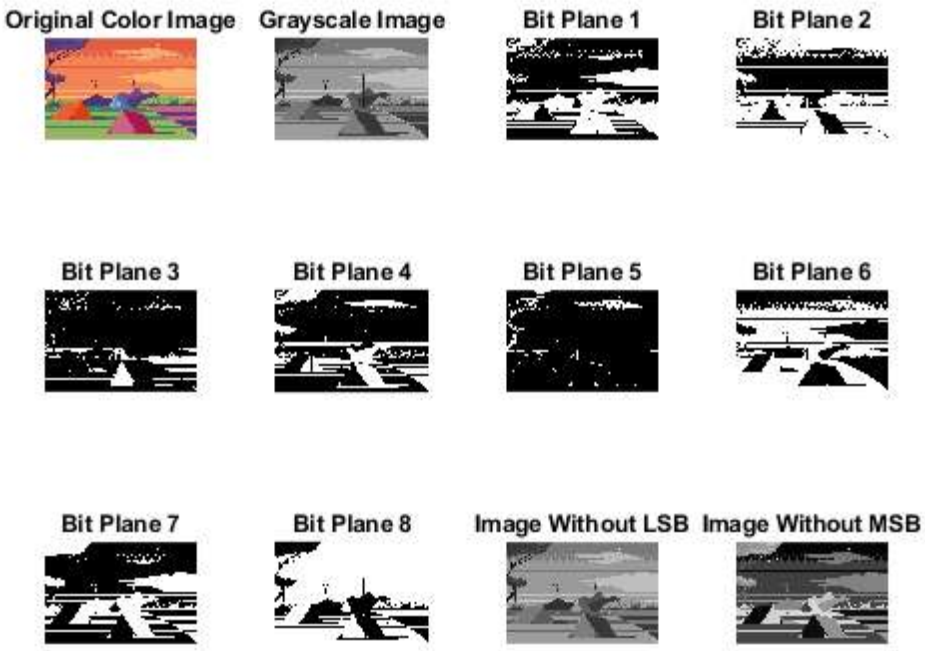
```
%Created by Soham Roy

% MATLAB code for bit plane slicing of an 8-bit image

% Read the image
image = imread('test.jpg');

% Display the original color image
figure;
subplot(3, 4, 1);
imshow(image);
title('Original Color Image');

% Convert the image to grayscale
gray_image = rgb2gray(image);
subplot(3, 4, 2);
imshow(gray_image, []);
title('Grayscale Image');

% Get the size of the grayscale image
[rows, cols] = size(gray_image);

% Initialize a cell array to hold each bit plane
bit_planes = cell(1, 8);

% Extract each bit plane
for bit = 1:8
    bit_planes{bit} = bitget(gray_image, bit); % Extract the bit plane using bitget
end

% Display all 8 bit planes
for bit = 1:8
    subplot(3, 4, bit + 2);
    imshow(logical(bit_planes{bit}));
    title(['Bit Plane ', num2str(bit)]);
end

% Combine higher-order bit planes (excluding the least significant bit, i.e., 1st bit)
combined_image1 = uint8(0);
for bit = 2:8
    combined_image1 = combined_image1 + uint8(bit_planes{bit} * 2^(bit - 1));
end

% Combine lower-order bit planes (excluding the most significant bit, i.e., 7st bit)
combined_image2 = uint8(0);
for bit = 1:7
    combined_image2 = combined_image2 + uint8(bit_planes{bit} * 2^(bit - 1));
end

% Display the combined image without the least significant bit
subplot(3, 4, 11);
imshow(combined_image1, []);
title('Image Without LSB');

% Display the combined image without the most significant bit
subplot(3, 4, 12);
imshow(combined_image2, []);
title('Image Without MSB');
```



Published with MATLAB® R2021a