

《数字图像处理》实验报告

实验名称 : 实验 3 数字图像的几何变换

实验日期 : 2022-9-16

姓 名 : 傅康

学 号 : 084520126

班 级 : 医信 20

成 绩 :

信息技术学院

南京中医药大学

实验目的:

1. 应用 MATLAB 语言编写实现一幅图像的镜像、旋转、剪切和缩放。
2. 设计合理的交互方式及操作流程。

实验内容和要求

建立一个名为“xxxxx 实验 3”的解决方案（xxxxx 为自己的学号）

1. 使用 GUIDE 或 AppDesigner 设计实现一个 GUI 程序, 实现图像读取、显示, 并用菜单实现彩色图像转换为灰度图像。
提示:
 - a) 利用 `imread()` 函数读取一幅图像, 用 `imshow()` 实现图像的显示;
 - b) 用 `global` 设置全局变量。
2. 利用 `rgb2gray` 命令实现彩色图像向灰度图像的转换操作; 设计代码实现图像镜像处理
3. 自定义一个 8×8 的矩阵, 使用 `imagesc` 显示
4. 利用 `imrotate()` 函数, 设置参数为 `nearest` 实现图像及自定义矩阵的最邻域旋转;
5. 利用 `imrotate()` 函数, 设置参数为 `bilinear` 实现图像及自定义矩阵的双线性旋转;
6. 利用 `imrotate()` 函数, 设置参数为 `bicubic` 实现图像及自定义矩阵的双立方旋转;
7. 利用 `imcrop()` 函数, 实现图像的剪切;
8. 利用 `imresize()` 函数, 实现图像及自定义矩阵的缩放;
9. 编写代码实现图像错切变换;
10. 编写代码实现图像平移变换;
11. 利用 `maketform()` 和 `imtransform()` 函数, 实现图像及自定义矩阵根据变换结构实现的变换。
12. 解析各种函数的处理结果, 并测量各算法执行时间

运行结果（写清题号）

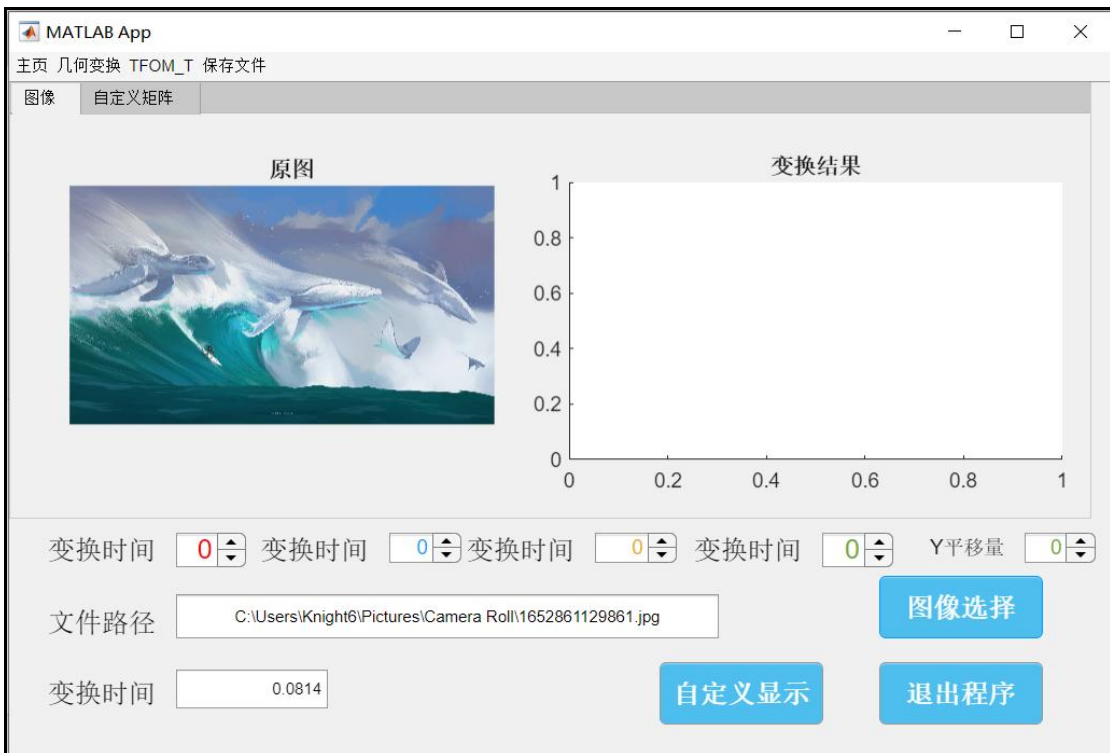
描述实验的基本步骤, 用数据和图片给出各个步骤中取得的实验结果和源代码, 并进行必要的讨论, 必须包括原始图像及其计算/处理后的图像。

1. 使用 GUIDE 或 AppDesigner 设计实现一个 GUI 程序, 实现图像读取、显示, 并用菜单实现彩色图像转换为灰度图像。

提示:

a) 利用 `imread()` 函数读取一幅图像, 用 `imshow()` 实现图像的显示;

```
function ButtonPushed(app, event)
%打开文件选择对话框,读入一张图片
[filename,pathname]=uigetfile({'*.jpg'; '*.bmp'; '*.tif'; '*.png'; '*.jif'; '*..*'}, '选择图像');
if isequal(filename,0) || isequal(pathname,0)
    errordlg('没有选中图片文件', '错误');
else
    app.file=strcat(pathname,filename);
    tic;
    im=imread(app.file);
    app.t=0+toc;
    imshow(im, 'Parent', app.UIAxes, 'InitialMagnification', 100);
    set(app.EditField, 'value', app.file);
    set(app.EditField_3, 'value', double(app.t));
end
end
```



b) 用 `global` 设置全局变量。

```
%全局变量,便于不同function调用共同属性
properties (Access = private)
    file % 图像文件
    %自定义8×8矩阵
    Custom_mat=magic(8);
    t=0;%测量时间
end
```

2. 利用 rgb2gray 命令实现彩色图像向灰度图像的转换操作；设计代码实现图像镜像处理。

```
function Rgb2GrayMenuSelected(app, event)
    im=imread(app.file);
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        tic;
        im=rgb2gray(im);
        app.t=0+toc;
        imshow(im,'Parent',app.UIAxes_2);
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
end
```

彩色图像转灰度图像



垂直镜像

```
function Menu_32Selected(app, event)
    %垂直镜像
    im=imread(app.file);
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        H=size(im);
        tic
        imB(1:H(1),1:H(2),1:H(3))=im(H(1):-1:1,1:H(2),1:H(3));
        app.t=0+toc;
        imshow(imB,'Parent',app.UIAxes_2);
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
end
```



水平镜像

```
function Menu_30Selected(app, event)
%水平镜像
im=imread(app.file);
if isempty(im)
    warndlg('请先读取一个图像');
else
    H=size(im);
    tic
    imB(1:H(1),1:H(2),1:H(3))=im(1:H(1),H(2):-1:1,1:H(3));
    app.t=0+toc;
    imshow(imB, 'Parent', app.UIAxes_2);
    set(app.EditField, 'value', app.file);
    set(app.EditField_3, 'value', double(app.t));
end
end
```



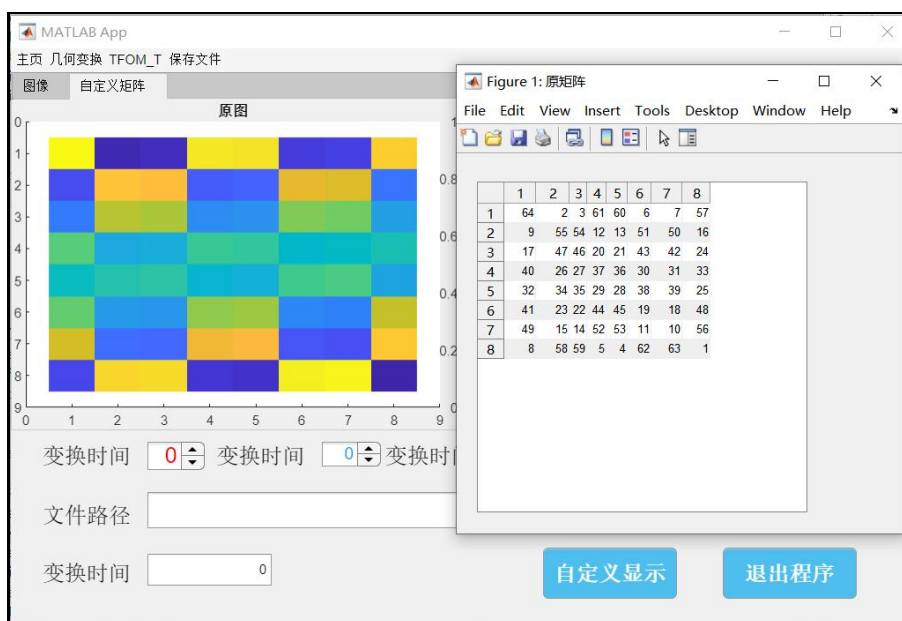
对角镜像

```
function Menu_31Selected(app, event)
    %对角镜像
    im=imread(app.file);
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        H=size(im);
        tic
        imB(1:H(1),1:H(2),1:H(3))=im(H(1):-1:1,H(2):-1:1,1:H(3));
        app.t=0+toc;
        imshow(imB,'Parent',app.UIAxes_2);
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
end
```




- 自定义一个 8×8 的矩阵，使用 `imagesc` 显示。
矩阵已在 1. b) 全局变量处使用 `magic` 幻方矩阵定义。

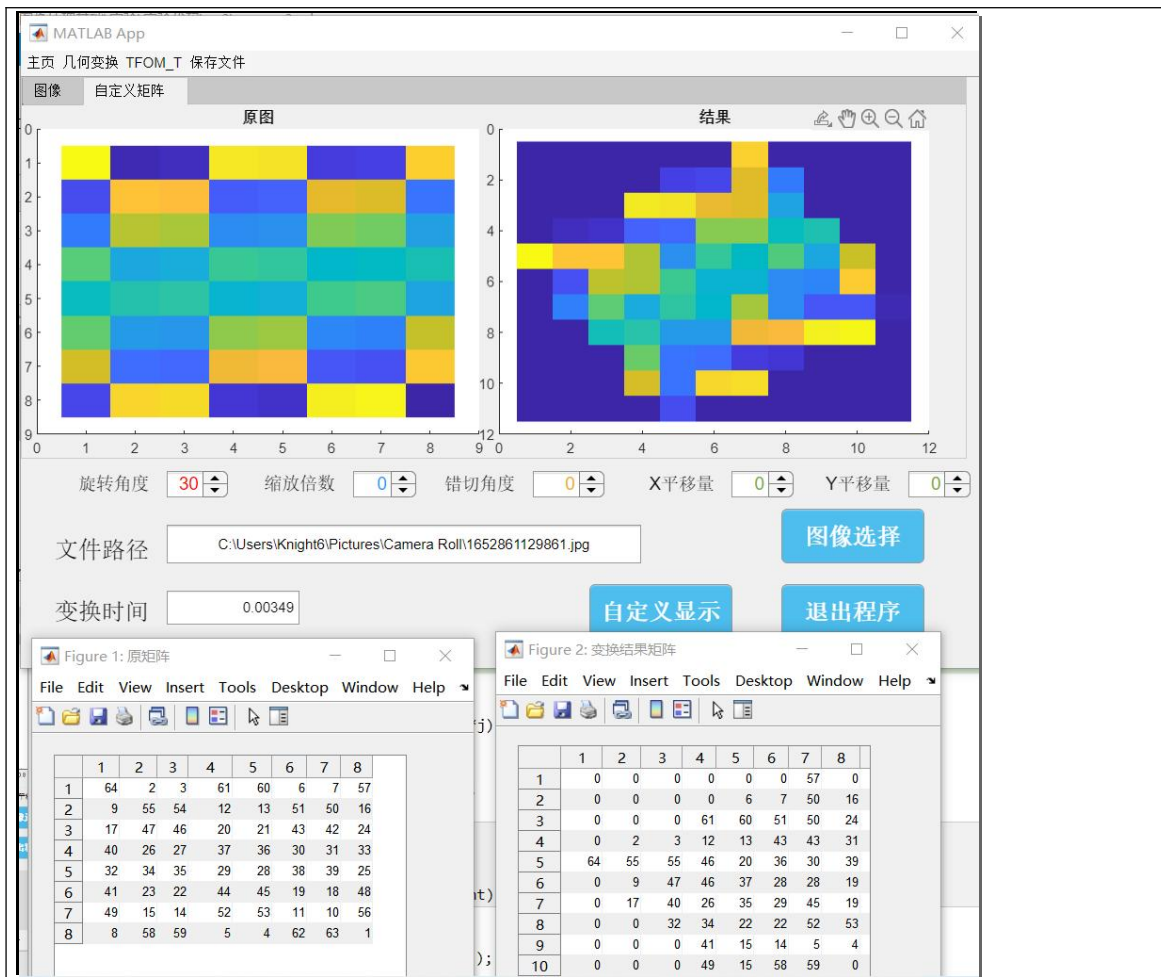
```
function Button_3Pushed2(app, event)
%自定义8×8矩阵Custom_mat
imagesc(app.UIAxes2,app.Custom_mat);
fig=figure('Name','原矩阵');
uitable('Parent',fig,'Data',app.Custom_mat);
end
```



4. 利用 `imrotate()` 函数, 设置参数为 `nearest` 实现图像及自定义矩阵的最邻域旋转;

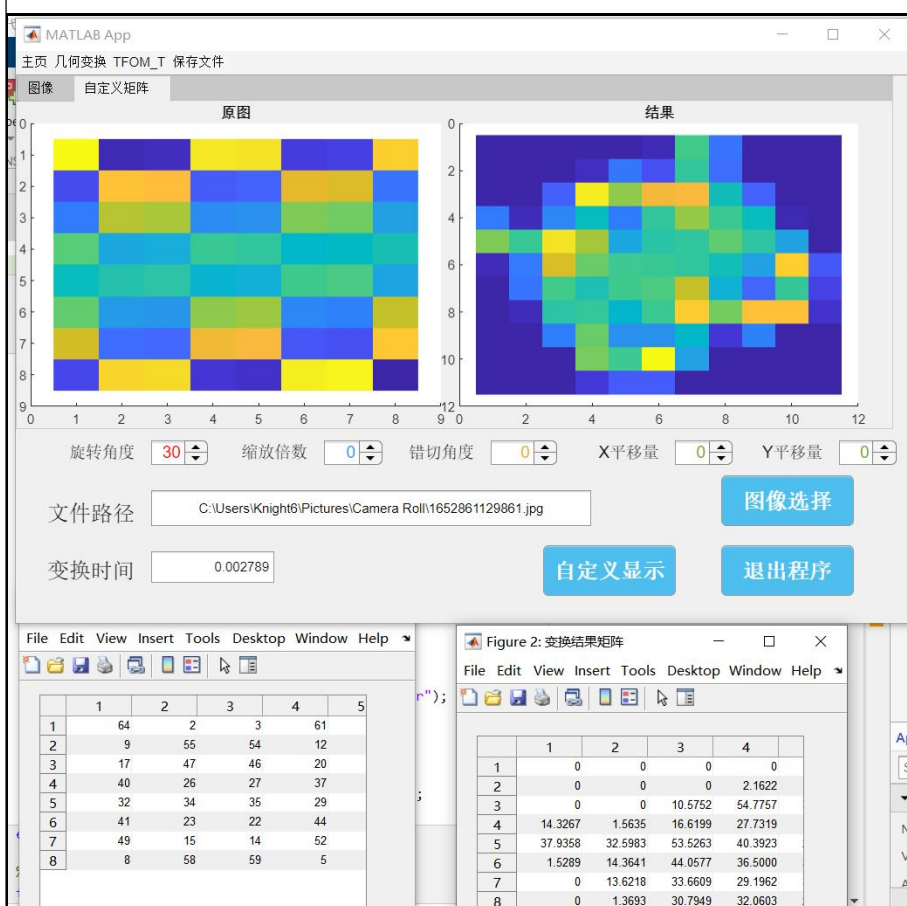
```
function Menu_19Selected(app, event)
%最邻域旋转
if app.TabGroup.SelectedTab==app.Tab
    im=imread(app.file);%选中图片
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        angle=app.Spinner.Value;
        tic;
        imB=imrotate(im,angle,"nearest");
        app.t=0+toc;
        imshow(imB,'Parent',app.UIAxes_2);
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
else%自定义矩阵
    angle=app.Spinner.Value;
    tic;
    imB=imrotate(app.Custom_mat,angle,"nearest");
    app.t=0+toc;
    imagesc(app.UIAxes2_2,imB);
    fig=figure('Name','变换结果矩阵');
    uitable('Parent',fig,"Data",imB);
    set(app.EditField_3,'value',double(app.t));
end
end
```





5. 利用 `imrotate()` 函数，设置参数为 `bilinear` 实现图像及自定义矩阵的双线性旋转；

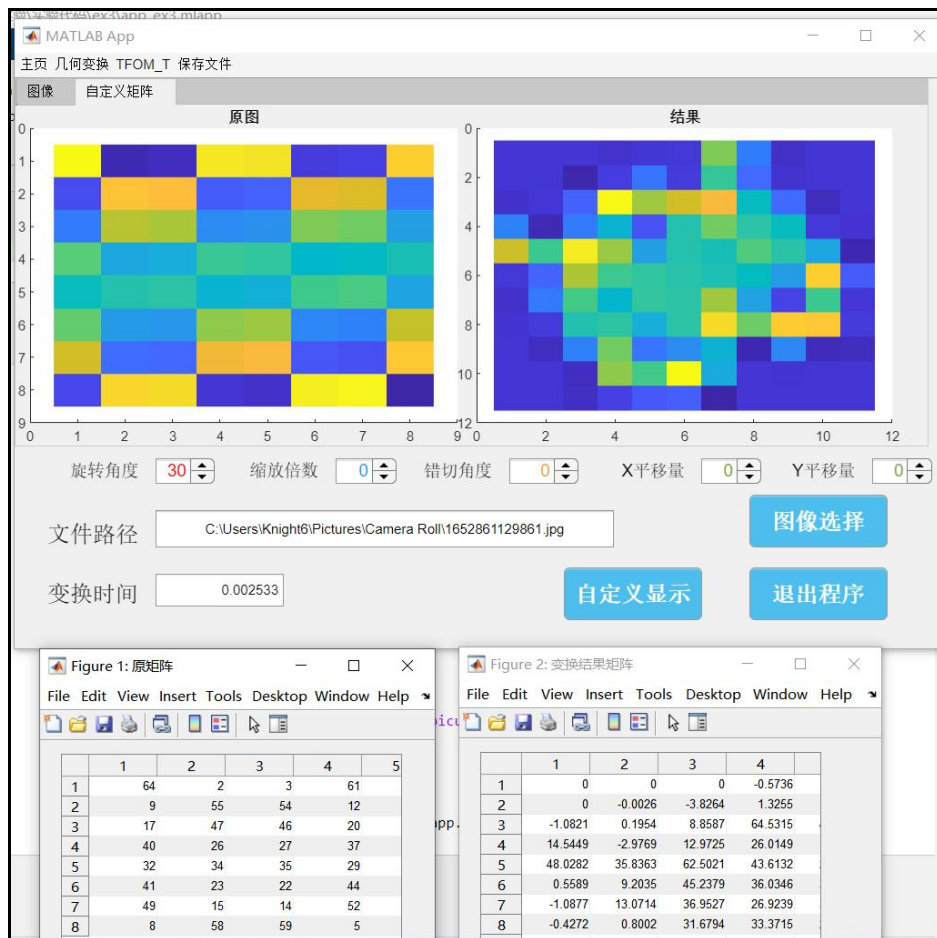
```
function Menu_20Selected(app, event)
%双线性旋转
if app.TabGroup.SelectedTab==app.Tab
im=imread(app.file);%选中图片
if isempty(im)
warndlg('请先读取一个图像');
else
angle=app.Spinner.Value;
tic;
imB=imrotate(im,angle,"bilinear");
app.t=0+toc;
imshow(imB,'Parent',app.UIAxes_2);
set(app.EditField,'value',app.file);
set(app.EditField_3,'value',double(app.t));
end
else%自定义矩阵
angle=app.Spinner.Value;
tic;
imB=imrotate(app.Custom_mat,angle,"bilinear");
app.t=0+toc;
imagesc(app.UIAxes2_2,imB);
fig=figure('Name','变换结果矩阵');
uitable('Parent',fig,'Data',imB);
set(app.EditField_3,'value',double(app.t));
end
end
```



6. 利用 `imrotate()` 函数，设置参数为 `bicubic` 实现图像及自定义矩阵的双立方旋转；

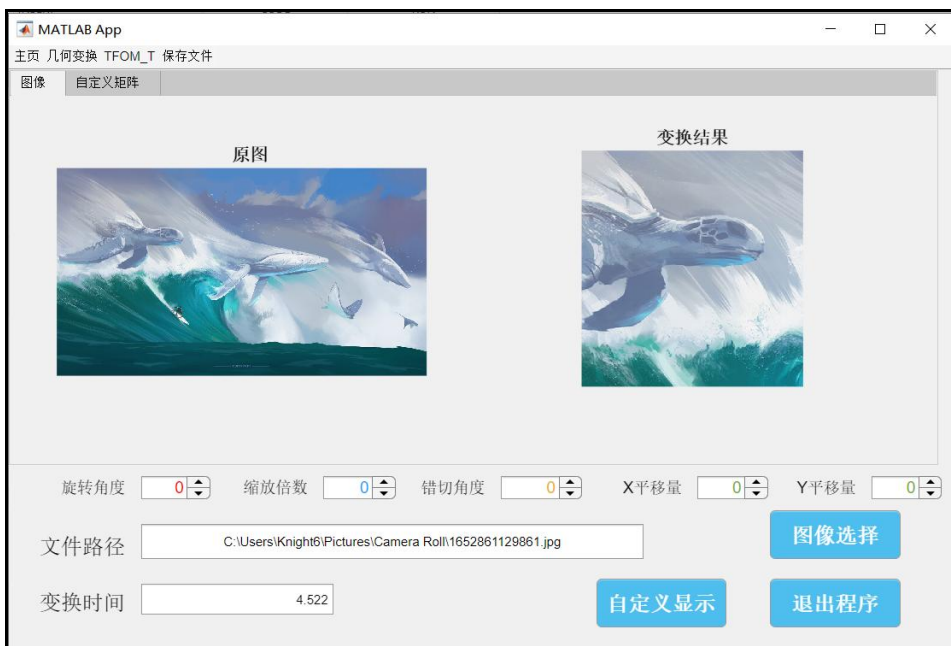
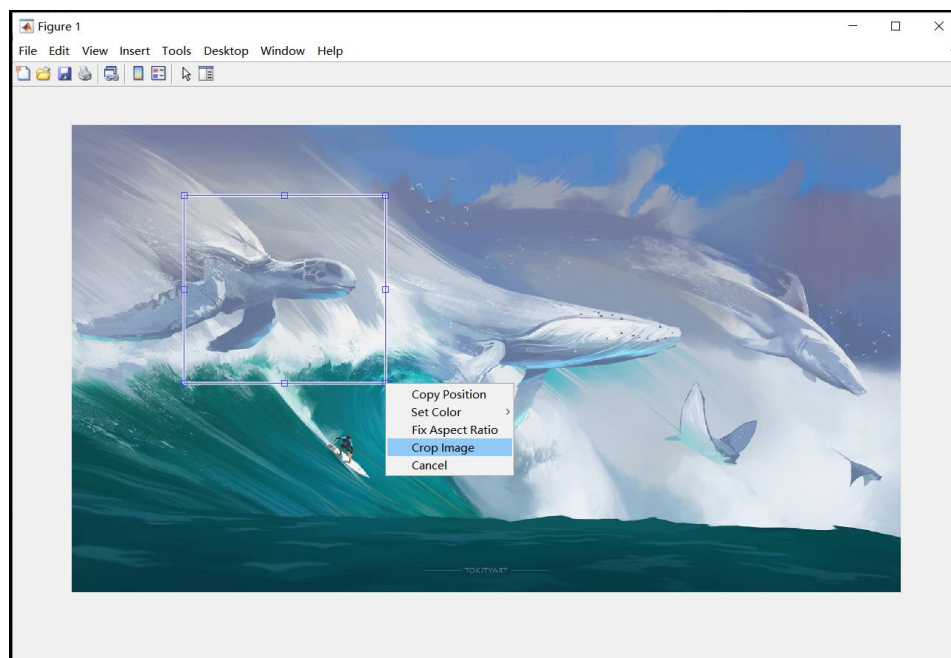
```
function Menu_21Selected(app, event)
%双立方旋转
if app.TabGroup.SelectedTab==app.Tab
    im=imread(app.file);%选中图片
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        angle=app.Spinner.Value;
        tic;
        imB=imrotate(im,angle,"bicubic");
        app.t=0+toc;
        imshow(imB,'Parent',app.UIAxes_2);
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
else%自定义矩阵
    angle=app.Spinner.Value;
    tic;
    imB=imrotate(app.Custom_mat,angle,"bicubic");
    app.t=0+toc;
    imagesc(app.UIAxes2_2,imB);
    fig=figure('Name','变换结果矩阵');
    uitable('Parent',fig,"Data",imB);
    set(app.EditField_3,'value',double(app.t));
end
end
```





7. 利用 `imcrop` () 函数，实现图像的剪切；

```
function Menu_23Selected(app, event)
    %图像剪切
    im=imread(app.file);
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        Msg=msgbox('请选中所需区域并右键裁剪','提示');
        waitfor(Msg);%等待用户选中再裁剪
        tic;
        imB=imcrop(im);
        app.t=0+toc;
        imshow(imB,'Parent',app.UIAxes_2);
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
end
```



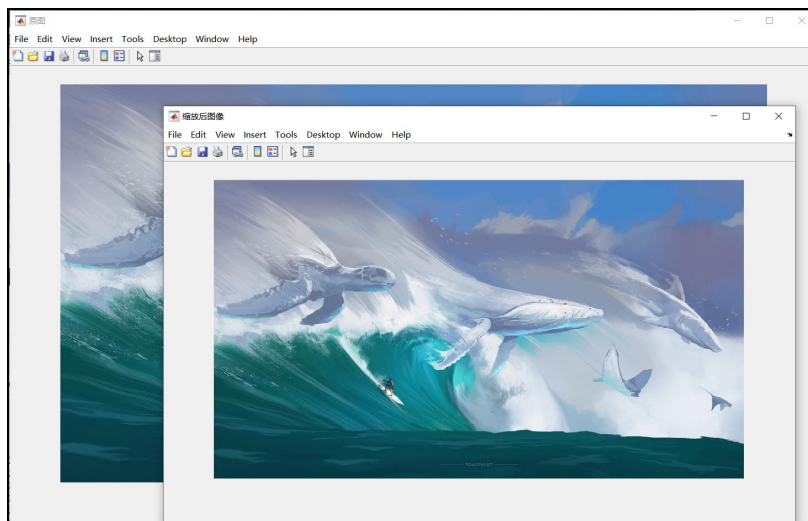
8. 利用 `imresize()` 函数，实现图像及自定义矩阵的缩放；

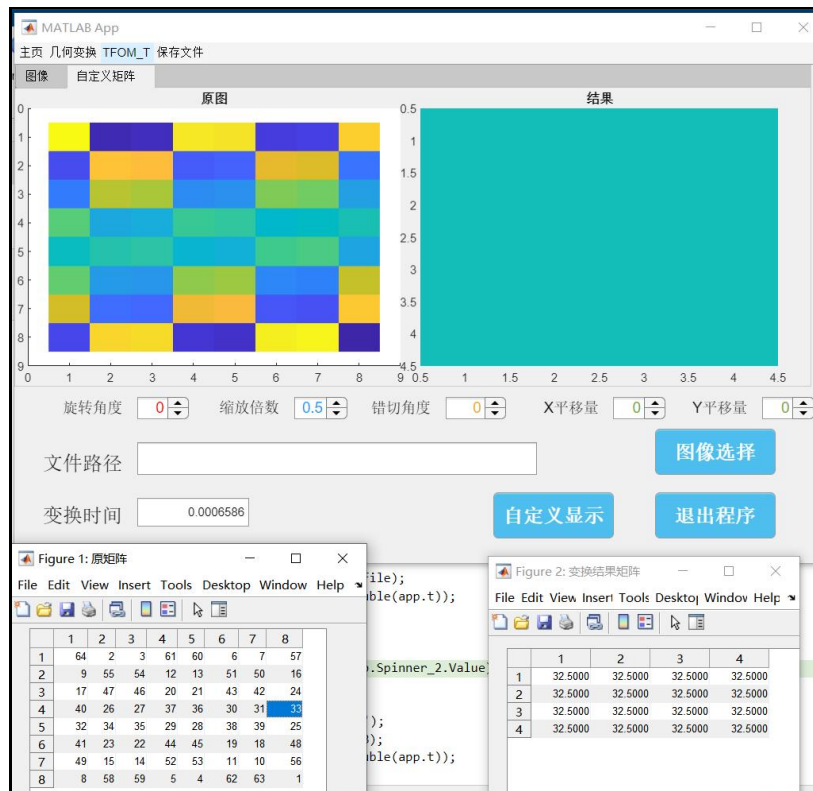

```
function Menu_24Selected(app, event)
%缩放
if app.TabGroup.SelectedTab==app.Tab
    im=imread(app.file);
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        %imshow自动优化图窗,用figure来单独显示
        figure('Name','原图','NumberTitle','off'),
        imshow(im);
        tic;
        imB=imresize(im,app.Spinner_2.Value);
        app.t=0+toc;
        figure('Name','缩放后图像','NumberTitle','off'),
        imshow(imB);
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
else
    tic;
    imB=imresize(app.Custom_mat,app.Spinner_2.Value);
    app.t=0+toc;
    imagesc(app.UIAxes2_2,imB);
    fig=figure('Name','变换结果矩阵');
    uitable('Parent',fig,"Data",imB);
    set(app.EditField_3,'value',double(app.t));
end
end
```

旋转角度 缩放倍数 错切角度 X平移量 Y平移量

文件路径

变换时间

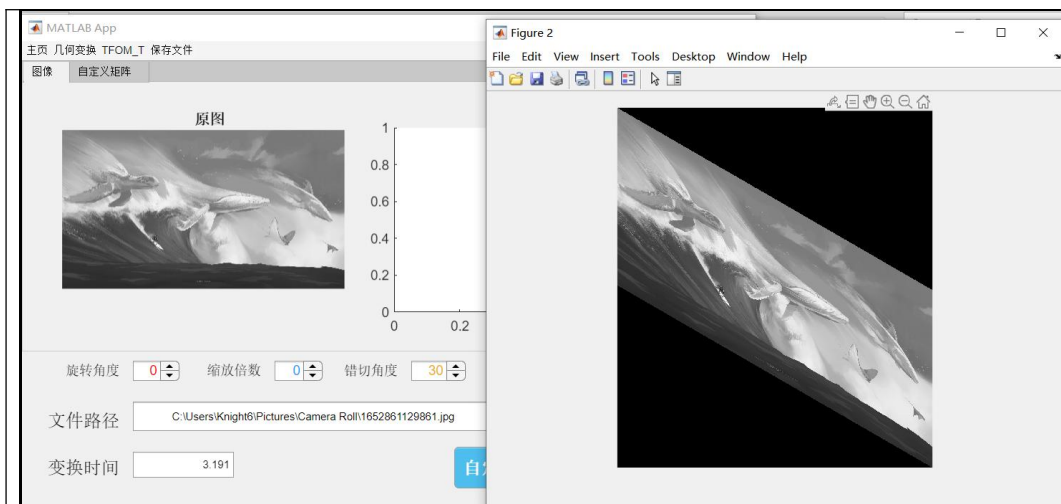




9. 编写代码实现图像错切变换： 垂直错切：

```
function Menu_33Selected(app, event)
    %垂直错切变换
    im=imread(app.file);
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        imB=rgb2gray(im);%转灰度变二维进行处理
        imshow(imB,'Parent',app.UIAxes);
        [row,col]=size(imB);

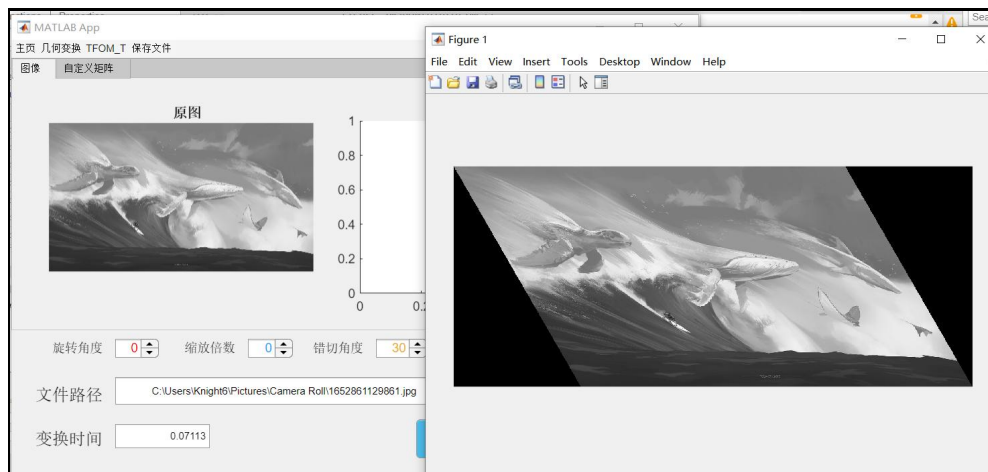
        imC=zeros(row,col);%先置零,提高程序运行效率
        angle=tan(app.Spinner_3.Value*pi/180);
        tic;
        for i=1:row
            for j=1:col
                imC(round(i+angle*j),j)=imB(i,j);%x不变,移动y
            end
        end
        app.t=0+toc;
        figure,imshow(uint8(imC));
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
end
```



水平错切:

```
function Menu_34Selected(app, event)
    %水平错切变换
    im=imread(app.file);
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        imB=rgb2gray(im);%转灰度变二维进行处理
        imshow(imB,'Parent',app.UIAxes);
        [row,col]=size(imB);

        imC=zeros(row,col);%先置零,提高程序运行效率
        angle=tan(app.Spinner_3.Value*pi/180);
        tic;
        for i=1:row
            for j=1:col
                imC(i,round(j+angle*i))=imB(i,j);%x移动,y不变
            end
        end
        app.t=0+toc;
        figure,imshow(uint8(imC));
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
end
```



10. 编写代码实现图像平移变换；

```
function Menu_27Selected(app, event)
    %平移变换
    im=imread(app.file);
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        imshow(im,'Parent',app.UIAxes);
        H=size(im);%原图矩阵
        moveX=app.YSpinner.Value;
        moveY=app.XSpinner.Value;
        imB=zeros(size(im));%先置零,提高程序运行效率
        tic;
        imB(moveX+1:H(1),moveY+1:H(2),1:H(3))=im(1:H(1)-moveX,1:H(2)-moveY,1:H(3));%移动
        app.t=0+toc;
        imshow(uint8(imB),'Parent',app.UIAxes_2);%转uint8不然会损失变黑白
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
end
```



11. 利用 maketform() 和 imtransform() 函数，实现图像及自定义矩阵根据变换结构实现的变换。

图像旋转

```
function Menu_35Selected(app, event)
%TFORM_T旋转
im=imread(app.file);
if isempty(im)
warndlg('请先读取一个图像');
else
angle=app.Spinner.Value;
T=maketform('affine',[cosd(angle) -sind(angle) 0; sind(angle) cosd(angle) 0; 0 0 1]);%变换矩阵
tic;
imB=imtransform(im,T);
app.t=0+toc;
imshow(imB,'Parent',app.UIAxes_2);
set(app.EditField,'value',app.file);
set(app.EditField_3,'value',double(app.t));
end
end
```

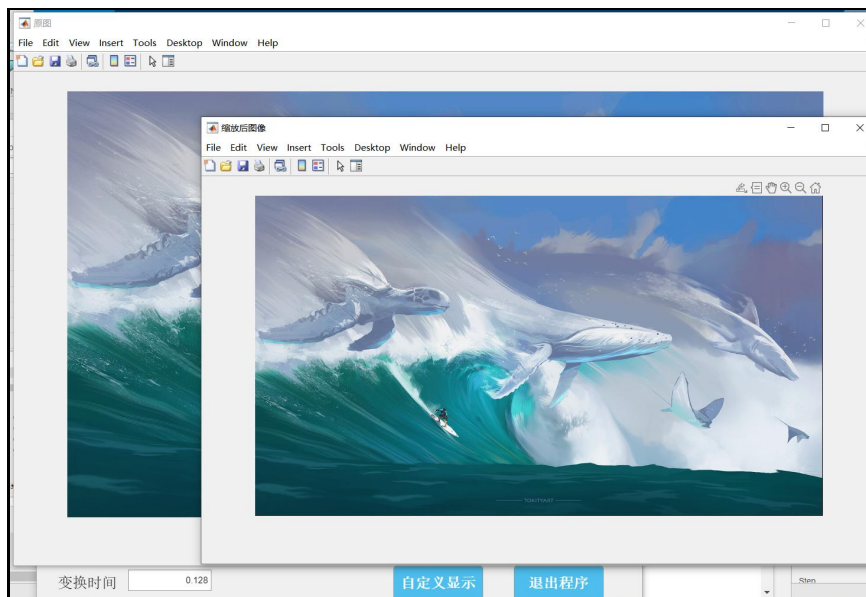


图像缩放

```

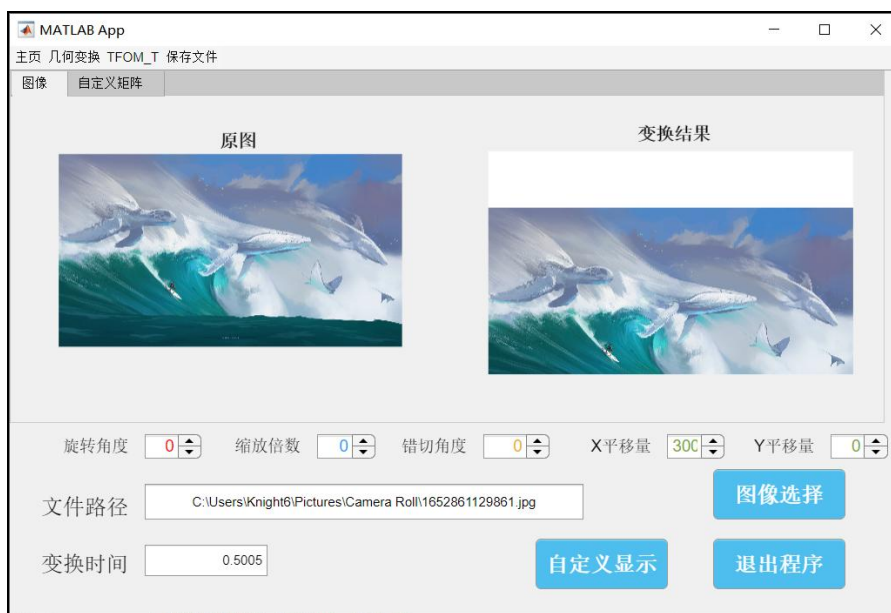
function Menu_36Selected(app, event)
    %TFORM_T缩放
    im=imread(app.file);
    if isempty(im)
        warndlg('请先读取一个图像');
    else
        figure('Name','原图','NumberTitle','off'),
        imshow(im);
        mul=app.Spinner_2.Value;
        T=maketform('affine',[mul 0 0; 0 mul 0; 0 0 1]);
        tic;
        imB=imtransform(im,T);
        app.t=0+toc;
        figure('Name','缩放后图像','NumberTitle','off');
        imshow(imB);
        set(app.EditField,'value',app.file);
        set(app.EditField_3,'value',double(app.t));
    end
end
end

```



图像平移


```
function Menu_37Selected(app, event)
%TFORM_T平移
im=imread(app.file);
if isempty(im)
    warndlg('请先读取一个图像');
else
    imshow(im,'Parent',app.UIAxes);
    moveX=app.YSpinner.Value;
    moveY=app.XSpinner.Value;
    imB=zeros(size(im));%先置零,提高程序运行效率
    TFORM=[1 0 moveX; 0 1 moveY; 0 0 1]';
    T=maketform('affine',TFORM);
    tic;
    imB=imtransform(im,T,'Xdata',[1 (size(im,2)+TFORM(3,1))], 'Ydata',[1 (size(im,1)+TFORM(3,2))], 'FillValues',255);
    %size(im,2)表示原始像素列数
    app.t=0+toc;
    imshow(uint8(imB), 'Parent', app.UIAxes_2);
    set(app.EditField,'value',app.file);
    set(app.EditField_3,'value',double(app.t));
end
end
```



图像切变

```
function Menu_38Selected(app, event)
%TFORM_T切变
im=imread(app.file);
if isempty(im)
    warndlg('请先读取一个图像');
else
    T=maketform('affine',[1 4 0;2 1 0;0 0 1]');
    tic;
    imB=imtransform(im,T);
    app.t=0+toc;
    imshow(imB, 'Parent', app.UIAxes_2);
    set(app.EditField,'value',app.file);
    set(app.EditField_3,'value',double(app.t));
end
end
```




12. 解析各种函数的处理结果，并测量各算法执行时间。

已显示在各图变换时间控件，解析见下讨论。

讨论/说明：

①`uigetfile` 可以指定打开选择图片，提升了程序的交互性，没有选中文件进行相应的报错处理。

②`Imshow` 可以显示在坐标系中，但对于 app 而言用 `image` 显示更好，因为 `imshow` 他会自动优化大小，特别是缩放和平移，会导致看不出图像的变化，所以这两块功能我尝试通过 `figure` 弹窗和 `image` 进行实现。

③同样对于错切等一些变换，我先进行了一个灰度图的转换，以便于处理 2-D 的矩阵。

④对比一些自写函数效率没有 matlab 高，私以为一个原因是自己写的，特别是对矩阵的操作用了大量 `for` 循环，二 matlab 里很多数值计算的函数都是用 fortran 语言所写并对算法进行了优化。

⑤可通过设置 `Accelerator` 来设置快捷方式，比如 `Ctrl+N`。

⑥读入图像 `double` 读出显示转 `uint8`，减少图像的损失。

实验的体会与思考题

1. 灵活的交互方式为软件的使用带来怎样的体验

用户体验感更好，整个软件对用户而言就类似于一个黑箱，其只要给定输入，选择功能，便可以得到想要的结果。封装层度更深，设计使用方便。