Environment Mapping

Environment Mapping and other lies.

- OK, its not possible to do environment mapping in real time unless you happen to be NASA

But there are some funky effects you can do…

- Chrome mapping
- Reflection mapping
- Water
- Specular Highlights

What is Environment Mapping?

- Environment mapping is used to model reflective surfaces
- The object is enclosed by a three dimensional surface onto which the environment is projected. Reflected Rays are projected from the object, hit the surface and then are indexed into the map.
- This isn't a realistic proposition on PS in real time. We don't have time to render all the things we'd like to see, let alone render a whole series of world views

What is Chrome Mapping?

- Instead of projecting the environment onto a 3d surface, we project a texture with a random chrome look to it, this is used to texture the object and from a distance it looks all shiny and reflective.

Advantages of Chrome Mapping

- With environment mapping you see the scene reflected in the objects surface
- With chrome mapping you see something that might look a little how the scene might look if it was reflected in the surface. It is more like a diffuse reflection
- Easy to implement
- Far less computationally expensive than environment mapping

Doing it on Playstation

- Chrome Mapping that is....

Choosing a 3 dimensional surface algorithm

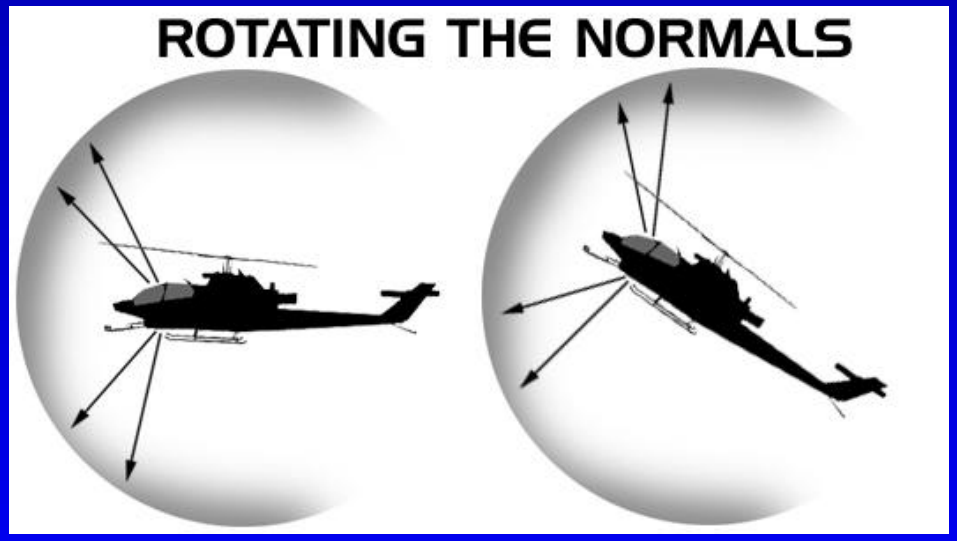
- TrueSpace offers cubic, cylindrical and spherical surrounding surfaces
- Spherical is the simplest to implement
- A sphere only has 1 surface, a cylinder has 3, a cube 6
- For more information on how to project the normals more accurately consult the references presented at the end

How to project rays out from the surface

- Use the normals
  - Use the vertex normals. As the vertex normals are shared by the surrounding polys this ensures the texture matches up between adjacent polygons
  - Normals are of Unit length (4096)

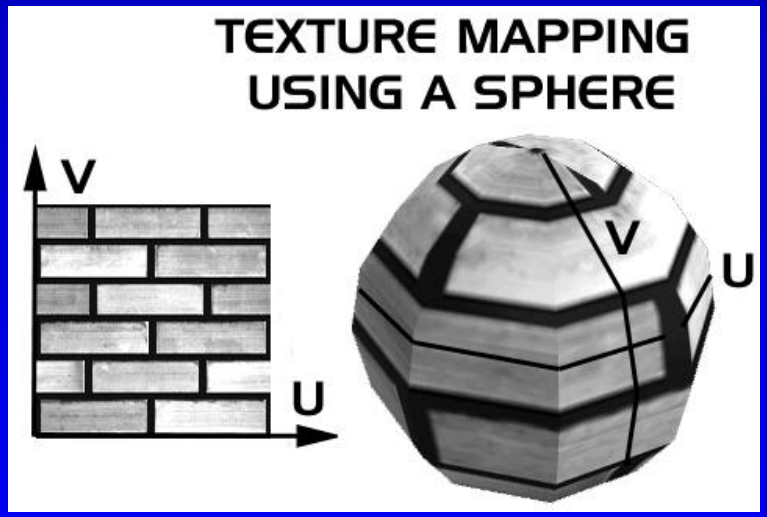
BUT….

- Have to apply the same rotation to the normals as to the vertices to get the right view
- Otherwise the texture will always stay the same



How to create the enclosing sphere

- Use Latitude-Longitude solution
  - Latitude (lines running east west) - Lines of constant v
  - Longitude (lines running north south) - Lines of constant u



So: How to do it (1)

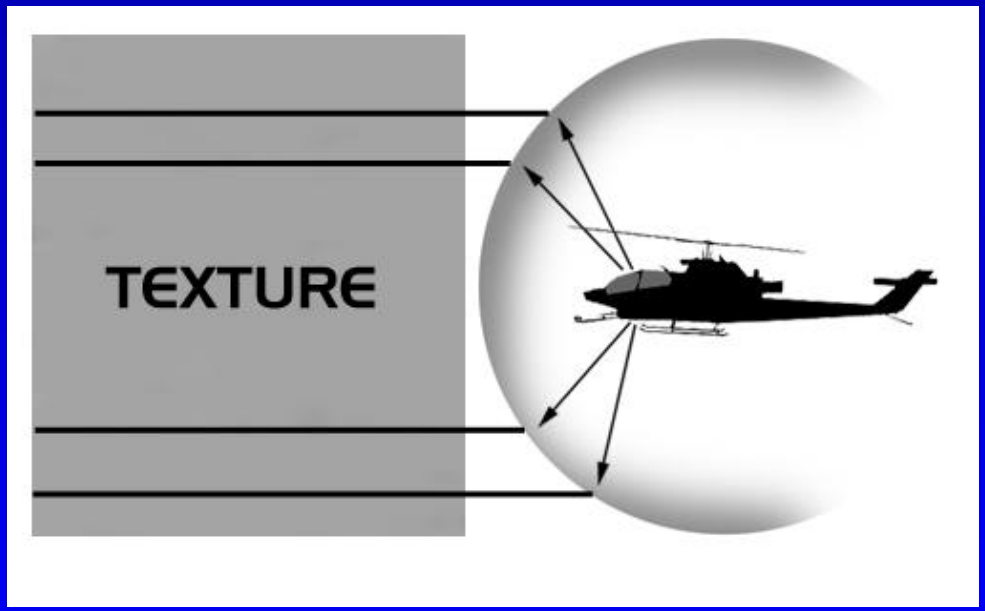
- Assume enclosing sphere has a radius of Unit length and object has a size of 0
- So normal vector components are in the range -4096 to +4096
- Texture coordinates in the range 0..255 (in my example)
- a normal has 3 components(X,Y,Z), a texture map has 2 components (U,V)

So: How to do it (2)

- We need to get 2 textures values e.g. (U,V)
- Take the Y vector and use it to obtain a V coordinate
- Combine the Z and X vector to obtain a U coordinate
- In practice only need to use the X vector to get a result that looks cool

Example

- Take the y component of the normal vector and use it to derive the v coordinate of the texture map



Disadvantages of this technique

- Glitching near the poles.
  - Use abstract soft textures
- Slow. Even this simple system is computationally expensive
  - Restrict its use to special objects, power ups etc.
  - If you use it too much, you'll wear it out...

Other things to consider

- Maybe possible to use a look up table for the u,v coordinates dependent on how free your viewing is...
- An animated texture could also produce a similar result for simple cases...
- Use an animated texture with chrome map
  - Could be used for Fiery reflections?

Reflection Maps

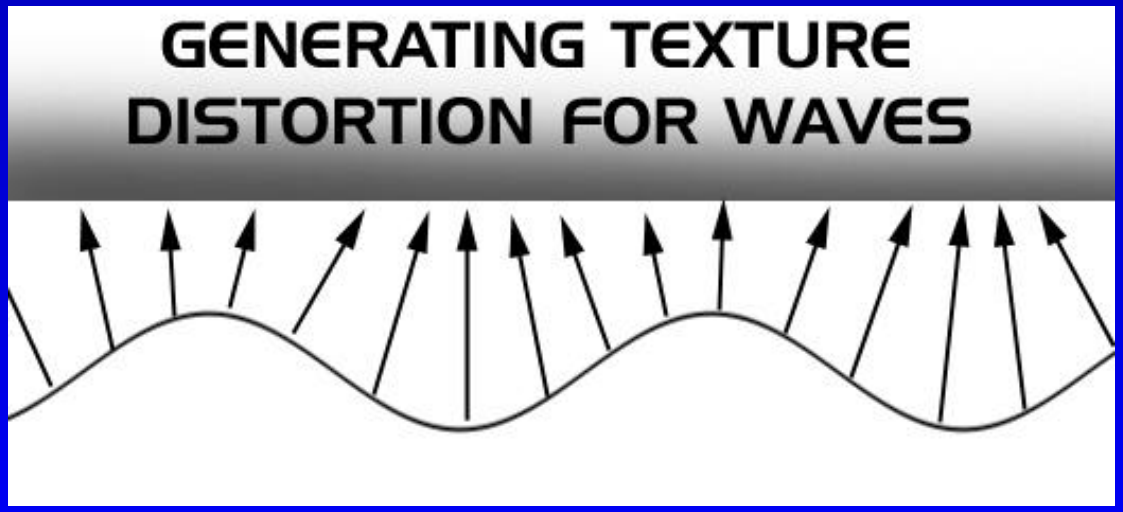
- What if it was possible to combine the texture map with the chrome map…
- Calculate the vertexes for the poly
- Render 1 poly solid with the texture on it
- Render a second poly, which is semi transparent with the chrome map texture on it, directly over the first poly

Refraction Maps

- Modeling the bending of light through different mediums
- Same structure as the chrome mapping but using a different algorithm to generate the texture coordinates
- Cool for water distortion effects

Water

- A water surface can be modeled using a derivation of chrome mapping
- Use the normals to calculate the texture for the reflection



Water (2)

- Combine reflection and refraction maps for the best simulation of water
  - Very expensive
  - Very nice!

Specular Highlights

- So while we're messing around with the normals……
  - Use to normal information to selectively boost the vertex RGB values
  - Use the normal information to locate the position for a highlight sprite
- Use your imagination!!

References

- Advanced Animation & Rendering Techniques
  - Allan and Mark Watt Addison-Wesley
  - ISBN 0-201-54412-1
- 3d Computer Animation
  - John Vince Addison-Wesley
  - ISBN 0-201-62756-6