Bit plane slicing

Aim:- To perform bit plane slicing on given gray scale image & perform water marking operation.

Theory:-

Bit plane slicing:

- **It indicates the contribution made by specific bits to the total image.**

The gray level of each pixel in a digital image is stored as one or more bytes in a computer. For an 8-bit image, 0 is encoded as 00000000 and 255 is encoded as 11111111. Any number between 0 t0 255 is encoded as one byte. The bit in the far-left side is referred as the most significant bit (MSB) because a change in that bit would significantly change the value encoded by the byte. The bit in the far right is referred as the least significant bit (LSB), because a change in this bit does not change the encoded gray value much. The bit plane representation of an eight-bit digital image is given by:

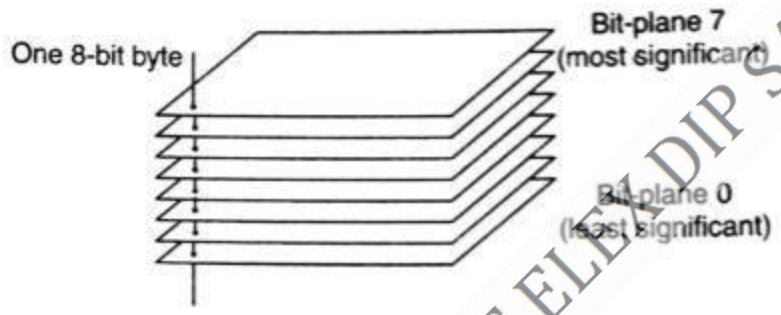


Fig no: 7

Bit plane slicing is a method of representing an image with one or more bits of the byte used for each pixel. One can use only MSB to represent the pixel, which reduces the original gray level to a binary image. The three main goals of bit plane slicing is:

- Converting a gray level image to a binary image.
- Representing an image with fewer bits and corresponding the image to a smaller size
- Enhancing the image by focusing.

6	7	6	6	7
0	0	0	1	2
1	1	1	2	3
4	5	5	4	2
6	6	6	7	7

Bit plane slicing:

Since the given image has a maximum grey level of 7, it is a 3-bit image. We convert the image to binary and separate the bit planes.

110	111	110	110	111
000	000	000	001	010
001	001	001	010	011
100	101	101	100	010
110	110	110	111	111

Separating the bit planes, we obtain

1	1	1	1	1
0	0	0	0	0
0	0	0	0	0
1	1	1	1	0
1	1	1	1	1

MSB plane

1	1	1	1	1
0	0	0	0	1
0	0	0	1	1
0	0	0	0	1
1	1	1	1	1

Centre bit plane

0	1	0	0	1
0	0	0	1	0
1	1	1	0	1
0	1	1	0	0
0	0	0	1	1

LSB plane

Application of Bit Plane Slicing

- **To convert a gray image to a set of binary images which may be more suitable for further processing (e.g. compression)**
- **Water Marking**
- **Steganography**

Implementation Instructions: -

Bit plane slicing:

1) Read the given gray scale image.
2) Convert pixel values to binary equivalent.
3) Separate nth bit of all the pixels as nth plane & save it as new image. Convert it to 8-bit number.
4) Perform above operation on all 8 bits of the pixels to generate 8-bit planes.
5) Display all the 8-bit planes.

Water Marking:

1) Perform bit plane slicing on the main image & image to be embedded.
2) Replace 4 LSBs of all pixels of main image with 4 MSBs of corresponding pixels of embed image.
3) Display the Water marked image.

Bit Plane Slicing

Code

Editor - C:\Users\djsce.student\Documents\B2-9497\exp2_dip.m

exp2_dip.m

```
%Experiment 2 - Bit Plane Slicing
%Krisha Lakhani - 60001200097
%Aryan Potdar - 60001200092
% clearing the output screen
clc;

% reading image's pixel in c
c = imread('cameraman.tif');

% storing image information in cd
cd = double(c);

% extracting all bit one by one
% from 1st to 8th in variable
% from c1 to c8 respectively
c1 = mod(cd, 2);
c2 = mod(floor(cd/2), 2);
c3 = mod(floor(cd/4), 2);
c4 = mod(floor(cd/8), 2);
c5 = mod(floor(cd/16), 2);
c6 = mod(floor(cd/32), 2);
c7 = mod(floor(cd/64), 2);
c8 = mod(floor(cd/128), 2);

% combining image again to form equivalent to original grayscale image
cc = (2 * (2 * (2 * (2 * (2 * (2 * (2 * c8 + c7) + c6) + c5) + c4) + c3) + c2) + c1);

% plotting original image in first subplot
subplot(2, 5, 1);
imshow(c);
title('Original Image');
```

```
% plotting binary image having extracted bit from 1st to 8th
% in subplot from 2nd to 9th
subplot(2, 5, 2);
imshow(c1);
title('Bit Plane 1');
subplot(2, 5, 3);
imshow(c2);
title('Bit Plane 2');
subplot(2, 5, 4);
imshow(c3);
title('Bit Plane 3');
subplot(2, 5, 5);
imshow(c4);
title('Bit Plane 4');
subplot(2, 5, 6);
imshow(c5);
title('Bit Plane 5');
subplot(2, 5, 7);
imshow(c6);
title('Bit Plane 6');
subplot(2, 5, 8);
imshow(c7);
title('Bit Plane 7');
subplot(2, 5, 9);
imshow(c8);
title('Bit Plane 8');

% plotting recombined image in 10th subplot
subplot(2, 5, 10);
imshow(uint8(cc));
title('Recombined Image');
```

Output

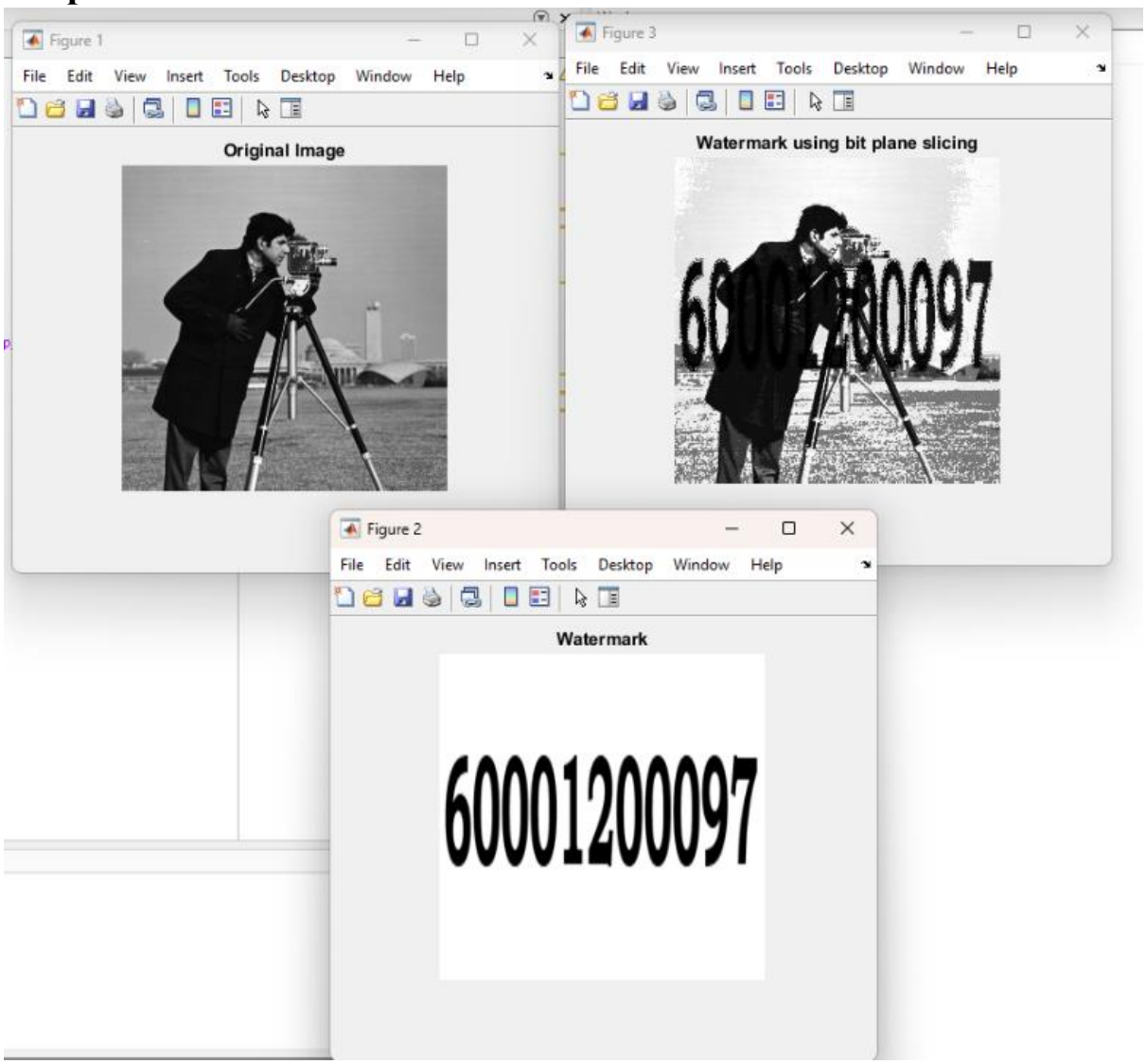
Watermark

Code

exp2a_dip.m | exp2b_dip.m | untitled2 *

```
%Experiment 2: Watermarking
%Krisha Lakhani - 60001200097
clc;
clear all;

a = imread('cameraman.tif');
f1 = figure;
b = a;

%%Original Image%%
f1 = imshow(a)
title('Original Image')

%%Watermark Resizing%%
I = imread('C:\Users\djsce.student\Downloads\sap_pic_white.png');
m = imresize(I,[256,256])
f2 = figure;
f2 = imshow(m)
title('Watermark')

%%Watermark using bit plane slicing%%
b=a;
% [k,l] = size(a)
for i=1:1:256
    for j= 1:1:256
        k = b(i,j);
        l = m(i,j);
        bb = dec2bin(k);
        wm = dec2bin(l,7);
        bb(1) = wm(4);
        bb(2) = wm(5);
        bb(3) = wm(6);
        k = bin2dec(bb);
        b(i,j) = k;
    end
end
f3 = figure;
f3 = imshow(b)
title('Watermark using bit plane slicing')
```

Output



Conclusion:

We have completed bit plane slicing experiment along with adding a watermark to the image.