using System;

using UnityEngine;

// This class implements simple ghosting type Motion Blur.

// If Extra Blur is selected, the scene will allways be a little blurred,

// as it is scaled to a smaller resolution.

// The effect works by accumulating the previous frames in an accumulation

// texture.

namespace UnityStandardAssets.ImageEffects

{

[ExecuteInEditMode]

[AddComponentMenu("Image Effects/Blur/Motion Blur (Color Accumulation)")]

[RequireComponent(typeof(Camera))]

public class MotionBlur : ImageEffectBase

{

public float blurAmount = 0.8f;

public bool extraBlur = false;

private RenderTexture accumTexture;

override protected void Start()

{

if (!SystemInfo.supportsRenderTextures)

{

enabled = false;

return;

}

base.Start();

}

override protected void OnDisable()

{

base.OnDisable();

DestroyImmediate(accumTexture);

}

// Called by camera to apply image effect

void OnRenderImage (RenderTexture source, RenderTexture destination)

{

// Create the accumulation texture

if (accumTexture == null || accumTexture.width != source.width || accumTexture.height != source.height)

{

DestroyImmediate(accumTexture);

accumTexture = new RenderTexture(source.width, source.height, 0);

accumTexture.hideFlags = HideFlags.HideAndDontSave;

Graphics.Blit( source, accumTexture );

}

// If Extra Blur is selected, downscale the texture to 4x4 smaller resolution.

if (extraBlur)

{

RenderTexture blurbuffer = RenderTexture.GetTemporary(source.width/4, source.height/4, 0);

accumTexture.MarkRestoreExpected();

Graphics.Blit(accumTexture, blurbuffer);

Graphics.Blit(blurbuffer,accumTexture);

RenderTexture.ReleaseTemporary(blurbuffer);

}

// Clamp the motion blur variable, so it can never leave permanent trails in the image

blurAmount = Mathf.Clamp( blurAmount, 0.0f, 0.92f );

// Setup the texture and floating point values in the shader

material.SetTexture("\_MainTex", accumTexture);

material.SetFloat("\_AccumOrig", 1.0F-blurAmount);

// We are accumulating motion over frames without clear/discard

// by design, so silence any performance warnings from Unity

accumTexture.MarkRestoreExpected();

// Render the image using the motion blur shader

Graphics.Blit (source, accumTexture, material);

Graphics.Blit (accumTexture, destination);

}

}

}