# ENGN4528/6528 Computer Vision – 2018
# Computer-Lab-1 (CLab-1)

**Objectives:**

This is CLab-1 for ENGN4528/6528. The objective of this lab is to help you familiar with basic image I/O functions in Matlab or Python. Note however, in all the lab task descriptions given below, we by default use Matlab as the preferred language. You are however free to choose Python if you are more comfortable with Python.

If you have not used Matlab or Python before, this lab is an opportunity to get you to quickly familiar with basic language usages of Matlab/Python for image processing and computer vision.

**Special Notes:**

(1) Each computer lab has two weeks: session-A and session-B. Tutors/Lab instructor will provide basic supervision to both sessions.

(2) 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 Friday evening of Week-2 session of your lab.

(3) 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 6 hours to finish all the tasks including finishing your Lab report. This suggests that, before attending the second lab session (in Week-2 of each CLab), you must make sure that you have almost complete 80% of the lab tasks.

## 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 dishonesty that occurs. Violations of the university policy can result in severe penalties.

# CLab-1 Tasks

## Task-1: Basic Image I/O. (2 marks)

In this task, you are asked to:

1. Take three frontal face photos of yourself, under different lighting conditions, against a white wall. The image should be in landscape shape (i.e. the longer side is in horizontal direction).

Hint: Make sure the three photos are under different, yet normal, lighting conditions. Try to avoid "extreme" lighting condition (e.g., pure dark, or pure white or saturated), otherwise you will add some unnecessary difficulties to your own "face recognition" task in a future CLab.

Note: Your photos will be used only for this class only, and for the C-Labs component of the course during this semester. All photo files uploaded to Wattle will be deleted at the end of this semester. By submitting your photos to Wattle, you agree you understand this. You are allowed to post any face photos to the public domain.

2. Re-scale the images to size of 1024 columns x 720 rows, and save them to JPG image files named "face _01_UId.jpg", "face _02_UId.jpg" and "face_03_UId.jpg". (replace UId with your uniID).

3. Choose one face image, for example, face_01 ; program a short computer code that does the following things:

3.1 Read this face image from its JPG file; resize the image to 768 x 512 in columns x rows.

3.2 Display the rescaled colour image on screen; (if you are using Matlab, you may use matlab functions imagesc() or imshow()).

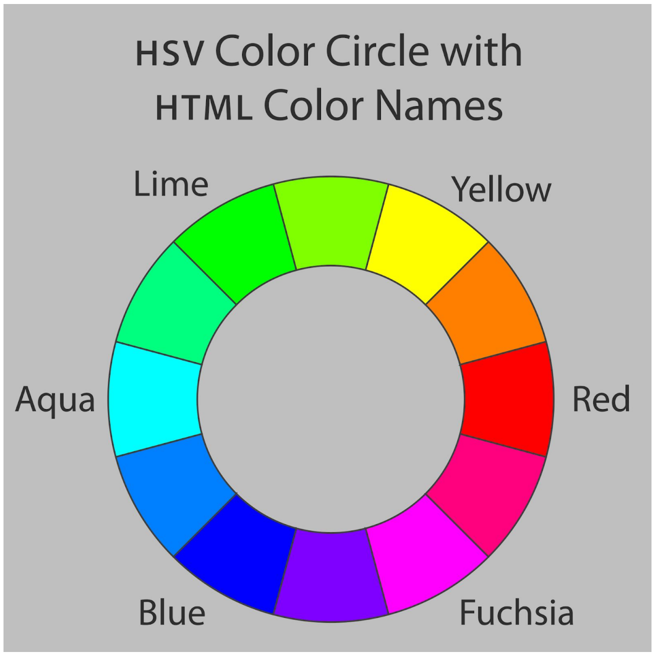
3.3 Convert the colour image into three grayscale channels, i.e., R,G,B images, and display each of the three channel grayscale images separately.

3.4 Compute the 3 histograms for each of the grayscale images, and display the 3 histograms.

3.5 Apply histogram equalisation to the resized images, and then repeat the above step 3.2, 3.3, 3.4. (hint: you may use Matlab's inbuilt histeq() function).

## Task-2: Colour Name Recognition. ( 2 marks)

Read in the following colour image (note: its original image file is on the Wattle site):



1 .Convert it to three images: H,S,V. (Check textbook or internet, to find out suitable formula for performing such RGB-to-HSV conversion).

2. Display the H image as a grayscale image, side by side to the original image.

3. Print the 12 average Hue-values next to the 12 colored regions in the image (Note: Please avoid the redundant noise).

## Task-3: Image denoise via Gaussian filter: ( 5 marks)

**Step-1: Read in one of your colour face image.** Crop a square image region corresponding to the central facial part of the image, resize it to 512 x512, and save this square region to a new gray-scale image. Make sure the pixel value range of this new image is within [0, 255].

**Step-2: Add Gaussian noise** to this new 512x512 image (if you are using Matlab, you can use function “imnoise”.).

Use the following Gaussian noise with zero mean, and standard deviation of 30. ( hint: make sure your input image range is within [0, 255]. Kindly, note that Matlab function imnoise() normalizes the input in the range [0,1] by default. So, normalize your variance accordingly like variance = (standard deviation^2)/(255^2) if you are using Matlab imnoise() in default setting).

**Step-3: Plot the histogram:** Plot the two histogram side by side, one before adding the noise and one after adding the noise.

**Step-4: Implement your own Matlab function that performs a 9x9 Gaussian filtering.**

Your function interface is: my_Gauss_filter( noisy_image, my_9x9_gausskernel, output_image)

NOTE: In doing this task you MUST NOT use any Matlab's in-built image filtering functions (e.g. imfilter(), filter2(), conv2(), or filter(), conv () ). In other words, you are required to code your own 2D filtering matlab code, based on the original mathematical definition for 2D convolution. However, you are allowed to use Matlab's function fspecial() to generate a 9x9 sized Gaussian kernel.

Apply your Gaussian filter to the above two noisy images, and display the resulted images and visually check their noise-removal effects. (Note: One of 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.)

**Step-5: Compare your result with that by Matlab's inbuilt 9x9 Gaussian filter,** e.g. filter2(), or imfilter(). Please show that the two results are nearly identical.

## Task 4: Implement your own 3x3 Median and Sobel Filter (3 marks)

**Step-1: Median Filter De-Noise.**

Add 10% salt and pepper noise to your original colour face image. I.e., to randomly pick 10% of the image pixels, and change their values to either pure white or pure black at 50-50 chance.

Implement your own 3x3 Median Filter, and use it to denoise your noisy version face image. Display the results.

Compare your result with Matlab's inbuilt 3x3 median filter.

Question and Answer: which filter (Gaussian or Median) is more suitable for removing salt-and-pepper noise ? Why ?

**Step-2: Implement your own 3x3 Sobel edge detector**.

Again, you must not use Matlab's inbuilt edge detection filter. You need to implement your own 3x3 Sobel filter).

Test it on one of your face images, and compare your result with Matlab's inbuilt Sobel edge detection function.

## Task-5: Image Morphology ( 3 marks)

1. Load in the image file "text.png"; resize it to size of 1024x1024.

Convert it to grayscale image I.

The image is also displayed here:

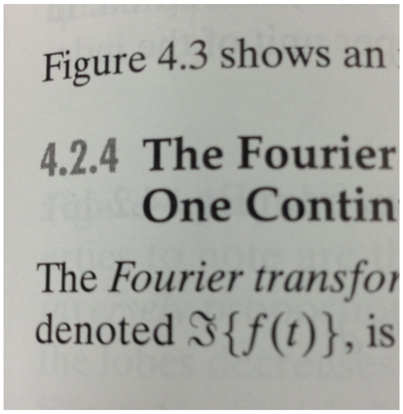


Figure 4.3 shows an

4.2.4 The Fourier
One Contin

The *Fourier transfor*
denoted $\Im\{f(t)\}$, is

2. Count the number of white pixels. Count the number of black pixels. Sum up the two numbers above.

3. Testing the effects of applying Matlab (or Python’s) inbuilt morphological operators of "erosion", "dilation", "opening" and "closing" to the binary image, and visually inspect the effects.

## Task-6: Geometric transformation (3 marks)

1. Implement your own Matlab function for Image-Rotation by any given angle between [-90, and 90] ; (use counter-clockwise rotation as the positive rotation).

2. Compare your result with Matlab's inbuilt image rotation function. Can you see any differences ? Explain the differences.

( hint: Compare forward and backward mapping methods; compare different interpolation methods. If you do not know what this hint is about, you will need to look it up in textbook or online. )

## Lab Report Requirement:

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

Your ZIP file must contain the following contents:

(a) Three JPG photos of yourself.

(b) 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 in the next page.

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

# Template and General Instructions for Lab Report

# CLAB-1 Lab Report

ENGN 4528/6528, Computer Vision 2018

College of Engineering and Computer Science

| | |
|---|---|
| Student Name: | Xxxxxx |
| UID: | Xxxxxx |

## 1. Instruction

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

## Task-1: Basic Image I/O

### 1. Your first question under this theme

Documentation, observations, results etc.

### 2. Your second question under this theme

Documentation, observations, results etc.

### 3. Your third question under this theme

Documentation, observations, results etc.

3.1 **Subquestion of the third question**

Your answers

3.2 **Subquestion of the third question**

Your answers

3.3 **Subquestion of the third question**

Your answers

3.4 **Subquestion of the third question**

Your answers

3.5 **Subquestion of the third question**

Your answers

## 2. 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. Few recommended fonts are Times New Roman, Times, which are quite widely used font to document projects and research papers. Too big or too small font sizes are not encouraged. Kindly, use appropriate font size for sections heading and its contents accordingly. For example, 14-15 point Times, boldface type for heading and 11-12 point 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.

### 2.1 Table, Figures and Plots

This is one of the important aspects of evaluating your assignment tasks. The 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 point Roman type, 10-11 point Helvetica non-boldface type. Kindly, adjust the size of the figure in the document appropriately such that it's 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 can zoom in or out to see tiny details on a graphs, plots, photographs, illustration, etc. Also, make sure the figures you include in your document is not a copyright image.*

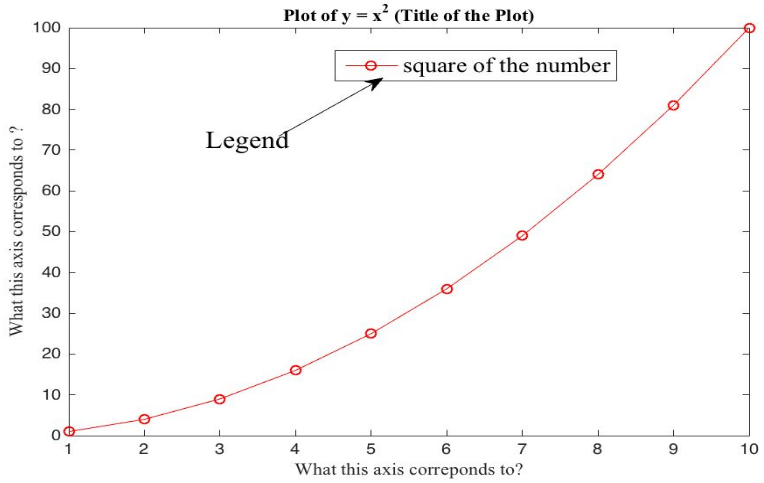


Fig. Variation in the y-axis corresponding to the values in the x-axis and What does this mean, your observations?

For table, 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.