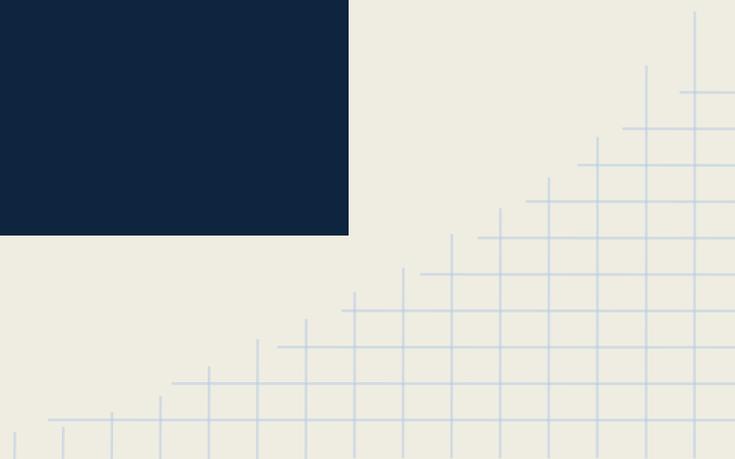




Shadows

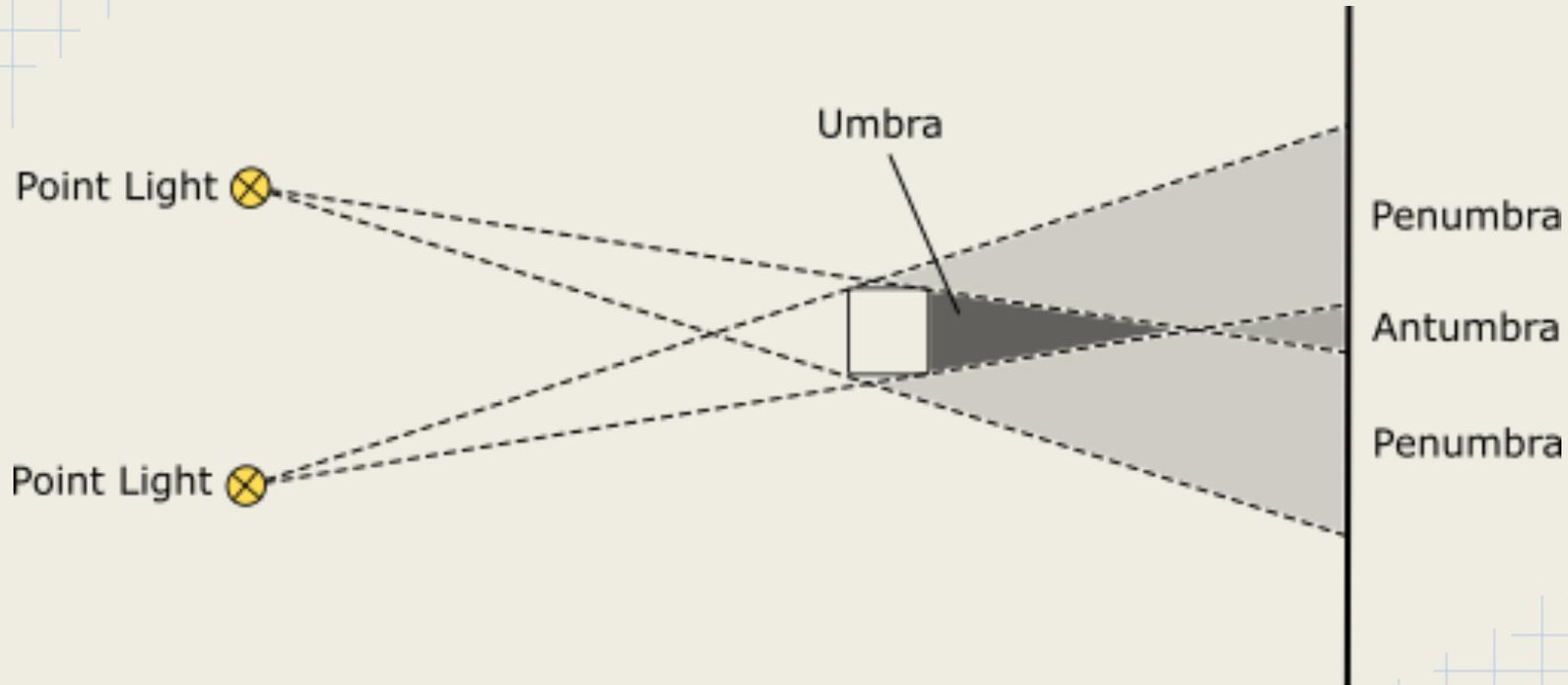
Margus Luik



Outline

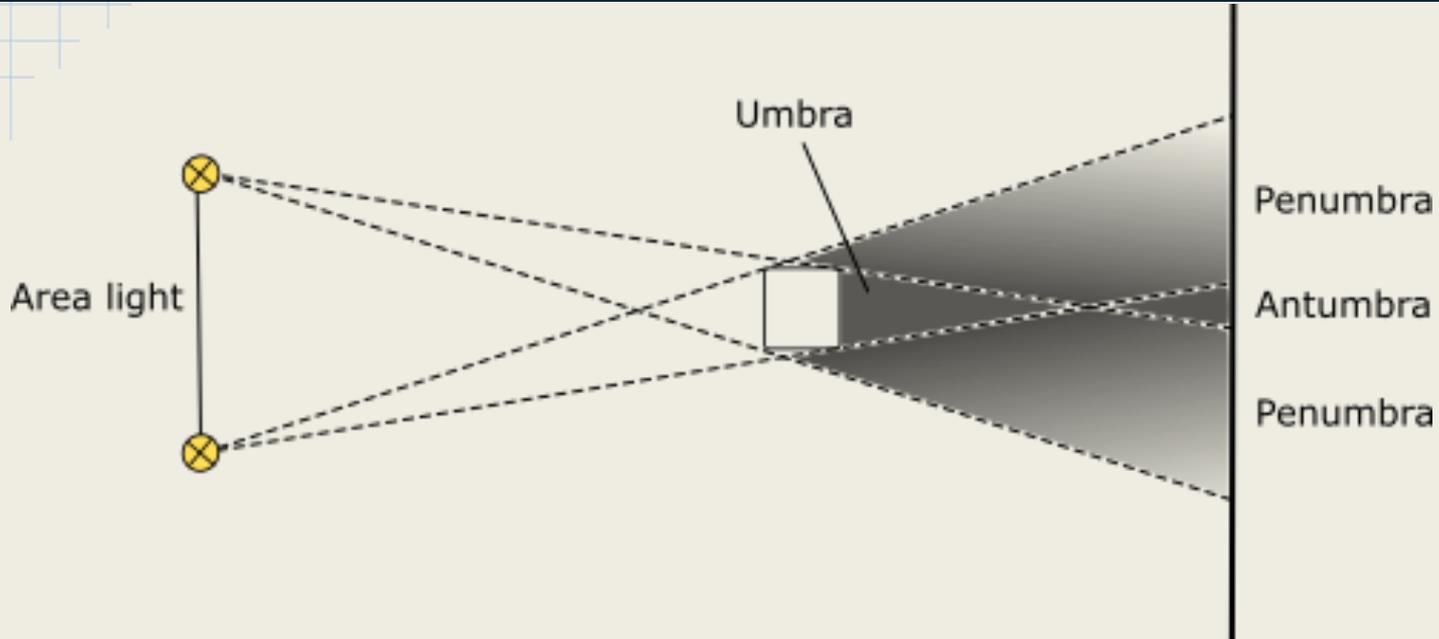
- Terminology
- Simple projected shadows
- Projection shadows
- Shadow mapping
- Stencil shadows

Terminology



But what if we used an area light?

Terminology



Soft shadows - edges of shadows are blurry

Hard shadows - edges of shadows are distinct. Caused by point lights

Terminology

- Static shadows - shadows from stationary objects in the scene.
- Dynamic shadows - shadows cast by objects which can move in the scene.
- Lightmap - static lighting and shadows can be baked into the texture assets.

Outline

- Terminology
- **Simple projected shadows**
- Projection shadows
- Shadow mapping
- Stencil shadows

Simple projected shadows

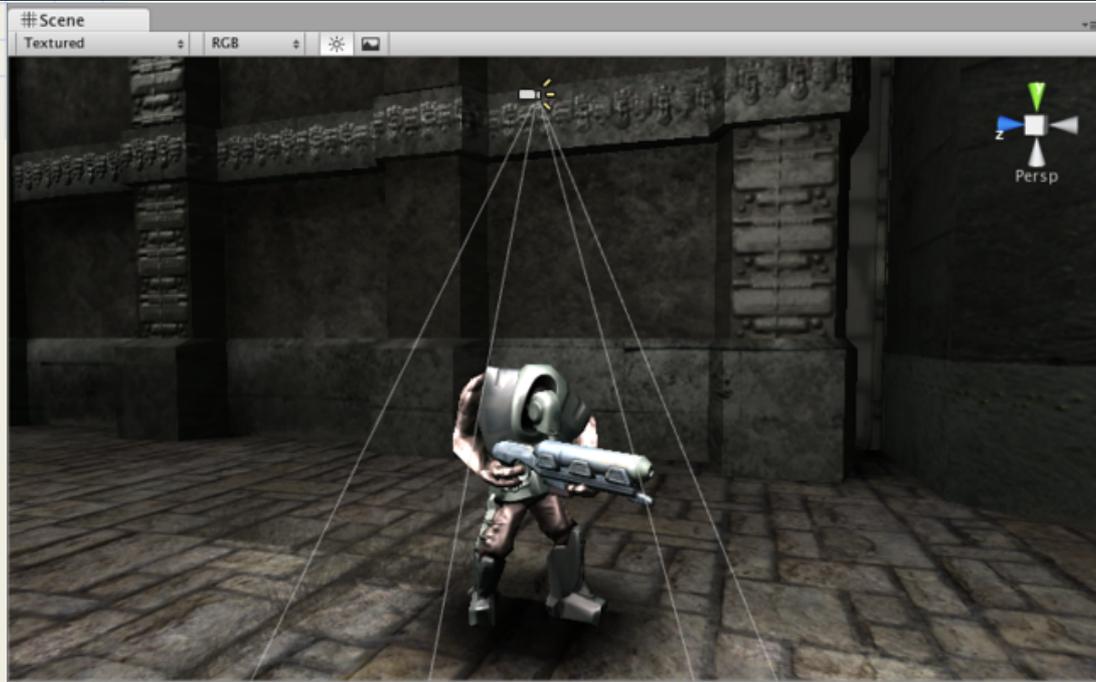
What is the simplest way you can think of to add shadows?

Simple projected shadows

What is the simplest way you can think of to add shadows?

Just create new objects for the shadows and add them to the scene.

Simple projected shadows



Using a projector to add shadow to a character in Unity 3D

Simple projected shadows

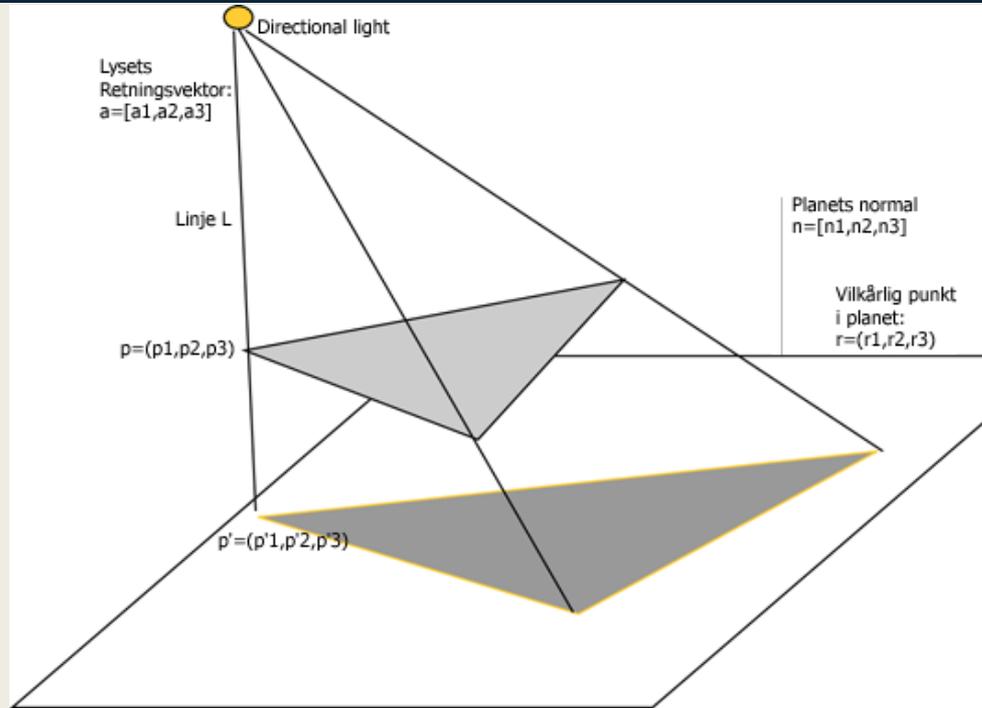
Pros/cons:

- Shadows are just additional objects placed into the scene.

Outline

- Terminology
- Simple projected shadows
- **Projection shadows**
- Shadow mapping
- Stencil shadows

Projection shadows



source: <http://www.it.hiof.no/~borres/j3d/explain/shadow/p-shadow.html>

Projection shadows

How-to:

1. Render the scene in usual way
2. Make second rendering pass with alterations:
 - a. squash the object into single plane using scaling;
 - b. make necessary transformations to position and rotate the shadow to a plane.
 - c. change the color of the objects.

See: <ftp://ftp.sgi.com/opengl/contrib/blythe/advanced99/notes/node192.html>

Projection shadows

Pros:

- Easy to implement
- Useable in games where viewer is far away and game takes place on a plane. Eg strategy or tycoon games.

Cons:

- Can only be cast on planar surfaces
- No self shadowing
- Can lead to z-fighting, unless properly offset
- Can cause double blending, unless stencil buffer is used

Outline

- Terminology
- Simple projected shadows
- Projection shadows
- **Shadow mapping**
- Stencil shadows

Shadow mapping

Shadows are created by testing whether a pixel is visible from the light source, by comparing it to a z-buffer or depth image of the light source's view, stored in the form of a texture.

Shadow mapping

How-to:

1. Enable only depth buffer.

For each light source render:

2. scene from the light source point of view and save the z buffer.
 - a. For spotlight perspective cameras can be used.
 - b. Point light may need multiple perspective cameras, if there are objects in multiple directions from the point light.
 - c. For directional light use orthographic camera.
3. Render the scene with shadows.
 - a. If currently drawn fragments z value is greater than the stored depth value from step 2, then the fragment is in shadow.
 - b. If z value is lesser or equal, then apply lighting.

Shadow mapping

Pros:

- Generally faster than shadow volume technique
- OpenGL provides samplers for shadow map textures.

Cons:

- It is difficult to set good resolution for the shadow map. Shadows close to the camera may have jagged edges, while at the same time shadows far away from camera have too high resolution.

Outline

- Terminology
- Simple projected shadows
- Projection shadows
- Shadow mapping
- **Stencil shadows**

Stencil shadows

- Also known as shadow volume
- Made popular in Doom III (i.e. only used in Doom III)
- Variations:
 - depth pass
 - depth fail
 - exclusive-or

Stencil shadows

How-to:

1. Render the scene with only ambient light

For every light:

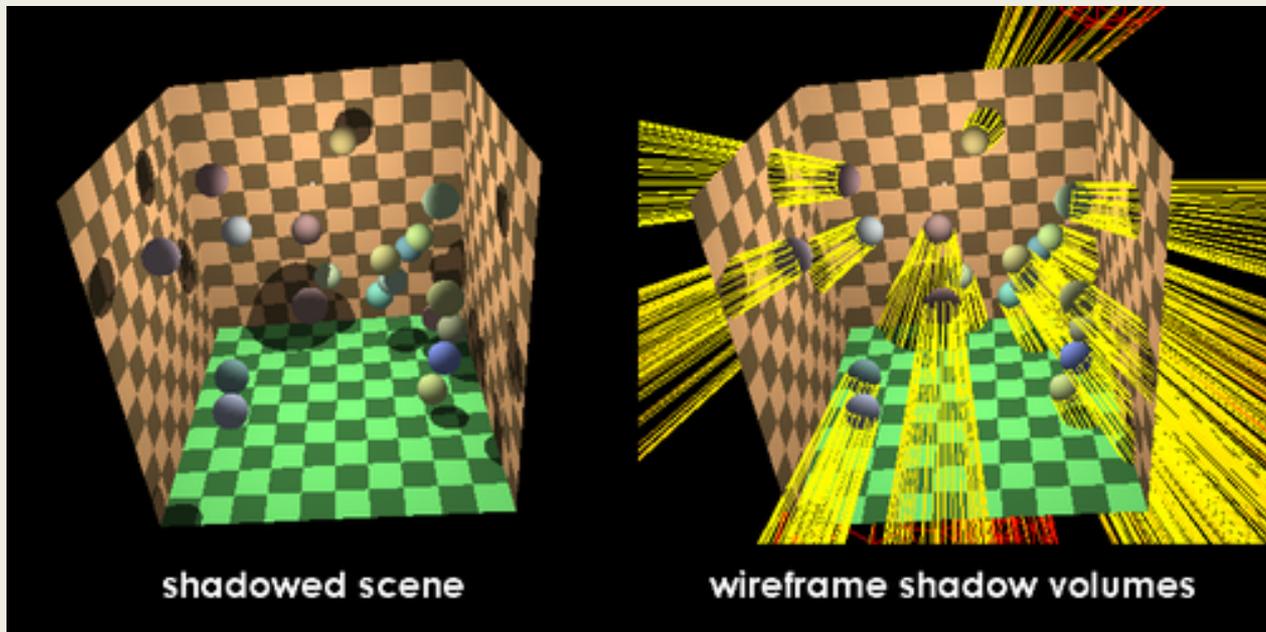
2. **Create shadow volumes**

- a. Find all silhouette edges
- b. Extend all silhouette edges away from light source
- c. Add a front cap and back cap to each surface to form a closed volume.

3. Render the shadow volumes into stencil buffer

4. Render the scene with diffuse and specular lights using the stencil buffer.

5. Blend the color buffer additively.



source: http://en.wikipedia.org/wiki/Shadow_volume

Stencil shadows

3. [Depth pass]Render the shadow volumes into stencil buffer
 - a. Disable writing to depth and color buffer.
 - b. Use back-face culling.
 - c. Set the stencil operation to increment on depth pass (only count shadows in front of the object).
 - d. Render the shadow volumes
 - e. Use front-face culling.
 - f. Set the stencil operation to decrement on depth pass.
 - g. Render the shadow volumes.

Stencil shadows

3. [Depth fail]Render the shadow volumes into stencil buffer
 - a. Disable writing to depth and color buffer.
 - b. Use front-face culling.
 - c. Set the stencil operation to increment on depth fail (only count shadows behind the object).
 - d. Render the shadow volumes.
 - e. Use back-face culling.
 - f. Set the stencil operation to decrement on depth fail.
 - g. Render the shadow volumes.

Stencil shadows

How-to:

1. Render the scene with only ambient light

For every light:

2. Create shadow volumes
3. Render the shadow volumes into stencil buffer
4. Render the scene with diffuse and specular lights using the stencil buffer.
 - a. Only use lighting on fragments where stencil buffer = 0.
5. Blend the color buffer additively.

Stencil shadows

Pros:

- Accurate to a pixel

Cons:

- CPU intensive. Have to create shadow volumes every frame for all light sources.
- Causes self-shadowing artifacts along the silhouette edge.