ACCESSING IMAGES FROM THE FOLDER AND STORING IT IN A STRUCTURE

```
mydir='/MATLAB Drive/Published/Birds-1/Class-1';
fileformat='*.jpg';
dd=dir(fullfile(mydir,fileformat));
assert(numel(dd) > 0, 'No file was found. Check that the path is correct');
my_img = struct('img', cell(size(dd)));
for zz=1:numel(dd)
    my_img(zz).img = imread(fullfile(mydir,dd(zz).name));
end
```

```
% CREATING A STRUCTURE FOR R COMPONENT
r_img = struct('img', cell(size(dd)));
for zz=1:numel(dd)
    r_img(zz).img = my_img(zz).img(:,:,1);
end

%CREATING A STRUCTURE FOR G COMPONENT
g_img = struct('img', cell(size(dd)));
for zz=1:numel(dd)
    g_img(zz).img = my_img(zz).img(:,:,2);
end

%CREATING A STRUCTURE FOR B COMPONENT
b_img = struct('img', cell(size(dd)));
for zz=1:numel(dd)
    r_img(zz).img = my_img(zz).img(:,:,3);
end

%CREATING A STRUCTURE FOR GRAY SCALE VERSION
gray_img = struct('img', cell(size(dd)));
for zz=1:numel(dd)
    gray_img(zz).img = rgb2gray(my_img(zz).img);
end

%CREATING A STRUCTURE FOR SHARPENED
edge_img = struct('img', cell(size(dd)));
for zz=1:numel(dd)
    edge_img(zz).img = fourrier(gray_img(zz).img,0.09,4);
end
```

```
%CODE TO EXTRACT FEACTURES IN THE FREQUENCY DOMAIN AFTER APPLYING FFT
%fft
%gray
for i=1:numel(dd)
    current=gray_img(i).img;     %r_img(i).img

    %fourier transform
    fft_img=fft2(current);

    %statistical measures
```

```
    av=real(mean(mean(fft_img)));
    med=real(median(median(fft_img)));
    st_dev=real(std(std(double(fft_img))));
    max_=real(max(max(fft_img)));
    min_=real(min(min(fft_img)));

    %Column Values
    rgb=[av,med,st_dev,max_,min_];
    writematrix(rgb,'IVA_FFT_gray.csv','WriteMode', 'append');
end
```

```
%CODE TO EXTRACT FEACTURES IN THE FREQUENCY DOMAIN AFTER APPLYING DCT
%dct
%gray
for i=1:numel(dd)
    current=gray_img(i).img; %r_img(i).img


    %dct
    dct_img=dct2(current);
    dc=dct_img(1,1);
    writematrix(dc,'IVA_DCT_gray.csv','WriteMode', 'append')
end
```

```
%CODE TO EXTRACT FEACTURES IN THE FREQUENCY DOMAIN AFTER APPLYING WAVELET TRANSFORM
%wavelet
%gray
for i=1:numel(dd)
    current=gray_img(i).img; %r_img(i).img

    %WAVELET transform
    wave_img=wave(current,'haar',3);

    %statistical measures
    av=real(mean(mean(wave_img)));
    med=real(median(median(wave_img)));
    st_dev=real(std(std(double(wave_img))));
    max_=real(max(max(wave_img)));
    min_=real(min(min(wave_img)));

    %Column Values
    rgb=[av,med,st_dev,max_,min_];
    writematrix(rgb,'IVA_wavelet_gray.csv','WriteMode', 'append');
end
```