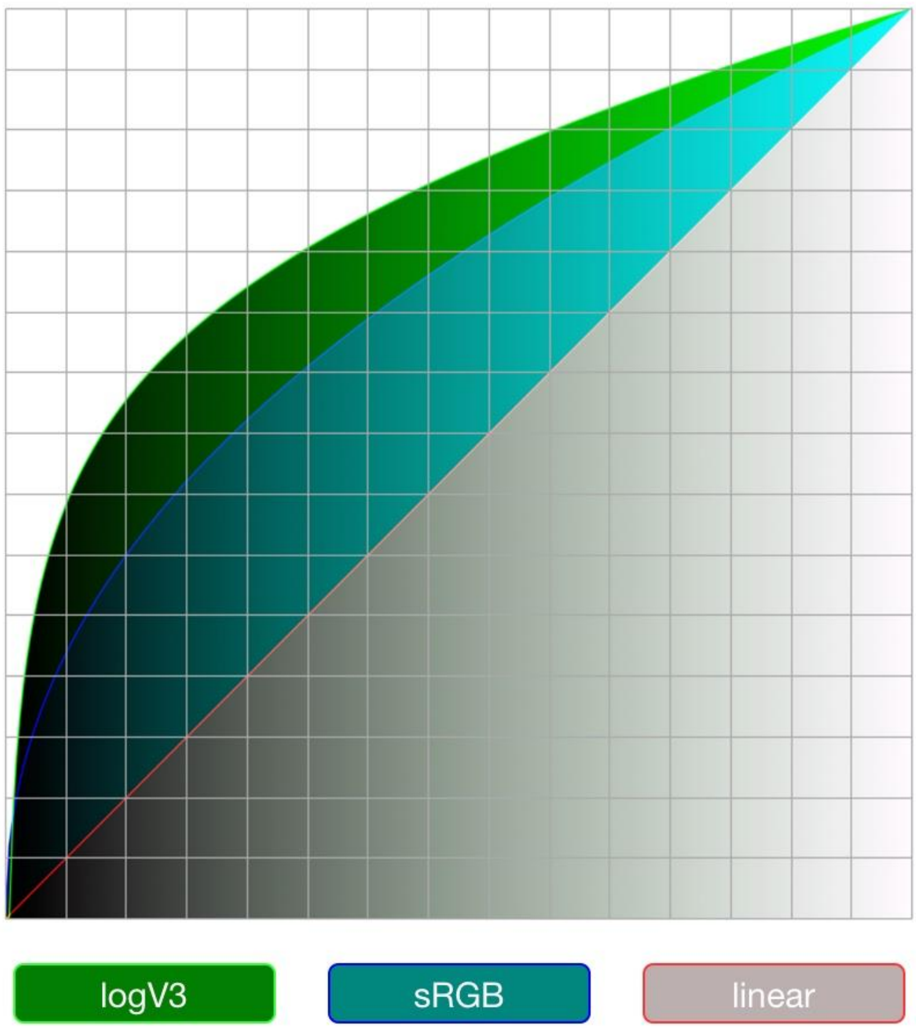
Revision History

Date	Author	Note
December 9, 2020	Christopher Cohen	Initial Draft
December 10, 2020	Christopher Cohen	• Added Curves Diagram to ‘Abstract’ section • Added “Benefits” section
December 14, 2020	Christopher Cohen	Added ‘Linearization’ section
April 5, 2021	Christopher Cohen	Added ‘HLG Conversion (Apple Only)’ section

Abstract

Filmic's 10-bit Logarithmic Encoding scheme (hereafter "LogV3") is a gamma transfer function that protectively remaps a linear (or linearized) image buffer.



LogV3 is intended to be used with 4:2:0 10-bit footage archived with a Group of Pictures (hereafter "GoP") video codec. Given the aggressive compression goals of GoP encoding, the tonal information of the image must be reorganized toward the mid-tones in order to avoid data loss. In addition to gamma vectorization, the chroma (color information independent of luminance) is remapped *independently* of the gamma response. This is done to minimize the appearance of unsightly artifacts in the shadows.

Benefits

The benefits of LogV3 are threefold: 1.) maximal dynamic range; 2.) high fidelity tonal gradation; and 3.) color accuracy.

1. ## Dynamic Range

   Note that logV3 does not *increase* dynamic range. It preserves it. By preemptively remapping the gamma before encoding, the resulting GoP file can better express the dynamic range of the raw source. When imported into a nonlinear editor (hereafter “NLE”), a LUT can be applied to restore perceptual color accuracy. This provides an excellent baseline for additional correction and modification.

2. ## Tonal Gradation

   Tonal gradation is preserved in the midtones in part due to the hybrid nature of the gamma spline. Put simply, the diagonality of the spline facilitates better gradient expression than the near horizontal “plateau” of a pure logarithm.

3. ## Color Accuracy

   Color accuracy also benefits from the profile of this hybrid log/gamma spline - particularly in the highlights. Compared to a typical log curve, highlights are afforded generous latitude on the curve: making them less prone to chromatic shifts or channel clipping.

Transfer Function

LogV3 linearly blends a binary logarithm with a gamma function of 3.45. This is rendered in the following pseudocode where a linear input is assumed:

```
vec3 l = (log2(inVec) + 8.f) * 0.125

vec3 x = mix(max(vec3(0.f), l), inVec, t: 0.1267)

vec3 y = vec3(0.28985507246)

vec3 outVec = max(vec3(0.f), mix(x, pow(inVec, y), t: inVec))
```

Linearization

LogV3 can be linearized with a function derived from the following pseudocode:

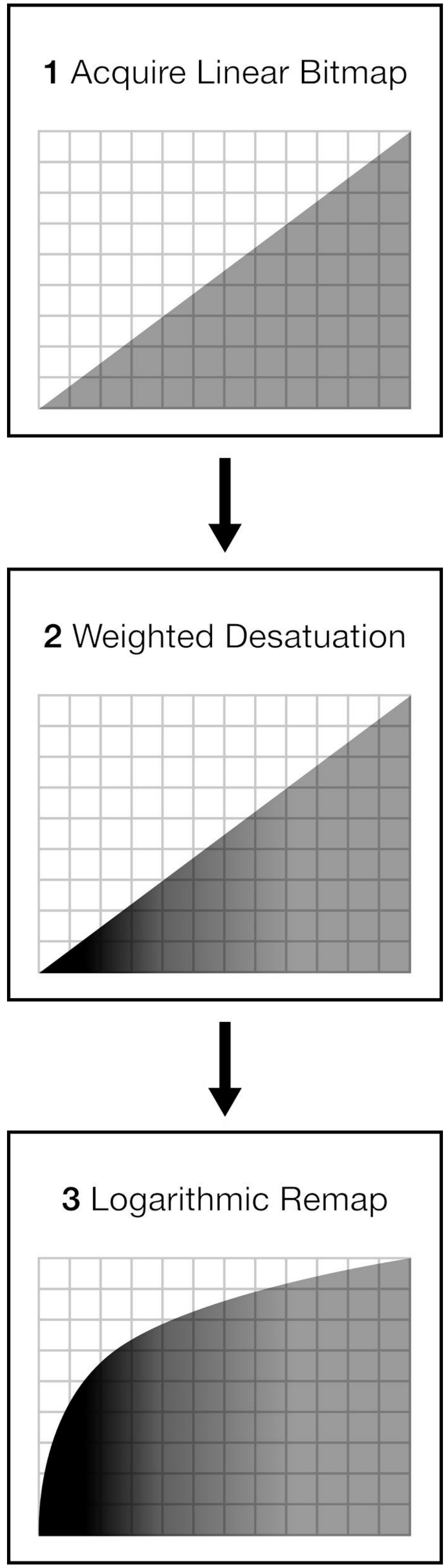
```
vec3 u = exp2((inVec / 0.125) - 8.f)

vec3 x = vec3(1.1612159730893894)

vec3 y = vec3(0.6090138106343165)

vec3 outVec = pow(u, mix(x, y, t: inVec))
```

HLG Conversion (Apple Only)

On Apple devices, dynamic tone mapping can make strict linearization challenging. Assuming typical conditions, LogV3 can be converted to an approximation of Hybrid Log Gamma (HLG) with a function derived from the following pseudocode:

```
vec3 w = vec3(0.2126, 0.7152, 0.0722)

vec3 l = vec3(dot(inVec, w))

float s = mix(l, inVec, t: 1.25)

vec3 g = vec3(1.48)

vec3 outVec = pow(s,g)
```

Remapping Process



1. Acquire Linear Bitmap

a. Acquire Bitmap

Acquire a raw bitmap directly from the device's camera module at a stage before the image is encoded.

b. Linearize Bitmap if needed

If a linear bitmap is not accessible, the inbound bitmap will be remapped to linear gamma using a shader or a lookup table.

2. Apply Luminance-Weighted Desaturation

a. Calculate Relative Luminance

In order to minimize chroma noise in critically underexposed regions of the buffer, the linear source is progressively desaturated with a relative luminance weight.

The relative luminance for an RGB pixel is produced by the following function:

$$\boldsymbol{p} = dot(\boldsymbol{s}, vec3(0.2126, 0.7152, 0.0722))$$

Where $\boldsymbol{s}$ represents a linear RGB vector and where $\boldsymbol{p}$ represents relative luminance.

b. Apply Weighted Saturation To Linear Buffer

With the intent of progressively desaturating the very darkest pixels (where chrominance typically decomposes in low bit-depth images), the linear RGB vector is blended with a linear monochromatic vector using a weight produced by the following formula:

$$\boldsymbol{o} = 0.16667 \times ln(\boldsymbol{p}) + 1.0$$

where $\boldsymbol{o}$ represents a relative-luminance weighted saturation bias.