**Unity3D教程：物体旋转和缩放的简单demo**

Posted on 2013年06月13日 by U3d / [Unity3D 基础教程](http://www.unitymanual.com/category/manual/unity3d-%e5%9f%ba%e7%a1%80%e6%95%99%e7%a8%8b)/被围观 54 次

一.旋转:

旋转有很多种函数，选择的是RotateAround函数；

Input.GetAxis(“Mouse X”)又正左负，Input.GetAxis(“Mouse Y”)上正下负，这两个函数你可以感受一下，他的大小会随着你的手势（或者鼠标）方向和速度改变的。

用这两个函数就能做出手势控制物体的旋转了，重点：你自己要体验一下Input.GetAxis(“Mouse X/Y”)的用法，然后就是注意一下旋转方向；

二.缩放：

缩放手势是用两个手指来模拟的，当两手之间的距离越来越大，那就是放大，反之缩小；

放大的方法有两种：

1.物体用localScale来放大物体。不知道为什么我用这种放大得不到我想要的效果；

2.拉近摄像机，这里面可就有一点学问了，我记得我在看OpenGL的时候，上面用铁轨来比喻一个视觉效果，实际当中铁轨是平行的不相交，但是我们视觉上的效果就是铁轨间的距离越来越近，最后相交了。Unity3D默认的是相交的效果（camera-->Projection-->perspective），当然也有不想交的效果了（camera-->Projection-->orthographic）。如果你想加入天空盒而且不想随着物体缩放，看到天空背景也移动的话，那就选第二种。我加了天空盒，所以用第二种了。Unity3D教程手册

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 001 | **var** rotatepos:Transform;*//rotatepos是该物体的一个子物体，放到物体的中心地方就行。* |
| 002 |  |
| 003 | **var** horizontalSpeed : **float** ; |
| 004 |  |
| 005 | **var** verticalSpeed : **float** ; |
| 006 |  |
| 007 | **var** flag:boolean; |
| 008 |  |
| 009 | **var** olddis:**float**=0; |
| 010 |  |
| 011 | **var** newdis:**float**=0; |
| 012 |  |
| 013 | **var** child:Transform; |
| 014 |  |
| 015 | function Start(){ |
| 016 |  |
| 017 | horizontalSpeed=0; |
| 018 |  |
| 019 | verticalSpeed=0; |
| 020 |  |
| 021 | flag=**true**; |
| 022 |  |
| 023 | } |
| 024 |  |
| 025 | function Update () { |
| 026 |  |
| 027 | **if**(flag){ |
| 028 |  |
| 029 | transform.RotateAround(rotatepos.position,Vector3.up,1); |
| 030 |  |
| 031 | }Unity3D教程手册 |
| 032 |  |
| 033 | *//物体的旋转* |
| 034 |  |
| 035 | **if**(Input.touchCount==1){ |
| 036 |  |
| 037 | **if**(Input.GetTouch(0).phase==TouchPhase.Moved){ |
| 038 |  |
| 039 | flag=**false**; |
| 040 |  |
| 041 | **var** h : **float**=Input.GetAxis(“Mouse X”);*//又正左负* |
| 042 |  |
| 043 | **var** v : **float** =Input.GetAxis(“Mouse Y”);*//上正下负* |
| 044 |  |
| 045 | **if**(Mathf.Abs(h)>=Mathf.Abs(v)){ |
| 046 |  |
| 047 | **if**(h<0){ |
| 048 |  |
| 049 | horizontalSpeed=6; |
| 050 |  |
| 051 | transform.RotateAround(rotatepos.position,Vector3.up,horizontalSpeed); |
| 052 |  |
| 053 | } |
| 054 |  |
| 055 | **if**(h>0){ |
| 056 |  |
| 057 | horizontalSpeed=6; |
| 058 |  |
| 059 | transform.RotateAround(rotatepos.position,-Vector3.up,horizontalSpeed); |
| 060 |  |
| 061 | } |
| 062 |  |
| 063 | } |
| 064 |  |
| 065 | **else**{ |
| 066 |  |
| 067 | **if**(v<0){ |
| 068 |  |
| 069 | verticalSpeed=6; |
| 070 |  |
| 071 | transform.RotateAround(rotatepos.position,-Vector3.right,verticalSpeed); |
| 072 |  |
| 073 | } |
| 074 |  |
| 075 | **if**(v>0){ |
| 076 |  |
| 077 | verticalSpeed=6; |
| 078 |  |
| 079 | transform.RotateAround(rotatepos.position,Vector3.right,verticalSpeed); |
| 080 |  |
| 081 | } |
| 082 |  |
| 083 | } |
| 084 |  |
| 085 | } |
| 086 |  |
| 087 | } |
| 088 |  |
| 089 | *//物体的缩放* |
| 090 |  |
| 091 | **if**(Input.touchCount>1){ |
| 092 |  |
| 093 | flag=**false**; |
| 094 |  |
| 095 | **if**(Input.GetTouch(0).phase==TouchPhase.Moved||Input.GetTouch(1).phase==TouchPhase.Moved){ |
| 096 |  |
| 097 | **var** pos1=Input.GetTouch(0).position; |
| 098 |  |
| 099 | **var** pos2=Input.GetTouch(1).position; |
| 100 |  |
| 101 | newdis=Vector2.Distance(pos1,pos2); |
| 102 |  |
| 103 | **if**(olddis!=**null**){ |
| 104 |  |
| 105 | **if**(newdis<olddis) { |
| 106 |  |
| 107 | Camera.main.camera.orthographicSize+=2; |
| 108 |  |
| 109 | } |
| 110 |  |
| 111 | **if**(newdis>olddis) { |
| 112 |  |
| 113 | Camera.main.camera.orthographicSize-=2; |
| 114 |  |
| 115 | } |
| 116 |  |
| 117 | } |
| 118 |  |
| 119 | olddis=newdis; |
| 120 |  |
| 121 | } |
| 122 |  |
| 123 | } |
| 124 |  |
| 125 | *//按返回键退出程序* |
| 126 |  |
| 127 | **if**(Input.GetKey(KeyCode.Escape)){ |
| 128 |  |
| 129 | Application.Quit(); |
| 130 |  |
| 131 | } |
| 132 |  |
| 133 | } |
| 134 |  |
| 135 | </pre> |
| 136 |  |
| 137 | <p>上面的代码有点局限，不适合所有物体，那么现在我就来改一下吧，让他适合每一种物体哦。</p> |
| 138 |  |
| 139 | <p></p> |
| 140 |  |
| 141 | <pre name=“code” **class**=“java”>**var** horizontalSpeed:**float**=0; |
| 142 |  |
| 143 | **var** newdis:**float**=0; |
| 144 |  |
| 145 | **var** olddis:**float**=0; |
| 146 |  |
| 147 | function Update () { |
| 148 |  |
| 149 | **if**(Input.touchCount==1){ |
| 150 |  |
| 151 | **if**(Input.GetTouch(0).phase==TouchPhase.Moved){ |
| 152 |  |
| 153 | **var** h : **float**=Input.GetAxis(“Mouse X”);*//又正左负* |
| 154 |  |
| 155 | **if**(h<0){ |
| 156 |  |
| 157 | horizontalSpeed=1.5; |
| 158 |  |
| 159 | transform.Rotate(transform.up\*horizontalSpeed); |
| 160 |  |
| 161 | } |
| 162 |  |
| 163 | **if**(h>0){ |
| 164 |  |
| 165 | horizontalSpeed=1.5; |
| 166 |  |
| 167 | transform.Rotate(-transform.up\*horizontalSpeed); |
| 168 |  |
| 169 | } |
| 170 |  |
| 171 | } |
| 172 |  |
| 173 | } |
| 174 |  |
| 175 | **if**(Input.touchCount>1){ |
| 176 |  |
| 177 | **if**(Input.GetTouch(0).phase==TouchPhase.Moved||Input.GetTouch(1).phase==TouchPhase.Moved){ |
| 178 |  |
| 179 | **var** pos1=Input.GetTouch(0).position; |
| 180 |  |
| 181 | **var** pos2=Input.GetTouch(1).position; |
| 182 |  |
| 183 | newdis=Vector2.Distance(pos1,pos2); |
| 184 |  |
| 185 | **if**(olddis!=**null**){ |
| 186 |  |
| 187 | **if**(newdis<olddis) { |
| 188 |  |
| 189 | Camera.main.camera.orthographicSize+=0.2; |
| 190 |  |
| 191 | } |
| 192 |  |
| 193 | **if**(newdis>olddis) { |
| 194 |  |
| 195 | Camera.main.camera.orthographicSize-=0.2; |
| 196 |  |
| 197 | } **if**( Camera.main.camera.orthographicSize<0){ |
| 198 |  |
| 199 | Camera.main.camera.orthographicSize=0; |
| 200 |  |
| 201 | } |
| 202 |  |
| 203 | **if**( Camera.main.camera.orthographicSize>10){ |
| 204 |  |
| 205 | Camera.main.camera.orthographicSize=10; |
| 206 |  |
| 207 | } |
| 208 |  |
| 209 | } |
| 210 |  |
| 211 | olddis=newdis; |
| 212 |  |
| 213 | } |
| 214 |  |
| 215 | } |
| 216 |  |
| 217 | *//按返回键退出程序* |
| 218 |  |
| 219 | **if**(Input.GetKey(KeyCode.Escape)){ |
| 220 |  |
| 221 | Application.Quit(); |
| 222 |  |
| 223 | } |