# EE420 – Digital Image Processing

# Final Take Home Exam

Assigned: Mar 11, 2021 Due: Mar 18, 2021

Maximum Possible Grade: 50 points

# General guidelines

Please upload your response to Canvas as a ZIP file with the following filename convention

* Name of ZIP file FIRSTNAME\_LASTNAME\_Homework5.zip
* Replace FIRSTNAME with your first name
* Replace LAST NAME with your last name

**Failure to adhere to the filename convention will result in deduction of points.**

Provide a detailed response to each question, including supporting mathematical arguments. Include screenshots of image before and after image processing. Failure to do will result in deduction of points.

# Submitting your solutions

Please ensure that the ZIP file uploaded to Canvas, includes the following components:

* PDF file of your write-up including response to all questions.
* Completed MATLAB Source code for my\_GrayThresh.m
* Completed MATLAB Source code for HWK5\_ImageThresholding.m
* Completed MATLAB Source code for HWK5\_LicensePlateRecongition.m

**Failure to adhere to the filename convention will result in deduction of points.**

# Objective

The homework is designed to help you appreciate the utility of thresholding and morphological processing. You will be implementing Otsu’s thresholding algorithm and use the function to binarize the image of a license plate and detect characters using the hit-or-miss transform discussed in Lecture-10.

# Otsu’s thresholding algorithm (15 points)

Implement the Otsu thresholding algorithm described in Lecture-10 (Slide 13). Use the function to binarize (convert from grayscale to binary by thresholding) the iceberg image (**iceberg.tif**) supplied with the assignment. Compare your threshold to MATLAB’s built-in implementation of the Otsu thresholding algorithm, called graythresh.

Please ensure that your implementation of the Otsu threshold adheres to the following conventions

***MATLAB Function name:*** my\_GrayThresh

***Inputs:*** user supplied image

***Outputs:*** result of thresholding, optimal value of threshold

Failure to adhere to these conventions will result in deduction of points.

The starter code HWK5\_GrayThresh.m supplied with the assignment includes a call to your implementation of my\_GrayThresh.

WARNING: Do not convert image to double using im2double as the threshold is supposed to a gray value between 0 and 255 for an 8-bit image.

WARNING: Read the MATLAB documentation associated with graythresh before attempting the assignment. The function returns a number that is inconsistent with the output of your function. Your task is to identify the mapping.

**Deliverables & Questions**

1. Completed MATLAB code for HWK5\_ImageThreesholding.m and my\_GrayThresh.m (10 points)
2. What is the mean gray level in the entire image? (1 point)
3. What is the gray value associated with the Otsu threshold? How does it compare to the mean value of the image? (2 points)
4. How does your Otsu threshold compare to MATLAB’s implementation of Otsu thresholding using the graythresh function? (2 points)
5. Screenshots of the original image, its histogram and the binarized image obtained using your Otsu thresholding. Label each screenshot clearly. Failure to do so will result in deduction of points. (5 points)

# License plate recognition (30 points)

In this portion of the assignment, you will study how to recognize the characters on a license plate using morphological image processing. Please use the image **LicensePlate\_Custom.png** included with the assignment to complete this portion of the homework. Also included with the assignment are image templates of alphanumeric characters **ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789** commonly found in a license plate. I have ensured that the characters on the license plate are the same size and font as the text **MUSTANG** written on the license plate.

It is strongly recommended that you read <https://en.wikipedia.org/wiki/Hit-or-miss_transform> before you attempt the assignment.

Approach:

* Identify a region of interest (set of rows, columns) in the image of the license plate where you are likely to see the characters **MUSTANG**. You will find use for this in a later step. This step is not unreasonable as the design of license plates and the location of characters is standardized by the Department of Motor Vehicles (DMV) in each state.
* The first step in recognizing characters in the license plate is to binarize the image of the license plate. Notice that the characters on the license plate are black and the background white. Binarize the image of the license plate so that the large characters in the middle of the plate appear as white and the background appears as black. An example is shown below:



Exercise caution when thresholding as you are looking to binarize the license plate so that the characters appear white and the background black. This may be realized by thresholding the license plate image so that all gray levels below the Otsu threshold are white. This is unlike what you did for the first question.

* Apply the same threshold to binarize the character templates as well. The starter code supplied with the assignment loads the image template of the characters into a cell array. Please read the MATLAB documentation to see what a cell array is and how you may find use for it in future tasks.
* For each character in the template, apply a hit-miss transform (MATLAB function: bwhitmiss) to see if the binary image of the character exists in the license plate. To use the hit-mass transform, you need to specify two structuring elements:
  + the character template, and
  + the outline of the character template which may be obtained by computing the difference of two dilated character templates (character template dilated with square character template dilated with square)

Use the MATLAB function imdilate to identify the outline of the character.

* Check if the output of the hit-or-miss transform is non-zero in the region of interest (ROI). This may be done by simply summing up the image intensities in the ROI. If the sum is non-zero, you must have detected a character. Now, display the detected character and its position in the image. An example is shown below:



You can accomplish this by dilating the output of the hit-or-miss transform with the image template of the detected character.

**Deliverables & Questions**

1. Completed MATLAB code for HWK5\_LicensePlateRecognition.m (20 points)
2. Screenshots of the binarized license plate and the 7 detected characters. Label each screenshot clearly. Failure to do so will result in deduction of points. (10 points)