

## cartzhang的专栏

三维，虚拟现实，游戏

目录视图摘要视图RSS 订阅

### 个人资料

译



cartzhang

关注发私信



访问：322673次

积分：4515

等级：BLOG 5

排名：第6002名

原创：105篇

转载：2篇

译文：21篇

评论：139条

### 博客专栏



图说VR入门

文章：9篇

阅读：25010



VR开发

文章：23篇

阅读：57550



Unity5学习记录

文章：33篇

阅读：109953



虚幻4基础学习

文章：16篇

阅读：47672

### 关于作者

cartzhang：三维平台开发，UE4 和 Unity 3D程序开发，VR游戏开发人员

联系方式：QQ：40278636

Email：cart033#sina.com 自行修改#为@

### 关于版权

关于版权



本作品采用署名-非商业性使用-禁止演

## Unity3D Asset文件导出3DMax 可编辑格式

标签：unity3d mesh 导出 3DMax OBJ

2017-03-08 18:34

1251人阅读

评论(1)

收藏

举报

分类：Unity ( 62 ) C# ( 47 )

版权声明：本文为博主cartzhang原创文章，未经博主允许不得转载。http://blog.csdn.net/cartzhang

目录(?)

[+]

本文章由cartzhang编写，转载请注明出处。 所有权利保留。

文章链接：<http://blog.csdn.net/cartzhang/article/details/60878354>

作者：cartzhang

## 一、前言

美术想要一个把unity中\*.asset的模型导出来，导出3D Max可以打开的模式，fbx或obj.

需要导出的格式：

| 名称                            | 修改日期           | 类型       | 大小     |
|-------------------------------|----------------|----------|--------|
| FH_Guard_Pole_Axe.asset       | 2014/1/19 6:47 | ASSET 文件 | 20 KB  |
| FH_Guard_sword_scabbard.asset | 2014/1/19 5:04 | ASSET 文件 | 12 KB  |
| FH_Knight.asset               | 2014/1/19 5:04 | ASSET 文件 | 162 KB |
| FH_Knight_hammer.asset        | 2014/1/19 5:04 | ASSET 文件 | 16 KB  |
| FH_Knight_shield.asset        | 2014/1/19 5:04 | ASSET 文件 | 13 KB  |
| FH_Knight_sword.asset         | 2014/1/19 5:04 | ASSET 文件 | 15 KB  |
| helmet_01.asset               | 2014/1/19 5:04 | ASSET 文件 | 12 KB  |
| helmet_02.asset               | 2014/1/19 5:04 | ASSET 文件 | 17 KB  |
| helmet_03.asset               | 2014/1/19 5:04 | ASSET 文件 | 15 KB  |

图1

也就是需要一个工具，个人觉得这个问题，肯定之前Unity的前辈就有解决方法了。于是乎网上一通下载和测试。

## 二、解包工具集合

网络上找来了各种测试，但是没有一个适合我的，很多都是失败，打不开。

参考宣雨松的博客，找了还是没有结果。

|   |                |                    |           |
|---|----------------|--------------------|-----------|
| AssetsBundleExtractor_1.7_64bit.zip     | 2017/3/6 18:41 | WinRAR ZIP 压缩...   | 1,055 KB  |
| disunity_v0.4.0.zip                     | 2017/3/7 14:33 | WinRAR ZIP 压缩...   | 1,091 KB  |
| disunity_v0.5.0.zip                     | 2017/3/6 18:54 | WinRAR ZIP 压缩...   | 2,987 KB  |
| DisunityGui_v0.2.1.7z                   | 2017/3/7 10:57 | WinRAR 压缩文件        | 35 KB     |
| DisunityGuiHelper-0.2.1.zip             | 2017/3/7 11:29 | WinRAR ZIP 压缩...   | 84 KB     |
| jcommander-jcommander-1.47.zip          | 2017/3/7 10:42 | WinRAR ZIP 压缩...   | 147 KB    |
| jre-8u121-windows-i586.exe              | 2017/3/7 11:42 | 应用程序               | 55,105 KB |
| Unity Studio.zip                        | 2017/3/7 12:19 | WinRAR ZIP 压缩...关闭 | 2,089 KB  |
| UnityAssetsExplorer 1.3 test %2808.0... | 2017/3/7 14:52 | WinRAR 压缩文件        | 623 KB    |
| UnityAssetsExplorer.exe                 | 2017/3/7 14:51 | 应用程序               | 924 KB    |
| UnityAssetsExplorer_1.5.rar             | 2017/3/7 13:18 | WinRAR 压缩文件        | 884 KB    |
| UnityFBXExporter-master.zip             | 2017/3/7 16:43 | WinRAR ZIP 压缩...   | 171 KB    |

绎 2.5 中国大陆进行许可。

作品 (cartzhang博文, 由 [cartzhang](#) 创作), 由 [cartzhang](#) 确认, 转载请注明版权。

文章分类

C++

(26)

C#

(48)

GIS

(1)

cmake

(3)

lua

(2)

openGL

(1)

android

(1)

OpenCV

(1)

Unreal

(28)

Kinect

(7)

VR

(29)

Unity

(63)

工具

(2)

管理

(3)

读技术类书

(1)

阅读排行

|                             |         |
|-----------------------------|---------|
| Unity 实现Log实时输出到屏...        | (15369) |
| Unity的Json解析<一>--读取...      | (13351) |
| UE4 Editor快捷键(ShortCut ...  | (8880)  |
| Unity5的关卡切换                 | (7839)  |
| 在 Oculus和 Gear VR上开发...     | (7302)  |
| 《图说VR入门》——google...         | (7047)  |
| VR室内定位系统小结                  | (6803)  |
| 热烈庆祝UE4完全免费Free---...       | (6301)  |
| Unity 实现Log实时输出到屏...        | (5724)  |
| UE4.5.0的Kinect插件(Plugin)... | (5618)  |

评论排行

|                             |      |
|-----------------------------|------|
| UE4.5.0的Kinect插件(Plugin)... | (17) |
| UE4制作插件的插件神器plug...         | (10) |
| Unity 实现Log实时输出到屏...        | (10) |
| 虚幻引擎4艺术大师 - 中文版...          | (8)  |
| 热烈庆祝UE4完全免费Free---...       | (7)  |
| 《图说VR入门》——google...         | (7)  |
| Unity的Json解析<一>--读取...      | (6)  |
| 虚幻引擎自带的创建插件的插件              | (6)  |
| Kinect 骨骼映射---Let me d...   | (5)  |
| 美轮美奂宇宙星空制作神器Sp...           | (4)  |

最新评论

虚幻引擎4艺术大师 - 中文版免费分享

dhaizhang

：灰常感谢

理解HTC Vive更新——控制相机旋转和...

cartzhang

：这个暂时没办法测试。你若方便测试下，请告诉我结果。每帧都硬扭回来，怎么会扭不动呢》 不合理啊

理解HTC Vive更新——控制相机旋转和...

zpciswinner

：@cartzhang:确定是修改后的代码（复制的图4下方的代码）。除了HTC的相机，还有其他两个相机...

Unity中Vive Controller手柄模型渲染流...

cartzhang

：若只是修改模型，最简单方法是使用模型替换。在有就是在代码中隐藏原有手柄模型，然后把自己新的模型显示出...

理解HTC Vive更新——控制相机旋转和...

cartzhang

：确定使用的不是我错误的代码，而是修改后的代码。有就是每帧都动态的修改当前其他相机的值。

理解HTC Vive更新——控制相机旋转和...

zpciswinner

：你好，博主。我使用了你的方法，为什么其他相机还是会跟随头盔旋转和移动（直接复制的代码）？

Unity中Vive Controller手柄模型渲染流...

天空之城-VR

：你好，我想问一下，我要是想改变手柄模型，当然我手柄上按钮是有事件的，需要从第几步开始啊，看着有点迷。

Unity 实现Log实时输出到屏幕或控制台...

图3

解包工具有很多种类，

disunity github地址: <https://github.com/ata4/disunity>

还有就是AssetAssetsExport，还有Unity Studio.

别人的博客里面都有比较多的介绍和说明，这里就详细说了。

最后还网上wiki里，找到了一个合适的我自己的解包。

<http://wiki.unity3d.com/index.php?title=ObjExporter>

### 三、初步成果

找到了一个网站：<http://wiki.unity3d.com/index.php?title=ObjExporter>

可以导出部分对象。

如下图：



图0

而原来unity中模型是这个样子的。

cartzhang : @ghl1390490928:没有的话，自己在Asset下新建一个Editor！！

Unity 实现Log实时输出到屏幕或控制台...  
ghl1390490928 : 大神你好，Editor文件夹在哪

Unity 实现Log实时输出到屏幕或控制台...  
ghl1390490928 : 大神你好，Editor文件夹在哪

统计



图4

导出的只有武器和头盔，没有人物主体body.

#### 四、bug修改

其实也不能算bug，也许人家没有这样的需要呢。

```
1 Component[] meshfilter = selection[i].GetComponentsInChildren<MeshFilter>();
2 MeshFilter[] mf = new MeshFilter[meshfilter.Length];
3 int m = 0;
4 for (; m < meshfilter.Length; m++)
5 {
6     exportedObjects++;
7     mf[m] = (MeshFilter)meshfilter[m];
8 }
```

代码中是要查找所有组件中的MeshFilter，发现SkinnedMeshRender组件居然没有这个MeshFilter这个组件，所以总会导出少一个，而这个居然是人的主体。

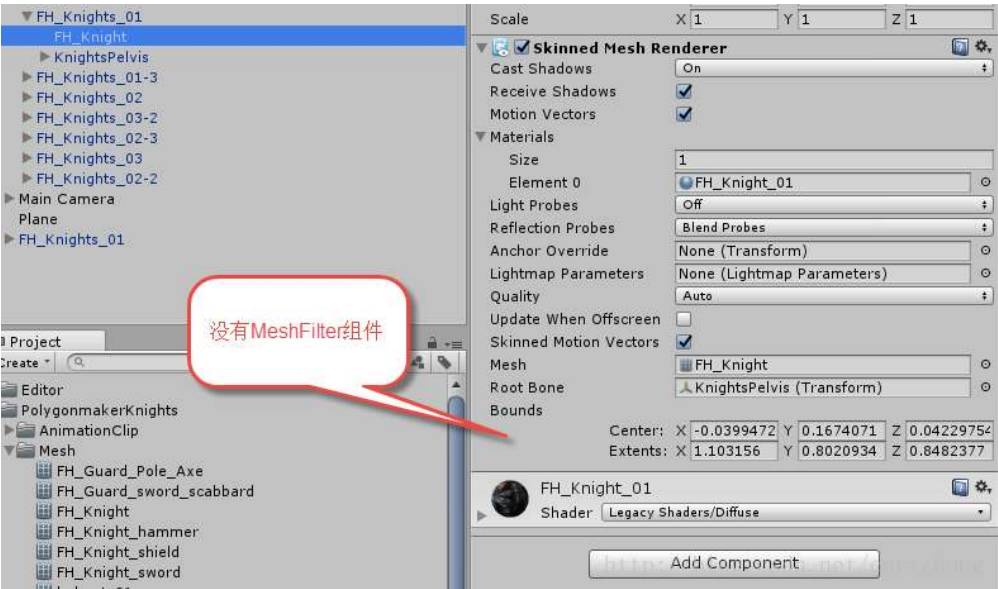


图5

本来说让美术自己添加一个MeshFilter组件，然后根据mesh render中的mesh自己来添加一个对应的mesh.

既然是程序，那就想办法，思路很明显，既然是有meshrender,就从这入手呗。

代码还是不难度。

```
1 // 没有meshFilter，添加一个meshFilter.
2 SkinnedMeshRender[] meshfilterRender = selection[i].GetComponentsInChildren<SkinnedMeshRender>();
3 for (int j = 0; j < meshfilterRender.Length; j++)
4 {
5     if (meshfilterRender[j].GetComponent<MeshFilter>() == null)
6     {
```

```
7 meshfilterRender[j].gameObject.AddComponent<MeshFilter>();
8 meshfilterRender[j].GetComponent<MeshFilter>().sharedMesh = Instantiate(meshfilterRender[j].sharedMesh);
9     }
10 }
```

这样修改过，就会自动在没有MeshFilter，但是有skinnedMeshRender组件的节点下，添加一个MeshFilter,然后就可以正常导出成.obj文件，与.FBX是类似的，都可以被3D max编辑使用。

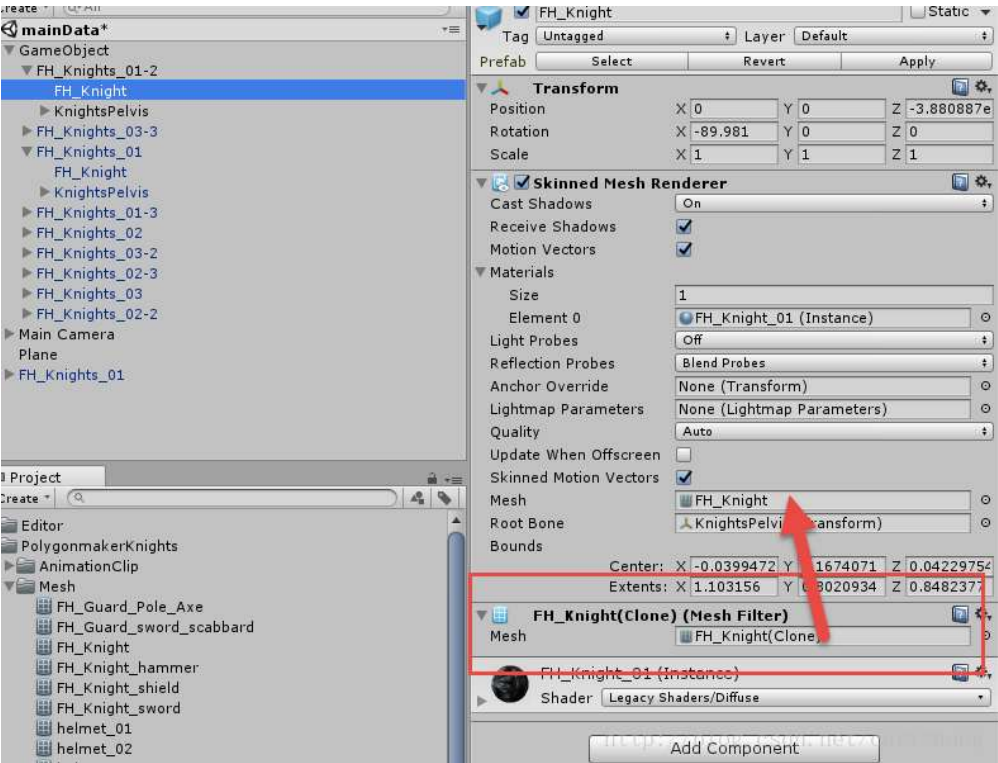


图7

最后的在VS中看的模型，因为没有安装3Dmax.

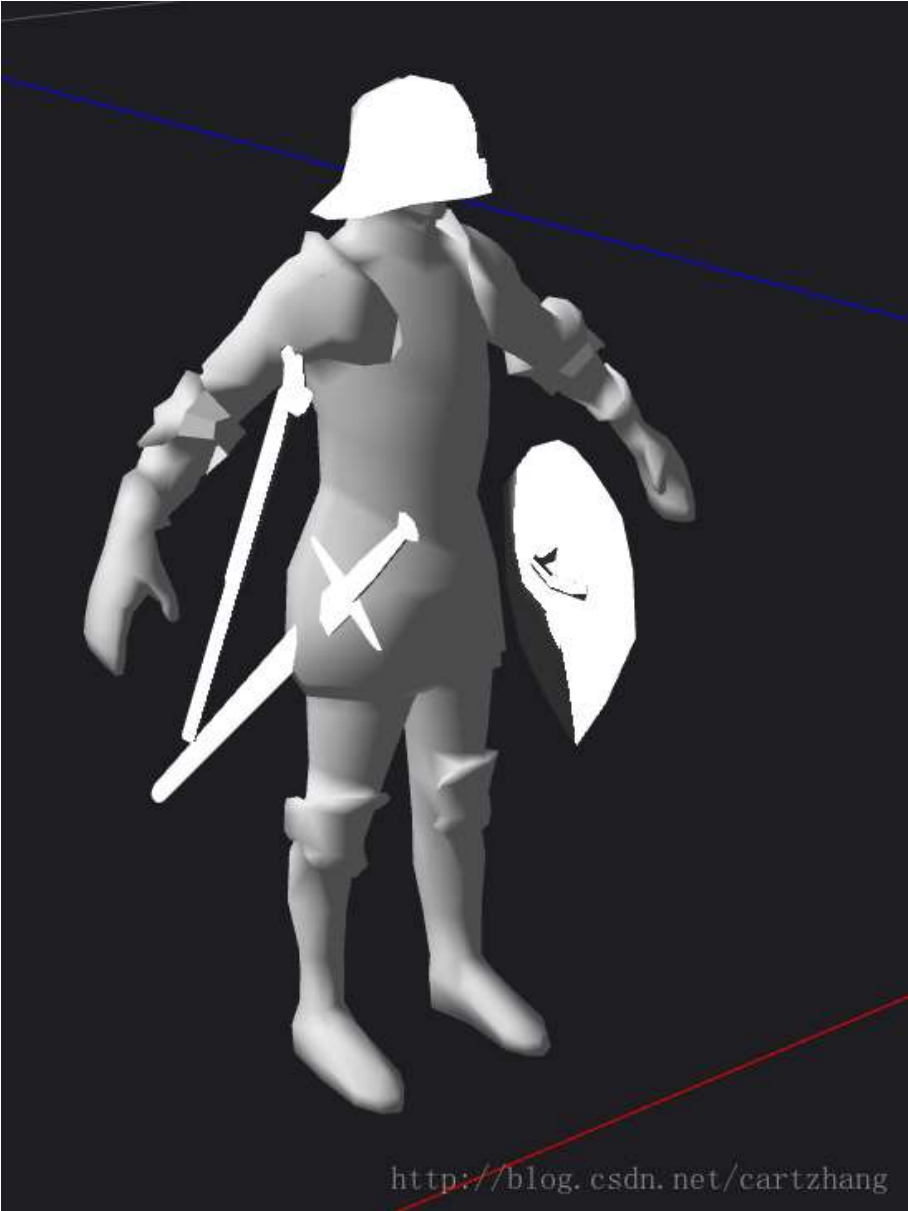


图6

虽然看起来简陋，但是满足他们小需要，就好了。

贴出主要的代码：

```
1 /*
2 Based on ObjExporter.cs, this "wrapper" lets you export to .OBJ directly from the editor menu.
3
```



```
4 This should be put in your "Editor"-folder. Use by selecting the objects you want to export, and select
5 the appropriate menu item from "Custom->Export". Exported models are put in a folder called
6 "ExportedObj" in the root of your Unity-project. Textures should also be copied and placed in the
7 same folder.
8 N.B. there may be a bug so if the custom option doesn't come up refer to this thread http://answers.unity3d.com/questions/31795
9
10 Updated for Unity 5.3
11
12 2017-03-07
13 @cartzhang
14 fixed can not create obj file in folder.
15 */
16
17 using UnityEngine;
18 using UnityEditor;
19 using UnityEditor.SceneManagement;
20 using System.Collections;
21 using System.Collections.Generic;
22 using System.IO;
23 using System.Text;
24 using System;
25
26 struct ObjMaterial
27 {
28     public string name;
29     public string textureName;
30 }
31
32 public class EditorObjExporter : ScriptableObject
33 {
34     private static int vertexOffset = 0;
35     private static int normalOffset = 0;
36     private static int uvOffset = 0;
37
38
39     //User should probably be able to change this. It is currently left as an exercise for
40     //the reader.
41     private static string targetFolder = "ExportedObj";
42
43
44     private static string MeshToString(MeshFilter mf, Dictionary<string, ObjMaterial> materialList)
45     {
46         Debug.Assert(null != mf);
47         Mesh m = mf.sharedMesh;
48         Material[] mats = mf.GetComponent<Renderer>().sharedMaterials;
49
50         StringBuilder sb = new StringBuilder();
51         if (null == m)
52             return sb.ToString();
53
54         sb.Append("g ").Append(mf.name).Append("\n");
55         foreach (Vector3 lv in m.vertices)
56         {
57             Vector3 wv = mf.transform.TransformPoint(lv);
58
59             //This is sort of ugly - inverting x-component since we're in
60             //a different coordinate system than "everyone" is "used to".
61             sb.Append(string.Format("v {0} {1} {2}\n", -wv.x, wv.y, wv.z));
62         }
63         sb.Append("\n");
64
65         foreach (Vector3 lv in m.normals)
66         {
67             Vector3 wv = mf.transform.TransformDirection(lv);
68
69             sb.Append(string.Format("vn {0} {1} {2}\n", -wv.x, wv.y, wv.z));
70         }
71         sb.Append("\n");
72
73         foreach (Vector3 v in m.uv)
74         {
75             sb.Append(string.Format("vt {0} {1}\n", v.x, v.y));
76         }
77
78         for (int material = 0; material < m.subMeshCount; material++)
79         {
80             sb.Append("\n");
81             sb.Append("usemtl ").Append(mats[material].name).Append("\n");
82             sb.Append("usemap ").Append(mats[material].name).Append("\n");
83
84             //See if this material is already in the materiallist.
85             try
86             {
87                 ObjMaterial objMaterial = new ObjMaterial();
88
89                 objMaterial.name = mats[material].name;
90
91                 if (mats[material].mainTexture)
92                     objMaterial.textureName = AssetDatabase.GetAssetPath(mats[material].mainTexture);
93                 else
94                     objMaterial.textureName = null;
```

```

95
96         materialList.Add(objMaterial.name, objMaterial);
97     }
98     catch (ArgumentException)
99     {
100         //Already in the dictionary
101     }
102
103
104     int[] triangles = m.GetTriangles(material);
105     for (int i = 0; i < triangles.Length; i += 3)
106     {
107         //Because we inverted the x-component, we also needed to alter the triangle winding.
108         sb.Append(string.Format("f {1}/{1}/{1} {0}/{0}/{0} {2}/{2}/{2}\n",
109             triangles[i] + 1 + vertexOffset, triangles[i + 1] + 1 + normalOffset, triangles[i + 2] + 1 + uvOffset));
110     }
111 }
112
113 vertexOffset += m.vertices.Length;
114 normalOffset += m.normals.Length;
115 uvOffset += m.uv.Length;
116
117 return sb.ToString();
118 }
119
120 private static void Clear()
121 {
122     vertexOffset = 0;
123     normalOffset = 0;
124     uvOffset = 0;
125 }
126
127 private static Dictionary<string, ObjMaterial> PrepareFileWrite()
128 {
129     Clear();
130
131     return new Dictionary<string, ObjMaterial>();
132 }
133
134 private static void MaterialsToFile(Dictionary<string, ObjMaterial> materialList, string folder, string filename)
135 {
136     using (StreamWriter sw = new StreamWriter(folder + Path.DirectorySeparatorChar + filename + ".mtl"))
137     {
138         foreach (KeyValuePair<string, ObjMaterial> kvp in materialList)
139         {
140             sw.Write("\n");
141             sw.Write("newmtl {0}\n", kvp.Key);
142             sw.Write("Ka 0.6 0.6 0.6\n");
143             sw.Write("Kd 0.6 0.6 0.6\n");
144             sw.Write("Ks 0.9 0.9 0.9\n");
145             sw.Write("d 1.0\n");
146             sw.Write("Ns 0.0\n");
147             sw.Write("illum 2\n");
148
149             if (kvp.Value.textureName != null)
150             {
151                 string destinationFile = kvp.Value.textureName;
152
153
154                 int stripIndex = destinationFile.LastIndexOf(Path.DirectorySeparatorChar);
155
156                 if (stripIndex >= 0)
157                     destinationFile = destinationFile.Substring(stripIndex + 1).Trim();
158
159                 string relativeFile = destinationFile;
160
161                 destinationFile = folder + Path.DirectorySeparatorChar + destinationFile;
162
163                 Debug.Log("Copying texture from " + kvp.Value.textureName + " to " + destinationFile);
164
165                 try
166                 {
167                     //Copy the source file
168                     File.Copy(kvp.Value.textureName, destinationFile);
169                 }
170                 catch
171                 {
172
173                 }
174
175                 sw.Write("map_Kd {0}", relativeFile);
176             }
177
178             sw.Write("\n\n\n");
179         }
180     }
181 }
182
183 }
184
185 private static void MeshToFile(MeshFilter mf, string folder, string filename)

```

```

186     {
187         Dictionary<string, ObjMaterial> materialList = PrepareFileWrite();
188
189         using (StreamWriter sw = new StreamWriter(folder + Path.DirectorySeparatorChar + filename + ".obj"))
190         {
191             sw.Write("mtllib ./" + filename + ".mtl\n");
192
193             sw.Write(MeshToString(mf, materialList));
194         }
195
196         MaterialsToFile(materialList, folder, filename);
197     }
198
199     private static void MeshesToFile(MeshFilter[] mf, string folder, string filename)
200     {
201         Dictionary<string, ObjMaterial> materialList = PrepareFileWrite();
202
203         using (StreamWriter sw = new StreamWriter(folder + Path.DirectorySeparatorChar + filename + ".obj"))
204         {
205             sw.Write("mtllib ./" + filename + ".mtl\n");
206
207             for (int i = 0; i < mf.Length; i++)
208             {
209                 sw.Write(MeshToString(mf[i], materialList));
210             }
211         }
212
213         MaterialsToFile(materialList, folder, filename);
214     }
215
216     private static bool CreateTargetFolder()
217     {
218         try
219         {
220             System.IO.Directory.CreateDirectory(targetFolder);
221         }
222         catch
223         {
224             EditorUtility.DisplayDialog("Error!", "Failed to create target folder!", "");
225             return false;
226         }
227
228         return true;
229     }
230
231     [MenuItem("Custom/Export/Export whole selection to single OBJ")]
232     static void ExportWholeSelectionToSingle()
233     {
234         if (!CreateTargetFolder())
235             return;
236
237
238         Transform[] selection = Selection.GetTransforms(SelectionMode.Editable | SelectionMode.ExcludePrefab);
239
240         if (selection.Length == 0)
241         {
242             EditorUtility.DisplayDialog("No source object selected!", "Please select one or more target objects", "");
243             return;
244         }
245
246         int exportedObjects = 0;
247
248         ArrayList mfList = new ArrayList();
249
250         for (int i = 0; i < selection.Length; i++)
251         {
252             Component[] meshfilter = selection[i].GetComponentsInChildren(typeof(MeshFilter));
253
254             for (int m = 0; m < meshfilter.Length; m++)
255             {
256                 exportedObjects++;
257                 mfList.Add(meshfilter[m]);
258             }
259         }
260
261         if (exportedObjects > 0)
262         {
263             MeshFilter[] mf = new MeshFilter[mfList.Count];
264
265             for (int i = 0; i < mfList.Count; i++)
266             {
267                 mf[i] = (MeshFilter)mfList[i];
268             }
269
270             string filename = EditorSceneManager.GetActiveScene().name + "_" + exportedObjects;
271
272             int stripIndex = filename.LastIndexOf(Path.DirectorySeparatorChar);
273
274             if (stripIndex >= 0)
275                 filename = filename.Substring(stripIndex + 1).Trim();
276

```

```
277         MeshesToFile(mf, targetFolder, filename);
278
279
280         EditorUtility.DisplayDialog("Objects exported", "Exported " + exportedObjects + " objects to " + filename, "");
281     }
282     else
283         EditorUtility.DisplayDialog("Objects not exported", "Make sure at least some of your selected objects have mesh fil
284 }
285
286
287
288 [MenuItem("Custom/Export/Export each selected to single OBJ")]
289 static void ExportEachSelectionToSingle()
290 {
291     if (!CreateTargetFolder())
292         return;
293
294     Transform[] selection = Selection.GetTransforms(SelectionMode.Editable | SelectionMode.ExcludePrefab);
295
296     if (selection.Length == 0)
297     {
298         EditorUtility.DisplayDialog("No source object selected!", "Please select one or more target objects", "");
299         return;
300     }
301
302     int exportedObjects = 0;
303
304
305     for (int i = 0; i < selection.Length; i++)
306     {
307         // 没有meshFilter, 添加一个meshFilter.
308         SkinnedMeshRenderer[] meshfilterRender = selection[i].GetComponentsInChildren<SkinnedMeshRenderer>();
309         for (int j = 0; j < meshfilterRender.Length; j++)
310         {
311             if (meshfilterRender[j].GetComponent<MeshFilter>() == null)
312             {
313                 meshfilterRender[j].gameObject.AddComponent<MeshFilter>();
314                 meshfilterRender[j].GetComponent<MeshFilter>().sharedMesh = Instantiate(meshfilterRender[j].sharedMesh);
315             }
316         }
317
318         Component[] meshfilter = selection[i].GetComponentsInChildren<MeshFilter>();
319         MeshFilter[] mf = new MeshFilter[meshfilter.Length];
320         int m = 0;
321         for (; m < meshfilter.Length; m++)
322         {
323             exportedObjects++;
324             mf[m] = (MeshFilter)meshfilter[m];
325         }
326
327         MeshesToFile(mf, targetFolder, selection[i].name + "_" + i);
328     }
329
330     if (exportedObjects > 0)
331     {
332         EditorUtility.DisplayDialog("Objects exported", "Exported " + exportedObjects + " objects", "");
333     }
334     else
335         EditorUtility.DisplayDialog("Objects not exported", "Make sure at least some of your selected objects have mesh fil
336 }
337
338 }
```

可以直接复制到直接的项目中使用。

## 五、怎么使用呢？

首先把代码拷贝到项目中，直接下载工程也行。

步骤一



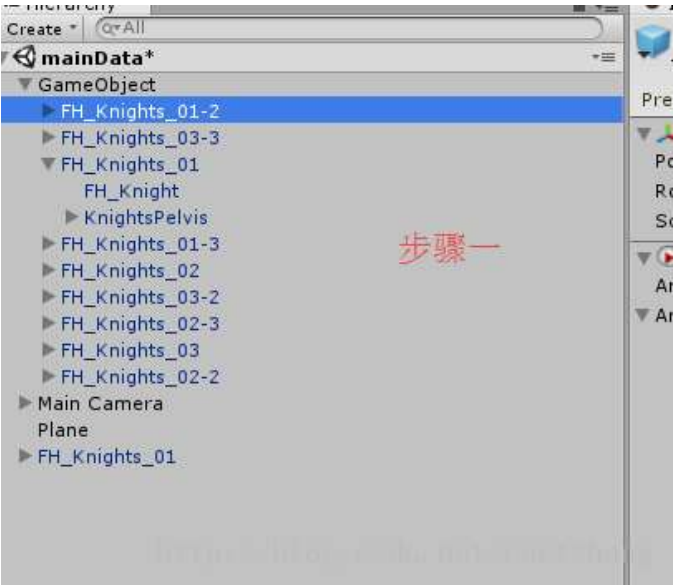


图8

步骤二



图10

步骤三



图11

就可以使用你的模型编辑工具来查看了。

## 五、源码和示例工程

源码地址：

[https://github.com/cartzhang/unity\\_lab/blob/master/ExportFbx/UnityAssetExportFBX/Assets/Editor/OBJExport/EditorObjExporter.cs](https://github.com/cartzhang/unity_lab/blob/master/ExportFbx/UnityAssetExportFBX/Assets/Editor/OBJExport/EditorObjExporter.cs)

示例工程地址：

[https://github.com/cartzhang/unity\\_lab/tree/master/ExportFbx/UnityAssetExportFBX](https://github.com/cartzhang/unity_lab/tree/master/ExportFbx/UnityAssetExportFBX)

博客图片地址：

[https://github.com/cartzhang/unity\\_lab/tree/master/ExportFbx/Img](https://github.com/cartzhang/unity_lab/tree/master/ExportFbx/Img)

Github readme：

[https://github.com/cartzhang/unity\\_lab/blob/master/ExportFbx/Unity%20asset%E6%96%87%E4%BB%B6%20%E5%AF%BC%E5%87%BAOBJ.md](https://github.com/cartzhang/unity_lab/blob/master/ExportFbx/Unity%20asset%E6%96%87%E4%BB%B6%20%E5%AF%BC%E5%87%BAOBJ.md)

## 六、参考

【1】<http://www.xuanyusong.com/archives/3618>

- 【2】<https://forums.inxile-entertainment.com/viewtopic.php?t=13724>
- 【3】<http://www.cnblogs.com/Niger123/p/4261763.html>
- 【4】<http://prog3.com/sbdm/download/download/akof1314/9097153>
- 【5】<http://wiki.unity3d.com/index.php?title=ObjExporter>
- 【6】<https://github.com/KellanHiggins/UnityFBXExporter/tree/master/Assets/Packages/UnityFBXExporter>

七，最后但不是不重要

Asset导出成FBX的格式：<https://github.com/cartzhang/UnityFBXExporter>

与上面介绍的不是一个方法，但是思路都一样。这个源码可以把**纹理和材质**都匹配上去，当然我也做了稍微的修改，修复了之前的小bug。

非常感谢，欢迎留言！！

顶

1

踩

0

- 上一篇

Unity中动态修改游戏中任意参数的框架
- 下一篇

Unity中Vive Controller手柄模型渲染流程和模型替换

相关文章推荐

unity基础

3DMAX导出unity3d中使用的FBX

3dMax导出fbx三维模型到Unity3D中

Unity3D asset bundle 格式简析

Unity3D asset bundle 格式简析

Unity3D asset bundle 格式解析

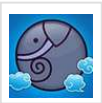
Unity3d 和 3dmax轴向问题以及快速解决方案

# Unity3d Mecanim动画如何应用3dmax中的动画...

Unity3D导入3DMax模型缩放单位问题深入分析

Unity3D加载火炬之光资源（3Dmax带动画）

参考知识库



PHP知识库  
6907 关注 | 1097 收录



Unity3D知识库  
3803 关注 | 511 收录



.NET知识库  
3986 关注 | 839 收录



软件测试知识库  
4814 关注 | 318 收录

猜你在找

- Python项目实训

Python编程基础视频教程(第四季)

Hadoop生态系统零基础入门

零基础学HTML 5实战开发(第一季)

Swift视频教程(第七季)
- Python编程基础视频教程(第三季)

Python编程基础视频教程(第七季)

HTML 5移动开发从入门到精通

Android入门实战教程

C语言及程序设计初步

查看评论



puppet\_master

感谢博主分享！

1楼 2017-03-09 19:24发表

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题    Hadoop    AWS    移动游戏    Java    Android    iOS    Swift    智能硬件    Docker    OpenStack    VPN    Spark    ERP

IE10    Eclipse    CRM    JavaScript    数据库    Ubuntu    NFC    WAP    jQuery    BI    HTML5    Spring    Apache    .NET

API    HTML    SDK    IIS    Fedora    XML    LBS    Unity    Splashtop    UML    components    Windows Mobile    Rails

QEMU    KDE    Cassandra    CloudStack    FTC    coremail    OPhone    CouchBase    云计算    iOS6    Rackspace    Web App

SpringSide    Maemo    Compuware    大数据    aptech    Perl    Tornado    Ruby    Hibernate    ThinkPHP    HBase    Pure    Solr

Angular    Cloud Foundry    Redis    Scala    Django    Bootstrap

公司简介 | 招贤纳士 | 广告服务 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服    杂志客服    微博客服    webmaster@csdn.net    400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 | 江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved 