
```
function main()
```

MAIN

Summary of this function goes here.

- Syntax

[] = MAIN()

- Examples:

Provide sample usage code here

- See also:

List related files here

- Author: Dmitrii Leliuhin
- Email: dleliuhin@mail.ru
- Date: 25/12/2018 19:30:05
- Version: 1.0 \$
- Requirements: PCWIN64, MatLab R2016a
- Warning:
 1. Warnings list.
- TODO:
 1. TODO list.

Code

```
clc;
clear;
close all;

addpath(strcat(pwd, '\images\original'), ...
        strcat(pwd, '\images\normalized'), '-end');
savepath;

% Reading original image files from folders.
f = dir('images/original/*.jpg');

% Creating array of image file names.
fileName = {f.name};

% Sort fileName using natuaral sorting.
```

```

fileName = natsortfiles(fileName);

se = strel('disk',15);

% ImageMaxPixels = ones(n,m)*255;
%
% rr = I(:,:,1);
% gr = I(:,:,2);
% br = I(:,:,3);
%
% rz = zeros(n,m);
% gz = zeros(n,m);
% bz = zeros(n,m);
%
% for i = 1:n
%     for j = 1:m
%         rz(i,j) = rr(i,j) / ImageMaxPixels(i,j);
%         gz(i,j) = gr(i,j) / ImageMaxPixels(i,j);
%         bz(i,j) = br(i,j) / ImageMaxPixels(i,j);
%     end
% end
%
% r = rz./(rz+gz+bz);
% g = gz./(rz+gz+bz);
% b = bz./(rz+gz+bz);

for i=1:length(fileName)

    I = imread(fullfile('images/original', fileName{i}));

    [n, m, k] = size(I);

    r = I(:,:,1);
    g = I(:,:,2);
    b = I(:,:,3);

    figure;
    % Maximize the figure.
    set(gcf, 'Position', get(0, 'ScreenSize'));

    subplot(4, 4, 1);
    imshow(I);
    drawnow;
    title('Original');

    subplot(4, 4, 2);
    G = rgb2gray(I);
    imshow(G);
    drawnow;
    title('Grayscaled');

    subplot(4, 4, 3);
    G = imgaussfilt(G);
    imshow(G);

```

```
drawnow;
title('Gauss Filtered');

subplot(4, 4, 4);
G = imadjust(G);
imshow(G);
drawnow;
title('Adjusted');

subplot(4, 4, 5);
NDI = (g+r)./(g-r);
imshow(NDI);
drawnow;
title('NDI = (G+R)/(G-R)');

subplot(4, 4, 6);
EG = (2).*g-r-b;
imshow(EG);
drawnow;
title('E*G = 2*G-R-B');

subplot(4, 4, 7);
ER = (1.4).*r-g;
imshow(ER);
drawnow;
title('E*R = 1.4*R-G');

subplot(4, 4, 8);
EGER = EG-ER;
imshow(EGER);
drawnow;
title('E*G - E*R');

subplot(4, 4, 9);
% NDI = imgaussfilt(NDI);
grayLvl = graythresh(NDI);
NDIOtsu = imbinarize(NDI, grayLvl);
NDIOtsu = imcomplement(NDIOtsu);
NDIOtsu = bwmorph(NDIOtsu, 'erode', 2);
stats = regionprops(NDIOtsu, 'Area');
max_blob = max( [stats.Area] );
NDIOtsu = bwareaopen(NDIOtsu, max_blob);
imshow(NDIOtsu);
drawnow;
title('NDI + Otsu Binary Image');

subplot(4, 4, 10);
% EG = imgaussfilt(EG);
grayLvl = graythresh(EG);
EGotsu = imbinarize(EG, grayLvl);
imshow(EGotsu);
drawnow;
title('E*G + Otsu Binary Image');
```

```

subplot(4, 4, 12);
EGER = imgaussfilt(EGER);
EGER = imbinarize(EGER, 0);
    EGER = bwmorph(EGER, 'erode', 4);
    stats = regionprops(EGER, 'Area');
    max_blob = max( [stats.Area] );
    EGER = bwareaopen(EGER, max_blob);
imshow(EGER);
drawnow;
title('E*G-E*R Binary Image');

EGER = NDIOtsu;

orientation = regionprops(EGER, 'Orientation');

subplot(4, 4, 13);
EGERC = imrotate(EGER, ...
    abs(orientation(1).Orientation) - 90, ...
    'bilinear', ...
    'crop');
imshow(EGERC);
drawnow;
title('Rotated Binary Image');

% height = regionprops(EGERC, 'MajorAxisLength');
% width = regionprops(EGERC, 'MinorAxisLength');
% pixelList = regionprops(EGERC, 'PixelList');

subplot(4, 4, 14);
IC = imrotate(I, ...
    abs(orientation.Orientation(1)) - 90, ...
    'bilinear', ...
    'crop');
imshow(IC);
drawnow;
title('Rotated Original Image');

extremums = regionprops(EGERC, 'Extrema');

[row,col] = find(EGERC);

% Find top left and bottom right corners
top_row = min(row);
top_col = min(col);
bottom_row = max(row);
bottom_col = max(col);

centers = regionprops(EGERC, 'Centroid');
height = length(top_row:bottom_row);
width = length(top_col:bottom_col);

left_edge = floor(centers(1).Centroid(1) - 50);
right_edge = floor(centers(1).Centroid(1) + 50);
top_edge = floor(centers(1).Centroid(2) - 150);

```

```
bottom_edge = floor(centers(1).Centroid(2) + 150);

subplot(4, 4, 16);
% Crop the image
RES = imcrop(IC, [left_edge, top_edge, ...
                  right_edge - left_edge - 1, ...
                  bottom_edge - top_edge - 1]);

imshow(RES);
drawnow;
title('Cropped Original Image');

imwrite(RES, fullfile('images', 'normilized', ...
                      strcat('norm_', num2str(i), '.jpg')));

clear I G r g b centers EG EGER EGERC EGotsu ER extremums IC NDI;
clear NDIotsu orientation RES stats col row ;
close all;
end
end
```

Published with MATLAB® R2016a