

码龄8年

暂无认证

88

7万+

1万+

28万+



原创

周排名

总排名

访问

等级

3372

73

219

59

305

积分

粉丝

获赞

评论

收藏







私信

关注

搜博文文章

Q

- 热门文章
- RGB图像转为灰度图

30882
- 字节数组byte[]和整型,浮点型数据的转换——Java代码

28334
- Microsoft Word 插件开发——Word外接应用程序开发

22342
- 文件打开的过程——调用fd=open()时操作系统所做的工作

11677
- Debug Assertion Failed! Expression: _pFirstBlock == pHead

11366

- 最新评论
- RGB图像转为灰度图

liuss1: 牛牛牛，真的学到了
- 字节数组byte[]和整型,浮点型数据的转换...

StarkBo: 能问一问为什么是这么写吗?
- 字节数组byte[]和整型,浮点型数据的转换...

cyy298: 楼主你好，如果使用ByteBuffer.putInt(i); 是否还需要转换?
- Socket通信——C++服务器端和Java客户端

船儿飞: 如果发送的结构体中有字符串类型的字段客户端该怎么处理
- RGB图像转为灰度图

Discipline※: 大佬好，这是MATLAB的吗?

您愿意向朋友推荐“博客详情页”吗？











强烈不推荐

不推荐

一般般

推荐

强烈推荐

- 最新文章
- [开发杂项][Ubuntu][常用命令杂项]
- [开发杂项][VS Code]remote-ssd retry failed
- [开发杂项][C/C++][g++ basics]

原创

xcy6666

于 2015-08-07 16:33:48 发布

30921

收藏 79

版权

分类专栏:

杂集

文章标签:

rgb



杂集

专栏收录该内容

0 订阅

30 篇文章

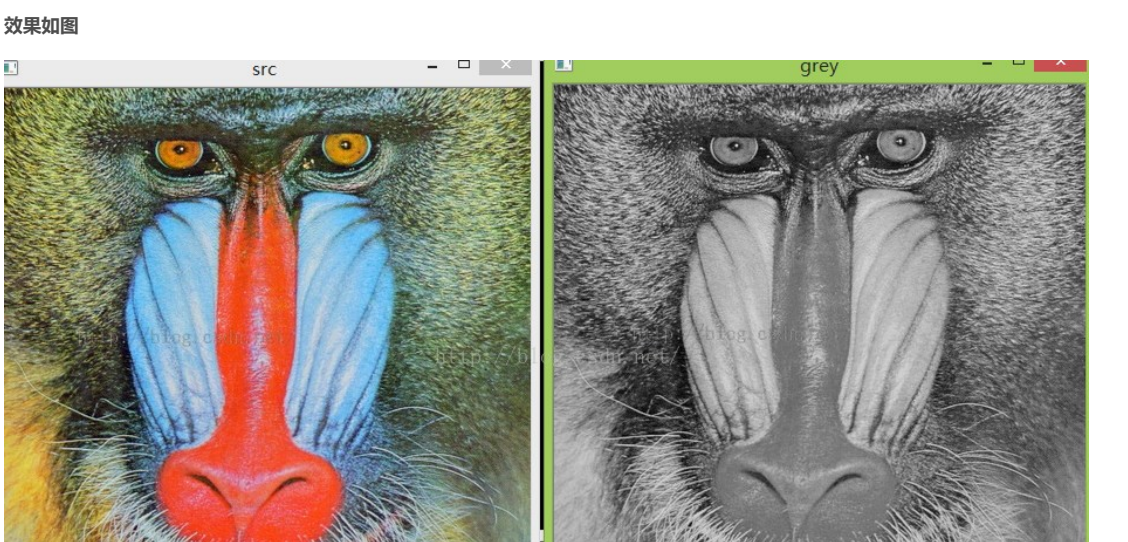
订阅专栏

最后结论：

$$\text{Grey} = (\text{R} \times 38 + \text{G} \times 75 + \text{B} \times 15) \gg 7$$

代码

```
1 #include <cv.h>
2 #include <highgui.h>
3
4 using namespace cv;
5 int main(){
6     Mat src= imread("C:\\Users\\Poplar\\Pictures\\ff.jpg");
7     Mat grey(src.rows, src.cols, CV_8UC1, Scalar(0));
8     for (inty = 0; y < src.rows; y++)
9     {
10         uchar*cp = src.ptr<uchar>(y);
11         uchar*gp = grey.ptr<uchar>(y);
12         for(int x = 0; x < src.cols; x++){
13             *gp= (15*cp[0] + 75*cp[1] + 38*cp[2]) >> 7;
14             cp+= 3;
15             gp++;
16         }
17     }
18     imshow("src",src);
19     imshow("grey",grey);
20     waitKey(0);
21     return0;
22 }
```



2021年 18篇	2020年 10篇
2019年 6篇	2018年 5篇
2017年 8篇	2016年 21篇
2015年 24篇	2014年 7篇



下面具体解释

RGB彩色图像中，一种彩色由R（红色）， G（绿色）， B（蓝色）三原色按比例混合而成。

图像的基本单元是一个像素，就像一个巨幅电子广告屏上远处看是衣服图像，走近你会看到一个一个的方格，这个方格的颜色是一种，从远处看，觉察不到这个方格的存在。

一个像素需要3块表示，分别代表R， G， B，如果8为表示一个颜色，就由0-255区分不同亮度的某种原色。

一张9像素的8位RGB图像，在计算机内存中的分布大概示意如下：

	B	G	R
0	0	0	255
	255	0	255
	255	0	0
	0	255	255
	255	255	0
	255	255	255
	0	255	0
	255	255	255
	0	0	0
26			

实际中数都是二进制形式的，并且未必按照R， G， B顺序，比如opencv是按照B,G,R顺序将三个色值保存在3个连续的字节里

灰度图像是用不同饱和度的黑色来表示每个图像点，比如用8位 0-255数字表示“灰色”程度，每个像素点只需要一个灰度值，8位即可，这样一个3X3的灰度图，只需要9个byte就能保存

RGB值和灰度的转换，实际上是人眼对于彩色的感觉到亮度感觉的转换，这是一个心理学问题，有一个公式：

$$\text{Grey} = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$$

根据这个公式，依次读取每个像素点的R， G， B值，进行计算灰度值（转换为整型数），将灰度值赋值给新图像的相应位置，所有像素点遍历一遍后完成转换。

一张**500X500**的图像转换为同样大小的灰度图需要进行**25万次**上述公式的计算。进行优化是很有必要的，这个简单的算法是O(n)复杂度的，应该是不能优化了（或者用并行进行优化，本文不涉及），但是Grey = 0.299*R + 0.587*G + 0.114*B有更加高效的等价形式。

在ALU中，位操作快于整数加法，整数加法快于整数乘法（快多少取决于有没有乘法电路，乘法电路的结构），整数运算又比浮点数运算快得多。

所以可以通过**将浮点数运算转化为整数运算，整数运算转换为位操作进行优化**

$$\text{Grey} = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$$

可以转化为

```
Grey = (299*R + 587*G + 114*B + 500) /1000;
```

整数运算会截断小数部分，加上500是为了四舍五入（找两个例子便可理解），减少精度损失。

这里的除法/ 即使是整数除法计算也是很耗时，转换为移位操作可以优化，那么**怎么转换为位操作**？左右移位对应于乘除2的幂，为了把除法转为右移操作，做如下处理：

```
Grey = 0.299*R + 0.587*G + 0.114*B

Grey = (299*R+ 587*G + 114*B) ÷ 1000

Grey = (1024*299*R+ 1024*587*G + 1024*114*B) ÷ (1024*1000)

Grey = (306176*R+601088*G + 116736*B) ÷ (1024*1000)

Grey = (306.176*R+601.088*G + 116.736*B) ÷ (1024)

Grey = (306*R+601*G + 116*B) ÷ (1024) //截断误差

Grey = (306*R+601*G + 116*B) >> 10;
```

误差最大是多少？

```
(0.176*255+0.088*255 + 0.736*255) ÷1024 = 255÷1024=0.249，可能会导致1个灰度值的波动
```

有一种计算方法可以降低误差

```
R 的系数  =1024*0.229= 306.176≈306

G的系数   =1024*0.587 + 0.176 =601.264 ≈601

B的系数   =1024*0.114 + 0.264 = 117
```

保留了小数部分的作用，可以得到一个误差较小的公式：

```
Grey = (306*R +601*G + 117*B) >> 10;
```

这样得来的是10位精度的。

同样的方法可以获得其他精度的，比如

```
Grey = (R*1 + G*2 + B*1) >> 2 ( Grey = (R + G<<1 + B) >> 2 )

Grey= (R*38 + G*75 + B*15) >> 7

Grey= (R*76 + G*150 + B*30) >> 8

Grey = (R*19595 + G*38469 + B*7472) >> 16
```

可以看出来，7位和8位精度是一样的，比较好用的是7位精度的公式。

实际编写代码时，还要考虑图像文件的读取问题，不同格式的RGB位图，结构不同，读取时也不同，本文不涉及图像读取问题，这里以openCV提供的图像读取方式，展示转灰度图的实际代码，见文章开头。

2-10位精度的公式

```
Grey = (R*1 + G*2 + B*1) >> 2
```

```
Grey= (R*2 + G*5 + B*1) >> 3
```

Grey= (R*4 + G*10 + B*2) >> 4

Grey = (R*9 + G*19 + B*4) >> 5

Grey = (R*19 + G*37 + B*8) >> 6

Grey= (R*38 + G*75 + B*15) >> 7

Grey= (R*76 + G*150 + B*30) >> 8

Grey = (R*153 + G*300 + B*59) >> 9

Grey = (R*306 + G*601 + B*117) >> 10

Grey = (R*612 + G*1202 + B*234) >> 11

Grey = (R*1224 + G*2405 + B*467) >> 12

Grey= (R*2449 + G*4809 + B*934) >> 13

Grey= (R*4898 + G*9618 + B*1868) >> 14

Grey = (R*9797 + G*19235 + B*3736) >> 15

Grey = (R*19595 + G*38469 + B*7472) >> 16

Grey = (R*39190 + G*76939 + B*14943) >> 17

Grey = (R*78381 + G*153878 + B*29885) >> 18

Grey =(R*156762 + G*307757 + B*59769) >> 19

Grey= (R*313524 + G*615514 + B*119538) >> 20

RGB图像转为灰度图的几种方法 01-20
RGB图像灰度化 RGB图像 我们来看看下面这张图，美不美 这么好看的图是咋个整出来的呢？哦！原来是这样：一幅图像由很多个像素点组成的，而每...

实现灰度图、二值图、rgb之间的转化 12-03
数字图像处理的基本操作：将rgb图像转为灰度、二值、索引等；实现图像的放大、缩小；明确灰度值的意义，对其进行操作和计算；等。

评论 6 您还未登录，请先 登录 后发表或查看评论

RGB图转灰度图_1lang_rgb转灰度图 3-2
我们知道,一般的彩色图像指的是在RGB颜色空间下的图像,这样的图像三个通道分别是R(red)G(green)B(blue)。而灰度图指的是单通道的,将三通道的图像...

RGB转灰度图_dannyisi的专栏_rgb变灰度图 3-15
RGB转灰度图 VC++写个图像处理的程序,挑了个简单的,RGB转灰度图,上网查了些资料,发现转化的方法还各种各样,于是写篇日志以祭之。 *** 方法一: 对...

RGB转灰度图 54
bool Rgb2Gray(const char * szFilePah) { if(NULL == szFilePath || '\0' == *szFilePath) { return false; } std::string strTemp = szFileP...

RGB彩色图像转灰度图算法——java实现 席木木的博客 1118
引文 关于如何通过编程实现将RGB图像转灰度图像，通过最近在网上查找了一些相关文章的积累，也逐渐形成了一些自己的理解，在此，我想与大家分享...

RGB转为灰度图然后二值化处理_zhuimengshaonian66的博... 3-8
RGB转为灰度图然后二值化处理 这里可以说很坑了,搞了很久才发现原来有些事情不能想当然 比如说空洞区域我们以为它的像素值为0.想提取出来可以让...

RGB图像转为灰度图的几种方法_努力学习的萝卜的博客-CS... 2-21
RGB图像转为灰度图的几种方法 RGB图像灰度化 RGB图像 我们来看看下面这张图,美不美 这么好看的图是咋个整出来的呢?哦!原来是这样:一幅图像由...

RGB转灰度的几种算法 weixin_30364325的博客 1043

这里我用的摄像头是OV2640，这款摄像头的像素是200万。通过对摄像头拍摄  的进行  的转化，先把  的格式转化为RGB565,然后在进行灰度的...			
BMP文件格式	lanstar200的专栏	3098	BMP文件格式分析 简介BMP(Bitmap-File)  形文件是Windows采用的  文件格式，在Windows环境下运行的所有  象处理软件都支持BMP  象文件格式...
色彩转换系列之RGB图转灰度图原理及实现_小武的博客-CS...		2-24	彩色  转灰度图在  像处理中应用非常非常广泛,而且很多算法只对灰度图有效,所以彩色  转灰度是十分重要和关键的。 RGB(红绿蓝)是依据人眼识别的...
RGB转换为灰度图_csg3140100993的博客		2-4	首先明白一点,灰度图并不是字面上的黑白  的意思,主要看是不是单通道,所以做实验的时候很多  片转换成灰度图之后看着并不像是黑白  。转换代码...
RGB图转为灰度图的方法 最新发布	m0_49089298的博客	915	我们日常的环境通常获得的是彩色  ，很多时候我们常常需要将彩色  转换成灰度  。也就是3个通道（RGB）转换成1个通道。（1）平均法 最简...
彩色  怎样转灰度  热门推荐	saltriver的专栏	7万+	我们日常的环境通常获得的是彩色  ，很多时候我们常常需要将彩色  转换成灰度  。也就是3个通道（RGB）转换成1个通道。（1）平均法 最简...
基于python的  格式转换（将RGB  转换为灰度  ）	little_yan的博客	6万+	1.将RGB  转换为灰度  ： <pre>from PIL import Image I = Image.open('F:\\pycharm\\picture_format\\data\\lena.jpg') I.show() L = I.convert('L') L.show()</pre> ...
将一幅  转换为灰度  灰度  是指用灰度表示的  ，灰度是在白色和黑色之间分的若干个等级，其中最常用的是256级，也就是256级灰度  。灰度  在医学、航天等领域有着...	weixin_30410999的博客	216	
 转灰度  使用C#编写，结合OpenCVSharp，将选择的  转化为灰度  ，保存灰度  ，完整代码，测试可运行，需要先配置OpenCVSharp		05-16	
RGB图转换为灰度  轻松将真彩色  转换为灰度  ，为你  处理打通第一道口		03-20	
详细的RGB转化成灰度计算过程-RGB转化成灰度 详细的RGB转化成灰度计算过程-RGB转化成灰度		11-17	
从RGB色转为灰度色算法 一、基础 对于彩色转灰度，有一个很著名的心理学公式： Gray = R*0.299 + G*0.587 + B*0.114 二、整数算...	lxiaoxiaot的专栏	2110	
RGB  转换为灰度  的原理 RGB彩色  中，一种彩色由R（红色），G（绿色），B（蓝色）三原色按比例混合而成。  的基本单元是一个像素，一个像素需要3块表示，分别代...	Bebr的博客	1万+	
RGB 转换为灰度  、二值化  from PIL import Image I = Image.open('lena.png') I.show() L = I.convert('L') #转化为灰度  L = I.convert('1') #转化为二值化  L.show() 附:PIL可以对  ...	小博要变强啊~	440	
opencv学习5: cvtColor RGB  转灰度  图像原理 简介 将彩色  转成为灰度  的过程成为  的灰度  化处理。彩色  中的每个像素的颜色有R、G、B三个分量决定，而每个分量有255中值可取，...	weixin_42713739的博客	3104	
©2022 CSDN 皮肤主题：大白 设计师：CSDN官方博客 返回首页			