Feature Extraction using MATLAB

1. Means of Red, Green and Blue.
2. Standard Deviation
3. Mean
4. Median
5. Maximum
6. Minimum
7. Entropy
8. Above & Below RGB: The number of pixels which are above and below the average values of Red, Green and Blue components separately.

The process is done for images after Noise Reduction and Edge Detection

```
mydir='/MATLAB Drive/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)));
k=numel(dd)+1;
for zz=1:numel(dd)
 my_img(zz).img = imread(fullfile(mydir,dd(zz).name));
end
```

Ordinary Images

```
for i=1:numel(dd)
    current=imresize(my_img(i).img,[400,400]);
    %rgb means
    r=mean(mean(current(:,:,1)));
    g=mean(mean(current(:,:,2)));
    b=mean(mean(current(:,:,3)));
    %grayscale
    g_img=double(rgb2gray(current));
    %statistical measures
    av=mean(mean(g_img));
    med=median(median(g_img));
    st_dev=std(std(double(g_img)));
    max_=max(max(g_img));
    min_=min(min(g_img));
    [M,N]=size(g_img);
    area=M*N;
    %entropy values
    e=entropy(g_img);

    %Above & Below


     c_r_1=0;
     c_r_2=0;
     c_g_1=0;
     c_g_2=0;
     c_b_1=0;
     c_b_2=0;

    for i = 1:M
        for j = 1:N
            if(current(i,j,1)>r)
                c_r_1=c_r_1+1;
            end
            if(current(i,j,2)>g)
                c_g_1=c_g_1+1;
            end
            if(current(i,j,3)>b)
                c_b_1=c_b_1+1;
            end
            if(current(i,j,1)<r)
                c_r_2=c_r_2+1;
            end
            if(current(i,j,2)<g)
                c_g_2=c_g_2+1;
            end
            if(current(i,j,1)<b)
                c_b_2=c_b_2+1;
            end
        end
    end
    %Column Values

rgb=[r,g,b,av,med,st_dev,max_,min_,area,e,c_r_1,c_r_2,c_g_1,c_g_2,c_b_1,c_b_2
];
    %Writing into Excel Sheets
    writematrix(rgb,'IVA.csv','WriteMode', 'append');

end
```

Noise Reduction: Median Filtering

```
for i=1:numel(dd)
```

```
    current=imresize(my_img(i).img,[400,400]);
    %rgb means
    r=mean(mean(current(:,:,1)));
    g=mean(mean(current(:,:,2)));
    b=mean(mean(current(:,:,3)));
    %grayscale
    g_img1=rgb2gray(current);
    g_img=double(medfilt2(g_img1));
    %statistical measures
    av=mean(mean(g_img));
    med=median(median(g_img));
    st_dev=std(std(double(g_img)));
    max_=max(max(g_img));
    min_=min(min(g_img));
    [M,N]=size(g_img);
    area=M*N;
    %entropy values
    e=entropy(g_img);
    %Above & Below
    c_r_1=0;
    c_r_2=0;
    c_g_1=0;
    c_g_2=0;
    c_b_1=0;
    c_b_2=0;

    %Above & Below
    for i = 1:M
        for j = 1:N
            if(current(i,j,1)>r)
                c_r_1=c_r_1+1;
            end
            if(current(i,j,2)>g)
                c_g_1=c_g_1+1;
            end
            if(current(i,j,3)>b)
                c_b_1=c_b_1+1;
            end
            if(current(i,j,1)<r)
                c_r_2=c_r_2+1;
            end
            if(current(i,j,2)<g)
                c_g_2=c_g_2+1;
            end
            if(current(i,j,1)<b)
                c_b_2=c_b_2+1;
            end
        end
    end
    %Column Values
```

```
rgb1=[r,g,b,av,med,st_dev,max_,min_,area,e,c_r_1,c_r_2,c_g_1,c_g_2,c_b_1,c_b_
2];
    %Writing into Excel Sheets
    writematrix(rgb1,'IVA1.csv','WriteMode', 'append');
end
```

Edge Detection: Canny Filter

```
for i=1:numel(dd)
    c_r_1=0;
    c_r_2=0;
    c_g_1=0;
    c_g_2=0;
    c_b_1=0;
    c_b_2=0;
    current=imresize(my_img(i).img,[400,400]);
    %rgb means
    r=mean(mean(current(:,:,1)));
    g=mean(mean(current(:,:,2)));
    b=mean(mean(current(:,:,3)));
    %grayscale
    g_img1=rgb2gray(current);
    g_img=double(edge(g_img1,'Canny',0.2));
    %statistical measures
    av=mean(mean(g_img));
    med=median(median(g_img));
    st_dev=std(std(double(g_img)));
    max_=max(max(g_img));
    min_=min(min(g_img));
    [M,N]=size(g_img);
    area=M*N;
    %entropy values
    e=entropy(g_img);

    %Above & Below
    for i = 1:M
        for j = 1:N
            R=current(i,j,1);
            G=current(i,j,2);
            B=current(i,j,3);
            if(R>r)
                c_r_1=c_r_1+1;
            end
            if(G>g)
                c_g_1=c_g_1+1;
            end
            if(B>b)
                c_b_1=c_b_1+1;
            end
            if(R<r)
                c_r_2=c_r_2+1;
            end
            if(G<g)
                c_g_2=c_g_2+1;
            end
            if(B<b)
                c_b_2=c_b_2+1;
            end
        end
    end
    %Column Values

rgb1=[r,g,b,av,med,st_dev,max_,min_,area,e,c_r_1,c_r_2,c_g_1,c_g_2,c_b_1,c_b_
2];
    %Writing into Excel Sheets
    writematrix(rgb1,'IVA2.csv','WriteMode', 'append');
end
```

```
    c_r_1=0;
    c_r_2=0;
    c_g_1=0;
    c_g_2=0;
    c_b_1=0;
    c_b_2=0;
    current=imresize(my_img(2).img,[400,400]);
    %rgb means
    r=mean(mean(current(:,:,1)));
    g=mean(mean(current(:,:,2)));
    b=mean(mean(current(:,:,3)));
    %grayscale
    g_img1=rgb2gray(current);
    g_img=double(edge(g_img1,'Canny',0.2));
    %statistical measures
    av=mean(mean(g_img));
    med=median(median(g_img));
    st_dev=std(std(double(g_img)));
    max_=max(max(g_img));
    min_=min(min(g_img));
    [M,N]=size(g_img);
    area=M*N;
    %entropy values
    e=entropy(g_img);
    %Above & Below
    for i = 1:M
        for j = 1:N
            R=current(i,j,1);
            G=current(i,j,2);
            B=current(i,j,3);
            if(R>r)
                c_r_1=c_r_1+1;
            end
            if(G>g)
                c_g_1=c_g_1+1;
            end
            if(B>b)
                c_b_1=c_b_1+1;
            end
            if(R<r)
                c_r_2=c_r_2+1;
            end
            if(G<g)
                c_g_2=c_g_2+1;
            end
            if(B<b)
                c_b_2=c_b_2+1;
            end
        end
    end
    %Column Values

rgb1=[r,g,b,av,med,st_dev,max_,min_,area,e,c_r_1,c_r_2,c_g_1,c_g_2,c_b_1,c_b_
2]
```