COMPUTER VISION – LAB (AI303L)

Class:	BS-AI-VI	**Semester:**	Spring 2024
Instructor:	Dr. Fiza Murtaza	**Email:**	fiza.murtaza@mail.au.edu.pk

Assignment # 02

Lab # 02

Image Pyrmaids

Guidelines for Attempting the Assignment:

- **The lab tasks should be done on individual basis.**
- **Lab tasks solution should consist of workable codes and a detailed report containing a discussion on different tasks and their solutions, along with input and output images.**
- **The report should be prepared as an electronic document, following a consistent format. Make use of headings and subheadings where necessary.**
- **Submit your lab tasks (code + report) within the due time, as late submissions will not be encouraged and will result in the deduction of your marks.**
- **There will be a viva voce regarding the lab tasks, and the obtained marks will also be based on the viva performance.**

LAB TASKS

[Task # 01]: Image Blending

Image blending is a way of splicing two images together so that the transition between them is smooth and natural-looking. For this task, you will be constructing a blended image from two different images, as shown in Figure 2. However, you are required to use your own unique images and design a custom mask that best supports your image blending experiments. This needs images to be of the same size, aligned to each other, and that you create a customized mask between them. You are required to perform image blending experiments using Python for at least 03 different sets of images and include their output in the report. This problem uses color images. You can think of color images as being three separate images, one for each color channel.
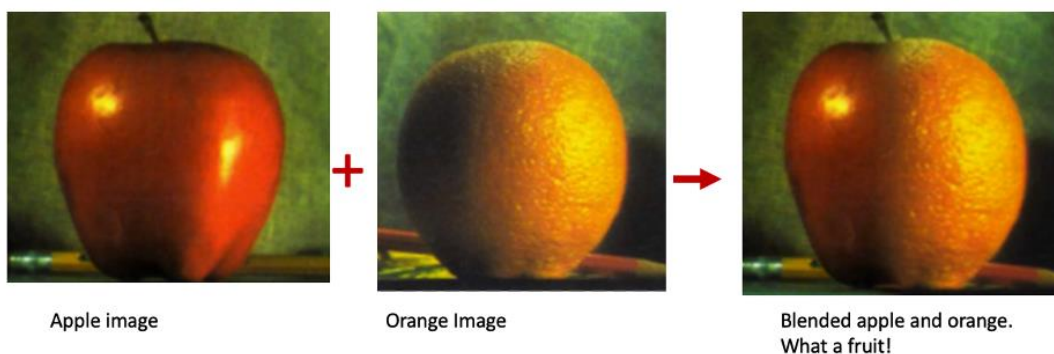

Apple image   Orange Image   Blended apple and orange. What a fruit!

Figure 1: Image Blending Example

[Task # 02]: Template Matching [Bonus Home Task]

Template matching refers to image processing where we find similar templates in a source image by giving a base template to compare. The template matching process is done by comparing each pixel value of the source image one at a time to the template image. The output would be an array of similarity values when compared to the template image. To look at the similar templates found on the image, we can find the peaks in the resulting array of the template matching. In this task, you are required to take an input image of your own choice, consisting of multiple objects. Then, create a template image by cropping an object from a small portion of the input image. Implement any template matching algorithm using Python/Matlab to find the template image in the original input image. Mark a bounding box where the template is found in the input image, as shown in Figure 3. Also, apply the template matching algorithm using image pyramids across 4-levels. Compare the results and computational cost of searching at different levels.

Figure 2: Template Matching Example