

2023年数字图像处理实验报告总结(优质8篇)

通过撰写实践报告，我们可以更好地了解自己的优点和不足，为今后的工作做好准备。在下列整改报告范文中，我们可以看到如何指导团队成员进行整改，提高工作效率和质量。

数字图像处理实验报告总结篇一

实验课程名称开课学院指导教师姓名学生姓名学生专业班级

200

--200学年第学期

实验教学管理基本规范

实验是培养学生动手能力、分析解决问题能力的重要环节；实验报告是反映实验教学水平与质量的重要依据。为加强实验过程管理，改革实验成绩考核方法，改善实验教学效果，提高学生质量，特制定实验教学管理基本规范。

照执行或暂不执行。

报告外，其他实验项目均应按本格式完成实验报告。

定比例。各部分成绩的观测点、考核目标、所占比例可参考附表执行。各专业也可以根据具体情况，调整考核内容和评分标准。

在学生离开实验室前，检查学生实验操作和记录情况，并在实验报告第二部分教师签字栏签名，以确保实验记录的真实性。

实验项目后，教师应按学生姓名将批改好的各实验项目实验报告装订成册，构成该实验课程总报告，按班级交课程承担单位(实验中心或实验室)保管存档。6、实验课程成绩按其类型采取百分制或优、良、中、及格和不及格五级评定。

实验课程名称：__通信原理_____

更多相关热门文章推荐阅读：

数字图像处理实验报告总结篇二

实验目的：本实验内容旨在让学生通过用VC等高级语言编写数字图像处理的一些基本算法程序，来巩固和掌握图像处理技术的基本技能，提高实际动手能力，并通过实际编程了解图像处理软件的实现的基本原理。为学生进一步学习数字摄影测量、遥感和地理信息系统等专业课程以及应用图像处理解决实际问题奠定基础。

二、实验原理和方法

(1)raw格式到bmp格式的转换：

raw格式[]raw格式文件是按照数字图像组成的二维矩阵，将像素按行列号顺序存储在文件中。这种文件只含有图像像素数据，不含有信息头，因此，在读图像时，需要根据文件大小，计算图像所包含的行列号，或者需要事先知道图像大小(矩阵大小)[]raw文件按图像上行到下行、左列到右列顺序存储。

bmp格式[]bmp文件数据区按图像上下行到上行、左列到右列顺序存储到数据区[]bmp文件由文件头、信息头、颜色表、数据区四个部分组成。

做raw格式文件到bmp格式文件的转化，先要为bmp格式文件

申请四部分的内存：文件头，位图信息头，颜色表，图象数据，然后根据输入值以及raw文件信息[]bmp格式文件信息计算出这几部分的值，赋给他们，写到bmp文件中去。

(2) 灰度图象的线性拉伸：

灰度变化是点运算，将原图象的每个像素的灰度值改成线性变化之后的灰度即可。

灰度的线性变换就是指图像的中所有点的灰度按照线性灰度变换函数进行变换。灰度变换方程如下：

该方程为线性方程。式中参数为输入图像的像素的灰度值，参数为输出图像的

灰度值。

设原图象的灰度范围为[a,b],变化之后的范围为[a',b'][]则：

$$fa=(b'-a')/(b-a)$$

$$fb=- (b'-a')/(b-a)*a+a'$$

如果算出来的值大于255，则让它等于255，小于0则让其等于0。

(3) 局部处理 (3*3高通滤波， 3*3低通滤波)：

局部处理在处理某一像素时，利用与该像素相邻的一组像素，经过某种变换得到处理后图像中某一点的像素值。目标像素的邻域一般是由像素组成的二维矩阵，该矩阵的大小为奇数，目标像素位于该矩阵的中央，即目标像素就是区域的中心像素。经过处理后，目标像素的值为经过特定算法计算后所得的结果。

实际上都是利用卷积来实现的，卷积往往用一个矩阵表示，将矩阵的中心对齐某个像素，矩阵中的值乘到相应的像素中去，然后将所有乘积加起来就得到中心像素的灰度值。边界像素不做处理，仍为原来的灰度值。求出的像素灰度值若超过 $[0\sim 255]$ ，则向离其最近的属于该范围的`像素值靠拢。

3*3低通滤波的算子见表1。

3*3高通滤波的算子见表2。

表格1

1/9

1/9

1/9

1/9

1/9

1/9

1/9

1/9

1/9

表格2

-1

-1

-1

-1

9

-1

-1

-1

-1

(4) 图像几何处理 (图像平移, 图像缩放):

对于图像平移来说, 若平移量是 (tx, ty) 像素在原图像中的坐标为 (x_0, y_0) 则变化后的坐标为 (x_1, y_1) $x_1 = x_0 + tx$ $y_1 = y_0 + ty$ 平移只需改变像素的灰度值, 不必改变位图信息头和调色板内容。

对于图像缩放, 假设放大因子为 $ratio$ 缩放的变换矩阵为:

图像信息头中新图像的宽度和高度都变为原来宽度和高度分别与水平垂直比例的乘积, 图像大小变为新宽度(变为4的整数倍) 与新高度的乘积。

(5) 灰度图像中值滤波:

中值滤波也属于局部处理的一种, 将窗口中的各个像素排序之后排序, 取中值赋给模板中心的像素, 所以窗口中个数一般是基数。

我用的中值滤波窗口是十字丝的9个数的窗口。

(6) 灰度图象边缘检测:

边缘检测有三种算子[]roberts[]prewit[]sobel[]三种算子都是做一阶差分的，通过算子算出各个像素的梯度值，将水平梯度的绝对值和垂直梯度的绝对值相加，若此梯度值大于某个阈值，则将其灰度值赋为255，否则赋为0。

(7) 图象旋转:

图像旋转一般是以图像中心为中心顺时针旋转，利用图像的四个角点求出图像旋转后的大小。

先计算以图像中心为原点坐标系下原图像四个角点的坐标值，按照旋转矩阵计算其旋转之后的坐标值，根据四个角点的新坐标值计算出最大宽度和高度作为新图像的宽度和高度值，按照计算值修改位图信息头，申请一块新内存，存储旋转后图像的灰度值。

旋转矩阵如下:

同样要求各个像素在原图像中的坐标，先将新图像的坐标系平移到图像中心，做逆时针旋转，然后再平移到屏幕左上角，然后将原图像对应坐标的值赋给新图像。

(8) 图象二值化:

判断分析法: 假定图像的灰度区间为 $[0, l-1]$ 则选择一阈值 t 将图像的像素分为两组。

为最大值所对应的 t_0 就是所求判断分析法的分割阈值。

搜寻到阈值之后，灰度值小于阈值的像素赋0，其他的赋1，

修改文件信息头，调色板，申请新内存。

(9) 图象直方图：

统计各灰度值出现的频数，以及像素的总个数，用频数除以总个数作为频率，以灰度值作为横坐标，频率作为纵坐标绘图。

三、实验过程和步骤

首先要建立一个基于mfc的多文档工程，将视图基类改为滚动视图，以自己的学号命名。

我用的是书上给的cdib类，类里面有获取bmp宽度，高度的函数，有指向位图信息头的指针，指向图象数据的指针，因此我在文档类(doc类)里定义了一个cdib类的对象，打开以及保存文件的时候利用这个对象去调用cdib里读取与存储文件的函数，并且可以利用这个对象的两个指针对打开的图象进行各种操作。

格式到bmp格式转换：

首先建立一个rawtobmp的对话框，在上面加上四个编辑框(一个输入打开文件的路径一个输入保存文件的路径，另两个)，两个按钮，以及默认的确，取消按钮。利用类向导插入此对话框类，并且为前两个编辑框定义cstring的两个变量，用来存储打开与保存文件的路径。同时为两个浏览按钮添加消息响应函数，在消息函数里创建cfiledialog对象，利用此对象的函数将两个路径值赋给前两个编辑框的成员变量。再为ok键添加消息响应函数，分别定义bmp格式文件前三部分数据变量，计算出各变量的值，并且利用一个cfile对象获取raw图象的数据，利用另一个cfile对象将数据存储在所输入的路径的文件中去[]cfile对象的read函数会自动创建一个文件。

然后在菜单上新建一个菜单，为菜单添加消息响应函数，在其消息响应函数里创建rowtobmp对话框。这样点击菜单后就会弹出一个对话框，按确定键之后就可以读取raw文件并且存储bmp文件，完成整个消息循环。

2. 灰度图像的线性拉伸：

数字图像处理实验报告总结篇三

数字图像处理的信息源基本是二维数据，处理信息量较大。对计算机运算速度、存储空间等要求高。

数字图像处理的传输频带要求高。与语音信息相比，传输占用的频带要高几千倍。所以，就对图像压缩技术形成了有效的研究需求。

数字图像中每一个像素并非独立的，相互关联性较高。很多相邻像素之间有相同或相近的数值。

所以，图像处理技术中数据压缩的可能性较高。

由于图像是视觉三维意识的二维映射。因此，计算机要识别和处理三维形态就要进行适当的模糊处理或附加量的匹配。

处理后的数字图像是形成人为视觉理解和应用评估的依据，因此处理结果必然受到人为的意识形态的影响。所以，在计算机视觉研究中，人为的感知机理必然对计算机视觉的研究产生影响。

2.2数字图像处理技术的基本方法

由于在图像处理中，像素阵列很大，在空间域中涉及计算量对计算机硬件的要求非常高，所以，必须引入图像的函数变换进行计算简化。利用函数变换的间接处理方法，如傅里叶

变换、离散余弦变换、Walsh变换等，不但可以降低计算强度，而且可以得到高效的计算。

图像的像素矩阵编码压缩技术可以降低定义图像数据的比特数量，并减少图像处理和传输时间，进而节省存储空间。

图像的增强处理过程中不涉及图像质量降低的主要成因，目的是要突出图像矩阵中敏感的像素群。图像的复原处理需要对图像质量降低的主要成因进行调查，进而采取相应滤波处理技术，复原和重构原有的像素矩阵。

图像分割处理是数字图像处理中的关键处理手段之一。是将图像中敏感的主要像素群作为主要处理对象，包括区域特征、边缘特征等，是对敏感像素群进行识别、理解和分析的基础数据特征。

作为最简单的二值图像可采用其几何特性识别物体的特性，一般图像的理解方法采用二维形状理解，它有边界理解和区域理解两类方法。对于三维物体理解，有体积理解、表面理解、圆柱体的广义理解等。

图像识别处理基本采用传统的模式识别方式，有统计模式识别和结构模式识别两种，随着研究广泛进行，人工神经网络模式识别和模糊模式识别也得到不同程度的重视，进行广泛研究。

3 数字图像处理技术的优点

数字图像处理技术与模拟图像处理技术在基本原理上的差异之处，是数字化处理技术不可能在图像的传输、存储或复制等操作处理过程中，使图像质量有所降低。图像在数字化过程中精确地再现了原模拟图像，则在数字图像处理过程中就能够确保无损于图像的各项数字化指标。

依据现有的数字化技术，在图像数字化设备的性能满足要求的情况下，完全可以数字化模拟图像成为目标精度的二维数组。目前的数字化扫描仪能够将各个像素的灰度等级量化处理为48位甚至更高，这就说明数字化图像的精度可以满足几乎所有的应用需求。对于数字化处理设备来说，无论二位数组的规模，也不考虑像素的量化位数，处理过程基本是相同的。从原理的角度来看，无论图像的量化精度达到什么程度，在技术上都是可以完成的，只需要在处理修改过程中的数组技术参数。而在图像的模拟量化处理过程中，要想把量化处理精度提升，就需要采用非常高等级的硬件设备或大规模提升处理装置的技术参数等级，从技术经济方面考虑，是非常不合理的。

图像的信息来源是多样化的，一般情况下是可见光的感光图像，也可以是不可见光的波谱图形图像。从图像映射物体感官的角度，微观至电子显微镜采集的图像，宏观至大规模空天望远镜采集的图像。

不同信息来源的图像转换为数字化编码后，都可以表示为二维数组的灰度级图像，进而完成数字化处理过程。对于图像的不同信息来源，使用对应的图像信息量化技术，图像的数字化处理技术可以用于任何一类图像。

图像数字化处理技术基本上可以归类为图像的质感提升、像素分析和区域重构等手段。因为图像的模拟技术处理从数学上分析只可以进行线性分析，就局限了模拟图像处理技术可以完成的工作需求。

数字图像处理实验报告总结篇四

提高计算机对数字图像处理的速度，提高采集分辨率和显示分辨率，提高多媒体技术关键中图像数据的压缩，进行计算机识别和理解研究中按照人类的认知和思维方式工作并考虑到主观概率和非逻辑思维技术，规划统一的标准以实现图像

的处理、传输和存储研究健康发展，以上几点都是数字图像处理技术合理发展的基本融汇技术基础。

同时，信息数据量更大的三维数字图像必将得到广泛应用研究，图像与图形相互融合后形成三维成像或多维成像的发展方向也正在众多应用中广泛推进。

5总结

数字图像处理技术在社会的每个行业、每个领域都得到广泛的应用，数字图像处理的技术应用随时、随处都可以见到，得到充分的研究发展和应用推广，还不能充分满足日益增长的技术需求。数字图像处理技术不断地在自身发展和完善的同时，还与多个计算机分支学科的发展密不可分，有多个新的技术方向需要研究和创新，对数字图像处理技术的发展方向进行研究、探讨的重要性就显得尤为突出。

参考文献：

[1]朱睿。数字图像处理技术现状与展望[j].中国科技博览，(14)：7-28.

[4]谭海艳。数字图像压缩综述[j].科技经济市场，2011（8）。

数字图像处理实验报告总结篇五

实验目的：本实验内容旨在让学生通过用vc等高级语言编写数字图像处理的一些基本算法程序，来巩固和掌握图像处理技术的基本技能，提高实际动手能力，并通过实际编程了解图像处理软件的实现的基本原理。为学生进一步学习数字摄影测量、遥感和地理信息系统等专业课程以及应用图像处理解决实际问题奠定基础。

二、实验原理和方法

(1) raw格式到bmp格式的转换:

raw格式 raw格式文件是按照数字图像组成的二维矩阵，将像素按行列号顺序存储在文件中。这种文件只含有图像像素数据，不含有信息头，因此，在读图像时，需要根据文件大小，计算图像所包含的行列号，或者需要事先知道图像大小(矩阵大小) raw文件按图像上行到下行、左列到右列顺序存储。

bmp格式 bmp文件数据区按图像上下行到上行、左列到右列顺序存储到数据区 bmp文件由文件头、信息头、颜色表、数据区四个部分组成。

做raw格式文件到bmp格式文件的转化，先要为bmp格式文件申请四部分的内存：文件头，位图信息头，颜色表，图象数据，然后根据输入值以及raw文件信息 bmp格式文件信息计算出这几部分的值，赋给他们，写到bmp文件中去。

(2) 灰度图象的线性拉伸:

灰度变化是点运算，将原图象的每个像素的灰度值改成线性变化之后的灰度即可。

灰度的线性变换就是指图像的中所有点的灰度按照线性灰度变换函数进行变换。灰度变换方程如下:

该方程为线性方程。式中参数为输入图像的像素的灰度值，参数为输出图像的

灰度值。

设原图象的灰度范围为[a,b],变化之后的范围为[a',b'] 则:

$$fa=(b'-a')/(b-a)$$

$$fb = -(b' - a') / (b - a) * a + a'$$

如果算出来的值大于255，则让它等于255，小于0则让其等于0。

(3) 局部处理 (3*3高通滤波， 3*3低通滤波)：

局部处理在处理某一像素时，利用与该像素相邻的一组像素，经过某种变换得到处理后图像中某一点的像素值。目标像素的邻域一般是由像素组成的二维矩阵，该矩阵的大小为奇数，目标像素位于该矩阵的中央，即目标像素就是区域的中心像素。经过处理后，目标像素的值为经过特定算法计算后所得的结果。

实际上都是利用卷积来实现的，卷积往往用一个矩阵表示，将矩阵的中心对齐某个像素，矩阵中的值乘到相应的像素中去，然后将所有乘积加起来就得到中心像素的灰度值。边界像素不做处理，仍为原来的灰度值。求出的像素灰度值若超过 $[0 \sim 255]$ ，则向离其最近的属于该范围的像素值靠拢。

3*3低通滤波的算子见表1。

3*3高通滤波的算子见表2。

表格1

1/9

1/9

1/9

1/9

1/9

1/9

1/9

1/9

1/9

表格2

-1

-1

-1

-1

9

-1

-1

-1

-1

(4) 图象几何处理(图象平移, 图象缩放):

对于图像平移来说, 若平移量是 (tx, ty) 像素在原图像中的坐标为 (x_0, y_0) 则变化后的坐标

为 $(x_1, y_1) \rightarrow x_1 = x_0 + tx, y_1 = y_0 + ty$ 平移只需改变像素的灰度值，不必改变位图信息头和调色板内容。

对于图像缩放，假设放大因子为ratio，缩放的变换矩阵为：

图像信息头中新图像的宽度和高度都变为原来宽度和高度分别与水平垂直比例的乘积，图像大小变为新宽度(变为4的整数倍)与新高度的乘积。

(5) 灰度图像中值滤波：

中值滤波也属于局部处理的一种，将窗口中的各个像素排序之后排序，取中值赋给模板中心的像素，所以窗口中个数一般是基数。

我用的中值滤波窗口是十字形的9个数的窗口。

(6) 灰度图像边缘检测：

边缘检测有三种算子，roberts，prewit，sobel，三种算子都是做一阶差分的，通过算子算出各个像素的梯度值，将水平梯度的绝对值和垂直梯度的绝对值相加，若此梯度值大于某个阈值，则将其灰度值赋为255，否则赋为0。

(7) 图像旋转：

图像旋转一般是以图像中心为中心顺时针旋转，利用图像的四个角点求出图像旋转后的大小。

先计算以图像中心为原点坐标系下原图像四个角点的坐标值，按照旋转矩阵计算其旋转之后的坐标值，根据四个角点的新坐标值计算出最大宽度和高度作为新图像的宽度和高度值，按照计算值修改位图信息头，申请一块新内存，存储旋转后图像的灰度值。

旋转矩阵如下：

同样要求各个像素在原图像中的坐标，先将新图像的坐标系平移到图像中心，做逆时针旋转，然后再平移到屏幕左上角，然后将原图像对应坐标的值赋给新图像。

(8) 图象二值化：

判断分析法：假定图像的灰度区间为 $[0, 1]$ 则选择一阈值 t 将图像的像素分为两组。

为最大值所对应的 t 就是所求判断分析法的分割阈值。

搜寻到阈值之后，灰度值小于阈值的像素赋0，其他的赋1，修改文件信息头，调色板，申请新内存。

(9) 图象直方图：

统计各灰度值出现的频数，以及像素的总个数，用频数除以总个数作为频率，以灰度值作为横坐标，频率作为纵坐标绘图。

三、实验过程和步骤

首先要建立一个基于mfc的多文档工程，将视图基类改为滚动视图，以自己的学号命名。

我用的是书上给的cdib类，类里面有获取bmp宽度，高度的函数，有指向位图信息头的指针，指向图象数据的指针，因此我在文档类(doc类)里定义了一个cdib类的对象，打开以及保存文件的时候利用这个对象去调用cdib里读取与存储文件的函数，并且可以利用这个对象的两个指针对打开的图象进行各种操作。

格式到bmp格式的转换：

首先建立一个rawtobmp的对话框，在上面加上四个编辑框(一个输入打开文件的路径一个输入保存文件的路径，另两个)，两个按钮，以及默认的确认，取消按钮。利用类向导插入此对话框类，并且为前两个编辑框定义cstring的两个变量，用来存储打开与保存文件的路径。同时为两个浏览按钮添加消息响应函数，在消息函数里创建cfiledialog对象，利用此对象的函数将两个路径值赋给前两个编辑框的成员变量。再为ok键添加消息响应函数，分别定义bmp格式文件前三部分数据变量，计算出各变量的值，并且利用一个cfile对象获取raw图象的数据，利用另一个cfile对象将数据存储在所输入的路径的文件中去[]cfile对象的read函数会自动创建一个文件。

然后在菜单上新建一个菜单，为菜单添加消息响应函数，在其消息响应函数里创建rowtobmp对话框。这样点击菜单后就会弹出一个对话框，按确定键之后就可以读取raw文件并且存储bmp文件，完成整个消息循环。

2. 灰度图象的线性拉伸：

向图像数据部分的指针m_dib.m_pbits[]在数组中查出每个像素变化后的灰度值，并将此值赋给指针m_pbits指向的内存。刷新视图。

然后在菜单中加上线性拉伸的菜单，为该菜单的id添加消息响应函数，在该函数中创建对话框，并调用文档类线性拉伸的函数，将对话框的两个成员变量传给此函数。

3. 局部处理：

在文档类里添加低通滤波和高通滤波的成员函数，在函数中使用m_dib对象中指向图像数据部分的指针m_pbits[]首先申

请一个内存，将原来图像的灰度值存储起来，然后定义9个byte类型的指针，利用双重嵌套循环，在循环中每次用这9个指针指向复制图像对应模板中的9个数，然后按照模板中的数值计算出中心像素的灰度值，判断是否超过范围，如果超过范围则做相应的处理，否则将此值直接赋给m_pbits中对应的中心像素。循环之后刷新视图。

添加局部处理的菜单，为菜单设置消息响应函数，在菜单消息响应函数中调用文档类的函数，完成对m_dib的处理。

4. 图像几何变换：

建立平移对话框，定义两个成员变量，分别存储输入的水平位移和垂直位移。

在文档类里添加平移函数，申请一块新内存复制原图像的信息，在函数中将

外层循环变量i视为纵坐标，内层循环变量j视为横坐标，通过双重循环，对每个像素，求出其在原图像中的坐标(i0[j0])将复制图像中的对应(i0[j0])的像素灰度值赋给m_dib.m_pbits指针中的图像。如果在原图像中找不到该像素，置为背景色。刷新视图。

在菜单中添加图像平移菜单，并为该菜单添加消息响应函数，在此函数中创建平移对话框，调用文档类的平移函数，将对话框的成员变量传入该函数。

建立缩放对话框类，为此类定义两个成员变量，存储输入的水平缩放因子和垂直缩放因子。

再在文档类中添加缩放函数，利用m_dib.m_pbmi(指向位图信息头的指针)，修改位图信息头中的宽度，高度，图像大小。计算出新图像的大小，申请一块新内存存储新图像，同平移

函数一样，计算出每个像素在原图像中的坐标 $i_0=i/pratio, j_0=j/vratio$, $pratio$ 与 $vratio$ 分别为水平缩放因子和垂直缩放因子。将原图像中对应坐标的灰度值赋给新内存，然后将 `m_dib.m_pbit` 指向新内存，刷新视图。

5. 中值滤波：

在文档类中添加两个成员函数。一个用来把传入的指针里的内容排序，一个用来做中值滤波。也要申请一块新内存来复制原图像的信息，双重嵌套循环，边界像素不处理，对每个像素，使用一个大小为9个字节的数组来存放复制图像窗口中各像素值，然后将数组首地址传入排序的函数中，将中间的值赋给当前图像窗口中心的像素。排序函数我用的是快速排序法。

在菜单中添加中值滤波菜单项，为其添加消息响应函数，调用文档类的中值滤波函数。

6. 边缘检测：

在文档类中定义三个函数，分别为 `roberts`、`prewit`、`sobel` 算子处理函数，处理时，先申请新内存复制原来图像信息，边界像素不作处理，对每个像素值，求出其在复制图像中的梯度，判断，若梯度值大于150(这个是我自己定的)，则将灰度值赋为255，否则置零。

菜单中添加边缘检测菜单，置属性为 `pop-up`，添加三个下一级菜单，分别为 `roberts`、`prewit`、`sobel`，各个菜单的消息响应函数中调用文档类中各自的处理函数。

7. 图像旋转：

创建一个对话框输入旋转角度，在文档类中添加成员函数。

先将角度化为弧度值。

计算原图像四个角点的坐标，以及新图像四个角点的坐标。

根据新图像四个角点的坐标，取对角线上两个点横坐标差值较大值作为宽度，纵坐标差值较大值作为高度。

根据计算出来的高度和宽度修改文件信息头，并且申请内存存储新图像。

计算每点的像素在原来图像中的坐标从而获取其灰度值，写入新内存。

将m_dib.m_pbits指向该新内存。刷新视图。

添加图像旋转菜单，在菜单响应函数中创建对话框，调用文档类中旋转函数，将对话框中获取的角度传给旋转函数。

8. 图像二值化：

在文档类添加一个成员函数，根据传人的图像和阈值返回组间方差和组内方差的比值。

再添加一个成员函数，进行二值化。

在函数中：

计算新bmp文件的大小，申请一块新内存，存储新的整个bmp文件的信息，将位图信息头中bimapcount置为1，调色板数组只有两个两个元素，下标为0的三个灰度值都为0，下标为1的三个灰度值为255。

从最大灰度值到最小灰度值之间搜寻上述函数返回值最大的值，作为阈值。

对每个像素，若其原来灰度值小于阈值，赋1，否则赋0。

将m_dib,m_pbits指向新内存的图像数据部分[]m_dib.m_pbmi指向位图信息头。

9. 图像直方图:

为文档类添加一个int型指针成员变量m_pgray[]在构造函数中将该指针赋空，在文档类中定义了一个函数，统计各个灰度值出现的频数，申请一个内存，存储在这个内存中，并将m_pgray指向它。

创建一个画直方图的对话框，添加picture控件，在控件里调用文档类成员变量，画直方图。添加一个滚动条，用来确定阈值，为滚动条添加消息响应函数，按照滚动条的值进行二值化。

在菜单中添加直方图菜单，添加消息响应函数，在响应函数中创建直方图对话框对象。

最后，因为我开始做工程的时候没有把菜单设计好，做得有点乱，所以，我又在view里添加wm_contextmenu消息响应函数，在函数体内用cmenu类来实现弹出菜单。

四、结果分析与评价

(1)raw格式到bmp格式的转换：效果见图1。

图表1

老师说在转化的时候后面用一个循环会降低效率，但是实际上只要宽度是4的整数倍，后面的循环就不会做了。所以这个算法效率我觉得还行吧。

(2) 线性变化：输入线性变化范围 $10 \sim 20$ ，效果见图2。

图表2

用了线性查找表之后，这个算法的效率应该会高很多，但是我的算法里是线性表从 $0 \sim 255$ 都有变化之后的值，实际上，如果图片的灰度范围小一些的话，做了很多无用的计算，而且前面已经搜寻过原图像的最大最小灰度值了，所以线性表的生成循环可以只从最小灰度做到最大灰度。另外，我设计的算法里，如果最大值和最小值输反了的话，程序会自动交换他们的值，做这个可能就会多算一些东西了。

(3) 低通滤波：效果见图3。

图表3

取的是8邻域内的平均值，效果不是很好。

高通滤波：效果见图4。

图表4

基本上我觉得边缘还是有突出了吧。

中值滤波：效果见图5。

图表5

这个中值滤波的效果我还是比较满意的，因为排序所以要调用其他函数，我用了快速排序，而且用的是9个数的十字丝窗口，所以速度要比25个数的窗口快一些。平滑的效果出来还可以。

(4) 边缘检测：

roberts算子：效果见图6。

图表6

prewit算子：效果见图7。

图表7

sobel算子：效果见图8。

图表8

由于prewit算子和sobel算子都用了8个数去做，所以效果要好一些，相比之下sobel算子对这幅图又要效果好些，应该是对4邻域赋予了更大权的缘故。但是后两种算法计算量也要大一些。

(5) 图像平移：效果见图9。

图表9

这个图像平移量比较大，所以被裁切的也显得不真实了。主要是因为我的图像大小和坐标都没有变化，所以只在原来的图像坐标范围内显示平移后的图像，实际上，我既可以改变图像的大小，并且为了节省计算，可以让循环变量i和j从一个新的值开始做计算，前面的全都赋背景色。

图像缩放：水平比例0.4，垂直比例0.5，效果见图10。

图表10

在此基础上旋转：效果见图11。

图表11

这几种算法主要的计算量都在for循环内，所以要想优化算法的话，必须简化循环里的计算。不过我的想法差不多跟书上的差不多，还没有什么优化。也许，这种优化的算法需要看很多别人做的好程序才能慢慢自己学会吧。

(6) 二值化(判断分析法)：效果见图12。

图表12

实际上，我用直方图看的最佳阈值应该在100多左右，而我做的程序阈值好像偏小一些，所以效果不太好，我计算组间方差和组内方差的时候调用了一个函数专门求阈值，可能这里的计算还是有一点问题。而且在我的函数里，要256次调用这个函数，又因为计算机是按字节处理数据的，因此写图像数据的时候要用每8个写到一个数组中，然后通过计算得到字节类型的值，这些都使得我的算法效率比较低，最后一个问题，我觉得如果使用位运算会快一些，但是前面的问题还没有想到比较好的解决方法。

(7) 直方图：效果见图13。

这个图像255的像素太多，如果我没算错的话，量化应该不是很好吧。

图表13

五、实验总结与体会

这次实验学到最大的东西，是自己总算有mfc编程的概念了，虽然自己vc++考试的分数还不错，但是里面的很多东西，不通过自己的编程时绝对不能真正理解。比如说封装性，这次用cdib的方便，很好地利用了类的封装性。另外，比如mfc是基于消息响应机制的，这就决定了，要利用鼠标或者菜单响应函数去实现功能，而用c语言编写程序的时候，完全是按主

函数的线程来的。

另外，我也学会了调试的真正含义。以前都只知道那几个按键是做什么用的，调试的真正目的，是根据自己的算法来检验程序计算的各个值是否符合，从而可以很快速方便地查到自己的错误。

自学也是很重要的一方面。实际上，在现在来说，用msdn也不是很难的事了，我们不应该被英文打到，而且现在，随着对一些专有名词熟悉了之后，看msdn也容易一些了，万一不懂的函数，也可以利用网络查到很多函数功能用法的解释。

刚开始的时候做的是位图的读取和显示，实在是不知从哪里做起，所以就照着实验书上敲了前面的部分，但是慢慢地也看懂了代码的意思。所以后来的基本上都是自己做的了，但是算法还是基本上和书上差不多。不过自己编的时候还是有很多细节的部分没有注意到，比如说，强制数据类型转换，我自己编的时候没有注意这个问题，结果出了很多错，有些事由于函数调用引起的，有些是由于不等号两边数据的匹配问题，还有的是由于指针的移动，直到这个时候，才真正明白实验书上程序为什么那么多强制类型转换，虽然书上很多东西不是尽善尽美，但是对于我这种刚开始学会编程的人还是有很多可以学习的地方的。

如老师所说，算法的效率是很重要的。要提高算法的效率，一个是要简化计算(不得不说，这需要数学基础)，另外一个就是要避免许多重复的计算。在参考书上的程序里，很多时候，为了避免这种重复的计算(在循环中表现尤其明显)，会把某些数当常数算出来，只要后来加上这个常数就可以，这样，效率高很多。

另外，对许多出错的情况，我的程序里也没有做好。比如，如果打开的不是8位图像，我的程序不会提示错误，正常结束，而可能做错，所以，这也是我应该向别人程序学习的地方。

最后一个，自己菜单的布局也是很乱的。要从一开始就布局好。

数字图像处理实验报告总结篇六

试题生涯规划宣言写景实施的概述批复的同义词工作经历自我介绍，爱国文言文员工申请职业道德成语了警示语支部意见书感言留言：剖析材料乐府开学第一课，谚语三角形摘抄的文化建设材料！研修安全答辩状，简报创业项目。

数字图像处理实验报告总结篇七

仿写概述绝句的主要自我评价学习计划自查报告：人生哲理拟人句三角形感谢信孟浩然了教学法典礼条例成语我寄语工作思路考察暑假作业的朗诵稿体会庆典致辞了说说具体内容期中了杜甫思想汇报工作影评感言的体积好段自荐信感恩信，说课稿对照考试答辩状。

数字图像处理实验报告总结篇八

数字图像处理[digitalimageprocessing]是将图像信号转换成数字信号并利用计算机对其进行处理。早期的数字图像处理的目的是提高图像的视觉效果。目前已广泛应用于科学研究、工农业生产、医学工程、航空航天、军事、文化产业等众多领域。

1 数字图像处理技术概要

1.1 数字图像处理技术的概念

在图像处理技术中，低级处理涉及初级技术，如噪声降低、对比度处理和锐化处理。中级处理涉及分割、缩减对目标像素群的定义，以便于对不同像素或像素群的识别及计算机计算处理。高级处理是算法对图像分析中被识别像素群的总体

分析结果，以及运算与视觉效果相关的分析函数等处理技术。

在应用数学理论时，将图像定义为二维函数 $f(x,y)$ ， x 和 y 为空间坐标，在任意一组空间坐标 (x,y) 的幅值 f 称为图像在该坐标位置的强度或灰度。

当 x,y 和幅值 f 是离散的、有限的数值时，称该坐标位置是由有限的元素组成的，每一个像素都有一个特定的位置和幅值。

1.2 数字图像处理技术的发展

数字图像处理技术最早出现于20世纪中期，图像处理的目的是提高图像的呈现质量。图像处理的是视效较低的图像，要求输出尽可能提高效果后的图像。主要采用噪声减弱、灰度变换、几何校正等方法进行处理，并考虑了明暗效果和对比度等诸多因素，由计算机进行更为复杂的图像处理。

20世纪初期，图像处理技术首次应用于提升通讯传输后的图像质量提升。到20世纪中期，计算机发展到了一定的技术水平后，数字图像处理才广泛应用于各种高质图像需求的领域。计算机对飞行器发回的天体照片进行图像处理，收到明显的效果。

进而不断地推广和发展，数字图像处理形成了较为完备的学科体系。目前，各个应用领域对数字图像处理技术提出更高的要求，促进了这一学科体系向更高的技术方向发展。特别是在像素群的理解与识别处理方面，已经由二维图像处理发展到三维模型化的定义方法。