**实现描边效果（轮廓边框）**

Posted on 2013年07月25日 by U3d / [Unity3D脚本/插件](http://www.unitymanual.com/category/script) /被围观 41 次

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

|  |  |
| --- | --- |
| 001 | Shader "Outlined/Silhouetted Diffuse" { |
| 002 | Properties { |
| 003 | \_Color ("Main Color", Color) = (1,1,1,1) |
| 004 | \_OutlineColor ("Outline Color", Color) = (0,0,0,1) *//改变这个能改变轮廓边的颜色* |
| 005 | \_Outline ("Outline width", Range (0.0, 0.03)) = 0.008 *//改变这个能改变轮廓边的粗细* |
| 006 | \_MainTex ("Base (RGB)", 2D) = "white" { } |
| 007 | } |
| 008 |  |
| 009 | CGINCLUDE |
| 010 | #include "UnityCG.cginc" |
| 011 |  |
| 012 | **struct** appdata { |
| 013 | float4 vertex : POSITION; |
| 014 | float3 normal : NORMAL; |
| 015 | }; |
| 016 |  |
| 017 | **struct** v2f { |
| 018 | float4 pos : POSITION; |
| 019 | float4 color : COLOR; |
| 020 | }; |
| 021 |  |
| 022 | uniform **float** \_Outline; |
| 023 | uniform float4 \_OutlineColor; |
| 024 |  |
| 025 | v2f vert(appdata v) { |
| 026 | *// just make a copy of incoming vertex data but scaled according to normal direction* |
| 027 | v2f o; |
| 028 | o.pos = mul(UNITY\_MATRIX\_MVP, v.vertex); |
| 029 |  |
| 030 | float3 norm = mul ((float3x3)UNITY\_MATRIX\_IT\_MV, v.normal); |
| 031 | float2 offset = TransformViewToProjection(norm.xy); |
| 032 |  |
| 033 | o.pos.xy += offset \* o.pos.z \* \_Outline; |
| 034 | o.color = \_OutlineColor; |
| 035 | **return** o; |
| 036 | } *//Unity3D教程手册：www.unitymanual.com* |
| 037 | ENDCG |
| 038 |  |
| 039 | SubShader { |
| 040 | Tags { "Queue" = "Transparent" } |
| 041 |  |
| 042 | *// note that a vertex shader is specified here but its using the one above* |
| 043 | Pass { |
| 044 | Name "OUTLINE" |
| 045 | Tags { "LightMode" = "Always" } |
| 046 | Cull Off |
| 047 | ZWrite Off |
| 048 | ZTest Always |
| 049 | ColorMask RGB *// alpha not used* |
| 050 |  |
| 051 | *// you can choose what kind of blending mode you want for the outline* |
| 052 | Blend SrcAlpha OneMinusSrcAlpha *// Normal* |
| 053 | *//Blend One One // Additive* |
| 054 | *//Blend One OneMinusDstColor // Soft Additive* |
| 055 | *//Blend DstColor Zero // Multiplicative* |
| 056 | *//Blend DstColor SrcColor // 2x Multiplicative* |
| 057 |  |
| 058 | CGPROGRAM |
| 059 | #pragma vertex vert |
| 060 | #pragma fragment frag |
| 061 |  |
| 062 | half4 frag(v2f i) :COLOR { |
| 063 | **return** i.color; |
| 064 | } |
| 065 | ENDCG |
| 066 | } |
| 067 |  |
| 068 | Pass { |
| 069 | Name "BASE" |
| 070 | ZWrite On |
| 071 | ZTest LEqual |
| 072 | Blend SrcAlpha OneMinusSrcAlpha |
| 073 | Material { |
| 074 | Diffuse [\_Color] |
| 075 | Ambient [\_Color] |
| 076 | } |
| 077 | Lighting On |
| 078 | SetTexture [\_MainTex] { |
| 079 | ConstantColor [\_Color] |
| 080 | Combine texture \* constant |
| 081 | } |
| 082 | SetTexture [\_MainTex] { |
| 083 | Combine previous \* primary **DOUBLE** |
| 084 | } *//Unity3D教程手册：www.unitymanual.com* |
| 085 | } |
| 086 | } |
| 087 |  |
| 088 | SubShader { |
| 089 | Tags { "Queue" = "Transparent" } |
| 090 |  |
| 091 | Pass { |
| 092 | Name "OUTLINE" |
| 093 | Tags { "LightMode" = "Always" } |
| 094 | Cull Front |
| 095 | ZWrite Off |
| 096 | ZTest Always |
| 097 | ColorMask RGB |
| 098 |  |
| 099 | *// you can choose what kind of blending mode you want for the outline* |
| 100 | Blend SrcAlpha OneMinusSrcAlpha *// Normal* |
| 101 | *//Blend One One // Additive* |
| 102 | *//Blend One OneMinusDstColor // Soft Additive* |
| 103 | *//Blend DstColor Zero // Multiplicative* |
| 104 | *//Blend DstColor SrcColor // 2x Multiplicative* |
| 105 |  |
| 106 | CGPROGRAM |
| 107 | #pragma vertex vert |
| 108 | #pragma exclude\_renderers gles xbox360 ps3 |
| 109 | ENDCG |
| 110 | SetTexture [\_MainTex] { combine primary } |
| 111 | } *//Unity3D教程手册：www.unitymanual.com* |
| 112 |  |
| 113 | Pass { |
| 114 | Name "BASE" |
| 115 | ZWrite On |
| 116 | ZTest LEqual |
| 117 | Blend SrcAlpha OneMinusSrcAlpha |
| 118 | Material { |
| 119 | Diffuse [\_Color] |
| 120 | Ambient [\_Color] |
| 121 | } |
| 122 | Lighting On |
| 123 | SetTexture [\_MainTex] { |
| 124 | ConstantColor [\_Color] |
| 125 | Combine texture \* constant |
| 126 | } |
| 127 | SetTexture [\_MainTex] { |
| 128 | Combine previous \* primary **DOUBLE** |
| 129 | } |
| 130 | } |
| 131 | } |
| 132 |  |
| 133 | Fallback "Diffuse" |
| 134 | } |
| 135 |  |
| 136 |  |