

# 实验二 图像几何变换实验

## 实验目的

1. 学习几种常见的图像几何变换，并通过实验体会几何变换的效果；
2. 掌握图像平移、剪切、缩放、旋转、镜像等几何变换的算法原理及编程实现；
3. 掌握 matlab 编程环境中基本的图像处理函数。

## 实验内容

1. 启动 MATLAB 程序，对图像文件分别进行平移、垂直镜像变换、水平镜像变换、缩放和 旋转操作。
2. 运行图像处理程序，并保存处理结果图像。

## 源代码

```
1  %平移
2  I=imread( 'lena.jpg' );
3  subplot(121),imshow(I);
4  title( 'before' );
5  I=double(I);
6  M=zeros(size(I));
7
8  N=size(I);
9  x=50;
10 y=50;
11 M(x+1:N(1),y+1:N(2))=I(1:N(1)-x,1:N(2)-y);
12 subplot(122),imshow(uint8(M));
13 title( 'after' );
14
15 %水平垂直镜像
16 I=imread( 'lena.jpg' );
17 subplot(131),imshow(I);
18 title( 'before' );
19 I=double(I);
20 A=zeros(size(I));
21 B=zeros(size(I));
22 M=size(I);
23 A(1:M(1),1:M(2))=I(M(1):-1:1,1:M(2));
24 B(1:M(1),1:M(2))=I(1:M(1),M(2):-1:1);
25 subplot(132),imshow(uint8(A));
26 title( '  竖直  ' );
27 subplot(133),imshow(uint8(B));
28 title( '  水平  ' );
29
30 %缩放
31 I=imread( 'lena.jpg' );
32 subplot(131),imshow(I);
33 title( 'before' );
34 I=double(I);
35 A=zeros(size(I));
36 B=zeros(size(I));
37 [m,n]=size(I);
38 x=1.8;
39 y=1.8;
40 x2=0.85;
41 y2=0.85;
```

```

42 for i=1:m
43     for j=1:n
44         i1=round(i*x);
45         j1=round(j*y);
46         i2=round(i*x2);
47         j2=round(j*y2);
48         if (j1>=0)&&(i1>=0)&&(i1<=m)&&(j1<=n)
49             A(i,j)=I(i1,j1);
50         end
51         if (j2>=0)&&(i2>=0)&&(i2<=m)&&(j2<=n)
52             B(i,j)=I(i2,j2);
53         end
54     end
55 end
56 subplot(132),imshow(uint8(A));
57 title( 'small' );
58 subplot(133),imshow(uint8(B));
59 title( 'big' );
60
61 %旋转
62 jiao=50;
63 M=imread( 'lena.jpg' );
64 imshow(M);
65 [h w]=size(M);
66
67 theta=jiao/180*pi;
68 rot=[cos(theta) -sin(theta) 0;sin(theta) cos(theta) 0;0 0 1]; % 建立变换矩阵
69 pix1=[1 1 1]*rot; % 变后左上
70 pix2=[1 w 1]*rot; % 变后右上
71 pix3=[h 1 1]*rot; % 变后左下
72 pix4=[h w 1]*rot; % 变后右下
73
74 height=round(max([abs(pix1(1)-pix4(1))+0.5 abs(pix2(1)-pix3(1))+0.5])); % 新宽
75 width=round(max([abs(pix1(2)-pix4(2))+0.5 abs(pix2(2)-pix3(2))+0.5])); % 新高
76 N=zeros(height,width);
77
78 chao_y=abs(min([pix1(1) pix2(1) pix3(1) pix4(1)])); %y 负轴超量
79 chao_x=abs(min([pix1(2) pix2(2) pix3(2) pix4(2)])); %x 负轴超量
80
81 for i=1-chao_y:height-chao_y
82     for j=1-chao_x:width-chao_x
83         pix=[i j 1]/rot; % 用变换后图像的点的坐标去寻找原图像点的坐标，
84
85         float_Y=pix(1)-floor(pix(1)); % 向下舍入
86         float_X=pix(2)-floor(pix(2));
87
88         if pix(1)>=1 && pix(2)>=1 && pix(1) <= h && pix(2) <= w
89
90             pix_up_left=[floor(pix(1)) floor(pix(2))]; % 四个相邻的点
91             pix_up_right=[floor(pix(1)) ceil(pix(2))];
92             pix_down_left=[ceil(pix(1)) floor(pix(2))];
93             pix_down_right=[ceil(pix(1)) ceil(pix(2))];
94
95             value_up_left=(1-float_X)*(1-float_Y); % 周围四个点权重
96             value_up_right=float_X*(1-float_Y);
97             value_down_left=(1-float_X)*float_Y;
98             value_down_right=float_X*float_Y;
99
100             N(i+chao_y,j+chao_x)=value_up_left*M(pix_up_left(1),pix_up_left(2))+
101                 ...
102                 value_up_right*M(pix_up_right(1),pix_up_right(2))+ ...
103                 value_down_left*M(pix_down_left(1),pix_down_left(2))+ ...
104                 value_down_right*M(pix_down_right(1),pix_down_right(2));
105         end

```

```
106     end
107 end
108
109 figure,imshow(uint8(N))
110
```

## 实验现象





