ELEC319 Assessment 1 Typical Answer

Assessment Criteria: The following variables were checked in the assessment

1. thresh_image
2. bgremove_image
3. hilb_image
4. ln_image
5. std_image
6. B1
7. let_T
8. B11
9. let_t
10. bound
11. skel
12. f
13. h2
14. im6_filt1
15. im6_filt2

The following is the code:

```
% Question 1: Image Pre-Processing Tests
% DON'T CHANGE VARIABLE NAMES


% Read image bone.tif
image = imread('bone.tif');
colormap(gray);




subplot(2,3,1),
% Use display image with scaled colors command (not 'imshow'). Put title
'Original'
% Add instruction(s) here:
imagesc(image), title ('Original');


% 1 - Image Thresholding Test
% Determine automatic threshold value 'thresh' by MATLAB itself, then use this
threshold
% value to get the threshold image 'thresh_image'


% Add instruction(s) here:
thresh = graythresh(image);
```

```
thresh_image = im2bw(image, thresh);


subplot(2,3,2),
% Use display image with scaled colors command (not 'imshow'). Put title
'Threshold'
% Add instruction(s) here:
imagesc(thresh_image), title ('Threshold');


% 2 - Image Background Removal Test
% Remove image background by calculating image mean 'average_image' then
% subtract it from the original image 'image' to get 'bgremove_image'
% Add instruction(s) here:
average_image = mean(mean(image));
bgremove_image = image - average_image;


subplot(2,3,3),
% Use display image with scaled colors command (not 'imshow'). Put title 'Bg
remove'
% Add instruction(s) here:
imagesc(bgremove_image), title ('Bg remove');

% 3 - Image Hilbert Transform Test
% Apply Hilbert transform on the image 'image' to get 'hilb_image'
% Note: Hilbert transform generates complex numbers outputs


% Add instruction(s) here:
hilb_image = abs(hilbert(image));


subplot(2,3,4),
% Use display image with scaled colors command (not 'imshow'). Put title
'Hilbert'
% Add instruction(s) here:
imagesc(hilb_image), title ('Hillbert');


% 4 - Image Natural Log Test
% Apply Natural Log function on the image 'image' to get 'ln_image'
% Note: You need to use mat2gray to convert the format of the image first
% Also, change negative values to positive for display purposes
% Add instruction(s) here:
ln_image = abs(log(mat2gray(image)));
```

```
subplot(2,3,5),
% Use display image with scaled colors command (not 'imshow'). Put title as
'Natural Log'
% Add instruction(s) here:
 imagesc(ln_image), title ('Natural Log');


% 5 - Image standard deviation Test
% Apply standard deviation filter on the image 'image' to get 'std_image'
% Add instruction(s) here:
std_image = stdfilt(image);


subplot(2,3,6),
% Use display image with scaled colors command (not 'imshow'). Put title as
'Std Dev'
% Add instruction(s) here:
imagesc(std_image), title ('Std Dev');


% ---------------------------------------------------------------------------
--------


% Question 2: Detect a letter in a text image
% DON'T CHANGE VARIABLE NAMES


% Read image text.png
im = imread('text.png');


% 1- It is required to detect upper case T in the text image (horizontal text
only)
% Use suitable structural elements (B1 and B2) of size 5x5 to do that


% Define the first structural element
% Add instruction(s) here:
B1 = [1 1 1 1 1;
      1 1 1 1 1;
      1 1 1 1 1;
      0 1 1 1 0;
      0 1 1 1 0];
```

```
% Define The second structural element
% Add instruction(s) here:
 B2 = ~B1;

 % Detect Upper case Letter T in the image. The output should be in 'Let_T'

% Add instruction(s) here:
let_T = bwhitmiss(im,B1, B2);


% Demonstrating results
figure, subplot(2,2,1)
imshow(im); title('Image');
subplot(2,2,2);
imshow(B1); title('B1');
subplot(2,2,3)
imshow(B2); title('B2');
subplot(2,2,4)
imshow(let_T); title('Letter T');


% 2- It is required to detect lower case t in the text image (horizontal text
only)
% Use suitable structural elements (B11, B22) of size 5x5 to do that


% Define The first structural element
% Add instruction(s) here:
B11 = [0 0 0 0 0;
       0 0 0 1 0;
       0 0 1 1 0;
       0 1 1 1 0;
       1 1 1 1 1];




% Define the second structural element
% Add instruction(s) here:
B22 = ~B11;

% Detect lower case t in the text image


% Add instruction(s) here:
let_t = bwhitmiss(im,B11, B22);
```

```
% Demonstrating results
figure, subplot(2,2,1)
imshow(im); title('Image');
subplot(2,2,2);
imshow(B11); title('B1');
subplot(2,2,3)
imshow(B22); title('B2');
subplot(2,2,4)
imshow(let_t); title('Letter t');


% ----------------------------------------------------------------------------
--------
% Question 3: Boundary Detection and Skeletonisation using morphological
operations
% DON'T CHANGE VARIABLE NAMES


% 1- Boundary Detection
% Read image cameraman.tif
image1 = imread('cameraman.tif');


% Define the structural element 3x3 cross or '+'
% Add instruction(s) here:


s = [0 1 0;
     1 1 1;
     0 1 0];


% Extract boundary of the image using suitable morphological operation and
% any additional mathematical operation. The output must be in 'bound'


% Add instruction(s) here:
bound = image1 - imerode(image1,s);


% Plotting results
figure, subplot(2,2,1), imshow(image1), title('Original');
subplot(2,2,2), imshow(bound), title('Boundary');
```

```matlab
% 2- Skeletonisation
% Read image spine.tif
image2 = imread('spine.tif');


% Find the skeleton of the image 'image2'. Put the result in 'skel'
% Add instruction(s) here:
skel = bwskel(imbinarize(image2));


% Plotting results
subplot(2,2,3), imshow(image2), title('Original');
subplot(2,2,4), imshow(skel), title('Skeleton');


% ------------------------------------------------------------------------------
--------
% Question 4: Filtering Algorithm with Morphological Operation
% Perform successive open-close filtering with a series
% of SEs of increasing size. Three successive open-close will be used.
% DON'T CHANGE VARIABLE NAMES


% Read image rice.png
image4 = imread('rice.png');


% Take a copy of the image for further processing
f = image4;


% Create a 'for' loop with three steps to apply the algorithm.
% The SEs must be of disk shape.
% The first SE is of 3x3 size, and it increases in the loop with each step
% Loop with three steps (i.e. loop counter is from 1 to 3 inclusive)


% Start with f, which is 'image4' then update its value in the loop with
% each step. Also, define the SE inside the loop
% Add instruction(s) here:
for k = 1:3
    se = strel('disk',k);
    f = imclose(imopen(f,se),se);
end


% Plotting Results
```

```
figure, subplot(1,2,1), imshow(image4), title('Original');
subplot(1,2,2), imshow(f), title('Filtered');



% ---------------------------------------------------------------------------
--------
% Question 5: Histogram Equalisation and Noise Filtering
% DON'T CHANGE VARIABLE NAMES


% 1- Histogram Equalistion


% Read image coins.png
image5 = imread('coins.png');


% Get the histogram for the image 'image5' and store in the variable 'h1'
% Add instruction(s) here:
h1=imhist(image5);


% Equalise image 'image5', put the result in 'im5_h'
% Add instruction(s) here:
im5_h = histeq(image5);


% Get the histogram for the equalised image and store it in 'h2'
h2 = imhist(im5_h);


% Plotting results
figure, subplot(2,2,1), imshow(image5); title('Original');
subplot(2,2,2), plot(h1); title('Hist-Original');
subplot(2,2,3), imshow(im5_h); title('Equalised');
subplot(2,2,4), plot(h2); title('Hist-Equalised');


% 2- Median Noise Filtering
% Read image cameraman.tif
image6 = imread('cameraman.tif');


% Add salt and pepper noise with 0.2 strength on the image 'image6'
% Add instruction(s) here:
```

```
im6_n1 = imnoise(image6, 'salt & pepper', 0.2);


% Add salt and pepper noise with 0.5 strength on the image 'image6'
% Add instruction(s) here:
im6_n2 = imnoise(image6, 'salt & pepper', 0.5);


% Use Median Filter with the default 3x3 neighborhood mask and
% apply it on the two noisy versions of the image
% Add instruction(s) here:
im6_filt1 = medfilt2(im6_n1);
im6_filt2 = medfilt2(im6_n2);


% Plotting Results
figure, subplot(2,2,1), imshow(im6_n1); title('Image with S-P Noise (0.2)');
subplot(2,2,2), imshow(im6_filt1); title('Median Filtered');
subplot(2,2,3), imshow(im6_n2); title('Image with S-P Noise (0.5)');
subplot(2,2,4), imshow(im6_filt2); title('Median Filtered');
```