Texture mapping review

1. Why use textures?

   Answer:
   Create effects that are much more complex and realistic without adding orders of magnitude to the geometry (number of polygons)

2. Describe the steps for applying a 2D texture to a surface.

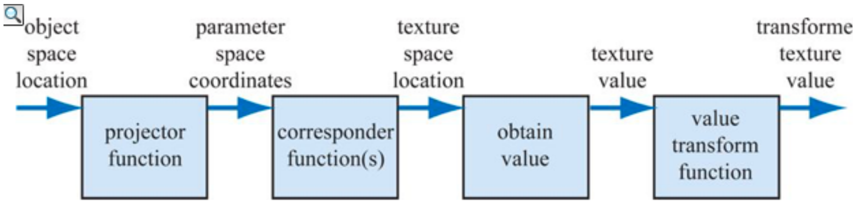
   Answer:

   (a) Project the object space coordinates into parameter space (u, v)

   (b) Use the (u, v) parameter coordinates and some corresponder function to get a texture space location

   (c) Get a texture value from that location
   (optional: transform the texture value)

   (d) Apply that value to the corresponding location on the surface.



Projector functions

3. Describe the steps for applying a 2D texture to a 3D object using an intermediate object.

   Answer:

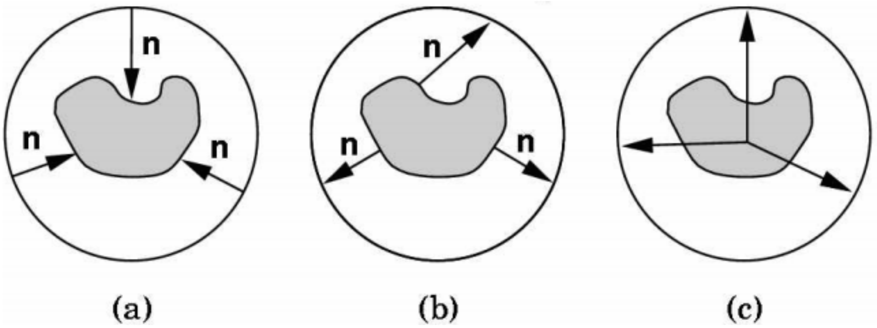
   (a) Map the texture to a 3D intermediate object for which a mapping is defined, such as a cube, cylinder, or sphere

   (b) Map the intermediate object to the target object by

      i. following the normal from each point on the intermediate object to a point on the surface

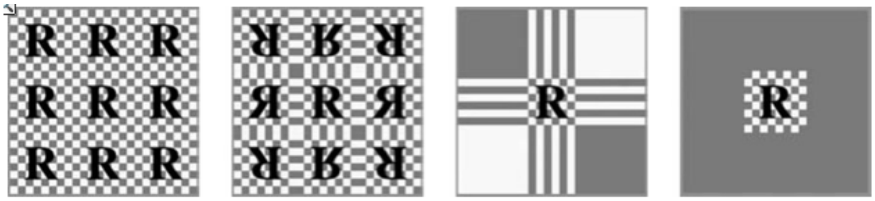
      ii. following the normal from each point on the surface to a point on the intermediate object

      iii. following a ray from the center of the target object through a point on the surface to a point on the intermediate object



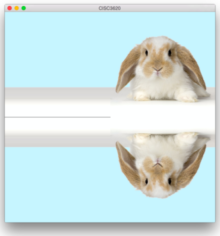
Corresponder functions

4. What are various ways of applying an image to a surface (handling texture coordinates outside of (0, 1))?

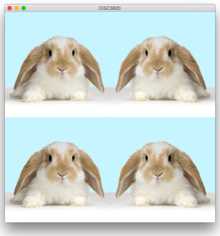
   Answer: 

   - **repeat, tile**: image repeats itself across the surface (drop the integer part of the parameter value)
   - **mirror**: image repeats itself across the surface and is mirrored (flipped) on every other repetition. Provides some continuity along the edges of the texture
   - **clamp**: values outside the range [0, 1) are clamped to this range.
   - **border**: parameter values outside [0, 1) are assigned a separately defined border color. Good for rendering decals onto surfaces

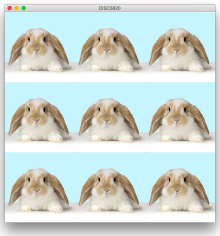
5. Match each image to the corresponding pair of OpenGL texture parameter commands.



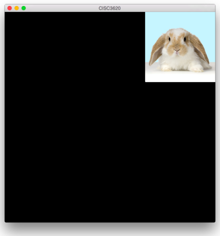
(a)



(b)



(c)



(d)

   1) glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_CLAMP_TO_BORDER);
      glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_CLAMP_TO_BORDER);
   2) glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_REPEAT);
      glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_REPEAT);
   3) glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_CLAMP_TO_EDGE);
      glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_MIRRORED_REPEAT);
   4) glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_MIRRORED_REPEAT);
      glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_REPEAT);

Magnification and minification

6. What is *magnification*?

   Answer: When a texture is mapped to an image of higher resolution, so each texel is mapped to numerous pixels.

7. What is *minification*?

   Answer: When a texture is mapped to an image of lower resolution, so several texels are mapped to a single pixel.

8. Given a texture of size (114, 114) and an image of size (56, 56), how many texels must cover each pixel?

   Answer: 2.04

Is this a problem of magnification or minification?

Answer: minification (texture is larger than image and must be minified)

9. What is mipmapping?

   Answer: Pre-generating downsampled versions of a texture at powers of two for fast lookup at runtime.

10. Given (u, v) coordinates of (0.90, 0.51) and a texture of size (1097, 1145), what texel will be chosen by nearest neighbor sampling?

    Answer: (987, 584)

$$0.90 \times 1097 = 987.3 \Rightarrow 987$$

$$0.51 \times 1145 = 583.95 \Rightarrow 584$$

11. Given (u, v) coordinates of (0.01, 0.46) and a texture with resolution 2048 x 1024, where the value at each (s, t) texture location is (s+t+3)/1, what is the value retrieved by bilinear interpolation?

    Answer: 494.52 $0.01 \times 2048 = 20.48, \;\; 0.46 \times 1024 = 471.04.$

    Value can come from pixel (20, 471), (21, 471), (20, 472), or (21, 472).

    Contribution of pixel (20, 471) should be (1-0.48)(1-0.04) = (0.52)(0.96) = 0.499.

    Contribution of pixel (21, 471) should be (0.48)(1-0.04) = 0.461.

    Contribution of pixel (20, 472) should be (0.52)(0.04) = 0.021.

    Contribution of pixel (21, 472) should be (0.48)(0.04) = 0.019.

$$\begin{aligned} 0.499 \times (20 + 471 + 3)/1 &= 246.51 \\ 0.461 \times (21 + 471 + 3)/1 &= 228.20 \\ 0.021 \times (20 + 472 + 3)/1 &= 10.395 \\ 0.019 \times (21 + 472 + 3)/1 &= 9.42 \\ 246.51 + 228.20 + 10.395 + 9.42 &= 494.525 \end{aligned}$$

12. Suppose a texture is applied to an area of size 85 x 85. What two levels of detail (powers of two) should be used for trilinear interpolation?

    Answer: 64, 128 (these are the powers of two adjacent to 85)

    How should each one be weighted?

    Answer: 0.67, 0.33

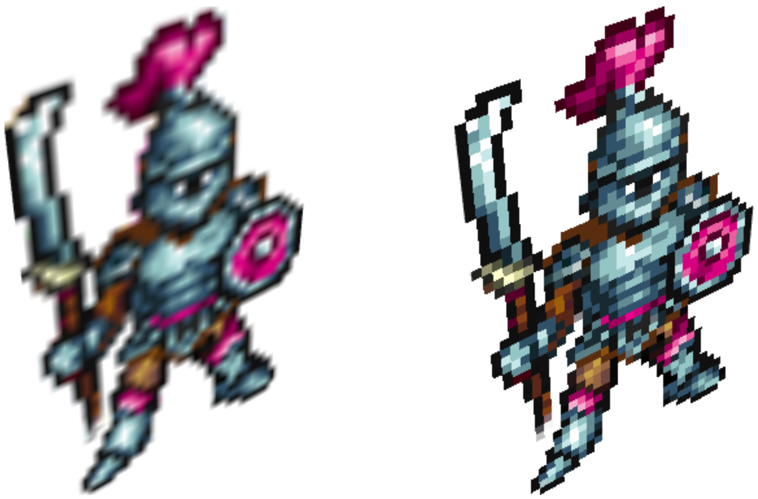
    85 is approximate 1/3 along the distance from 64 to 128.

$$\frac{85 - 64}{128 - 64} \approx 0.33$$

weight of 64 is 1-0.33 = 0.67

weight of 128 is 0.33

13. Which of these figures is likely to have been rendered by nearest neighbor sampling?



Advanced texturing techniques

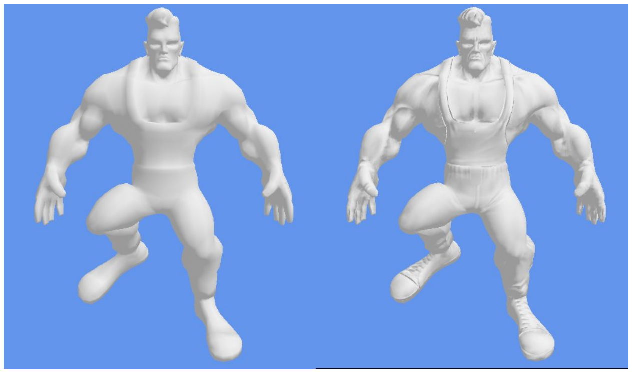
14. Decaling an image onto a surface, or rendering an object with a complex silhouette, can be accomplished by what kind of mapping?

    Answer: alpha mapping: storing alpha values from 0 (transparent) to 1 (opaque)

15. How do bump maps work?

    Answer: Normals are stored in a texture map and retrieved during lighting calculation to be used instead of the surface normal. The light will then interact with the surface as if it were bumpy.

16. What technique can add the detail in the figure on the right without adding to the geometry?

    Answer: Bump mapping

17. What complexity does bump mapping add to an image that a diffuse color map does not?

    Answer: Interactions with the light

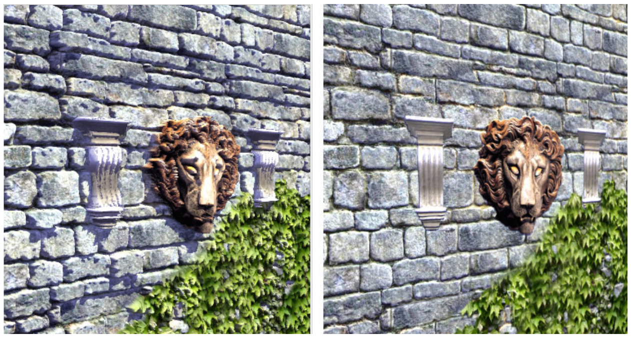
18. What complexity does relief mapping add to an image that bump mapping does not?

    Answer: Self-shadowing, occlusion

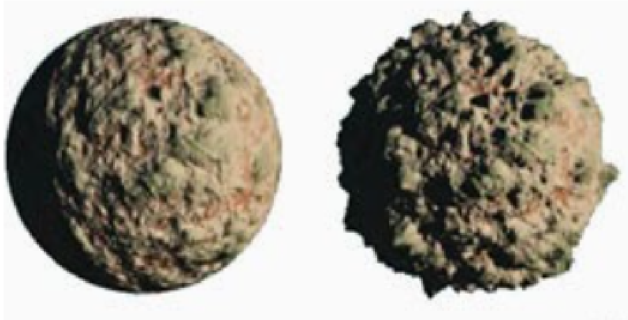
19. What complexity does displacement mapping add to an image that relief mapping does not?

    Answer: Textured silhouette

20. Which of the following two walls is likely to have been generated using relief mapping rather than bump mapping?



21. Which of the following two balls is likely to have been generated using displacement mapping rather than simple bump mapping?



22. From what viewpoint is an environment map initially rendered?

    Answer: The mirror

23. From what viewpoint is a shadow map initially rendered?

    Answer: The light

24. Describe the process of environment cube mapping

    Answer:

    (a) First pass: Render the scene 6 times from the center of the scene: top, bottom, left, right, front, back

    (b) Store the rendered image as a cube texture map. This can then be used for any scenes rendered in this environment.

    (c) During scene rendering, cast a ray in the direction of a perfect reflector from the viewpoint

    (d) Intersect the ray with the cube map at point E

    (e) Illuminate the fragment as if point E were a light source with that position and color

25. Describe the process of shadow mapping.

    Answer:

    (a) First pass: render the scene from the viewpoint of the light.

    (b) Store z (depth) values of the fragments closest to the light

    (c) When rasterizing scene with real view, also get screen coordinates + depth from light view

    (d) If depth is the same as in the shadow depth map at those screen coordinates, that is the visible fragment and should be lit

    (e) Otherwise shadow