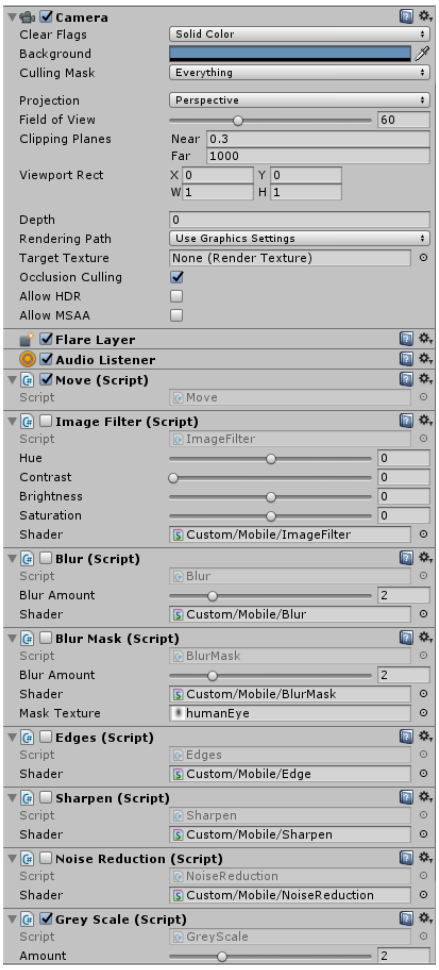
MOBILE COLOR GRADING

In this package u have 7 color grading and image filtering shaders which can be whether applied for mobile or desktop applicaitons. Most of the shaders were optimized to run smothly on mobile, with idea of keeping the proper quality.

How to apply:

1. **Add any of the effect which you want to apply to Camera object**



☑ Camera	
Clear Flags	Solid Color
Background	
Culling Mask	Everything
Projection	Perspective
Field of View	60
Clipping Planes	Near 0.3 Far 1000
Viewport Rect	X 0 Y 0 W 1 H 1
Depth	0
Rendering Path	Use Graphics Settings
Target Texture	None (Render Texture)
Occlusion Culling	☑
Allow HDR	□
Allow MSAA	□
☑ Flare Layer	
☑ Audio Listener	
☑ Move (Script)	
Script	Move
□ Image Filter (Script)	
Script	ImageFilter
Hue	0
Contrast	0
Brightness	0
Saturation	0
Shader	Custom/Mobile/ImageFilter
□ Blur (Script)	
Script	Blur
Blur Amount	2
Shader	Custom/Mobile/Blur
□ Blur Mask (Script)	
Script	BlurMask
Blur Amount	2
Shader	Custom/Mobile/BlurMask
Mask Texture	humanEye
□ Edges (Script)	
Script	Edges
Shader	Custom/Mobile/Edge
□ Sharpen (Script)	
Script	Sharpen
Shader	Custom/Mobile/Sharpen
□ Noise Reduction (Script)	
Script	NoiseReduction
Shader	Custom/Mobile/NoiseReduction
☑ Grey Scale (Script)	
Script	GreyScale
Amount	2

2. Attach the shader for the script relatively to its name F.e: Greyscale.cs-> Grayscale.shader etc.

You can find all of the scripts attaches to camera in the Example scene of the Color Grading package.

Shaders

- **IMAGE FILTER –** this shader is for color grading, allowing to affect the final with contrast, hue, brightness and saturation filters.
- **SHARPEN –** simple sharpening filter
- **EDGE –** darken the inners of the image and exhibits only the edges of the objects in the scene. Used mostly for image recognition purposes. P.s better to use along with greyscale to whiten the edges.
- **BLUR –** very efficient blur rendered with 2 iteration algorithm in a single pass.
- **BLUR MASK –** same efficient blur, with mask applied, work a bit slower than simple blur if mask is completely white(better to use just blur for that case)
- **NOISE REDUCTION –** used 3 iterational gauusian blur for noise reduction, works efficiently on mobile devices. Mostly used after sharpen to blur out the oversharpened edges.
- **GREYSCALE –** simple greyscaling filter. Works faster than image filter with 0 saturation(greyscale).

Tested in the 40k polugonal scene, with 68 materials applied to 50 gameobjects and one Directional light we have this results on Meizu M2 Note(Octa-core 1.3 GHZ ARM Cortex-A53, Mediatek MT6753, GPU Mali-T720MP3, RAM 2 GB)

Filters work approximately fps depending on complexitysome of them were running 40-45 fps(image filter, noise reduction) others 45-55 fps.