**计算机图形学实验**

**姓 名：白文强**

**学 号：20191060064**

**专 业：计算机科学与技术**

**教 师：钱文华**

实验八 建模实验

时间：2020年12月11日

地点：信息学院2202机房

1. 实验内容

使用opengl，教材P541，颜色编码建模显示。

2、实验目的

调用函数完成颜色编码建模实验。

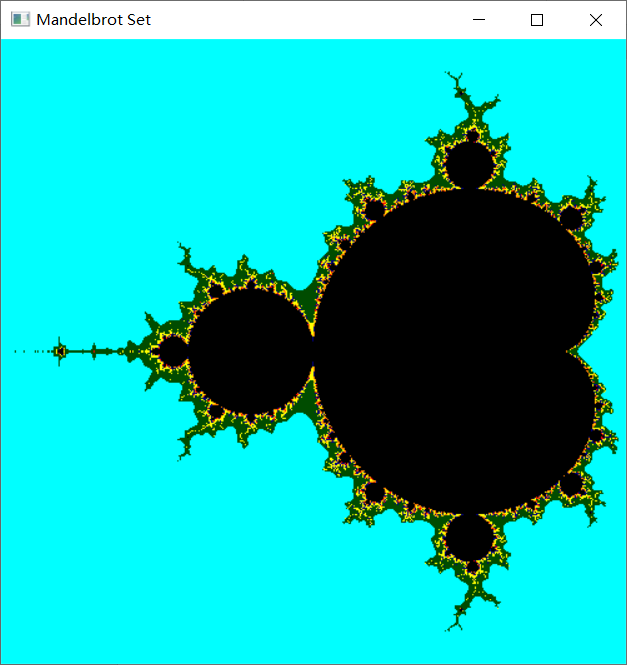
3、实验代码

1. #include <windows.h>
2. #include <GL/glut.h>
4. /\*Set initial size of the display window\*/
5. GLsizei winWidth = 500, winHeight = 500;
7. GLfloat xComplexMin = -2.00, xComplexMax = 0.50;
8. GLfloat yComplexMin = -1.25, yComplexMax = 1.25;
10. GLfloat complexWidth = xComplexMax - xComplexMin;
11. GLfloat complexHeight = yComplexMax - yComplexMin;
13. **class** complexNum
14. {
15. **public**:
16. GLfloat x, y;
17. };
18. **struct** color
19. {
20. GLfloat r, g, b;
21. };
22. **void** init(**void**)
23. {
24. glClearColor(1.0, 1.0, 1.0, 0.0);
25. }
26. **void** plotPoint(complexNum z)
27. {
28. glBegin(GL\_POINTS);
29. glVertex2f(z.x, z.y);
30. glEnd();
31. }

34. complexNum complexSquare(complexNum z)
35. {
36. complexNum zSquare;
37. zSquare.x = z.x \* z.x - z.y \* z.y;
38. zSquare.y = 2 \* z.x \* z.y;
39. **return** zSquare;
40. }

43. GLint mandelSqTransf(complexNum z0, GLint maxIter)
44. {
45. complexNum z = z0;
46. GLint count = 0;
47. **while**((z.x \* z.x + z.y \* z.y <= 4.0) && (count < maxIter))
48. {
49. z = complexSquare(z);
50. z.x += z0.x;
51. z.y += z0.y;
52. count++;
53. }
54. **return** count;
55. }
56. **void** mandelbrot(GLint nx, GLint ny, GLint maxIter)
57. {
58. complexNum z, zIncr;
59. color ptColor;
60. GLint iterCount;
61. zIncr.x = complexWidth / GLfloat(nx);
62. zIncr.y = complexHeight / GLfloat(ny);
63. **for**(z.x = xComplexMin ; z.x < xComplexMax; z.x += zIncr.x)
64. {
65. **for**(z.y = yComplexMin ; z.y < yComplexMax; z.y += zIncr.y)
66. {
67. iterCount = mandelSqTransf(z, maxIter);
68. **if**(iterCount >= maxIter)
69. ptColor.r = ptColor.g = ptColor.b = 0.0;
70. **else** **if**(iterCount > maxIter / 8)
71. {
72. ptColor.r = 1.0;
73. ptColor.g = 0.5;
74. ptColor.b = 0.0;
75. }
76. **else** **if**(iterCount > (maxIter / 10))
77. {
78. ptColor.r = 1.0;
79. ptColor.g = ptColor.b = 0.0;
80. }
81. **else** **if**(iterCount > (maxIter / 20))
82. {
83. ptColor.b = 0.5;
84. ptColor.r = ptColor.g = 0.0;
85. }
86. **else** **if**(iterCount > (maxIter / 40))
87. {
88. ptColor.r = ptColor.g = 1.0;
89. ptColor.b = 0.0;
90. }
91. **else** **if**(iterCount > (maxIter / 100))
92. {
93. ptColor.r  = ptColor.b = 0.0;
94. ptColor.g = 0.3;
95. }
96. **else**
97. {
98. ptColor.r = 0.0;
99. ptColor.g = ptColor.b = 1.0;
100. }
101. glColor3f(ptColor.r, ptColor.g, ptColor.b);
102. plotPoint(z);
103. }
104. }
105. }
106. **void** displayFcn(**void**)
107. {
108. GLint nx = 1000, ny = 1000, maxIter = 1000;
109. glClear(GL\_COLOR\_BUFFER\_BIT);   // Clear display window.
110. mandelbrot(nx, ny, maxIter);
111. glFlush();
112. }
113. **void** winReshapeFcn(GLint newWidth, GLint newHeight)
114. {
115. glViewport(0, 0, newHeight, newHeight);
116. glMatrixMode(GL\_PROJECTION);
117. glLoadIdentity();
118. gluOrtho2D(xComplexMin, xComplexMax, yComplexMin, yComplexMax);
119. glClear(GL\_COLOR\_BUFFER\_BIT);
120. }
121. **int** main(**int** argc, **char**\*\*argv)
122. {
123. glutInit(&argc, argv);
124. glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);
125. glutInitWindowPosition(50, 50);
126. glutInitWindowSize(winWidth, winHeight);
127. glutCreateWindow("Mandelbrot Set");
128. init();
129. glutDisplayFunc(displayFcn);
130. glutReshapeFunc(winReshapeFcn);
131. glutMainLoop();
132. **return** 0;
133. }

4、实验结果



5、实验总结

通过颜色编码，可以在复平面上选择更小的矩形区域并放大所选区域的图示。

旋转后缩放

旋转后

原图