

# Camera 图像处理原理及实例分析

作者：刘旭晖 [colorant@163.com](mailto:colorant@163.com) 转载请注明出处

BLOG：<http://blog.csdn.net/colorant/>

主页：<http://rgbboner.googlepages.com/>

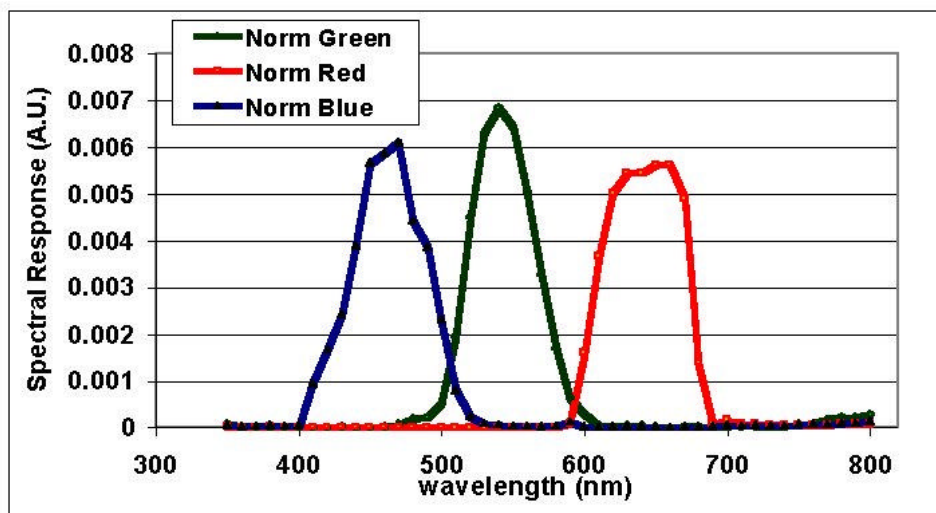
做为拍照手机的核心模块之一，camera sensor 效果的调整，涉及到众多的参数，如果对基本的光学原理及 sensor 软/硬件对图像处理的原理能有深入的理解和把握的话，对我们的工作将会起到事半功倍的效果。否则，缺乏了理论的指导，只能是凭感觉和经验去碰，往往无法准确的把握问题的关键，不能掌握 sensor 调试的核心技术，无法根本的解决问题。

## 1.1 色彩感应及校正

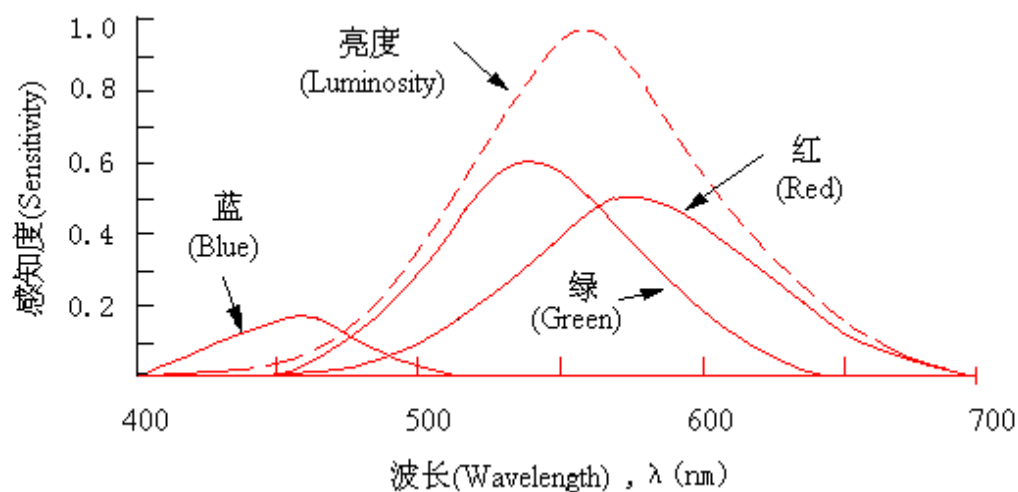
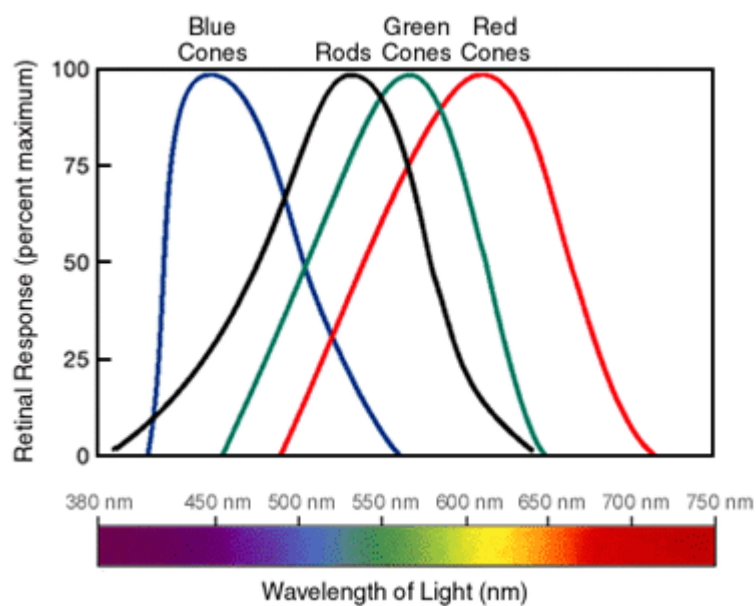
### 1.1.1 原理

人眼对色彩的识别，是基于人眼对光谱存在三种不同的感应单元，不同的感应单元对不同波段的光有不同的响应曲线的原理，通过大脑的合成得到色彩的感知。一般来说，我们可以通俗的用 RGB 三基色的概念来理解颜色的分解和合成。

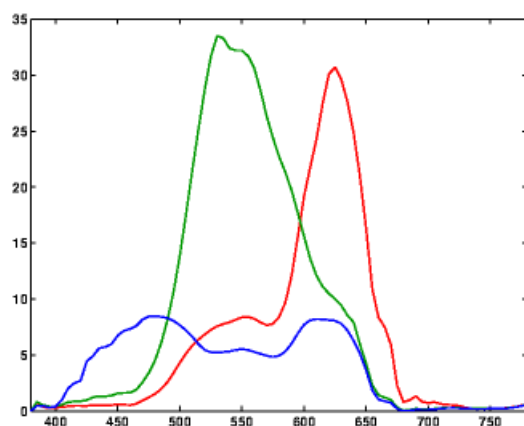
理论上，如果人眼和 sensor 对光谱的色光的响应，在光谱上的体现如下的话，基本上对三色光的响应，相互之间不会发生影响，没有所谓的交叉效应。



但是，实际情况并没有如此理想，下图表示了人眼的三色感应系统对光谱的响应情况。可见 RGB 的响应并不是完全独立的。



下图则表示了 Kodak 某相机光谱的响应。可见其与人眼的响应曲线有较大的区别。



### 1.1.2 对 sensor 的色彩感应的校正

既然我们已经看到 sensor 对光谱的响应，在 RGB 各分量上与人眼对光谱的响应通常是有偏差的，当然就需要对其进行校正。不光是在交叉效应上，同样对色彩各分量的响应强度

也需要校正。通常的做法是通过一个色彩校正矩阵对颜色进行一次校正。

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \begin{bmatrix} CMC11 & CMC12 & CMC13 \\ CMC21 & CMC22 & CMC23 \\ CMC31 & CMC32 & CMC33 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

该色彩校正的运算通常由 ISP 完成，软件通过修改相关寄存器得到正确的校正结果。值得注意的一点是，由于 RGB -> YUV 的转换也是通过一个 3\*3 的变换矩阵来实现的，所以有时候这两个矩阵在 ISP 处理的过程中会合并在一起，通过一次矩阵运算操作完成色彩的校正和颜色空间的转换。

## 1.2 颜色空间

### 1.2.1 分类

实际上颜色的描述是非常复杂的，比如 RGB 三基色加光系统就不能涵盖所有可能的颜色，出于各种色彩表达，以及色彩变换和软硬件应用的需求，存在各种各样的颜色模型及色彩空间的表达方式。这些颜色模型，根据不同的划分标准，可以按不同的原则划分为不同的类别。

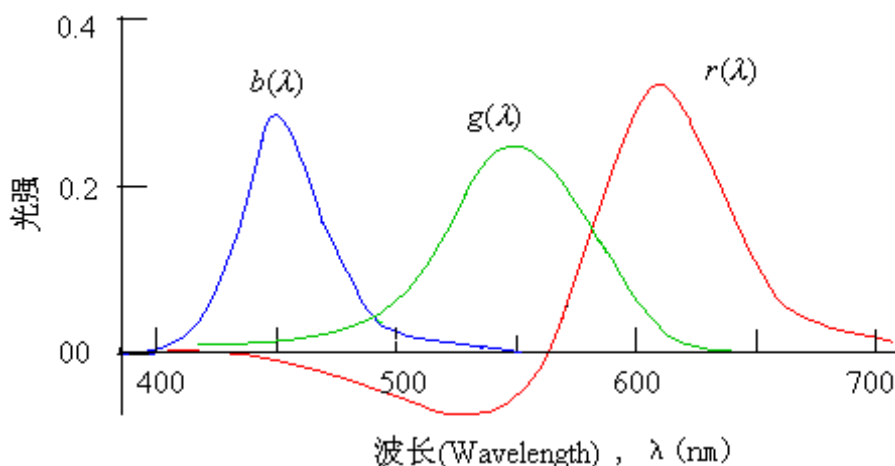


图1 匹配任意可见光所需的三原色光比例曲线

对于 sensor 来说，我们经常接触到的色彩空间的概念，主要是 RGB, YUV 这两种（实际上这两种体系包含了许多种不同的颜色表达方式和模型，如 sRGB, Adobe RGB, YUV422, YUV420 ...），RGB 如前所述就是按三基色加光系统的原理来描述颜色，而 YUV 则是按照亮度，色差的原理来描述颜色。

#### 1.2.1.1 RGB <-> YUV 的转换

不比其它颜色空间的转换有一个标准的转换公式，因为 YUV 在很大程度上是与硬件相关的，所以 RGB 与 YUV 的转换公式通常会多个版本，略有不同。

常见的公式如下

PAL 制：

$$Y = 0.299R + 0.587G + 0.114B$$

$$U = 0.493(B - Y) = -0.15R - 0.29G + 0.44B$$

$$V = 0.877(R - Y) = 0.62R - 0.52G - 0.10B$$

此外，还有，NTSC 制：

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

但是这样获得的 YUV 值存在着负值以及取值范围上下限之差不为 255 等等问题，不利于计算机处理，所以根据不同的理解和需求，通常在软件处理中会用到各种不同的变形的公式

$$\begin{bmatrix} Y \\ Cr \\ Cb \end{bmatrix} = \begin{bmatrix} 0.299 & 0.578 & 0.114 \\ 0.500 & -0.4187 & -0.0813 \\ -0.1687 & -0.3313 & 0.500 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix}$$

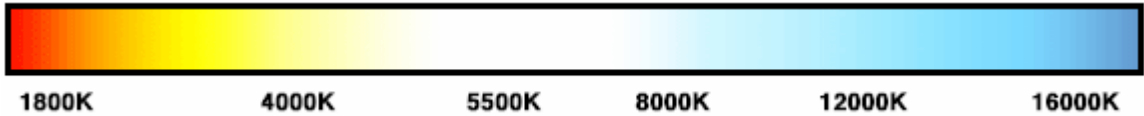
体现在 Sensor 上，我们也会发现有些 Sensor 可以设置 YUV 的输出取值范围。原因就在于此。

从公式中，我们关键要理解的一点是，UV 信号实际上就是蓝色差信号和红色差信号，进而言之，实际上一定程度上间接的代表了蓝色和红色的强度，理解这一点对于我们理解各种颜色变换处理的过程会有很大的帮助。

### 1.3 白平衡

#### 1.3.1 色温

色温的定义：将黑体从绝对零度开始加温，温度每升高一度称为 1 开氏度(用字母 K 来表示)，当温度升高到一定程度时候，黑体便辐射出可见光，其光谱成份以及给人的感觉也会随着温度的不断升高发生相应的变化。于是，就把黑体辐射一定色光的温度定为发射相同色光光源的色温。



常见光源色温：

光源	色温 (K)
钨丝灯 (白炽灯)	2500-3200k
碳棒灯	4000-5500k
荧光灯 (日光灯, 节能灯)	4500-6500k
氙灯	5600 k
炭精灯	5500 ~ 6500k
日光平均	5400k
有云天气下的日光	6500-7000k
阴天日光	12000-18000k

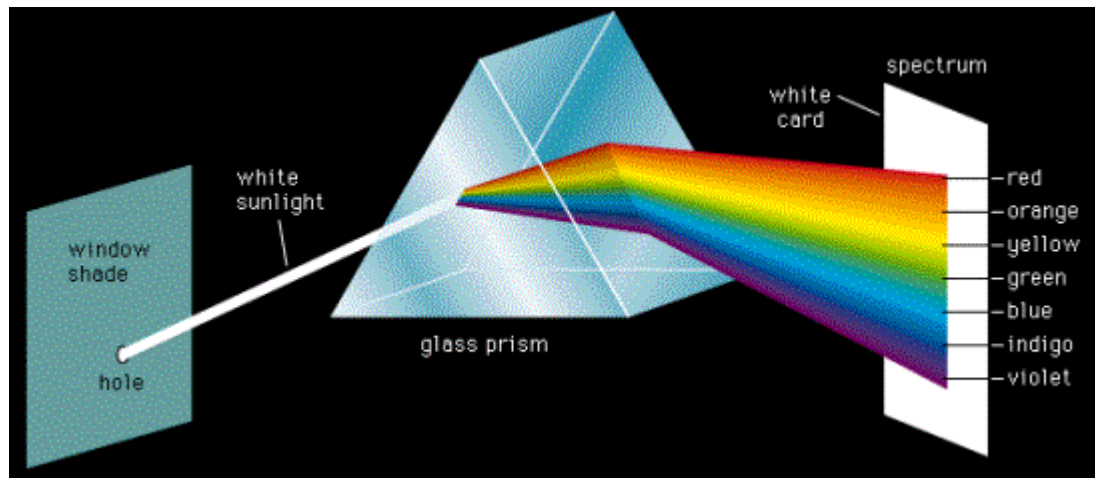
随着色温的升高，光源的颜色由暖色向冷色过渡，光源中的能量分布也由红光端向蓝光端偏移。

值得注意的是，实际光源的光谱分布各不相同，而色温只是代表了能量的偏重程度，并不反映具体的光谱分布，所以即使相同色温的光源，也可能引起不同的色彩反应。

人眼及大脑对色温有一定的生理和心理的自适应性，所以看到的颜色受色温偏移的影响较小，而 camera 的 sensor 没有这种能力，所以拍出来的照片不经过白平衡处理的话，和人

眼看到的颜色会有较大的偏差（虽然人眼看到的和白光下真实的色彩也有偏差）。

太阳光色温随天气和时间变化的原因，与不同频率光的折射率有关：



波长长的光线，折射率小，透射能力强，波长短的光线，折射率大，容易被散射，折射率低，这也就是为什么交通灯用红色，防雾灯通常是黄色，天空为什么是蓝色的等等现象的原因。

知道了这一点，太阳光色温变化的规律和原因也就可以理解和分析了，留给大家自己思考。

### 1.3.2 色温变化时的色彩校正

所以从理论上可以看出，随着色温的升高，要对色温进行校正，否则，物体在这样的光线条件下所表现出来的颜色就会偏离其正常的颜色，因此需要降低 sensor 对红色的增益，增加 sensor 对蓝光的增益。同时在调整参数时一定要考虑到整体亮度的要保持大致的不变，即以 YUV 来衡量时，Y 值要基本保持不变，理论上认为可以参考 RGB→YUV 变换公式中，RGB 三分量对 Y 值的贡献，从而确定 RGAIN 和 BGAIN 的变化的比例关系。但实际情况比这还要复杂一些，要考虑到不同 sensor 对 R,B 的感光的交叉影响和非线性，所以最佳值可能和理论值会有一些偏差。

### 1.3.3 自动白平衡原理

#### 1.3.3.1 原理

自动白平衡是基于假设场景的色彩的平均值落在一个特定的范围内，如果测量得到结果偏离该范围，则调整对应参数，校正直到其均值落入指定范围。该处理过程可能基于 YUV 空间，也可能基于 RGB 空间来进行。对于 Sensor 来说，通常的处理方式是通过校正 R/B 增益，使得 UV 值落在一个指定的范围内。从而实现自动白平衡。

#### 1.3.3.2 特殊情况的处理

在自动白平衡中，容易遇到的问题是，如果拍摄的场景，排除光线色温的影响，其本身颜色就是偏离平均颜色值的，比如大面积的偏向某种颜色的图案如：草地，红旗，蓝天等等，这时候，强制白平衡将其平均颜色调整到灰色附近，图像颜色就会严重失真。

因此，通常的做法是：在处理自动白平衡时，除了做为目标结果的预期颜色范围外，另外再设置一对源图像的颜色范围阈值，如果未经处理的图像其颜色均值超出了该阈值的话，根本就不对其做自动白平衡处理。由此保证了上述特殊情况的正确处理。

可见，这两对阈值的确定对于自动白平衡的效果起着关键性的作用。

### 1.3.4 某代码的例子

英文代码	中文界面	色温	色温	RGAIN, GGAIN, BGAIN
cloud	阴天	7500k	0x1D4C,	0x00CD, 0x0085, 0x0080
daylight	日光	6500k	0x1964,	0x00A3, 0x0080, 0x0088
INCANDESCENCE	白热光	5000k	0x1388,	0x00A5, 0x0080, 0x0088
FLUORESCENT	日光灯	4400k	0x1130,	0x0098, 0x0080, 0x00A8
TUNGSTEN	钨丝灯	2800k	0x0AF0,	0x0080, 0x0081, 0x00A4

可以看到随着色温的升高，其变化规律基本符合上节中的理论分析。不过这里多数参数与理论值都有一些偏差，其中日光灯的色温参数设置与理论值有较大的偏差，实际效果也证明该日光灯的参数设置使得在家用日光灯环境下拍摄得到的照片颜色偏蓝。修改其参数后实拍效果明显改善。（再查一些资料可以看到通常会有两种荧光灯色温 4000 和 5000K，目前我所接触到的应该是 5000K 居多）

### 1.3.5 调试和验证

具体参数的调整，应该在灯箱环境下，使用各种已知色温的标准光源对标准色卡拍摄，在 Pc 机上由取色工具测量得到其与标准色板的 RGB 分量上的色彩偏差，相应的调整各分量增益的比例关系。为了更精确的得到结果，曝光量增益的设置在此之前应该相对准确的校正过。

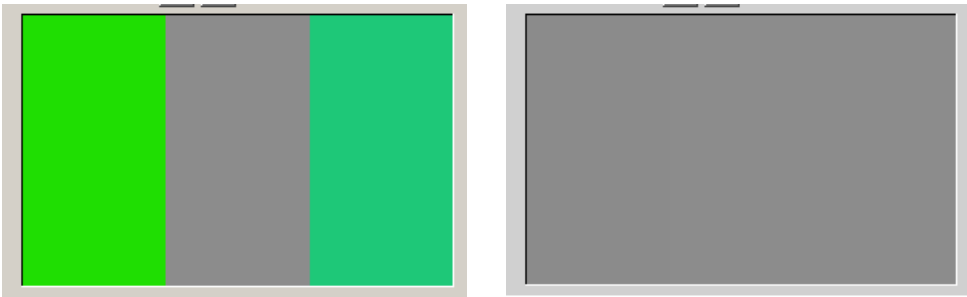
## 1.4 颜色相关特效处理

### 1.4.1 grayscale（灰阶）

灰阶图的效果就是将彩色图片转换为黑白图片。

#### 1.4.1.1 理论

理论上，在 YUV 空间，将 UV 分量丢弃，只保留 Y 分量，这样就可以得到黑白图像，这也是彩色电视信号能兼容黑白电视机的原理。如下图理论上 Y 值一样的颜色（右边是用 acdsee 转成灰度图的效果），在 grayscale 模式下看应该是一样的颜色。



算法上的操作，理论上应该把 UV 值改成灰色对应数值就可以了。不过根据软件算法和硬件结构的不同，具体代码也会有不同。

### 1.4.2 sepia / sepiagreen / sepiablue

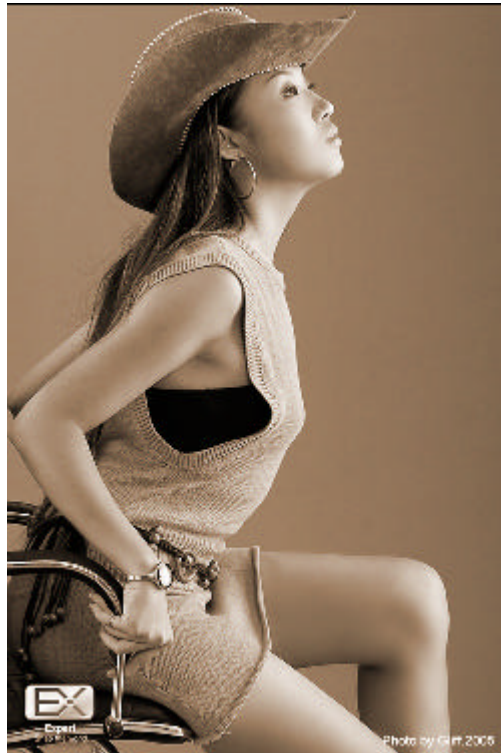
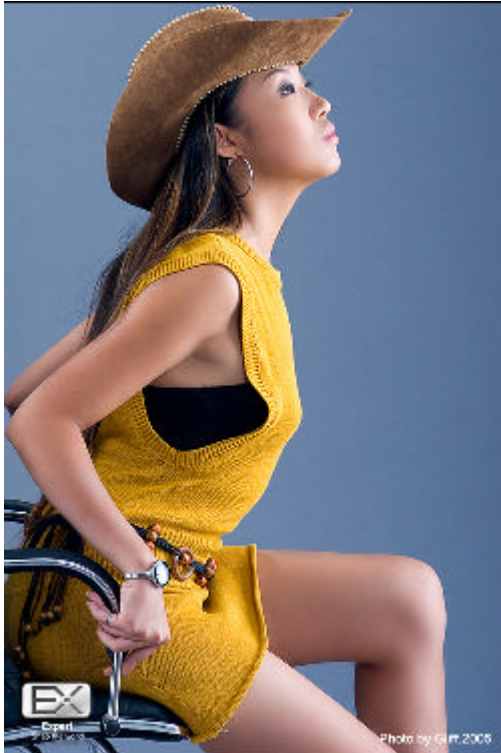
某平台手机上所谓的复古（绿，蓝）就是在灰阶的基础上，对 UV 值做了一个 offset，



将灰度图转换成某种颜色的梯度图。理论上为了获得蓝色效果，应该增加蓝色差信号，减小红色差信号。即增大  $U$ ，减小  $V$ 。

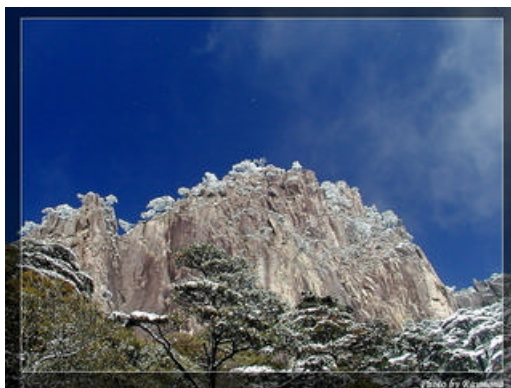
以 sepiablue 效果为例，这里的字节的 MSB 表示符号位：所以 88 为 88，158 为 -30。

```
SET_HUE_U_GAIN(0);  
SET_HUE_V_GAIN(0);  
SET_HUE_U_OFFSET(88);  
SET_HUE_V_OFFSET(158);
```



### 1.4.3 negative

所谓负片效果，就是将图像的颜色反转，看起来就像是在看胶片底片时的效果。这从理论上也很容易理解 and 处理，就是在 RGB 空间，取其补色，具体的操作就是用 255 分别减去 RGB 得到新的 RGB 值。通常会在 ISP 中实现该功能。



理解了原理，要做出其它颜色变换方面的效果就很容易了。  
基本上，在颜色校正和处理方面，需要考虑的相关参数大致包括：

自动 WB 上下限，自动白平衡时的目标范围，RGB gain, UV gain, UV offset, color correction.有些还会有 saturation 和 hue 相关的设置。

从 sensor 或 ISP 硬件处理的流程上说，通常方向是先做 RGB gain，再做 color correction，最后做 YUV 空间的处理。所以调整效果的时候，为了减少参数之间的相互影响，基本上也可以按这个顺序来调整参数。

## 1.5 亮度感应及曝光

### 1.5.1 感光宽容度

从最明亮到最黑暗，假设人眼能够看到一定的范围，那么胶片（或 CCD 等电子感光器件）所能表现的远比人眼看到的范围小的多，而这个有限的范围就是感光宽容度。

人眼的感光宽容度比胶片要高很多，而胶片的感光宽容度要比数码相机的 ccd 高出很多！了解这个概念之后，我们就不难了解，为什么在逆光的条件下，人眼能看清背光的建筑物以及耀眼的天空云彩。而一旦拍摄出来，要么就是云彩颜色绚烂而建筑物变成了黑糊糊的剪影，要么就是建筑物色彩细节清楚而原本美丽的云彩却成了白色的一片

再看人眼的结构，有瞳孔可以控制通光量，有杆状感光细胞和椎状感光细胞以适应不同的光强，可见即使人眼有着很高的感光宽容度，依然有亮度调节系统，以适应光强变化。

那么对于 camera sensor 来说，正确的曝光就更为重要了！

### 1.5.2 自动曝光和 18%灰

对于 sensor 来说，又是如何来判断曝光是否正确呢？很标准的做法就是在 YUV 空间计算当前图像的 Y 值的均值。调节各种曝光参数设定（自动或手动），使得该均值落在一个目标值附近的时候，就认为得到了正确的曝光。

那么如何确定这个 Y 的均值，以及如何调整参数使得 sensor 能够将当前图像的亮度调整到这个范围呢？

这就涉及到一个概念 18%灰，一般认为室内室外的景物，在通常的情况下，其平均的反光系数大约为 18%，而色彩均值，如前所述，可以认为是一种中灰的色调。这样，可以通过对反光率为 18%的灰板拍摄，调整曝光参数，使其颜色接近为中等亮度的灰色（Y 值为 128）。然后，对于通常的景物，就能自动的得到正确的曝光了。





当然这种自动判断曝光参数的 AE 功能不是万能的，对于反光率偏离通常均值的场景，比如雪景，夜景等，用这种方法就无法得到正确的曝光量了。所以在 sensor 的软件处理模块中，通常还会提供曝光级别的设定功能，强制改变自动曝光的判断标准。比如改变预期的亮度均值等。

### 1.5.3 曝光级别设定

在多数数码相机和拍照手机上都可以看到曝光级别设定的功能，如前所述，这种设定实际上是在自动曝光的基础上给用户一定的曝光控制能力，强制改变 camera sensor 的曝光判断标准，获得用户想要的效果。

通常的做法就是改变 Y 值均值的预期值，使得 sensor 在自动曝光时以新的 Y 预期值为目标，自动调整 Exptime 和 AG。

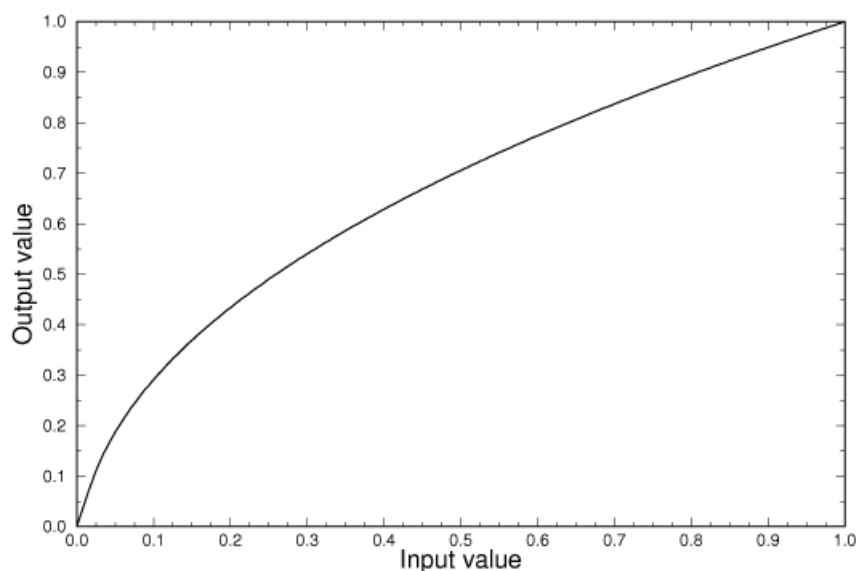
### 1.5.4 gamma 校正

曝光的均值正确了，不代表整体图像的亮度分布就和人眼所看到的保持一致了。

事实上，人眼对亮度的响应并不是一个线性的比例关系，而各种涉及到光电转换的设备的输入输出特性曲线一般也是非线性的，且表现为幂函数的形式： $y = x^n$ ，所以整个图像系统的传递函数是一个幂函数。 $g = g_1 \times g_2 \times \dots \times g_n$

对于 sensor 来说，其响应倒是接近为线性关系，所以为了在各种设备上正确输出符合人眼对亮度的响应的图像，就需要进行校正。

幂函数的指数的倒数就是通常所说的 gamma 值。



归一化的 gamma 曲线

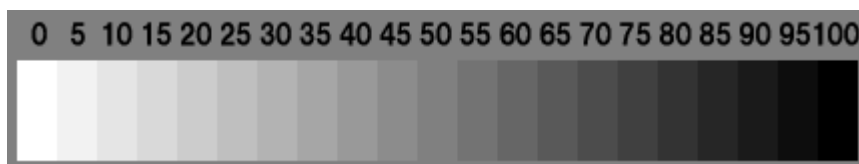
校正的函数可以表示为  $y = 255 * (x/255)^{\frac{1}{r}}$ ，通常对于 Window 的输出显示系统，gamma 值为 2.2，而对于苹果的输出显示系统和打印系统来说，gamma 值为 1.8。

实际上，sensor 在做 gamma 校正的时候，通常也一并作了从 raw 格式的 10bit 的数据到 8bit 数据的转换，所以这时候的公式可以表示为  $y = 255 * (x/1023)^{\frac{1}{r}}$ 。

由于指数运算需要消耗大量的 CPU 时间，所以实际的做法，往往是将 gamma 曲线用比

如 12 段线段进行拟合。这样只需要保存 13 个点的数据，用线性变换或查表的方式进行 gamma 校正。我司多数平台的 gamma 校正就是采用了后者的方法。要调整 gamma 校正实际上也就是调整这 13 个点的数值。

原则上，在拍摄如下图灰阶的色板时，能分辨出的级数越多越好。



### 1.5.5 对比度

对比度的调整在一定程度上说，其实也就是对 gamma 曲线的调整，增大对比度就是提高 Gamma 值。对于图像处理来说，也有在硬件 gamma 校正后，单独由软件再进行一次类似的幂函数变换来调整对比度。

### 1.5.6 曝光参数的调整

曝光强度的调整，可以通过改变曝光时间，也可以通过改变亮度增益 AG 来实现。

曝光时间受到帧频的限制，比如摄像时要求 15 帧每秒的话，这时候曝光时间最长就不能超过  $1/15s$ ，可能还有别的条件限制，实际的曝光时间还要短，在光线弱的情况下，单独调整曝光时间就无法满足帧频的需要了。

这时候还可以调整增益 AG，来控制曝光的增益，降低曝光时间。但是，这样做的缺点是以牺牲图像质量为代价的，AG 的增强，伴随的必然是信噪比的降低，图像噪声的增强。

所以，以图像质量为优先考虑的时候，曝光参数的调节通常是优先考虑调节曝光时间，其次在考虑曝光增益。当然曝光时间也不能过长以免由于抖动造成图像的模糊，而在拍摄运动场景时，对曝光时间的要求就更高了。

## 1.6 抗噪处理

AG 的增大，不可避免的带来噪点的增多，此外，如果光线较暗，曝光时间过长，也会增加噪点的数目（从数码相机上看，主要是因为长时间曝光，感光元件温度升高，电流噪声造成感光元件噪点的增多），而感光元件本身的缺陷也是噪点甚至坏点的来源之一。因此，通常 sensor 集成或后端的 ISP 都带有降噪功能的相关设置。

### 1.6.1 启动时机

根据噪点形成的原因，主要是 AG 或 Exptime 超过一定值后需要启动降噪功能，因此通常需要确定这两个参数的阈值，过小和过大都不好。

从下面的降噪处理的办法将会看到，降噪势附带的带来图像质量的下降，所以过早启动降噪功能，在不必要的情况下做降噪处理不但增加处理器或 ISP 的负担，还有可能适得其反。而过迟启动降噪功能，则在原本需要它的时候，起不到相应的作用。

### 1.6.2 判定原则和处理方式

那么如何判定一个点是否是噪点呢？我们从人是如何识别噪点的开始讨论，对于人眼来说，判定一个点是噪点，无外乎就是这一点的亮度或颜色与边上大部分的点差异过大。从噪点产生的机制来说，颜色的异常应该是总是伴随着亮度的异常，而且对亮度异常的处理工作量比颜色异常要小，所以通常 sensor ISP 的判定原则是一个点的亮度与周围点的亮度的差值

大于一个阈值的时候，就认为该点是一个噪点。

处理的方式，通常是对周围的点取均值来替代原先的值，这种做法并不增加信息量，类似于一个模糊算法。

对于高端的数码相机，拥有较强的图像处理芯片，在判定和处理方面是否有更复杂的算法，估计也是有可能的。比如亮度和颜色综合作为标准来判定噪点，采用运算量更大的插值算法做补偿，对于 sensor 固有的坏点，噪点，采用屏蔽的方式抛弃其数据（Nikon 就是这么做的，其它厂商应该也如此）等等。

### 1.6.3 效果

对于手机 sensor 来说，这种降噪处理的作用有多大，笔者个人认为应该很有限，毕竟相对数码相机，手机 sensor 的镜头太小，通光量小，所以其基准 AG 势必就比相机的增益要大（比如相当于普通家用数码相机 ISO800 的水平），这样才能获得同样的亮度，所以电流噪声带来的影响也就要大得多。这样一来，即使最佳情况，噪点也会很多，数据本身的波动就很大，这也就造成我们在手机照片上势必会看到的密密麻麻的花点，如果全部做平均，降低了噪点的同时，图像也会变得模糊，所以手机噪点的判断阈值会设得比较高，以免涉及面过大，模糊了整体图像。这样一来一是数据本身就差，二是降噪的标准也降低了，造成总体效果不佳。

## 1.7 数码变焦

数码变焦可以有两种形式：

其一，是通过插值算法，对图像进行插值运算，将图像的尺寸扩大到所需的规格，这种算法就其效果而言，并不理想，尤其是当使用在手机上的时候，手机上的摄像头本身得到的数据就有较大的噪声，再插值的话，得到的图像几乎没法使用。实际上，即使是数码相机的数码变焦功能也没有太大的实用价值。如果插值算法没有硬件支持，则需要在应用层实现。我司某平台的数码变焦用的就是该种办法。

其二，其实是一种伪数码变焦的形式，当摄像头不处在最大分辨率格式的情况下，比如 130 万像素的 sensor 使用 640\*480 的规格拍照时，仍旧设置 sensor 工作在 1280\*960 的分辨率下，而后通过采集中央部分的图像来获取 640\*480 的照片，使得在手机上看来所拍物体尺寸被放大了一倍。也有很多手机采用的是这种数码变焦方式，这种办法几乎不需要额外的算法支持，对图像质量也没有影响，缺点是只有小尺寸情况下可以采用。此外在 DV 方式下也可以实现所谓的数码变焦放大拍摄功能。（这应该是一个卖点，对 Dv 来说，这种数码变焦还是有实际意义的）

要采用这种变焦模式，驱动需要支持 windowing 功能，获取所需部分的 sensor 图像数据。

## 1.8 频闪抑制功能

### 1.8.1 何谓频闪

日常使用的普通光源如白炽灯、日光灯、石英灯等都是直接用 220/50Hz 交流电工作，每秒钟内正负半周各变化 50 次，因而导致灯光在 1 秒钟内发生 100（ $50 \times 2$ ）次的闪烁，再加上市电电压的不稳定，灯光忽明忽暗，这样就产生了所谓的“频闪”。

下表显示了几种光源的光强波动情况：

光源类型	工作频率	波动深度
白炽灯	50Hz	5~15%
钠灯或金卤灯	50Hz	80~130%
荧光汞灯（水银灯）	50Hz	65%
普通荧光灯（铁芯）	50Hz	55%
电子荧光灯（含 CFL）	20~35KHz	30~20%
	40~50KHz	15~10%
	80KHz	3%
三相交流荧光灯	50Hz	3~5%

因为人眼对光强变化有一定的迟滞和适应性，所以通常看不出光源的亮度变化。但是依然还是会增加眼睛的疲劳程度。所以市场上才会有所谓的无频闪灯销售。

## 1.8.2 对频闪的抑制

对于 camera sensor 来说，没有人眼的迟滞和适应过程，所以对光源亮度的变化是比较敏感的。如果不加抑制，在预览和 DV 模式下，可能会有明显的图像的明亮变化闪烁的现象发生。

如何解决呢？考虑到频闪的周期性，在一个周期内，光源亮度的累积值，应该是大体一致的，所以，如果控制曝光的时间是频闪周期的整倍数，那么每一帧图像的亮度就大体是一致的了，这样就可以有效地抑制频闪对图像亮度的影响。

所以，在自动曝光的模式下，sensor 会根据频闪的频率，调整曝光时间为其周期的整倍数。因为各地的交流电的频率不同，所以有 50Hz/60Hz 之分。

在具体设置相关 Sensor 寄存器的时候，要根据电流频率和 sensor 的时钟频率，分辨率等，计算出频闪周期对应的时钟周期数等。