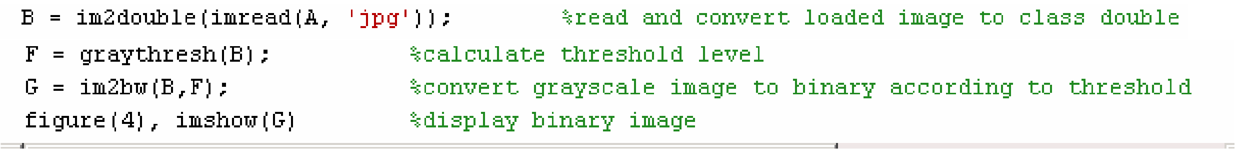
Computer Vision Exercises

10: Image Threshold and morphology

November 28, 2019

1. Load the image ‘coins.png’.

2. Calculate the binary image of the image. Use the graythresold function of Matlab and verify the selected value for thresh and displayed it in the MATLAB Command Window. MATLAB chooses a value for thresh that minimizes the intraclass variance of black and white pixels. If this value does not meet your expectations, use a different value when using the **im2bw** function. Do not forget to use the function **im2double**. This function converts the image from its current class to class double. Many MATLAB functions cannot perform operations on class unit8 or unit16, so they must first be converted into class double. This is due to the unsigned nature of class unit. Certain mathematical functions must be able to output to a floating point array in order to operate. When writing an image, MATLAB converts the data back to class unit.



```
B = im2double(imread(A, 'jpg'));        %read and convert loaded image to class double
 F = graythresh(B);          %calculate threshold level
 G = im2bw(B,F);             %convert grayscale image to binary according to threshold
 figure(4), imshow(G)        %display binary image
```

3. Add Gaussian noise to the image;

4. Build the histogram of the image.

5. Implement the Otsu’s method to segment the original and noisy image.

6. Now, use *graythresh* function (again) and compare the results with the implemented method.

   What are the differences?

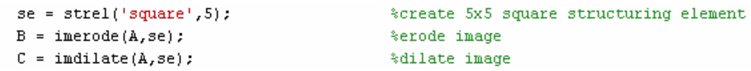
Erosion and dilation

Erosion and Dilation – Erosion and Dilation are similar operations to median filtering in that they both are neighborhood operations. The erosion operation examines the value of a pixel and its neighbors and sets the output value equal to the minimum of the input pixel values. Dilation, on the other hand, examines the same pixels and outputs the maximum of these pixels. In MATLAB erosion and dilation can be accomplished by the **imerode** and **imdilate** functions, respectively, accompanied by the **strel** function. Example demonstrates erosion and dilation.

Example
In order to erode or dilate and image you must first specify to what extent and in what way you would like to erode or dilate the image. This is accomplished by creating a

structured element by using the **strel** function. There are many types of structuring elements, each with their own unique properties. For this example, the square shape provides a 5x5 square structuring element. To find other shapes for structuring elements, look up **strel** in MATLAB’s help.

Below contains the M-commands for this example. The image used in this example is the Coins3 image. In actual applications the structuring element must be configured to process the image according to desired results.



```
se = strel('square',5);                %create 5x5 square structuring element
B = imerode(A,se);                     %erode image
C = imdilate(A,se);                    %dilate image
```

Play with the strel function to define different structured elements and obtain different results.