



自动白平衡算法技术文档

1. 自动白平衡概述

白平衡是图像处理中的关键技术，通过调整图像的颜色通道来消除由不同光源造成的色差，使图像中的白色物体呈现真实的白色。

2. 算法介绍

1. 简单均值白平衡 (white_balance_1)

原理：基于灰度世界假设，认为图像的平均颜色应该是灰色的。通过计算各通道的平均值，然后调整各通道增益，使得三个通道的平均值相等。

实现步骤：

1. 分离图像的B、G、R通道
2. 计算每个通道的平均值
3. 计算三个通道平均值的平均值 (K值)
4. 计算各通道的增益：K值除以各自通道的平均值
5. 应用增益调整各通道
6. 合并通道返回结果

关键参数：

- 无特殊参数
- 时间复杂度： $O(n)$ ，适用于实时处理

2. 完美反射白平衡 (white_balance_2)

原理：完美全反射理论perfect Reflector假设图像上最亮点就是白点，并以此白点为参考对图像进行自动白平衡，最亮点定义为R+G+B的最大值。

实现步骤：

1. 分离图像通道

2. 计算每个像素的RGB和
3. 取RGB和值在前1%的像素 (默认ratio=0.01)
4. 计算这些高亮像素的通道平均值
5. 调整各通道使高亮区域变为中性色
6. 合并通道返回结果

关键参数：

- `ratio` : 高亮区域百分比阈值 (默认0.01)
- 时间复杂度: $O(n \log n)$, 适用于中等分辨率图像

3. 灰度世界假设 (white_balance_3)

原理: 灰度世界算法 (Gray World) 是以灰度世界假设为基础的,该假设认为对于一幅有着大量色彩变化的图像, R、 G、 B 三个分量的平均值趋于同一个灰度K。一般有两种方法来确定该灰度。

实现步骤:

1. 分离通道并转换为float32
2. 计算各通道平均值Raver,Gaver,Baver
3. 计算目标值 $K = (Raver+Gaver+Baver)/3$ (三通道平均值之和/3)
4. 计算各通道增益
 $Kr=K/Raver;$
 $Kg=K/Gaver;$
 $Kb=K/Baver;$
5. 根据Von Kries 对角模型,对于图像中的每个像素R、G、B, 计算其结果值。 (应用增益并限制像素值范围)
 $Rnew = R * Kr;$
 $Gnew = G * Kg;$
 $Bnew = B * Kb;$

对于上式, 计算中可能会存在溢出 (>255 , 不会出现小于0的)现象, 处理方式有两种。

a、 直接将像素设置为255, 这可能会造成图像整体偏白。

b、 计算所有Rnew、Gnew、Bnew的最大值, 然后利用该最大值将计算后数据重新线性映射到 $[0, 255]$ 。

6. 合并通道返回结果

特点：

- 避免整数截断误差
- 处理速度较快

4. 基于图像分析的偏色检测及颜色校正方法 (white_balance_4)

原理：通过建立通道间的二次回归关系来校正颜色偏差。

实现步骤：

1. 分离通道并转换为浮点数
2. 计算各通道的平方值和统计量
3. 构建线性方程组求解校正系数
4. 应用二次回归校正公式
5. 合并通道返回结果

特点：

- 处理复杂色偏效果好
- 数学计算较复杂

5. 动态阈值算法 (white_balance_5)

原理：在YCrCb颜色空间中分析，选择特定颜色范围内的亮点作为参考白点。

实现步骤：

白点检测

1. 转换到YCrCb颜色空间
2. 计算Cb和Cr通道的平均值Mb/Mr，统计特征：绝对差的累积值Db/Dr

$$D_b = \sum_{i,j} (|C_b(i,j) - M_b|) / N$$

$$D_r = \sum_{i,j} (|C_r(i,j) - M_r|) / N$$

上式中N为每个区域的像素数。统计对于除了符合第四条的其他区域的Mb/Mr/Db/Dr的平均值作为整幅图像的Mb/Mr/Db/Dr值

3. 创建颜色置信区域掩码 (初步确定哪些点是属于白色参考点)

$$|C_b(i,j) - (M_b + D_b \times sign(M_b))| < 1.5 \times D_b$$

$$|C_r(i,j) - (1.5 \times M_r + D_r \times sign(M_r))| < 1.5 \times D_r$$

4. 在掩码区域内选择亮度前10%的像素 (最终确定的白色参考点)

白点调整

5. 计算这些像素的平均值Raver,Gaver,Baver
6. 应用增益并限制最大增益值

$$R_{gain} = Y_{max} / R_{avew}$$

$$G_{gain} = Y_{max} / G_{avew}$$

$$B_{gain} = Y_{max} / B_{avew}$$

式中，Ymax就是YCbCr颜色空间中Y分量的最大值。

7. 合并通道返回结果

$$R' = R \times R_{gain}$$

$$G' = G \times G_{gain}$$

$$B' = B \times B_{gain}$$

其中R/G/B为在原始的颜色空间中的值，注意这里要进行溢出检测的

关键参数：

- `max_gain`：最大允许增益（默认2.0，防止过度曝光）
- `radio`：颜色偏差阈值（默认0.5）
- 最先进的自适应算法，适用于多种场景

3. 测试结果（用comparison.png综合展示）

输出说明

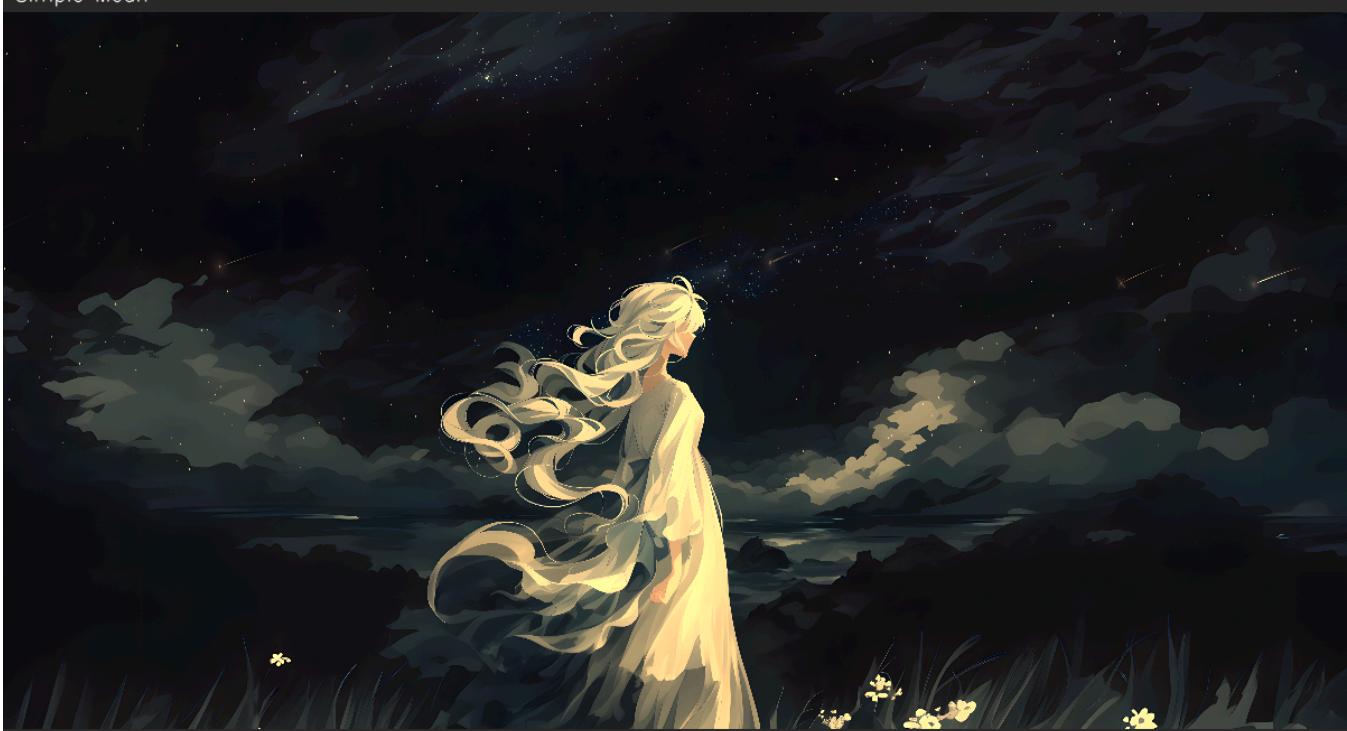
执行程序后会生成以下文件：

1. `original.png`：原始输入图像
2. `simple_mean.png`：简单均值白平衡结果
3. `perfect_reflection.png`：完美反射白平衡结果
4. `gray_world.png`：灰度世界假设结果
5. `color_deviation.png`：偏色检测校正结果
6. `dynamic_threshold.png`：动态阈值算法结果
7. `comparison.png`：所有结果的垂直堆叠对比图（带标签）

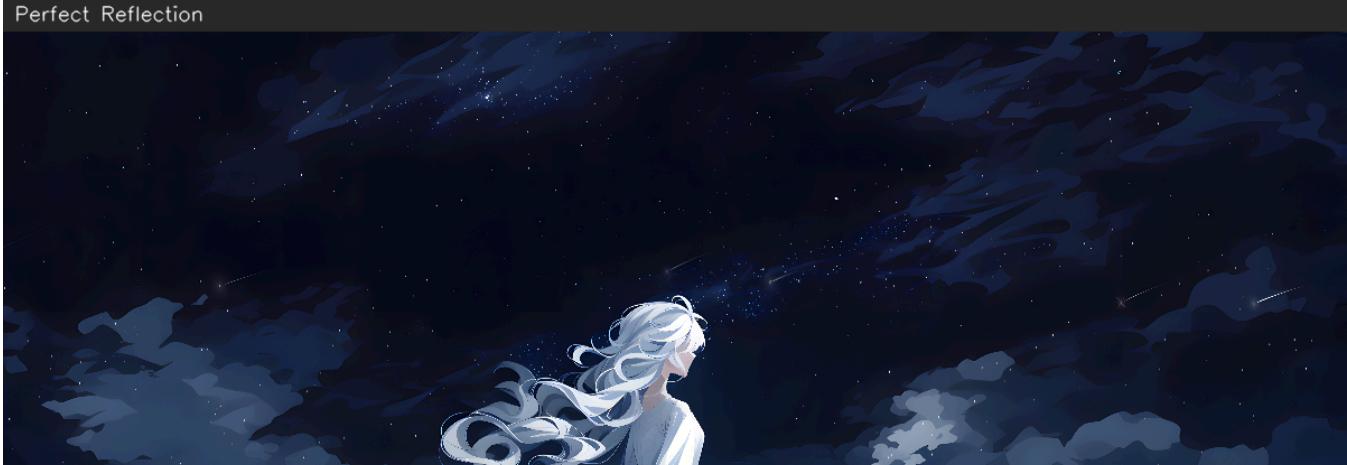
Original



Simple Mean



Perfect Reflection



4. 算法选择

1. 算法复杂度：

- 简单均值法: $O(n)$
- 完美反射法: $O(n \log n)$ (由于排序操作)
- 动态阈值法: $O(n)$ (优化后)

2. 实时性建议：

- 对于实时应用，推荐使用简单均值法或灰度世界法
- 对于质量要求高的应用，推荐动态阈值法

3. 参数调优：

- 夜景/低照度：降低 `max_gain` (1.5-1.8)
- 高对比度场景：调整 `ratio` (完美反射法)
- 混合光源：调整 `ratio` (动态阈值法)

注意事项

1. 所有算法都要求输入图像为BGR格式 (OpenCV默认格式)
2. 动态阈值算法在低照度场景下可能需要降低 `max_gain` 值
3. 完美反射法在高光区域不足时效果不佳
4. 偏色检测法在图像色彩单调时可能不适用

扩展建议

1. 可添加场景检测功能，自动选择最佳算法
2. 可集成到图像处理流水线中，作为ISP的一部分
3. 可优化为FPGA硬件实现，提高处理速度

本实现为各种场景提供了全面的白平衡解决方案，用户可根据具体需求选择合适的算法和参数。