EE4704 Image Processing and Analysis, CA2

AY22/23 Sem 1

This assignment is to be done with Matlab. Students are required to submit a zip file containing a report and Matlab code. Please see Section F for details.

A. Feature Measurement (30 marks)

1. The test image is test1.bmp (image *I*).
2. [**15 Marks**] Use image *I* as input. Implement the intermeans algorithm to calculate the threshold *T* and use it to threshold *I*. The output image is *I*1. [intermeans.m]
3. [**15 Marks**] Use image *I*1 as input. Calculate these features: perimeter, area, compactness, centroid, invariant moment ϕ_1. [features.m].
4. Note that:
   (a) You can check your implementation intermeans.m by testing it on image "letter.bmp". The calculated threshold value should be 87.
   (b) Perimeter and area – calculate using the method given in the notes or employ some other method that gives more accurate results.
   (c) Centroid – calculate from the moment values.

B. Feature Invariance (20 marks)

1. The test image is test2.bmp (image *J*).
2. [**7 Marks**] What do you think is the optimum threshold T_{Opt} for segmenting the object accurately?
3. [**5 Marks**] Obtain the intermeans threshold T_2 using intermeans.m.
4. [**2 Marks**] Threshold *J* using T_2 and measure the features using features.m.
5. [**2 Marks**] Threshold *J* with threshold T_{Opt} and measure the features using features.m.
6. [**4 Marks**] Compare the segmentation results obtained with T_2 and T_{Opt} . Discuss the sensitivity of the measured feature values to the threshold values.

C. Hough Transform (30 marks)

1. The test image is letter.bmp.
2. [**4 Marks**] Compute the edge map of the test image. **You may call any off-the-shelf matlab functions to compute the edge**.
3. [**20 Marks**] Use Hough Transform to detect straight lines in the image. You may choose either the slope-intercept or the normal representation to denote and detect straight lines. Also, you need decide the sizes of the bins in your accumulator array. Please note, you have to implement this part yourself. **You cannot call off-the-shelf matlab functions to conduct Hough Transform**.

4. [**6 Marks**] Plot the straight lines on letter.bmp, and save the new image, with the name letter_line.bmp.
5. For this part, there is no need to write your code as functions. You may simply put all your code under run_D.m.

D. Boundary Plot (20 marks)

1. Input image is the boundary image test3.bmp (image *K*).
2. [**20 Marks**] Calculate the $r - \theta$ values and plot the graph. [rtheta.m]

E. Matlab Code

1. The same Matlab code is to be used for Sections A and B.
2. As part of the assessment process, your code may be tested on images other than test1.bmp, test2.bmp and test3.bmp to check that it is reasonably robust.
3. Follow the templates below to write your code as functions.

intermeans.m

```
% To calculate the intermeans threshold;
% input is the gray level image 'test1.bmp'
% output is the threshold value T and the binary thresholded
% image Iout.

function [T,Iout] = intermeans(Iin)
%
% put your code here
%
end
```

features.m

```
% To compute the features;
% input is the binary thresholded image
% outputs are the feature values
function [P, A, C, xbar, ybar, phione] = features(Iin)
%
% put your code here
%
end
```

rtheta.m

```
% To compute the r-theta plot;
```

```
% input is a boundary image ‘test3.bmp’
% output is the array containing the r-theta value
function [r, theta] = rtheta(Iin)
%
% put your code here
%
end
```

4. In addition to the above, you will also have to provide scripts to test your functions.

run_A.m

```
%%%%% Section A %%%%%
% This m file is used to test your code for Section A
% Ensure that when you run this script file, the output images
are generated and displayed correctly
%--- 1. Display the thresholded image and the threshold
I = imread('./test1.bmp');
[T, IT] = intermeans(I);
imshow(IT) % display image IT
output = T % display the intermeans threshold
%--- 2. Display the measured feature values
[P, A, C, xbar, ybar, phione] = features(IT)
```

run_B.m

```
%%%%% Section B %%%%%
% This m file is used to test your code for Section B
% Ensure that when you run this script file, the output images
are generated and displayed correctly
%--- 1.
I = imread('./test2.bmp');
[T, IT] = intermeans(I);
imshow(IT) % display image IT
output = T % display the intermeans threshold
%--- 2
% display the measured feature values
[P, A, C, xbar, ybar, phione] = features(IT)
%--- 3
Iopt = I >= Topt; % threshold J with Topt
imshow(Iopt) % display image Iopt
% display the measured feature values
[P, A, C, xbar, ybar, phione] = features(Iopt)
```

run_C.m

```
%%%%% Section C %%%%%
% This m file is used to test your code for Section D
% Ensure that when you run this script file
%--- 1.
I = imread('./letter.bmp');
% Continue your code here.
```

run_D.m

```
%%%%% Section D %%%%%
% This m file is used to test your code for Section C
% Ensure that when you run this script file, the r-theta plot
is displayed correctly
%--- 1.
I = imread('./test3.bmp');
[r, theta] = rtheta(I); % calculate r and theta
% plot r-theta graph
```

F. Report

1. Your report should focus on the results, observations, explanations and discussion. Relevant images should be included.
2. If you use any algorithms that are not from the lecture, you should explain how they work.
3. The softcopy of the report (pdf file) and the Matlab m-files are to be zipped and submitted to the "CA2-submission" folder on the EE4704 module LumiNUS website.
4. The file is to be named as follows:
   matric number_full name.zip (e.g., A010134J_Tan_Shu_King.zip).
5. ***The results and report must entirely be your own work. Plagiarism is a serious offence.***