

OPPO O-Log

Logarithmic Color Space

SPECIFICATION

版本修订 Document Version History		
版本 Version	作者 Author(s)	修订日志 Change Note
2025-5-31	赖泽民 Thom Qu	初始化 Initial document version
2025-9-29	Thom Qu	ACES

目录 Table of Contents

前言

1 简介 Introduction

2 编码 Hardware Encoding

2.1 编码格式 Encoding Format

2.2 编码质量 Encoding Quality

3 传递函数曲线 OPPO O-Log Curve

3.1 编码函数 Encoding Function

3.2 解码函数 Decoding Function

4 色彩空间 OPPO Wide Gamut

5 动态范围 Dynamic Range

6 ACES 转换 ACES transform

前言

Log 视频是为了最大化保留视频信息，在视频预处理后获得类似照片 RAW 图像的效果（Log 视频对图像做了简单的处理，与照片 DNG 更相似）。

由于视频片段对于图像原始信息处理量远高于单张照片，通常需要借助专用硬件模块协助处理。当前的设备在拍摄端都集成了专用硬件图像处理模块，因此 Log 视频在前端会进行一部分的预处理（Log 视频并不是普通的每帧视频 raw 压缩，而是对视频 raw 进行了基础预处理）。

视频图像数据处理过程使用对数曲线和宽色域。同时通过拟合人眼对于亮度的感知（指数变化）和传感器特性（线性变化），结合传感器图像的数据位宽（10bit ~ 14bit），使用了基于对数的 Log 曲线进行压缩（14bit 原始线性数据，压缩到 10bit 的 Log 数据。此部分为有损压缩，只存在量化误差）。

使用 Log 视频配置的目的是在捕获时以最少的处理来最大化地保存视频数据。Log 视频不适用于直接观看，而是作为后期制作工作流中使用的源剪辑。

1 简介 Introduction

本文档描述了 OPPO O-Log 对数色彩空间（以下简称 O-Log）及其应用场景。

OPPO O-Log 专为移动影像小型传感器增强动态范围的编码精度与制作适用性而设计。

O-Log 曲线是 OPPO 摄像机使用的一种对数图像编码方式。这种编码具有类似人眼感知的灰度特性。但由于数字摄像机与人眼感知存在本质差异，二者的色彩特性仍有不同。

对数编码意味着：在较宽范围内，以档位测量的曝光值与编码信号之间呈线性（直线）关系。每增加一档曝光，信号值就会增加相同幅度。曲线的这段斜率被称为伽马值。可以看到曲线底部存在趾部区域，这是因为传感器对低光量级的量化精度无法达到高光区域的水平。该曲线的整体形态与人眼感知曲线十分相似。

2 硬件编码 Hardware Encoding

本节概述了摄影机内部使用的 O-Log 硬件编码曲线。提供的硬件规范仅作为与 O-Log 传感器信号编码相关的参考材料，并非用于软件实现。

2.1 编码格式 Encoding Format

为了兼顾存储和编码效率，OPPO O-Log 视频使用了有损压缩编码 (H.265 YUV420)

。

为了便于后期对素材进行二次调色创作。O-Log 视频处理色域使用通用的广色域 BT2020 转换矩阵进行处理。

2.2 编码质量 Encoding Quality

视频编码码率使用动态码率进行。参考标准为 4K60fps 编码码率 120Mbps ($\pm 20\%$)

。

拍摄 1 分钟 O-Log 视频占用存储空间：

1080P30fps 约 240MB，1080P60fps 约 360MB，1080P120fps 约 540MB，
4K30fps 约 600MB，4K60fps 900MB，4K120fps 约 1350MB。

3 传递函数 OPPO O-Log Curve

本节正式开始定义 OPPO O-Log——这是一种基于场景参考的对数色彩空间，由传递函数 OPPO O-Log 曲线和原色系统 OPPO Wide Gamut (以下简称 O-Gamut) 共同构成。

3.1 编码函数 Encoding Function

O-Log 曲线实际上是一组针对不同曝光指数的曲线簇。默认自动曝光会将实际场景 18%反射率的部分映射到 O-Log 值 0.4901589 (近似 502/1023)。下图以线性-线性坐标展示了 O-Log 曲线，其中横轴代表传感器信号 (按照 10EV 输入信号归一到 1)。

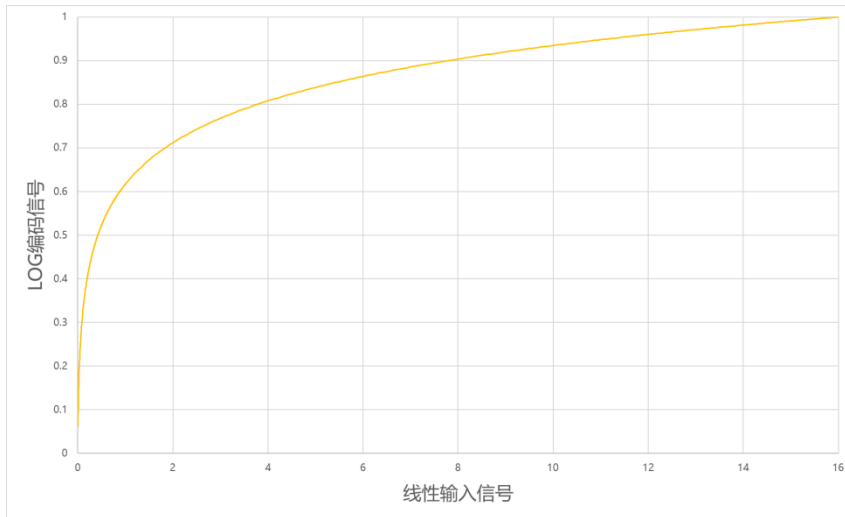


图 1：应用于线性信号的硬件编码曲线

场景反射率 Scene reflection	线性输入信号 Linear Signal	编码信号 Log Signal	
		归一化浮点值	10 bit 整型 (full range)
0%	0	0.0631271	64
18%	0.01125	0.3895463	399
39%	0.0244	0.4901589	502
1600%	1	1	1023

表 1：应用于线性输入信号的 Log 曲线编码信号

Log 视频在实际场景中使用的压缩曲线会根据场景亮度进行动态调整。为了便于理解，下图中仅展示一条完整的范围的 Log 曲线。

编码函数定义如下：

$$P = f(R) = \gamma * \log_e (X + \beta) + \delta , 0 \leq R \leq 16$$

其中

$$\gamma = 0.139$$

$$\beta = 0.019$$

$$\delta = 0.614$$

e 为自然对数，取近似值 2.7182818

整个编码函数使用单段函数完成映射。

3.2 解码函数 Decoding Function

解码函数可用于还原 Log 画面信息到线性场景反射信号，由以下方程定义：

$$R = f^{-1}(P) = e^{(P - \delta) / \gamma} - \beta , R_0 \leq P \leq 1$$

其中

$$R_0 = 0.0631271$$

$$\gamma = 0.139$$

$$\beta = 0.019$$

$$\delta = 0.614$$

e 为自然对数，取近似值 2.7182818

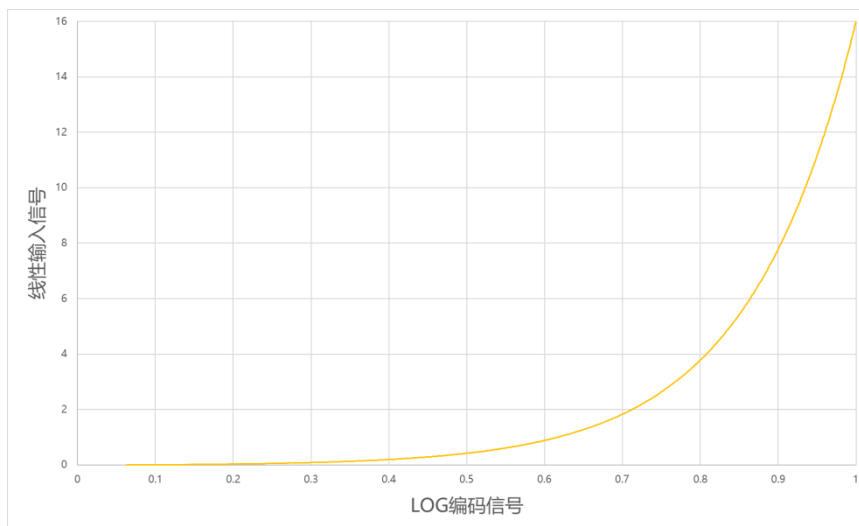


图 2：应用于 Log 信号的硬件解码曲线

4 色彩空间 OPPO Wide Gamut

色彩空间是对颜色的组织方式。O-Gamut 采用 ITU-R BT.2020-2 规范一致的 RGB 原色 (RGB primaries) 与色彩转换矩阵。白点与 D65 标准光源一致，所有色度值均使用 CIE 1931 2° 标准观察者比色法进行规定。

CIE 色度坐标		
	X	Y
红色原色 (R)	0.708	0.292
绿色原色 (G)	0.170	0.797
蓝色原色 (B)	0.131	0.046
白点 (D65)	0.3127	0.3290

RGB 和亮度-色度 YC_bC_r 表示之间的转换由以下公式控制：

$$Y = 0.2627 * R + 0.6780 * G + 0.0593 * B$$

$$C_b = (B - Y) / 1.8814$$

$$C_r = (R - Y) / 1.4746$$

5 动态范围 Dynamic Range

5.1 环境动态范围 Dynamic Range in the Scene

图像中的色调值，用物理术语来说，就是场景中反射或发射不同光量的表面。表面亮度以“光亮度”（luminance）衡量，其国际单位是坎德拉每平方米（ cd/m^2 ）。在中国通常使用同义词“尼特”（nits）表示。当最亮物体光亮度为 $4096 \text{ cd}/\text{m}^2$ ，最暗物体为 $0.5 \text{ cd}/\text{m}^2$ 时，场景的对比度或动态范围即为 8192:1（ $4096:0.5$ ）。摄影师和电影摄影师用“档位”（stops）衡量这个比率，本例中为 13 档（ $2^{13}=8192$ ）。

点测光表测量的物理量就是光亮度，但测光表会以选定 ISO 感光度对应的光圈和曝光时间组合来呈现结果。场景的绝对亮度相对次要，因为它可以通过多种方式补偿：调整光圈、使用中性密度（ND）滤镜，或调整相机的曝光指数（EI）设置。

若上例中两个物体的光亮度都增加一倍（8192:1 而非 4096:0.5），对比度仍保持不变。缩小一档光圈在色调范围方面会产生相同效果。场景中各元素的相对光亮度会线性转换为传感器上的曝光量——接收最多光线的感光单元将比接收最少光线的感光单元多 8192 倍的光量。

5.2 传感器动态范围 Dynamic Range of the Sensor

传感器的饱和信号与灵敏度阈值之比即为动态范围。这一概念并非 OPPO 首创，而是符合 ISO 157392 标准或 EMVA Standard 12883 标准中概述的通用工程实践规范。

工程师根据线性原始信号值来测量传感器的动态范围。而电影摄影师更关注摄像机的动态范围，因此将摄像机的图像处理流程纳入考量更为合理。大多数摄像机会将图像处理为类对数 RGB 编码（如 OPPO 采用 O-Log 曲线）。当通过测试图的目视检查来推导动态范围时，这就产生了另一个问题——所有厂商使用不同的编码曲线（甚至不同版本间也存在差异），导致测量结果难以直接比较。

要对信噪比（SNR）进行定量测量，需将图像数据转换回线性域。OPPO 官方公布的摄像机动态范围值正是通过这种方法测得：先将图像处理为 O-Log 格式，再用相应曲线的逆函数将图像数据线性化。基于这些线性数据，我们计算每次曝光的信噪比，从而确定灵敏度阈值和饱和点。这里要注意，小型传感器由于技术方案的限制，在不同传感器、分辨率以及不同帧率下，动态范围会有差异，下表按照最大动态范围展示。

机型 Products	动态范围 Dynamic Range (SNR=1)
OPPO Find X8 Ultra	14 stops
OnePlus 15	13 stops
OPPO Find X9	14 stops
OPPO Find X9 Pro	14 stops

关于动态范围：工程师将灵敏度阈值视为信号下限，此时噪点强度与信号相当。但对摄影师而言，这种高相对噪点水平可能难以接受。除感光元件动态范围外，还需考量其他特性，例如暗部不应出现线性噪波或偏色现象。

部分从业者倾向采用 SNR=2 作为动态范围测量下限，该标准得出的数值会比 SNR=1 的标称值至少低 1 档曝光。

6 ACES 转换 ACES transform

```
// O-Log REC2020 to ACES AP0 CTL

//define rec2020_to_xyz make_mat3(make_float3(0.6370f, 0.1446f, 0.1689f),
make_float3(0.2627f, 0.6780f, 0.0593f), make_float3(0.0f, 0.0281f, 1.0610f))

//define d65_to_d60_cat02 make_mat3(make_float3(1.01174414f, 0.00770577991f, -
0.0157216747f), make_float3(0.00555788933f, 1.00153586f,-0.00626219941f),
make_float3(-0.000334059457f,-0.00104828776f, 0.927569778f))

//define xyz_to_acesAP0 make_mat3(make_float3(1.0498110175f, 0.0f, -0.0001f),
make_float3(-0.4959030231f, 1.3733130458f, 0.0982400361f), make_float3(0.0f, 0.0f,
0.9912520182f))

// OLog Curve Encoding Function
float relativeSceneLinearToNormalizedOLog( float x) {
    return (0.139f * log(x * 16+ 0.019f)/log(_expf(1)) + 0.614f);
}

// OLog Curve Decoding Function
float normalizedOLogToRelativeSceneLinear( float x) {
    return (_expf((x - 0.614f)/0.139f) - 0.019f)/16.0f*7.37235f;;
}

void OLogToACES
(   input varying float rIn,
    input varying float gIn,
    input varying float bIn,
    input varying float aIn,
    output varying float rOut,
    output varying float gOut,
    output varying float bOut,
    output varying float aOut)
{
```

```
float r_lin = normalizedOLogToRelativeSceneLinear(rIn);
float g_lin = normalizedOLogToRelativeSceneLinear(gIn);
float b_lin = normalizedOLogToRelativeSceneLinear(bIn);

float x_D65 = r_lin * 0.6370 + g_lin * 0.1446 + b_lin * 0.1689;
float y_D65 = r_lin * 0.2627 + g_lin * 0.6780 + b_lin * 0.0593;
float z_D65 = r_lin * 0.0 + g_lin * 0.0281 + b_lin * 1.0610;

float x_D60 = x_D65 * 1.01174414 + y_D65 * 0.00770577991 + z_D65 * -0.0157216747;
float y_D60 = x_D65 * 0.00555788933 + y_D65 * 1.00153586 + z_D65 * -
0.00626219941;
float z_D60 = x_D65 * -0.000334059457 + y_D65 * -0.00104828776 + z_D65 *
0.927569778;

rOut = x_D60 * 1.0498110175 + y_D60 * 0.0 + z_D60 * -0.0001;
gOut = x_D60 * -0.4959030231 + y_D60 * 1.3733130458 + z_D60 * 0.0982400361;
bOut = x_D60 * 0.0 + y_D60 * 0.0 + z_D60 * 0.9912520182;
aOut = 1.0;

}
```