
```
% Bit Plane Slicing
% Created on: 23/01/25
% By Author: Supratit Datta, BT22ECE127

% Dynamic Input
[file, path] = uigetfile({'*.*', 'All Image Files';
 '*.png;*.jpg;*.jpeg;*.bmp', 'Supported Image Formats'});
if isequal(file, 0)
    disp('No file selected');
    return;
end
inputImage = imread(fullfile(path, file));

% Convert the image to grayscale if it is a color image
if size(inputImage, 3) == 3
    grayImage = rgb2gray(inputImage);
else
    grayImage = inputImage;
end

% Display the original grayscale image
figure;
subplot(3, 3, 1);
imshow(grayImage, []);
title('Original Image');

% Initialize a cell array to store bit planes
bitPlanes = cell(1, 8);

% Perform bit-plane slicing
for i = 1:8
    % Extract the i-th bit plane
    bitPlanes{i} = bitget(grayImage, i) * 255;

    % Display the bit plane
    subplot(3, 3, i+1); % Place in the subplot
    imshow(bitPlanes{i}, []);
    title(['Bit Plane ', num2str(i)]);
end

% Reconstruct the image by removing the LSB (set the 1st bit to 0)
% Perform bit manipulation to remove the 1st bit (LSB)
reconstructedImage = bitand(grayImage, bitcmp(1, 'uint8'));

% Display the original and reconstructed images in a new figure
figure;
% Original grayscale image
subplot(2, 1, 1);
imshow(grayImage, []);
title('Original Grayscale Image');

% Reconstructed image without LSB
```

```
subplot(2, 1, 2);  
imshow(reconstructedImage, []);  
title('Reconstructed Image (No LSB)');
```

Original Image



Bit Plane 1



Bit Plane 2



Bit Plane 3



Bit Plane 4



Bit Plane 5



Bit Plane 6



Bit Plane 7



Bit Plane 8



Original Grayscale Image



Reconstructed Image (No LSB)

