

```

clear;
clc;
close all;

%Change this value to change compression level%
%Image is only smaller below 11%
%Max achievable compression is 4X%
comp_ratio = 100;

[file, folder] = uigetfile('*');
img = imread( [folder,file] );

s1 = whos( 'img' );
img_size = s1.bytes;

figure;
imshow( img );
title( 'Original Image' );

%% Compress %%
img_r = img(:,:,1);
img_g = img(:,:,2);
img_b = img(:,:,3);

[w,h] = size( img_r );

img_r_fft = fft2( img_r );
img_g_fft = fft2( img_g );
img_b_fft = fft2( img_b );

% Filter Red Frequencies %
r1 = reshape( img_r_fft, [1, w*h] );
r1 = sort( abs(r1) );
comp_range = 1 - 1/comp_ratio;
comp_index = comp_range*length(r1);
comp_index = round( comp_index );
lim = r1( comp_index );

r_mat = abs(img_r_fft) > lim;
r_val = img_r_fft( r_mat );

% Filter Green Frequencies %
g1 = reshape( img_g_fft, [1, w*h] );
g1 = sort( abs(g1) );
comp_range = 1 - 1/comp_ratio;
comp_index = comp_range*length(g1);
comp_index = round( comp_index );
lim = g1( comp_index );

g_mat = abs(img_g_fft) > lim;
g_val = img_g_fft( g_mat );

% Filter Blue Frequencies %
b1 = reshape( img_b_fft, [1, w*h] );
b1 = sort( abs(b1) );

```

```

comp_range = 1 - 1/comp_ratio;
comp_index = comp_range*length(b1);
comp_index = round( comp_index );
lim = b1( comp_index );

b_mat = abs(img_b_fft) > lim;
b_val = img_b_fft( b_mat );

%Data needed to reconstruct image:
% w, h, r_mat, r_val, g_mat, g_val, b_mat, b_val %

%% Size Calculation %%
w0 = cast( w, 'uint16' );
h0 = cast( h, 'uint16' );

s0 = whos( 'w0' );
s1 = whos( 'h0' );

size_new = s0.bytes + s1.bytes;

%index bits size%
size_new = 3*(w*h)/4 + size_new;

%Data Size%
r_val = cast( r_val, 'single' );
g_val = cast( g_val, 'single' );
b_val = cast( b_val, 'single' );

sr = whos( 'r_val' );
sg = whos( 'g_val' );
sb = whos( 'b_val' );

size_new = size_new + sr.bytes + sg.bytes + sb.bytes;

comp = img_size/size_new;
fprintf( 'Image Compressed at Compression Ratio of %f\n', comp );

%% Reconstruct Image %%
img_r_newf = zeros( w, h );
img_b_newf = zeros( w, h );
img_g_newf = zeros( w, h );

img_r_newf( r_mat ) = r_val;
img_b_newf( b_mat ) = b_val;
img_g_newf( g_mat ) = g_val;

img_r_new = ifft2( img_r_newf );
img_b_new = ifft2( img_b_newf );
img_g_new = ifft2( img_g_newf );

img_new = zeros( w, h, 3 );

img_new(:, :, 1) = img_r_new;
img_new(:, :, 2) = img_g_new;
img_new(:, :, 3) = img_b_new;

img_new = cast( img_new, 'uint8' );

```

```
figure;  
imshow( img_new );  
str = sprintf( 'Image Compressed %f Times' , comp );  
title( str );
```