```
17
18 VideoInfo = GetVideoStruct('C:\Users\senthilkumar\
      Desktop\Chandra_Mohan_LAB\
      Digital_Image_ProcessingLab\red-car-video.avi');
19
20 VideoFilePointer = OpenVideoFile('C:\Users\
      senthilkumar\Desktop\Chandra_Mohan_LAB\
      Digital_Image_ProcessingLab\red-car-video.avi');
21
22 figure();
23
24 for n = 1 : VideoInfo.NumberOfFrames
25
26  RGB = ReadImage(VideoFilePointer);
27
28  ShowColorImage(RGB, VideoPath);
29
30 end;
31
32 CloseVideoFile(VideoFilePointer);
```

Appendix

Scilab code AP 1

```
function [B] = rot90f(A,k)
[%nargout,%nargin] = argn(0)
//ROT90  Rotate matrix 90 degrees.
//   ROT90(A) is the 90 degree counterclockwise rotation of matrix A.
//   ROT90(A,K) is the K*90 degree rotation of A, K = +-1,+-2,...
[m,n] = size(A);
if %nargin==1 then
  k = 1;
else
  k = k-fix(k/4).*4;
  if(k<0) then
    k = k+4;
  end
end
if k == 1
             A = A.';
             B = A(n:-1:1,:);
elseif k == 2
             B = A(m:-1:1,n:-1:1);
elseif k == 3
             B = A(m:-1:1,:);
             B = B.';
else
             B = A;
end
```

```
endfunction
```

Rotate Image 90 degree

Scilab code AP 12

```
function [hea,b]=histbw12(a)
    //a=imread('C:\Users\senthilkumar\Desktop\
       Chandra_Mohan_LAB\Digital_Image_ProcessingLab
       \tire.jpeg')
    //a- original image
    //b- histogram
    //hea- histogram equalized image
    [m n]=size(a);
    for i=1:256
        b(i)=length(find(a==(i-1)));
    end
    pbb=b/(m*n);
    pb(1)=pbb(1);
    for i=2:256
        pb(i)=pb(i-1)+pbb(i);
    end

    s=pb*255;
    sb=uint8(round(s));
    index =0;
    for i=1:m
        for j=1:n
            index = double(a(i,j))+1;//convert it to
                double
            //otherwise index = 255+1 =0
            hea(i,j)= sb(index);//histogram
               equalization
        end
    end
endfunction
//note:
//First run this function
//type the following commands in scilab console
   window
```

```
30 //a=imread('C:\Users\senthilkumar\Desktop\
     Chandra_Mohan_LAB\Digital_Image_ProcessingLab\
     tire.jpeg')
31 //[hea,b] = histbw_12(a);
32 //figure,
33 //ShowImage(a,'Original Image')//IPD toolbox
34 //title('Original Image')
35 //figure
36 //plot2d3('gnn',[1:256],b)
37 //title('Histogram of the Image')
38 //figure
39 //ShowImage(hea,'Image after Histogram equalization
     ')//IPD toolbox
40 //title('Image after Histogram equalization')
```

Histogram of Gray images

Scilab code AP 13 `function [out] =imgenh11(a,ks)`

```
2     [m n]=size(a);
3     a1=zeros(m+ks-1,n+ks-1);
4     [m1 n1]=size(a1);
5     x=floor(ks/2);
6     a1(1+x:m1-x,1+x:n1-x)=a;
7     out=[];
8
9     for i=1+x:m1-x
10        for j=1+x:n1-x
11            t=a1(i-x:i+x,j-x:j+x);
12            med=median(t(:));
13            out(i-x,j-x)=med;
14        end
15    end
16 endfunction
```

Image Enhancement

Scilab code AP 14 `function [result] = izigzag5(data)`

```
2 //inverse ZigZag scanning of input data
```

```
3   N= sqrt(size(data,1));
4   z = 1;
5   count = 0;
6   row = 1;
7   col = 0;
8   for (x = 2:2*N),
9       if (x <= N+1),
10          y = x + 1;
11          if(modulo(x,2) == 0)
12            col = col + 1;
13          else
14            row = row + 1;
15          end
16      else
17          y = N+1;
18          if (modulo(x,2) == 0)
19             row = row - 1;
20             col = col + 2;
21          else
22             row = row + 2;
23             col = col - 1;
24          end
25      end
26
27      while((row < y)&(col < y)&(row > 0)&(col > 0))
28          result(row,col) = data(z);
29          z = z + 1;
30
31          if(modulo(x,2) == 0)
32            row = row - 1;
33            col = col + 1;
34          else
35            row = row + 1;
36            col = col - 1;
37          end
38      end
39  end
40  endfunction
```

Scilab code AP 15

```
1 function [result] = zigzag5(data)
2 // ZigZag scanning of input data
3 N= size(data,1);
4 z = 1;
5 count = 0;
6 row = 1;
7 col = 0;
8
9 for (x = 2:2*N),
10      if (x <= N+1)
11          y = x + 1;
12          if(modulo(x,2) == 0)
13              col = col + 1;
14          else
15               row = row + 1;
16          end
17      else
18          y = N+1;
19          if(modulo(x,2) == 0)
20              row = row - 1;
21              col = col + 2;
22          else
23              row = row + 2;
24              col = col - 1;
25          end
26      end
27
28      while((row < y)&(col < y)&(row > 0)&(col > 0))
29          result(z) = data(row,col);
30          z = z + 1;
31          if(modulo(x,2) == 0)
32              row = row - 1;
33              col = col + 1;
34          else
35              row = row + 1;
```

```
36              col = col - 1;
37          end
38      end
39  end
40  endfunction
```

Zig Zag Scanning of Pixels

Scilab code AP 16

```
1  function [a2] = fft2d(a)
2  //a = any real or complex 2D matrix
3  //a2 = 2D-DFT of 2D matrix  'a'
4  m=size(a,1)
5  n=size(a,2)
6  // fourier transform along the rows
7  for i=1:n
8  a1(:,i)=exp(-2*%i*%pi*(0:m-1)'.*.(0:m-1)/m)*a(:,i)
9  end
10 // fourier transform along the columns
11 for j=1:m
12 a2temp=exp(-2*%i*%pi*(0:n-1)'.*.(0:n-1)/n)*(a1(j,:))
       .'
13 a2(j,:)=a2temp.'
14 end
15 for i = 1:m
16     for j = 1:n
17         if((abs(real(a2(i,j)))<0.0001)&(abs(imag(a2(
               i,j)))<0.0001))
18             a2(i,j)=0;
19         elseif(abs(real(a2(i,j)))<0.0001)
20             a2(i,j)= 0+%i*imag(a2(i,j));
21         elseif(abs(imag(a2(i,j)))<0.0001)
22             a2(i,j)= real(a2(i,j))+0;
23         end
24     end
25 end
```

2D Fast Fourier Trasnform

Scilab code AP 17

```
1 function [a] =ifft2d(a2)
2 //a2 = 2D-DFT of any real or complex 2D matrix
3 //a = 2D-IDFT of a2
4 m=size(a2,1)
5 n=size(a2,2)
6 //Inverse Fourier transform along the rows
7 for i=1:n
8 a1(:,i)=exp(2*%i*%pi*(0:m-1)'.*.(0:m-1)/m)*a2(:,i)
9 end
10 //Inverse fourier transform along the columns
11 for j=1:m
12 atemp=exp(2*%i*%pi*(0:n-1)'.*.(0:n-1)/n)*(a1(j,:)).'
13 a(j,:)=atemp.'
14 end
15 a = a/(m*n)
16 a = real(a)
17 endfunction
```

Inverse 2D Fast Fourier Transform