COMP/ENGN6528 Computer Vision – 2022

Computer Lab-1 (CLab1)

Objectives:

The goal of this lab is to help you become familiar with, and practice Python-based or Matlab-based basic image processing operations, image analysis, and image filtering.

This is CLab-1 for COMP/ENGN6528. The objective of this lab is to help you familiar with basic image I/O functions in either Matlab or Python. Note, in this lab, you are free to choose any of those two languages as you prefer, and lab task descriptions are provided with both languages.

If you have not used Python or Matlab before, this lab is an opportunity to get you to be quickly familiar with basic language usages and relevant libraries for image processing and computer vision. *Please note that Python is now increasingly used in computer vision, and we encourage you to practice it in this course.*

Special Notes:

1. Each Computer Lab task lasts for three weeks and has two lab sessions: session-A and session-B in the first two weeks. Tutors/Lab instructor will provide basic supervision to both sessions. The third week has no lab, which is for you to complete and submit the lab report.

2. Before you start to work on **writing lab report**, we strongly recommend you to watch the video '**how to write a good lab report**' in Computer Labs section on Wattle. Keep in mind, Lab markers would not be interested in a pure collection of your experiment results. (Imagine that you are demonstrating your lab works to others. You need appropriate interpretations for experimental results.)

3. The requirement of Lab submission is attached in the last pages. Please ensure your submission meets the requirement.

4. Your Lab will be marked based on the overall quality of your lab report. The report is to be uploaded to Wattle site before the due time, which is usually on the Sunday evening of the third week after the announcing of computer lab tasks. (e.g. Sunday of Week5 for Clab1). Please see the lab policy for the penalty for late submission. Note, all students should consider problems of their own personal Internet access in advance.

5. It is normal if you cannot finish all the tasks within the two 2-hour sessions — these tasks are designed so that you will have to spend about 9 hours to finish all the tasks including finishing your lab report. This suggests that, before attending the second lab session (e.g. the Lab in Week4 for Clab1), you must make sure that you have almost completed 80%.

Academic Integrity

You are expected to comply with the University Policy on Academic Integrity and Plagiarism. You are allowed to talk with / work with other students on lab and project assignments. You can share ideas but not code, you should submit your own work. Your course instructors reserve the right to determine an appropriate penalty based on the violation of academic integrity that occurs. Violations of the university policy can result in severe penalties.

C-Lab-1 Tasks

Task-1:   Matlab/Python Warm-up. (2 marks):

Describe (in words where appropriate) the result/function of each of the following commands of your preferred language in report. Please utilize the inbuilt *help()* command if you are unfamiliar with the these functions.

Note: Different from Matlab, Python users need to import external libraries by themselves. And we assume you already know some common package abbreviations (e.g. *numpy = np*). (0.2 marks each)

| Matlab	Python
(1) a = [1, 2, 3; 5, 2, 20] ;	(1) a = np.array([[1, 2, 3],[5, 2, 20]])
(2) b = a(1,:);	(2) b = a[0, :]
(3) f = randn(200, 1);	(3) f = np.random.randn(200,1)
(4) g = f(find(f > 0)) ;	(4) g = f[f > 0]
(5) x = zeros (1,20) + 0.50 ;	(5) x = np.zeros (20) + 0.5
(6) y = 0.5 .* ones(1,length(x));	(6) y = 0.5 * np.ones([1, len(x)])
(7) z = x + y;	(7) z = x + y
(8) a = [1: 100];	(8) a = np.linspace(1,100)
(9) b = a([end: -1:1]);	(9) b = a[: :-1]
(10) b(b <25)=0;	(10) b[b <25]=0

Hint:
Do the necessary typecasting (uint8 and double) when processing and displaying the image data in the following tasks. For Python, please be aware of the default datatype of different libraries (e.g. *Image*, *matplotlib*, *cv2*). An improper datatype of image will cause many troubles when you want to display the image.

Task-2:   Basic Image I/O (2 marks)
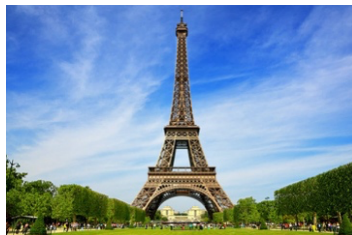
(a)   (b)   (c)

Figure 1

In this task, you are asked to:

Given the three images shown in Figure 1(you can find them in the CLAB folder on Wattle):

1. Read those images, and save them to JPG image files named 'image1.jpg', 'image2.jpg' and 'image3.jpg'. (0 marks).

2. **Using image1.jpg**, develop short computer code that does the following tasks:

a.   Read this image from its JPG file, and resize the image to 384 x 256 (in *columns* x *rows*) (0.2 marks).

b.   Convert the colour image into three grayscale channels, i.e., R-channel, G-channel, B-channel images, and display each of the three grayscale images separately (0.2 marks for each channel, 0.6 marks in total).

c.   Compute the histograms for each of the grayscale images, and display the 3 histograms (0.2 marks for each histogram, 0.6 marks in total).

d.   Convert the colour image to a grayscale image and then apply the histogram equalisation to this grayscale image, and the R-channel, G-Channel, and B-Channel images (mentioned in b.), respectively. Then display the 4 images after applying the histogram equalisation process, and their corresponding histograms. (0.15 marks for each (image, histogram pair), 0.6 marks in total). (**Hint:** you can use inbuilt functions for implementing histogram equalisation. e.g. *histeq()* in Matlab or *cv2.equalizeHist()* in Python).

Task-3:   Image Denoising via a Gaussian Filter (5 marks)

In this task, you are asked to:

1. Read in **image2.jpg**. Crop a square image region corresponding to the central part of the image, resize it to 512×512, and save this square region to a new grayscale image. Please display the two images. Make sure the pixel value range of this new image is within [0, 255] (0.5 marks).

2. Add Gaussian noise to this new 512x512 image (Review how you generate random number in Task-1). Use Gaussian noise with zero mean, and standard deviation of 15 (0.5 marks). *The intensity values of the Generated image should be within [0, 255].*

Hint: Make sure your input image range is within [0, 255]. Kindly, you may need *np.random.randn()* in Python. While Matlab provides a convenient function *imnoise()*. Please check the default setting of these inbuilt function.

3. Display the two histograms side by side, one before adding the noise and one after adding the noise (0.25 marks for each histogram, 0.5 marks in total).

4. Implement your own Matlab function that performs a 5x5 Gaussian filtering (1.5 marks). Your function interface is:

   my_Gauss_filter()

   input: *noisy_image, my* 5x5 *gaussian kernel*

   output: *output_image*

5. Apply your Gaussian filter to the above noisy image, and display the smoothed images and visually check their noise-removal effects (0.5 marks in total).

   One of the key parameters to choose for the task of image filtering is the standard deviation of your Gaussian filter. You may need to test and compare different Gaussian kernels with different standard deviations (1.0 marks).

   Note: In doing this task you **MUST NOT** use any Python's (or Matlab's) inbuilt image filtering functions (e.g. *cv2.filter2D()* in Python or *imfilter()*, *filter2()* in Matlab,). In other words, you are required to code your own 2D filtering code, based on the original mathematical definition for 2D convolution. However, you are allowed to generate a 5x5 sized Gaussian kernel with inbuilt functions.

6. Compare your result with that by Python's or Matlab's inbuilt 5x5 Gaussian

filter (e.g. conveniently *cv2.GaussianBlur()* in Python or *filter2(), imfilter()* in Matlab). Please show whether the two results are nearly identical (0.5 marks).

Further reading material: http://setosa.io/ev/image-kernels/

Task -4: Implement your own 3x3 Sobel filter in Matlab/Python (3 marks)

Sobel edge detector.

You need to implement your own 3x3 Sobel filter. Again, you must not use inbuilt functions or any inbuilt edge detection filter. The implementation of the filter should be clearly presented with your code.

Test it on the image, **image3.jpg**, and compare your result with that using the inbuilt Sobel edge detection function by visualising both results (0.5 marks). Briefly explain the function of Sobel filter and how it can achieve the edge-detection function (0.5 marks).

Investigate the accuracy of the Sobel filter in terms of predicting the orientation of the edge. If it is used for edge detection, what might cause any inaccuracy of the edge orientation estimation that you see? Look at the histogram of the gradient orientation and find out the major edge orientations in the image. Please discuss whether the gradient orientation histogram aligns with the real expected edges for the image or not.

You can use the inbuilt Sobel function to discuss this question. (2.0 marks)

Task-5:   Image Rotation (3 marks)

In this task, you are asked to do:

Take one frontal face photo of yourself, under normal lighting condition, against a white wall. (i.e., as if a passport photo). The image should be in landscape shape (i.e., the longer side is in the horizontal direction), and resized to 384x 512. (Height x Width) (0 marks).

1. Implement your own function *my_rotation*() for image rotation by any given angle between [-90°, 90°]. Display images rotated by -90°, -45°, -15°, 45°, and 90° (0.20 for each image, 1.0 mark in total).

    Note: positive for clockwise. Negative for anti-clockwise. Eg. -45° means rotate 45° in anti-clockwise direction.

2. Compare forward and backward mapping and analyze their difference (1.0 mark). [Hints: you are required to write my_rotation(), using, forward mapping and backward mapping, respectively Obtain the images after rotating using two different methods.]

3. Compare the methods using bilinear interpolation or using nearest neighbor method in the inverse mapping/warping method and analyze their differences (1.0 mark).

Submission Quality (1 mark)

Note: Completing the experiment does not mean getting full marks, please leave enough time for documenting your Lab.

- Successful submission, correctly submitting the whole lab package. (0 mark)

- Report quality, including clear interpretation, presentation and correct report format (e.g. table captions and reference). (0.5 mark)

- Code quality, including clear demonstration, comments. (0.5 mark)

==================== END of C-Lab-1 ====================

Clab-* Report

COMP/ENGN6528

name
UID Master

dd/mm/yyyy

Lab Report Requirement

1 Files

Upload a single ZIP file by the due date. You must use the following file name: CLab-1/2/3-Report-Uxxxxxx.zip, replacing Uxxxxx with your uni-ID.

Your ZIP file must contain the following contents:

(a) A PDF of your Lab Report. The report generally contains sample results from all the Lab Tasks, along with necessary comments and descriptions, questions, and answers. For more detail. Please refer to the following Template and General Instructions for Lab Report on the next page.

(b) A file named "code" to include all your *.m or *.py files.

2 Lab Report

Kindly document different question under respective headings provided with the assignment. For example:

Task-1: The Question
1. Your first question under this theme

Documentation, observations, results, analysis etc.

2. Your second question under this theme

Documentation, observations, results, analysis etc.

3. Your third question under this theme

Documentation, observations, results, analysis etc.

2.1 General Formatting Instruction

Kindly use the same font single-spaced type for the entire document as much as possible, you may use the bold and italic version of the same font to highlight the important points. In the report, you need to use Times New Roman, which is quite widely used font to document projects and research papers.

• Kindly, use appropriate font size for sections heading and its contents accordingly. For example, 15 points Times, boldface type for heading and 12 points single-spaced type for the content is one of the widely used font sizes for documenting research papers.

• Please number all your sections and subsections of the tasks as provided in the assignment.

• Please show the images mentioned in each task to make your answer clearer.

2.2 Table, Figures and Plots

This is one of the important aspects of evaluating your report. Figures and the caption of the tables must be appropriately addressed. The figure should have an appropriate title if required. All the legends in the figure should be properly highlighted. The caption of the figure should explain your observation and understanding which may comprise of quantitative or qualitative evaluation to endorse your observation. Some of the widely used font to caption your figure, table and callouts are 10-11 points Roman type, 10-11 points Helvetica non-boldface type. Kindly, adjust the size of the figure in the document appropriately such that its clearly visible and perfectly eligible to illustrate your observation. We encourage you to look into the below example for reference. Note: You **cannot insist** we should zoom in or out to see tiny details on the graphs, plots, photographs, illustration, etc. Also, make sure the figures you include in your document is ***not a copyright image***.
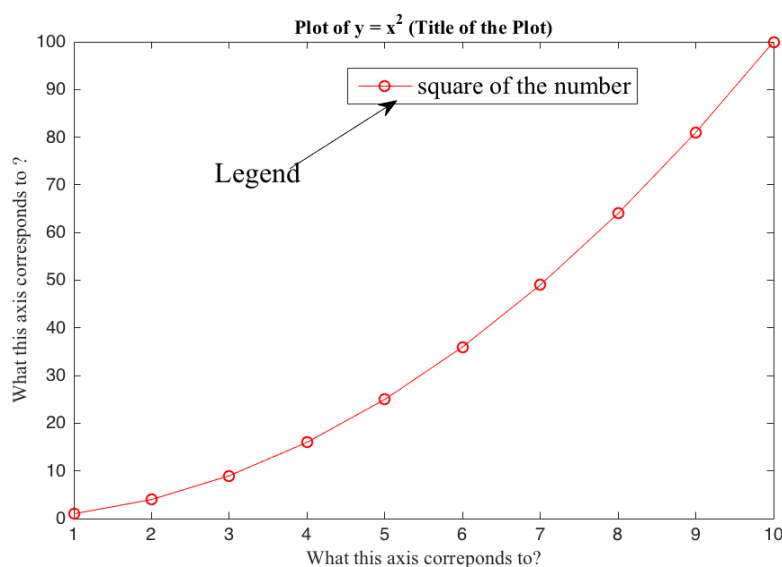
Caption: Variation in the y-axis corresponding to the values in the x-axis and What does this mean, your observations?

For tables, graphs and others as well, kindly document the purpose of the statistical illustration which should include titles and proper labelling of the data and statistics.

Please follow the requirements to write your own Lab report.

=================   END of C-Lab-1 report   =================