

昵称：白了少年头
园龄：4年11个月
粉丝：125
关注：13
[+加关注](#)

2010年10月						
日	一	二	三	四	五	六
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

搜索

常用链接

- [我的随笔](#)
- [我的评论](#)
- [我的参与](#)
- [最新评论](#)
- [我的标签](#)
- [更多链接](#)

随笔分类

- [数据库原理\(1\)](#)
- [08计算机软件\(10\)](#)
- [3d打印\(2\)](#)
- [android\(1\)](#)
- [C++\(13\)](#)
- [Directx\(1\)](#)
- [Java\(33\)](#)
- [Java3D\(1\)](#)
- [Jquery\(3\)](#)
- [linux\(7\)](#)
- [matlab\(1\)](#)
- [Oracle\(3\)](#)
- [VTK\(5\)](#)
- [电信crm\(5\)](#)
- [分布式数据库](#)
- [服务器\(1\)](#)
- [计算机理论\(4\)](#)
- [计算机软件工程\(1\)](#)
- [计算机史](#)
- [生活与编程\(3\)](#)
- [体绘制\(2\)](#)
- [体绘制 \(volume render\)](#)
- [图形学\(26\)](#)
- [未来的程序员](#)

随笔档案

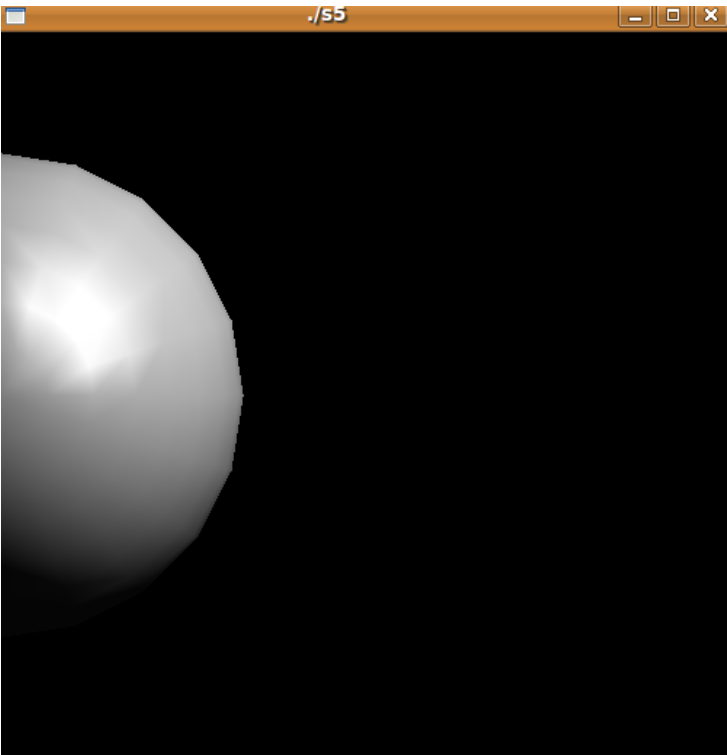
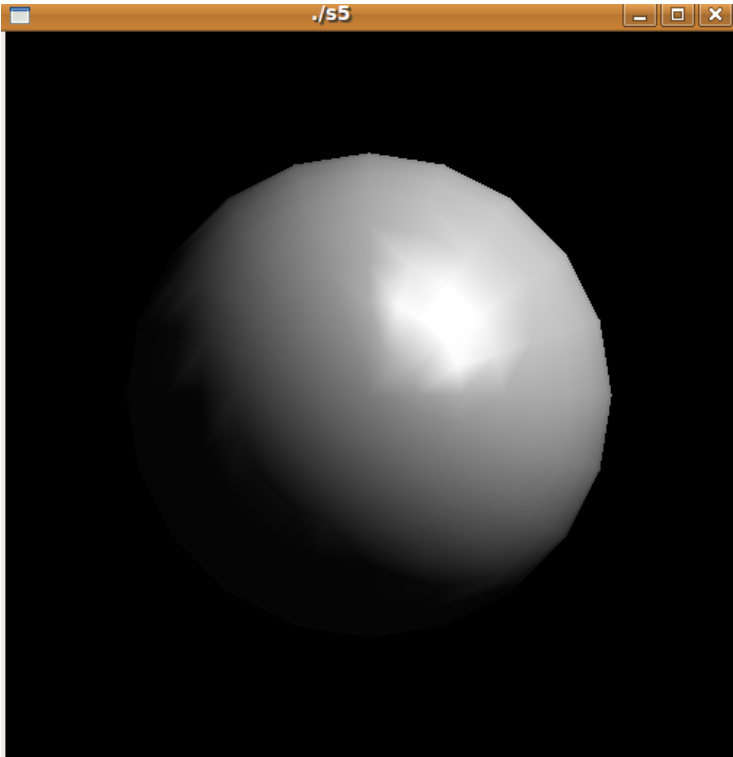
- [2015年5月 \(2\)](#)
- [2015年4月 \(2\)](#)
- [2015年2月 \(1\)](#)
- [2015年1月 \(2\)](#)
- [2014年11月 \(2\)](#)
- [2014年10月 \(1\)](#)
- [2014年7月 \(2\)](#)
- [2014年6月 \(5\)](#)
- [2014年5月 \(7\)](#)
- [2014年4月 \(1\)](#)
- [2014年1月 \(2\)](#)
- [2013年12月 \(1\)](#)
- [2013年1月 \(1\)](#)
- [2012年10月 \(2\)](#)
- [2012年9月 \(2\)](#)
- [2012年8月 \(9\)](#)
- [2012年6月 \(1\)](#)
- [2012年5月 \(2\)](#)
- [2012年4月 \(3\)](#)

glViewport()函数和glOrtho()函数的理解(转)

在OpenGL中有两个比较重要的投影变换函数，glViewport和glOrtho。

glOrtho是创建一个正交平行的视景体。一般用于物体不会因为离屏幕的远近而产生大小的变换的情况。比如，常用的工程中的制图等。需要比较精确的显示。而作为它的对立情况，glFrustum则产生一个透视投影。这是一种模拟真是生活中，人们视野观测物体的真实情况。例如：观察两条平行的火车到，在过了很远之后，这两条铁轨是会相交于一处的。还有，离眼睛近的物体看起来大一些，远的物体看起来小一些。

glOrtho(left, right, bottom, top, near, far), left表示视景体左面的坐标，right表示右面的坐标，bottom表示下面的，top表示上面的。这个函数简单理解起来，就是一个物体摆在那里，你怎么去截取他。这里，我们先抛开glViewport函数不看。先单独理解glOrtho的功能。假设有一个球体，半径为1，圆心在(0, 0, 0)，那么，我们设定glOrtho(-1.5, 1.5, -1.5, 1.5, -10, 10);就表示用一个宽高都是3的框框把这个球体整个都装了进来。如果设定glOrtho(0.0, 1.5, -1.5, 1.5, -10, 10);就表示用一个宽是1.5，高是3的框框把整个球体的右面装进来;如果设定glOrtho(0.0, 1.5, 0.0, 1.5, -10, 10);就表示用一个宽和高都是1.5的框框把球体的右上角装了进来。上述三种情况可以见图：



- 2012年3月 (1)
- 2012年2月 (6)
- 2011年12月 (1)
- 2011年11月 (9)
- 2011年8月 (2)
- 2011年6月 (1)
- 2011年4月 (4)
- 2011年3月 (10)
- 2011年2月 (5)
- 2011年1月 (1)
- 2010年12月 (1)
- 2010年11月 (13)
- 2010年10月 (16)
- 2010年9月 (13)

文章分类

数学(3)

相册

JOGL(1)
opengl(1)

最新评论

1. Re:JAVA深复制(深克隆)与浅复制(浅克隆)

写得很清楚!

--程序驾

2. Re:短暂的计算机职业生涯

楼主加油~~~

--Gru--

3. Re:JAVA深复制(深克隆)与浅复制(浅克隆)

@金良酒庄恩, 是的, 所以我说他的例子中有一处是错误的, 如下: class Profess or { String name; int age; Profess.....

--beijishiqidu

4. Re:JAVA深复制(深克隆)与浅复制(浅克隆)

@beijishiqidu我试了下, 不用Cloneable接口是不能用clone方法的。...

--金良酒庄

5. Re:glViewport()函数和glOrtho()函数的理解(转)

你好想问下这个球体的原点时如何设置的?

--ASunDay

阅读排行榜

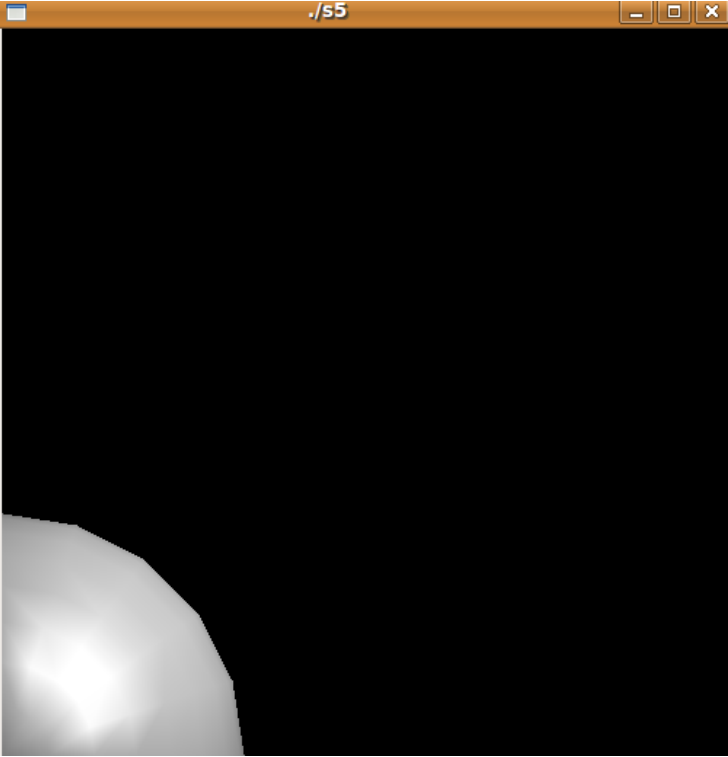
- 1. std::vector介绍(40207)
- 2. Matlab曲线拟合工具箱(37182)
- 3. JAVA深复制(深克隆)与浅复制(浅克隆)(24342)
- 4. glViewport()函数和glOrtho()函数的理解(转)(24090)
- 5. windows系统下JDK1.6环境变量配置(17271)

评论排行榜

- 1. JAVA深复制(深克隆)与浅复制(浅克隆)(9)
- 2. glViewport()函数和glOrtho()函数的理解(转)(9)
- 3. OpenGL总结(9)
- 4. 短暂的计算机职业生涯(2)
- 5. 使用Eclipse RCP进行桌面程序开发(2)

推荐排行榜

- 1. glViewport()函数和glOrtho()函数的理解(转)(12)
- 2. OpenGL总结(8)
- 3. windows系统下JDK1.6环境变量配置(3)
- 4. JAVA深复制(深克隆)与浅复制(浅克隆)(3)
- 5. Java中的equals()(3)



从上述三种情况, 我们可以大致了解glOrtho函数的用法。glOrtho函数只是负责使用什么样的视景体来截取图像, 并不负责使用某种规则把图像呈现在屏幕上。

glViewport主要完成这样的功能。它负责把视景体截取的图像按照怎样的高和宽显示到屏幕上。

比如: 如果我们使用glut库建立一个窗体:glutInitWindowSize(500, 500); 然后使用glutReshapeFunc(reshape); reshape代码如下:

```
void reshape(int width, int height)
{
    glViewport(0, 0, (GLsizei)width, (GLsizei)height);

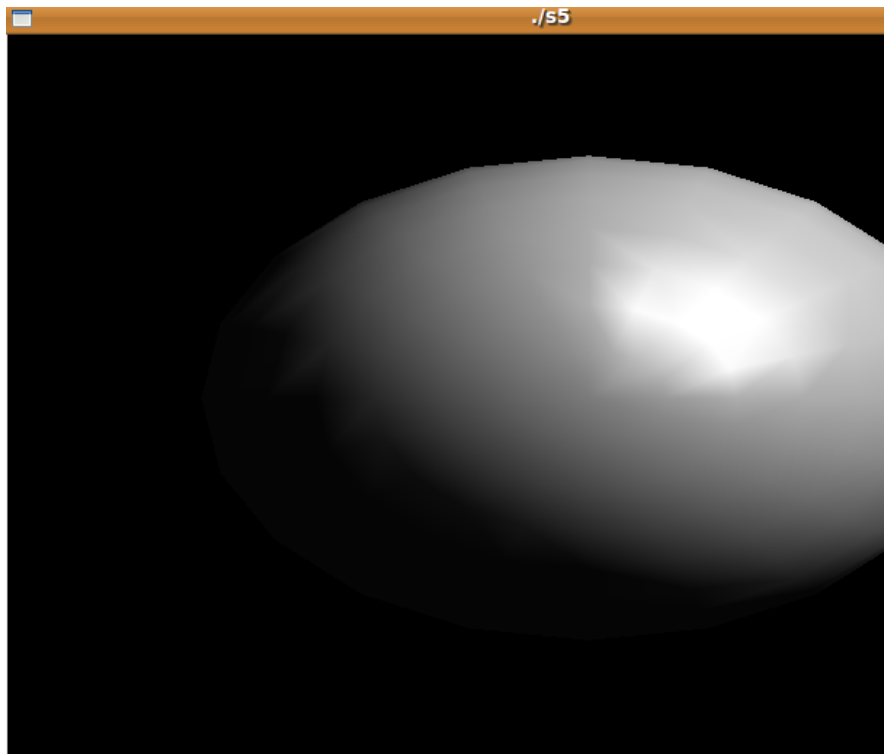
    glMatrixMode(GL_PROJECTION);

    glLoadIdentity();

    glOrtho(-1.5, 1.5, -1.5, 1.5, -10, 10);

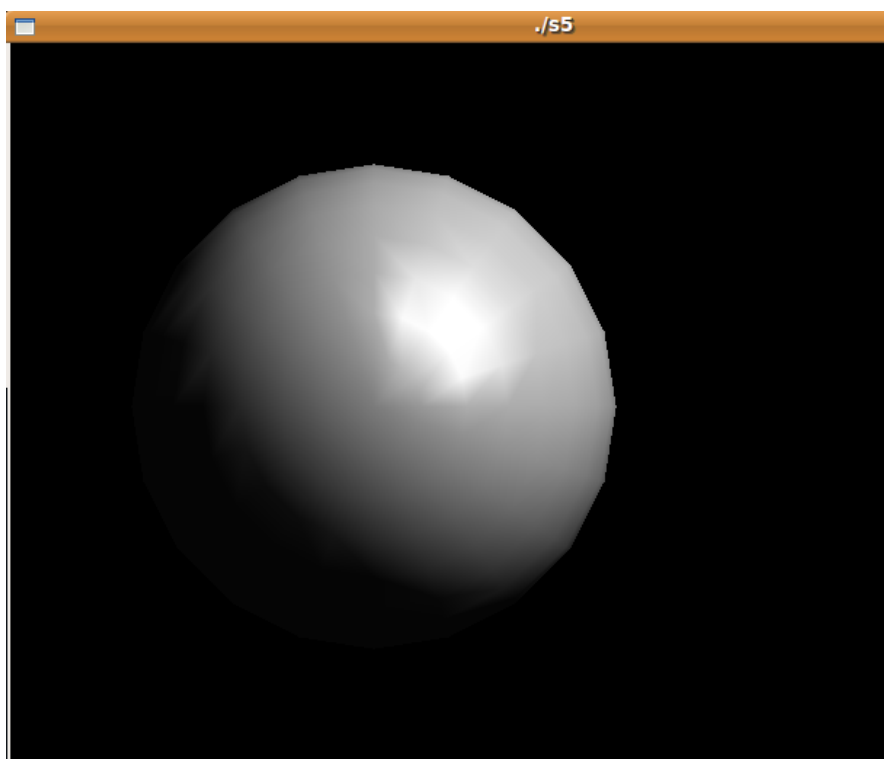
    ....
}
```

这样是可以看到一个正常的球体的。但是, 如果我们创建窗体时glutInitWindowSize(800, 500),那么看到的图像就是变形的。上述情况见图。



因为我们是用一个正方形截面的视景体截取的图像，但是拉伸到屏幕上显示的时候，就变成了 `glViewport(0, 0, 800, 500)`;也就是显示屏变宽了，倒是显示的时候把一个正方形的图像“活生生的给拉宽了”。就会产生变形。这样，就需要我们调整我们的OpenGL显示屏了。我们可以不用800那么宽，因为我们用的是正方形的视景体，所以虽然窗体是800宽，但是我们只用其中的500就够了。修改一下程序。

```
void reshape(int width, int height)
{
    int dis = width < height ? width : height;
    glViewport(0, 0, dis, dis); /*这里dis应该是500*/
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-1.5, 1.5, -1.5, 1.5, -10, 10);
    .....
}
```



OK。如果你能看明白我写的内容。你可能对glViewport函数有个大致的了解。

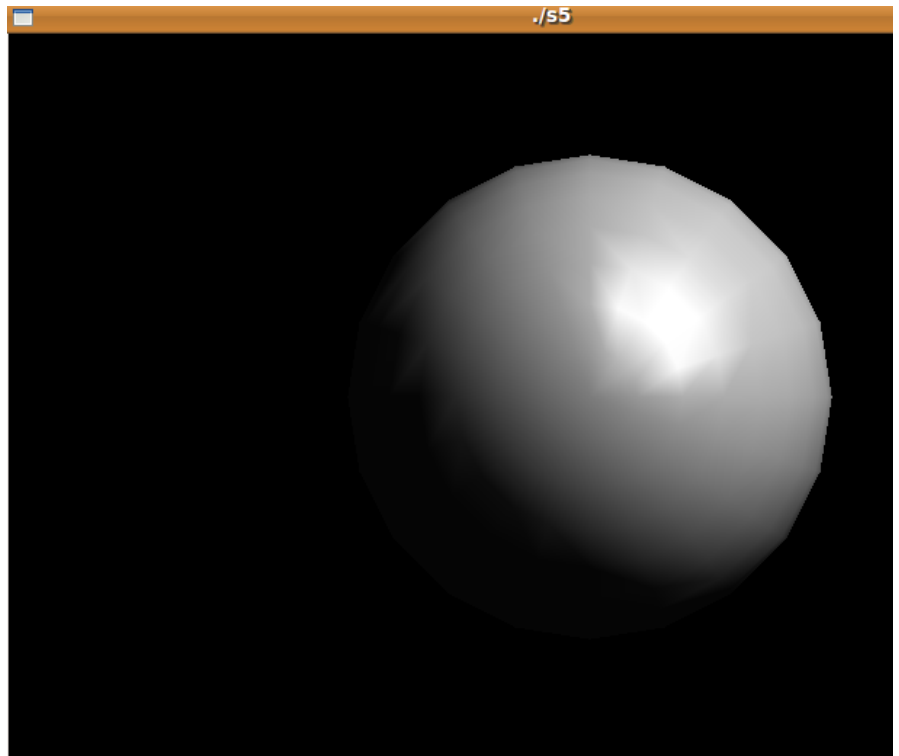
不过，我们采用上面的办法，就是只使用了原来屏幕的一部分（宽度从501到800我们没有用来显示图像）。如果我们想用整个OpenGL屏幕显示图像，但是又不使图像变形怎么办？

那就只能修改glOrtho函数了。也就是说，我们使用一个和窗体一样比例的视景体（而不再是正方形的视景体）来截取图像。例如，对于(800, 500)的窗体，我们使用glOrtho(-1.5 * 800/500, 1.5 * 800/500, -1.5, 1.5, -10, 10)，就是截取的时候，我们就使用一个“扁扁”的视景体截取，那么，显示的到OpenGL屏幕时(800, 500)，我们只要正常把这个扁扁的截取图像显示（扁扁的截取图像是指整个截取的图像，包括球形四周的黑色部分。球形还是正常圆形的），就可以了。如：

```
void reshape(int width , int height)
{
    glViewport(width, height); //按照窗体大小制作OpenGL屏幕

    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();

    if (width <= height)
        glOrtho(-1.5, 1.5, -1.5 * (GLfloat)height/(GLfloat)width, 1.5 * (GLfloat)height/(GLfloat)width, -10.0, 10.0);
    else
        glOrtho(-1.5*(GLfloat)width/(GLfloat)height, 1.5*(GLfloat)w/(GLfloat)h, -1.5, 1.5, -10.0, 10.0);
    ....
}
```



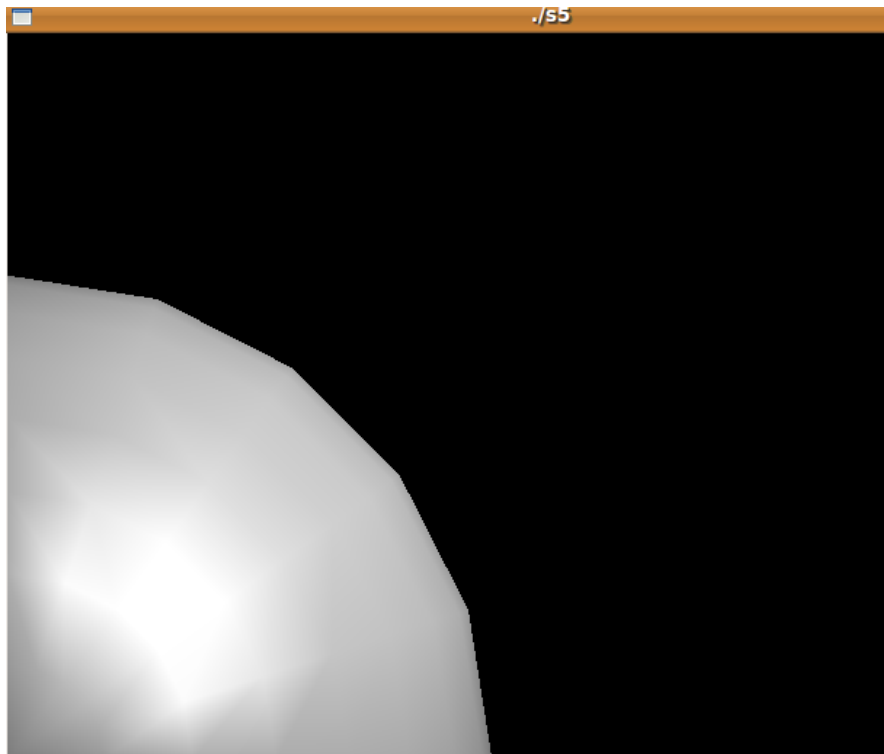
另外，关于glViewport()函数，我们还可以用来调整图像的分辨率。例如，保持目前的窗体大小不变，我们如果用这个size来只显示整个物体的一部分，那么图像的分辨率就必然会增大。例如：

```
void reshape(int w, int h)
{
    glViewport(0, 0, (GLsizei)w, (GLsizei)h);

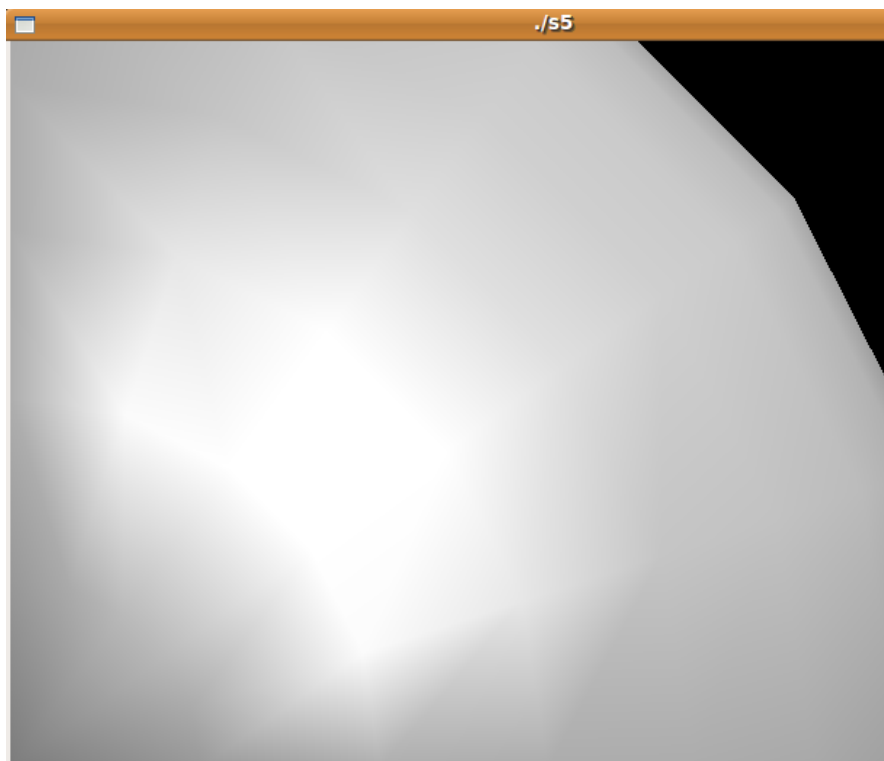
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();

    if (w <= h)
        glOrtho(0, 1.5, 0, 1.5 * (GLfloat)h/(GLfloat)w, -10.0, 10.0);
    else
        glOrtho(0, 1.5*(GLfloat)w/(GLfloat)h, 0, 1.5, -10.0, 10.0);
}
```

可以把分辨率扩大4倍。



而如果再修改一下glViewport(0, 0, 2 * (GLsizei)w, 2 * (GLsizei)h); 则可以把分辨率扩大16倍。



完整的测试程序：

```
/*Build on ubuntu 9.04*/
#include <GL/gl.h>
#include <GL/glu.h>
#include <GL/glut.h>
void init(void)
{
    GLfloat mat_specular[] = {1.0, 1.0, 1.0, 1.0};
    GLfloat mat_shininess[] = {50.0};
    GLfloat light_position[] = {1.0, 1.0f, 1.0, 0.0};
    GLfloat white_light[] = {1.0, 1.0, 1.0, 1.0};
    GLfloat lmodel_ambient[] = {0.1, 0.1, 0.1, 1.0};
```

```

glClearColor(0.0, 0.0, 0.0, 0.0);

glShadeModel(GL_SMOOTH);

glMaterialfv(GL_FRONT, GL_SPECULAR, mat_specular);

glMaterialfv(GL_FRONT, GL_SHININESS, mat_shininess);

glLightfv(GL_LIGHT0, GL_POSITION, light_position);

glLightfv(GL_LIGHT0, GL_DIFFUSE, white_light);

glLightfv(GL_LIGHT0, GL_SPECULAR, white_light);

glLightModelfv(GL_LIGHT_MODEL_AMBIENT, lmodel_ambient);

glEnable(GL_LIGHTING);

glEnable(GL_LIGHT0);

glEnable(GL_DEPTH_TEST);

}

void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    glutSolidSphere(1.0, 20, 16);

    glFlush();
}

void reshape(int w, int h)
{
    glViewport(0, 0, (GLsizei)w, (GLsizei)h);

    glMatrixMode(GL_PROJECTION);

    glLoadIdentity();

    if (w <= h)
        glOrtho(-1.5, 1.5, -1.5 * (GLfloat)h/(GLfloat)w, 1.5 * (GLfloat)h/(GLfloat)w, -10.0, 10.0);
    else
        glOrtho(-1.5*(GLfloat)w/(GLfloat)h, 1.5*(GLfloat)w/(GLfloat)h, -1.5, 1.5, -10.0, 10.0);

    glMatrixMode(GL_MODELVIEW);

    glLoadIdentity();
}

int main(int argc, char **argv)
{
    glutInit(&argc, argv);

    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB | GLUT_DEPTH);

    glutInitWindowSize(500, 500);

    glutInitWindowPosition(100, 100);

    glutCreateWindow(argv[0]);

    init();

    glutDisplayFunc(display);

    glutReshapeFunc(reshape);

    glutMainLoop();

    return 0;
}

/*CMakeLists.txt*/

PROJECT(s5)

CMAKE_MINIMUM_REQUIRED(VERSION 2.6)

ADD_EXECUTABLE(s5 main.cpp)

FIND_PACKAGE(OpenGL)

FIND_PACKAGE(GLUT)

```

```
IF(OPENGLE_FOUND)
    INCLUDE_DIRECTORIES(${OPENGLE_INCLUDE_DIR})
    TARGET_LINK_LIBRARIES(${PROJECT_NAME} ${OPENGLE_LIBRARIES})
ELSE(OPENGLE_FOUND)
    MESSAGE(FATAL_ERROR "OpenGL not found")
ENDIF(OPENGLE_FOUND)

IF(GLUT_FOUND)
    INCLUDE_DIRECTORIES(${GLUT_INCLUDE_DIR})
    TARGET_LINK_LIBRARIES(${PROJECT_NAME} ${GLUT_LIBRARIES})
ELSE(GLUT_FOUND)
    MESSAGE(FATAL_ERROR "GLUT not found")
ENDIF(GLUT_FOUND)
```

分类: [图形学](#)

绿色通道:

好文要顶

关注我

收藏该文

与我联系

[白了少年头](#)
[关注 - 13](#)
[粉丝 - 125](#)
[+加关注](#)

120

(请您对文章做出评价)

« 上一篇: [OpenGL基础技术讲座--体系结构](#)
» 下一篇: [OpenGL数据处理过程](#)

posted @ 2010-10-30 21:57 白了少年头 阅读(24090) 评论(9) 编辑 收藏

评论列表

- #1楼 2011-11-01 11:08 lihui_yy
写的很详细~~
支持(0) 反对(0)
- #2楼 2011-12-12 17:02 guoning
博主 有个问题想请教, glOrtho(-1.5, 1.5, -1.5 * (GLfloat)h/(GLfloat)w, 1.5 * (GLfloat)h/(GLfloat)w, -10.0, 10.0);这个1.5这个值是怎么来得?
支持(0) 反对(0)
- #3楼[楼主] 2011-12-12 17:27 白了少年头
@guoning
是自己设定的, 球的半径是1, 要把球能够完全显示出来的话, 那就要屏幕大于1, 然后取个数1.5, 其实设置1.6, 1.7, 等都可以, 你可以试下。
支持(1) 反对(0)
- #4楼 2011-12-12 18:19 guoning
@白了少年头
多谢解答, 刚开始学opengl, 希望博主可以多多指点。
支持(0) 反对(0)
- #5楼 2012-10-09 05:59 ricekeller
很好理解, 通俗易懂, 而且把要点都解释的很清楚。赞~!
支持(0) 反对(0)
- #6楼 2014-02-27 11:26 Sure::D
写的很清楚!
支持(0) 反对(0)
- #7楼 2014-03-09 20:49 dahongdahong
受益匪浅, 谢谢~~~
支持(0) 反对(0)
- #8楼 2014-03-14 19:55 唐小妹
你好, 我也试了这个函数, 但是我的程序像是glOrtho函数不起作用一样怎么样都不随着窗口变化变成正