CIS565 GPU Programming and Architecture

Final Project Milestone III

Forward + with Vulkan
Zimeng Yang & Liang Peng

Instructor: Patrick Cozzi
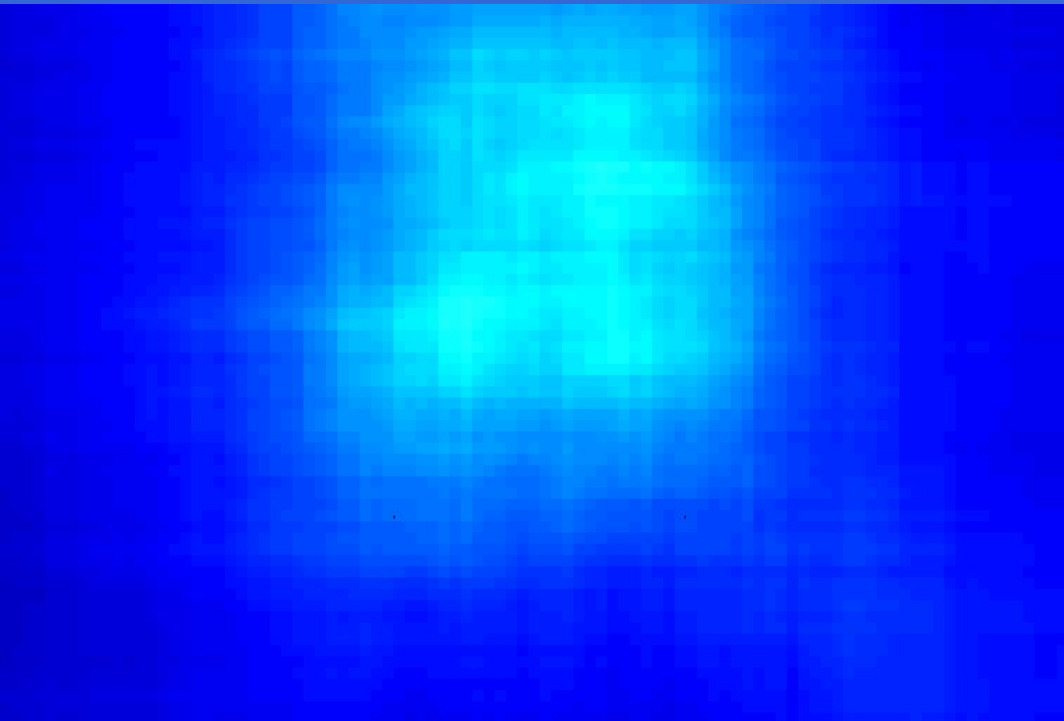
Finished Tasks

- Vulkan compute pipelines

  - Compute grid frustums pipeline

  - Compute light list pipeline

- Light culling

- Shading

GIF

- GTX 750m

- 1600 point lights

- Tiles of 8x8 pixels

GIF - Light Heat Map

- GTX 750m

- 1600 point lights

- Tiles of 8x8 pixels

- Brighter means more lights in tile

Working Code

```glsl
vec4 ComputePlane( vec3 p0, vec3 p1, vec3 p2 )
{
    vec4 plane;

    vec3 v0 = p1 - p0;
    vec3 v2 = p2 - p0;

    plane.xyz = normalize( cross( v0, v2 ) );

    // Compute the distance to the origin using p0.
    plane.w = dot( plane.xyz, p0 );

    return plane;
}
```

Plane representation
- Normal direction
- Distance to origin

Frustum representation
- 4 planes

Point light representation
- Sphere

Light culling
- Detect sphere frustum intersection

Working Code

```
uint lightIndexBegin = index * MAX_NUM_LIGHTS_PER_TILE;
uint lightNum = lightGrid[index];

// LightGrid[index] = Lights need to be considered in tile
for(int i = 0; i < lightNum; ++i) {
    int lightIndex = lightIndices[i + lightIndexBegin];

    Light currentLight = lights[lightIndex];

    vec3 beginPos = currentLight.beginPos.xyz;
    vec3 endPos = currentLight.endPos.xyz;
    float t = sin(params.time * lightIndex * .0005f);

    lightPos = (1 - t) * beginPos + t * endPos;
    lightColor = currentLight.color.xyz;
    lightDir = lightPos - fragPosWorldSpace;
    lightIntensity = currentLight.beginPos.w;
    lightRadius = currentLight.endPos.w;
```

Light list usage

- A global list of all lights
- A list of light indices for each tile
- A list of light count for each tile
- Shading: only iterate on the lights in the light list for each tile

Upcoming Milestones

- Final Presentation
  - Optimization
  - Code refactoring
  - Resource management
  - Performance Analysis

Thank you!