California Polytechnic State
University, San Luis Obispo

Real-Time Volumetric Cloud Rendering

Jaafer Sheriff

Senior Project

June 2018

Supervisor
Dr. Christian Eckhardt

ABSTRACT

short abstract

Contents

1 Introduction

introduction/motivation (1 page)

2 Voxelization

2.1 Spherical Billboards

Spherical billboard distribution

```
// Calculate distance from fragment to billboard center
float dist = distance(center, fragPos);
// Calculate linear distribution
float distribution = distToCenter / radius;
// Convert linear distribution to spherical
distribution = sqrt(max(0, 1 - distribution * distribution));
```

first_voxelize.glsl, 55

2.2 Position Map

Initial pos map writing

2.2.1 Concurrency

Now with concurrency

```
ivec2 texCoords = ivec2(gl_FragCoord.xy);
vec3 dir = normalize(fragNor);
float dist = radius * sphereContrib;
vec4 nearestPos = imageLoad(positionMap, texCoords);
if (nearestPos.a < 1.f) {
    nearestPos = vec4(FLT_MAX, FLT_MAX, FLT_MAX, 0.f);
}

/* Write to volume in spherical shape from billboard to light source */
vec3 start = fragPos - dir * dist;
for(float i = 0; i < 2*dist; i += stepSize) {
    vec3 worldPos = start + dir * i;
    ivec3 voxelIndex = calculateVoxelIndex(worldPos);
    imageAtomicAdd(volume, voxelIndex, f16vec4(0, 0, 0, 1));

    /* Keep track of nearest voxel position */
    if (distance(worldPos, lightPos) < distance(nearestPos.xyz,
        lightPos)) {
        nearestPos = vec4(worldPos, 1.f);
    }
}
/* Write nearest voxel position to position map */
imageStore(positionMap, texCoords, nearestPos);
```

first_voxelize.glsl, 55

2.3 Voxel viewing debug tools

2.4 Multiple Billboards

2.4.1 Concurrency

2.5 Optimizations

implementation (5-6 pages) where you argue in detail how you implemented what i.e. code

3 Voxel Cone Tracing

implementation (5-6 pages) where you argue in detail how you implemented what i.e. code

4 Noise Generation

4.1 Parameter playing

implementation (5-6 pages) where you argue in detail how you implemented what i.e. code

5 Results

3 page result (including pictures).