**实习一 BMP灰度图像压缩**

**一、实习目的与要求**

**【问题描述】**

灰度图像的像素值范围在[0，255]之间，如果采用一个像素一个字节的存储方式，势必会造成空间的浪费。如果采用一定的无损压缩算法，可以大大提高减小文件大小，减少存储空间。本课题要求针对提供的256色（8位）位图数据，采用教材上第15章动态规划中图像压缩算法（图像分段合并的思想），设计一个类，实现灰度位图数据的压缩和解压过程。

**【基本要求】**

一个完整的灰度图像类应具有以下功能：

（1）对8位位图数据的读功能，提供ReadBitmap方法。

ReadBitmap方法有一个参数为输入位图文件名(\*.bmp)，它能解析8位位图文件格式，获取位图BITMAPINFOHEADER信息和每个像素的数据信息，放入内存中。

（2）对8位位图数据的写功能，提供WriteBitmap方法。

WriteBitmap方法有一个参数为输出位图文件名(\*.bmp)，它能将内存中的位图文件信息，按照位图格式，写到位图文件中保存。

（3）灰度图像压缩功能，提供Compress方法。

Compress方法有一个参数为输出压缩文件名(\*.img) ，它能将已经装入到内存中的8位位图信息，进行压缩，形成段标题和以变长格式存储的像素的二进制串，写入到文件中（注意：Img文件格式自行定义）。

（4）灰度图像解压功能，提供UnCompress方法。

UnCompress方法有一个参数为输入压缩文件名(\*.img)，它能解析Img文件格式，将其在内存中解压缩为8位位图信息，以便输出为位图文件。

（5）以上是该灰度图像类基本的四个方法，在实现时可根据需要扩充其他方法。在设计时，要使用面向对象的思想，考虑各个成员的访问权限。

**【提高要求】**

（1）基于Windows对话框界面，可选择输入/输出文件名，有压缩进度条显示。

（2）采用不同的数据集，比较其压缩比，采用最有效的压缩方式。

**【测试数据】**

lena.bmp，512\*512\*8

**【测试用例】**

类的测试用例如下：

CCompressImage Test;

Test. ReadBitmap(“数字化.bmp”); 读原始位图

Test. Compress(“Out.img”); 压缩

Test. UnCompress(“Out.img”); 解压

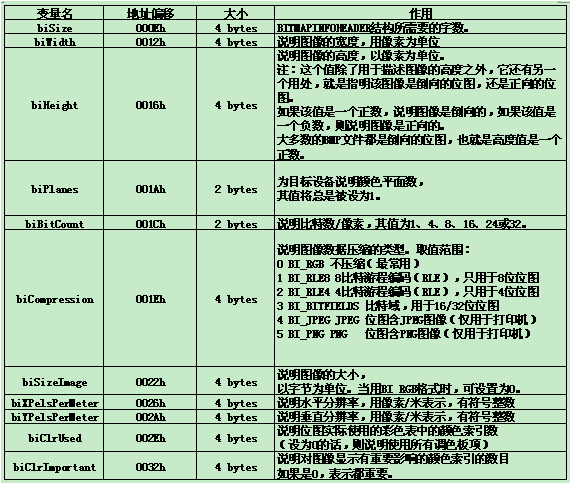
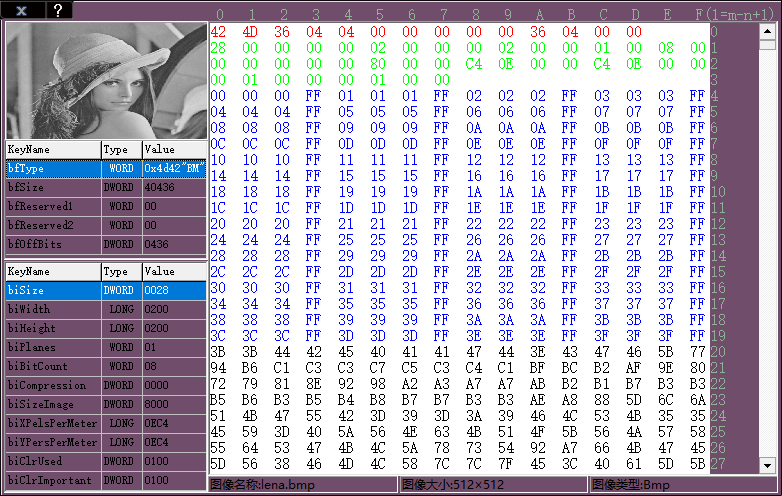
Test. WriteBitmap(“Out.bmp”); 还原位图信息

【测试结果】

可以使用MD5比较解压后的图与原图是否一样，验证你所实现的灰度图像类是否做到了无损压缩。

**【实现提示】**

有关8位的位图格式可以参考MSDN中BITMAPINFOHEADER结构的说明文档，注意其中biBitCount=8的说明。



**二、分析与设计**

1. **需求分析与类设计**
2. **需求分析：**

对于位深度为8位（256色）的位图文件，像素值范围在[0,225]间，每个像素占用一个字节存储会造成空间的浪费。采用图像分段合并的思想的压缩算法，实现该图形的压缩和解压。

1. **类设计：**

类的声明如下：

Class Bitmap{ //Bitmap类

public:

Bitmap(QString name); //将文件名传入类的构造函数

void readBitmap(); //读取位图，获取图象文件头、图象信息头、

//调色板数据、图像数据

void writeBitmap(QString fileName); //写入位图，写入图象文件头、图象信息头、

//调色板数据、图像数据

void Compress(QString fileName); //压缩位图

void UnCompress(); //解压位图

/\*以下是辅助类型、位数等转换的函数\*/

QString BYTEtoBit(BYTE data,int bit);//将unsigned char转为二进制数位保存在QString中

BYTE StrtoBYTE(QString str,int bit); //将QString中的二进制数位转为unsigned char

int StrtoInt(QString str,int bit); //将QString中的二进制数位转为int

private:

QString fileName; //打开的文件名

BITMAPFILEHEADER fileHeader; //图像文件头

BITMAPINFOHEADER infoHeader; //图像信息头

RGBQUAD rgb[256]; //调色板

int width; //像素宽度

int height; //像素长度

BYTE \*\*pixelData = NULL; //保存像素信息

BYTE \*pixel = NULL; //蛇形保存像素信息

QFile outFile; //输出文件

QString LengthBits=""; //暂时存储段头信息

QString AllPixels=""; //暂时存储像素信息

QString All=""; //存储头信息和像素信息

signals:

void Signal(int value); //发送信号，更新进度条

**}；**

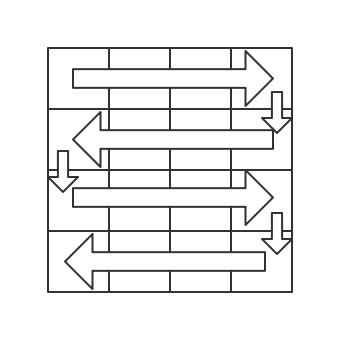
其中，readBitmap()，将构造类时输入位图文件名(\*.bmp)传入，解析8位位图文件格式，获取图象文件头、图象信息头、调色板数据、图像数据，放入内存中，以供之后压缩使用。

writeBitmap(QString fileName)，将内存中的图象文件头、图象信息头、调色板数据、图像数据输出，输出位图文件(\*.bmp)。

Compress(QString fileName)，输出压缩文件(\*.Img) 将已经装入到内存中的8位位图信息，进行压缩，形成段标题和以变长格式存储的像素的二进制串，写入到文件中。

UnCompress(),输入压缩文件(\*.Img)，解析Img文件格式，将其在内存中解压缩为8位位图信息，从而进行输出为位图文件。

1. **算法设计与分析**



首先读bmp文件，前三部分内容图象文件头、图象信息头、调色板数据、图像数据不进行更改，保存图像数据部分内容，以上图所示蛇形获取到像素值。

int size = height \* width; //总像素的个数

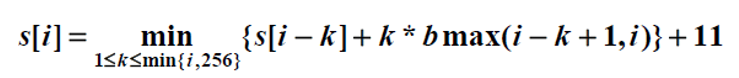
int \*l = new int[size+1]; //保存分段信息

int \*b = new int[size+1]; //保存每个像素所占用位数

int \*s = new int[size+1]; //保存当前最优分段所需的存储位数

int Lmax = 256; //单段最长为256个元素

int header = 11; //每段的断头长11位，其中段长8位，段内元素位数长3位



for (int i = 1; i <= size; i++){

b[i] = length(pixel[i-1]);

bmax = b[i];

s[i] = s[i - 1] + bmax;

l[i] = 1;

for (int j = 2; j <= i && j <= Lmax; j++){

if (bmax<b[i - j + 1]){

bmax = b[i - j + 1];

}

if (s[i]>s[i - j] + j \* bmax){

s[i] = s[i - j] + j \* bmax;

l[i] = j;

}

}

s[i] += header;

}

/\*上述获取像素的位数函数\*/

int length(int i){

int k = 1;

i = i / 2;

while (i>0){

k++;

i = i / 2;

}

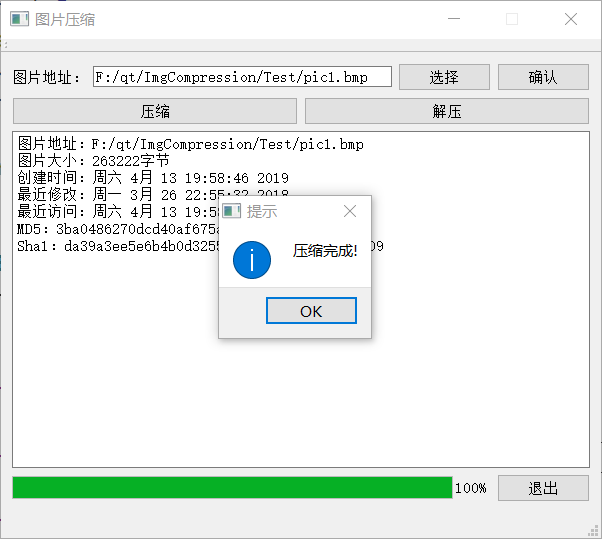
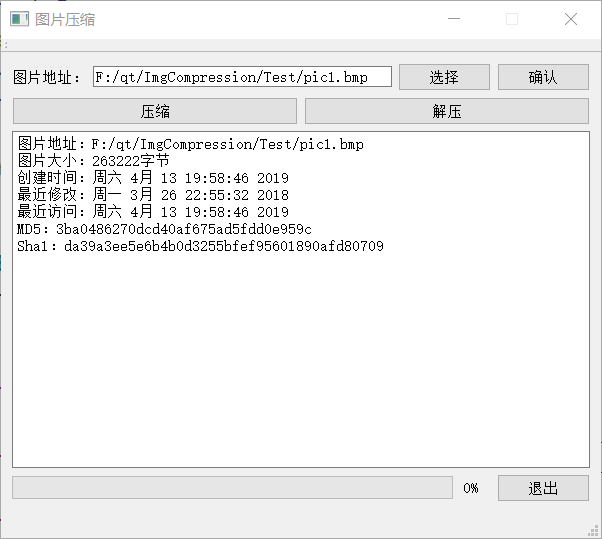
return k;

}

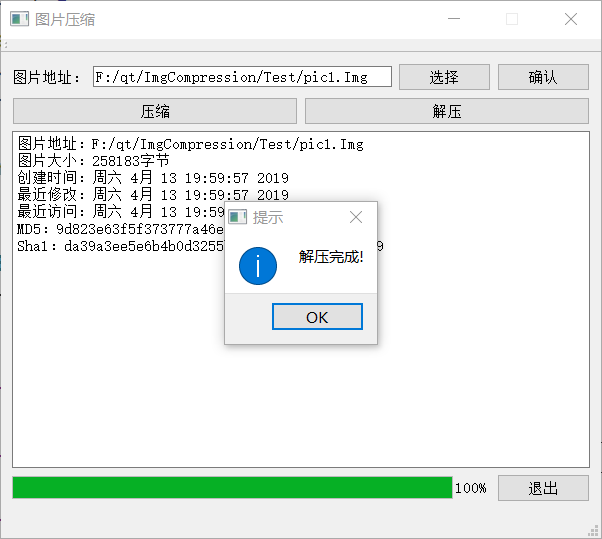
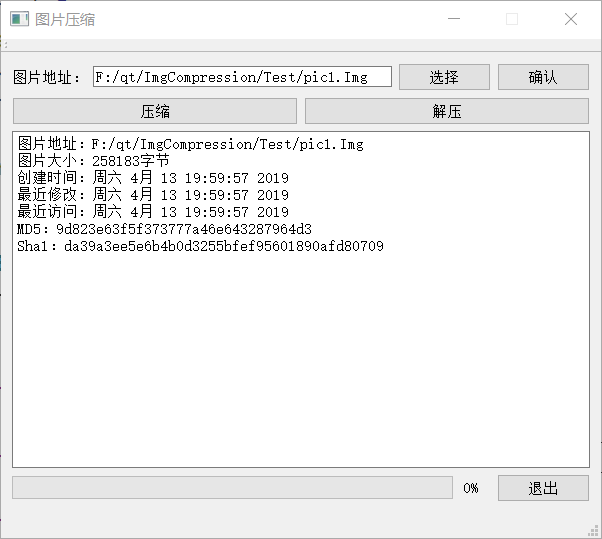
将l[]数组中存储断长的元素取出转为8位，再选出每段中最大的b[]元素转为3位，变为二进制值存入QString中，再将元素值按照上述值变为二进制值存入QString中，两个字符串合并，再存入unsigned char数组中，将其写入压缩文件中。为了解压的方便，同时写入段数数据和unsigned char数组元素个数数据。解压读取上述值后反向操作即可。

**3、功能测试与改进**

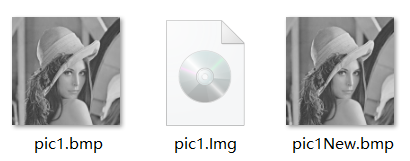
压缩：



解压：



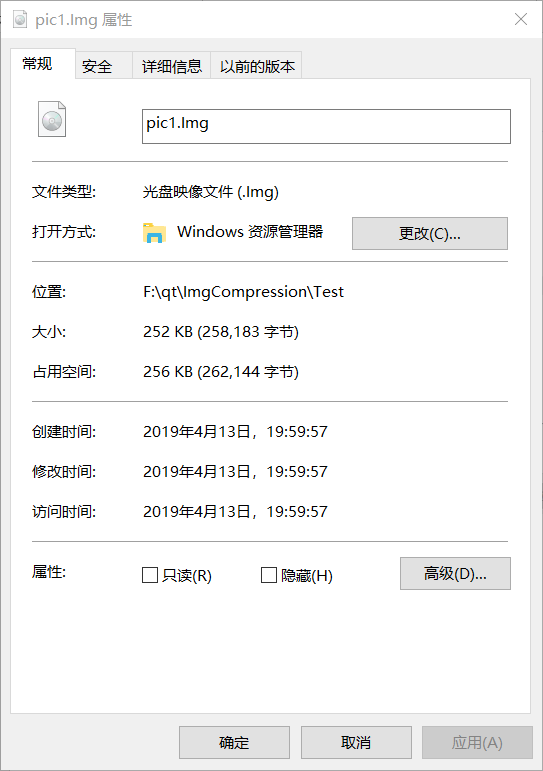
文件：



解压前后文件对比：



压缩文件信息：



**三、实习小结**

对于本次实习，主要可以分为两块：位图的读写和压缩算法。第一次接触文图文件的相关代码，刚开始还是不了解的，学习相关内容后就很好操作了，测试进行度写位图没有问题的话位图部分基本就是没有问题了。压缩算法这块的动态转移方程相对也是较难的，需要跟着相关代码自己构建一遍dp数组的过程才能更好的理解。实习完成了要求的压缩和解压过程，并添加了上学期的哈夫曼压缩的部分，但是发现哈夫曼部分的代码可能存在些问题，压缩率不是很高，也有时候会出现负压缩的情况，所有最后将哈夫曼的部分屏蔽掉了。进度条的使用上，对于QT的多线程不是很熟悉，用了信号与槽，在压缩和解压途中，通过发送进度信息从而更新进度条。

-------------------------------

成绩评定：

教师签名：

批改日期：