

Implementation of Relationships between Pixels

Neighbour of 4,8 and Diagonal point

% To find Neighbour of a given Pixel

```
a=magic(5);  
disp('a='); disp(a);  
b=input('Enter the row < size of the Matrix');  
c=input(' Enter the Column < size of matrix');  
disp('Element'); disp(a(b,c));
```

% 4 Point Neighbour

```
N4=[a(b+1,c), a(b-1,c), a(b,c+1), a(b,c-1)];  
disp('N4='); disp(N4);
```

%8 Point Neighbour

```
N8=[a(b+1,c), a(b-1,c), a(b,c+1), a(b,c-1), a(b+1,c+1), a(b+1,c-1), a(b-1,c-1), a(b-1,c+1)]; disp('N8=');  
disp(N8);
```

%Diagonal Neighbour

```
ND=[ a(b+1,c+1), a(b+1,c-1), a(b-1,c-1), a(b-1,c+1)];  
disp('ND='); disp(ND);
```

Output

a=

```
17 24 1 8 15  
23 5 7 14 16  
4 6 13 20 22  
10 12 19 21 3  
11 18 25 2 9
```

Enter the row < size of the Matrix 3

Enter the Column < size of matrix 3

Element =

13

N4=

```
19 7 20 6
```

N8=

```
19 7 20 6 21 12 5 14
```

ND=

```
21 12 5 14
```

Implementation of Transformations of an Image

%Scaling & Rotation

% Scaling (Resize)

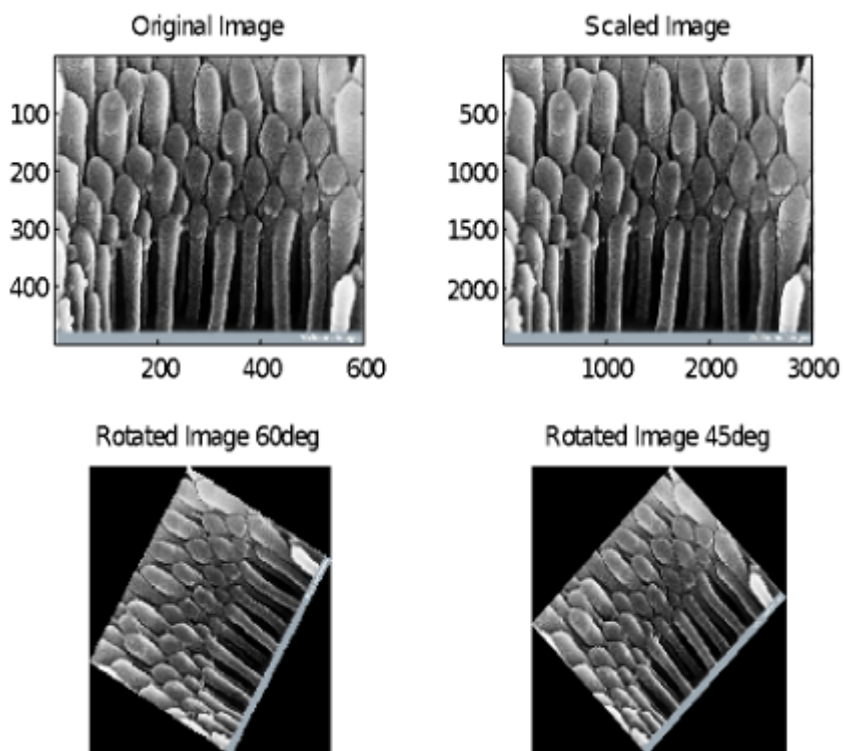
```
I=imread('earcell.jpg');  
subplot(2,2,1); subimage(I); title('Original Image');
```

```
s=input('Enter Scaling Factor');  
j=imresize(I,s);  
subplot(2,2,2); subimage(j); title('Scaled Image');
```

%Rotation

```
K=imrotate(j,60);  
subplot(2,2,3); imshow(K); title('Rotated Image 60deg');
```

```
R=imrotate(j,45);  
subplot(2,2,4); imshow(R); title('Rotated Image 45deg');
```



%Display the color image and its Resized images by different methods

%Display the color image

```
I=imread('embryo.jpg');  
figure,  
subplot(2,2,1);  
subimage(I);  
title('Original Image');
```

%Display Resized image by Bilinear method

```
B=imresize(I,5);  
subplot(2,2,2);  
subimage(B); title('Bilinear Image');
```

%Display Resized image by Nearest method

```
C=imresize(I,5,'nearest');  
subplot(2,2,3);  
subimage(C); title('Nearest Image');
```

%Display Resized image by Bicubic method

```
D=imresize(I,5,'Bicubic');  
subplot(2,2,4);  
subimage(D); title('Bicubic Image');
```

