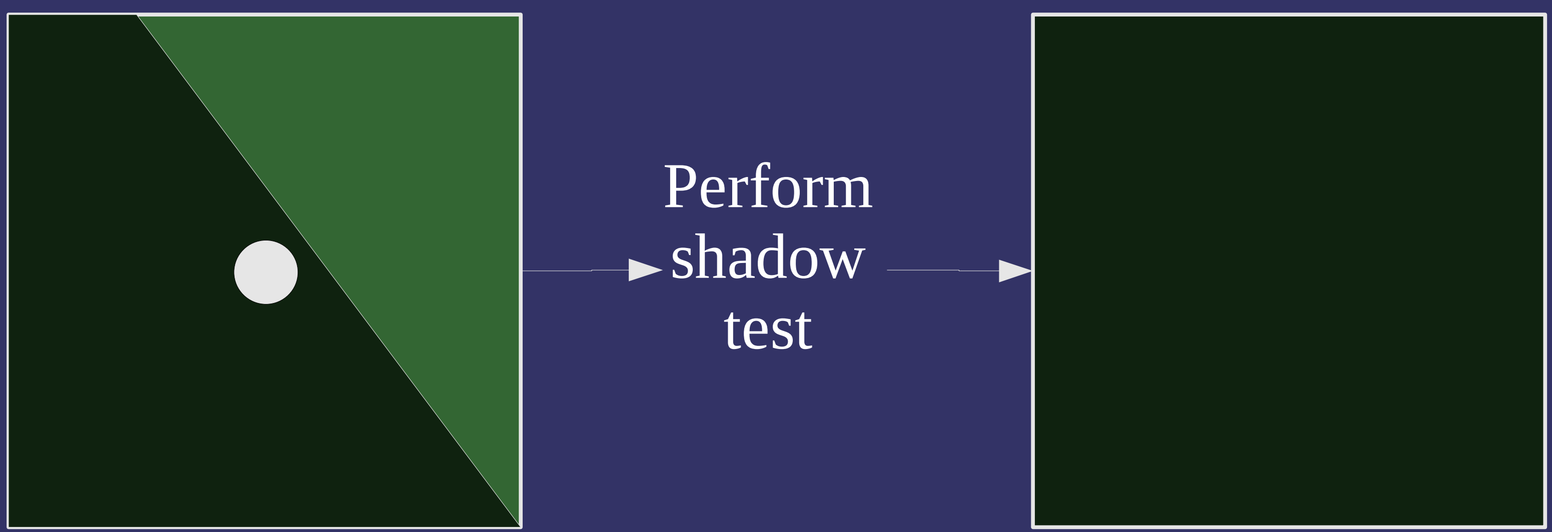
VGP353 – Week 4

- Agenda:
  - Fighting shadow map aliasing
    - Percentage closer filtering (PCF)
    - Perspective shadow maps

22-April-2008

Shadow Map Texture Filtering

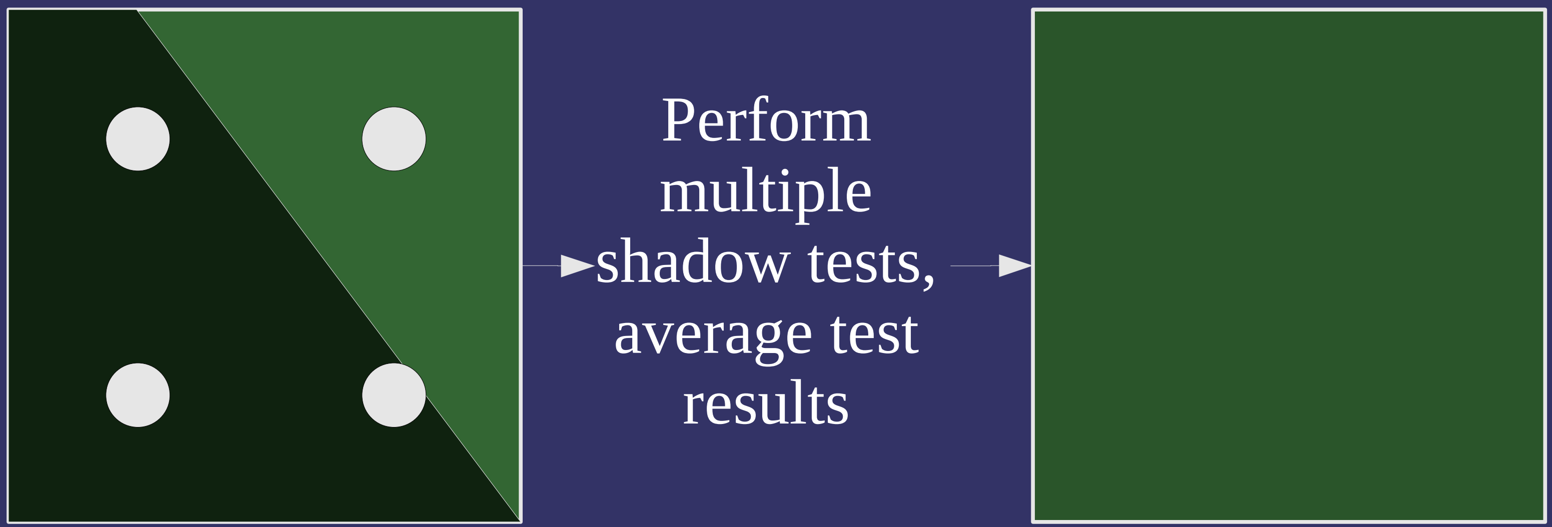
- Shadow test samples shadow map once per fragment
  - Results in two possible light levels: fully lit or fully shadowed



22-April-2008

Shadow Map Texture Filtering

- Percentage closer filtering (PCF) reads multiple samples, performs one test per sample, averages test results
  - Results in n+1 possible light levels, where n is the the number of samples



Shadow Map Texture Filtering

- Straight forward implementation in GLSL:

```
uniform vec2 bias;
uniform sampler2DShadow map;

void main()
{
    vec3 proj = coord.xyz / coord.w;
    vec3 p0 = proj - vec3((0.5 * bias.xy), 0.0);

    vec4 shadow = shadow2D(map, p0);
    shadow += shadow2D(map, p0 + vec3(bias.x, 0.0, 0.0));
    shadow += shadow2D(map, p0 + vec3(0.0, bias.y, 0.0));
    shadow += shadow2D(map, p0 + vec3(bias.x, bias.y, 0.0));

    shadow /= 4.0;

    ...
}
```

Percentage Closer Filtering

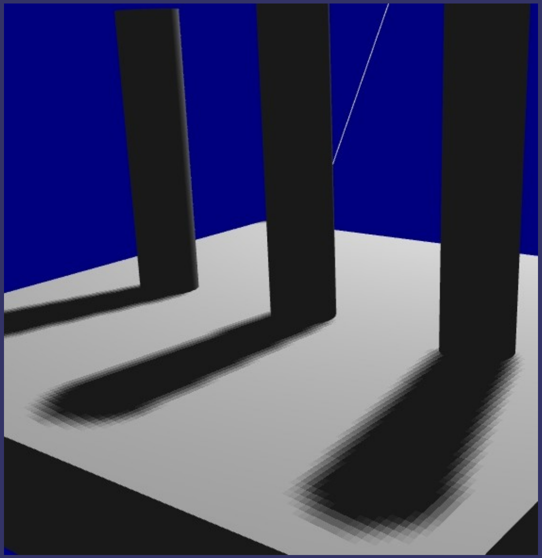
- The good news:
  - Improves quality
  - Larger filter kernels can be used to enable soft shadows
  - Some hardware can do 2x2 PCF nearly for free
    - Just enable `GL_LINEAR` filter on Nvidia hardware

Percentage Closer Filtering

- The good news:
  - Improves quality
  - Larger filter kernels can be used to enable soft shadows
  - Some hardware can do 2x2 PCF nearly for free
    - Just enable `GL_LINEAR` filter on Nvidia hardware
- The bad news:
  - Larger filter kernels are expensive
  - Grid-based sampling has artifacts

Grid-Based Sampling

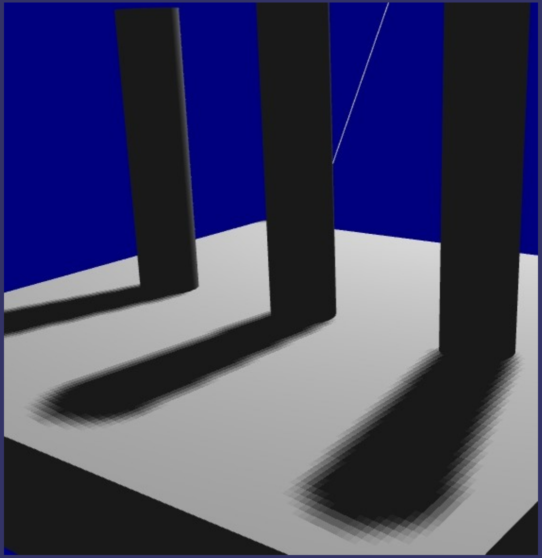
- Grid-based sampling artifacts have regular shape and are easily noticed by the eye

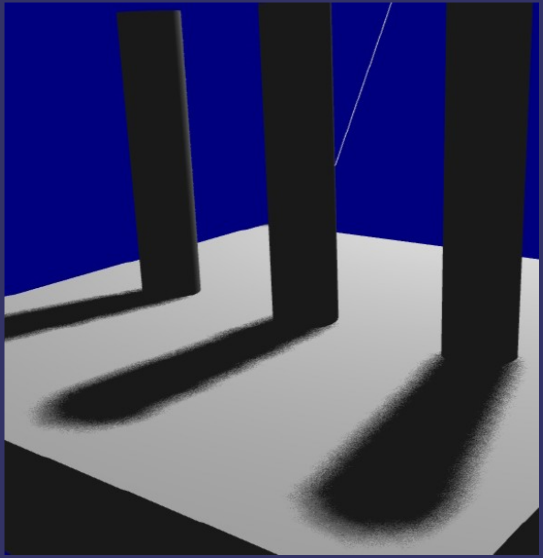


Images from http://ati.amd.com/developer/SIGGRAPH05/ShadingCourse_ATI.pdf

Grid-Based Sampling

- Grid-based sampling artifacts have regular shape and are easily noticed by the eye
- Irregular sample patterns are more easily accepted by the eye
  - Can even use fewer samples in the same size area

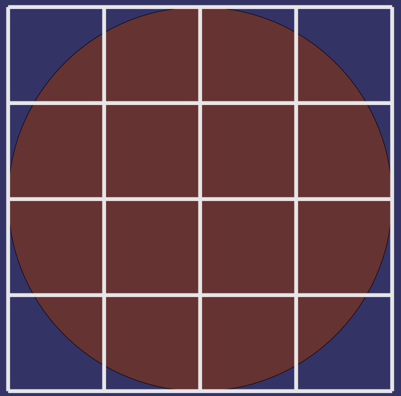




Images from http://ati.amd.com/developer/SIGGRAPH05/ShadingCourse_ATI.pdf

Irregular Sampling

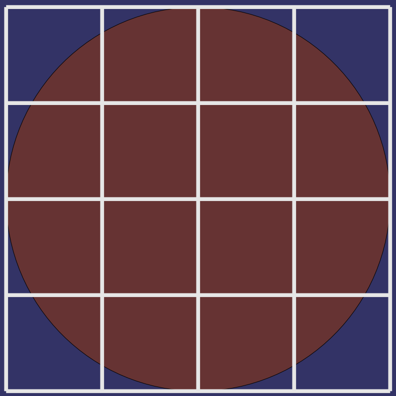
- Select filter area



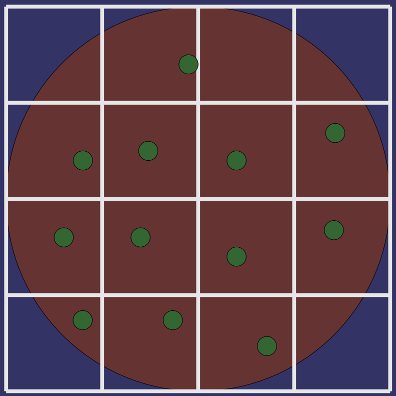
22-April-2008

Irregular Sampling

- Select filter area

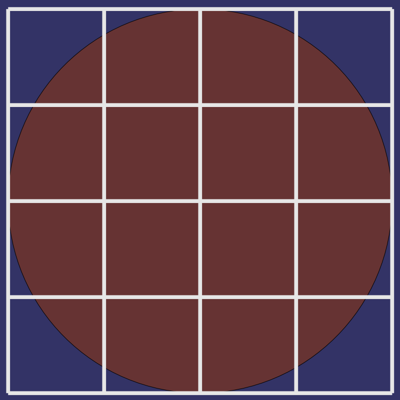


- Select random sample locations within area

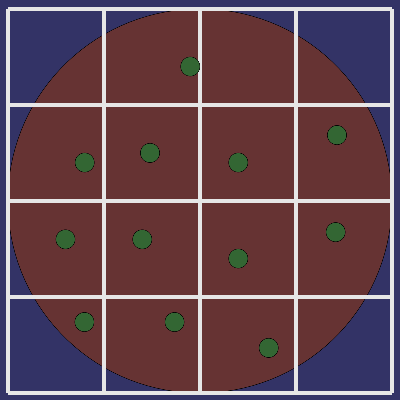


Irregular Sampling

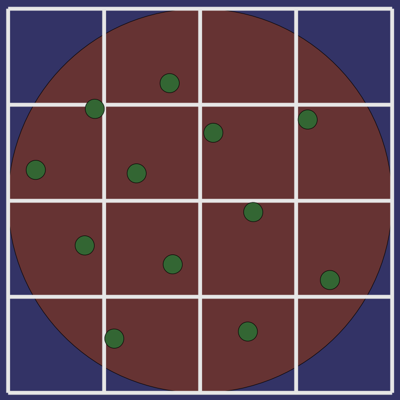
- Select filter area



- Select random sample locations within area



- Randomly rotate sample locations
  - Rotation based on screen location



Filter Cost

- 12 or 16 samples per fragment is expensive

Filter Cost

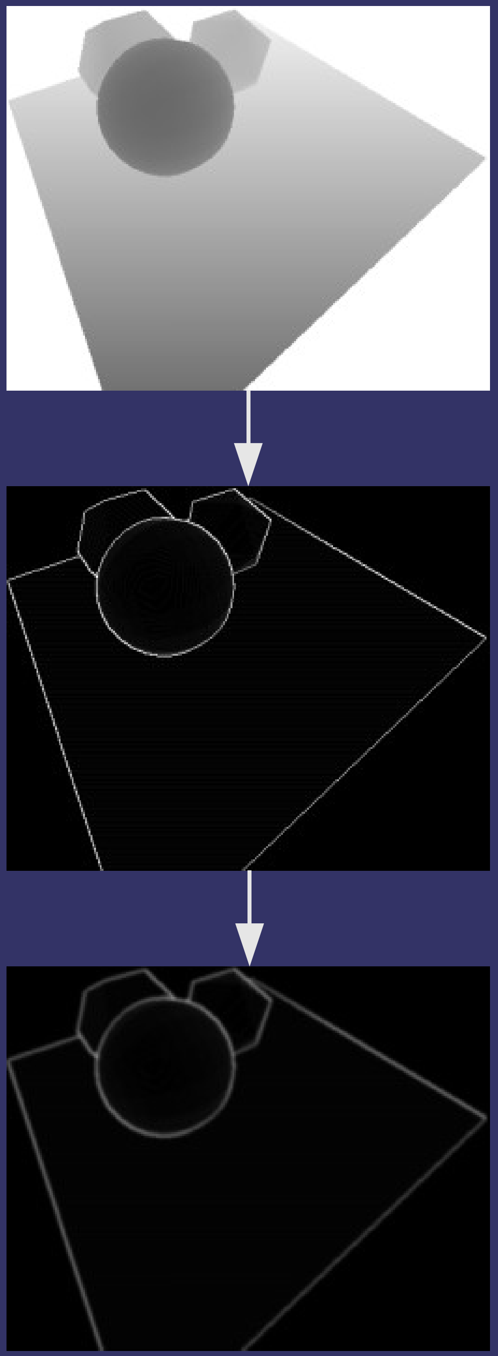
- 12 or 16 samples per fragment is expensive
  - In most of the final image, expensive sampling is unnecessary
  - Nyquist–Shannon sampling theorem tells us that areas with only low-frequency information need fewer samples than areas with high-frequency information

Filter Cost

- 12 or 16 samples per fragment is expensive
  - In most of the final image, expensive sampling is unnecessary
  - Nyquist–Shannon sampling theorem tells us that areas with only low-frequency information need fewer samples than areas with high-frequency information
    - The only high-frequency information is near the shadow boundaries!

Shadow Boundary Map

- Find boundaries in shadow map using edge detection filter
  - The edges in the map are the regions where the expensive filter should be applied
- Blur edge map using a blur kernel equal in size to the shadow map sample filter
  - This increases the area where the expensive filter will be applied and ensures that it will be applied everywhere that it needs to be



22-April-2008

Shadow Boundary Map

- Use the shadow boundary map to determine whether to use one or many shadow samples

```
if (texture2DProj(boundary_map, proj) > 0.0) {
    shadow = pcf_shadow_filter(shadow_map, proj);
} else {
    shadow = shadow2DProj(shadow_map, proj)
}
```

  - On hardware that support dynamic flow control, this can be a *big* win
    - DFC is a required part of DX 9.0c Shader Model 3.0
      - Geforce6 and later
      - Radeon X1xxx (R500) and later
      - Intel GMA X3000 (G965) and later

References

Sander, P. and Isidoro, J. *Explicit Early-Z Culling and Dynamic Flow Control on Graphics Hardware*. ATI Corporation, 2005, accessed 20 April 2008; available from http://ati.amd.com/developer/techpapers.html

Break

Shadow Map Aliasing

- A shadow map texel represents an area $d_s \times d_s$
  - d_s is the reciprocal of the shadow map resolution
  - As shadow map resolution increases, d_s decreases
  - The projected size of a surface at distance r_s is approximately:

$$\frac{d_s r_s}{N \cdot L}$$

Shadow Map Aliasing

- An image pixel represents an area $d_i \times d_i$
  - d_i is the reciprocal of the image resolution
  - As image resolution increases, d_i decreases
  - The projected size of a surface at distance r_i is approximately:

$$\frac{d_i r_i}{N \cdot V}$$

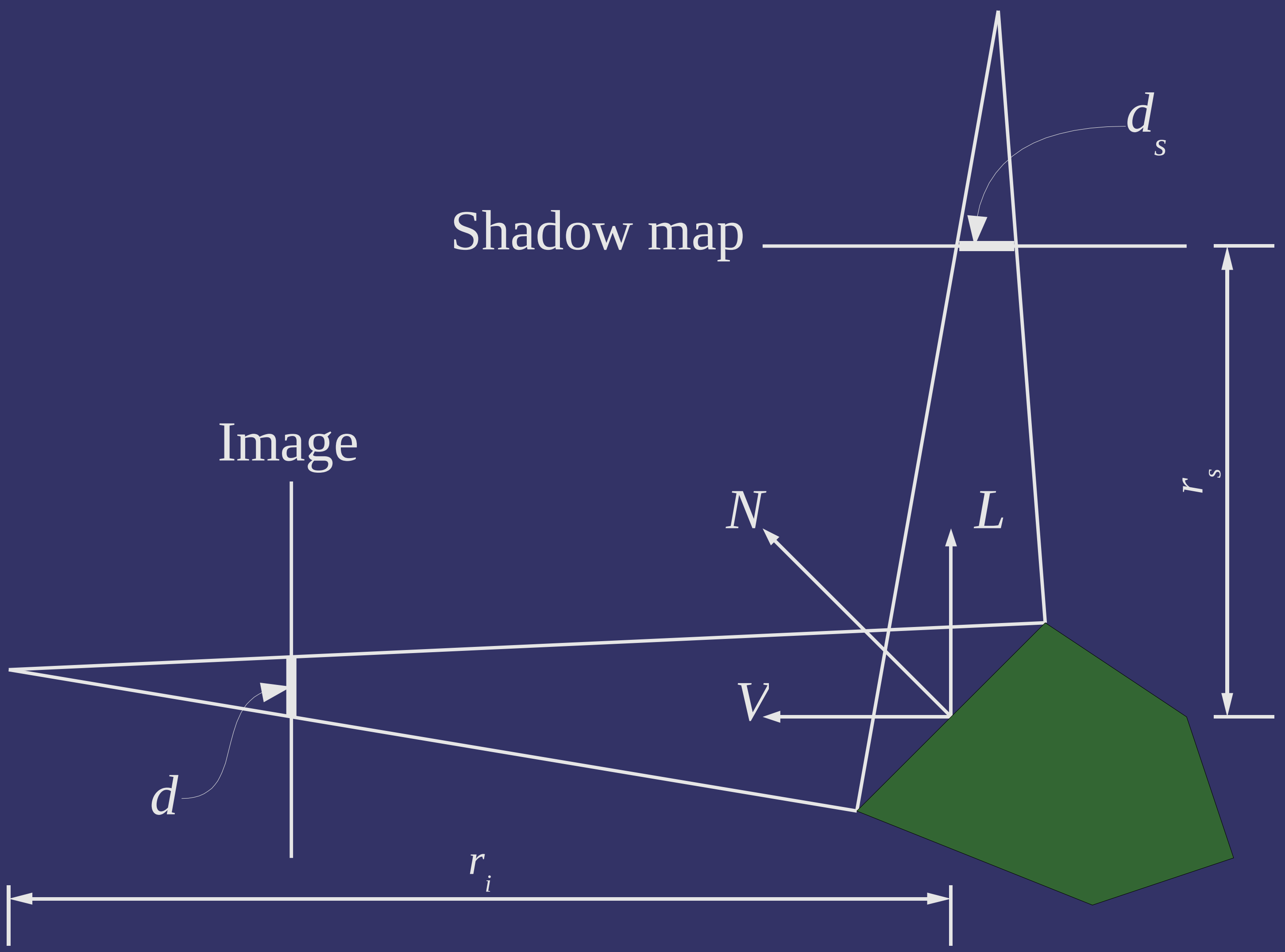
Shadow Map Aliasing

- The size of the projection of the shadow texel in the final image is:

$$d = d_s \frac{r_s}{r_i} \frac{N \cdot V}{N \cdot L}$$

  - Aliasing occurs when $d > d_i$

Shadow Map Aliasing



22-April-2008

Shadow Map Aliasing

- The size of the projection of the shadow texel in the final image is:

$$d = d_s \frac{r_s}{r_i} \frac{N \cdot V}{N \cdot L}$$

  - Aliasing occurs when $d > d_i$
  - Intuitively, if the shadow area is small in the shadow map, but large in the final image, there will be aliasing

Shadow Map Aliasing

$$d_s \frac{r_s}{r_i} \frac{N \cdot V}{N \cdot L}$$

Large when light rays are nearly tangent to surface geometry, but surface geometry faces towards the viewer

- This is called *projection aliasing*
- Dependent on orientation of scene geometry
- Can change even when light and viewer are stationary
- Difficult to fix!

Shadow Map Aliasing

$$d_s \frac{r_s}{r_i} \frac{N \cdot V}{N \cdot L}$$

Occurs when the view is close to individual texels of the shadow map

- This is called *perspective aliasing*
- Occurs if the shadow map is too small (i.e., d_s is large)
- Can only increase shadow map size so much!
- Also occurs if $r_s \gg r_i$

Large when light rays are nearly tangent to surface geometry, but surface geometry faces towards the viewer

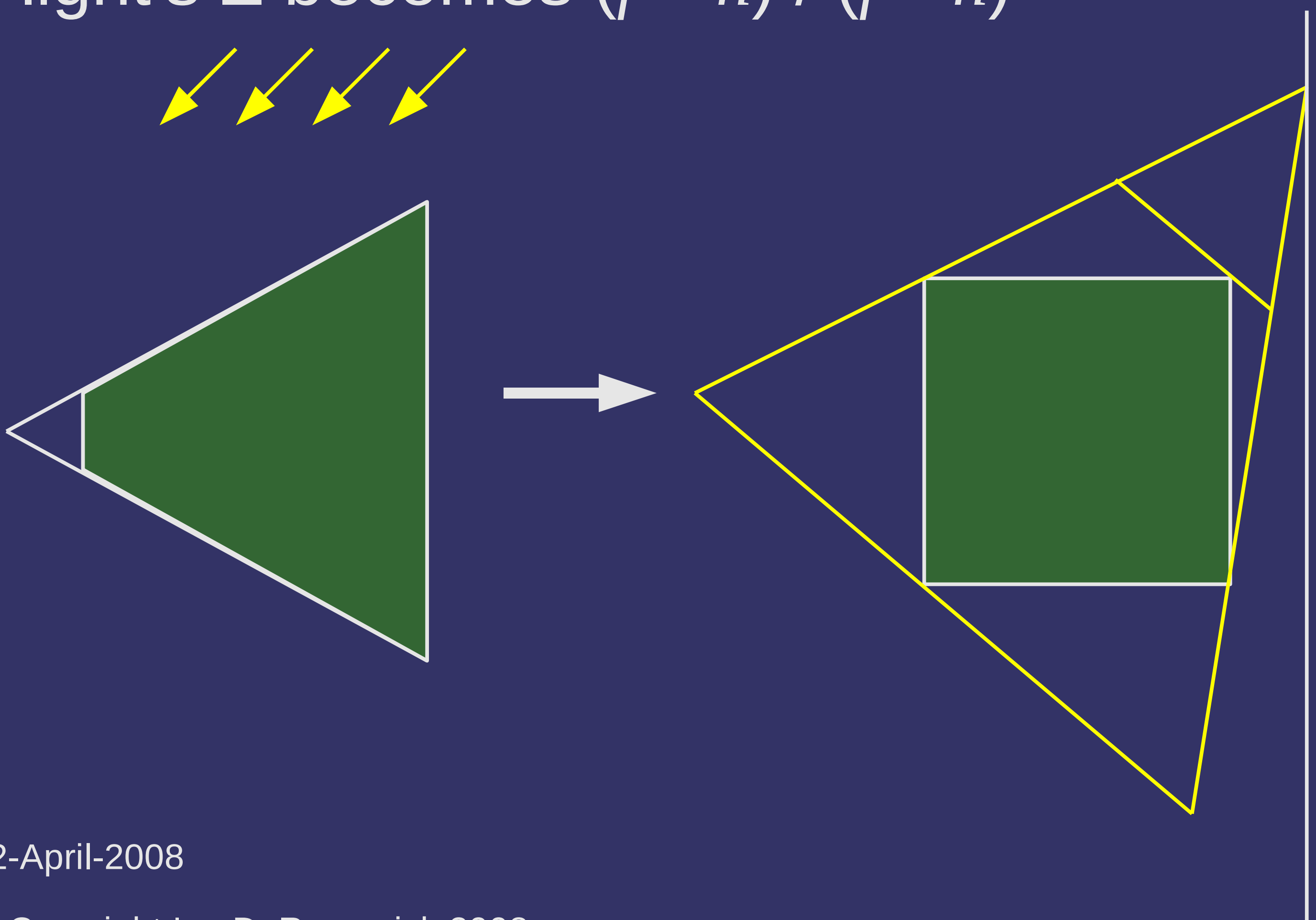
- This is called *projection aliasing*
- Dependent on orientation of scene geometry
- Can change even when light and viewer are stationary
- Difficult to fix!

Perspective Shadow Maps

- If the problem stems from the relationship between the camera frustum and light frustum, then the solution make take both frusta into account
  - Perform shadow map calculations in post-projection camera space *instead of* world space
  - The projection remaps the frustum volume to a cube, this cube is then sampled to create the shadow map
    - Applying this to the world before applying the light's view effectively changes the “shape” of the light

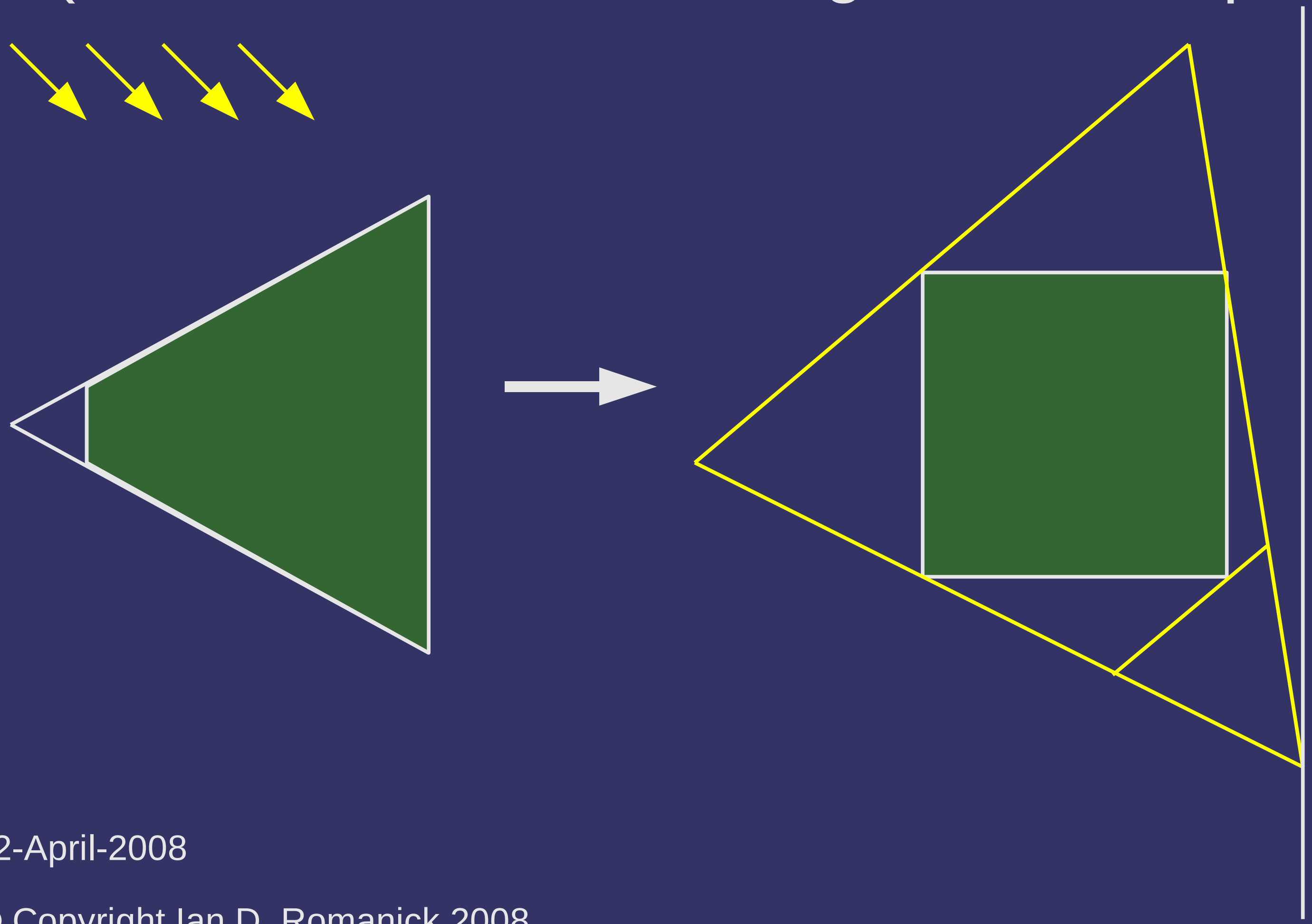
Perspective Shadow Maps

- Directional lights become point lights “on the infinity plane”
  - The light's Z becomes $(f + n) / (f - n)$



Perspective Shadow Maps

- Directional front-lights become inverted
  - Reverse the order of the usual depth and shadow tests (i.e., less-than becomes greater-or-equal)

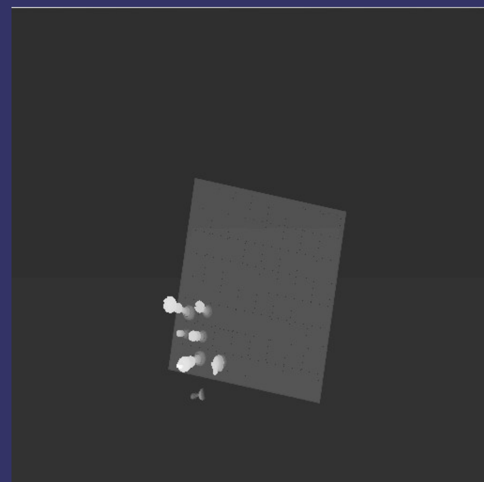


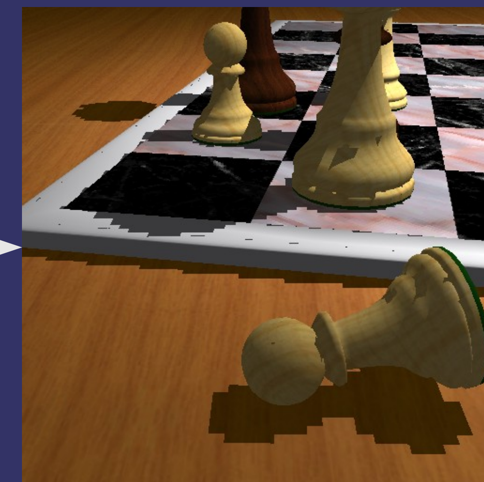
Perspective Shadow Maps

- Directional lights have other quirks
  - The more parallel the light and view direction, the lower the quality
    - A directional light pointing in the exact opposite direction of the view direction degrades back to the classic shadow map case
  - Casters behind the viewer (i.e., negative Z) are inverted and projected past the far plane
    - Several methods to handle this special case
  - Point lights have similar issues

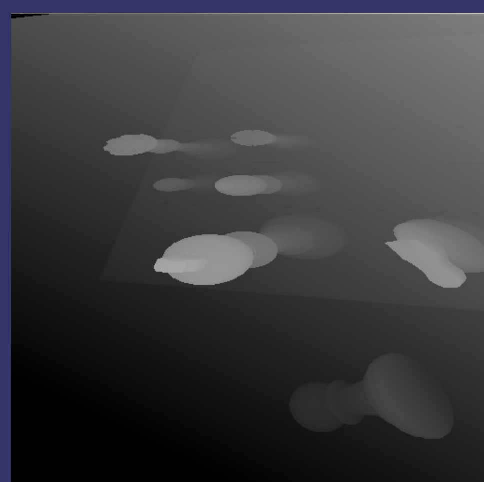
Perspective Shadow Map

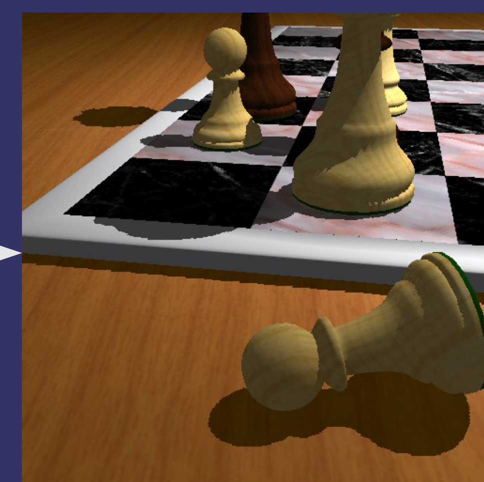
Standard
shadow map





Perspective
shadow map





Images from http://www-sop.inria.fr/reves/publications/data/2002/SD02/index.gb.html

22-April-2008

Perspective Shadow Maps

- Advantages:
  - Improves quality for many common cases
  - Easy to implement for directional light sources
- Disadvantages:
  - Shadow maps are view dependent, and must be regenerated when the camera moves (instead of just when the light or objects move)
  - Dual perspective transforms exaggerate shadow acne
  - As the viewer moves, the quality of the shadow map changes...even if the rest of the scene is static
    - For *most* games, this is the deal breaker

References

Stamminger, M. and Drettakis, G. 2002. Perspective shadow maps. In *Proceedings of the 29th Annual Conference on Computer Graphics and interactive Techniques* (San Antonio, Texas, July 23 - 26, 2002). SIGGRAPH '02. ACM, New York, NY, 557-562. http://www-sop.inria.fr/reves/publications/data/2002/SD02/index.gb.html

Next week...

- End of shadow maps
  - Percentage closer soft shadows
  - Parallel split shadow maps
- Quiz #2
  - Week 3 and week 4 material
- Assignments:
  - Programming assignment #2 due
  - Programming assignment #3 begins
  - First reading presentation!

Legal Statement

This work represents the view of the authors and does not necessarily represent the view of IBM or the Art Institute of Portland.

OpenGL is a trademark of Silicon Graphics, Inc. in the United States, other countries, or both.

Khronos and OpenGL ES are trademarks of the Khronos Group.

Other company, product, and service names may be trademarks or service marks of others.