

Practical 9

Aim: Image Compression

Code:

```
close;
clear;
clc;
x=[65,75,80,70;72,75,82,68;84,72,62,65;66,68,72,80];
disp(x,"Original Block is x = ");
[ m1 n1 ]=size(x);
blk = input("Enter the block size: ");
for i = 1:blk:m1
    for j = 1:blk:n1
        y=x(i:i+(blk-1),j:j+(blk-1));
        m=mean(mean(y));
        disp(m,"Mean value is m = ");
        sig=stdev(y);
        disp(sig,"Standard Deviation of the block is = ");
        b=y>m;
        disp(b,"Binary allocation matrix is B = ");
        K = sum(sum(b));
        disp(K,"number of ones = ");
        if(K~=blk^2)&(K~=0)
            ml=m-sig*sqrt(K/((blk^2)-K));
            disp(ml,"The value of a = ");
            mu=m+sig*sqrt(((blk^2)-K)/K);
            disp(mu,"The value of b = ");
            x(i:i+(blk-1),j:j+(blk-1))=b*mu+(1-b)*ml;
        end
    end
end
```

```
disp(round(x),"Reconstructed blk is x = ");
```

Output:

```

65.  75.  80.  70.
72.  75.  82.  68.
84.  72.  62.  65.
66.  68.  72.  80.

"Original Block is x = "
Enter the block size: 4

72.25

"Mean value is m = "

6.6282225

"Standard Deviation of the block is = "

F T T F
F T T F
T F F F
F F F T

"Binary allocation matrix is B = "

6.

"number of ones = "

67.115801

"The value of a = "

80.806998

"The value of b = "

67.  81.  81.  67.
67.  81.  81.  67.
81.  67.  67.  67.
67.  67.  67.  81.

"Reconstructed blk is x = "
```