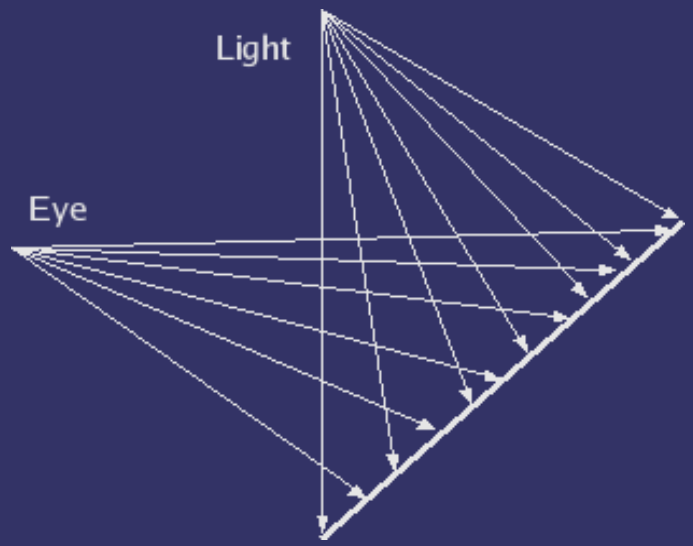
Shadow Maps, part 3

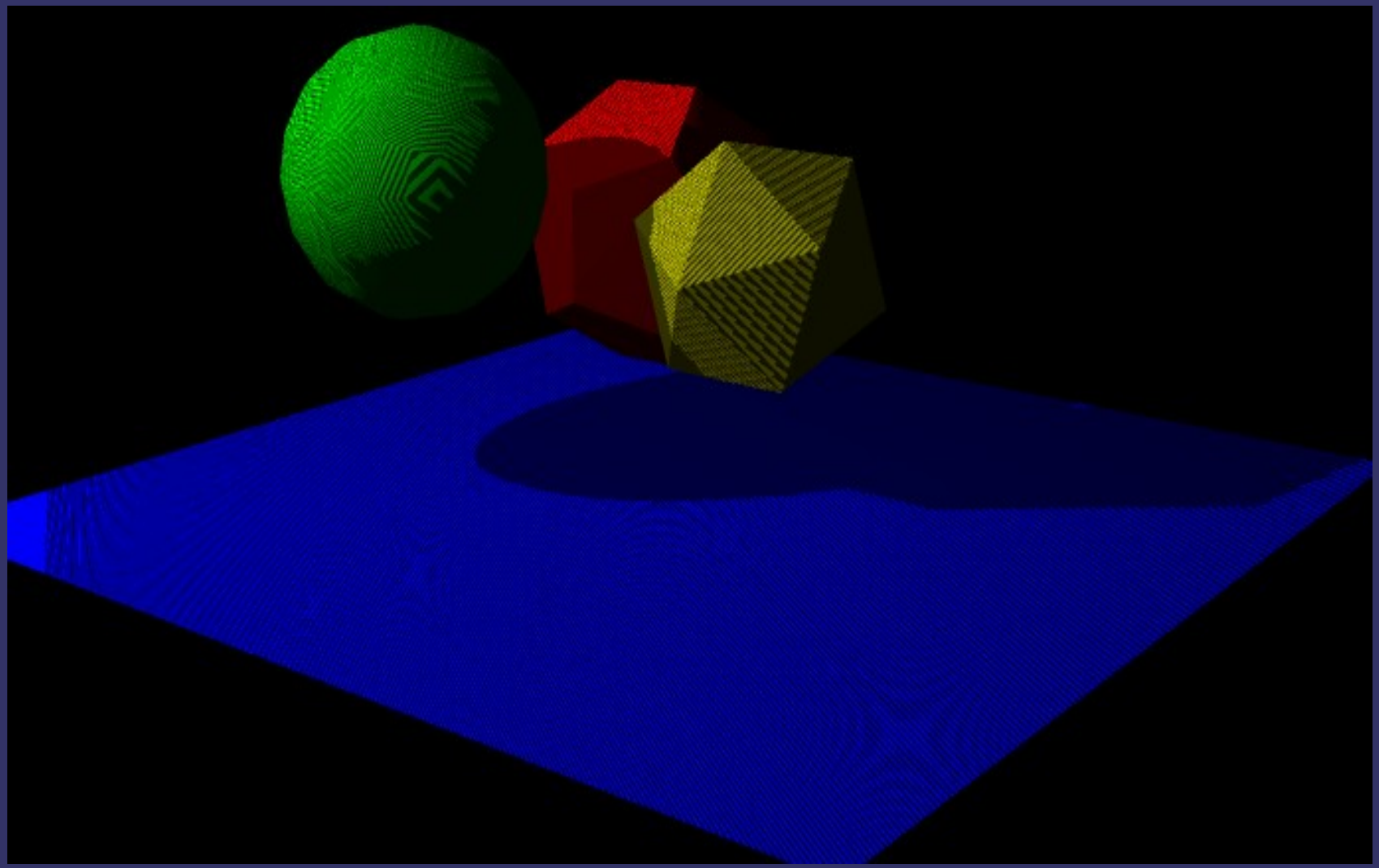
- Agenda:
  - Reading presentation
  - Quiz #2!
  - Wrap up shadow maps
  - Introduce the stencil buffer
  - Lab time:
    - Work on assignment #2
    - Give out assignment #3

One last bit of basic shadow maps...

- Due to sampling issues, surfaces incorrectly self-shadow
  - Drawing the surface to the shadow map samples one set of (surface space) positions, but drawing to the screen samples a different set



Result: “shadow acne”



Depth bias

- Common fix is to use *polygon offset* functionality to bias depth values
  - Add small depth bias to all fragments on the polygon to guarantee the depth value is $\leq$ shadow map depth value
  - Very tricky to get right! Movie fx companies spend *lots* of time tweaking every frame to eliminate artifacts[1]

[1] G. King, "Shadow Mapping Algorithms." NVIDIA. 2004. ftp://download.nvidia.com/developer/presentations/2004/GPU_Jackpot/Shadow_Mapping.pdf

Using polygon offset in OpenGL

- Parameters set with `glPolygonOffset`
  - Two values are set: *factor* and *units*
- Depth value is adjusted by (*factor* $\times$ DZ) + (r $\times$ *units*)
  - DZ is a measure of the Z slope of the polygon
    - The more the polygon slopes, the more it will be biased
  - r is the (implementation specific) smallest value that will cause a measurable change

Using polygon offset in OpenGL (cont.)

- Common technique is to enable minimum offset via `glPolygonOffset(0.0, 1.0)`
- May achieve better results using *factor*, but requires more tuning
- Note: must enable for your primitive types
  - Call `glEnable` with one of `GL_POLYGON_OFFSET_FILL`, `GL_POLYGON_OFFSET_LINE`, or `GL_POLYGON_OFFSET_POINT`

What is the stencil buffer?

- An extra per-pixel buffer containing integer values
- Stencil buffer is often stored interleaved with depth buffer
  - 8-bit stencil with 24-bit depth is most common, but 1-bit stencil with 15-bit depth is sometimes available

What can you do with it?

- Write values to it! Several operations available:
  - GL_KEEP – leave the value alone
  - GL_ZERO – clear value to zero
  - GL_REPLACE – replace value with preset value
  - GL_INCR – increment value, clamp to max value
    - GL_INCR_WRAP increments but warps to zero
  - GL_DECR – decrement value, clamp to zero
    - GL_DECR_WRAP decrements but warps to max value
  - GL_INVERT – bitwise inversion of value

Writing values to the stencil buffer

- A different operation can be set for pixels that pass the Z test, fail the Z test, or fail the stencil test (see next slide)
  - `glStencilOp` sets all three operations
  - Several extensions and OpenGL 2.1 add the ability to perform a *different* set of operations for front facing and back facing polygons
    - We'll talk about this functionality in a few weeks

Miscellaneous stencil functions

- `glClearStencil` clears the stencil buffer to some value
- `glStencilMask` controls which bits can be written by stencil operations

Stencil testing

- `glStencilFunc` sets the operation, reference value, and a mask
  - The usual depth test values are available: `GL_NEVER`, `GL_LESS`, `GL_LEQUAL`, `GL_GREATER`, `GL_GEQUAL`, `GL_EQUAL`, `GL_NOTEQUAL`, and `GL_ALWAYS`
- Per-pixel, `(ref & mask) op (stencil & mask)` is used *before* the depth test to determine whether or not to write to the color buffer

Example

```
glClearStencil(0);
glEnable(GL_STENCIL_TEST);

/* Write 1 to stencil where polygon is drawn.
 */
glStencilFunc(GL_ALWAYS, 1, ~0);
glStencilOp(GL_KEEP, GL_KEEP, GL_REPLACE);
draw_some_polygon();

/* Draw scene only where stencil buffer is 1.
 */
glStencilFunc(GL_EQUAL, 1, ~0);
glStencilOp(GL_KEEP, GL_KEEP, GL_KEEP);
draw_scene();
```

Questions?

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.