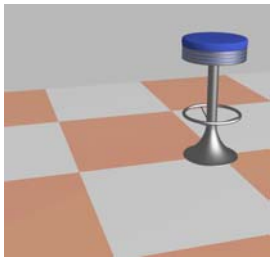
University of
British Columbia

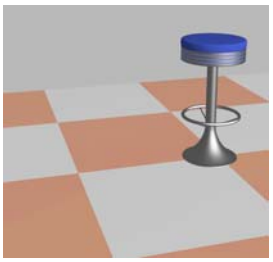
Chapter 10

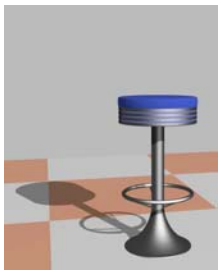
Shadows

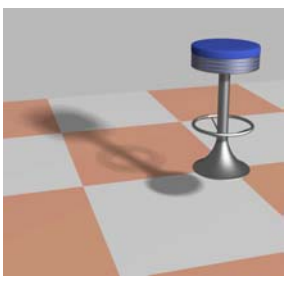


Shadows

- Realistic illumination includes shadows cast by objects
- Simple shadow generation methods
  - Z-buffer extension
  - Shadow volumes
- Advanced – ray-tracing & radiosity



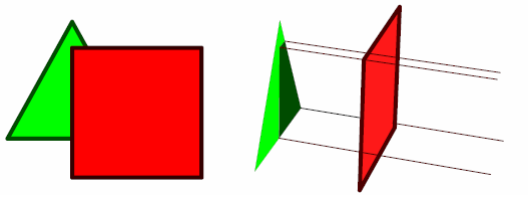




University of
British Columbia

Z-Buffer Shadow Generation

- Object is in shadow if not "seen" by light source
- Idea – compute visibility from light source to decide if shadowed



University of British Columbia

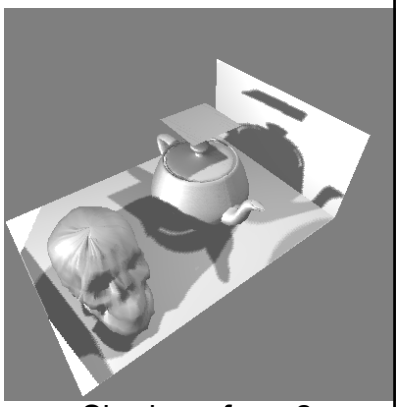
Z-Buffer Shadow Algorithm

- Render scene from light-source "viewpoint"
- For each pixel save *z* depth instead of color
- Render scene from eye view point
- Map every (visible) non-background pixel to light source space (perspective transformation)
- Compare z values
  - If identical – pixel illuminated by light source (add light source to its illumination equation)
  - If not, it is shadowed
- Need to repeat rendering & projection for each light source

University of British Columbia

Properties

- Can shadow ANY scene which can be rendered using Z-buffer
- However -requires separate memory buffer for each light source
- Every polygon rendered **N+1** times (for **N** light sources)
  - **N** views do not need lighting calculations

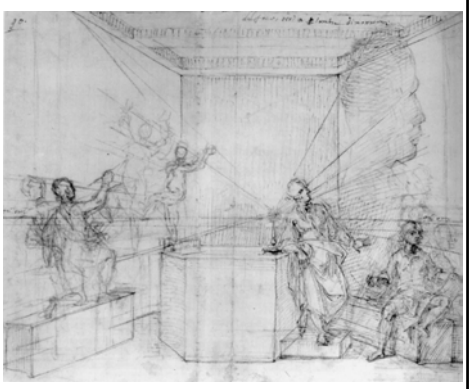


Shadows from 2 light-sources

University of British Columbia

Shadow Volumes

- Shadow – boundary between illuminated & shades space
- Compute as extrusion of silhouettes along light direction
- Compute intersection of extruded volume with other objects

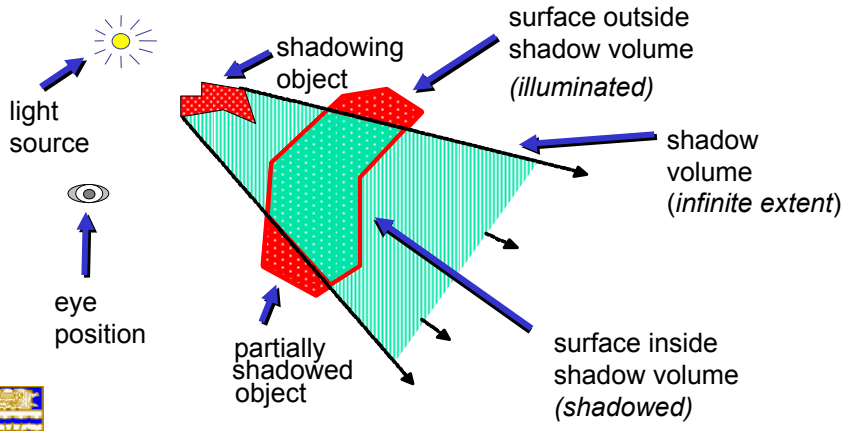


Shadow volumes circa Leonardo daVinci

University of British Columbia

Shadow Volumes

- Shadow Volumes illustrated (2D)



University of British Columbia

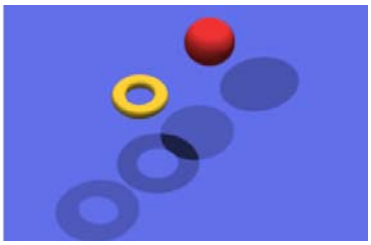
Algorithm

- For each object and light source compute object silhouette from light source viewpoint
- Extend each silhouette to form semi-infinite volumes
- Feed boundaries into regular Z-buffer as fully transparent polygons
- Front facing shadow polygons cause object behind to be shadowed
- Back facing shadow polygons cancel effect of front facing ones
- Consider vector from viewpoint to point on object – point is shadowed if vector intersects more front facing polygons than back facing

University of British Columbia

Properties

- Object space – does not depend on view point
- High complexity per object
- Time - function of scene complexity
- Requires modeling methods
  - Silhouette computation
  - Extrusion



University of
British Columbia