照片管理与编辑系统需求分析报告

C语言课程设计



专业班级： 自动化1802

小组成员： 王子毅 学号U201814426

牛宝健 学号U201814418

指导老师： 周纯洁、何顶新、彭刚、周凯波、桑农、

左峥嵘、高常鑫、汪国有、陈忠

上交时间：2019年9月29日

[引言 4](#_Toc20078522)

[1.1 编写目的 4](#_Toc20078523)

[1.2 背景 4](#_Toc20078524)

[1.3 参考资料 4](#_Toc20078525)

[任务概述 5](#_Toc20078526)

[2.1 目标 5](#_Toc20078527)

[2.2 用户的特点 5](#_Toc20078528)

[需求规定 6](#_Toc20078529)

[3.1 照片管理模块 6](#_Toc20078530)

[3.2 照片编辑模块 7](#_Toc20078531)

[1. 画笔 7](#_Toc20078532)

[2. 粗细 7](#_Toc20078533)

[3. 裁剪 8](#_Toc20078534)

[4. 选色 8](#_Toc20078535)

[3.3 照片修饰模块 9](#_Toc20078536)

[1. 放大 9](#_Toc20078537)

[2. 缩小 9](#_Toc20078538)

[3. 放缩模式 9](#_Toc20078539)

[4. 旋转 9](#_Toc20078540)

[5. 镜像 10](#_Toc20078541)

[运行环境规定 11](#_Toc20078542)

[4.1 设备 11](#_Toc20078543)

[4.2 支持软件 11](#_Toc20078544)

[4.3 接口 11](#_Toc20078545)

[4.4 控制 12](#_Toc20078546)

[系统设计 13](#_Toc20078547)

[5.1 功能需求描述 13](#_Toc20078548)

[5.1.1 照片管理模块 13](#_Toc20078549)

[1. 照片打开窗口 13](#_Toc20078550)

[2. 照片保存窗口 13](#_Toc20078551)

[3. 图片新建窗口 14](#_Toc20078552)

[4. 图库窗口 15](#_Toc20078553)

[5.1.2 照片编辑模块 16](#_Toc20078554)

[1. 画笔 16](#_Toc20078555)

[2. 裁剪 16](#_Toc20078556)

[3. 粗细 17](#_Toc20078557)

[4. 选色 18](#_Toc20078558)

[5.1.3 照片修饰模块 19](#_Toc20078559)

[1. 放大 19](#_Toc20078560)

[2. 缩小 19](#_Toc20078561)

[3. 放缩模式 19](#_Toc20078562)

[4. 旋转 20](#_Toc20078563)

[5. 镜像 20](#_Toc20078564)

[5.2 界面初步设计 21](#_Toc20078565)

[·5.2.1初始界面 21](#_Toc20078566)

[5.2.2打窗口开 21](#_Toc20078567)

[5.2.3保存窗口 22](#_Toc20078568)

[5.2.4新建窗口 22](#_Toc20078569)

[5.2.5图库窗口 23](#_Toc20078570)

[5.2.6粗细窗口 23](#_Toc20078571)

[5.2.7取色窗口 24](#_Toc20078572)

[5.2.8模式窗口 24](#_Toc20078573)

[5.2.9警告窗口 25](#_Toc20078574)

[5.2.10效果展示 25](#_Toc20078575)

[5.3 软件结构 26](#_Toc20078576)

[1. 照片管理 26](#_Toc20078577)

[2. 属性设置 29](#_Toc20078578)

[3. 图像编辑 30](#_Toc20078579)

[4. 图像修饰 32](#_Toc20078580)

[组员分工及时间安排 34](#_Toc20078581)

[6.1 组员分工 34](#_Toc20078582)

[6.1 时间安排 34](#_Toc20078583)

## 引言

#### 1.1 编写目的

通过一定量的有关 “照片管理与编辑系统”的需求者和使用者的调查与分析，以及通过使用已有的照片管理与编辑系统和网上资料查询，在对项目可行性和功能的详细分析和纠正的基础上，特提出了这份软件需求分析和系统设计报告。

此份需求分析与系统设计报告对“照片管理与编辑系统”项目做了全面的用户需求分析，明确此所要开发的软件应具有的功能、性能与界面，设计了程序流程，使系统分析人员及软件开发人员能清楚地了解用户的需求，并在此基础上进一步提出系统设计报告和完成后续设计与开发工作。本报告的预期读者为项目委托部门、业务或需求分析人员、测试人员。

#### 1.2 背景

当今时代是信息化时代，而信息的数字化也越来越为研究人员所重视。数码照片是数字化的摄影作品，通常指采用数码相机进行创作的摄影作品。数字图像处理(Digital Image Processing)是通过计算机对图像进行去除噪声、增强、复原、分割、提取特征等处理的方法和技术。照片管理与编辑系统主要处理以像素所构成的数字图像，使用众多的编修与修饰工具，可以有效地进行图片编辑工作。

#### 1.3 参考资料

* 王士元. C高级实用程序设计. 北京: 清华大学出版社. 1996
* 周纯杰，刘正林等. 标准C语言程序及应用. 武汉: 华中科技大学出版社. 2008
* VESA BIOS EXTENSION (VBE) Core Function Standard Version: 2.0 Document Revision: 1.1 Ratification Date: November 18, 1994

## 任务概述

#### 2.1 目标

本产品可实现“照片管理与编辑系统”的基本功能。产品用户可在最开始的界面进打开或新建照片。普通用户使用该软件可进行的操作：打开照片、保存照片、新建照片、调用图库、编辑照片、裁剪照片、绘制图形、调节画笔颜色、调节画笔粗细、调节图形粗细、旋转照片、翻转照片、放大照片、缩小照片。

#### 2.2 用户的特点

本系统操作简单界面美观逻辑清晰。面向的用户是广大的照片编辑者，只要具备计算机的基本使用技能和照片编辑的思路就可以使用该系统，用户进入该系统时，按照指示窗口进行操作，即可完成照片管理、修饰、编辑的全过程。对于维护人员，需要对C语言和数据结构有一定的了解，以便于在软件出错时及时维护软件，并可以根据需要增添相关的功能。预计该系统投入市场后反应较好，会得到消费者以及项目开发人员的肯定与支持。

## 需求规定

#### 3.1 照片管理模块

1. **文件打开**

输入文件名称，打开文件。

如果文件名称输入正确并且文件类型符合要求，则正常打开图像；否则，弹出报错窗口。

1. **文件保存**

输入文件名称，保存文件。

如果已有图像打开并且输入的文件名称未超出字符长度限制，则正常保存图像；否则，弹出报错窗口。

1. **文件新建**

输入尺寸，选择颜色，新建该尺寸、该色彩的图像。

1. 输入新建图像的尺寸：图像的长度以及图像的宽度（单位：像素）。
2. 如果没有输入尺寸，则选择默认尺寸（400像素×300像素）。
3. 如果尺寸不符合要求，则弹出警告；
4. 选择新建图像的填充色。
5. 如果没有选择颜色，则选择默认颜色（灰白色）。
6. **图库**

点击文件，查看文件信息，打开或删除文件。

1. 显示基本信息：文件名称、文件类型、修改时间、图像长度、图像宽度。
2. 显示照片缩略图。
3. 点击“打开”按钮，直接打开图片。
4. 点击“删除”按钮，直接删除文件。

#### 照片编辑及修饰模块

###### 画笔

通过画笔对图像进行修饰。

1. 在图像有效区域，可以绘制。
2. 在图像区域外，不能绘制。
3. 可以调用选色、粗细调整功能修改画笔的属性。

###### 裁剪

裁剪图像。

1. 检测用户是否打开图像，如果没有，则提醒用户。
2. 如果裁剪后的图像太小，则提醒用户是否要裁剪。

###### 放大

点击“放大”按钮，可对图像放大。

如果图像过大，或已经放到最大程度，则不进行图像缩小处理。

###### 缩小

点击“缩小”按钮，可对图像缩小。

如果图像过小，或已经缩小到最小程度，则不进行图像缩小处理。

###### 旋转

点击“旋转”按钮，将图像逆时针旋转90度。

如果旋转后的图像超出工作区域，则旋转操作无效。

###### 镜像

点击“镜像”按钮，将图像沿中心竖直线翻转。

#### 工具设置模块

###### 粗细大小

设置画笔的大小和图形的粗细。

1. 画笔的大小和图形的粗细具有最大值（分别为8，5）。
2. 画笔的大小和图形的粗细具有默认值（分别为5，1）。

###### 选色

通过调色板设置画笔、图形等工具的颜色。

1. 调色板分为两个主要部件：色相-饱和度面板，亮度面板。
2. 调色板会以RGB数值的形式显示用户当前选择的颜色信息。

例如：R:125 G:155 B:176。

1. 调色板可以选择的颜色多达16,777,216‬ 种颜色（256\*256\*256）。

###### 放缩模式

设置图像的放缩模式：双线性插值法、临近点插值法（默认）。

## 运行环境规定

#### 4.1 设备

本系统是对“图片管理与编辑系统”基本模拟，因此，对此系统的运行环境将是在计算机上面进行，即在电脑上模拟实现，系统运行的设备需求如下:

1. 一台 586 以上的微机及兼容
2. 内存 16MB 及其以上
3. 彩色显示器一台

#### 4.2 支持软件

根据“图片管理与编辑系统”设计环境要求，即在 DOS 环境下，用 C 语言编译实现，使用 TC 或 BC 开发软件，因此，此模拟系统可在绝大多数现形计算机系统上运行，包括最常见的 Windows XP 操作系统以及 Windows 7 32 位机上运行，但由于兼容性问题，此“证券量化投资系统”在windows 7 64 位操作系统的计算机上运行可能出现意想不到的问题或根本不能运行，因而，不推荐在包含 64 位操作系统的计算机上测试本系统。

#### 4.3 接口

1. 用户接口:本系统采用一般性架构，所有界面使用简易风格。
2. 硬件接口:无特殊要求。
3. 软件接口:无特殊要求。
4. 通信接口:无特殊要求。

#### 4.4 控制

本系统主要通过鼠标直接进行操控，部分内容可以通过键盘操作。用户将鼠标移至需要操作的功能区点击，进行相应的操作，操作进行完后，点击相应位置退出系统。控制信号来源于用户鼠标的位置，以及用户使用鼠标进行的操作。

## 系统设计

#### 5.1 功能需求详细描述

###### 5.1.1 照片管理模块

1. 打开
2. 调用：

点击“打开”图标，弹出“打开”窗口。

1. 设置快捷键及其他键：
2. ESC键关闭窗口。
3. Enter键打开图片。
4. Backspace键删除字符。
5. 打开图片：
6. 如果不存在指定文件则弹出警告窗口。
7. 如果指定文件类型不符弹出警告窗口。
8. 正常情况关闭“打开”窗口并打开图片。
9. 关闭：

点击“关闭”图标关闭窗口。

1. 保存
2. 调用：

①如果工作区域已有图像打开，弹出“保存”窗口。

②如果工作区域没有图像打开，弹出“未图片打开”警告。

1. 设置快捷键：
2. ESC键关闭窗口。
3. Enter键保存图片。
4. Backspace键删除字符。
5. 保存图片：

点击“确认”按钮，保存图片。

1. 关闭：

点击“关闭”图标，关闭窗口。

1. 新建
2. 调用：

点击“新建”图标，弹出“新建”窗口。

1. 设置快捷键：
2. ESC键关闭窗口。
3. Enter键新建图像。
4. Backspace键删除数字。
5. Tab键切换输入框。（宽度输入框和高度输入框）。
6. 输入尺寸：

点击“宽度”输入框，宽度输入框被聚焦。

点击“高度”输入框，高度输入框被聚焦。

1. 选择颜色：

点击“选色”按钮，弹出调色板窗口，选择颜色。

1. 新建图片：
2. 如果输入的宽度不符合要求，弹出“宽度超出或不足”警告窗口。
3. 如果输入的高度不符合要求，弹出“高度超出或不足”警告窗口。
4. 如果未进行选色操作，则选择默认填充色（灰白色）。
5. 如果未输入尺寸，则使用默认尺寸（400像素\*300像素）。
6. 关闭：

点击“关闭”图标，关闭窗口。

1. 图库
2. 调用：

点击“图库”图标，弹出“照片管理”窗口。

1. 显示文件列表：

显示文件夹中的部分文件。（如果文件数量不多，则显示全部）。

1. 加载文件信息：

点击文件列表中的文件，显示文件信息：

1. 文件名称
2. 文件类型
3. 文件大小（单位：KB）
4. 修改日期（包括：年、月、日、时、分、秒）
5. 图像尺寸（包括：图像宽度、图像高度，单位：像素）。
6. 显示照片缩略图：

点击文件列表中的文件，显示图片的缩略图。

1. 翻动文件列表：

点击“上翻”按钮向上翻动文件列表。

点击“下翻”按钮向下翻动文件列表。

1. 打开图片：

点击“打开”按钮打开图片。

1. 删除图片：

点击“删除”按钮删除文件。

1. 关闭：

点击“关闭”图标，关闭窗口。

###### 5.1.2 照片编辑及修饰模块

1. 画笔
2. 调用：

点击“画笔”图标，启动画笔功能。

如果图像未打开，画笔功能无法启动。

1. 绘制：

在图像区域内点击，开始绘制。

如果不在图像有效区域内点击，绘制无效。

1. 退出：

再次点击“画笔”图标退出画笔功能。

如果直接点击其他功能按钮，将会直接切换至对应功能。

1. 裁剪
2. 点击“裁剪”图标，启动“裁剪”功能。

如果图像未打开，裁剪功能无法启动。

1. 在图像区域内按下鼠标左键记录坐标(X1,Y1)。

如果不在图像有效区域内点击鼠标，记录无效。

1. 拖动鼠标移至图像内另一点。
2. 松开鼠标左键，记录坐标（X2，Y2）。

如果已经移除鼠标有效区域，则记录无效。

1. 对图像进行裁剪，保留两坐标以内的内容，其余内容删去。
2. 再次点击“裁剪”图标退出画笔功能。
3. 如果直接点击其他功能按钮，将会直接切换至对应功能。
4. 放大
5. 调用：

点击“放大”图标，进行图像放大处理。

1. 放大：

如果工作界面没有图像打开，则操作无效，并提醒用户。

如果图像放大后超出上限，则操作无效，并提醒用户。

正常情况下，以1.25倍率放大原图像。

1. 缩小
2. 调用：

点击“缩小”图标，进行图像缩小处理。

1. 缩小：

如果工作区域没有图像打开，则操作无效。

如果图像缩小后超出下限，则操作无效。

正常情况下，以0.80倍率缩小原有图像。

1. 旋转
2. 调用：

点击“选择”图标，进行图像旋转处理。

1. 旋转：

如果工作区域没有图像打开，则操作无效。

如果旋转后的图像，宽度或高度超出限制范围，则操作无效。

正常情况下，图像进行逆时针旋转90度。

1. 镜像
2. 调用：

点击“镜像”图标，进行图像镜像处理。

1. 镜像：

如果工作区域没有图像打开，则操作无效。

理论上，不存在镜像后图像长宽超出限制范围的情况。

正常情况下，图像进行水平镜像。

###### 5.1.2 工具设置模块

1. 粗细大小
2. 调用：

点击“粗细”图标，弹出“粗细调整”窗口。

1. 调节画笔大小：

点击“画笔粗细”后的“+”，画笔大小增加。

点击“画笔粗细”后的“-”，画笔大小减少。

默认画笔大小为5，最大值为8，最小值为1。

1. 调节图形粗细：

点击“线条粗细”后的“+”，线条宽度增加。

点击“线条粗细”后的“-”，线条宽度减少。

默认线条粗细为1，最大值为5，最小值为1。

1. 确认：

点击“确认”按钮，保存本次修改后的粗细值，并退出窗口。

1. 关闭：

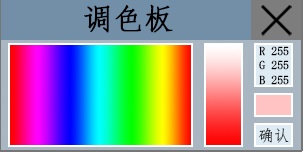
点击“取消”按钮，不保存修改值，关闭窗口。

点击“关闭”图标，不保存修改值，关闭窗口。。

1. 选色
2. 调用：

点击“选色”图标，弹出“选色”窗口。

1. 选色：

选色窗口包含两个面板（色相-饱和度面板、亮度面板），两个信息输出框（RGB色彩信息框、呈色框）组成。

1. 色相-饱和度面板：横向为色相轴（0~180），纵向为饱和度轴（0~100）。鼠标点击色相-饱和度面板，会触发程序捕获相应的色相（H）、饱和度（S）数值，并将HSL变量（亮度L默认为0.5）转换为RGB变量。
2. 亮度面板：纵向为亮度轴（0~100）。鼠标点击亮度面板，会触发程序捕获相应的亮度（L），并将HSL变量（H、S默认为0）转换为RGB变量。
3. RGB色彩信息框：在每次色彩空间转换完成后，RGB变量将会格式化输出到色彩信息框内，以便用户更充分地了解色彩信息。
4. 呈色框：在每次色彩空间转换完成后，呈色框将呈现出所选颜色，以便用户更直接地获得所选颜色信息。
5. 确认：

点击“确认”按钮，保存所选颜色，关闭窗口。

1. 关闭：

点击“关闭”图标，直接关闭窗口，不保存修改值，关闭窗口。

1. 放缩模式
2. 调用：

点击“设置”图标，弹出“模式”窗口。

默认情况下选择临近点插值法。

1. 确认：

点击“确认”按钮，保存选择的模式，并关闭窗口。

1. 关闭：

点击“取消”按钮，放弃选择的模式，并关闭窗口。

点击“关闭”图标，放弃选择的模式，并关闭窗口。

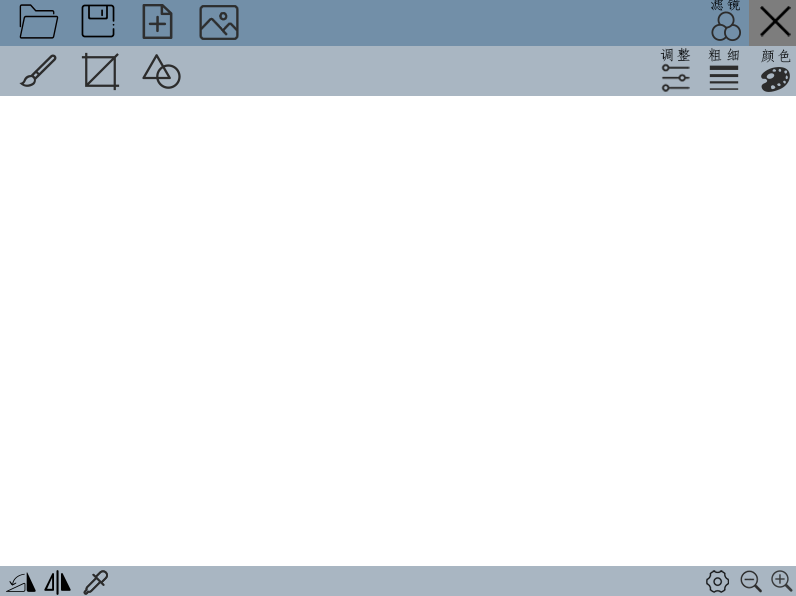
#### 5.2 界面设计

###### 欢迎界面



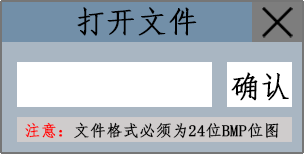
欢迎界面由标题和背景组成，标题使用汉字库制作，背景使用渐变函数制作（无贴图），界面内有动画显示。

###### 工作界面



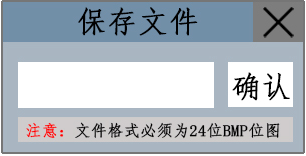
工作界面由各种工具以及中心工作区域组成。

###### 打窗口开



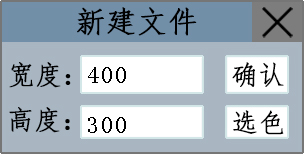
打开窗口由输入框、提示框、确认按钮组成。

###### 保存窗口



保存窗口由输入框、提示框、确认按钮组成。

###### 新建窗口



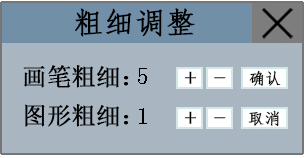
新建窗口由宽度输入框、高度输入框、确认按钮、选色按钮组成。

###### 图库窗口



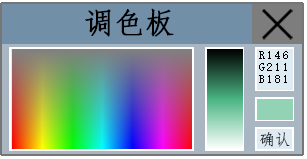
照片管理窗口由文件列表、信息框、缩略图、打开按钮、删除按钮组成。

###### 粗细调整窗口



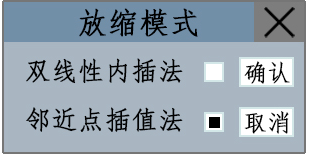
粗细调整窗口由画笔粗细调整按钮、图形粗细调整按钮、确认按钮、取消按钮组成。

###### 选色窗口



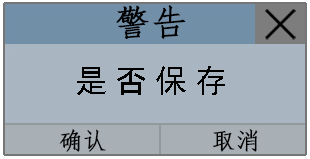
调色板窗口由色相-饱和度面板、亮度面板、RGB信息面板、呈色面板组成。

###### 放缩模式窗口



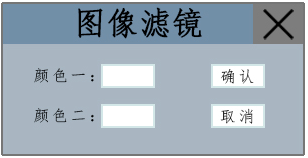
放缩模式窗口由模式选项部件、确认按钮、取消按钮组成。

###### 警告窗口

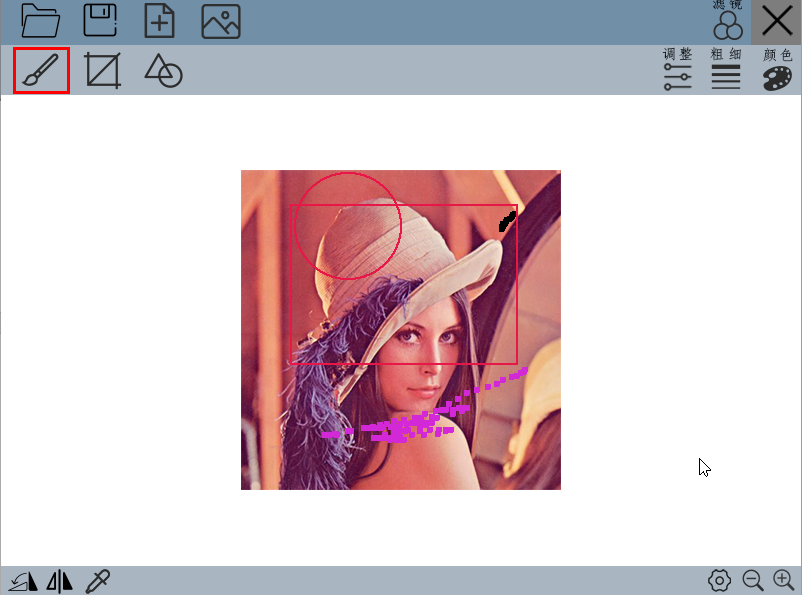


警告窗口由警告信息、确认按钮、取消按钮组成。

###### 调整窗口



###### 效果展示



通过点击不同的功能图标，可实现不同的功能。

#### 5.3 软件结构

用一览表及框图的形式说明系统的各个模块（各层模块、子程序、公用程序）的划分，扼要说明每个块的功能，分层次地给出各模块之间的控制与被控制关系。

###### 文件结构

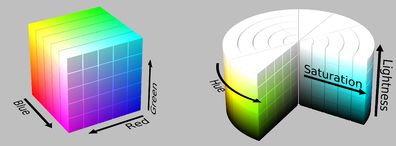
###### 函数库关系

###### 消息机制

###### 功能调用

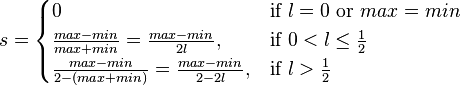
## 算法

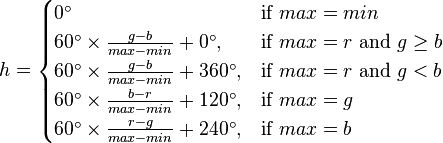
#### 色彩空间转换算法



图一： 左图： RGB模型 右图： HSL模型

###### 从 RGB 到 HSL 的转换





1. **int** RGB2HSL(RGB rgb, HSL\* hsl)
2. {
3. **double** rr, gg, bb;
4. **double** Max, Min;
5. **int**\* h = &hsl->h;
6. **double**\* s = &hsl->s;
7. **double**\* l = &hsl->l;
9. rr = (**double**)rgb.r / 255.0;
10. gg = (**double**)rgb.g / 255.0;
11. bb = (**double**)rgb.b / 255.0;
13. Max = rr;
14. **if** (Max < gg)
15. {
16. Max = gg;
17. }
18. **if** (Max < bb)
19. {
20. Max = bb;
21. }
23. Min = rr;
24. **if** (Min > gg)
25. {
26. Min = gg;
27. }
28. **if** (Min > bb)
29. {
30. Min = bb;
31. }
33. //计算亮度
34. \*l = (Max + Min) / 2.0;
36. //计算色相
37. **if** (Max == Min)
38. {
39. \*h = 0;
41. }
42. **else** **if** (Max == rr && gg >= bb)
43. {
44. \*h = 60 \* (gg - bb) / (Max - Min) + 0;
45. }
46. **else** **if** (Max == rr && gg < bb)
47. {
48. \*h = 60 \* (gg - bb) / (Max - Min) + 360;
49. }
50. **else** **if** (Max == gg)
51. {
52. \*h = 60 \* (bb - rr) / (Max - Min) + 120;
53. }
54. **else** **if** (Max == bb)
55. {
56. \*h = 60 \* (rr - gg) / (Max - Min) + 240;
57. }
58. **else**
59. {
60. **return** 0;
61. }

64. //计算饱和度
65. **if** (\*l == 0 || Max == Min)
66. {
67. \*s = 0;
68. }
69. **else** **if** (\*l > 0 && \*l <= 0.5)
70. {
71. \*s = (Max - Min) / (Max + Min);
72. }
73. **else** **if** (\*l > 0.5)
74. {
75. \*s = (Max - Min) / (2 - (Max + Min));
76. }
77. **else**
78. {
79. **return** 0;
80. }
82. **return** 1;
83. }

###### 从 HSL 到 RGB 的转换





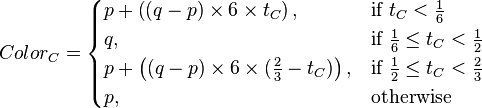










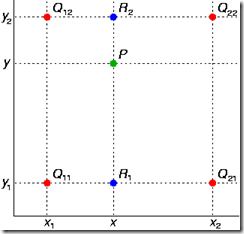




1. **int** HSL2RGB(RGB\* rgb, HSL hsl)
2. {
3. **double** tR, tG, tB;
4. **double** p, q;
5. **double** hk;
6. u8\* r = &rgb->r;
7. u8\* g = &rgb->g;
8. u8\* b = &rgb->b;
10. //如果S为0，则为灰度图
11. **if** (hsl.s == 0)
12. {
13. tR = hsl.l;
14. tG = hsl.l;
15. tB = hsl.l;
16. }
17. **else**
18. {
19. **if** (hsl.l < 0.5)
20. {
21. q = hsl.l \* (1.0 + hsl.s);
22. }
23. **else**
24. {
25. q = hsl.l + hsl.s - (hsl.l \* hsl.s);
26. }
27. p = 2.0 \* hsl.l - q;
28. hk = (**double**)hsl.h / 360.0;
29. tR = hk + 1.0 / 3.0;
30. tG = hk + 0;
31. tB = hk - 1.0 / 3.0;
32. tR = ToRGB(p, q, tR);
33. tG = ToRGB(p, q, tG);
34. tB = ToRGB(p, q, tB);
35. }
36. \*r = (**int**)(tR \* 255.0 + 0.5);
37. \*g = (**int**)(tG \* 255.0 + 0.5);
38. \*b = (**int**)(tB \* 255.0 + 0.5);
39. **return** 1;
40. }
42. **double** ToRGB(**double** p, **double** q, **double** tC)
43. {
44. **double** ColorC;
45. **if** (tC < 0)
46. {
47. tC = tC + 1.0;
48. }
49. **else** **if** (tC > 1)
50. {
51. tC = tC - 1.0;
52. }
53. **else**
54. {
55. ;
56. }
57. **if** (tC < (1.0 / 6.0))
58. {
59. ColorC = p + ((q - p) \* 6 \* tC);
60. }
61. **else** **if** (tC >= (1.0 / 6.0) && tC < 0.5)
62. {
63. ColorC = q;
64. }
65. **else** **if** (tC >= 0.5 && tC < (2.0 / 3.0))
66. {
67. ColorC = p + ((q - p) \* 6 \* (2.0 / 3.0 - tC));
68. }
69. **else**
70. {
71. ColorC = p;
72. }
74. **return** ColorC;
75. }

#### 图像放缩算法

###### 双线性内插法



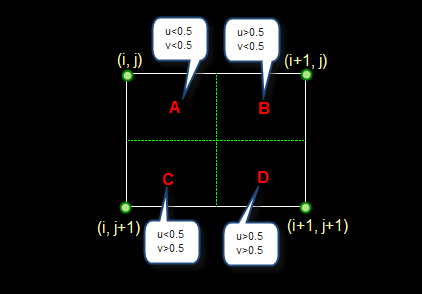
图二：双线性内插法



图三：双线性内插法效果

（双线性内插法代码见下）

###### 临近点插值法



图四：临界点插值法

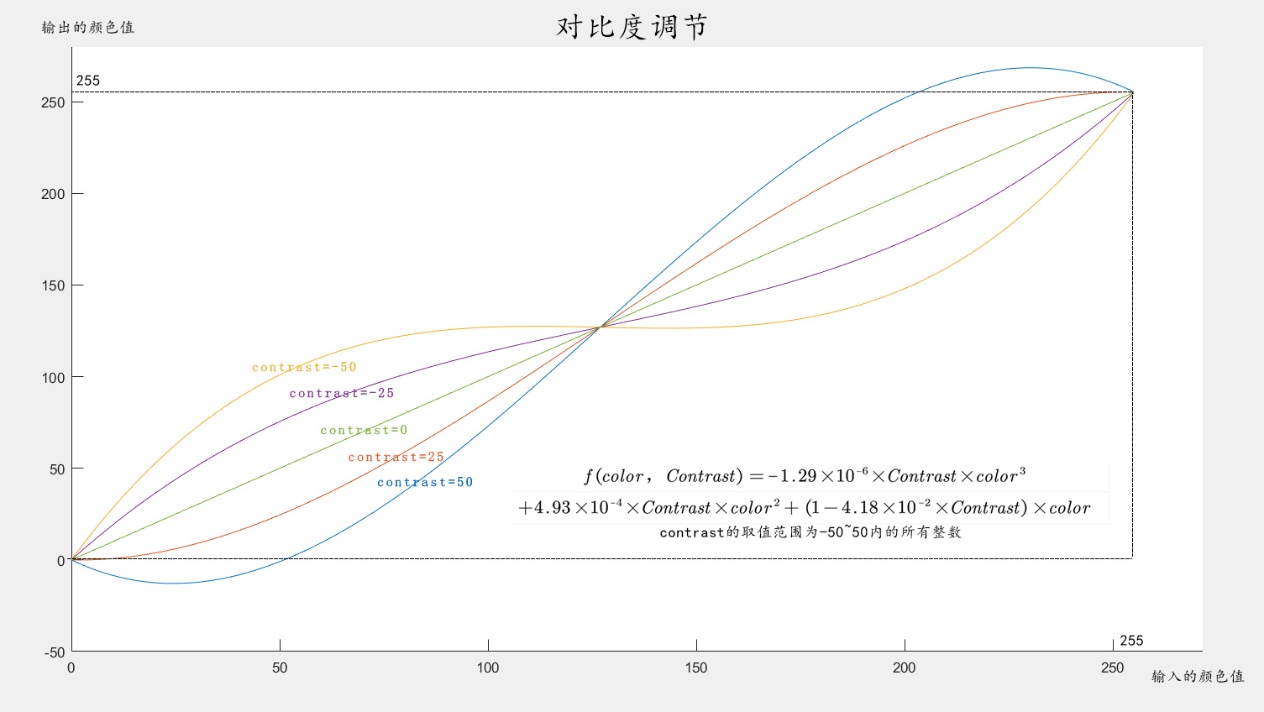


图五：临界点插值法效果图

1. u32 ImageInterpolat(**int** x1, **int** x2, **int** y1, **int** y2, **double** x, **double** y, u8 patton)
2. {
3. **int** i;
4. u32 color[4];
5. RGB rgb[4];
6. **double** r1,r2,r,g1,g2,g,b1,b2,b;
8. color[0] = GetPixel(x1, y1);
9. color[1] = GetPixel(x2, y1);
10. color[2] = GetPixel(x1, y2);
11. color[3] = GetPixel(x2, y2);
13. **for** (i = 0; i < 4; i++)
14. {
15. rgb[i].r = (u8)((color[i] & 0xff0000) >> 16);
16. rgb[i].g = (u8)((color[i] & 0x00ff00) >> 8);
17. rgb[i].b = (u8)((color[i] & 0x0000ff));
18. }
20. **if** (patton)
21. {
22. //双线性内插法
23. r1 = ((x2 - x) \* rgb[0].r + (x - x1) \* rgb[1].r) / (x2 - x1);
24. r2 = ((x2 - x) \* rgb[2].r + (x - x1) \* rgb[3].r) / (x2 - x1);
25. r = ((y2 - y) \* r1 + (y - y1) \* r2) / (y2 - y1);
26. g1 = ((x2 - x) \* rgb[0].g + (x - x1) \* rgb[1].g) / (x2 - x1);
27. g2 = ((x2 - x) \* rgb[2].g + (x - x1) \* rgb[3].g) / (x2 - x1);
28. g = ((y2 - y) \* g1 + (y - y1) \* g2) / (y2 - y1);
29. b1 = ((x2 - x) \* rgb[0].b + (x - x1) \* rgb[1].b) / (x2 - x1);
30. b2 = ((x2 - x) \* rgb[2].b + (x - x1) \* rgb[3].b) / (x2 - x1);
31. b = ((y2 - y) \* b1 + (y - y1) \* b2) / (y2 - y1);
32. }
34. **else**
35. {
36. //最邻近点插值法
37. **if** (x >= x1 && x <= x1 + 0.5 && y >= y1 && y <= y1 + 0.5)
38. {
39. r = rgb[0].r;
40. g = rgb[0].g;
41. b = rgb[0].b;
42. }
43. **else** **if** (x > x1 + 0.5 && x <= x2 && y >= y1 && y <= y1 + 0.5)
44. {
45. r = rgb[1].r;
46. g = rgb[1].g;
47. b = rgb[1].b;
48. }
49. **else** **if** (x >= x1 && x <= x1 + 0.5 && y > y1 + 0.5 && y <= y2)
50. {
51. r = rgb[2].r;
52. g = rgb[2].g;
53. b = rgb[2].b;
54. }
55. **else** **if** (x > x1 + 0.5 && x <= x2 && y > y1 + 0.5 && y <= y2)
56. {
57. r = rgb[3].r;
58. g = rgb[3].g;
59. b = rgb[3].b;
60. }
61. **else**
62. {
63. **return** 0;
64. }
65. }
67. **return** RGB2U32(r, g, b);
68. }

#### 图像调节算法

###### 对比度调节算法



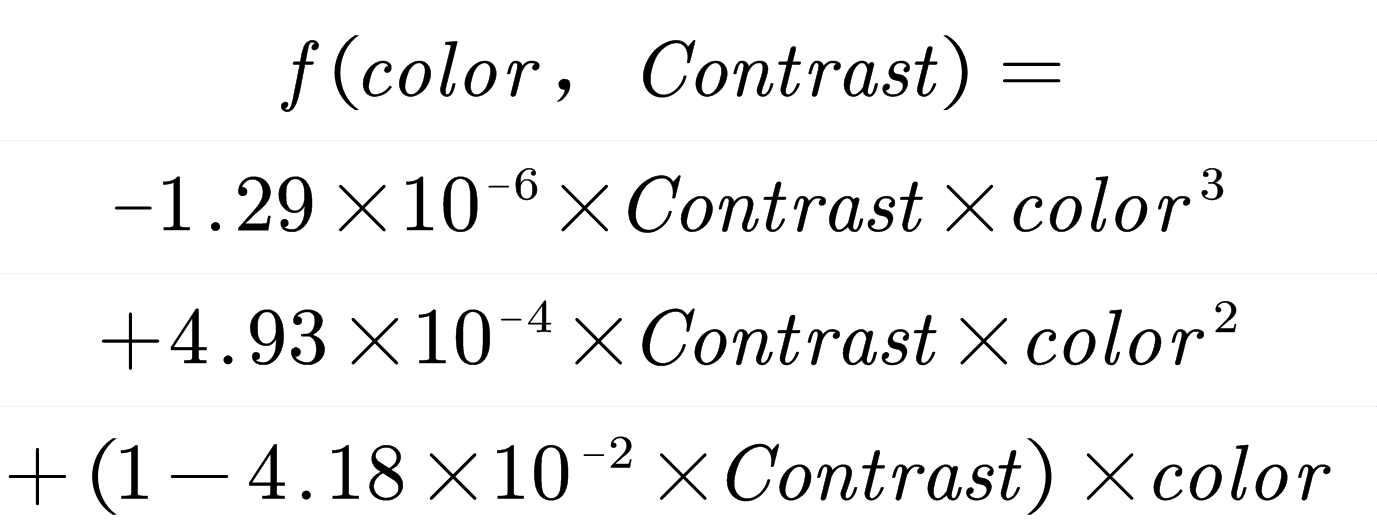
图六：对比度函数的图像

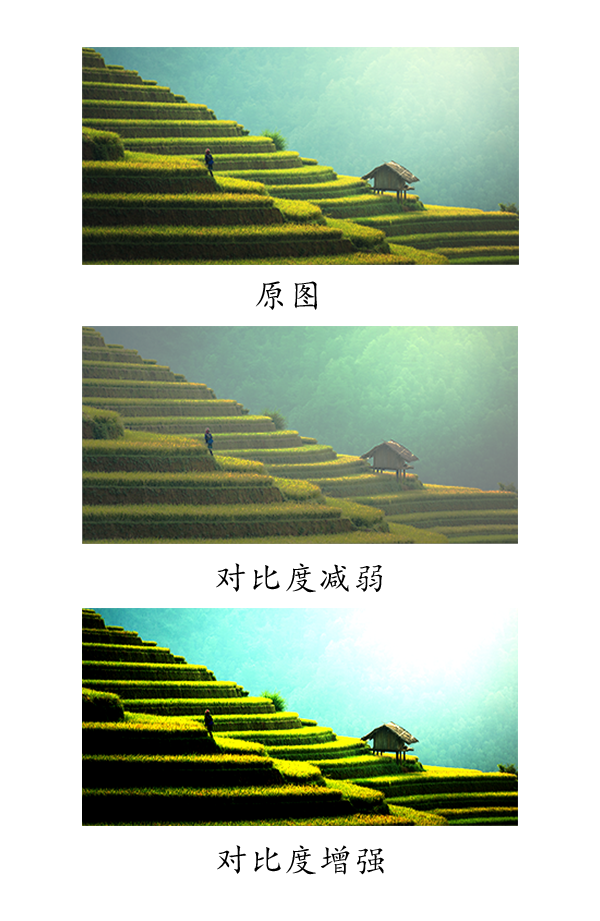
对比度实现的原理：

对比度调节函数通过模拟“S”型曲线，改变图像的色彩值：如果contrast < 0，则图像中小与阈值（127）的像素的值将增加，大于阈值的像素的值将减小；如果contrast > 0，则图像中小于阈值（127）的像素的值将减小，大于阈值的像素的值将增加；如果contrast = 0，则图像不变。

对比度调节的过程分为两步：第一步，判断当前图像的对比度是否为零，如果不为零，通过逆向变换得到对比度为零时的原图；如果为零，则直接进入第二步。第二步，通过正向变化得到调整后的图像。

对比度调节公式见下：

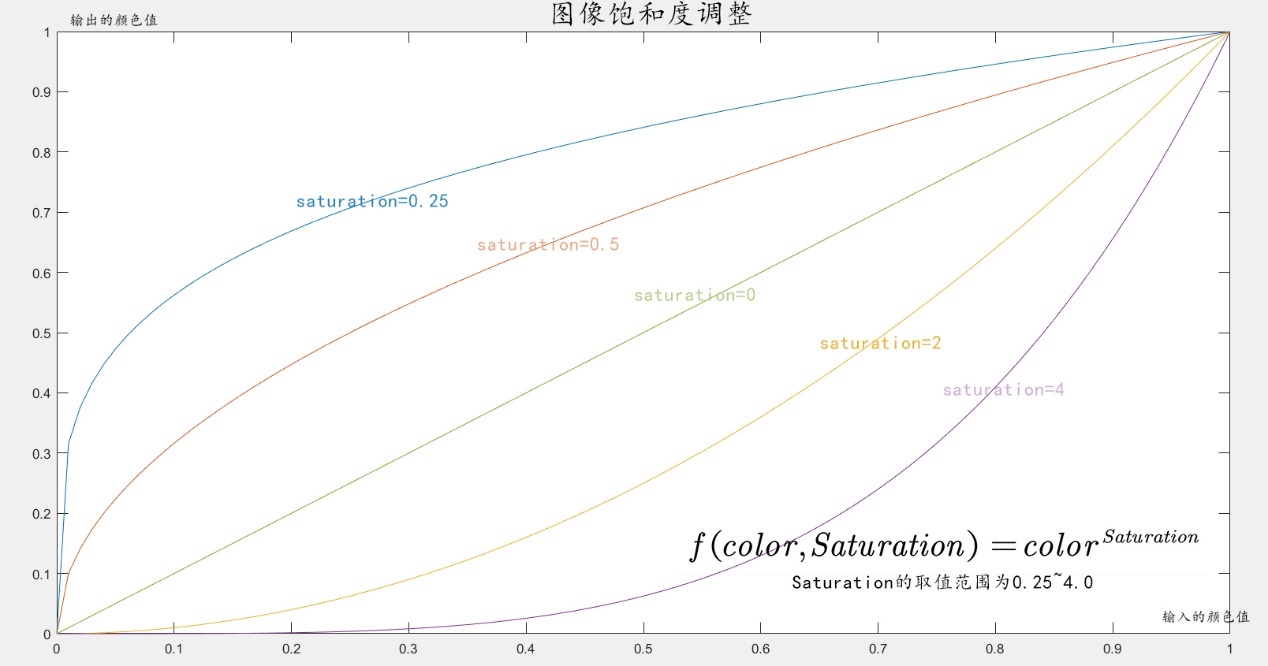




图七：对比度调节效果图

1. **int** ImageContrast(BMPATTR attr,**int** value)
2. {
3. **int** i, j;       //循环变量
4. u32 oldcolor;   //色彩变量
5. RGB rgb;        //色彩分量
7. **if** (value >= -50 && value <= 50)
8. {
9. **for** (i = attr.y1; i < attr.y2; i++)
10. {
11. **for** (j = attr.x1; j < attr.x2; j++)
12. {
13. oldcolor = GetPixel(j, i);
14. U32TRGB(&rgb, oldcolor);
15. //计算公式
16. rgb.r = ContrastCalc(rgb.r, value);
17. rgb.g = ContrastCalc(rgb.g, value);
18. rgb.b = ContrastCalc(rgb.b, value);
19. //类型转换
20. oldcolor = RGB2U32(rgb.r, rgb.g, rgb.b);
21. PutPixel(j, i, oldcolor);
22. }
23. }
24. }
25. **else**
26. {
27. //范围超出
28. **return** -1;
29. }
30. **return** 0;
31. }
33. u8 ContrastCalc(u8 color,**int** param)
34. {
35. **int** tcolor;
36. **if** (param > 50 || param < -50)
37. {
38. //如果输入调节参数超出限制范围
39. **return** -1;
40. }
41. tcolor = -1.29 \* param \* pow(color, 3) \* 1e-6 + 4.93 \* param \* pow(color, 2) \* 1e-4 + (1 - 4.18 \* param \* 1e-2) \* color;
42. //函数原型为：f(x)=a\*x^3+bx^2+cx，通过此函数模拟“S”型灰度曲线。
43. **if** (tcolor > 255)
44. {
45. **return** 255;
46. }
47. **else** **if** (tcolor < 0)
48. {
49. **return** 0;
50. }
51. **else**
52. {
53. **return** tcolor;
54. }
55. }

###### 饱和度调节算法



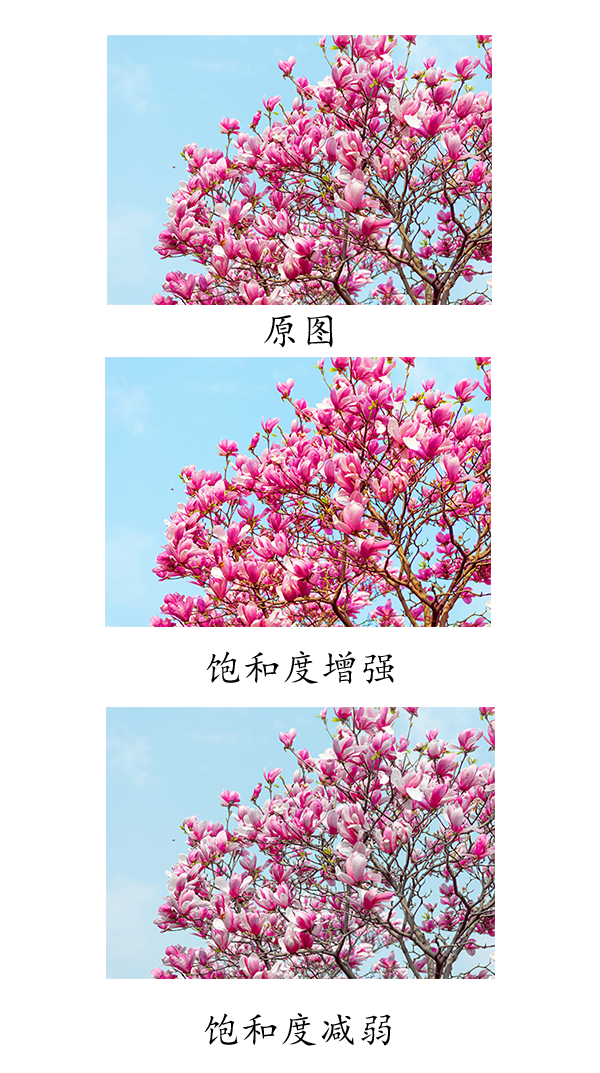
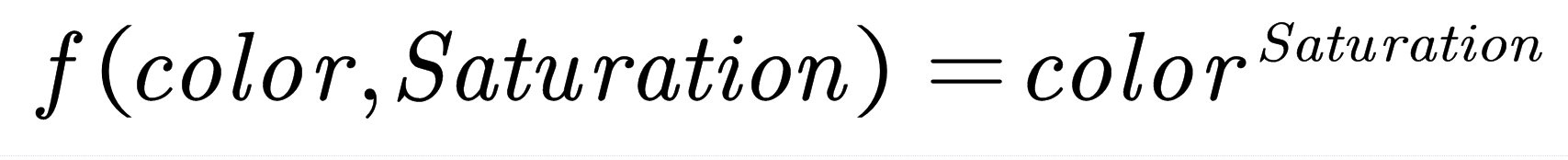
图八：饱和度调节函数曲线

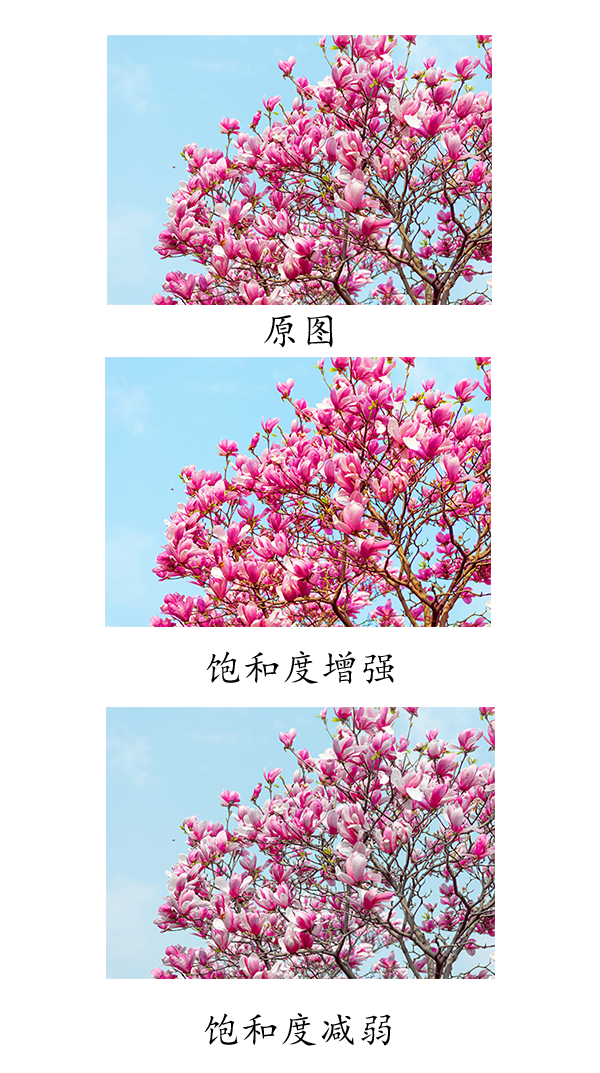
饱和度实现的原理：

饱和度调节函数通过模拟指数型曲线，改变图像的色彩值：如果saturation < 1，则图像中所有像素的饱和度将增加；如果saturation > 1，则图像中所有像素的饱和度将减小；如果saturation = 1，则图像不变。

饱和度调节的过程：第一步，判断当前图像的饱和度是否为1，如果不为1，则计算饱和度变化的倍率（本次的饱和度=选定的饱和度/上次的饱和度）；如果为1，则直接进入第二步。第二步，将计算后的倍率作为参数调节图像的饱和度。

饱和度调节公式见下：

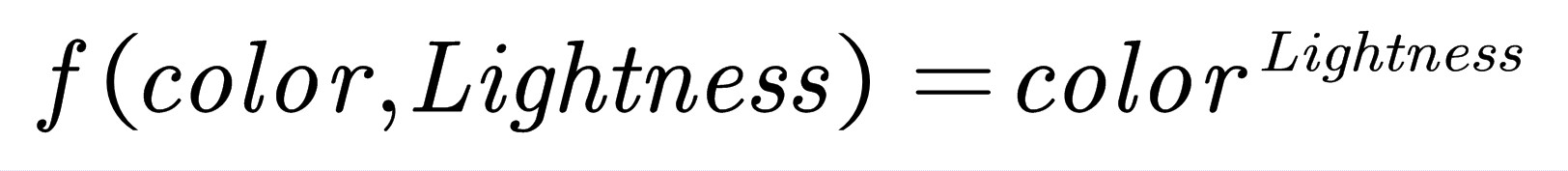


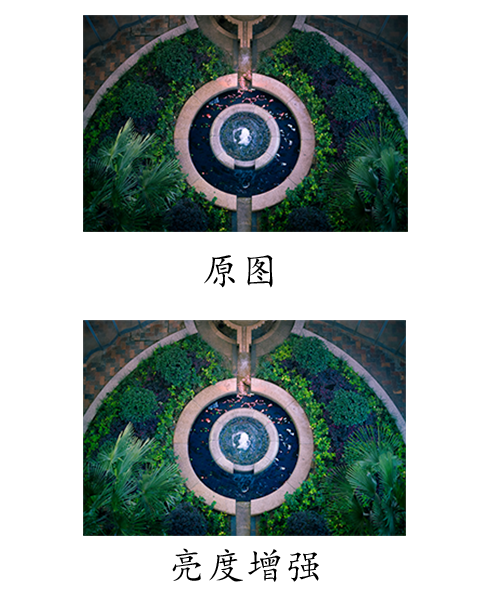


图九：饱和度调节效果图

###### 亮度调节算法

亮度调节与饱和度调节相似，公式为：





图十：亮度调节效果图

1. **int** ImageLightness(BMPATTR attr, **double** times)
2. {
3. **int** i, j;       //循环变量
4. u32 oldcolor;   //色彩变量
5. RGB rgb;        //色彩分量
6. HSL hsl;        //色彩分量
8. **if** (times <= 4 && times >= 0.25)
9. {
10. **for** (i = attr.y1; i < attr.y2; i++)
11. {
12. **for** (j = attr.x1; j < attr.x2; j++)
13. {
14. oldcolor = GetPixel(j, i);
15. //类型转换
16. U32TRGB(&rgb, oldcolor);
17. RGB2HSL(rgb, &hsl);
19. //计算公式
20. hsl.l = pow(hsl.l, 1 / times);
22. //类型转换
23. HSL2RGB(&rgb, hsl);
24. oldcolor = RGB2U32(rgb.r, rgb.g, rgb.b);
26. PutPixel(j, i, oldcolor);
27. }
28. }
29. }
30. **else**
31. {
32. //范围超出
33. **return** -1;
34. }
35. **return** 0;
36. }

#### 渐变滤镜算法

## 组员分工及时间安排

###### 6.1 组员分工

|  |  |  |
| --- | --- | --- |
| 姓名 | 分工 | 备注 |
| 王子毅 | 照片管理模块  图像修饰模块 |  |
| 牛宝健 | 图像编辑模块 |  |

###### 6.1 时间安排

|  |  |  |
| --- | --- | --- |
| 时间 | 任务 | 备注 |
| 第一周 | 查找相关资料，设计程序结构。 |  |
| 第二周 | 查找相关资料，编写SVGA模块。 |  |
| 第三周 | 设计图形界面。 |  |
| 第四周 | 编写图像文件读写、图像处理等内容。 |  |
| 第五周 | 编写图像修饰、图像编辑模块。 |  |
| 第六周 | 编写照片管理模块。 |  |