

Lab 2: The objective of this lab is to implement spatial operation techniques for the purpose of:

- Edge detection
- Image morphology
- Histogram equalization
- Region growing
- Fourier Transform

### Learning objectives:

- Utilize existing computer vision libraries (i.e. OpenCV) to:
  - Implement edge detection and sharpening techniques
  - Implement image morphology techniques
  - Implement image histogram equalization techniques

### Procedure:

- Download the zip file called *Lab2* and extract it. This zip file contains:
  - An empty python code for you to fill in
  - A test image for you to practice on
  - This manual for the lab
- You may open the images to view them and see what they look like
- The python code consists of several comments that indicate where you should fill in.  
Note that there are three functions that you should complete:
  - imgFilter (the function used for image sharpening and edge detection)
    - Use at least three kernels for image sharpening
    - For edge detection, make use of the following kernels:
      - Sobel operator
      - Prewitt operator
      - Laplacian operator
  - imgMorph (the function used for image morphing)
  - imgHist (the function used for histogram equalization)
- Note that you must implement each of the above functions by using the built-in methods specified in the code's comments. Look at their documents to understand how to use them, and to learn which Python libraries you should import.
- Also follow any other instructions present in the code's comments.

When you have finished implementing the code, please answer the following questions:

1. For imgFilter(...)
  - Remember that a key characteristic of a sharpening kernel that it places more weight towards the center pixel. One examples of sharpening kernel is  $[0 -1 0; -1 5 -1; 0 -1 0]$ . Make use of similar kernels and observe what they do.
    - What are some sharpening kernels that you used?

- Pay attention to the total sum of the kernel. How does their total sum affect the image? (e.g. how does the image look like if the sum=1? what if sum < 1? what if sum > 1?)
  - What are some observations that you can make about each type of edge detection kernel?
- 2. For imgMorph(...):
  - Vary the kernels that are used for morphological operations. What do you observe? Make use of at least three kernels
- 3. For imgHist(...):
  - Perform histogram equalization using the following two built-in methods. Do not worry about the output being the exact same between them.
    - equalizeHist(...) in opencv
    - match\_histograms(...) in sklearn
  - Is there any difference between the two outputs? Please describe.

**Deliverables:**

- Your code that fills in the blanks as indicated by the comments in *Lab4120\_Week4\_empty.py*
- A report that details the results you have found. It should contain the following information:
  - Introduction to the image preprocessing step of the computer vision pipeline
  - A description that highlights your understanding of spatial operations used in image preprocessing techniques
  - Answers to the above questions. Include the results from your code (as specified by the questions)
  - Concluding remark of what you have learned from this lab

**Your implemented code should have the following (you will be marked for both):**

- Correctness: The code should compile without error
- Style: There should be comments and proper formatting