

```

% date 25/01/2026
% created by - Saurabh Singh

clc;
clear;
close all;

% Read a color image
I = imread('I.jpg');

figure;
imshow(I);
title('Original Color Image');

% Convert RGB image to grayscale
G = rgb2gray(I);
[r, c] = size(G);

figure;
imshow(G);
title('Grayscale Image');

% Convert grayscale image into column vector
v = G(:);
d = v;

% Find unique gray levels
s = unique(v);

% Calculate frequency of each symbol
f = histcounts(v, [s; max(s)+1]);

% Calculate probability
p = f / sum(f);

% Sort symbols based on probability
[p_s, id] = sort(p, 'descend');
s_s = s(id);

% Shannon code length calculation
L = ceil(-log2(p_s));

% Generate Shannon codes
sc = cell(length(s_s),1);
cp = 0;

for i = 1:length(s_s)
    sc{i} = dec2bin(floor(cp * 2^L(i)), L(i));
    cp = cp + p_s(i);
end

% Encoding
eb = cell(length(v),1);

for i = 1:length(v)
    k = find(s_s == v(i), 1);
    eb{i} = sc{k};
end

```

```

% Final encoded bitstream
es = [eb{:}];

% Reconstructed image
R = reshape(v, r, c);

figure;
imshow(R);
title('Reconstructed Image after Shannon Coding');

% Compression calculation
ob = numel(v) * 8;
cb = length(es);
cr = ob / cb;

fprintf('By using Shannon Method\n');
fprintf('Original size (bits) : %d\n', ob);
fprintf('Compressed size (bits) : %d\n', cb);
fprintf('Compression Ratio : %.2f\n', cr);

% ----- HUFFMAN CODING -----
pv = unique(d);
pc = histc(d, pv);
pp = pc / sum(pc);

% Create Huffman dictionary
hd = huffmandict(pv, pp);

% Encode using Huffman coding
hes = huffmanenco(d, hd);

% Decode
dd = huffmandeco(hes, hd);

% Convert decoded data back into image form
Ri = uint8(reshape(dd, r, c));

figure;
imshow(Ri);
title('Reconstructed Image after Huffman Coding');

% Compression ratio
ob = numel(d) * 8;
cb = length(hes);
cr = ob / cb;

fprintf('By using Huffman Method\n');
fprintf('Original size (bits) : %d\n', ob);
fprintf('Compressed size (bits) : %d\n', cb);
fprintf('Compression Ratio : %.2f\n', cr);

```

By using Shannon Method
Original size (bits) : 3180640
Compressed size (bits) : 3057614
Compression Ratio : 1.04
By using Huffman Method
Original size (bits) : 3180640

Compressed size (bits) : 2856436
Compression Ratio : 1.11

Original Color Image



Grayscale Image



Reconstructed Image after Shannon Coding



Reconstructed Image after Huffman Coding

