

第八次作业



2023E8019482015 姚江瑜

1.1 生成一个200x200的灰度图，分析灰度图的数据格式（各部分的起始地址、大小、各字段的意义等）

200x200的灰度图如下：



通过十六进制查看可以得到该灰度图的数据如下：

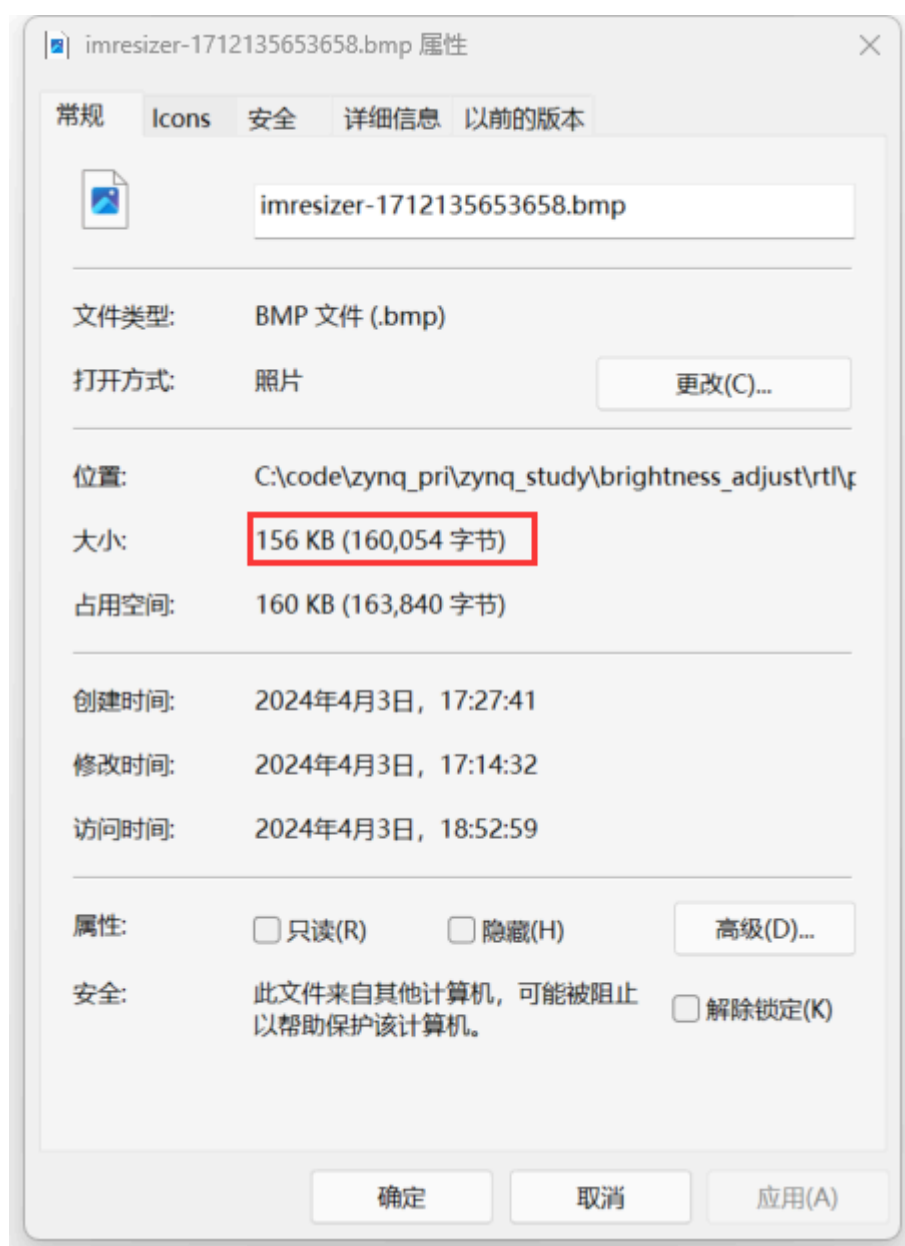
brightness_adjust > doc >  imresizer-1712135653658.bmp																	已解码的文本
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000000	42	4D	36	71	02	00	00	00	00	00	36	00	00	00	28	00	B M 6 q 6 . . . (.
00000010	00	00	C8	00	00	00	C8	00	00	00	01	00	20	00	00	00
00000020	00	00	00	00	00	00	C4	0E	00	00	C4	0E	00	00	00	00
00000030	00	00	00	00	00	00	00	00	00	FF	01	01	01	FF	03	03
00000040	03	FF	00	00	00	FF	00	00	00	FF	03	03	03	FF	08	08
00000050	08	FF	0A	0A	0A	FF	0A	0A	0A	FF	0A	0A	0A	FF	0A	0A
00000060	0A	FF	0C	0C	0C	FF	10	10	10	FF	10	10	10	FF	0E	0E
00000070	0E	FF	0C	0C	0C	FF	0C	0C	0C	FF	0A	0A	0A	FF	09	09
00000080	09	FF	0A	0A	0A	FF	0E	0E	0E	FF	0F	0F	0F	FF	0D	0D
00000090	0D	FF	0A	0A	0A	FF	0B	0B	0B	FF	0B	0B	0B	FF	0D	0D
000000A0	0D	FF	10	10	10	FF	0B	0B	0B	FF	05	05	05	FF	0B	0B
000000B0	0B	FF	17	17	17	FF	1B	1B	1B	FF	20	20	20	FF	27	27 ' '
000000C0	27	FF	15	15	15	FF	1E	1E	1E	FF	2D	2D	2D	FF	15	15	' - - - . . .
000000D0	15	FF	0D	0D	0D	FF	02	02	02	FF	06	06	06	FF	08	08
000000E0	08	FF	1B	1B	1B	FF	27	27	27	FF	29	29	29	FF	36	36 ' ' ' .))) . 6 6
000000F0	36	FF	39	39	39	FF	42	42	42	FF	82	82	82	FF	76	76	6 . 9 9 9 . B B B v v
00000100	76	FF	2E	2E	2E	FF	40	40	40	FF	4C	4C	4C	FF	52	52	v @ @ @ . L L L . R R
00000110	52	FF	59	59	59	FF	3E	3E	3E	FF	76	76	76	FF	20	20	R . Y Y Y . > > > . v v v .
00000120	20	FF	21	21	21	FF	1F	1F	1F	FF	18	18	18	FF	28	28	. ! ! ! ((
00000130	28	FF	2F	2F	2F	FF	30	30	30	FF	07	07	07	FF	0D	0D	(. / / / . 0 0 0
00000140	0D	FF	04	04	04	FF	09	09	09	FF	08	08	08	FF	17	17
00000150	17	FF	33	33	33	FF	30	30	30	FF	20	20	20	FF	25	25	. . 3 3 3 . 0 0 0 % %
00000160	25	FF	3F	3F	3F	FF	66	66	66	FF	8F	8F	8F	FF	5A	5A	% . ? ? ? . f f f Z Z

从上图可以看出，灰度图的文件头数据分析如下：

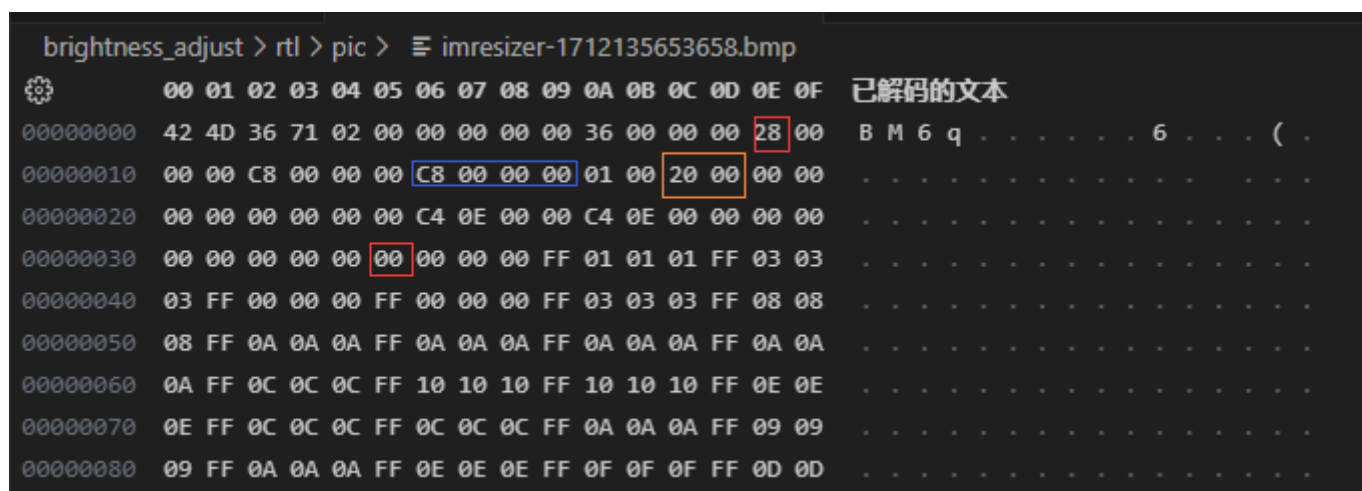
brightness_adjust > rtl > pic > imresizer-1712135653658.bmp

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	已解码的文本
00000000	42	4D	36	71	02	00	00	00	00	00	36	00	00	00	28	00	B M 6 q 6 (.
00000010	00	00	C8	00	00	00	C8	00	00	00	01	00	20	00	00	00
00000020	00	00	00	00	00	00	C4	0E	00	00	C4	0E	00	00	00	00
00000030	00	00	00	00	00	00	00	00	00	FF	01	01	01	FF	03	03
00000040	03	FF	00	00	00	FF	00	00	00	FF	03	03	03	FF	08	08
00000050	08	FF	0A	0A	0A	FF	0A	0A	0A	FF	0A	0A	0A	FF	0A	0A
00000060	0A	FF	0C	0C	0C	FF	10	10	10	FF	10	10	10	FF	0E	0E
00000070	0E	FF	0C	0C	0C	FF	0C	0C	0C	FF	0A	0A	0A	FF	09	09
00000080	09	FF	0A	0A	0A	FF	0E	0E	0E	FF	0F	0F	0F	FF	0D	0D

很明显，文件头的前两个字节是文件类型“BM”；接下来4个字节是文件大小， $0x027136=160054$ ；然后4个字节是保留字段，全为0；最后4个字节是偏移量0x36，也就是说从0x00 00 00 36开始为图像数据。



接下来是信息头数据，分析如下：



信息头的前4个字节是信息头的大小， $0x28=40$ ；接下来4个字节是图像的宽度， $0x00C8=200$ ；再接下来4个字节是图像的高度， $0x00C8=200$ ，也说明了该图像存储是从下到上；然后是颜色平面数， $0x01=1$ ；接下来是每个像素的位数， $0x0020=32$ ；最后是压缩方式， $0x00=0$ ，说明没有压缩。

32位图像数据，不需要调色板数据，每个像素占4个字节，分别是B、G、R、A，所以每个像素的数据是4个字节，也就是32位。所以， 200×200 的图像数据大小是 $200 \times 200 \times 4=160000$ 字节，加上文件头和信息头的大小，总共是**160054**字节，与文件大小一致。

1.2 对200x200的灰度图进行亮度调节

亮度调节的原理：

通过改变仿真输入文件，将200x200文件输入到模块中，模块对输入的灰度图进行亮度调节，然后将调节后的图像输出到仿真输出文件中。将b改为b+60，即将每个像素点值都加60，这样就可以实现亮度调节。

```
//fileId = $fopen("E:\\Class_ex\\modelsim\\read_bmpfile_br_adj\\400x400gray.bmp","rb");
fileId = $fopen("../pic/imresizer-1712135653658.bmp","rb");
//out_file = $fopen("E:\\Class_ex\\modelsim\\read_bmpfile_br_adj\\output_file.bmp","wb");
out_file = $fopen("../pic/output_file.bmp","wb");

// pixels add 30
for(index=data_start_index;index<=bmp_size;index=index+1)
begin
    temp=bmp_data[index];
    //if((bmp_data[index]<245) &(bmp_data[index]>105)) bmp_data_add[index]=bmp_data[index]+10;
    //if(bmp_data[index]<8'd200) bmp_data_add[index]=bmp_data[index]+8'd10;

    if(temp+8'd60>8'd255) bmp_data_add[index]=8'd255;
    else bmp_data_add[index]=bmp_data[index]+8'd60;
end
```

对200x200的灰度图进行亮度调节，调节后的图像如下：



从上图可以看出，左边是原始的200x200的灰度图，右边是亮度调节后的200x200的灰度图。

很明显，调节后的图像亮度更高，整体变亮了。