落羽の殇

本人专注于图像算法以及软件安全的实现与优化。

联系Email:
gaozhihan@vip.qq.com
To get where you want to go, you have to know where you are.
昵称：落羽の殇
园龄：9个月
粉丝：32
关注：0
+加关注

< 2016年8月 >

日	一	二	三	四	五	六
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

搜索
找找看
谷歌搜索

常用链接
我的随笔
我的评论
我的参与
最新评论
我的标签
更多链接

我的标签
12306车票 抢票软件(1)
3A算法 白平衡(1)
bilateralFilter(1)
demos 冷暖自知(1)

相机中白平衡的算法模拟实现

相机主要技术点为3A算法。

而3A算法主要指的是自动对焦(AF)、自动曝光(AE)及自动白平衡(AWB)。
自动白平衡:根据光源条件调整图片颜色的保真程度。

网上时常有类似招聘如下的招聘信息:

Camera/ISP 算法工程师
摄像机3A算法软件工程师

这里随机摘录一些具体要求。

任职要求:
1、本科以上学历，天文，物理，机电、工业自动化，电子相关专业，硕士学历优先考虑;
2、本科毕业3年以上，硕士毕业1年以上的相关行业相关工作经验要求;
3、熟练掌握C/C++或者FPGA 开发语言,数据结构,MATLAB,信号和系统;
4、掌握数字色度学，数字图像处理，数字影像处理的基本知识;
5、熟悉摄像机成像原理;
6、掌握3A(AF,AE,AWB)算法之一;
7、对于自动化控制，数字信号采样，滤波，负反馈，PID算法有实际经验;
8、理解从镜头到SENSOR,电机，ISP，编码器，采集，显示通道一些列变化。

任职要求:
1. 精通camera的3A（AE,AWB,AF）算法原理和设计思路， 有3A算法的设计经验为佳
2. 具备丰富ISP（图象处理器） 开发经验，熟悉MTK，QUALCOMM, OV等便携式终端上应用的ISP开发环境。有上述环境下开发经验为佳。
3. 精通数字图像处理原理和基础知识。
4. 熟悉C/C++语言，有开发经验为佳
5. 有手机/便携式相机3A算法实现/应用经验
6. 精通CMOS sensor的工作原理

而这类职位一般都是高薪待遇。

然后问题来了，市面上3A算法相关资料都非常稀少，就连相关书籍都很少提及算法细节，而他们基本上都会要求精通3A算法至少之一。

而关于白平衡算法，比较不错的资料是这份:

基于灰度世界、完美反射、动态阈值等图像自动白平衡算法的原理、实现及效果

之前多次与博主laviewpbt探讨相关的知识，受益匪浅。

而据我所知，绝大多数的相机采用的基础算法便是灰度世界算法,然后在这算法的基础上再改进。

贴一下《基于灰度世界、完美反射、动态阈值等图像自动白平衡算法的原理、实现及效果》灰度世界法的大概内容。

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。

（1)直接给定为固定值, 取其各通道最大值的一半,即取为127或128;

高斯模糊 C++ 高斯算法(1)

高斯模糊 快速高斯模糊 模糊算法(1)

谷歌开源项目 Google Preview Image Extractor(PIEX) 无损图片格(1)

阶段性总结 一键修图(1)

卷积(1)

均值模糊算法 快速均值模糊 图像处理(1)

更多

随笔分类

图像处理(16)

网络安全(3)

随笔档案

2016年4月 (4)

2016年3月 (2)

2016年1月 (3)

2015年12月 (2)

2015年10月 (8)

最新评论

1. Re:半径无关单核单线程最快速高斯模糊实现(附完整C代码)

@无悔的青春不好意思，我没学过 计算机图形学。...

--落羽の殇

2. Re:半径无关单核单线程最快速高斯模糊实现(附完整C代码)

博主，有没有《计算机图形学》的课程设计报告，能给我发一份吗？邮箱：1324728812@qq.com。真心对这门课不感兴趣，学期快结束了，要交作业了.....

--无悔的青春

3. Re:这一路走来，冷暖自知 (附算法

（2)令 K = (Raver+Gaver+Baver)/3,其中Raver,Gaver,Baver分别表示红、 绿、 蓝三个通道的平均值。

算法的第二步是分别计算各通道的增益：

Kr=K/Raver;

Kg=K/Gaver;

Kb=K/Baver;

算法第三步为根据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：

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将计算后数据重新线性映射到[0,255]内。实践证明这种方式将会使图像整体偏暗，建议采用第一种方案。

---------- ----------

算法的大概思路就是评估一张图片RGB三个通道的中最能表达该通道富含信息的值，然后以该值为基准重新调整像素。

这样就会存在评估不够准确的问题，导致各通道像素信息差距过大，形成噪点以及偏色等现象。

因为如果采用取最大值的方案就会导致在特定情况明显不均衡，例如该通道大多数的值落在最小值周围，而却存在一个遥远处的最大值，那么就会导致像素信息差距过大，就很糟糕了。

所以在第二种思路上进行进一步改进比较稳妥，因为可用的信息比较多，不容易出问题。

第二种思路，最简单的另一种改进就是采用灰度法。

均值法: K = (Raver+Gaver+Baver)/3

我们知道常用的视频采集编码是YUV。

YUV相关见百度百科:YUV

其中的Y为:

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

故灰度法相应可对应为:

K=0.299*Raver + 0.587*Gaver+0.114*Baver

经过实测，这样的处理后效果还不错。

贴上对比图：

原图

demos)

@迷失的白马大一，一块处女地等着你去开发。加油。...

--落羽の殇

4. Re:这一路走来，冷暖自知 (附算法demos)

单这阅读量我就服了，况且我是大一学生，作为学生一年所读书籍都那么那么少，真是非常惭愧！感谢博主分享！

--迷失的白马

5. Re:这一路走来，冷暖自知 (附算法demos)

@韩子迟额，俺菜鸟一枚，见笑了。...

--落羽の殇

阅读排行榜

1. 这一路走来，冷暖自知 (附算法demos)(934)

2. 谷歌开源项目Google Preview Image Extractor(PIEX) (附上完整demo代码)(767)

3. 传统高斯模糊与优化算法(附完整C++代码)(584)

4. 相机中白平衡的算法模拟实现(555)

5. 半径无关单核单线程最快速高斯模糊实现(附完整C代码)(547)

评论排行榜

1. 这一路走来，冷暖自知 (附算法demos)(14)

2. 学习图像算法阶段性总结 (附一键修图Demo) 2016.04.19更新demo(6)

3. 快速均值模糊算法(3)

4. 半径无关单核单线程最快速高斯模糊实现(附完整C代码)(2)

5. 双边滤波算法的简易实现bilateralFilter(2)

均值法

灰度法

单从肉眼上去分辨两张图片,的确很难分出优劣。

不过我也只是大概点一下这个思路而已，有所积累的人，看到这，应该可以发散出更多的想法。

接下来我要说的是具体相机中的钨丝灯等手动白平衡是如何实现的。

简单的说就是色温调节。

那么基于灰度世界这个白平衡算法可以怎么实现这种调节呢？！

这里贴出简单实现的C代码:

```
switch (preset)
    {
    case AUTO:
        Raver = (SumR / numberOfPixels);
        Gaver = (SumG / numberOfPixels);
        Baver = (SumB / numberOfPixels);
        break;
    case CLOUDY:
        Raver = (SumR *1.953125 / numberOfPixels);
        Gaver = (SumG*1.0390625 / numberOfPixels);
        Baver = (SumB / numberOfPixels);
        break;
    case DAYLIGHT:
        Raver = (SumR *1.2734375 / numberOfPixels);
        Gaver = (SumG / numberOfPixels);
        Baver = (SumB*1.0625 / numberOfPixels);
        break;
```

```
    case INCANDESCENCE:
        Raver = (SumR *1.2890625 / numberOfPixels);
        Gaver = (SumG / numberOfPixels);
        Baver = (SumB*1.0625 / numberOfPixels);
        break;
    case FLUORESCENT:
        Raver = (SumR *1.1875 / numberOfPixels);
        Gaver = (SumG / numberOfPixels);
        Baver = (SumB*1.3125 / numberOfPixels);
        break;
    case TUNGSTEN:
        Raver = (SumR / numberOfPixels);
        Gaver = (SumG*1.0078125 / numberOfPixels);
        Baver = (SumB*1.28125 / numberOfPixels);
        break;
    default:
        break;
    }
```

```
enum WB_PRESET{
    //自动白平衡
    AUTO,
    //阴天 7500k
    CLOUDY,
    //日光 6500k
    DAYLIGHT,
    //白热光 5000k
    INCANDESCENCE,
    //日光灯 4400k
    FLUORESCENT,
    //钨丝灯 2800k
    TUNGSTEN,
};
```

阴天

日光

白热光

日光灯

钨丝灯

这里只是起到一个演示作用，具体的参数，可按实际需求酌情进行修改。

本文只是抛砖引玉一下，若有其他相关问题或者需求也可以邮件联系我探讨。

邮箱地址是:

gaozhihan@vip.qq.com

分类: 图像处理

标签: 3A算法 白平衡

落羽の殇
关注 - 0
粉丝 - 32
+加关注

« 上一篇：双边滤波算法的简易实现bilateralFilter
» 下一篇：传统高斯模糊与优化算法(附完整C++代码)

posted @ 2016-01-05 15:43 落羽の殇 阅读(554) 评论(0) 编辑 收藏