Filmic Tonemapping DELUXE

Documentation V. 1.2

Support
mikael.paroxe@gmail.com

Table of contents

RELEASE NOTES

Version 1.2

- MAJOR FEATURE: Histogram based eye adaptation have been added
- The documentation have been updated
- A curve presets combo box have been added to the tonemapper

Version 1.1

- The editor have been changed to a more intuitive and user friendly interface
- Luminance tonemapping have been added
- Extended luminance tonemapping have been added

When should a tonemapper be used?

Tonemapping is very useful when the HDR option is enabled. HDR allows RGBA values over one (over 255) to be rendered in the RenderTexture of the Unity camera but those values can't be treated by the standard computer screens and a tonemapper is needed to make those values visible.

Tonemapping can also be used as a color correction tool to make a scene look better by adding depth to the colors where it matters, even if HDR is disabled. A filmic tonemapper can adjust colors in a way that can't be done with color correction tools.

About Filmic Tonemapping DELUXE

Filmic Tonemapping DELUXE is inspired by the S-shaped tonemapping algorithm used in the film industry to add a more definitions to the raw image captured by the camera. Nowadays, most AAA games uses similar algorithms to add depth to the rendered image, and to make HDR values visible.

Filmic Tonemapping DELUXE offers an easy and intuitive way to edit the tonemapping curve to fits your game and application needs.

Since version 1.2, a histogram based eye adaptation effect have been added. This effect can simulate how the eye react to lights by changing the exposure relatively to the average brightness of a specified range of pixel intensity.

HOW TO: Add Filmic Tonemapping to your camera

Go to ***Component/Filmic Tonemapping/Add Tonemapper*** *to add the tonemapper to the selected camera*

HOW TO: Add Eye Adaptation to your camera

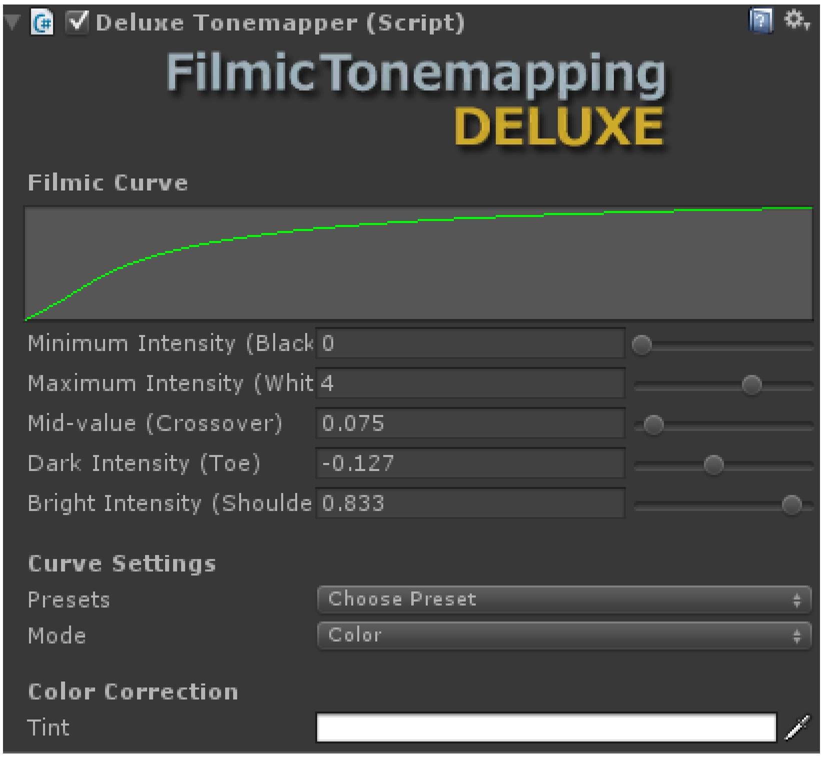
Go to ***Component/Filmic Tonemapping/Add Eye Adaptation*** *to add the eye adaptation component to the selected camera. You must place the eye adaptation component* ***before*** *filmic tonemapping in the component list.*

Where should Tonemapping and Eye Adaptation be placed in the camera post process pipeline?

The Eye Adaptation component should be placed before the tonemapper, while the tonemapper should be place at the end of your graphics pipeline, just before color correction (if you use any). In-between the eye adaptation and the tonemapper, you can place some effects that are dependent on the pixels luminance (i.e. a bloom effect). Bloom could also be placed right before the eye adaptation effect, but I personally think it looks better when placed in-between eye adaptation and tonemapping.

Filmic Tonemapping

Tonemapping parameters



Deluxe Tonemapper (Script)

FilmicTonemapping DELUXE

Filmic Curve

Minimum Intensity (Black 0

Maximum Intensity (Whit 4

Mid-value (Crossover) 0.075

Dark Intensity (Toe) -0.127

Bright Intensity (Shoulde 0.833

Curve Settings

Presets Choose Preset

Mode Color

Color Correction

Tint

Filmic Curve

Min Intensity (Black Point) [0,100]: The color range start value (minimum value)

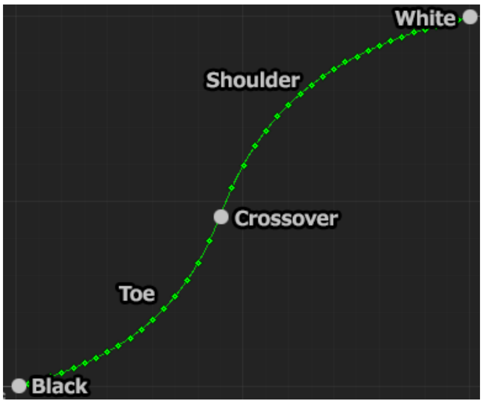
Black Intensity (Toe Strength) [-1,1]: The strength of the Toe curve (how the dark values will be rendered).

Mid-Value (Crossover Point) [0,1]: The intersection point between the Toe and the Shoulder curve. This value is a linear interpolation between the black point (=0) and the white point (=1).

Bright Intensity (Shoulder Strength) [-1,1]: the strength of the Shoulder curve (how the bright values will be rendered)

Max Intensity (White point)[0,100]: The color range end value (maximum value)

Luminance Saturation[0,6] (Extended Luminance mode only) : The white point of the luminance. This is an extension to the luminance processing to preserve white values.



Curve Settings

Presets: You can select a predefined settings as a starting point to configure the tonemapper

Tonemapping processing mode

Since version 1.1, the tonemapping curve can either be computed on pixel luminance or on pixel colors. An extended luminance mode have been added because classical luminance processing will quickly saturate colors and white colors won't be preserved. The extended luminance mode will fix those issues by introducing a white luminance saturation point.

In the films industry, tonemapping is processed on colors. However, applying some luminosity correction on the image add more tools to add depth to the image in a unique way. You can even stack luminosity correction and color correction together.

Tonemapping Guidelines

In most cases, the black point should be 0. You rarely want to saturate your black colors.

If you are not using HDR, the white point value should not be set to a value greater than 1. In this context, the render textures used in a non-HDR context will saturate values beyond 1 anyway.

If you are using HDR, a good value for the white point could be around 4-5.

To get a great contrast between your dark colors and your bright colors, the crossover point value should be between 0.001 and 0.2.

You can visualize the curve by clicking on the curve editor located in the component inspector, but you won't be able to edit the curve that way.

Eye Adaptation

How it works

There is two popular ways to compute eye adaptation. The most straightforward way is to compute the average luminance of the source image. The exposure is then generated by offsetting this value and this how it is done in the standard Unity eye adaptation effect (and probably all the others Unity eye adaptation implementations).

This method has two big fallbacks: it causes unstable exposure adaptation since even a small part of the screen can affect the computed exposure, and it will create poor contrast because it has no notion of how bright are the brightest pixels are and how dark are the darkest pixel. It only knows the average luminance of the source image.

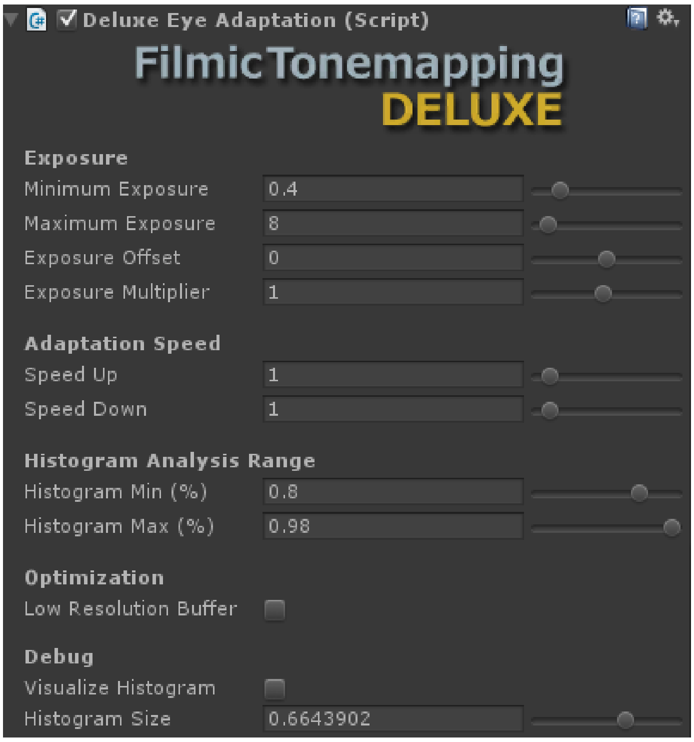
The second method (Filmic Tonemapping Deluxe, Cryengine and Unreal Engine use this one), is based on a GPU generated histogram. The histogram contains the number of pixels inside every luminance buckets (in this case, there are 64 different buckets representing 64 different luminance range).

Based on this histogram, a target exposure is generated by computing the average luminance of a specified pixel range. The pixel range start at the XX% (i.e. 80%) brightest pixels and it ends at the XX% (i.e. 98%) brightest pixels.

As an example, with this method, if the camera is moving inside a medieval style corridor with torches on the sides, the exposure will likely stay stable since the top XX% most brightest pixels will have similar values every frames (until the torches gets far away). With the average luminance method, the computed exposure would get unstable since the average luminance of all the pixels would change every frames. And you really want to know what luminance are the brightest pixels to compute the optimal whitest point.

The histogram based method ensure a better image stability since only the most meaningful pixels luminance are averaged. It knows where the brightest pixels are located, and as a consequence, it gives a better image contrast.

Eye Adaptation parameters



☑ Deluxe Eye Adaptation (Script)

FilmicTonemapping DELUXE

Exposure

Minimum Exposure	0.4
Maximum Exposure	8
Exposure Offset	0
Exposure Multiplier	1

Adaptation Speed

Speed Up	1
Speed Down	1

Histogram Analysis Range

Histogram Min (%)	0.8
Histogram Max (%)	0.98

Optimization

Low Resolution Buffer	□

Debug

Visualize Histogram	□
Histogram Size	0.6643902

Exposure

Minimum Exposure: The minimum exposure the effect will adapt to.

Maximum Exposure: The maximum exposure the effect will adapt to.

Exposure Offset: An offset added to the computed target exposure

Exposure Multiplier: A multiplier to the result exposure

Speed

Speed Up: The speed at which the effect will adapt from dark to bright

Speed Down: The speed at which the effect will adapt from bright to dark

Ideally, in a fast paced game, the speed could be around 5-6 (a racing game, or FPS game). In a slower paced game, the speed could be around 1 (exploration, adventure).

Histogram Analysis Range

Histogram Min (%): The minimum threshold in terms of percentage of pixels at which the average luminance will be computed (A good value should be between 70% and 90%).

Histogram Max (%): The maximum threshold in terms of percentage of pixels at which the average luminance will be computed (A good value should be > 96%).

Histogram Luminance Range: The range of pixel luminance to be analyzed. This parameter only appears if Filmic Tonemapping is not set on the camera, otherwise it will be set to the tonemapper white range.

Read the “how it works” section for more details.

Optimization

Low Resolution Buffer: Compute the histogram from a lower resolution buffer. You should get a couple more FPS from this with some small lost in precision. Ideal for lower end hardware.

Debug

Visualize Histogram: If enabled, the generated histogram will be drawn of the screen. It's useful to calibrate your luminance range.

Histogram Size: The size of the histogram on the screen.