**批量修改贴图导入设置工具脚本**

Posted on 2013年07月08日 by U3d / [Unity3D脚本/插件](http://www.unitymanual.com/category/script) /被围观 35 次

这个Unity3D批量修改贴图导入设置工具脚本十分小巧，但是威力大。特别针对大批量贴图要调整尺寸等等的时候作用尤为明显。在菜单中添加“Custom→Texture”的方式来批量改变所选的贴图导入设置。Unity本身只能一次打开一张图片进行导入设置，目前这个脚本可以批量更改贴图格式，是否开启MipMap，调整纹理最大尺寸，是否可读等等。

用法是把脚本放在你项目的资源目录的Editor文件夹下。然后选择你要批处理的纹理。到菜单中选择要处理的类型就可以了。

ChangeTextureImportSettings。csforUnity2.x

程序代码csharp代码：

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

|  |  |
| --- | --- |
| 001 | **using** UnityEngine; |
| 002 | **using** UnityEditor; |
| 003 |  |
| 004 | *// /////////////////////////////////////////////////////////////////////////////////////////////////////////* |
| 005 | *//* |
| 006 | *// Batch Texture import settings modifier.* |
| 007 | *//* |
| 008 | *// Modifies all selected textures in the project window and applies the requested modification on the* |
| 009 | *// textures. Idea was to have the same choices for multiple files as you would have if you open the* |
| 010 | *// import settings of a single texture. Put this into Assets/Editor and once compiled by Unity you find* |
| 011 | *// the new functionality in Custom -> Texture. Enjoy! :-)* |
| 012 | *//* |
| 013 | *// Based on the great work of benblo in this thread:* |
| 014 | *// http://forum.unity3d.com/viewtopic.php?t=16079&start=0&postdays=0&postorder=asc&highlight=textureimporter* |
| 015 | *//* |
| 016 | *// Developed by Martin Schultz, Decane in August 2009* |
| 017 | *// e-mail: ms@decane.net* |
| 018 | *//* |
| 019 | *// /////////////////////////////////////////////////////////////////////////////////////////////////////////* |
| 020 | **public** **class** ChangeTextureImportSettings : ScriptableObject { |
| 021 |  |
| 022 | [MenuItem ("Custom/Texture/Change Texture Format/Auto")] |
| 023 | **static** **void** ChangeTextureFormat\_Auto() { |
| 024 | SelectedChangeTextureFormatSettings(TextureImporterFormat.Automatic); |
| 025 | } |
| 026 |  |
| 027 | [MenuItem ("Custom/Texture/Change Texture Format/RGB Compressed DXT1")] |
| 028 | **static** **void** ChangeTextureFormat\_RGB\_DXT1() { |
| 029 | SelectedChangeTextureFormatSettings(TextureImporterFormat.DXT1); |
| 030 | } |
| 031 |  |
| 032 | [MenuItem ("Custom/Texture/Change Texture Format/RGB Compressed DXT5")] |
| 033 | **static** **void** ChangeTextureFormat\_RGB\_DXT5() { |
| 034 | SelectedChangeTextureFormatSettings(TextureImporterFormat.DXT5); |
| 035 | } |
| 036 |  |
| 037 | [MenuItem ("Custom/Texture/Change Texture Format/RGB 16 bit")] |
| 038 | **static** **void** ChangeTextureFormat\_RGB\_16bit() { |
| 039 | SelectedChangeTextureFormatSettings(TextureImporterFormat.RGB16); |
| 040 | } |
| 041 |  |
| 042 | [MenuItem ("Custom/Texture/Change Texture Format/RGB 24 bit")] |
| 043 | **static** **void** ChangeTextureFormat\_RGB\_24bit() { |
| 044 | SelectedChangeTextureFormatSettings(TextureImporterFormat.RGB24); |
| 045 | } |
| 046 |  |
| 047 | [MenuItem ("Custom/Texture/Change Texture Format/Alpha 8 bit")] |
| 048 | **static** **void** ChangeTextureFormat\_Alpha\_8bit() { |
| 049 | SelectedChangeTextureFormatSettings(TextureImporterFormat.Alpha8); |
| 050 | } |
| 051 |  |
| 052 | [MenuItem ("Custom/Texture/Change Texture Format/RGBA 16 bit")] |
| 053 | **static** **void** ChangeTextureFormat\_RGBA\_16bit() { |
| 054 | SelectedChangeTextureFormatSettings(TextureImporterFormat.ARGB16); |
| 055 | } |
| 056 |  |
| 057 | [MenuItem ("Custom/Texture/Change Texture Format/RGBA 32 bit")] |
| 058 | **static** **void** ChangeTextureFormat\_RGBA\_32bit() { |
| 059 | SelectedChangeTextureFormatSettings(TextureImporterFormat.ARGB32); |
| 060 | } |
| 061 |  |
| 062 | *// ----------------------------------------------------------------------------* |
| 063 |  |
| 064 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/32")] |
| 065 | **static** **void** ChangeTextureSize\_32() { |
| 066 | SelectedChangeMaxTextureSize(32); |
| 067 | } |
| 068 |  |
| 069 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/64")] |
| 070 | **static** **void** ChangeTextureSize\_64() { |
| 071 | SelectedChangeMaxTextureSize(64); |
| 072 | } |
| 073 |  |
| 074 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/128")] |
| 075 | **static** **void** ChangeTextureSize\_128() { |
| 076 | SelectedChangeMaxTextureSize(128); |
| 077 | } |
| 078 |  |
| 079 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/256")] |
| 080 | **static** **void** ChangeTextureSize\_256() { |
| 081 | SelectedChangeMaxTextureSize(256); |
| 082 | } |
| 083 |  |
| 084 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/512")] |
| 085 | **static** **void** ChangeTextureSize\_512() { |
| 086 | SelectedChangeMaxTextureSize(512); |
| 087 | } |
| 088 | *//Unity3D教程手册：www.unitymanual.com* |
| 089 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/1024")] |
| 090 | **static** **void** ChangeTextureSize\_1024() { |
| 091 | SelectedChangeMaxTextureSize(1024); |
| 092 | } |
| 093 |  |
| 094 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/2048")] |
| 095 | **static** **void** ChangeTextureSize\_2048() { |
| 096 | SelectedChangeMaxTextureSize(2048); |
| 097 | } |
| 098 |  |
| 099 | *// ----------------------------------------------------------------------------* |
| 100 |  |
| 101 | [MenuItem ("Custom/Texture/Change MipMap/Enable MipMap")] |
| 102 | **static** **void** ChangeMipMap\_On() { |
| 103 | SelectedChangeMimMap(**true**); |
| 104 | } |
| 105 |  |
| 106 | [MenuItem ("Custom/Texture/Change MipMap/Disable MipMap")] |
| 107 | **static** **void** ChangeMipMap\_Off() { |
| 108 | SelectedChangeMimMap(**false**); |
| 109 | } |
| 110 |  |
| 111 | *// ----------------------------------------------------------------------------* |
| 112 |  |
| 113 | **static** **void** SelectedChangeMimMap(**bool** enabled) { |
| 114 |  |
| 115 | **Object**[] textures = GetSelectedTextures(); |
| 116 | Selection.objects = new **Object**[0]; |
| 117 | **foreach** (Texture2D texture **in** textures) { |
| 118 | **string** path = AssetDatabase.GetAssetPath(texture); |
| 119 | TextureImporter textureImporter = AssetImporter.GetAtPath(path) **as** TextureImporter; |
| 120 | textureImporter.mipmapEnabled = enabled; |
| 121 | AssetDatabase.ImportAsset(path); |
| 122 | } |
| 123 | } |
| 124 | *//Unity3D教程手册：www.unitymanual.com* |
| 125 | **static** **void** SelectedChangeMaxTextureSize(**int** size) { |
| 126 |  |
| 127 | **Object**[] textures = GetSelectedTextures(); |
| 128 | Selection.objects = new **Object**[0]; |
| 129 | **foreach** (Texture2D texture **in** textures) { |
| 130 | **string** path = AssetDatabase.GetAssetPath(texture); |
| 131 | TextureImporter textureImporter = AssetImporter.GetAtPath(path) **as** TextureImporter; |
| 132 | textureImporter.maxTextureSize = size; |
| 133 | AssetDatabase.ImportAsset(path); |
| 134 | } |
| 135 | } |
| 136 |  |
| 137 | **static** **void** SelectedChangeTextureFormatSettings(TextureImporterFormat newFormat) { |
| 138 |  |
| 139 | **Object**[] textures = GetSelectedTextures(); |
| 140 | Selection.objects = new **Object**[0]; |
| 141 | **foreach** (Texture2D texture **in** textures) { |
| 142 | **string** path = AssetDatabase.GetAssetPath(texture); |
| 143 | *//Debug.Log("path: " + path);* |
| 144 | TextureImporter textureImporter = AssetImporter.GetAtPath(path) **as** TextureImporter; |
| 145 | textureImporter.textureFormat = newFormat; |
| 146 | AssetDatabase.ImportAsset(path); |
| 147 | } |
| 148 | } |
| 149 |  |
| 150 | **static** **Object**[] GetSelectedTextures() |
| 151 | { |
| 152 | **return** Selection.GetFiltered(typeof(Texture2D), SelectionMode.DeepAssets); |
| 153 | } |
| 154 | } |

ChangeTextureImportSettingsUnity3 for Unity3.x

程序代码csharp代码：

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

|  |  |
| --- | --- |
| 001 | **using** UnityEngine; |
| 002 | **using** UnityEditor; |
| 003 |  |
| 004 | *// /////////////////////////////////////////////////////////////////////////////////////////////////////////* |
| 005 | *//* |
| 006 | *// Batch Texture import settings modifier.* |
| 007 | *//* |
| 008 | *// Modifies all selected textures in the project window and applies the requested modification on the* |
| 009 | *// textures. Idea was to have the same choices for multiple files as you would have if you open the* |
| 010 | *// import settings of a single texture. Put this into Assets/Editor and once compiled by Unity you find* |
| 011 | *// the new functionality in Custom -> Texture. Enjoy! :-)* |
| 012 | *//* |
| 013 | *// Based on the great work of benblo in this thread:* |
| 014 | *// http://forum.unity3d.com/viewtopic.php?t=16079&start=0&postdays=0&postorder=asc&highlight=textureimporter* |
| 015 | *//* |
| 016 | *// Developed by Martin Schultz, Decane in August 2009* |
| 017 | *// e-mail: ms@decane.net* |
| 018 | *//* |
| 019 | *// Updated for Unity 3.0 by col000r in August 2010* |
| 020 | *// http://col000r.blogspot.com* |
| 021 | *//* |
| 022 | *// /////////////////////////////////////////////////////////////////////////////////////////////////////////* |
| 023 | **public** **class** ChangeTextureImportSettingsUnity3 : ScriptableObject { |
| 024 |  |
| 025 | [MenuItem ("Custom/Texture/Change Texture Format/Auto Compressed")] |
| 026 | **static** **void** ChangeTextureFormat\_AutoCompressed() { |
| 027 | SelectedChangeTextureFormatSettings(TextureImporterFormat.AutomaticCompressed); |
| 028 | } |
| 029 |  |
| 030 | [MenuItem ("Custom/Texture/Change Texture Format/Auto 16bit")] |
| 031 | **static** **void** ChangeTextureFormat\_Auto16Bit() { |
| 032 | SelectedChangeTextureFormatSettings(TextureImporterFormat.Automatic16bit); |
| 033 | } |
| 034 |  |
| 035 | [MenuItem ("Custom/Texture/Change Texture Format/Auto Truecolor")] |
| 036 | **static** **void** ChangeTextureFormat\_AutoTruecolor() { |
| 037 | SelectedChangeTextureFormatSettings(TextureImporterFormat.AutomaticTruecolor); |
| 038 | } |
| 039 |  |
| 040 | [MenuItem ("Custom/Texture/Change Texture Format/RGB Compressed DXT1")] |
| 041 | **static** **void** ChangeTextureFormat\_RGB\_DXT1() { |
| 042 | SelectedChangeTextureFormatSettings(TextureImporterFormat.DXT1); |
| 043 | } |
| 044 |  |
| 045 | [MenuItem ("Custom/Texture/Change Texture Format/RGB Compressed DXT5")] |
| 046 | **static** **void** ChangeTextureFormat\_RGB\_DXT5() { |
| 047 | SelectedChangeTextureFormatSettings(TextureImporterFormat.DXT5); |
| 048 | } |
| 049 |  |
| 050 | [MenuItem ("Custom/Texture/Change Texture Format/RGB 16 bit")] |
| 051 | **static** **void** ChangeTextureFormat\_RGB\_16bit() { |
| 052 | SelectedChangeTextureFormatSettings(TextureImporterFormat.RGB16); |
| 053 | } |
| 054 |  |
| 055 | [MenuItem ("Custom/Texture/Change Texture Format/RGB 24 bit")] |
| 056 | **static** **void** ChangeTextureFormat\_RGB\_24bit() { |
| 057 | SelectedChangeTextureFormatSettings(TextureImporterFormat.RGB24); |
| 058 | } |
| 059 |  |
| 060 | [MenuItem ("Custom/Texture/Change Texture Format/Alpha 8 bit")] |
| 061 | **static** **void** ChangeTextureFormat\_Alpha\_8bit() { |
| 062 | SelectedChangeTextureFormatSettings(TextureImporterFormat.Alpha8); |
| 063 | } |
| 064 |  |
| 065 | [MenuItem ("Custom/Texture/Change Texture Format/ARGB 16 bit")] |
| 066 | **static** **void** ChangeTextureFormat\_RGBA\_16bit() { |
| 067 | SelectedChangeTextureFormatSettings(TextureImporterFormat.ARGB16); |
| 068 | } |
| 069 |  |
| 070 | [MenuItem ("Custom/Texture/Change Texture Format/RGBA 32 bit")] |
| 071 | **static** **void** ChangeTextureFormat\_RGBA\_32bit() { |
| 072 | SelectedChangeTextureFormatSettings(TextureImporterFormat.RGBA32); |
| 073 | } |
| 074 |  |
| 075 | [MenuItem ("Custom/Texture/Change Texture Format/ARGB 32 bit")] |
| 076 | **static** **void** ChangeTextureFormat\_ARGB\_32bit() { |
| 077 | SelectedChangeTextureFormatSettings(TextureImporterFormat.ARGB32); |
| 078 | } |
| 079 |  |
| 080 | [MenuItem ("Custom/Texture/Change Texture Format/RGB PVRTC 2bit")] |
| 081 | **static** **void** ChangeTextureFormat\_RGB\_PVRTC\_2bit() { |
| 082 | SelectedChangeTextureFormatSettings(TextureImporterFormat.PVRTC\_RGB2); |
| 083 | } |
| 084 |  |
| 085 | [MenuItem ("Custom/Texture/Change Texture Format/RGBA PVRTC 2bit")] |
| 086 | **static** **void** ChangeTextureFormat\_RGBA\_PVRTC\_2bit() { |
| 087 | SelectedChangeTextureFormatSettings(TextureImporterFormat.PVRTC\_RGBA2); |
| 088 | } |
| 089 |  |
| 090 | [MenuItem ("Custom/Texture/Change Texture Format/RGB PVRTC 4bit")] |
| 091 | **static** **void** ChangeTextureFormat\_RGB\_PVRTC\_4bit() { |
| 092 | SelectedChangeTextureFormatSettings(TextureImporterFormat.PVRTC\_RGB4); |
| 093 | } |
| 094 |  |
| 095 | [MenuItem ("Custom/Texture/Change Texture Format/RGBA PVRTC 4bit")] |
| 096 | **static** **void** ChangeTextureFormat\_RGBA\_PVRTC\_4bit() { |
| 097 | SelectedChangeTextureFormatSettings(TextureImporterFormat.PVRTC\_RGBA4); |
| 098 | } |
| 099 |  |
| 100 | *// ----------------------------------------------------------------------------* |
| 101 |  |
| 102 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/32")] |
| 103 | **static** **void** ChangeTextureSize\_32() { |
| 104 | SelectedChangeMaxTextureSize(32); |
| 105 | } |
| 106 |  |
| 107 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/64")] |
| 108 | **static** **void** ChangeTextureSize\_64() { |
| 109 | SelectedChangeMaxTextureSize(64); |
| 110 | } |
| 111 |  |
| 112 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/128")] |
| 113 | **static** **void** ChangeTextureSize\_128() { |
| 114 | SelectedChangeMaxTextureSize(128); |
| 115 | } |
| 116 |  |
| 117 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/256")] |
| 118 | **static** **void** ChangeTextureSize\_256() { |
| 119 | SelectedChangeMaxTextureSize(256); |
| 120 | } |
| 121 |  |
| 122 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/512")] |
| 123 | **static** **void** ChangeTextureSize\_512() { |
| 124 | SelectedChangeMaxTextureSize(512); |
| 125 | } |
| 126 |  |
| 127 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/1024")] |
| 128 | **static** **void** ChangeTextureSize\_1024() { |
| 129 | SelectedChangeMaxTextureSize(1024); |
| 130 | } |
| 131 |  |
| 132 | [MenuItem ("Custom/Texture/Change Texture Size/Change Max Texture Size/2048")] |
| 133 | **static** **void** ChangeTextureSize\_2048() { |
| 134 | SelectedChangeMaxTextureSize(2048); |
| 135 | } |
| 136 |  |
| 137 | *// ----------------------------------------------------------------------------* |
| 138 |  |
| 139 | [MenuItem ("Custom/Texture/Change MipMap/Enable MipMap")] |
| 140 | **static** **void** ChangeMipMap\_On() { |
| 141 | SelectedChangeMimMap(**true**); |
| 142 | } |
| 143 |  |
| 144 | [MenuItem ("Custom/Texture/Change MipMap/Disable MipMap")] |
| 145 | **static** **void** ChangeMipMap\_Off() { |
| 146 | SelectedChangeMimMap(**false**); |
| 147 | } |
| 148 |  |
| 149 | *// ----------------------------------------------------------------------------* |
| 150 |  |
| 151 |  |
| 152 | [MenuItem ("Custom/Texture/Change Non Power of 2/None")] |
| 153 | **static** **void** ChangeNPOT\_None() { |
| 154 | SelectedChangeNonPowerOf2(TextureImporterNPOTScale.None); |
| 155 | } |
| 156 |  |
| 157 | [MenuItem ("Custom/Texture/Change Non Power of 2/ToNearest")] |
| 158 | **static** **void** ChangeNPOT\_ToNearest() { |
| 159 | SelectedChangeNonPowerOf2(TextureImporterNPOTScale.ToNearest); |
| 160 | } |
| 161 |  |
| 162 | [MenuItem ("Custom/Texture/Change Non Power of 2/ToLarger")] |
| 163 | **static** **void** ChangeNPOT\_ToLarger() { |
| 164 | SelectedChangeNonPowerOf2(TextureImporterNPOTScale.ToLarger); |
| 165 | } |
| 166 |  |
| 167 | [MenuItem ("Custom/Texture/Change Non Power of 2/ToSmaller")] |
| 168 | **static** **void** ChangeNPOT\_ToSmaller() { |
| 169 | SelectedChangeNonPowerOf2(TextureImporterNPOTScale.ToSmaller); |
| 170 | } |
| 171 |  |
| 172 | *// ----------------------------------------------------------------------------* |
| 173 |  |
| 174 | [MenuItem ("Custom/Texture/Change Is Readable/Enable")] |
| 175 | **static** **void** ChangeIsReadable\_Yes() { |
| 176 | SelectedChangeIsReadable(**true**); |
| 177 | } |
| 178 |  |
| 179 | [MenuItem ("Custom/Texture/Change Is Readable/Disable")] |
| 180 | **static** **void** ChangeIsReadable\_No() { |
| 181 | SelectedChangeIsReadable(**false**); |
| 182 | } *//Unity3D教程手册：www.unitymanual.com* |
| 183 |  |
| 184 | *// ----------------------------------------------------------------------------* |
| 185 |  |
| 186 | **static** **void** SelectedChangeIsReadable(**bool** enabled) { |
| 187 |  |
| 188 | **Object**[] textures = GetSelectedTextures(); |
| 189 | Selection.objects = new **Object**[0]; |
| 190 | **foreach** (Texture2D texture **in** textures) { |
| 191 | **string** path = AssetDatabase.GetAssetPath(texture); |
| 192 | TextureImporter textureImporter = AssetImporter.GetAtPath(path) **as** TextureImporter; |
| 193 | textureImporter.isReadable = enabled; |
| 194 | AssetDatabase.ImportAsset(path); |
| 195 | } |
| 196 | } |
| 197 |  |
| 198 |  |
| 199 | **static** **void** SelectedChangeNonPowerOf2(TextureImporterNPOTScale npot) { |
| 200 |  |
| 201 | **Object**[] textures = GetSelectedTextures(); |
| 202 | Selection.objects = new **Object**[0]; |
| 203 | **foreach** (Texture2D texture **in** textures) { |
| 204 | **string** path = AssetDatabase.GetAssetPath(texture); |
| 205 | TextureImporter textureImporter = AssetImporter.GetAtPath(path) **as** TextureImporter; |
| 206 | textureImporter.npotScale = npot; |
| 207 | AssetDatabase.ImportAsset(path); |
| 208 | } |
| 209 | } |
| 210 |  |
| 211 | **static** **void** SelectedChangeMimMap(**bool** enabled) { |
| 212 |  |
| 213 | **Object**[] textures = GetSelectedTextures(); |
| 214 | Selection.objects = new **Object**[0]; |
| 215 | **foreach** (Texture2D texture **in** textures) { |
| 216 | **string** path = AssetDatabase.GetAssetPath(texture); |
| 217 | TextureImporter textureImporter = AssetImporter.GetAtPath(path) **as** TextureImporter; |
| 218 | textureImporter.mipmapEnabled = enabled; |
| 219 | AssetDatabase.ImportAsset(path); |
| 220 | } |
| 221 | } |
| 222 | *//Unity3D教程手册：www.unitymanual.com* |
| 223 | **static** **void** SelectedChangeMaxTextureSize(**int** size) { |
| 224 |  |
| 225 | **Object**[] textures = GetSelectedTextures(); |
| 226 | Selection.objects = new **Object**[0]; |
| 227 | **foreach** (Texture2D texture **in** textures) { |
| 228 | **string** path = AssetDatabase.GetAssetPath(texture); |
| 229 | TextureImporter textureImporter = AssetImporter.GetAtPath(path) **as** TextureImporter; |
| 230 | textureImporter.maxTextureSize = size; |
| 231 | AssetDatabase.ImportAsset(path); |
| 232 | } |
| 233 | } |
| 234 |  |
| 235 | **static** **void** SelectedChangeTextureFormatSettings(TextureImporterFormat newFormat) { |
| 236 |  |
| 237 | **Object**[] textures = GetSelectedTextures(); |
| 238 | Selection.objects = new **Object**[0]; |
| 239 | **foreach** (Texture2D texture **in** textures) { |
| 240 | **string** path = AssetDatabase.GetAssetPath(texture); |
| 241 | *//Debug.Log("path: " + path);* |
| 242 | TextureImporter textureImporter = AssetImporter.GetAtPath(path) **as** TextureImporter; |
| 243 | textureImporter.textureFormat = newFormat; |
| 244 | AssetDatabase.ImportAsset(path); |
| 245 | } |
| 246 | } |
| 247 |  |
| 248 | **static** **Object**[] GetSelectedTextures() |
| 249 | { |
| 250 | **return** Selection.GetFiltered(typeof(Texture2D), SelectionMode.DeepAssets); |
| 251 | } |
| 252 | } |