

CS 270: DIGITAL IMAGE PROCESSING (SPRING 2024)

# Assignment 1

**Due: 23:59, April 7, 2024**

## Notes

- This assignment has **100 points** in total.
- Please prepare all your solutions in English.
- Please prepare your report with digital typesetting software (LaTeX, Microsoft Word, etc.). Hand-written reports, including digital handwriting on iPad etc., will not be accepted.
- Please submit your assignment to **Blackboard** as a **zip** file with its name formatted as `DIP2024_HW1_ID_ChineseName.zip`. The zip file should contain 3 things:
  1. Your report named as `HW1_Report_ID_Name.pdf`;
  2. A folder named as `code` that stores your codes;
  3. A folder named as `images` that stores the original images in your report.

For each problem, you should provide a separate code file that corresponds to it, with its name like `p3.m` (for Problem 3) or `p6a.m` (for Problem 6 (a)). Please make sure all paths in your codes are relative paths, so that we can run your codes and get your results without any modification.

## Policy on Plagiarism

This is an individual homework. You can discuss the ideas and algorithms, but:

- You cannot read, modify, and submit the codes of other students, nor allow other students to read, modify, and submit your codes.
- You cannot directly use generative AI tools to produce codes for submission. While you may consult generative AI for understanding the ideas and algorithms, the code you submit must be the result of your own individual understanding and efforts.

We will utilize automated tools to check for plagiarism, and any violations will result in a zero score for this assignment and 20% discount on total course grade.

## I. CODING PART

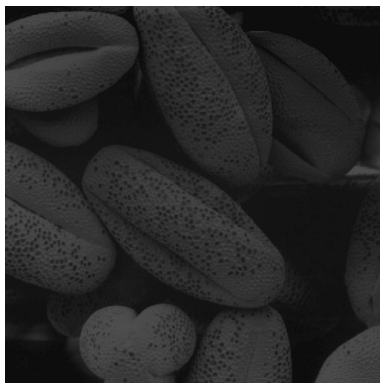
Please complete all the coding assignments using MATLAB. Make sure your results in the report are the same as the results of your codes. For general operations, the following functions may be useful:

`load`, `imread`, `double`, `im2double`, `uint8`, `imshow`, `zeros`, `size`, `montage`, `subplot`, `bar`.

You must implement the core code in each question **WITHOUT** using relevant build-in functions, e.g, `fspecial`, `imfilter`, `filter2`, `conv2`, `imsharpen`, `ordfilt2`, `hist`, `imadjust`, `histeq`, `adapthisteq`, etc. You can type `help FunctionName` in Command Window of MATLAB for detailed help text for the functionality specified by `FunctionName`.

### Problem 1: Histogram Equalization

- (a) Compute the histogram of `grain.tif` and show the histogram image in your report. (Built-in functions `hist` and `histogram` are not allowed). (5 points)
- (b) Implement histogram equalization on `grain.tif`. Show the histogram equalized image and the histogram of equalized image in your report. (Built-in function `histeq` is not allowed). (10 points)
- (c) Implement contrast limited adaptive histogram equalization (CLAHE) on `tire.tif`, where you traverse every pixel with a  $41 \times 41$  patch and process histogram equalization within each patch and update the center patch of  $3 \times 3$ . Each time you move the patch with a step of 1 pixel. The clip limit for CLAHE is 0.02, which means that after the normalization of the histogram, amplification above 0.02 should be clipped and evenly distributed to other parts of the histogram. Show the CLAHE processed images and corresponding histogram in your report. (Built-in function `adapthisteq` is not allowed) (20 points)



## Problem 2: Image Sharpening

- (a) Apply the  $3 \times 3$  Laplacian kernels ( $x$ -direction and  $y$ -direction) on `moon.jpg` to obtain the details of the image. Since the Laplacian kernels are separable, please separate them into combinations of 1-D kernels and then apply them to the image sequentially. Show the separated Laplacian kernels and the corresponding processed images. (Implement your own **convolution** operator.) (15 points)
- (b) Apply the  $3 \times 3$  Laplacian kernels (not separated) to sharpen the image by 2-D convolution. Show the results in your report. (10 points)
- (c) Use `unsharpen mask` to sharpen the image. Show the results in your report. (10 points)

The pixel intensity of the result images should be normalized to `uint8` values within  $[0, 255]$ .



### Problem 3: Nonlinear Filter

Apply the median filter and Gaussian filter to the image `lena_noisy.tif`. In your report, Please show the images `processed by the median filter and Gaussian filter` respectively, and analyze the `cause of the results`. (Filtering operations cannot be implemented by calling functions, Hint: Choose the best kernel size you think,  $\sigma = 1$ ) (30 points)

