图像处理学习笔记（一）-灰度和对比度处理

## 灰度处理

在RGB的领域里， 图像元素的颜色都由红、绿、蓝三原色组成。假如原来某点的颜色为RGB(R，G，B)。我们通过一系列的方法求得Gray后，将原来的RGB(R,G,B)中的R,G,B全部用Gray替换，形成新的颜色RGB(Gray,Gray,Gray)，用它替换原来的RGB(R,G,B)，这就是就是灰度图了。通常我们可以通过下面几种方法，将其转换为灰度：  
     浮点算法：Gray=R\*0.3+G\*0.59+B\*0.11  
     整数方法：Gray=(R\*30+G\*59+B\*11)/100  
     移位方法：Gray=(R\*28+G\*151+B\*77)>>8;

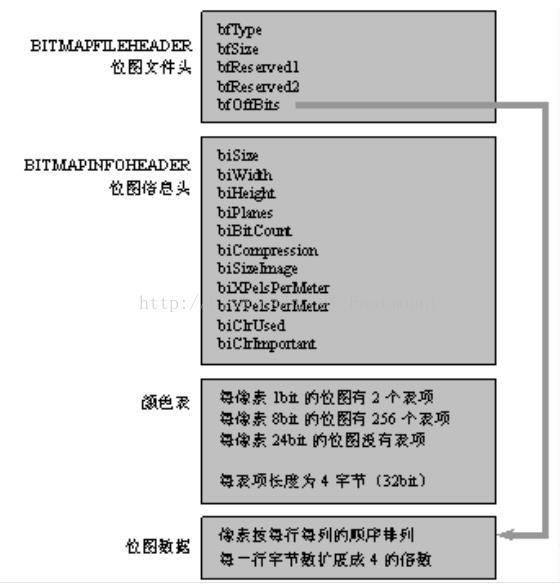
     仅取绿色：Gray=G；

     加权平均值算法：根据光的亮度特性，灰度公式应当是： R=G=B=R\*0.299+G\*0.587+B0.144。

     平均值法：Gray=（R+G+B）/3（我们灰度处理的实现，采用了此方法将彩色图转变为灰度图）；  
     24位真彩图与8位的灰度图的区别就在于，真彩图文件中没有调色板，灰度图有调色板，真彩图中的象素矩阵是RGB值，灰度图中的象素矩阵是调色板索引值。我们只简单的改变象素矩阵的RGB值，来达到彩色图转为灰度图，并没有添加调色板；程序未实现添加调色板。

程序是通过VC++ MFC单文档视图实现显示BMP格式图片，并通过Bitmap进行灰度处理功能。由于还处于学习的初步阶段，图像的灰度处理，只实现了BMP位图的转换，没有进一步完成其它图像格式——〉位图——〉灰度处理。后续可以进一步研究图像间的转换，并和灰度处理结合起来。

位图的图像信息包含在位图中，所以，学习图像处理，必须了解位图的图片格式：BMP图片格式如下图所示。获取BMP图像的一些数据，如图像宽度、高度、像素大小等，后面的处理与之相关。



BMP位图文件格式

位图的灰度处理过程中，对位图文件的读取和位图的显示使用了MFC固有的库函数，其它算法处理过程，均为代码实现。

首先，新建MFC单文档项目"MFC AppWizard(exe)"，项目名为ImgPro。在"资源视图"中，点击"Menu->IDR\_MAINFRAM"，并修改菜单视图，增加了DIP和灰度处理的菜单按钮。

其次，在ImgProcView的类中建立成员函数，包括ShowBitmap()、ReadBmp()、SaveBmp()以及相关的成员变量。实现读取位图信息，显示以及存储。实现过程在ImgProcView.cpp中。

最后，在ImgProcView的类中实现对按钮DIP——>灰度处理的方法函数OnShowHd()，上一步中的ReadBmp()为我们准备了BMP位图头的详细信息，从而利用灰度值算法，使用RGB均值作为灰度值，实现灰度处理，并显示在屏幕中。

## 对比度处理：

算法原理：g(x) = f(x)\*alpha + 亮度均值\*(beta-alpha)

对比度算法过程：

1.计算图像的RGB像素均值；

2.对图像的每个像素点与均值求差值；

3.对去掉平均值以后的像素点的RGB值，乘以对比度系数

4.对上一步像素RGB值，加上均值乘以亮度系数的积

5.从而得到像素点新的RGB值

实现过程在 CImgProcView::OnShowDbd()中。即菜单上—>按钮DIP——>对比度处理的实现函数OnShowHd()