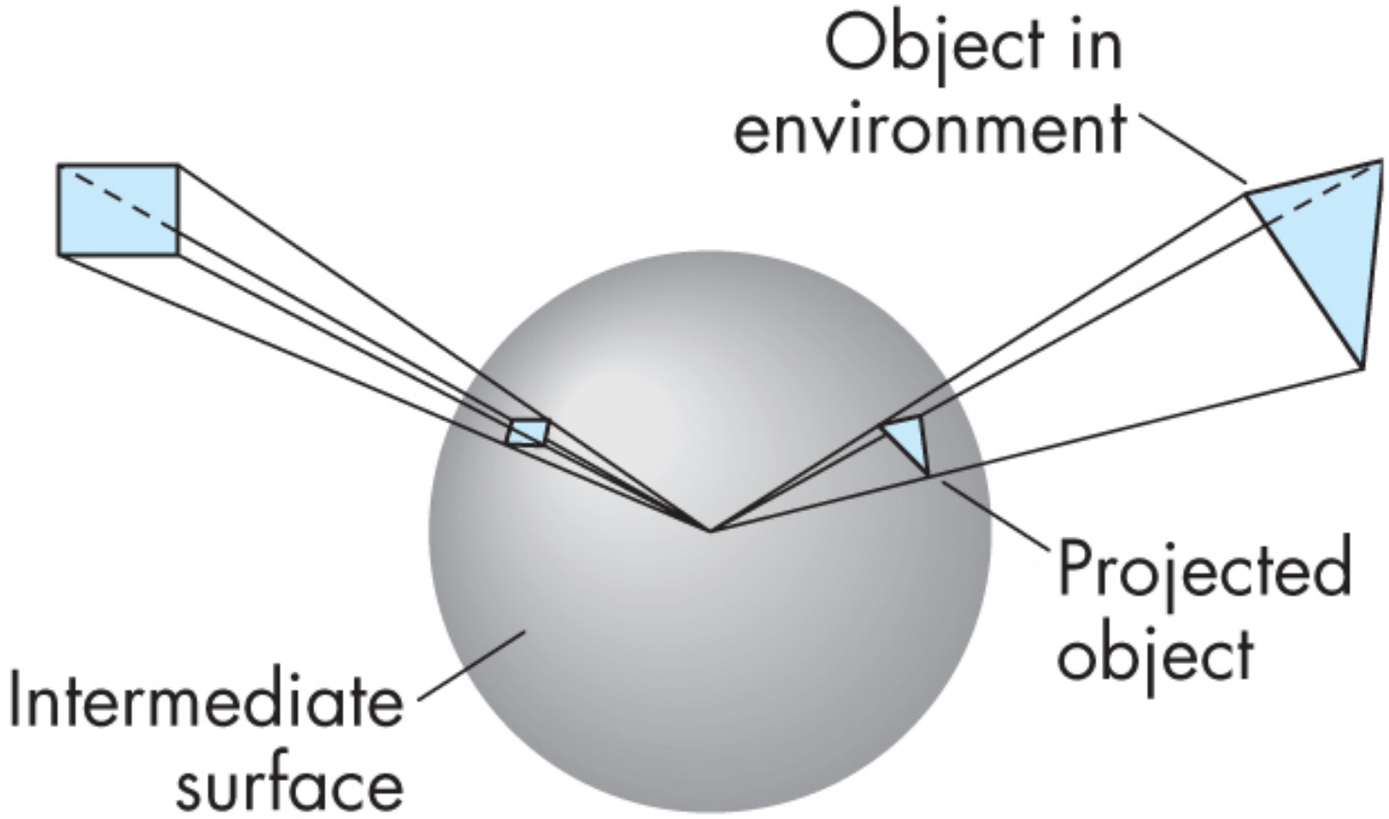




Wikimedia Commons

Sphere Mapping

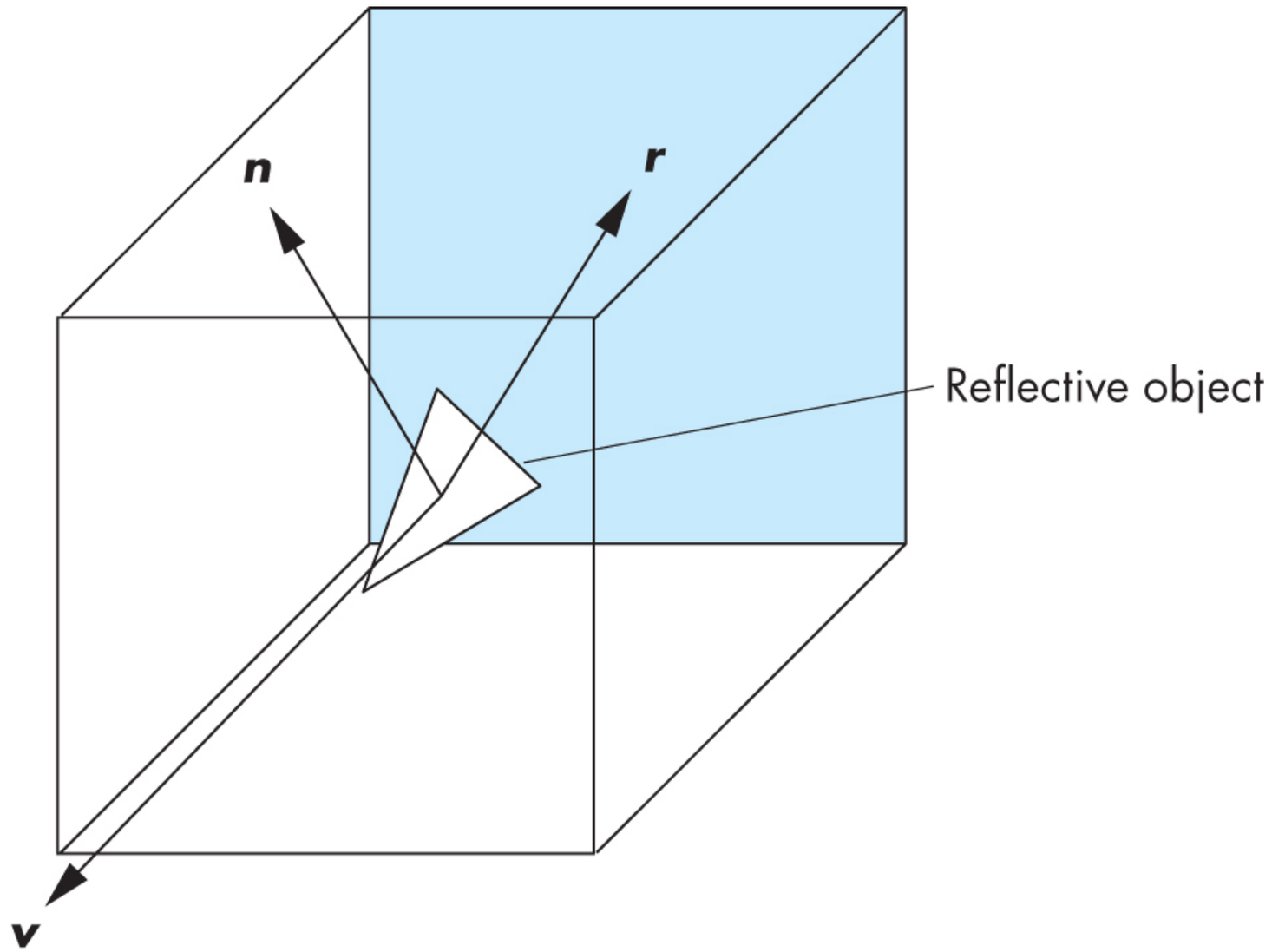
- Project objects in the environment onto sphere centered at eye
- unwrap and store as texture
- use reflection direction to lookup texture value



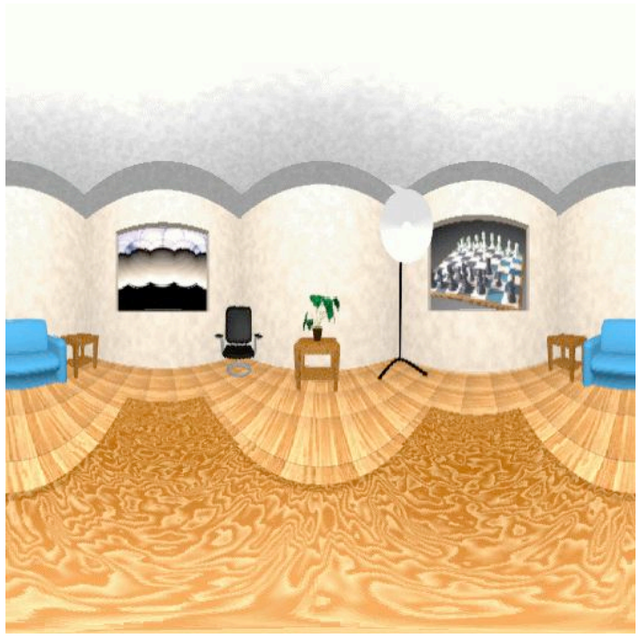
How is environment mapping different from ray tracing?
 - typically only the direction of the reflection vector is used to look up the texture value- this doesn't reproduce the true intersection of the reflected ray the the object it hits
- Note: realism of environment map degrades as model is displaced from where the textures were generated

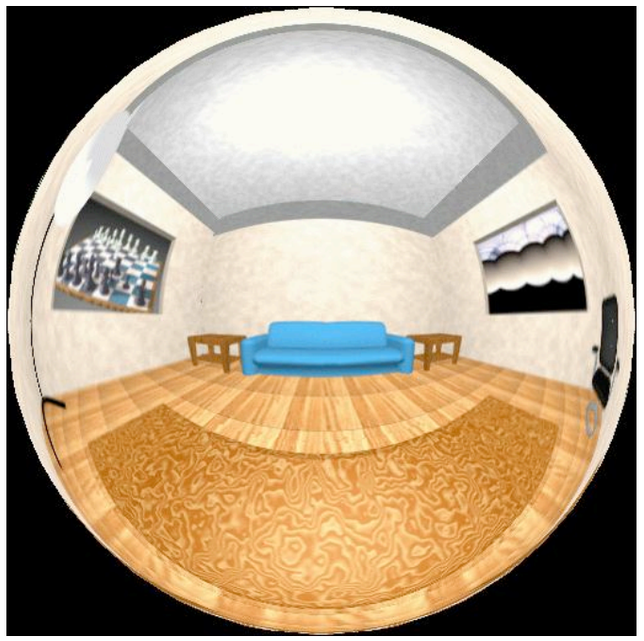
Cube Mapping

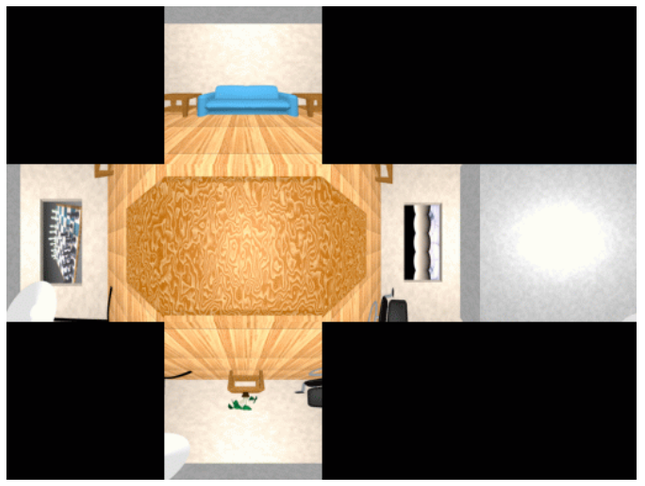
- Compute six projections, one for each wall
- store as texture
- use reflection direction to lookup texture value



Different environment maps







www.reindelsoftware.com

Blinn/Newell latitude mapping

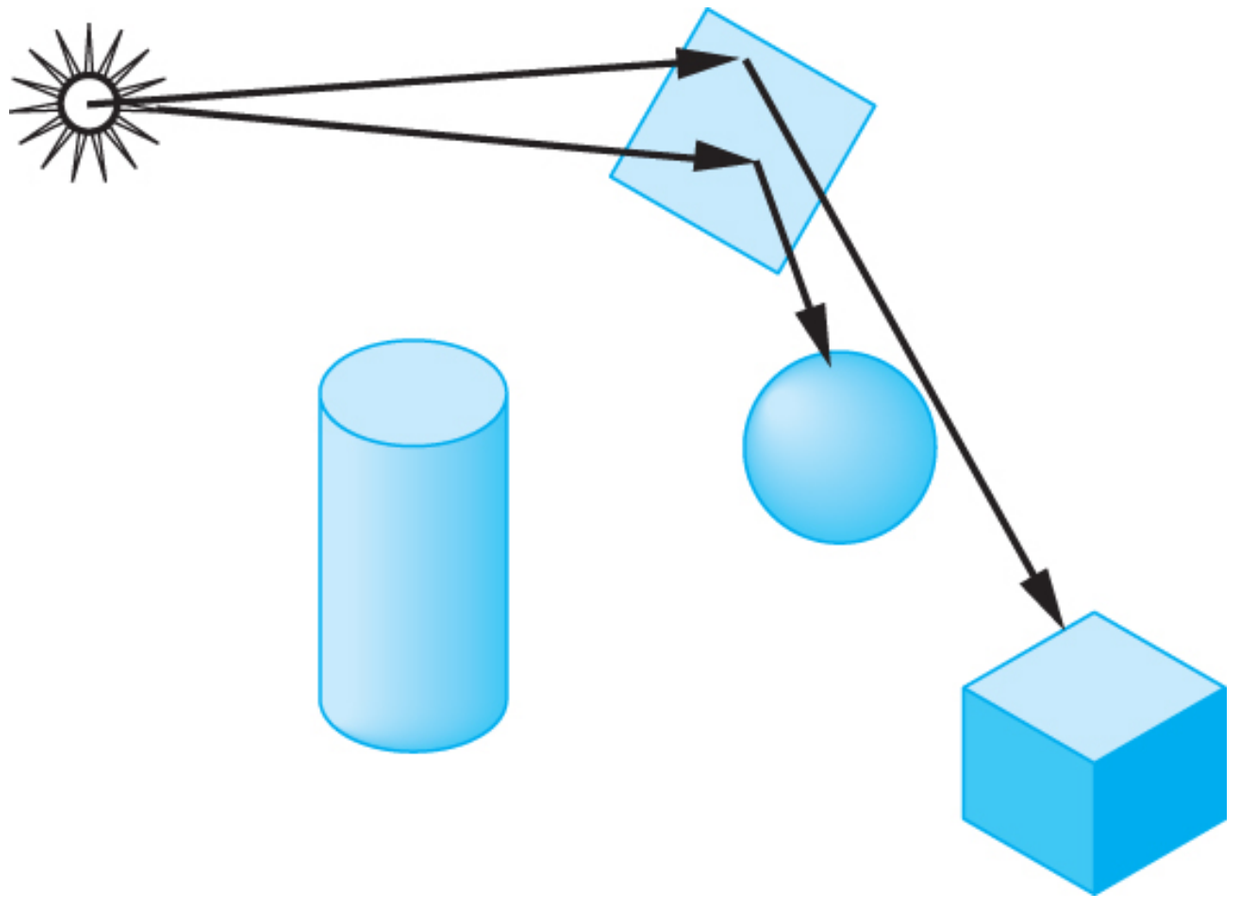
OpenGL spherical mapping

Cube mapping

Environment Mapping

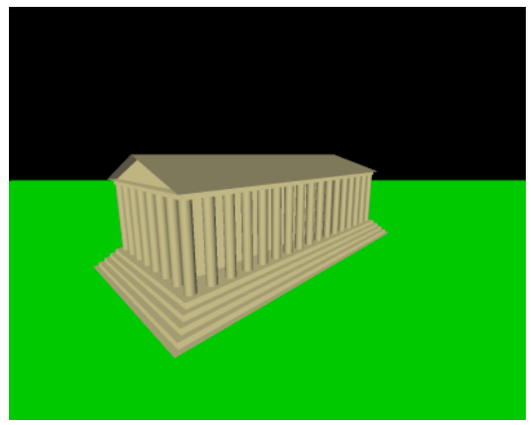
Create the effect of a mirror with two-pass rendering

1. First pass: render the scene from the perspective of the mirror
2. Second pass: render from original pov; use the first image as a texture for the mirror

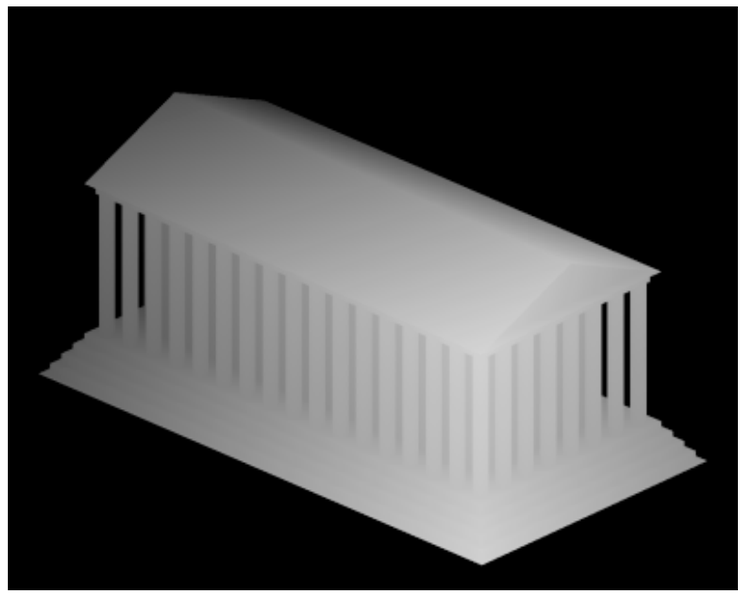


Shadow Mapping

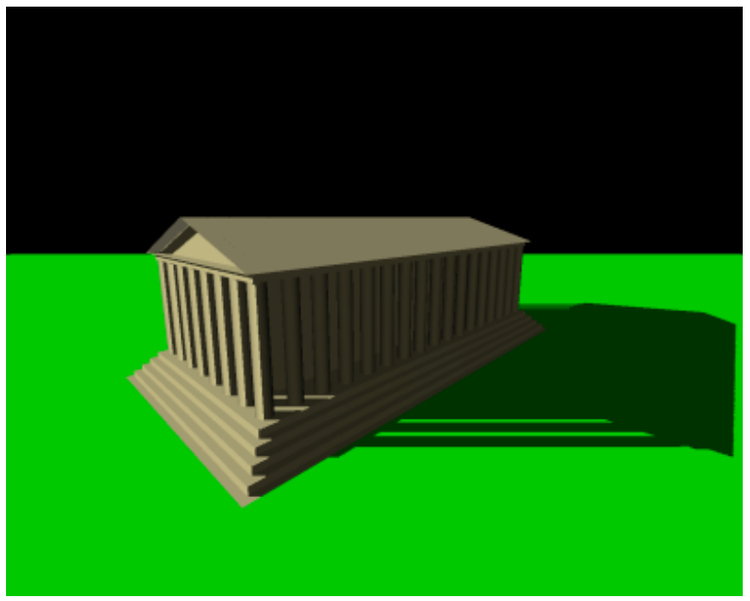
first pass from light's perspective



1. render scene from pov of light and store z-buffer in a texture



2. render scene from desired pov, and test pixel against light's z-buffer

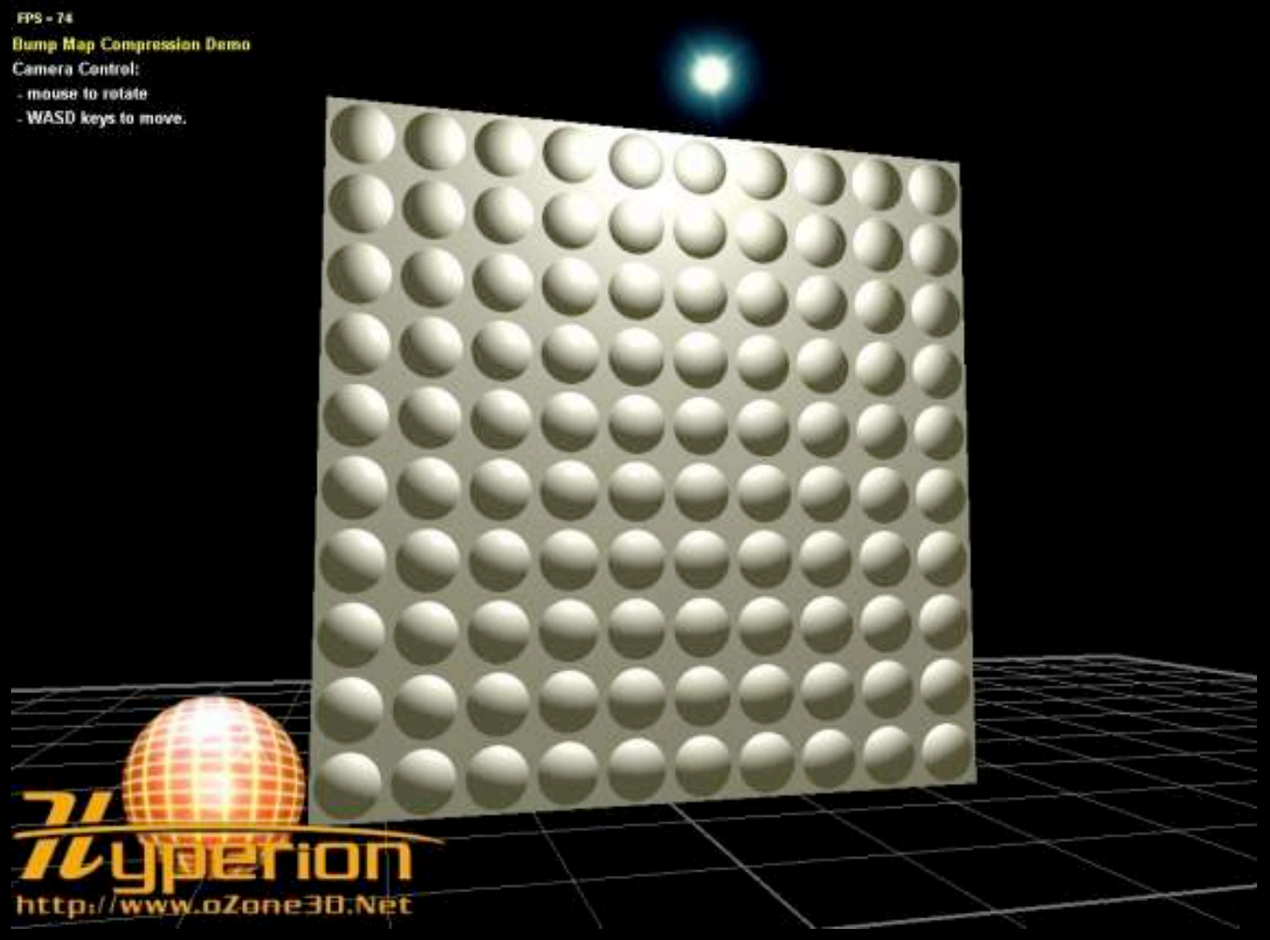


Bump Mapping

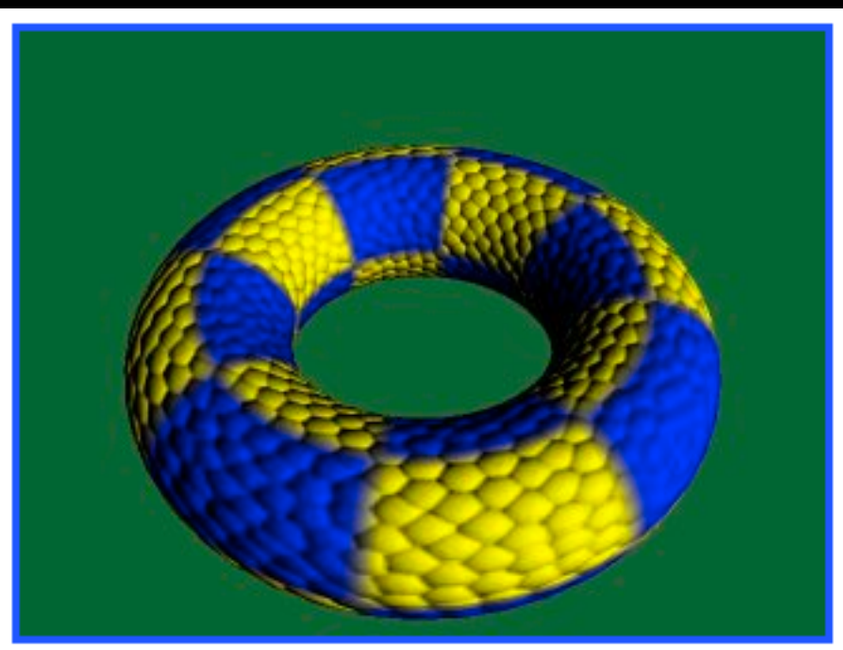
perturb
normal
vectors

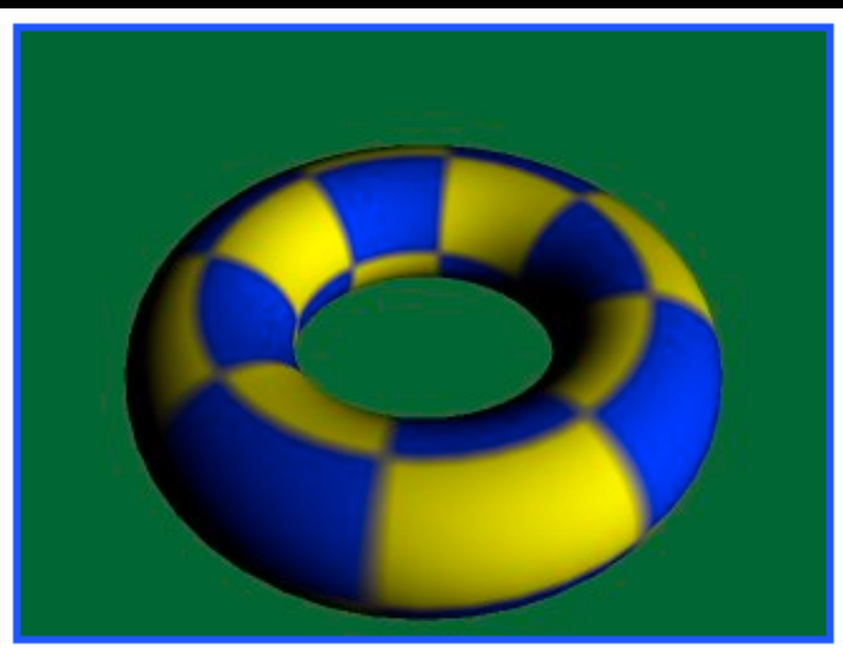
doesn't
affect
silhouette

[DirectXTutorial.com]



http://www.lg.clanhost.cz





http://www.paulsprojects.net/tutorials/simplebump/simplebump.html