

# 北 京 林 业 大 学

## 2019 学年—2020 学年第 2 学期计算机算法设计与实践实验报告书

专业： 计算机科学与技术(创新实验班) 班级： 计创 18

姓 名： 连月菡 学 号： 181002222

实验地点： 家 任课教师： 王忠芝

实验题目： 实验 2 图像处理程序设计

实验环境： Visual Studio 2019 Community

### 一、 实验目的

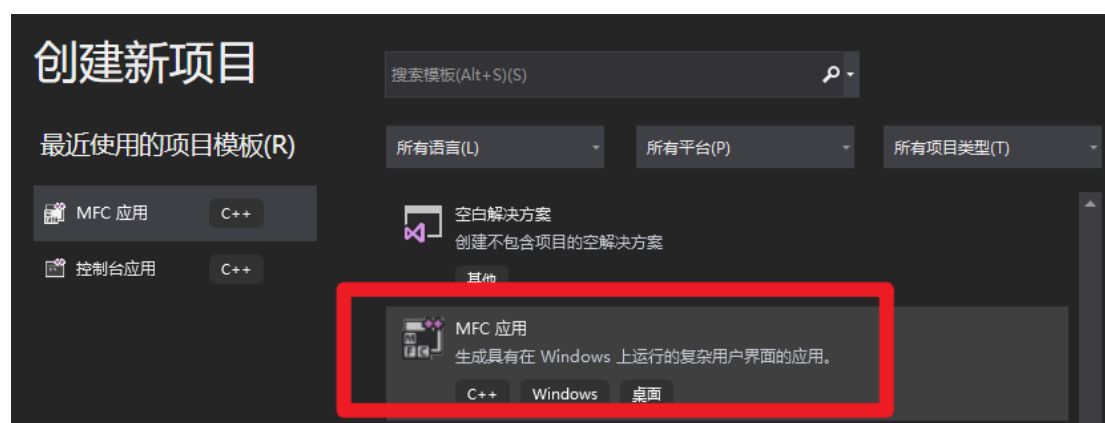
利用 c/c++ 作为编程环境进行图像处理的编程实践,掌握 Windows 下的 BMP 图像的线性变换编程方法。

### 二、 实验内容

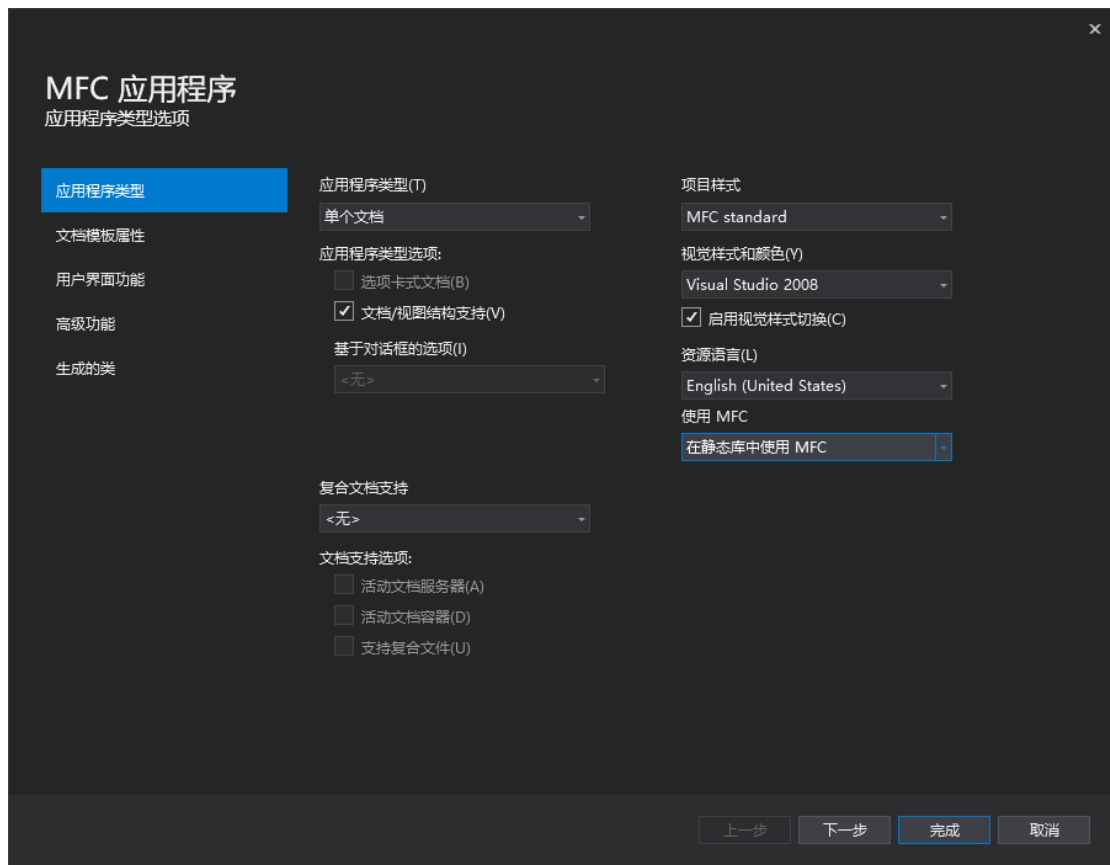
- 1、 编写程序处理 BMP 格式的图像。
- 2、 掌握图像线性变换的处理方法。
- 3、 实现 3\_1, 3\_2 任务。

### 三、 实验步骤

#### 1. 新建 MFC 程序



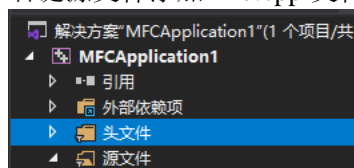
并选择一些必要的选项



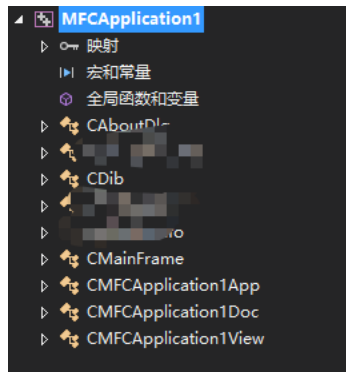
点击完成,生成成功。  
右键头文件添加 Dib.h 文件。



右键源文件添加 Dib.cpp 文件。



2.修改 DOC 和 VIEW 类的代码  
点击上方的视图, 选择类视图,找到这两个类



```
1. // MFCApplication1Doc.h: CMFCApplication1Doc 类的接口
2.
3. #include "../MFC_DIP/Dib.h"
4. extern CDib m_dib;
5.
6. #pragma once
7.
8.
9.
10. class CMFCApplication1Doc : public CDocument
11. {
12. protected: // 仅从序列化创建
13.     CMFCApplication1Doc() noexcept;
14.     DECLARE_DYNCREATE(CMFCApplication1Doc)
15.
16. // 特性
17. public:
18.     CDib m_dib;
19. // 操作
20. public:
21.     HGLOBAL GetHObject() const // 获取 Dib 对象的句柄
22.     {
23.         return m_hDIB;
24.     }
25.     CPalette* GetDocPal() const // 获取调色板指针
26.     {
27.         return m_palDIB;
28.     }
29.     CSize GetDocDimension() const // 获取调色板大小
30.     {
31.         return m_sizeDoc;
32.     }
33.     void UpdateObject(HGLOBAL hDIB); // 更新 dib 对象
34.
```

```

35. // 重写
36. public:
37.     virtual BOOL OnNewDocument();
38.     virtual void Serialize(CArchive& ar);
39. #ifdef SHARED_HANDLERS
40.     virtual void InitializeSearchContent();
41.     virtual void OnDrawThumbnail(CDC& dc, LPRECT lprcBounds);
42. #endif // SHARED_HANDLERS
43.
44. // 实现
45. public:
46.     void SetDib();           //初始化 dib 对象
47.     virtual ~CMFCApplication1Doc();
48. #ifdef _DEBUG
49.     virtual void AssertValid() const;
50.     virtual void Dump(CDumpContext& dc) const;
51. #endif
52.
53. protected:
54.     HGLOBAL m_hDIB;
55.     CPalette* m_palDIB;
56.     CSize m_sizeDoc;
57. // 生成的消息映射函数
58. protected:
59.     DECLARE_MESSAGE_MAP()
60.
61. #ifdef SHARED_HANDLERS
62.     // 用于为搜索处理程序设置搜索内容的 Helper 函数
63.     void SetSearchContent(const CString& value);
64. #endif // SHARED_HANDLERS
65. public:
66.     virtual BOOL OnOpenDocument(LPCTSTR lpszPathName);
67.     virtual BOOL OnSaveDocument(LPCTSTR lpszPathName);
68. };

```

## Doc.cpp 修改为

```

1. // MFCApplication1Doc.cpp: CMFCApplication1Doc 类的实现
2. //
3.
4. #include "pch.h"
5. #include "framework.h"
6. // SHARED_HANDLERS 可以在实现预览、缩略图和搜索筛选器句柄的
7. // ATL 项目中进行定义，并允许与该项目共享文档代码。

```

```
8. #ifndef SHARED_HANDLERS
9. #include "MFCApplication1.h"
10. #endif
11.
12. #include "MFCApplication1Doc.h"
13.
14. #include <propkey.h>
15.
16. #ifdef _DEBUG
17. #define new DEBUG_NEW
18. #endif
19.
20. // CMFCApplication1Doc
21.
22. IMPLEMENT_DYNCREATE(CMFCApplication1Doc, CDocument)
23.
24. BEGIN_MESSAGE_MAP(CMFCApplication1Doc, CDocument)
25. END_MESSAGE_MAP()
26.
27.
28. // CMFCApplication1Doc 构造/析构
29.
30. CMFCApplication1Doc::CMFCApplication1Doc() noexcept
31. {
32.     // TODO: 在此添加一次性构造代码
33.     m_hDIB = NULL; // 初始化变量
34.     m_palDIB = NULL;
35.     m_sizeDoc = CSize(1, 1);
36.
37. }
38.
39. CMFCApplication1Doc::~CMFCApplication1Doc()
40. {
41.     if (m_hDIB != NULL) // 判断是否有 DIB 对象
42.     {
43.         ::GlobalFree((HGLOBAL)m_hDIB);
44.     }
45.
46.
47.     if (m_palDIB != NULL) // 判断调色板是否存在
48.     {
49.         delete m_palDIB;
50.     }
51.
```

```
52. }
53.
54. BOOL CMFCApplication1Doc::OnNewDocument()
55. {
56.     if (!CDocument::OnNewDocument())
57.         return FALSE;
58.
59.     // TODO: 在此添加重新初始化代码
60.     // (SDI 文档将重用该文档)
61.
62.     return TRUE;
63. }
64.
65.
66.
67.
68. // CMFCApplication1Doc 序列化
69.
70. void CMFCApplication1Doc::Serialize(CArchive& ar)
71. {
72.     if (ar.IsStoring())
73.     {
74.         // TODO: 在此添加存储代码
75.     }
76.     else
77.     {
78.         // TODO: 在此添加加载代码
79.     }
80. }
81.
82. #ifdef SHARED_HANDLERS
83.
84. // 缩略图的支持
85. void CMFCApplication1Doc::OnDrawThumbnail(CDC& dc, LPRECT lprcBounds)
86. {
87.     // 修改此代码以绘制文档数据
88.     dc.FillSolidRect(lprcBounds, RGB(255, 255, 255));
89.
90.     CString strText = _T("TODO: implement thumbnail drawing here");
91.     LOGFONT lf;
92.
93.     CFont* pDefaultGUIFont = CFont::FromHandle((HFONT) GetStockObject(DEFAULT_GUI_FONT));
94.     pDefaultGUIFont->GetLogFont(&lf);
```

```

95.     lf.lfHeight = 36;
96.
97.     CFont fontDraw;
98.     fontDraw.CreateFontIndirect(&lf);
99.
100.    CFont* pOldFont = dc.SelectObject(&fontDraw);
101.    dc.DrawText(strText, lprcBounds, DT_CENTER | DT_WORDBREAK);
102.    dc.SelectObject(pOldFont);
103. }
104.
105. // 搜索处理程序的支持
106. void CMFCApplication1Doc::InitializeSearchContent()
107. {
108.     CString strSearchContent;
109.     // 从文档数据设置搜索内容。
110.     // 内容部分应由“;”分隔
111.
112.     // 例
113.     如: strSearchContent = _T("point;rectangle;circle;ole object;");
114.     SetSearchContent(strSearchContent);
115. }
116. void CMFCApplication1Doc::SetSearchContent(const CString& value)
117. {
118.     if (value.IsEmpty())
119.     {
120.         RemoveChunk(PKEY_Search_Contents.fmtid, PKEY_Search_Contents.pid);
121.     }
122.     else
123.     {
124.         CMFCFilterChunkValueImpl *pChunk = nullptr;
125.         ATLTRY(pChunk = new CMFCFilterChunkValueImpl);
126.         if (pChunk != nullptr)
127.         {
128.             pChunk->SetTextValue(PKEY_Search_Contents, value, CHUNK_TEXT);
129.             SetChunkValue(pChunk);
130.         }
131.     }
132. }
133.
134. #endif // SHARED_HANDLERS
135.

```

```
136. // CMFCApplication1Doc 诊断
137.
138. #ifdef _DEBUG
139. void CMFCApplication1Doc::AssertValid() const
140. {
141.     CDocument::AssertValid();
142. }
143.
144. void CMFCApplication1Doc::Dump(CDumpContext& dc) const
145. {
146.     CDocument::Dump(dc);
147. }
148. #endif //_DEBUG
149.
150.
151. // CMFCApplication1Doc 命令
152.
153.
154. BOOL CMFCApplication1Doc::OnOpenDocument(LPCTSTR lpszPathName)
155. {
156.
157.     if (!CDocument::OnOpenDocument(lpszPathName))
158.         return FALSE;
159.
160.     // TODO: 在此添加您专用的创建代码
161.     CFile file;
162.     if (!file.Open(lpszPathName, CFile::modeRead |
163.         CFile::shareDenyWrite))// 打开文件
164.     {
165.         return FALSE;// 返回 FALSE
166.     }
167.     DeleteContents();
168.     m_hDIB = m_dib.LoadFile(file);
169.     if (m_hDIB == NULL)// 调用 LoadFile()读取图象
170.     {
171.         return FALSE;
172.     }
173.     SetDib();// 初始化 DIB
174.     if (m_hDIB == NULL) // 判断读取文件是否成功
175.     {
176.         AfxMessageBox(L"读取图像时出错");// 提示出错
177.         return FALSE;// 返回 FALSE
178.     }
179.     SetPathName(lpszPathName);// 设置文件名称
```



```

180.     SetModifiedFlag(FALSE); // 初始化修改标记为 FALSE
181.     return TRUE; // 返回 TRUE
182.
183. }
184.
185.
186. BOOL CMFCApplication1Doc::OnSaveDocument(LPCTSTR lpszPathName)
187. {
188.     // TODO: 在此添加专用代码和/或调用基类
189.     CFile file;
190.     if (!file.Open(lpszPathName, CFile::modeCreate | // 打开文件
191.         CFile::modeReadWrite | CFile::shareExclusive))
192.     {
193.         return FALSE; // 返回 FALSE
194.     }
195.     BOOL bSuccess = FALSE;
196.     bSuccess = m_dib.SaveFile(m_hDIB, file); // 保存图象
197.     file.Close(); // 关闭文件
198.     SetModifiedFlag(FALSE); // 重置脏标记为 FALSE
199.     if (!bSuccess)
200.     {
201.         AfxMessageBox(L"保存 BMP 图象时出错"); // 提示出错
202.     }
203.     return bSuccess;
204.     //return CDocument::OnSaveDocument(lpszPathName);
205. }
206.
207. void CMFCApplication1Doc::SetDib()
208. {
209.     LPSTR lpdib = (LPSTR) ::GlobalLock((HGLOBAL)m_hDIB);
210.
211.     if (m_dib.GetWidth(lpdib) > INT_MAX || m_dib.GetHeight(lpdib) > INT_MAX
212.         ) // 判断图象是否过大
213.     {
214.         ::GlobalUnlock((HGLOBAL)m_hDIB);
215.         ::GlobalFree((HGLOBAL)m_hDIB); // 释放 DIB 对象
216.         m_hDIB = NULL; // 设置 DIB 为空
217.         AfxMessageBox(L"初始化失败");
218.         return;
219.     }
220.     m_sizeDoc = CSize((int)m_dib.GetWidth(lpdib), (int)m_dib.GetHeight(lpdib)); // 设置文档大小
221.     ::GlobalUnlock((HGLOBAL)m_hDIB);
222.     m_palDIB = new CPalette; // 创建新调色板

```

```

222.     if (m_palDIB == NULL)// 判断是否创建成功
223.     {
224.         ::GlobalFree((HGLOBAL)m_hDIB); // 失败
225.         m_hDIB = NULL;// 设置 DIB 对象为空
226.         return;
227.     }
228.     // 调用 CreateDIBPalette 来创建调色板
229.     if (m_dib.ConstructPalette(m_hDIB, m_palDIB) == NULL)
230.     {
231.         delete m_palDIB;// 删除
232.         m_palDIB = NULL;// 设置为空
233.         return;// 返回空
234.     }
235. }
236.
237. void CMFCApplication1Doc::UpdateObject(HGLOBAL hDIB)
238. {
239.     if (m_hDIB != NULL) // 判断 DIB 是否为空
240.     {
241.         ::GlobalFree((HGLOBAL)m_hDIB);// 非空，则清除
242.     }
243.     m_hDIB = hDIB; // 替换成新的 DIB 对象
244. }

```

## View.cpp

```

1. // MFCApplication1View.cpp: CMFCApplication1View 类的实现
2. //
3.
4. #include "pch.h"
5. #include "framework.h"
6. // SHARED_HANDLERS 可以在实现预览、缩略图和搜索筛选器句柄的
7. // ATL 项目中进行定义，并允许与该项目共享文档代码。
8. #ifndef SHARED_HANDLERS
9. #include "MFCApplication1.h"
10. #endif
11.
12. #include "MFCApplication1Doc.h"
13. #include "MFCApplication1View.h"
14. #include "CdlgImgInfo.h"
15.
16. #ifdef _DEBUG
17. #define new DEBUG_NEW
18. #endif

```

```

19. #include "CChangeLights.h"
20.
21.
22. // CMFCApplication1View
23.
24. IMPLEMENT_DYNCREATE(CMFCApplication1View, CView)
25.
26. BEGIN_MESSAGE_MAP(CMFCApplication1View, CView)
27.     // 标准打印命令
28.     ON_COMMAND(ID_FILE_PRINT, &CView::OnFilePrint)
29.     ON_COMMAND(ID_FILE_PRINT_DIRECT, &CView::OnFilePrint)
30.     ON_COMMAND(ID_FILE_PRINT_PREVIEW, &CView::OnFilePrintPreview)
31.     ON_COMMAND(ID_DispImgInfo, &CMFCApplication1View::OnDispimginfo)
32.     ON_COMMAND(ID_ChangeLight, &CMFCApplication1View::OnChangelight)
33. END_MESSAGE_MAP()
34.
35. // CMFCApplication1View 构造/析构
36.
37. CMFCApplication1View::CMFCApplication1View() noexcept
38. {
39.     // TODO: 在此处添加构造代码
40.
41. }
42.
43. CMFCApplication1View::~~CMFCApplication1View()
44. {
45. }
46.
47. BOOL CMFCApplication1View::PreCreateWindow(CREATESTRUCT& cs)
48. {
49.     // TODO: 在此处通过修改
50.     // CREATESTRUCT cs 来修改窗口类或样式
51.
52.     return CView::PreCreateWindow(cs);
53. }
54.
55. // CMFCApplication1View 绘图
56.
57. void CMFCApplication1View::OnDraw(CDC* pDC)
58. {
59.     CMFCApplication1Doc* pDoc = GetDocument();
60.     ASSERT_VALID(pDoc);
61.     if (!pDoc)
62.         return;

```

```

63.     // TODO: 在此处为本机数据添加绘制代码
64.     HGLOBAL hDIB = pDoc->GetHObject();
65.
66.     // 判断 DIB 是否为空
67.     if (hDIB != NULL)
68.     {
69.         LPSTR lpDibSection = (LPSTR) ::GlobalLock((HGLOBAL)hDIB);
70.
71.         // 获取 DIB 宽度
72.         int cxDIB = (int)pDoc->m_dib.GetWidth(lpDibSection);
73.
74.         // 获取 DIB 高度
75.         int cyDIB = (int)pDoc->m_dib.GetHeight(lpDibSection);
76.         ::GlobalUnlock((HGLOBAL)hDIB);
77.         CRect rcDIB;
78.         rcDIB.top = rcDIB.left = 0;
79.         rcDIB.right = cxDIB;
80.         rcDIB.bottom = cyDIB;
81.         CRect rcDest = rcDIB;
82.         // 输出 DIB
83.         pDoc->m_dib.DrawDib(pDC->m_hDC, &rcDest, pDoc->GetHObject(),
84.             &rcDIB, pDoc->GetDocPal());
85.     }
86.
87. }
88.
89.
90. // CMFCApplication1View 打印
91.
92. BOOL CMFCApplication1View::OnPreparePrinting(CPrintInfo* pInfo)
93. {
94.     // 默认准备
95.     return DoPreparePrinting(pInfo);
96. }
97.
98. void CMFCApplication1View::OnBeginPrinting(CDC* /*pDC*/, CPrintInfo* /*pInfo*
    */)
99. {
100.     // TODO: 添加额外的打印前进行的初始化过程
101. }
102.
103. void CMFCApplication1View::OnEndPrinting(CDC* /*pDC*/, CPrintInfo* /*pInfo*
    */)
104. {

```

```

105.     // TODO: 添加打印后进行的清理过程
106. }
107.
108.
109. // CMFCApplication1View 诊断
110.
111. #ifdef _DEBUG
112. void CMFCApplication1View::AssertValid() const
113. {
114.     CView::AssertValid();
115. }
116.
117. void CMFCApplication1View::Dump(CDumpContext& dc) const
118. {
119.     CView::Dump(dc);
120. }
121.
122. CMFCApplication1Doc* CMFCApplication1View::GetDocument() const // 非调试版本
    是内联的
123. {
124.     ASSERT(m_pDocument->IsKindOf(RUNTIME_CLASS(CMFCApplication1Doc)));
125.     return (CMFCApplication1Doc*)m_pDocument;
126. }
127. #endif // _DEBUG
128.
129.
130. // CMFCApplication1View 消息处理程序
131.
132.
133. void CMFCApplication1View::OnDispiminfo()
134. {
135.     // TODO: 在此添加命令处理程序代码
136.     // 显示图像基本信息
137.     CMFCApplication1Doc* pDoc = GetDocument(); // 获取文档
138.     long lSrcLineBytes; // 图像每行的字节数
139.     long lSrcWidth; // 图像的宽度和高度
140.     long lSrcHeight;
141.     int lpSrcBitCount; // 图像的位深
142.     LPSTR lpSrcDib; // 指向源图像的指针
143.     LPSTR lpSrcStartBits; // 指向源像素的指针
144.     lpSrcDib = (LPSTR)::GlobalLock((HGLOBAL)pDoc->GetHObject()); // 锁定
        DIB
145.     if (!lpSrcDib) return;

```

```

146.     lpSrcStartBits = pDoc->m_dib.GetBits(lpSrcDib); // 找到 DIB 图像像素起始位置
147.     lSrcWidth = pDoc->m_dib.GetWidth(lpSrcDib); // 获取图像的宽度
148.     lSrcHeight = pDoc->m_dib.GetHeight(lpSrcDib); // 获取图像的高度
149.     lpSrcBitCount = pDoc->m_dib.GetBitCount(lpSrcDib); // 获取图像位深
150.     lSrcLineBytes = pDoc->m_dib.GetReqByteWidth(lSrcWidth * lpSrcBitCount);
        // 计算图像每行的字节数
151.     //////////////////////////////////////
        //////////////////////////////////////
152.     CdlgImgInfo dlgPara; // 创建对话框
153.     dlgPara.m_Width = lSrcWidth;
154.     dlgPara.m_Height = lSrcHeight;
155.     dlgPara.m_bitCount = lpSrcBitCount;
156.     if (dlgPara.DoModal() != IDOK) // 显示对话框,
157.     {
158.         return;
159.     }
160.     ::GlobalUnlock((HGLOBAL)pDoc->GetHObject()); // 解除锁定
161.
162. }
163.
164.
165.
166.
167.
168. void CMFCApplication1View::OnChangeLight()
169. {
170.     // TODO: 在此添加命令处理程序代码
171.
172.     //::MessageBox(NULL, L"hello", L"welcome", MB_OK);
173.
174.
175.     // 显示图像基本信息
176.     CMFCApplication1Doc* pDoc = GetDocument(); // 获取文档
177.
        //////////////////////////////////////
        //////////////////////////////////////
178.     long lSrcLineBytes; // 图像每行的字节数
179.     long lSrcWidth; // 图像的宽度
180.     long lSrcHeight; // 图像的高度
181.     int lpSrcBitCount; // 图像的位深
182.     LPSTR lpSrcDib; // 指向源图像的指针
183.     LPSTR lpSrcStartBits; // 指向源像素的指针
184.

```

```

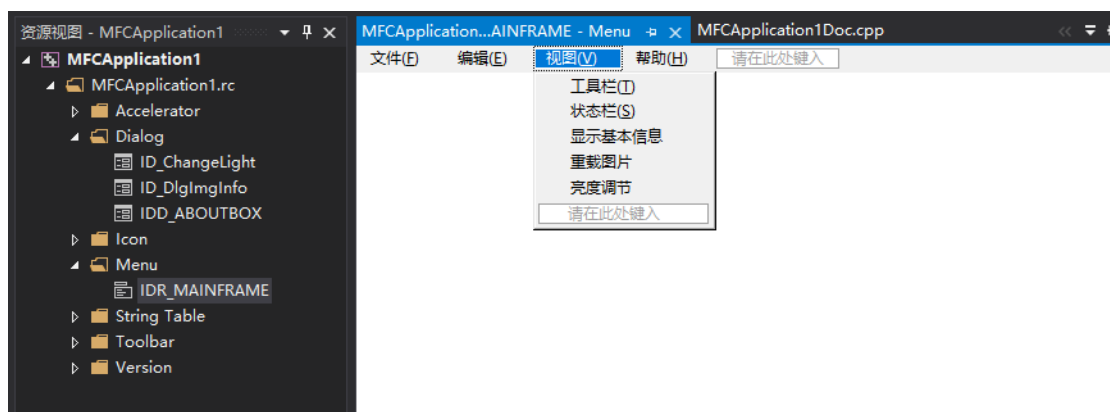
185.     lpSrcDib = (LPSTR) ::GlobalLock((HGLOBAL)pDoc->GetHObject()); // 锁定
        DIB
186.     if (!lpSrcDib)
187.     {
188.         ::MessageBox(NULL, L"hello", L"welcome", MB_OK);
189.         return;
190.     }
191.
192.
193.     /*
194.     if (pDoc->m_dib.GetColorNum(lpSrcDib) != 256) // 判断是否是 8-bmp 位图
195.     {
196.         AfxMessageBox(L"对不起, 不是 256 色位图!"); // 警告
197.         ::GlobalUnlock((HGLOBAL) pDoc->GetHObject()); // 解除锁定
198.         return; // 返回
199.     } // 判断是否是 8-bpp 位图, 不是则返回
200.     */
201.
202.
203.     //////////////////////////////////////
        //////////////////////////////////////
204.
205.     lpSrcStartBits = pDoc->m_dib.GetBits(lpSrcDib); // 找到 DIB 图象像素起始位
        置
206.     lSrcWidth = pDoc->m_dib.GetWidth(lpSrcDib); // 获取图象的宽度
207.     lSrcHeight = pDoc->m_dib.GetHeight(lpSrcDib); // 获取图象的高度
208.     lpSrcBitCount = pDoc->m_dib.GetBitCount(lpSrcDib); // 获取图像位深
209.     lSrcLineBytes = pDoc->m_dib.GetReqByteWidth(lSrcWidth * lpSrcBitCount);
        // 计算图像每行的字节数
210.
        //////////////////////////////////////
        //////////////////////////////////////
211.     CChangeLights dlgPara; // 创建对话框
212.
213.
214.
215.     if (dlgPara.DoModal() != IDOK) // 显示对话框
216.     {
217.
218.         return;
219.     }
220.
221.
222.

```

```

223.
224.     for (int i = 0; i < lSrcLineBytes * lSrcHeight; i++)
225.     {
226.
227.
228.         int temp = (unsigned char)*lpSrcStartBits;
229.
230.         if (temp + dlgPara.light < 0)
231.             temp = 0;
232.         else if (temp + dlgPara.light > 255)
233.             temp = 255;
234.         else temp += dlgPara.light;
235.
236.         *lpSrcStartBits = temp;
237.
238.         lpSrcStartBits++;
239.     }
240.
241.
242.     //设置文档修改标志
243.     pDoc->SetModifiedFlag(true);
244.     //更新视图
245.     pDoc->UpdateAllViews(NULL);
246.
247.     ::GlobalUnlock((HGLOBAL)pDoc->GetHObject()); // 解除锁定
248.
249.
250. }

```



添加对话框 ID 显示基本信息 和亮度调节。  
其他步骤同试验任务书。



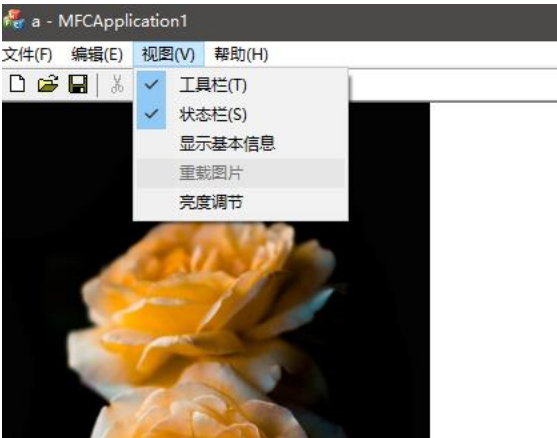
进行编译运行。  
打开一张图片。

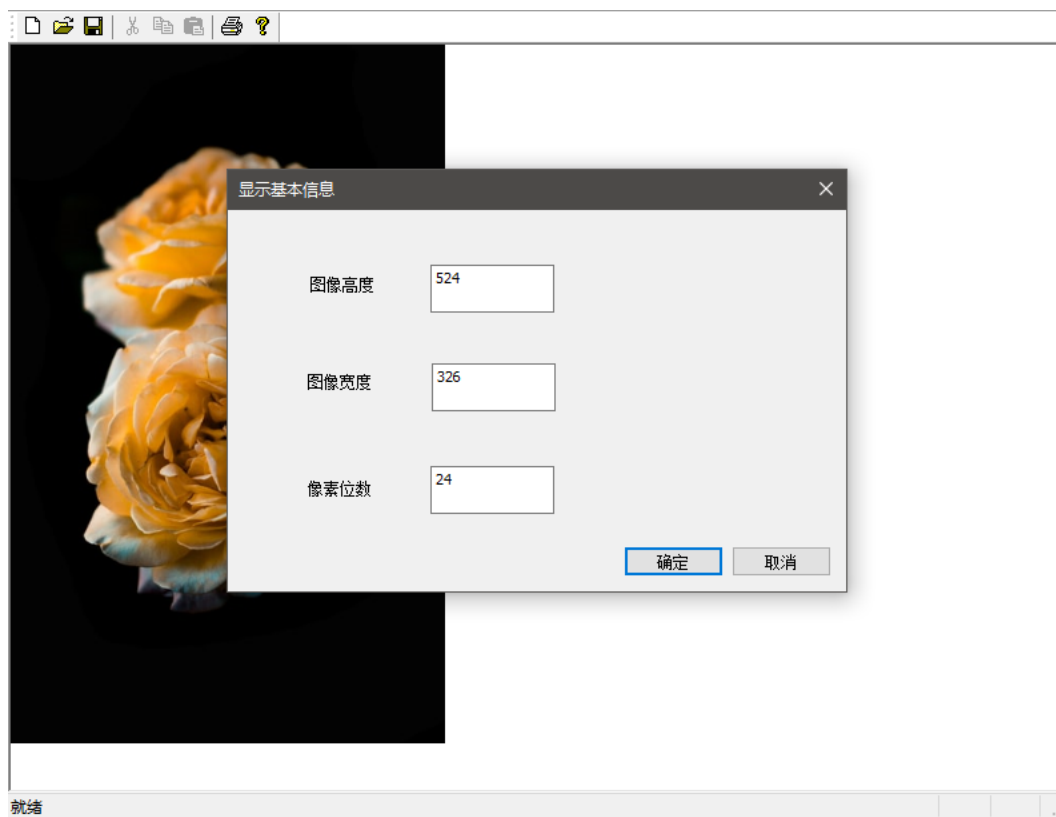


1.显示图片



2,显示基本信息





### 3.调节亮度



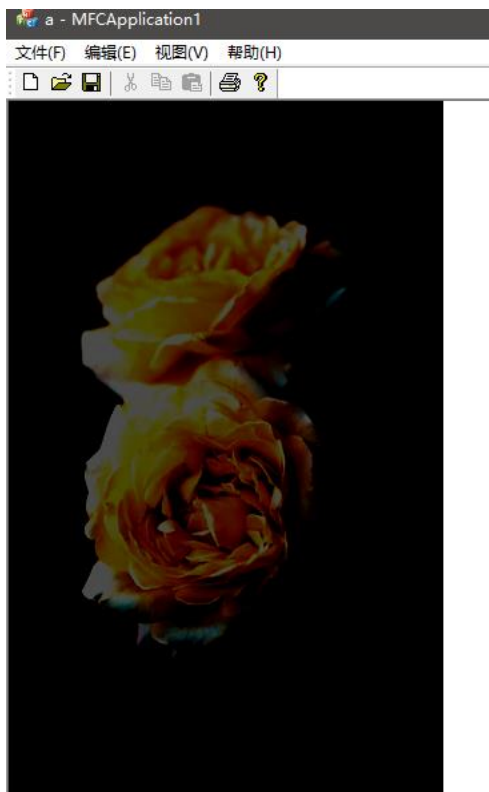
图片变亮了



若输入 90,则是如下结果



输入-90,则是如下结果



#### 四、实验心得

通过本次实验,我对 MFC 框架有了初步的认识,并结合数字图像处理专业课程的内容,对图片进行增量减暗的处理,还能保存显示图片,增加了新的计算机技能。