

# CS GY 6643 - Computer Vision, Fall 2024

## Project 1

Due: 2024/10/04 11:59 PM

---

*Note: For the project you are not allowed to use pre-built libraries to implement the core algorithm of the problem. You can use pre-built libraries to load/process the images. You have to submit a final jupyter notebook and a pdf report as a submission. Your pdf report should be informative enough on its own. We will use your jupyter notebook file to check your implementation of the code. Use markdown cells to write text explanation wherever needed. Upload the jupyter notebook to google drive and add the sharable link to brightspace.*

---

## 1 Histogram Equalisation

Imagine you're a detective analyzing a mysterious photograph below 1 where a person is sitting on something, but you can't quite tell what it is. The image is washed out, with a bright laptop screen stealing all the attention, leaving the rest of the scene in low contrast and hidden details. Your mission, should you choose to accept it, is to enhance the image and reveal the secret!

Using the concept of histogram equalization, your task is to write a function `histogram_equalisation` that takes this grayscale image as input and enhances the contrast, making the scene clearer. Once you process the image, uncover what the person is sitting on. Can you bring the hidden details to light and solve the mystery?



Figure 1

Tasks to execute:

- Brief introduction of histogram of an image and what it does. **(3)**
- Implement the following functions: `def create_pdf(im_in)` which returns the pdf, `def create_cdf(pdf)` which returns the cdf and `def histogram_equalization(im_in)` which returns the equalized image utilizing your previously implemented functions of pdf and cdf. **(7)**

- Plot pdf and cdf before and after histogram equalization. Plot them side-by-side for easy comparison. **(10)**
- Provide a comparative discussion of the PDF and CDF plots, and relate them to the image's contrast and intensity. **(5)**

## 2 Image Thresholding

Transmission electron microscopy (TEM) provides resolutions on the order of a nanometer. Hence, it is a critical imaging modality for biomedical analysis at the sub-cellular level. One of the problems associated with TEM images is variations in brightness due to electron imaging defects or non-uniform support films and specimen staining. These variations render image processing operations such as segmentation more difficult. The correction requires estimation of the global illumination field in either the background and foreground and other corrections. Below 2 is a TEM image of the C-Elegans ventral nerve cord. Can you, by image thresholding, enhance the cell boundaries which can then be used to segment the cell further in the pipeline?

*Note: You just need to perform image thresholding to enhance the cell boundaries.*

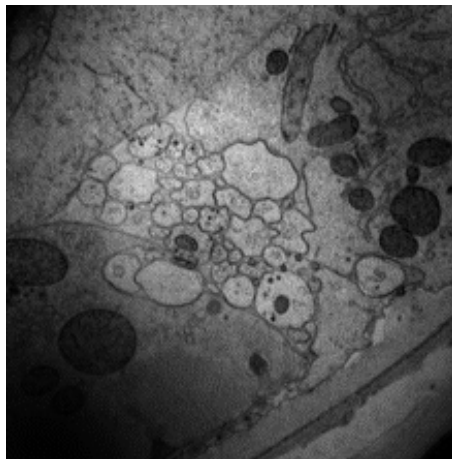


Figure 2

Tasks to execute:

- Show histogram of the image. **(3)**
- Implement and apply Otsu's thresholding and generate a plot of the inter-class variance as a function of the chosen threshold (i.e., x-axis with each possible threshold from 0-255, y-axis with the resulting variance) and highlight the threshold value where it is maximised. **(6)**
- Show the binary image after thresholding and discuss the results. Explain why do you think the results are they way they are. **(3)**
- Implement and apply Niblack thresholding algorithm and show how changing the filter shape affect the results. Show the thresholded binary image with 3 examples of changing filter size. **(10)**
- Comparatively discuss the two thresholding based on the results you got. **(3)**

## 3 Template Matching

Have you ever played the classic game "Where's Waldo?". Waldo, with his iconic red-and-white striped shirt and glasses, loves to hide in the most crowded and chaotic places, making him tricky to find. Now, it is your turn to take on the challenge – but instead of searching with just your eyes, you will use computer vision to find him! In this task, you will apply template matching to locate Wilma, Waldo's

friend, in a large, cluttered image 3. Let's see how quickly you can track her down using some clever image processing!

The template 4 provided is of Wilma whom you need to find.

*Note: The image is colored; you may need to adjust your approach to account for that during template matching.*

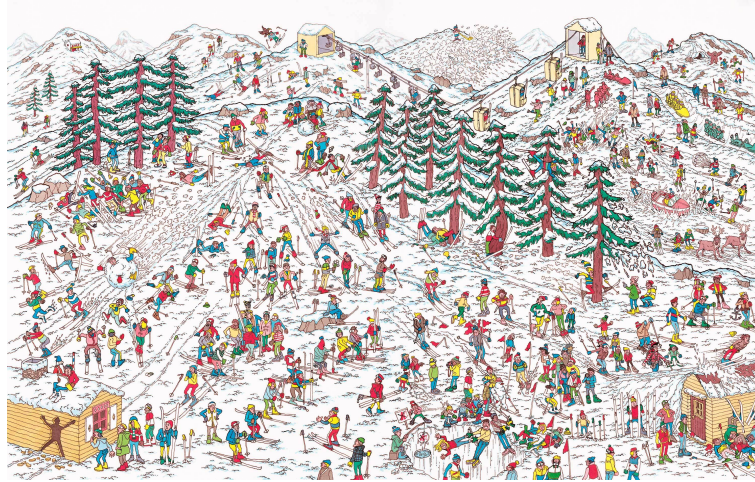


Figure 3

While performing the template matching, you might have to take care of the scale of the template. At the end of your task, once the location is identified, mark the same pixel location on the image with a box around the pixel or color the center of the pixel.

*Note: While translating the pixel position of the location where Wilma is found in the cluttered image, take into account the reduction in size that will occur when you apply the template to find a match. Adjust the same in the cluttered image before marking the final location.*



Figure 4

Tasks to execute:

- Implement cross-correlation with the original image and the template as mask. (8)
- Parse the correlation image to detect the maximum peak value and its (x,y) location. You may mark this location with an overlay of a circle or just manually painting an arrow or else. Discuss if this location matches your expectations. Discuss the appearance and cause of other peaks, and how they compare to the global peak, you may just manually check other peaks and annotate them on the peak image. (10)
- Show the image which helps you identify the peaks (cross-correlation output) with a mark on the highest peak.(2)
- Overlay the found peak on the original image. (5)

## 4 Creative Section

Points **(25)** If you beat this section, you will get **(5)** bonus points.

You are free to choose any method to find Waldo in the image below 5. It should be quicker (in execution time) than the previous implementation, which was done using cross-correlation to get full points. The top 10% (in execution time) of the students will receive bonus points. We will normalise the runtime in terms of image size to do a fair comparison between the two images.

Make your code generic enough, such that it accepts two images(original image and template image) and returns a bounding box around the matched template on the original image.

We will pass a test image with a template to your function and use the runtime to determine the score for bonus points.

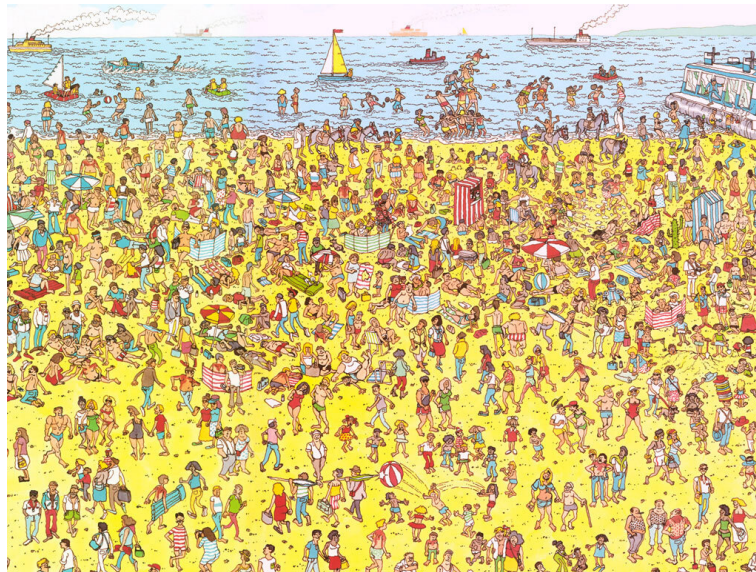


Figure 5

The template is below 6.

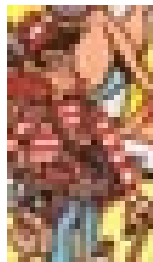


Figure 6