```
function[rgbImage]=Image_compressor(k)

rgbImage= im2double(imread('img.jpg'));
redChannel = rgbImage(:, :, 1);

greenChannel = rgbImage(:, :, 2);

blueChannel = rgbImage(:, :, 3);

mean_redChannel=mean(redChannel(:));
redChannel=redChannel-mean_redChannel;

mean_greenChannel=mean(greenChannel(:));
greenChannel=greenChannel-mean_greenChannel;

mean_blueChannel=mean(blueChannel(:));
blueChannel=blueChannel-mean_blueChannel;


[coeff,score]=pca(redChannel,'Numcomponents',k);
A=reshape(score*coeff' , size(redChannel)) +mean_redChannel;

[coeff,score]=pca(greenChannel,'Numcomponents',k);
B=reshape(score*coeff' , size(greenChannel)) +mean_greenChannel;

[coeff,score]=pca(blueChannel,'Numcomponents',k);
C=reshape(score*coeff' , size(blueChannel)) +mean_blueChannel;

rgbImage = cat(3,A,B,C);

end

```

```
arr=[50 100 500 1000];
rgbImage= im2double(imread('img.jpg'));
imshow(rgbImage);
for i=1:4
    rgbImage=Image_compressor(arr(1,i));
    figure;
    imshow(rgbImage);
    title(sprintf('image using pca %d',arr(1,i)));
end

```

**image using pca 50**

**image using pca 100**

**image using pca 500**

image using pca 1000

```
1  %Power method for eigen value
2  function [] = SelfPower()
3  A=input('Enter the matrix A : \n');
4  MaxNumIter=2000;
5  len=length(A);
6  vec=ones(len,1);
7  tol=power(10,-8);
8  val1=1;
9
10 fprintf("Iteration lambda\n");
11
12 for i=1:MaxNumIter
13     v=A*vec;
14     val2=max(abs(v));
15     temp=abs(val1-val2);
16     val1=val2;
17
18     vec=v/val2;
19     if(temp<=tol*(abs(val2)))
20         break;
21     end
22
23
24     if ismember(i, [1, 2, 3, 4, 5, 10, 30, 50, 100, 200, 500, 1000, 2000])
25         fprintf('%d : %f\n', i, val2);
26     end
27
28 end
29
30 end
31
32
33
```

```
>> SelfPower
Enter the matrix A :
[1 3 -1;3 2 4;-1 4 10]
Iteration lambda
1 : 13.000000
2 : 12.538462
3 : 11.834356
4 : 11.738206
5 : 11.682595
10 : 11.662078
>>
```

```
%QR Iteration method to get eigen values of a matrix A
function[]=SelfQRIter()
A=input('Enter Matrix A: ');
temp=Inf;
maxNumIter=1000;
tol=power(10,-8);
Ak=A;
for i=1:maxNumIter
    if(temp<tol*norm(Ak,'fro'))
        break;
    end
    Ak_prev=Ak;
    [Q,R]=qr(Ak);
    Ak=R*Q;
    temp=norm(tril(Ak)-tril(Ak_prev),'fro');
end
disp('Eigen values by inbuilt eig fucntion');
eig(A)
disp('Eigen values by QR iteration method');
diag(Ak)
end


```

```
>> SelfQRIter
Enter Matrix 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 8]
Eigen values by inbuilt eig fucntion

ans =

   64.8024
  -21.6787
  -13.1557
   21.2892
   12.7429

Eigen values by QR iteration method

ans =

   64.8024
  -21.6787
   21.2892
  -13.1557
   12.7429

>>
```