using System;

using UnityEditor;

using UnityEngine;

namespace UnityStandardAssets.ImageEffects

{

[CustomEditor (typeof(VignetteAndChromaticAberration))]

class VignetteAndChromaticAberrationEditor : Editor

{

private SerializedObject m\_SerObj;

private SerializedProperty m\_Mode;

private SerializedProperty m\_Intensity; // intensity == 0 disables pre pass (optimization)

private SerializedProperty m\_ChromaticAberration;

private SerializedProperty m\_AxialAberration;

private SerializedProperty m\_Blur; // blur == 0 disables blur pass (optimization)

private SerializedProperty m\_BlurSpread;

private SerializedProperty m\_BlurDistance;

private SerializedProperty m\_LuminanceDependency;

void OnEnable ()

{

m\_SerObj = new SerializedObject (target);

m\_Mode = m\_SerObj.FindProperty ("mode");

m\_Intensity = m\_SerObj.FindProperty ("intensity");

m\_ChromaticAberration = m\_SerObj.FindProperty ("chromaticAberration");

m\_AxialAberration = m\_SerObj.FindProperty ("axialAberration");

m\_Blur = m\_SerObj.FindProperty ("blur");

m\_BlurSpread = m\_SerObj.FindProperty ("blurSpread");

m\_LuminanceDependency = m\_SerObj.FindProperty ("luminanceDependency");

m\_BlurDistance = m\_SerObj.FindProperty ("blurDistance");

}

public override void OnInspectorGUI ()

{

m\_SerObj.Update ();

EditorGUILayout.LabelField("Simulates the common lens artifacts 'Vignette' and 'Aberration'", EditorStyles.miniLabel);

EditorGUILayout.PropertyField (m\_Intensity, new GUIContent("Vignetting"));

EditorGUILayout.PropertyField (m\_Blur, new GUIContent(" Blurred Corners"));

if (m\_Blur.floatValue>0.0f)

EditorGUILayout.PropertyField (m\_BlurSpread, new GUIContent(" Blur Distance"));

EditorGUILayout.Separator ();

EditorGUILayout.PropertyField (m\_Mode, new GUIContent("Aberration"));

if (m\_Mode.intValue>0)

{

EditorGUILayout.PropertyField (m\_ChromaticAberration, new GUIContent(" Tangential Aberration"));

EditorGUILayout.PropertyField (m\_AxialAberration, new GUIContent(" Axial Aberration"));

m\_LuminanceDependency.floatValue = EditorGUILayout.Slider(" Contrast Dependency", m\_LuminanceDependency.floatValue, 0.001f, 1.0f);

m\_BlurDistance.floatValue = EditorGUILayout.Slider(" Blur Distance", m\_BlurDistance.floatValue, 0.001f, 5.0f);

}

else

EditorGUILayout.PropertyField (m\_ChromaticAberration, new GUIContent(" Chromatic Aberration"));

m\_SerObj.ApplyModifiedProperties();

}

}

}