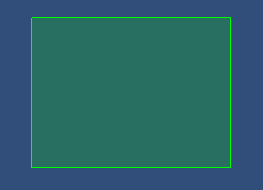
Unity3D使用的图形引擎是DirectX，OpenGL和自带的APi（Wii）

我们这里使用OpenGL的渲染方式



代码：

（使用过OpenGl的应该相对容易理解

另外：代码中使用Shader是因为矩形框中部的透明部分需要）

**[csharp]** [view plaincopy](http://blog.csdn.net/xv_ly15/article/details/9047509)

1. **using** UnityEngine;
2. **using** System.Collections;
4. **public** **class** DrawRect : MonoBehaviour {
6. **private** Vector2 mMouseStart, mMouseEnd;
7. **private** **bool** mBDrawMouseRect;
9. **private** Material rectMat = **null**;//画线的材质 不设定系统会用当前材质画线 结果不可控
11. **void** Start()
12. {
14. mBDrawMouseRect = **false**;
16. rectMat = **new** Material("Shader \"Lines/Colored Blended\" {" +
17. "SubShader { Pass { " +
18. "    Blend SrcAlpha OneMinusSrcAlpha " +
19. "    ZWrite Off Cull Off Fog { Mode Off } " +
20. "    BindChannels {" +
21. "      Bind \"vertex\", vertex Bind \"color\", color }" +
22. "} } }");//生成画线的材质
23. rectMat.hideFlags = HideFlags.HideAndDontSave;
24. rectMat.shader.hideFlags = HideFlags.HideAndDontSave;
25. }
27. **void** Update()
28. {
29. **if** (Input.GetMouseButtonDown(0))
30. //按下鼠标左键
31. {
32. Vector3 mousePosition = Input.mousePosition;
33. mMouseStart = **new** Vector2(mousePosition.x, mousePosition.y);
34. }
36. **if** (Input.GetMouseButton(0))
37. //持续按下鼠标左键
38. {
39. mBDrawMouseRect = **true**;
40. Vector3 mousePosition = Input.mousePosition;
41. mMouseEnd = **new** Vector2(mousePosition.x, mousePosition.y);
42. }
44. **if** (Input.GetMouseButtonUp(0))
45. {
46. mBDrawMouseRect = **false**;
47. }
48. }
50. **void** OnGUI()
51. {
52. **if** (mBDrawMouseRect)
53. Draw(mMouseStart, mMouseEnd);
54. }
56. //渲染2D框
57. **void** Draw(Vector2 start, Vector2 end)
58. {
59. rectMat.SetPass(0);
61. GL.PushMatrix();//保存摄像机变换矩阵
63. Color clr = Color.green;
64. clr.a = 0.1f;
66. GL.LoadPixelMatrix();//设置用屏幕坐标绘图
67. //透明框
68. GL.Begin(GL.QUADS);
69. GL.Color(clr);
70. GL.Vertex3(start.x, start.y, 0);
71. GL.Vertex3(end.x, start.y, 0);
72. GL.Vertex3(end.x, end.y, 0);
73. GL.Vertex3(start.x, end.y, 0);
74. GL.End();
76. //线
77. //上
78. GL.Begin(GL.LINES);
79. GL.Color(Color.green);
80. GL.Vertex3(start.x, start.y, 0);
81. GL.Vertex3(end.x, start.y, 0);
82. GL.End();
84. //下
85. GL.Begin(GL.LINES);
86. GL.Color(Color.green);
87. GL.Vertex3(start.x, end.y, 0);
88. GL.Vertex3(end.x, end.y, 0);
89. GL.End();
91. //左
92. GL.Begin(GL.LINES);
93. GL.Color(Color.green);
94. GL.Vertex3(start.x, start.y, 0);
95. GL.Vertex3(start.x, end.y, 0);
96. GL.End();
98. //右
99. GL.Begin(GL.LINES);
100. GL.Color(Color.green);
101. GL.Vertex3(end.x, start.y, 0);
102. GL.Vertex3(end.x, end.y, 0);
103. GL.End();
105. GL.PopMatrix();//还原
106. }
107. }

http://blog.csdn.net/xv\_ly15/article/details/9047509