



北京交通大学  
BEIJING JIAOTONG UNIVERSITY

# 《数字图像处理》 实验报告

实验名称:	实验 2
学 号:	
姓 名:	
学 院:	计算机与信息技术学院
日 期:	2022 年 12 月 25 日

## 目录

1. 实验目的 .....	3
2. 实验内容 .....	3
3. 实验要求 .....	3
4. 实验思路 .....	3
5. 实验过程 .....	4
5.1. 图形界面设计 .....	4
5.2. BMP 图片的打开.....	4
5.3. 彩色图灰度化 .....	5
5.4. 直方图均衡化 .....	6
6. 实验汇总 .....	7

## 1. 实验目的

完成对 BMP 图像的文件解析和对图像数据施加直方图均衡化算法。

## 2. 实验内容

- 1) BMP 图像文件解析功能的例程为“实验 2 附件.pdf”文件第 10 页开始的 LoadBitmap ( )函数所示。
- 2) 对图像数据矩阵施加直方图均衡化算法的例程为“实验 2 附件.pdf”文件第 13 页开始的“五、为“直方图均衡化”按钮添加响应”中的函数。
- 3) 然后结合程序运行的效果，对自己实现的直方图均衡化算法的应用效果进行对分析。

## 3. 实验要求

该实验属于模仿性实验，需个人完成，编程环境不限（任何 VS 版本或其他 IDE 环境均可）。不能调用任何图像处理相关的函数、工具包，需仿照或改写“实验 2 附件.pdf”的例程进行

## 4. 实验思路

实验过程如下：

- 1) 学习 BMP 文件格式
- 2) 学习 pyqt 架构
- 3) 画出实验程序架构图
- 4) 代码编写

实验选取 python 语言进行编写，GUI 框架选用 pyqt 进行编写。核心就是对 BMP 文件读取，这需要掌握 BMP 格式。编写相关提取函数，把图像数据提取为三维矩阵。建立图像类把图像显示在图像窗口上，把直方图均衡化函数封装为按钮函数，把处理后的图像显示出来即可。

## 5. 实验过程

### 5.1. 图形界面设计

本实验的编程语言是 python3.9, gui 工具是 pyqt6。基本界面设计使用 qtdesigner 工具进行拖拽设计。



图表 5-1 图形界面设计

考虑到实际实验要求，我们把实验 2 和实验 3 的要求进行统一在一个程序上完成。每一个按钮对应一个功能。实验 2 核心功能为，打开 BMP 图像和直方图均衡化。

### 5.2. BMP 图片的打开

BMP（全称 Bitmap）是 Windows 操作系统中的标准图像文件格式，可以分成两类：设备相关位图（DDB）和设备无关位图（DIB），使用非常广。它采用位映射存储格式，除了图像深度可选以外，不采用其他任何压缩，因此，BMP 文件所占用的空间很大。BMP 文件的图像深度可选 8bit 及 24bit。BMP 文件存储数据时，图像的扫描方式是按从左到右、从下到上的顺序。由于 BMP 文件格式是 Windows 环境中交换与图有关的数据的一种标准，因此在 Windows 环境中运行的图形图像软件都支持 BMP 图像格式。

#### 思路：

灰度图和彩色图同一的部分直接读取，但是不同色位后边的数据部分分开不同情况进行讨论。8bit 的灰度图是需要进行颜色表的映射，24bit 直接在数据存储 RGB 信息。这样就可以直接打开灰度图和彩色图，增加了程序的健壮性。

#### 程序验证：



图表 5-2 打开灰度图



图表 5-3 打开彩色图

程序验证成功。

### 5.3. 彩色图灰度化

我们通过组合 RGB 三个通道的方法进行转换彩色图为灰度图。  
具体比例如下：

$$0.11 * r + 0.59 * g + 0.3 * b$$

通过这样的方法就可以直接得到灰度图，并以此为基准，进行下边的变换。



图表 5-4 彩图灰度化

选取的图片是上一节展示的彩色老虎图，现在可以直接转换为灰度图。同时这里在程序整体逻辑上有一点不同。其他的算法变换，改变的施加算法之后的图像数据，但是灰度化是直接修改了原始的图像数据，从而实现其他算法那是在灰度化的基础上进行的。

## 5.4. 直方图均衡化

算法原理：

- ①依次扫描原始灰度图像的每一个像素，计算出图像的灰度直方图；
- ②计算灰度直方图的累积分布函数；
- ③根据累积分布函数和直方图均衡化原理得到输入与输出之间的映射关系。
- ④最后根据映射关系得到结果进行图像变换

其中 python 中 numpy 对这样的数据统计计算支持的很好，可以通过向量化很快的解决问题。



图表 5-5 直方图均衡化的效果

## 6. 实验汇总

本实验主要是认识性实验，借着这次实验，完成了对课程主要内容的梳理和回顾。特别是在有实操工具的情况下，加深了对课程内容的理解。

- 1) 一开始读取 BMP 文件时，文件默认的长宽方向和最后 PyQt 的默认方向是不一样的，一开始不知道，显示出来的图片永远是雪花状。网上也少有人提到这个，后来是在官方文档中发现示例代码表示顺序，才修改，遂成功。
- 2) 显示在 GUI 界面上。一开始尝试了很多方法，可能是没有太多入门的指导，网上对于直接显示图像数据流的指引很少，也可能是没有找到相关有用信息。最后锁定了 QImage、QPixmap、QGraphics 一组类。
- 3) 数据格式一定要注意。Numpy 数组默认 int32，但是我们图像中是用 8bit 表示一位，需要使用 uint8。这种 bug 基本很难发现，本人第一次遇到这种需要专门指定数据长度的 bug。