Shadow Mapping

Why Shadows?

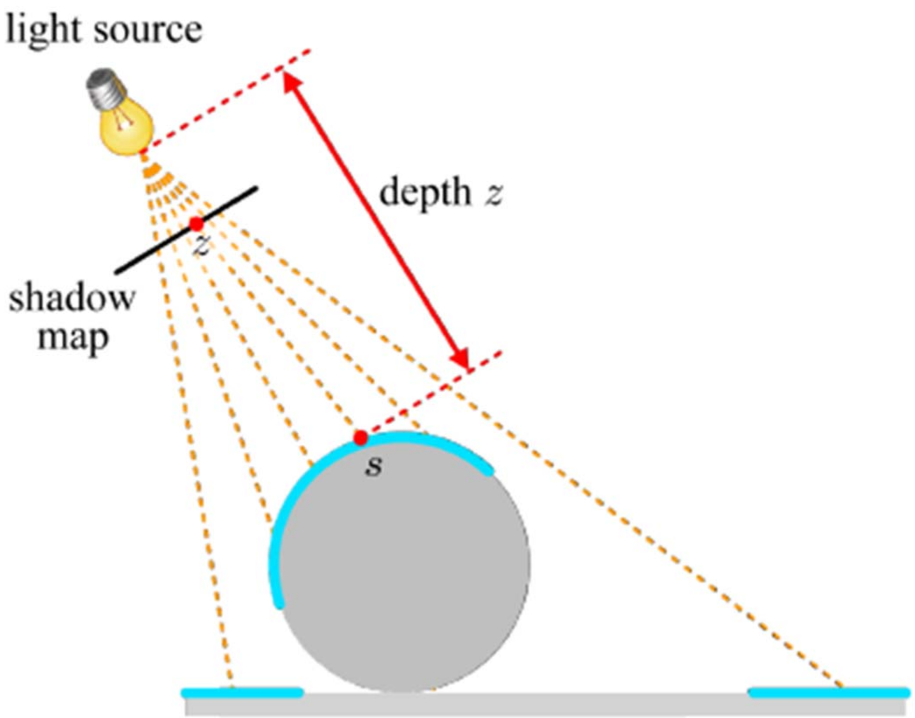
- Shadows help us understand the spatial relationships among objects in a scene.

(a)

(b)

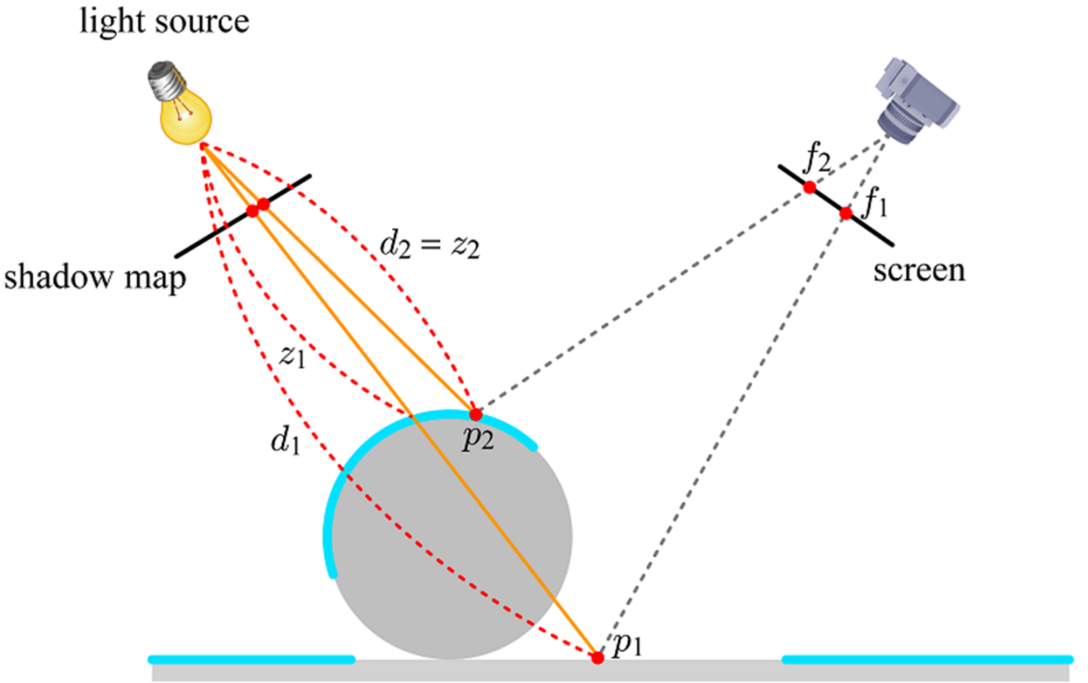
Shadow Mapping

- Two-pass algorithm
- Pass 1 – not real rendering
  - Render the scene *from the position of the light source.*
  - Store only the depths into the *shadow map*, which is a *depth map* with respect to the light source.



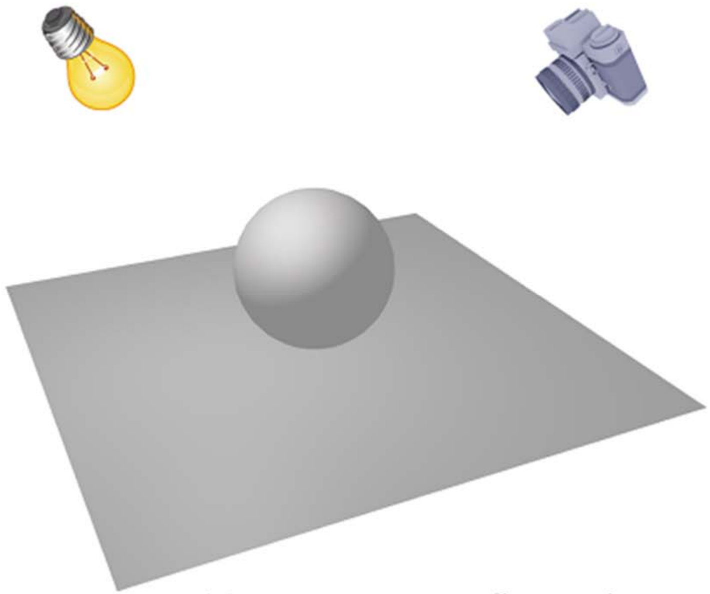
Shadow Mapping (cont'd)

- Pass 2
  - Render the scene from the camera position. It's real rendering.
  - For each pixel, compare its distance *d* to the light source with the depth value *z* in the shadow map.
    - If $d > z$, the pixel is in shadows.
    - Otherwise, the pixel is lit.

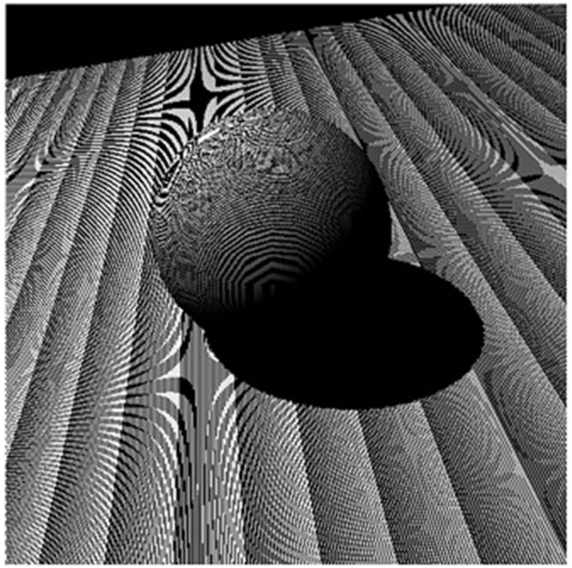


Shadow Mapping – Surface Acne

- A brute-force implementation of shadow mapping suffers from two major problems.
  - Surface acne - a mixture of shadowed and lit areas
  - Shadow-map filtering



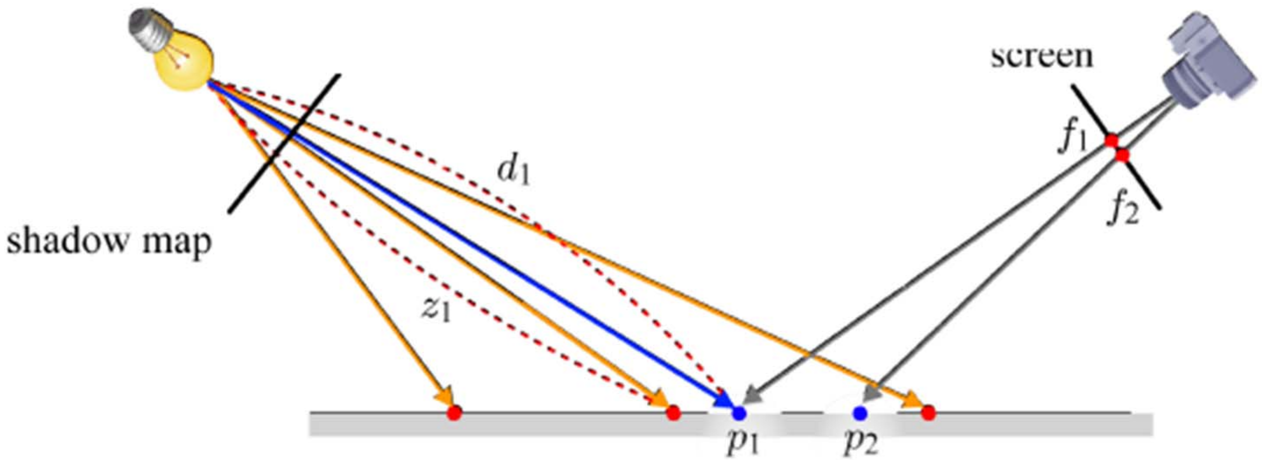
(a) A scene configuration



(b) Surface acne artefact

Shadow Mapping – Surface Acne (cont'd)

- What's the problem? The shadowed and lit pixels coexist on a surface area that should be entirely lit.
- Note that the scene points sampled at the second pass are usually different from the scene points sampled at the first pass.
- Suppose the nearest point sampling.

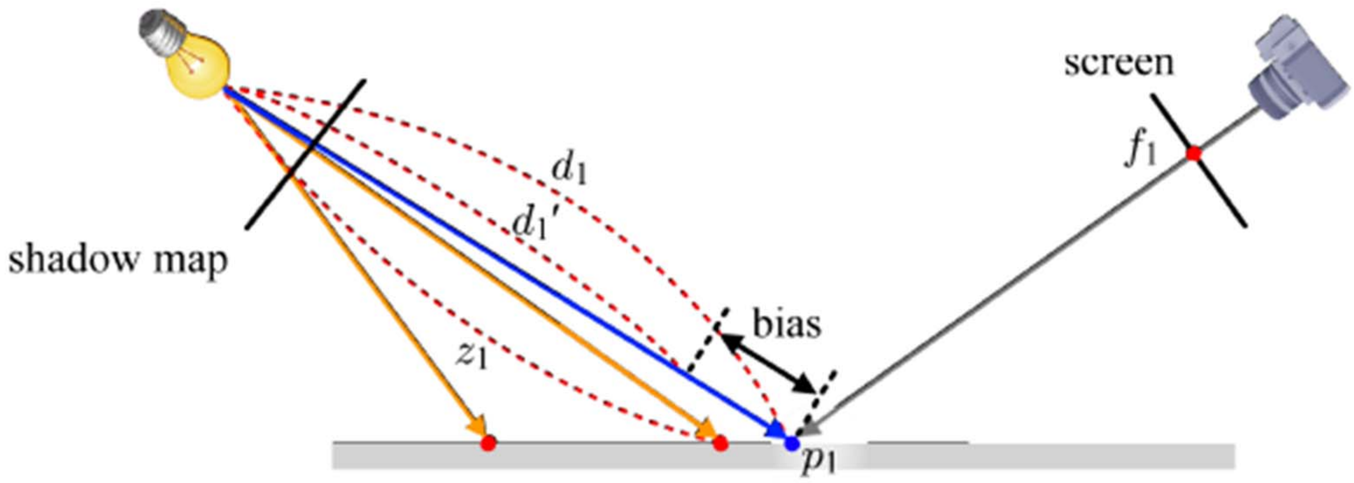


(c) Surface point p_1 is shadowed whereas p_2 is lit.

- In the example, p_1 is to be lit, but judged to be in shadows because $d_1 > z_1$.

Shadow Mapping – Surface Acne (cont'd)

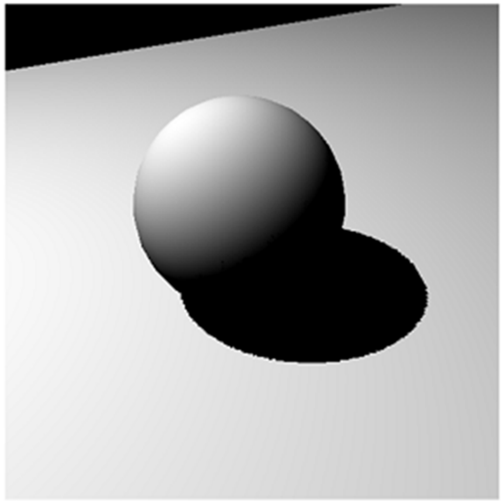
- At the 2nd pass, subtract a small bias value from d_1 such that $d_1' < z_1$.



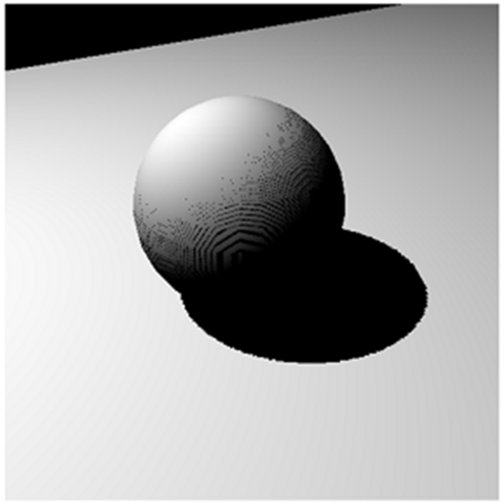
(d) Bias is subtracted from the distance to the light source

Shadow Mapping – Surface Acne (cont'd)

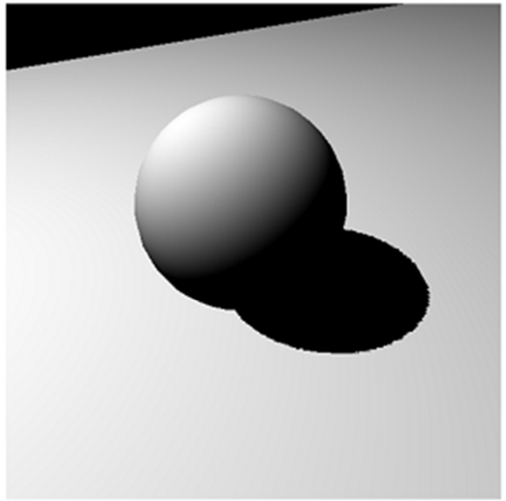
- The bias value is usually fixed after a few trials.



(e) Biased shadow mapping



(f) Too small bias

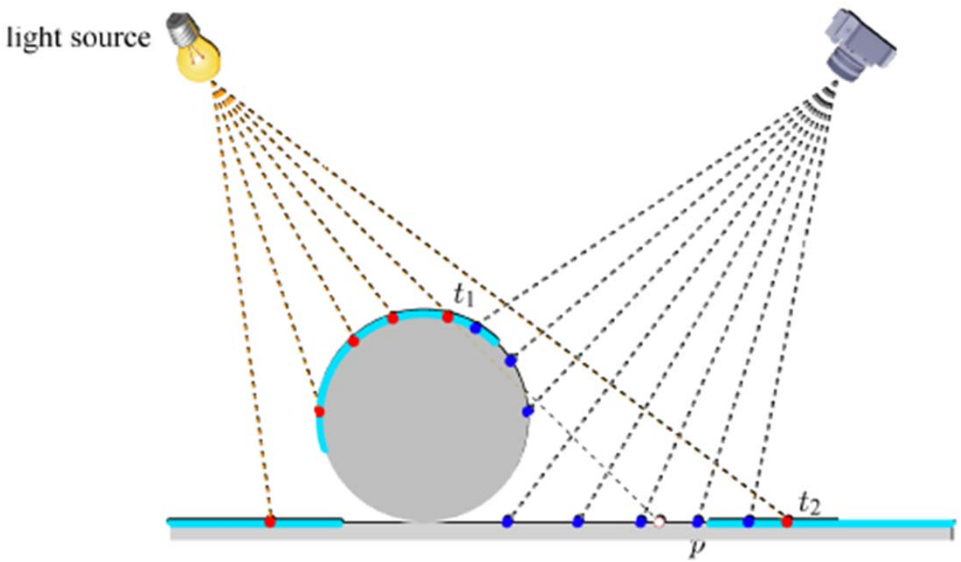


(g) Too large bias

- Anyway, the surface acne problem has been largely resolved.

Shadow Mapping – Shadow Map Filtering

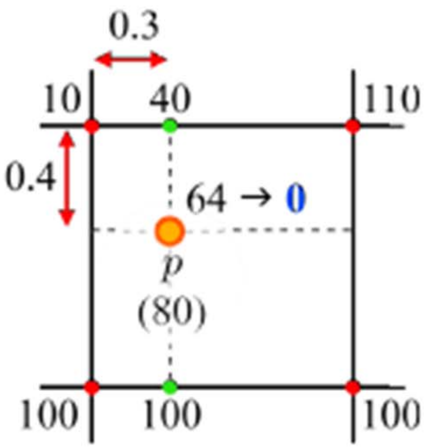
- If the resolution of a shadow map is not high enough, multiple pixels may be mapped to a single texel of the shadow map.
- This is a magnification case. Suppose you choose the nearest point sampling.



- Consider pixel p. Nearest point sampling makes texel t_1 compared with p whereas bilinear interpolation makes t_1 and t_2 interpolated and then compared with p.

Shadow Mapping – Shadow Map Filtering (cont'd)

- Nearest point sampling leads to the well-known artifact.
- Unfortunately, bilinear interpolation doesn't help: A pixel is either fully lit or fully shadowed and consequently the shadow quality is not improved at all by choosing bilinear interpolation.



Shadow Mapping – Shadow Map Filtering (cont'd)

- A solution to this problem is to first determine the visibilities of a pixel with respect to the four texels, and then interpolate the visibilities. This value is taken as the "degree of being lit."
- The technique of taking multiple texels from the shadow map and blending the pixel's visibilities against the texels is named percentage closer filtering (PCF). In the example, *p*'s visibility is set to 0.58.