

Parallel Programming

CS575

Chao Zhang

Project #7

1. Source listing

```
12 GLuint SphereList;  
13 GLuint SphereList2;  
14  
15  
16  
17  
18 SphereList2 = glGenLists( 1 );  
19 glNewList( SphereList2, GL_COMPILE );  
20     glColor3f( .9f, 0.f, 0.f );  
21     glPushMatrix( );  
22         glTranslatef( 1000., -500., 0. );  
23         glutWireSphere( 600., 100., 100. );  
24     glPopMatrix( );  
25 glEndList( );  
26
```

In the cpp file, I add one more sphere as SphereList2, I give the color and the translate of it.

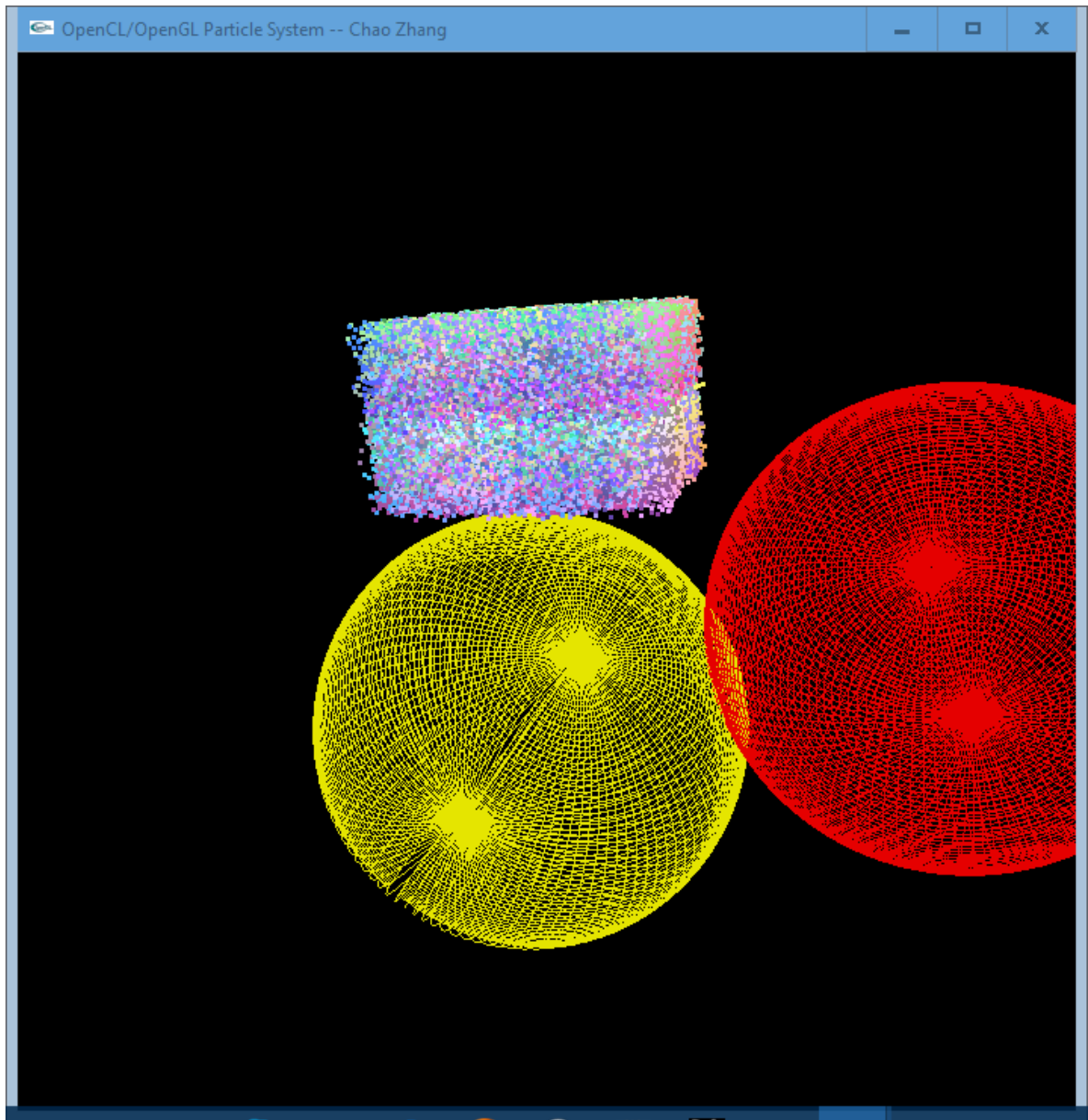
```
color c = dCobj[gid];  
color cp = c + (color)( -0.02*DT, -0.02*DT, 0.02*DT, 0. );
```

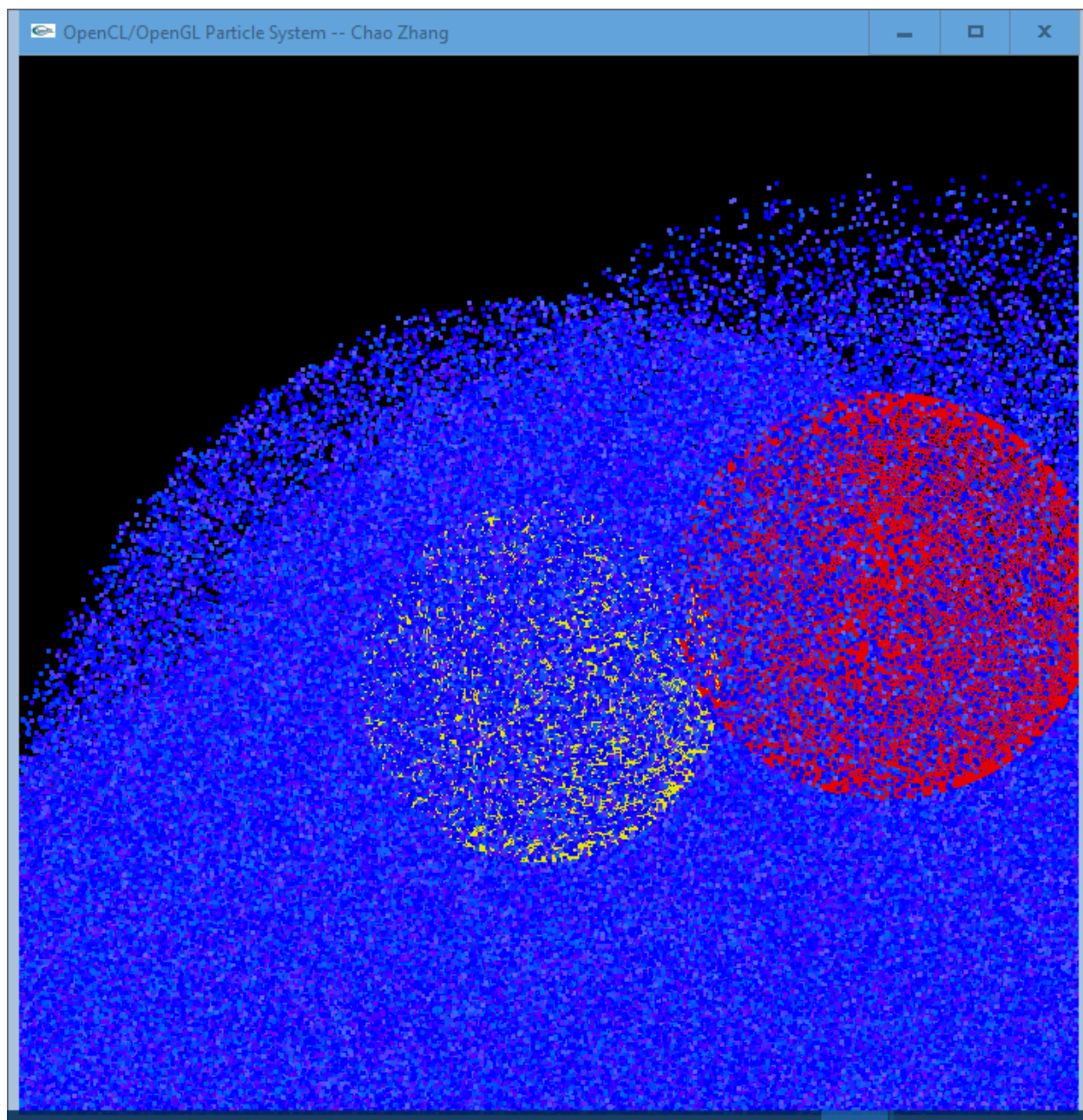
```
if( IsInsideSphere( pp, Sphere2 ) )  
{  
    vp = BounceSphere( p, v, Sphere2 );  
    pp = p + vp*DT + .5*DT*DT*G;  
}
```

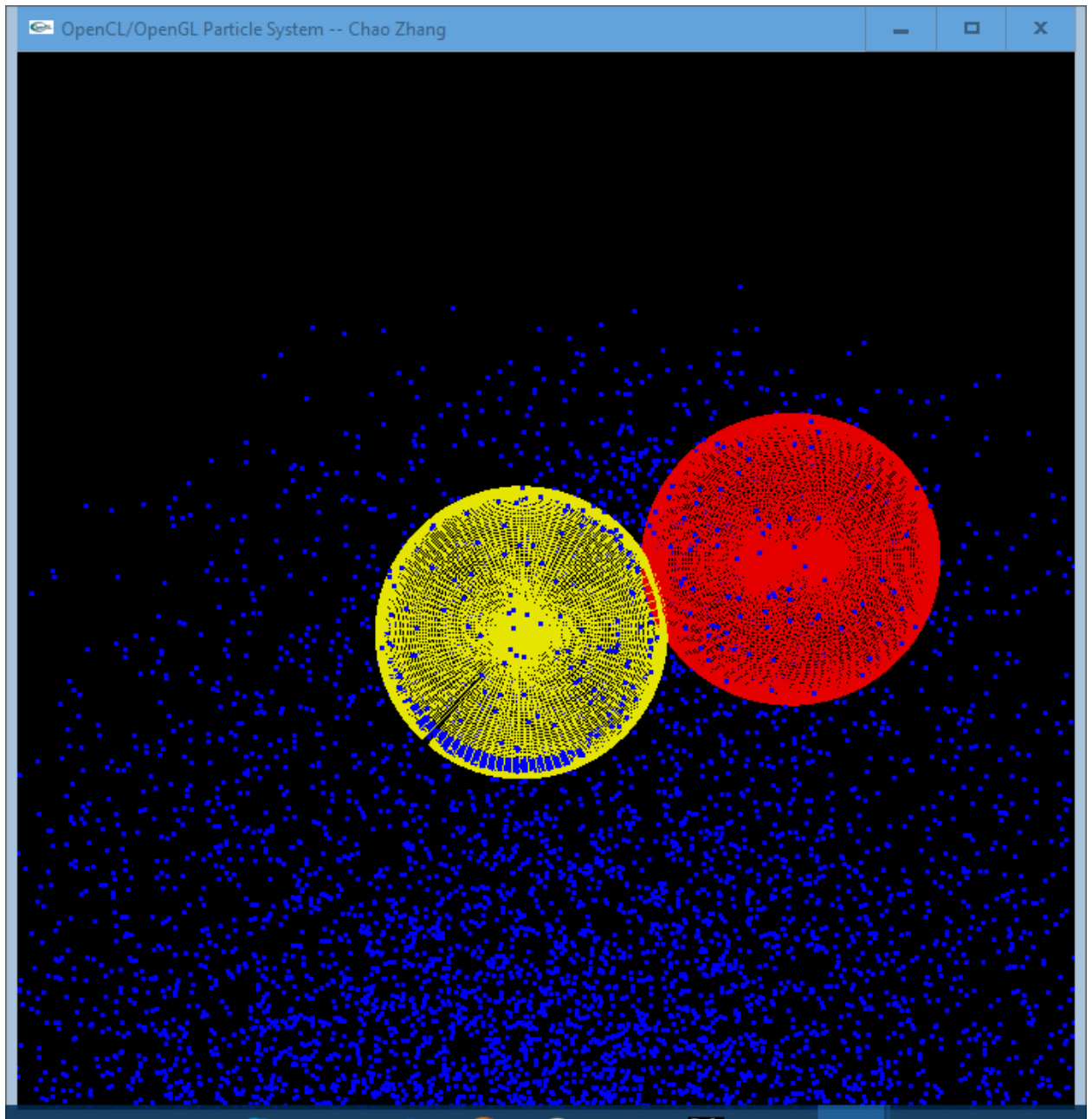
Those are added in the cl file, it will bounce when inside the sphere. The $cp = c + (color)(0.02*DT, -0.01*DT, -0.01*DT, 0.)$; is used to change the color during the time.

2. Result and analysis

I run this project on my own PC with visual studio. My PC has i3-3120m 2.5GHz CPU and Intel HD4000 GPU with 6G memory.

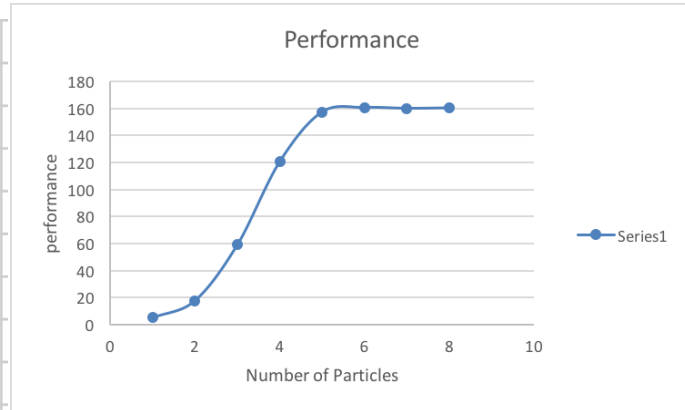






Picture of the Results

Performance	
1	4.994
2	17.657
3	58.936
4	120.508
5	157.247
6	160.476
7	159.861
8	160.174



Performance Table and Graph

We can get the performance increase with the increase of the number of particles until it reaches the limit. I think the reason is we can use the GPU parallel computing which can use the GPU and CPU together to do the calculate. The GPU and CPU has different advantages so use them both can take all the advantages to dealing with the data.