



重慶理工大學

实验报告

实验名称: 数字图像基本处理实验

专业班级:

学 号:

姓 名:

联系电话:

指导老师:

实验时间: 第7周星期三

重庆理工大学电气与工程学院
电气与控制工程实验中心

【成绩】

【教师签名】

【实验目的】

1. 了解BMP图像基本格式和数字图像处理的基本原理;
2. 了解图像二值化自适应阈值法的基本原理;
3. 掌握用C语言编写DSP图像二值化自适应阈值算法程序的方法。

【实验原理及内容】

1. 图像反色原理:

反色的实际含义是将R、G、B值反转。若颜色的量化级别是256,则新图的R、G、B值为255减去原图的R、G、B值。这是针对的是所有图,包括真彩图、带调色板的彩色图(又称为伪彩色图)和灰度图。真彩图不带调色板,每个像素用3个字节,表示R、G、B三个分量。带调色板的彩色图,其位图中的数据只是对应调色板中的一个索引值,只需将调色板中的颜色反转,形成新调色板,而位图数据不用动,就能实现反转。

2. 图像二值化原理:

图像二值化是将一幅灰度图像转换为黑白二值图像的过程。其基本原理是将图像中的灰度级别进行分类,将其分为两个区域,其中一个区域的灰度值被赋为黑色,另一个区域则被赋为白色。

3. 数字图像的表达方法:

将图像分割为N行M列的小方块,叫做“像素点”;比如将一张有14400个像素点的图片放大后可以看到由许多小方块(像素点)组成。每个点需要记录RGB三种颜色,若每一种颜色用8位(256种取值),则一个点需要3个字节。

4. 灰度图:

每个像素点的RGB三种颜色取值相等,看起来就只有亮度的变化,只需要记录一个值,叫做灰度值。如果是8位灰度图,则每个像素点需要1个字节。比如某一点灰度值为100,表示这一点RGB(100, 100, 100);黑色对应灰度值为0: RGB(0, 0, 0);白色对应灰度值为255。

5. 灰度图的二维数组表示:

图像数据一般是线性存储(即按顺序以一维的形式存储), $f(i, j)$,也就是第i行j列的像素点的灰度值存储的偏移地址为 $i * x + j$ (其中, x 表示图像的宽度)。

【实验设备】

一台装有 CCS 3.3 的电脑。

【实验方案及步骤】

1. 把 "imagefun1.c" 源文件中的函数 "alloc_mem(buffer_size)" 中 "if (buffer_size > 0x2000)" 一行修改为 "if (buffer_size > 0x4000)" 以使其能支持 120x120 像素的 bmp 图像。
2. 编写图像二值化处理程序基础上修改, 由于数据量增加较大, 需要修改 CMD 链接命令文件, 增加代码段长度和 "heap" 长度, 否则编译会出错。
3. 要观测的数据必须定义为全局变量, 即只能在函数体外部定义。

二、图像反色运算:

1. "Build" 工程, 加载 .out 文件 (注意: 在 "debug" 文件夹中)。
2. 从 "image" 文件夹中选择一张 120*120 大小的灰度图, 复制到 "debug" 文件夹中。
3. 运行 (RUN 或 F5), 按提示输入文件名 (注意: 带后缀名 bmp)。
4. 查看输入和输出图像数据并记录图像数据。

三、图像二值化运算:

1. 先将文件 "imagefun1.c" 备份, 将反色运算函数 Fanse (ip, jp, k, ly) 的功能改为二值化运算。
2. 输入图像灰度级归一直方图 $h(i)$ 其表示就是表示每种灰度级所占的比例, 可以先算出每种灰度级的点数 $GrayNum[i]$, 再除以总的像素点数 $TotalPixel$ 即可 (计算 $GrayNum[i]$ 的方法, 避免增大时间复杂度: $GrayNum[i]$ 除以 $TotalPixel$ 时要避免整除)。
3. 计算直方图的零阶累积矩 $w(k)$ 和一阶累积矩 $\mu(k)$ 时, 可对公式进行变形: $w(0) = h(0)$, $w(k) = w(k-1) + h(k)$ $k=1, 2, 3, \dots, 255$ 这样可以通过一重循环即可算出 $w(k)$ 。 $\mu(k)$ 的公式也可以做类似的变形。
4. 类分离指标 $\delta(k)$, 公式中分子的计算可以变形为:
$$temp = \mu(w(k)) - \mu(k) \quad \delta(k) = \frac{temp^2}{w(k)[1-w(k)]}$$
5. 比较三种二值化计算的优劣。
6. 理解自适应阈值算法的物理意义, 什么情况下效果好? 什么情况下效果较差?

【实验电路图】

【实验数据处理及分析】

1. 有实验中对比观察了固定阈值($T=128$)、灰度均值作为阈值($\mu_T=107.0231$, $T=107$)、自适应阈值($T=142$)三种算法对图像进行二值化处理的效果,发现灰度均值作为阈值时图像中人物轮廓最清晰,阈值过低或过高都会使原图中的特征无法很好地被分割出来。利用直方图分析可以帮助我们找到最佳阈值,使二值化后的图像具有最佳的黑白分离性。

2. ① 8位灰度图像:每个像素由一个8位(1字节)表示灰度值,范围从0到255。读取和设置像素灰度值时,根据图像宽度计算出当前像素在行内的偏移量,然后直接读取或修改相应字节即可。

② 4位灰度图像:每个像素由4位表示灰度值,共16种可能的灰阶。由于计算机存储以字节为单位,因此通常两个像素共用一个字节。读取时需先定位到正确的字节,然后通过位操作提取或设置指定像素的4位灰度值。③ 1位灰度图像:每个像素由1位表示灰度值,只有黑(0)和白(1)两种状态。同样地,一个字节通常会包含8个像素的信息,需进行位操作读取或设置单个像素值。

④ 要读取4位灰度图像的第 i 个像素,需先计算出所在字节的偏移量,然后读取该字节,通过位移和掩码操作获取灰度值;设置像素值时类似,先读取原字节,修改对应位,然后写回内存。

【实验结论】

1. 直方图均衡化能有效改善图像的整体对比度,尤其是针对曝光不足或过曝的图像,它能扩大图像灰度动态范围。
2. 不同的阈值选择会导致二值图像的质量和内容解析度大相径庭,最佳阈值的选择可以根据直方图分布等来确定。
3. 对于大多数灰度图像来说,图像中的物体和背景是有明显区别的。通过选择阈值,区分图像和背景,以便对物体进行处理。设定一个阈值,若像素的颜色值大于阈值则取255,否则就取0。

【思考题】

1. $h(i)$ 的下标 i 表示的是灰度级,取值范围是 $0 \sim 255$ 。
2. ①固定阈值:优点是简单易行,计算速度快,当图像背景和前景对比度较高时能得到满意的结果。缺点是对图像的光照条件变化敏感。②灰度级均值作为阈值:优点是在一定程度上考虑了图像的整体亮度分布。缺点是无法适应图像亮度不均匀的情况。③自适应阈值:优点是根据图像的局部特性来确定阈值,二值化分割的准确性高。缺点是计算量相比固定阈值要大。
3. ①自适应阈值算法能根据不同区域的局部特性来自动调整该区域的阈值;②允许在图像的不同部位设定不同阈值,类似生物体对外界刺激的自适应反应机制;③可计算每个像素及其邻域的灰度统计特性,提高准确性。
4. 归一化直方图 $h(i)$ 的分布特性较好地反映出图像的内在灰度结构和对比特性,自适应阈值算法就越有可能取得更好的分割效果。物理意义:直方图的分布特性直接影响着图像中不同区域的灰度对比度和边界清晰度。