**使用说明书**

目录

**PPM、PGM和PBM图像处理1**

**RAW图像处理7**

**BMP图像处理9**

**其它处理29**

**PPM、PGM和PBM图像处理**

|  |  |
| --- | --- |
| void OTSUBinarization(char\* input,char\* output) | OTSU二值化。input是输入文件名，output是输出文件名。支持P5格式的PGM图像。 |
| void SegmentsOTSUBinarization(char\* input,char\* output) | OTSU二值化划分。input是输入文件名，output是输出文件名。支持P5格式的PGM图像。 |
| void P3PPMBlur(char\* input,char\* output) | PPM图像模糊，input是输入文件名，output是输出文件名。支持P3格式的PPM图像。 |
| unsigned char\*\* ReadPBM(char\* input) | 读取PBM图像并返回图像数据。input是要读取的PBM图像文件名。支持P4格式的PBM图像。 |
| void WritePBM(unsigned char\*\* Input,char\* output) | 保存PBM图像。Input是输入的图像数据，output是输出文件名。支持P4格式的PBM图像。 |
| void PGMSobel(char\* input,char\* output,int Mx[3][3],int My[3][3],int max,int min) | Sobel算子，input是输入文件名，output是输出文件名。支持P5格式的PGM文件。  参考模板：  int Mx[3][3] = {{-1, 0, 1}, {-2, 0, 2}, {-1, 0, 1}}  int My[3][3] = {{-1, -2, -1}, {0, 0, 0}, {1, 2, 1}}  int max = -9999  int min = 9999 |
| void PGMSobelX(char\* input,char\* output,int Mx[3][3],int My[3][3],int max,int min) | X方向滤波，input是输入文件名，output是输出文件名。支持P5格式的PGM文件。  参考模板：  int Mx[3][3] = {{-1, 0, 1}, {-2, 0, 2}, {-1, 0, 1}}  int My[3][3] = {{-1, -2, -1}, {0, 0, 0}, {1, 2, 1}}  int max = -9999  int min = 9999 |
| void PGMSobelY(char\* input,char\* output,int Mx[3][3],int My[3][3],int max,int min) | Y方向滤波，input是输入文件名，output是输出文件名。支持P5格式的PGM文件。  参考模板：  int Mx[3][3] = {{-1, 0, 1}, {-2, 0, 2}, {-1, 0, 1}}  int My[3][3] = {{-1, -2, -1}, {0, 0, 0}, {1, 2, 1}}  int max = -9999  int min = 9999 |
| void PGMSobel1(char\* input,char\* output,int min,int max,int mx[3][3],int my[3][3]) | Sobel算子，input是输入文件名，output是输出文件名。min和max是图像归一化相关参数，如min = 1000000, max = 0；mx和my分别是Sobel算子的X和Y方向模板。支持P2和P5格式的PGM。  参考模板 ：  int mx[3][3] = {  {-1, 0, 1},  {-2, 0, 2},  {-1, 0, 1}  };  int my[3][3] = {  {-1, -2, -1},  {0, 0, 0},  {1, 2, 1}  }; |
| void PGMSobelX1(char\* input,char\* output,int min,int max,int mx[3][3],int my[3][3]) | X方向梯度，input是输入文件名，output是输出文件名。min和max是图像归一化相关参数，如min = 1000000, max = 0；mx和my分别是Sobel算子的X和Y方向模板。支持P2和P5格式的PGM。  参考模板 ：  int mx[3][3] = {  {-1, 0, 1},  {-2, 0, 2},  {-1, 0, 1}  };  int my[3][3] = {  {-1, -2, -1},  {0, 0, 0},  {1, 2, 1}  }; |
| void PGMSobelY1(char\* input,char\* output,int min,int max,int mx[3][3],int my[3][3]) | Y方向梯度，input是输入文件名，output是输出文件名。min和max是图像归一化相关参数，如min = 1000000, max = 0；mx和my分别是Sobel算子的X和Y方向模板。支持P2和P5格式的PGM。  参考模板 ：  int mx[3][3] = {  {-1, 0, 1},  {-2, 0, 2},  {-1, 0, 1}  };  int my[3][3] = {  {-1, -2, -1},  {0, 0, 0},  {1, 2, 1}  }; |
| void PGMSobel2(char\* input,char\* XOutput,char\* YOutput,char\* SobelOutput,int sobel\_x[3][3],int sobel\_y[3][3],int min,int max) | Sobel算子，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。XOutput是输出的X方向的梯度图像，YOutput是输出的Y方向的梯度图像，SobelOutput是输出的整幅图像的Sobel算子计算结果，min和max是图像归一化的相关参数，如min=100,max=0。  参考模板：  int sobel\_x[3][3]={{-1, 0, 1},{-2, 0, 2},{-1, 0, 1}};  int sobel\_y[3][3]={{1, 2, 1}, {0, 0, 0},{-1,-2,-1}}; |
| void HorizSobel(char\* input,char\* output) | 水平Sobel算子，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。 |
| void VertSobel(char\* input,char\* output) | 垂直Sobel算子，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。 |
| void PGMSobel1(char\* input,char\* output,int threshold) | Sobel算子，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。threshold是目标阈值，如threshold=80。 |
| void PGMHistogramEqualization(char\* input,char\* output) | 直方图均衡化，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。 |
| PPMImage\* ReadPPM(char\* input) | PPM图像读取，input是要读取的PPM图像文件名。支持P6格式的PPM图像。  需要引入的结构体：  typedef struct {  unsigned char red, green, blue; //像素的颜色由RGB（红/绿/蓝）表示  } PPMPixel;  typedef struct {  unsigned int width, height; // 图像的宽度和高度（以像素为单位）  PPMPixel \*data; // 构成图像的像素  } PPMImage; |
| void WritePPM(char\* output,PPMImage\* img) | PPM图像保存， output是输出的PPM图像文件名，img是输入的图像数据。支持P6格式的PPM图像。  需要引入的结构体：  typedef struct {  unsigned char red, green, blue; //像素的颜色由RGB（红/绿/蓝）表示  } PPMPixel;  typedef struct {  unsigned int width, height; // 图像的宽度和高度（以像素为单位）  PPMPixel \*data; // 构成图像的像素  } PPMImage; |
| void InvertColor(char\* input,char\* output) | 负滤波器，input是输入文件名，output是输出文件名。支持P6格式的PPM图像。 |
| void GrayFilter(char\* input,char\* output) | 灰度过滤器，input是输入文件名，output是输出文件名。支持P6格式的PPM图像。 |
| void SepiaFilter(char\* input,char\* output) | 乌贼墨过滤器，input是输入文件名，output是输出文件名。支持P6格式的PPM图像。 |
| void AdjustSaturation(char\* input,char\* output,double a) | 调整图像饱和度，input是输入文件名，output是输出文件名。a是目标饱和度，如a=30。支持P6格式的PPM图像。 |
| void Resize(char\* input,char\* output,unsigned int NewWidth, unsigned int NewHeight) | 调整图像大小，input是输入文件名，output是输出文件名。NewWidth和NewHeight分别是输出图像的宽和高。支持P6格式的PPM图像。 |
| void AdjustHue(char\* input,char\* output,int a) | 调整图像的色调，input是输入文件名，output是输出文件名。a是目标色调，如a=125。支持P6格式的PPM图像。 |
| void AdjustBrightness(char\* input,char\* output,double a) | 调整图像亮度，input是输入文件名，output是输出文件名。a是目标亮度，如a=60。支持P6格式的PPM图像。 |
| void AdjustContrast(char\* input,char\* output,double a) | 调整图像对比度，input是输入文件名，output是输出文件名。a是目标对比度，如a=60。支持P6格式的PPM图像。 |
| void AdjustBlur(char\* input,char\* output,double a) | 通过sigma因子模糊图像，input是输入文件名，output是输出文件名。a是sigma因子，如a=5。支持P6格式的PPM图像。 |
| void MeanGrayFilter(char\* input,char\* output,double a) | 平均灰度滤波器，input是输入文件名，output是输出文件名。a是平均值系数，如a=3。支持P6格式的PPM图像。 |
| void Pixelate(char\* input,char\* output,unsigned int a) | 像素化，input是输入文件名，output是输出文件名。a是幅度值，如a=8。支持P6格式的PPM图像。 |
| void Rotate(char\* input,char\* output,short a) | 旋转图像，input是输入文件名，output是输出文件名。a是旋转的角度，如a=45。支持P6格式的PPM图像。 |
| void GammaCorrection(char\* input,char\* output,double a) | 伽马校正，input是输入文件名，output是输出文件名。a是gamma数，如a=0.5。支持P6格式的PPM图像。 |
| void GrayAndChannelSeparation(char\* input,char\* Grayoutput,char\* Routput,char\* Goutput,char\* Boutput) | 生成灰度图以及RGB通道分离，input是输入的P6格式的PPM图像；Grayoutput是输出的灰度图文件名，Routput、Goutput和Boutput分别是输出的R、G和B通道的图像文件名，输出都是PGM格式文件。 |
| void PGMBin(char\* input,char\* output,int threshold) | 灰度图像二值化，输入是灰度图像，输入和输出都是PGM文件，threshold是阈值，如threshold=125。 |
| void Brightening(char\* input,char\* output,int a) | 彩色图像增亮，输入和输出都是P6格式的PPM图像，a是增亮系数，如a=80。 |
| void GrayBrightening(char\* input,char\* output,int a) | 灰度图像增亮，输入和输出都是PGM图像，a是增亮系数，如a=80。 |
| void PPMFilter(char\* input,char\* output) | 彩色图像滤波，输入和输出都是P6格式的PPM文件。 |
| void PGMGrayFilter(char\* input,char\* output) | 灰度图像滤波，输入和输出都是PGM图像。 |
| void PPMtoBMP(char\* input,char\* output) | PPM图像转BMP图像，input是输入文件名，output是输出文件名。支持P6格式的PPM图像。 |
| void YFiltering(char\* input,char\* output,int sobel\_x[3][3],int sobel\_y[3][3]) | Y方向滤波，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。  参考模板：  int sobel\_x[3][3] = { { 1, 0, -1},  { 2, 0, -2},  { 1, 0, -1}};  int sobel\_y[3][3] = { { 1, 2, 1},  { 0, 0, 0},  {-1, -2, -1}}; |
| void XFiltering(char\* input,char\* output,int sobel\_x[3][3],int sobel\_y[3][3]) | X方向滤波，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。  参考模板：  int sobel\_x[3][3] = { { 1, 0, -1},  { 2, 0, -2},  { 1, 0, -1}};  int sobel\_y[3][3] = { { 1, 2, 1},  { 0, 0, 0},  {-1, -2, -1}}; |
| void SobelFiltering(char\* input,char\* output,int sobel\_x[3][3],int sobel\_y[3][3]) | Sobel算子，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。  参考模板：  int sobel\_x[3][3] = { { 1, 0, -1},  { 2, 0, -2},  { 1, 0, -1}};  int sobel\_y[3][3] = { { 1, 2, 1},  { 0, 0, 0},  {-1, -2, -1}}; |
| void PrewittFiltering(char\* input,char\* output,int prewitt\_x[3][3],int prewitt\_y[3][3]) | Prewitt算子，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。  参考模板：  int prewitt\_x[3][3] = { { 5, 5, 5},  { -3, 0, -3},  { -3, -3, -3}};  int prewitt\_y[3][3] = { { 5, -3, -3},  { 5, 0, -3},  {5, -3, -3}}; |
| void LaplacianFiltering(char\* input,char\* output,int laplacian[3][3]) | Laplace算子，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。laplacian是Laplacian算子模板。  参考模板：  int laplacian[3][3] = { { 1, 1, 1},  { 1, -8, 1},  { 1, 1, 1}}; |
| void PGMOtsuThreshold(string input,char\* output) | 大津阈值法，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。 |
| void PGMLocalisedOtsuThreshold(string input,char\* output) | 局部大津阈值，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。 |
| void PGMSauvolaThreshold(string input,char\* output,double a,double b,double c) | 索沃拉阈值，input是输入文件名，output是输出文件名。a、b和c的参考值如：a=0.01,b=15,c=225。支持P5格式的PGM图像。 |
| void PGMThreshold(string input,char\* output,int thresh) | 阈值法，input是输入文件名，output是输出文件名。支持P5格式的PGM图像。thresh是阈值，如：thresh=5。 |

**RAW图像处理**

|  |  |
| --- | --- |
| void RAWSobelEdge(char\* input,char\* output,int ROWS,int COLS,int M,float sobelX[3][3],float sobelY[3][3]) | Sobel算子，input是输入文件名，output是输出文件名。ROWS是图像的行，COLS是图像的列，M是滤波相关参数，如M=1。支持RAW图像。  参考模板：  float sobelX[3][3] = {{-1,0,1},  {-2,0,2},  {-1,0,1}};  float sobelY[3][3] = {{-1,-2,-1},  {0,0,0},  {1,2,1}}; |
| void RAWPlaceHolder(char\* input,char\* output,int ROWS,int COLS,int M,float mask[3][3]) | 边缘检测，input是输入文件名，output是输出文件名。ROWS是图像的行，COLS是图像的列，M是滤波相关参数，如M=1。支持RAW图像。  参考模板：  float mask[3][3] = {{-1,-2,-1},  {0,0,0},  {1,2,1}}; |
| void RAWLaplacialSharpeningFilter(char\* input,char\* output,int ROWS,int COLS,int M,float w,float mask[3][3]) | 拉普拉斯锐化滤波器，input是输入文件名，output是输出文件名。ROWS是图像的行大小，COLS是图像的列大小，M和w是滤波相关参数，如M=1，w=1；mask是滤波器模板。支持RAW图像。  参考模板：  float mask[3][3] = {{0,1,0},  {1,-4,1},  {0,1,0}}; |
| void RawLaplacianEnhancement(char\* input1,char\* output1,int width,int height) | 拉普拉斯算子增强，input1是输入的RAW图像文件名，output1是输出的RAW图像文件名，width是输入图像的宽，height是输入图像的高。支持RAW图像。 |
| void RawPowerTransformation(char\* input,char\* output,int width,int height,int c,float v) | 幂次变换，input是输入的RAW图像文件名，output是输出的RAW图像文件名，width是输入图像的宽，height是输入图像的高。默认c=1，v=0.6。支持RAW图像。 |
| void RAWAvgFilter(char\* input,char\* output,int ROWS,int COLS,int M,float mask[3][3]) | 平均滤波器，input是输入文件名，output是输出文件名。ROWS是图像的行大小，COLS是图像的列大小，M是滤波相关参数，如M=1；mask是滤波器模板。支持RAW图像。  参考模板：  float mask[3][3] = {{0.1111,0.1111,0.1111},  {0.1111,0.1111,0.1111},  {0.1111,0.1111,0.1111}}; |
| void RawImageInversion(char\* input,char\* output,int width,int height) | 图像反相，input是输入的RAW图像文件名，output是输出的RAW图像文件名，width是输入图像的宽，height是输入图像的高。支持RAW图像。 |
| void RawHistogramEqualization(char\* input,char\* output,int width,int height) | 直方图均衡化，input是输入的RAW图像文件名，output是输出的RAW图像文件名，width是输入图像的宽，height是输入图像的高。支持RAW图像。 |
| void RAWMedianFilter(char\* input,char\* output,int ROWS,int COLS,int M,int sequence[9]) | 中值滤波，input是输入文件名，output是输出文件名。ROWS是图像的行，COLS是图像的列，M是滤波相关参数，如M=1。支持RAW图像。  参考模板：  int sequence[9]={0,0,0,0,0,0,0,0,0}; |
| void RawtoBmp1(char\* input, char\* output,unsigned long Width, unsigned long Height) | RAW图像转为BMP图像，input是输入文件名，output是输出文件名。Width和Height是输入文件的宽和高。 |
| void RawToBmp(char\* input,char\* output,int imageWidth,int imageHigth) | RAW图像转为BMP图像，input是输入文件名，output是输出文件名。支持宽和高相等的图像。 |
| void RGBtoHSI(char\* input,char\* output) | RGB模型转为HIS模型，input是输入文件名，output是输出文件名。支持24位BMP图像。 |

**BMP图像处理**

|  |  |
| --- | --- |
| void DCMtoBMP(string input,char\* output) | DCM图像转BMP图像。input是输入文件名，output是输出文件名。 |
| void SobelOperator(char\* input,char\* output) | Sobel算子，耗时较长，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void ClosedOperation(char\* input,char\* output) | 闭运算，input是输入文件名，output是输出文件名。支持4位BMP图像。 |
| void EdgeDetection(char\* input,char\* output) | 边缘检测，input是输入文件名，output是输出文件名。支持4位BMP图像。 |
| void EdgeDetection1(char\* input,char\* output,short sharpen[3][3]) | 边缘检测，input是输入文件名，output是输出文件名。支持8位BMP图像。  参考模板：  short sharpen[3][3] = {{1, 1, 1},  {1, -8, 1},  {1, 1, 1}}; |
| void AdjustPixel(char\* input,char\* output,int a) | 调整像素值，input是输入文件名，output是输出文件名。a是用于设置图像像素的相关参数，如a=3。支持24位BMP图像。 |
| void EdgeDetection2(char\* input,char\* output,int a) | 边缘检测，input是输入文件名，output是输出文件名。a是用于设置图像像素的相关参数，如a=3。支持24位BMP图像。 |
| void EdgeDetection3(char\* input,char\* output,int a) | 边缘检测，input是输入文件名，output是输出文件名。a是用于设置图像像素的相关参数，如a=3。支持24位BMP图像。 |
| void EdgeDetection4(char\* input,char\* output,int a) | 边缘检测，input是输入文件名，output是输出文件名。a是用于设置图像像素的相关参数，如a=3。支持24位BMP图像。 |
| void GrayImageConversion8(char\* input,char\* output) | 生成图像的灰度图，支持8位BMP图像。input是输入文件名，output是输出文件名。 |
| void Gray(char\* input,char\* output) | 灰度图转换，支持24位BMP图像。input是输入文件名，output是输出文件名。 |
| void GrayImageConversion(char\* input,char\* output) | 彩色图转灰度图，input是要处理的彩色图像，output是处理后生成的灰度图名称。支持24位BMP图像。 |
| void OTSU(char\* input,char\* output,int BeforeThreshold) | 大津算法，input是输入文件名，output是输出文件名。BeforeThreshold是初始阈值，如BeforeThreshold=10。支持8位BMP图像。 |
| void LowerBrightness(char\* input,char\* output,int a,int b) | 调低亮度，input是输入文件名，output是输出文件名。支持24位BMP图像。a和b的参考值可为a=100，b=0。 |
| void HightBrightness(char\* input,char\* output,int a,int b) | 调高亮度，input是输入文件名，output是输出文件名。支持24位BMP图像。a和b的参考值可为a=100，b=0。 |
| void IterativeThresholdSelection(char\* input,char\* output) | 迭代阈值选择，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void DitheringMethod(char\* input,char\* output) | 抖动法，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void LogTransformation(char\* input,char\* output,int constant) | 对数变换，input是输入文件名，output是输出文件名。支持8位BMP图像。constant是相关参数，如constant=15。 |
| void LogarithmicTransformation(char\* input,char\* output) | 对数变换，input是输入文件名，output是输出文件名。支持BMP图像。 |
| void HistogramEqualization(char\* input,char\* output) | 直方图均衡化，input是输入文件名，output是输出文件名。支持BMP图像。 |
| void Binarization(char\* input,char\* output,int threshold) | 二值化，input是输入文件名，output是输出文件名。支持24位BMP图像。threshold是阈值，如：threshold=128。 |
| void LowPassFilter(char\* input,char\* output) | 低通滤波器，input是输入文件名，output是输出文件名。支持BMP图像。 |
| void HighPassFilter(char\* input,char\* output) | 高通滤波器，input是输入文件名，output是输出文件名。支持BMP图像。 |
| void Thinning(char\* input,char\* output) | 图像细化，input是输入文件名，output是输出文件名。支持BMP图像。 |
| void ThinningLine(char\* input,char\* output) | 图像细化且线条化，input是输入文件名，output是输出文件名。支持BMP图像。 |
| void Corrosion(char\* input,char\* output) | 腐蚀，input是输入文件名，output是输出文件名。支持4位BMP图像。 |
| void Corrosion1(char\* input,char\* output,int \*TempBuf, int TempH, int TempW) | 腐蚀，input是输入文件名，output是输出文件名。支持24位BMP图像。TempBuf是腐蚀模板，TempH和TempW分别是TempBuf的高和宽，如TempH=4,TempW=4，则有TempBuf[4][4]。 |
| void Expand(char\* input,char\* output,int \*TempBuf, int TempH, int TempW) | 膨胀，input是输入文件名，output是输出文件名。支持24位BMP图像。TempBuf是膨胀模板，TempH和TempW分别是TempBuf的高和宽，如TempH=4,TempW=4，则有TempBuf[4][4]。 |
| void GaussianFiltering(char\* input,char\* output) | 高斯滤波，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void LaplaceEnhancement(char\* input,char\* output) | 拉普拉斯增强，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void Residual(char\* input,char\* output) | 求残差，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void Compress(char\* input,char\* output) | 压缩，支持多种文件。input是要压缩的文件名，output是压缩后的文件名。 |
| void Decompression(char\* input, char\* output) | 解压缩，支持多种文件。input是要解压缩的文件名，output是解压缩后的文件名。 |
| void BlackWhite(char\* input,char\* output) | 黑白化，input是输入的原图像，output是输出的黑白图像。支持24位BMP图像。 |
| void Underexposure(char\* input,char\* output) | 图像欠曝光，input是输入的原图像，output是输出的欠曝光图像。支持24位BMP图像。 |
| void Overexposure(char\* input,char\* output) | 图像过曝光，input是输入的原图像，output是输出的过曝光图像。支持24位BMP图像。 |
| void GammaTransform(char\* input,char\* output) | 伽马变换，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void GrayScale(char\* input,char\* output) | 灰度化，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void GrayImageBinarization(char\* input,char\* output,int bit,int threshold) | 灰度图二值化，bit用于设定位数，如bit=8；threshold是阈值，如threshold=200。支持8位BMP图像。 |
| void GreyPesudoColor(char\* input,char\* output) | 灰度图伪彩色化，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void CalculateCumulativeHistogramMap(char\* input,char\* outfile) | 计算累加直方图并映射，input是输入文件名，outfile是输出文件名。支持24位BMP图像。 |
| void Translation(string input,char\* output,int dx,int dy) | 图像平移，input是输入的文件，dx和dy是横向及纵向的移动距离（像素），负值是向左 / 向下移动；output是平移操作后的结果文件名。支持BMP图像。 |
| void Mirrored(string input,char\* output,char axis) | 镜像变换，input是输入的文件，output是镜像操作后的结果文件名，axis是镜像变换的方向（以X或Y表示）。支持BMP图像。 |
| void Sheared(string input,char\* output,char axis,double Coef) | 错切变换，input是输入的文件，output是错切操作后的结果文件名，axis和Coef分别是错切变换的方向（以X或Y表示）和错切系数，负值是向左 / 向下偏移。支持BMP图像。 |
| void Scaled(string input,char\* output,double cx,double cy) | 缩放操作，input是输入的文件，output是缩放操作后的结果文件名，cx和cy分别是横向及纵向的缩放系数，系数大于1表示拉伸，小于1表示压缩。支持BMP图像。 |
| void Rotated1(string input,char\* output,double angle) | 图像旋转，input是输入的文件，output是图像旋转后的结果文件名，angle是旋转角度，弧度制。支持BMP图像。 |
| void SaltNoise(char\* input,char\* output,int a,int b,int c,int d) | 添加椒盐噪声，a和b是噪声相关参数，如a=3，b=3；c和d是颜色相关参数，如c=0，d=255。支持8位BMP图像 |
| void MeanFiltering(char\* input,char\* output) | 均值滤波，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void MeanFltering1(char\* input,char\* output) | 均值滤波，input是输入文件名，output是输出文件名。支持8位和24位BMP图像。 |
| void KapoorAlgorithm(char\* input,char\* output,int BeforeThreshold) | 卡普尔算法，input是输入文件名，output是输出文件名。BeforeThreshold是初始阈值，如BeforeThreshold=150。支持8位BMP图像。 |
| void OpenOperation(char\* input,char\* output) | 开运算，input是输入文件名，output是输出文件名。支持4位BMP图像。 |
| void LapulasFiltering(char\* readPath,char\* writePath,float CoefArray[9],float coef) | 拉普拉斯滤波，readPath是原图像，writePath是处理后的图像文件名。支持8位BMP图像。  各参数参考值：  定义\*3的模板（拉普拉斯）：  float CoefArray[9]={1.0f,2.0f,1.0f,2.0f,4.0f,2.0f,1.0f,2.0f,1.0f};  定义模板前乘的系数（拉普拉斯）：  float coef=(float)(1.0/16.0); |
| void ImageFiltering(char\* input,char\* output,float kernel[3][3]) | 图像滤波，input是输入文件名，output是输出文件名。kernel是模糊内核。支持24位BMP图像。 |
| void LinearFiltering(char\* input,char\* output,short average[3][3]) | 线性滤波，input是输入文件名，output是输出文件名。支持8位BMP图像。  参考模板：  short average[3][3] = {{1, 2, 1},  {2, 4, 2},  {1, 2, 1}}; |
| void MedianFiltering(char\* input,char\* output,short average[3][3]) | 中值滤波，input是输入文件名，output是输出文件名。支持8位BMP图像。  参考模板：  short average[3][3] = {{1, 2, 1},  {2, 4, 2},  {1, 2, 1}}; |
| void SharpeningFiltering(char\* input,char\* output,short average[3][3],short sharpen[3][3]) | 锐化滤波，input是输入文件名，output是输出文件名。支持8位BMP图像。  参考模板：  short average[3][3] = {{1, 2, 1},  {2, 4, 2},  {1, 2, 1}};  short sharpen[3][3] = {{-1, -1, -1},  {-1, 8, -1},  {-1, -1, -1}}; |
| void GradientSharpening(char\* input,char\* output,short average[3][3],short soble1[3][3],short soble2[3][3]) | 梯度锐化，input是输入文件名，output是输出文件名。支持8位BMP图像。  参考模板：  short average[3][3] = {{1, 2, 1},  {2, 4, 2},  {1, 2, 1}};  short soble1[3][3] = {{-1, -2, -1},  {0, 0, 0},  {1, 2, 1}};  short soble2[3][3] = {{-1, 0, 1},  {-2, 0, 2},  {-1, 0, 1}}; |
| void ArithmeticMeanFilter(char\* input,char\* output) | 算术平均滤波器，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void GeometricMeanFilter(char\* input,char\* output) | 几何平均滤波器，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void HarmonicMeanFilter(char\* input,char\* output) | 调和平均滤波器，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void ContraHarmonicMeanFilter(char\* input,char\* output) | 反调和平均滤波器，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void Filter(char\* input,char\* output) | 滤波，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void Mosaic(char\* input,char\* output,int x) | 马赛克化图像，input是输入文件名，output是输出文件名。x是马赛克处理的块的大小。支持24位BMP图像 |
| void Expansion(char\* input,char\* output) | 膨胀，input是输入文件名，output是输出文件名。支持4位BMP图像。 |
| void SmoothSharpen(char\* input,char\* output,int Template[3][3],int coefficient) | 平滑，input是输入文件名，output是输出文件名。Template是平滑模板，均一化处理，coefficient1 = 9。支持24位BMP图像。 |
| void GaussSmoothSharpen(char\* input,char\* output,int Template[3][3],int coefficient) | 高斯平滑，input是输入文件名，output是输出文件名。Template是高斯平滑模板，coefficient=16。支持24位BMP图像。 |
| void SobelSharpen(char\* input,char\* output,int Templatex[3][3],int Templatey[3][3],int coefficient1,int coefficient2) | Sobel算子，input是输入文件名，output是输出文件名。Templatex是laplace锐化模板,4邻域，Templatey是laplace锐化模板,8邻域，coefficient1 = 9，coefficient2 = 16。支持24位BMP图像。 |
| void MidSmoothing(char\* input,char\* output) | 中值滤波器，input是输入文件名，output是输出文件名。支持8位BMP图像 |
| void AvgSmoothing(char\* input,char\* output) | 均值滤波器，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void Averaging(char\* input1,char\* input2,char\* input3,char\* output,int a) | 图像平均化，input是输入文件名，output是输出文件名。a是平均化相关参数，如a=3。支持8位BMP图像。 |
| void PlaneSlicing(char\* input,char\* output) | 平面切片，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void SharpeningSpatialFiltering8(char\* input,char\* output,int model[9]) | 锐化空间滤波器，input是输入文件名，output是输出文件名。model是锐化模板。支持8位灰度图像。 |
| void PseudoGrayscale(char\* input,char\* output) | 伪灰度化，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void TwoColors(char\* input,char\* output,int threshold,unsigned char color1,unsigned char color2) | 二色化，input是输入文件名，output是输出文件名。threshold是阈值，如threshold=115；color1和color2是要填充的两个颜色。支持24位BMP图像。 |
| void PNGImageGeneration(char\* filename,const unsigned char img[],unsigned W,unsigned H,int x) | filename是生成的PNG图像文件名；img是图像的像素数据，W是图像的宽，H是图像的高，x=0选择生成RGB图像，x=1选择生成RGBA图像。 |
| void BilateralFiltering(string input,char\* output,double ssd, double sdid) | 双边滤波，input是输入文件名，output是输出文件名。支持24位BMP图像。ssd和sdid分别是空间域标准差与强度域标准差。 |
| int Equal(char\* input1,char\* input2,double c) | 若比对图像的梯度幅相似性偏差值等于c则通过。input1和input2是要比对的两个图像。c是参考的阈值。支持24位BMP图像。 |
| int GreaterThan(char\* input1,char\* input2,double c) | 若比对图像的梯度幅相似性偏差值大于c则通过。input1和input2是要比对的两个图像。c是参考的阈值。支持24位BMP图像。 |
| int LessThan(char\* input1,char\* input2,double c) | 若比对图像的梯度幅相似性偏差值小于c则通过。input1和input2是要比对的两个图像。c是参考的阈值。支持24位BMP图像。 |
| double GMSD(char\* input1, char\* input2) | 求两幅图像的梯度幅相似性偏差值并返回结果。input1和input2是要比对的两个图像。支持24位BMP图像。 |
| void AddGaussNoise(char\* input,char\* output) | 添加高斯噪声，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void AddSaltPepperNoise(char\* input,char\* output) | 添加椒盐噪声，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void ChannelSeparation(char\* input,char\* Routput,char\* Goutput,char\* Boutput) | 通道分离，input是输入文件名，Routput是红色通道图像，Goutput是绿色通道图像，Boutput是绿色通道图像。支持24位BMP图像。 |
| void PatternMethod(char\* input,char\* output,unsigned char Template[8][8]) | 图案法，input是输入文件名，output是输出文件名。Template是模板数组。支持8位BMP图像。 |
| void BMP24LossyCompression(char\* input,char\* output) | 图像有损压缩，input是待压缩的BMP文件名，output是有损压缩后输出的文件名。支持24位BMP图像。 |
| void BMP24LossyDecompression(char\* input,char\* output) | 图像有损解压，input是待解压的文件名，output是输出解压后的BMP文件名。支持24位BMP图像。 |
| void BMP24LosslessCompression(char\* input,char\* output) | 图像无损压缩，input是待压缩的BMP文件名，output是无损压缩后输出的文件名。支持24位BMP图像。 |
| void BMP24LosslessDecompression(char\* input,char\* output) | 图像无损解压，input是待解压的文件名，output是输出解压后的BMP文件名。支持24位BMP图像。 |
| void ImageDiscoloration(char\* input,char\* output,double a,double b,double c) | 图像变色，input是输入文件名，output是输出文件名。如：a=0.2126，b=0.7152, c=0.0722。支持24位BMP图像。 |
| void ImageCutting(char\* input,char\* output,int leftdownx,int leftdowny,int rightupx,int rightupy) | 图像裁剪，input是输入文件名，output是输出文件名。leftdownx, leftdowny, rightupx, rightupy是要裁剪的矩形区域的左下角和右上角的坐标（连续四个整数值，如50 50 300 300)。支持24位BMP图像。 |
| void RGBtoGraywithoutLUT(char\* input,char\* output) | 图像无LUT的灰度化，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void RGBtoGraywithLUT(char\* input,char\* output) | 图像有LUT的灰度化，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void PiecewiseLinearTransform(char\* input,char\* output) | 分段线性变换，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void PowerConvertion(char\* input,char\* output,double c,double g) | 功率转换，input是输入文件名，output是输出文件名。如: c = 1.2，g = 0.5。支持8位BMP图像。 |
| void LaplacianEnhancement(char\* input,char\* output,int N,int LaplMask[3][3]) | 拉普拉斯图像增强，input是输入文件名，output是输出文件名。如：N=1。支持8位BMP图像。  参考模板：  int LaplMask[3][3] = {  0,1,0,  1, -4, 1,  0, 1, 0  }; |
| void Smooth(char\* input,char\* output) | 平滑，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void LaplaceSmooth(char\* input,char\* output,int N,int LaplMask[3][3]) | 拉普拉斯平滑，input是输入文件名，output是输出文件名。如：N=1。支持8位BMP图像。  参考模板：  int LaplMask[3][3] = {  0,1,0,  1, -4, 1,  0, 1, 0  }; |
| void Sobel1(char\* input,char\* output,int N,int SblMask1[3][3],int SblMask2[3][3]) | Sobel算子，input是输入文件名，output是输出文件名。如：N=1。支持8位BMP图像。  参考模板：  int SblMask1[3][3] = {  -1,-2,-1,  0, 0, 0,  1, 2, 1  };  int SblMask2[3][3] = {  -1, 0, 1,  -2, 0, 2,  -1, 0, 1  }; |
| void SobelSmooth(char\* input,char\* output,int N,int SblMask1[3][3],int SblMask2[3][3]) | Sobel平滑，input是输入文件名，output是输出文件名。如：N=1。支持8位BMP图像。  参考模板：  int SblMask1[3][3] = {  -1,-2,-1,  0, 0, 0,  1, 2, 1  };  int SblMask2[3][3] = {  -1, 0, 1,  -2, 0, 2,  -1, 0, 1  }; |
| void Multiply(char\* input,char\* output,int N,int SblMask1[3][3],int SblMask2[3][3],int LaplMask[3][3]) | 图像倍增化，input是输入文件名，output是输出文件名。如：N=1。支持8位BMP图像。  参考模板：  int LaplMask[3][3] = {  0,1,0,  1, -4, 1,  0, 1, 0  };  int SblMask1[3][3] = {  -1,-2,-1,  0, 0, 0,  1, 2, 1  };  int SblMask2[3][3] = {  -1, 0, 1,  -2, 0, 2,  -1, 0, 1  }; |
| void Add(char\* input,char\* output,int N,int SblMask1[3][3],int SblMask2[3][3],int LaplMask[3][3]) | 图像添加，input是输入文件名，output是输出文件名。如：N=1。支持8位BMP图像。  参考模板：  int LaplMask[3][3] = {  0,1,0,  1, -4, 1,  0, 1, 0  };  int SblMask1[3][3] = {  -1,-2,-1,  0, 0, 0,  1, 2, 1  };  int SblMask2[3][3] = {  -1, 0, 1,  -2, 0, 2,  -1, 0, 1  }; |
| void PowerConvertion1(char\* input,char\* output,double c,double g,int N,int SblMask1[3][3],int SblMask2[3][3],int LaplMask[3][3]) | 功率变换，input是输入文件名，output是输出文件名。如: c = 1.2，g = 0.5，N=1。支持8位BMP图像。  int LaplMask[3][3] = {  0,1,0,  1, -4, 1,  0, 1, 0  };  int SblMask1[3][3] = {  -1,-2,-1,  0, 0, 0,  1, 2, 1  };  int SblMask2[3][3] = {  -1, 0, 1,  -2, 0, 2,  -1, 0, 1  }; |
| void BlackWhite(char\* input,char\* output) | 黑白化图像，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void RandomOperation(char\* input,char\* output,unsigned char treshold1,unsigned char treshold2,unsigned char treshold3,unsigned char treshold4,unsigned char treshold5,unsigned char treshold6,unsigned char red,unsigned char green,unsigned char blue,int color1,int color2,int color3,int color4,int color5,int color6,int color7,int color8) | 随意操作，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void SpecialEffects1(char\* input,char\* output,unsigned char red,unsigned char green,unsigned char blue) | 图像特效，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void GrayAVS(char\* input,char\* output,float k,float b) | input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void HistogramEqualize24(char\* input,char\* output) | 直方图均衡化，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void AvrFilter(char\* input,char\* output1,char\* output2,int M,int N) | input是输入文件名，output是输出文件名。如M=21，N=1。支持8位BMP图像。 |
| void GryOppositionSSE(char\* input,char\* output) | input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void MedianFilter(char\* input,char\* output,int M,int N) | 中值滤波器，input是输入文件名，output是输出文件名。如M=5，N=5。支持8位BMP图像。 |
| void EdgeSharpeningGry(char\* input,char\* output) | input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void SJGryandRiceTest(char\* input,char\* output) | input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void TextTest(char\* input,char\* output) | input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void RedChannel(char\* input,char\* output) | 生成图像的红色通道图像，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void GreenChannel(char\* input,char\* output) | 生成图像的绿色通道图像，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void BlueChannel(char\* input,char\* output) | 生成图像的蓝色通道图像，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void HistogramStatistics(char\* input,char\* output) | 直方图统计，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void HistogramEqualization1(char\* input,char\* output) | 直方图均衡化，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void ReflectionRay(char\* input,char\* output) | 反射线，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void MeanFiltering24(char\* input,char\* output) | 均值滤波，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void MedianFiltering24(char\* input,char\* output) | 中值滤波，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void ZoomOutAndZoomIn(char\* input,char\* output,double value) | 缩放（双线性插值），input是输入文件名，output是输出文件名。value是放大倍数，如value=0.5。支持24位BMP图像。 |
| void Translation24(char\* input,char\* output,int x,int y) | 平移，input是输入文件名，output是输出文件名。x是横轴的平移量，y是纵轴的平移量，如x=-10，y=-30。支持24位BMP图像。 |
| void Mirror24(char\* input,char\* output) | 镜像，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void Rotate24(char\* input,char\* output,double degree) | 旋转，input是输入文件名，output是输出文件名。degree是旋转的度数。支持24位BMP图像。 |
| void GivenThresholdMethod(char\* input,char\* output,int threshold) | 给定阈值法处理图像，使图片黑白化，input是输入文件名，output是输出文件名。threshold是给定的阈值，如threshold=100。支持24位BMP图像。 |
| void IterativeThresholdMethod(char\* input,char\* output) | 迭代阈值法处理图像，使图片黑白化，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void OstuThresholdSegmentationMethod(char\* input,char\* output) | Ostu（大津法）阈值分割，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void Repudiation(char\* input,char\* output) | 将伪彩图片反白，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void Gray1(char\* input,char\* output) | 将彩色图片变成灰度图片，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void CorrectMethod(char\* input,char\* output) | 正确法，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void ChannelSeparation1(char\* input,char\* Routput,char\* Goutput,char\* Boutput) | 对图像分理出其中的RGB分量并分别保存为独立的图像，input是输入文件名，Routput是红色通道图像，Goutput是绿色通道图像，Boutput是绿色通道图像。支持24位BMP图像。 |
| void ReverseColor(char\* input,char\* output) | 对灰度图进行反色，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| Image\* LoadImage1(char\* input) | BMP图像读取，input是输入文件名。支持8位和24位BMP图像。  返回Image型数据，Image型数据的结构如下：  typedef struct  {  int width;  int height;  int channels; //图像通道数  unsigned char\* Data; //像素数据  }Image; |
| void SaveImage1(char\* output,Image\* img) | 将Image型数据保存为BMP图像，output是生成的BMP图像文件名，img是要保存的图像数据。支持8位和24位BMP图像。  Image型数据的结构如下：  typedef struct  {  int width;  int height;  int channels; //图像通道数  unsigned char\* Data; //像素数据  }Image; |
| void ImageContrastExtension(char\* input,char\* output,double m,double g1,double g2,double a) | 图像对比度扩展，input是输入文件名，output是输出文件名。  其中，可参考：double m=1.5,g1=100.0,g2=200.0; m对应斜率  double a=(255.0-m\*(g2-g1))/(255.0-(g2-g1));  支持8位BMP图像。 |
| void Binaryzation(char\* input,char\* output,int threshold) | 图像二值化，input是输入文件名，output是输出文件名。threshold是将灰度值转化为二值的阈值，如threshold=80。支持24位BMP图像。 |
| void GlobalBinarization(char\* input,char\* output) | 全局二值化，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void AdaptiveBinarization(char\* input,char\* output) | 自适应二值化，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void ExpansionOperation(char\* input,char\* output) | 膨胀操作，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void CorrosionOperation(char\* input,char\* output) | 腐蚀操作，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void Operation1(char\* input,char\* output) | 开操作，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void Closed1(char\* input,char\* output) | 闭操作，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void Negative1(char\* input,char\* output) | 图像反色，input是输入文件名，output是输出文件名。支持24位BMP图像。 |
| void Negative(char\* input,char\* output) | 图像反色，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void ImageEncryption(char\* inFileName,char\* outFileName,char key) | 图像加密，inFileName是原图图像文件名，outFileName是解密图像文件名，key是密钥，如key=255。支持8位、24位和32位BMP图像。 |
| void ImageDecryption(char\* inFileName,char\* outFileName,char key) | 图像解密，inFileName是加密图像文件名，outFileName是解密图像文件名，key是密钥，如key=255。支持8位、24位和32位BMP图像。 |
| void EncryptionDecryption(char\* input,char\* output,int Key,int a) | 图像加解密，Key是密钥，a=1时执行加密，a=0时执行解密。支持24位BMP图像。 |
| void Encryption(char\* input,char\* output,int Key) | 图像加密，input是输入文件名，output是输出文件名。Key是密钥。支持24位BMP图像。 |
| void Decryption(char\* input,char\* output,int Key) | 图像解密，input是输入文件名，output是输出文件名。Key是密钥。支持24位BMP图像。 |
| void Compress8(string input,string output) | 图像压缩，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void Decompression(string input,string output) | 图像解压，input是输入文件名，output是输出文件名。支持8位BMP图像压缩后的结果文件。 |
| void HorizontalMirror(char\* input,char\* output) | 水平镜像，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void MirrorVertically(char\* input,char\* output) | 垂直镜像，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void XMirroring(char\* input,char\* output) | X镜像，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void YMirroring(char\* input,char\* output) | Y镜像，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void ImageConvolution(char\* input,char\* output,double\*\* Kernel,int n,int m) | 图像卷积，input是输入文件名，output是输出文件名。Kernel是卷积核，如double Kernel[3][3] = {{-0.225,-0.225-0.225},{-0.225,1,-0.225},{-0.225,-0.225,-0.225}}；n是Kernel的第一维的大小，m是Kernel的第二维的大小，形如Kernel[n][m]。支持24位BMP图像。 |
| void Translation(char\* input,char\* output,int x,int y,unsigned char color) | 图像平移，input是输入文件名，output是输出文件名。x和y是在X轴和Y轴平移的量，以右为正向，color是平移后非原图区域填充的颜色，如color=100。支持8位BMP图像。 |
| void Nesting(char\* Biginput,char\* Smallinput,char\* output) | 图像嵌套，Biginput是输入的大图，Smallinput是输入的小图。支持24位BMP图像。 |
| void Blend(char\* input1,char\* input2,char\* output) | 图像融合之混合化，input1和input2是输入的两个要融合的图像，output是输出文件名。支持24位BMP图像。 |
| void Checker(char\* input1,char\* input2,char\* output) | 图像融合之棋盘化，input1和input2是输入的两个要融合的图像，output是输出文件名。支持24位BMP图像。 |
| void Blend1(char\* input1,char\* input2,char\* output) | 图像融合之混合化，input1和input2是输入的两个要融合的图像，output是输出文件名。支持24位BMP图像。 |
| void Checker1(char\* input1,char\* input2,char\* output) | 图像融合之棋盘化，input1和input2是输入的两个要融合的图像，output是输出文件名。支持24位BMP图像。 |
| void ImageSharpening(char\* input,char\* output) | 图像锐化，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void DrawRectangle(char\* input,char\* output,int x1,int y1,int x2,int y2,unsigned char red,unsigned char green,unsigned char blue) | 在24位BMP图像上通过传入的参数画一个矩形。input是输入文件名，output是输出文件名。(x1,y1)是矩形坐上顶点的坐标，(x2,y2)是矩形右下顶点的坐标；red是矩形线框的红色分量，green是矩形线框的绿色分量，blue是矩形的蓝色分量。 |
| void GenerateBmp(unsigned char\* pData,int width,int height,char\* filename) | 生成BMP图像，pData是图像的像素数据，width和height是图像的宽和高，filename是生成的图像的文件名。 |
| void Jpg24ImageGeneration(char\* filename,unsigned int width, unsigned int height, unsigned char\* img) | JPG图像生成，filename是生成的JPG图像文件名，width是图像的宽，height是图像的高，img是图像的像素数据。 |
| void ImageScalingNearestNeighborInterpolation(char\* input,char\* output,float lx,float ly) | 最近邻插值法去栅格，input是输入文件名，output是输出文件名。lx和ly是长和宽需要缩放的倍数。支持8位BMP图像。 |
| void ImageScalingBilinearInterpolation(char\* input,char\* output,float lx,float ly) | 双线性插值法去栅格，input是输入文件名，output是输出文件名。lx和ly是长和宽需要缩放的倍数。支持8位BMP图像。 |
| void BilinearInterpolationScaling(char\* input,char\* output,float ExpScalValue) | 双线性插值，input是输入文件名，output是输出文件名。ExpScalValue是期望的缩放倍数（允许小数）。支持BMP图像。 |
| void NearestNeighborInterpolationScaling(char\* input,char\* output,float ExpScalValue) | 最近邻插值，input是输入文件名，output是输出文件名。ExpScalValue是期望的缩放倍数（允许小数）。支持BMP图像。 |
| void RotateRight90Degrees(char\* input,char\* output) | input是输入文件名，output是输出文件名。支持8位BMP图像，向右旋转90度。 |
| void RotateLeft90Degrees(char\* input,char\* output) | input是输入文件名，output是输出文件名。支持8位BMP图像，向左旋转90度。 |
| void ImageRotation(char\* input,char\* output,double angle) | 图像旋转，input是输入文件名，output是输出文件名。支持8位BMP图像。angle是要旋转的角度。 |
| void Rotation8(char\* input,char\* output,double Angle,int x1,int y1,int x2,int y2,unsigned char color) | 图像旋转，input是输入文件名，output是输出文件名。支持8位BMP图像。Angle是要旋转的角度数；x1、y1、x2、y2是旋转所围绕的中心点的坐标，color是旋转后非原图区域的填充颜色。 |
| void Rotation24(char\* input,char\* output,double Angle,int x1,int y1,int x2,int y2,unsigned char red,unsigned char green,unsigned char blue) | 图像旋转，input是输入文件名，output是输出文件名。支持24位BMP图像。Angle是要旋转的角度数；x1、y1、x2、y2是旋转所围绕的中心点的坐标；red、green、blue分别是旋转后非原图区域要填充的颜色的红绿蓝分量。 |
| void Rotation(char\* input,char\* output,int angle,unsigned char color) | 图像旋转，input是输入文件名，output是输出文件名。支持8位BMP图像。angle是旋转的角度，color是旋转后非原图区域填充的颜色，如color=100。 |
| void Rotate(char\* input,char\* output,int angle) | 图像旋转，input是输入文件名，output是输出文件名。支持BMP图像。angle是旋转的角度。 |
| void FileWrite(char\* BMP,char\* TXT) | 图像隐写之文件写入，将文本文件写入图像。支持32位BMP图像。BMP是要写入的图像文件名，TXT是要写入图像的文本文件名。 |
| void FileWriteOut(char\* BMP,char\* TXT) | 图像隐写之文件写出，将文本文件从图像中取出来。支持32位BMP图像。BMP是要写出的图像文件名，TXT是写出图像后信息保存的文本文件名。 |
| void grayToColor(FILE\* input,FILE\* output) | 灰色转伪彩色，input是输入文件，output是输出文件。支持8位和24位BMP图像。 |
| void ImageThinning(char\* input,char\* output,char\*\* str,int n,int m1,int a,int b) | 图像细化，input是输入文件名，output是输出文件名。支持4位BMP图像。n是str第一维的大小，m1是第二维的大小，形如str[n][m1]；a和b是相关的调节参数，可以为a=3，b=5。  参考模板：  char str[6][8] = { { 0, 0, 0, 0, 0, 0, 0, 0, }, { 255, 0, 255, 0, 0, 255, 0, 0 },  { 255, 0, 255, 255, 0, 255, 0, 255 }, { 255, 255, 255, 0, 0, 255, 255, 255 },  { 255, 0, 255, 255, 0, 255, 255, 255 }, { 0, 255, 255, 255, 255, 255, 255, 255 } }; |
| int MinimumValueOfImagePixels(char\* filename) | 返回图像像素的最小值，filename是输入的图像文件名。支持8位和24位BMP图像。 |
| int MaximumValueOfImagePixels(char\* filename) | 返回图像像素的最大值，filename是输入的图像文件名。支持8位和24位BMP图像。 |
| float AverageValueOfImagePixels(char\* filename) | 返回图像像素的均值，filename是输入的图像文件名。支持8位和24位BMP图像。 |
| double StandardDeviationOfImagePixels(char\* filename) | 返回图像像素的标准差，filename是输入的图像文件名。支持8位和24位BMP图像。 |
| double EntropyOfImage(char\* filename) | 返回图像的熵，支持8位和24位BMP图像。 |
| float\* CountTheFrequencyOfPixels(char\* filename) | filename是输入的图像文件名。存储每个像素的频率，像素值为0~255，返回值数组中的元素序号即为像素值，该序号在数组下的值即为这个像素的频率。支持8位和24位BMP图像。 |
| void HistogramEqualization2(char\* input,char\* output,int imgBit) | 直方图均衡化，input是输入文件名，output是输出文件名。支持8位和24位BMP图像。imgBit是输入图像的位数。 |
| void HistogramEqualization3(char\* input,char\* output) | 直方图均衡化，input是输入文件名，output是输出文件名。支持8位和24位BMP图像。 |
| void HistogramEqualization4(char\* input,char\* output) | 直方图均衡化，input是输入文件名，output是输出文件名。支持8位和24位BMP图像。input是输入文件名称，out是输出文件名称。 |
| void MedianFiltering1(char\* input,char\* output) | 中值滤波，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void MedianFiltering2(char\* input,char\* output) | 中值滤波，input是输入文件名，output是输出文件名。支持8位和24位BMP图像。 |
|  |  |
| void ThresholdProcessing(char\* input,char\* output,int Threshold) | 阈值处理，input是输入文件名，output是输出文件名。支持8位BMP图像。Threshold是阈值相关参数，如Threshold=0.001 |
| void OTSUProcessing(char\* input,char\* output) | 大津法处理，input是输入文件名，output是输出文件名。支持8位BMP图像。 |
| void HistogramEqualization5(char\* input,char\* output) | 直方图均衡，支持8位和16位BMP。 |
| void Resize(char\* input,char\* output,int Height,int Width) | 图片缩放，支持8位和16位BMP。Height和Width是输出图像的高和宽。 |
| double MeanBrightness(char\* input) | 求图像的平均亮度，支持8位和16位BMP。 |
| void KMeans(string input,unsigned int Clusters,char\* output) | K-Means聚类，input是输入文件名，Clusters是聚类的种类数目，output是输出文件名。支持PNG等多种图像。 |

**其他处理**

|  |  |
| --- | --- |
| void DES\_Encrypt(char \*PlainFile, char \*Key,char \*CipherFile) | DES加密函数，支持多种文件。PlainFile是原文件的文件名，Key是密钥字符，CipherFile是加密后的文件名。 |
| void DES\_Decrypt(char \*CipherFile, char \*Key,char \*PlainFile) | DES解密函数，支持多种文件。CipherFile是已加密文件的文件名，Key是密钥字符，PlainFile是解密后的文件名。 |
| void Encode(char\* input,char\* output) | 文本文件压缩，input是输入文件名，output是输出文件名。 |
| void Decode(char\* input,char\* output) | 文本文件压缩结果解压缩，input是输入文件名，output是输出文件名。 |
| void FileCompress(char \*input , char \*output) | 文件压缩，input是输入文件名，output是输出文件名。 |
| void FileDecompression(char \*input , char \*output) | 文件压缩结果解压缩，input是输入文件名，output是输出文件名。 |