

# 全彩屏校正软件设计

卿培

qingpei@sansitech.com

2014 年 9 月 24 日

# 目录

1	图像采集	4
2	显示质量评价	5
2.1	亮度均匀性 . . . . .	5
2.2	色彩准确度 . . . . .	6
3	亮度校正	7

## Todo list

了解这方面的现有标准 . . . . .	5
优化目标的选择也需要查一查显示器生产厂家是如何做的。 . . . .	6

# 1 图像采集

待整理。

## 2 显示质量评价

目前我们关注两个方面的显示质量：亮度均匀性和色彩准确度。

### 2.1 亮度均匀性

定义 2.1. 全彩屏上一个像素  $P$  的亮度  $L_P$  由采集到的照片中该像素范围内各个像素  $p_i$  在 CIELAB 空间  $L$  通道的几何平均值<sup>1</sup>表示。

$$L_P = \left( \prod_{p_i \in P} L_{p_i} \right)^{1/n} \quad (2.1)$$

定义 2.2. 全彩屏的亮度均匀性为其中最大亮度像素与最小亮度像素的亮度比。

$$Luminance\ uniformity = \frac{L_{max}}{L_{min}} \quad (2.2)$$

理想的亮度均匀性为 1。这个比值越大，表示亮度均匀性越差。

这时需要一个检验标准来判定亮度均匀性是否合格，这个标准可以是 1.1，可以是 1.3，当然也可以是 1.05 等很严格的数值。

了解这方面的现有标准

另外一种评价方式是基于区域的。将一个显示屏分为  $N$  个区域，每个区域求出各自的平均亮度  $L_i$ ，所有区域有一个全局的平均亮度  $L_{avg}$ ，此时给出一个评价标准  $\delta$ ，要求各个区域都符合标准。

$$|L_i - L_{avg}| < \delta \quad (2.3)$$

<sup>1</sup>相比于算术平均，几何平均对少量噪点更不敏感。

同样，这里的  $\delta$  如何取值与上文一样需要商榷。

## 2.2 色彩准确度

**定义 2.3.** 全彩屏上一个像素  $P_i$  的颜色  $(L_{P_i}, a_{P_i}, b_{P_i})$  为采集到的照片中该像素范围内各个像素  $p_j$  在 CIELAB 空间  $L, a, b$  通道上的几何平均值。

**定义 2.4.** 全彩屏上一个像素  $P_i$  显示某个参考色时的色差  $\Delta E_{P_i, color}$  为该像素的颜色与参考色的颜色由 CIE DE2000<sup>2</sup>定义的色差。

理想情况下色差为 0，数值越大表示偏色越严重。

**定义 2.5.** 全彩屏的色彩准确度分为两项，一是平均色差  $\Delta E_{avg, color}$ ，二是最大色差  $\Delta E_{max, color}$ 。

$$\Delta E_{avg, color} = \frac{\sum_{i=1}^N \Delta E_{P_i, color}}{N} \quad (2.4)$$

$$\Delta E_{max, color} = \max \left( \sum \Delta E_{P, color} \right) \quad (2.5)$$

理想情况下色差为 0。一般来说  $\Delta E < 2.3$  属于被认为肉眼不可见。优化目标选择平均色差还是最大色差有待商榷。

优化目标的选择也需要查一查显示器生产厂家是如何做的。

<sup>2</sup>CIE DE2000 很复杂，如果放宽要求可以采用 CIE94 甚至 CIE76 定义的色差。

### 3 亮度校正

经过亮度均匀性评价，我们得到每个像素的亮度表示。由于 HoughCircles 搜索到的圆并非按顺序排列的，需要将其对应到行和列。

然后将亮度数据重排成矩阵形式  $B$ 。

对于亮度矩阵，我们可以得到一个全像素平均亮度  $L_{overall}$ 。此时可以求出一个校正矩阵  $C$  使得

$$C \times B = L_{overall} \cdot I \quad (3.1)$$

将校正矩阵  $C$  应用于传输给控制盒的输出信号，等价于在硬件端进行校正。假设我们发给屏的信号亮度为  $B_{orig}$ ，采集到屏实际输出是上文提到的  $B$ ，那么屏的输入输出转化矩阵即为  $K$ ，它满足  $B_{orig} \times K = B$ 。

由于矩阵乘法满足结合律，下式也同样成立：

$$C \times B_{orig} \times K = C \times B = L_{overall} \cdot I \quad (3.2)$$

这样就可以通过校正输入信号来预览校正结果了。