2016320272 권영진

CUDA PARTICLE

1. 파티클 모션 제어

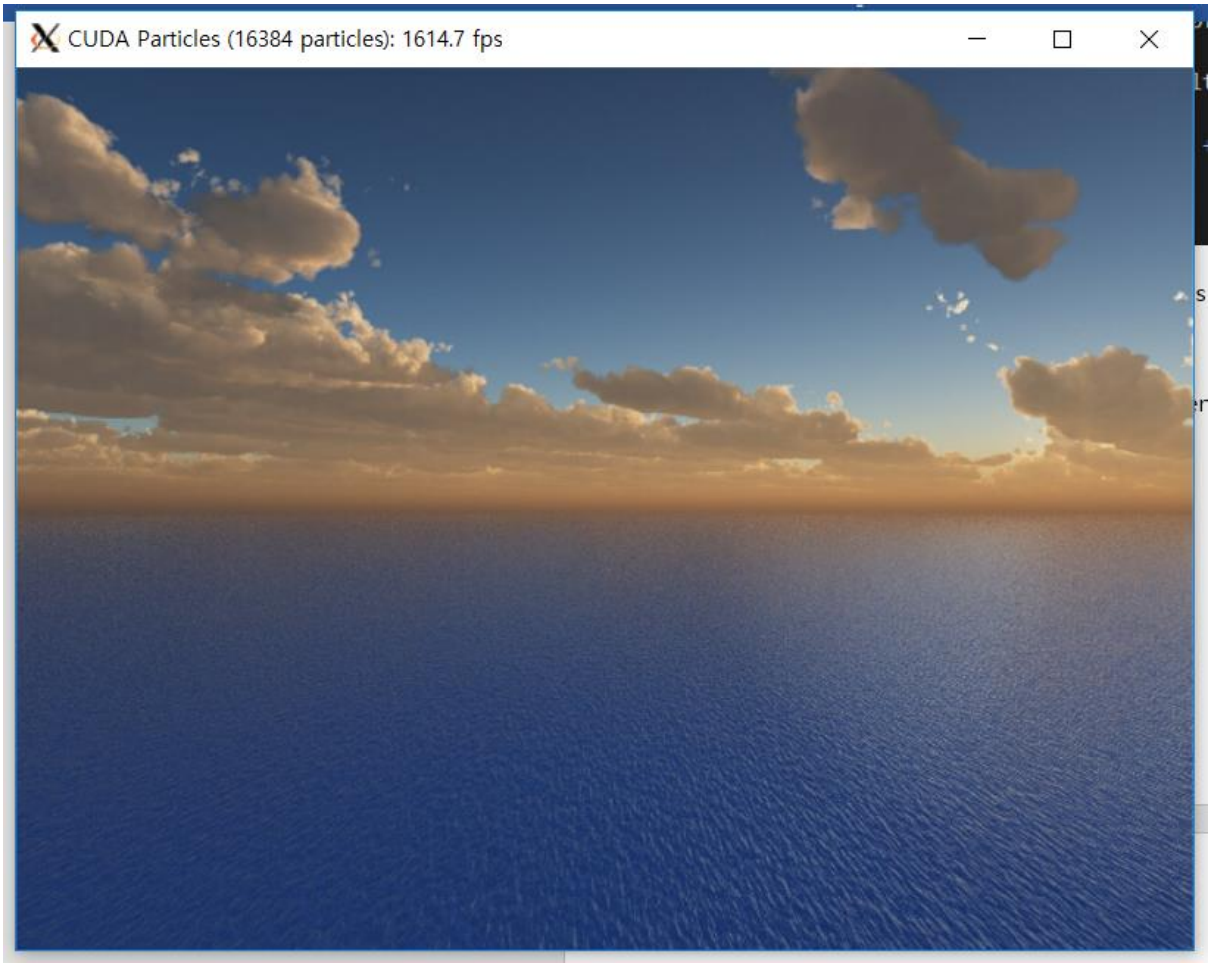
```
__host__ __device__
integrate_functor(float delta_time) : deltaTime(delta_time) {}

template <typename Tuple>
__device__
void operator()(Tuple t)
{
    /* 1. 위아래로 가는 모션 (video_topdown)
    volatile float4 posData = thrust::get<0>(t);
    volatile float4 velData = thrust::get<1>(t);
    float3 pos = make_float3(cos(40.0*posData.x), sin(40.0*posData.y), posData.z);
    float3 vel = make_float3(velData.x, velData.y, velData.z);

    vel += params.gravity * deltaTime;
    vel *= params.globalDamping;

    // new position = old position + velocity * deltaTime
    pos += vel * deltaTime;
    */

    /* 2. 중점으로 모이는 모션 (video_center)
    volatile float4 posData = thrust::get<0>(t);
    volatile float4 velData = thrust::get<1>(t);
    float3 pos = make_float3(posData.x, posData.y, posData.z);
    float3 vel = make_float3(velData.x, velData.y, velData.z);
    float3 center = make_float3(0.0f,0.0f,0.0f);

    vel += (center-pos) * 0.1 * deltaTime;

    // new position = old position + velocity * deltaTime
    pos += vel * deltaTime;
    */
```

I)위와 아래 두층으로 몰리는 모션 : pos값을 이용하여 삼각함수를 이용해 반복하여 위아래로 몰립니다.

II) 중점으로 모이며 충돌하는 모션 : center 값을 중점으로 초기화해 현재 위치와 중점간 거리를 계산하여 속도에 곱하였습니다.

2. 렌더링

스카이박스 맵핑




```
//function declaration
void ReadOBJ(const char* filename);
void initGL();
void createProgram();
char* ReadFile(char *filename);
GLuint createShader(char* src, GLenum type);
void ObjectDataTransfer();
void CubeMapTexture();
void SkyBoxDataTransfer();
void display();
void ExitProgram();
void cleanUp();
```

옆 사진과 같이 13_Environment 의 SampleCode를 참조하여 28_CUDA_Particle 의 particles.cpp에 가져와서 사용하였습니다. 오브젝트 관련 코드는 제거하였습니다.