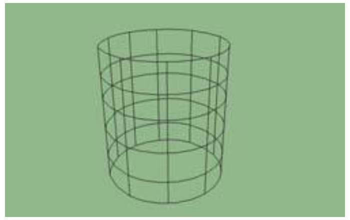
Texture Mapping

Introduction

- So far, every geometric primitive has been drawn as either a solid color or smoothly shaded between the colors at its vertices, that is, they've been drawn without texture mapping.
- Texture mapping allows you to glue an image to a polygon.
- Texture mapping ensures that all the right things happen as the polygon is transformed and rendered.



Introduction

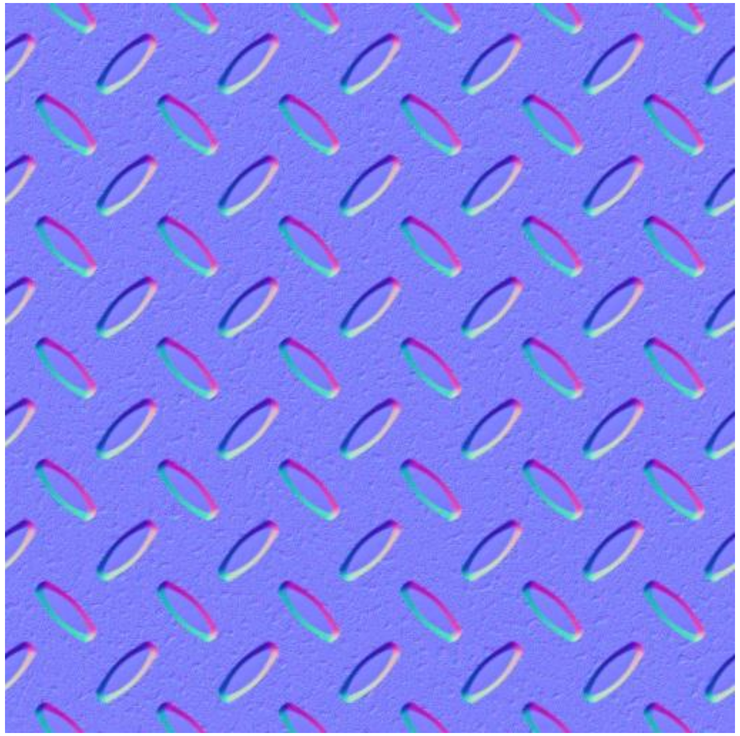
- Other uses for texture mapping include depicting vegetation on large polygons representing the ground in flight simulation; wallpaper patterns; and textures that make polygons look like natural substances such as marble, wood, and cloth.

Introduction

- Texture mapping is a fairly large, complex subject, and you must make several programming choices when using it.
- Most people intuitively understand a two-dimensional texture, but a texture may be **one-dimensional** or even **three-dimensional**.
- You can map textures to surfaces made of a set of polygons or to curved surfaces, and you can repeat a texture in one, two, or three directions (depending on how many dimensions the texture is described in) to cover the surface.

Introduction

- You can automatically map a texture onto an object in such a way that the texture indicates contours or other properties of the item being viewed.



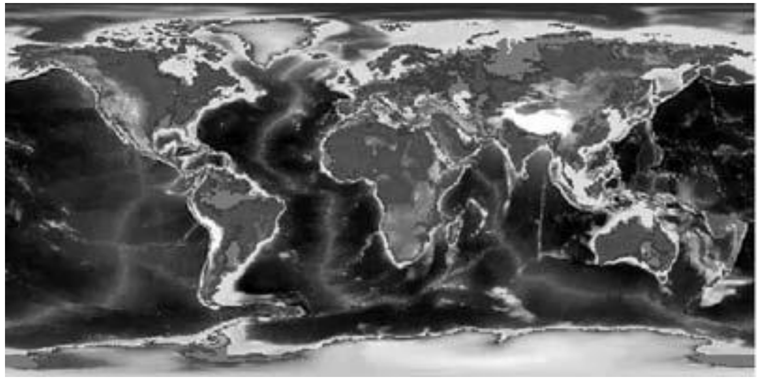
2D Texture Mapping

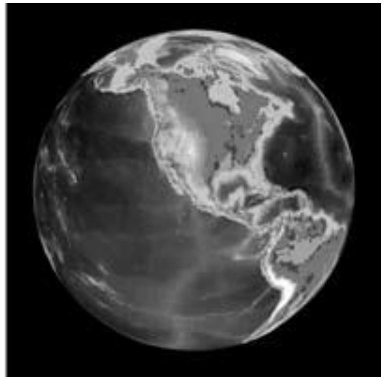
- For 2D texture mapping, we use a 2D coordinate, often called **uv**, which is used to create a reflectance R(u, v).
- The key is to take an image and associate a (u, v) coordinate system on it so that it can, in turn, be associated with points on a 3D surface.

2D Texture Mapping

Example:

- If the latitudes and longitudes on the world map are associated with a polar coordinate system on the sphere, we get a globe





2D Texture Mapping

- As a convention, the coordinate system on the image is set to be the unit square $(u, v) \in [0, 1]^2$.
- For (u, v) outside of this square, only the fractional parts of the coordinates are used resulting in a tiling of the plane.
- Note that the image has a different number of pixels horizontally and vertically, so the image pixels have a non-uniform aspect ratio in (u, v) space.

2D Texture Mapping

- To map this $(u, v) \in [0, 1]^2$ image onto a sphere, we first compute the polar coordinates.
- Recall the spherical coordinate system described by Equations

$$x = r \cos \varphi \sin \vartheta,$$
$$y = r \sin \varphi \sin \vartheta,$$
$$z = r \cos \vartheta.$$

2D Texture Mapping

- For a sphere of radius R with center (cx, cy, cz), the parametric equation of the sphere is
- $x = xc + R\cos\phi \sin\theta$,
- $y = yc + R\sin\phi \sin\theta$,
- $z = zc + R\cos\theta$.

2D Texture Mapping

- We can find (θ, ϕ):
- $\theta = arcos(\frac{z - z_c}{R})$
- $\phi = \text{arctan2}(y - y_c, x - x_c)$
- where arctan2(a, b) is the atan2 of most math libraries which returns the arctangent of a/b.

2D Texture Mapping

- Because $(\theta, \phi) \in [0, \pi] \times [-\pi, \pi]$, we convert to (u, v) as
- follows, after first adding 2π to ϕ if it is negative:
- $u = \dfrac{\phi}{2\pi}$,
- $v = \dfrac{\pi - \theta}{\pi}$