Lab 1– Spring 2023

Course Code	*Course*		
CSE 483	**Computer Vision**	**Due Date:**	**16 / 5 / 2023**
Instructor/s	**Dr. Mahmoud Khalil, Eng. Mahmoud Soheil**		

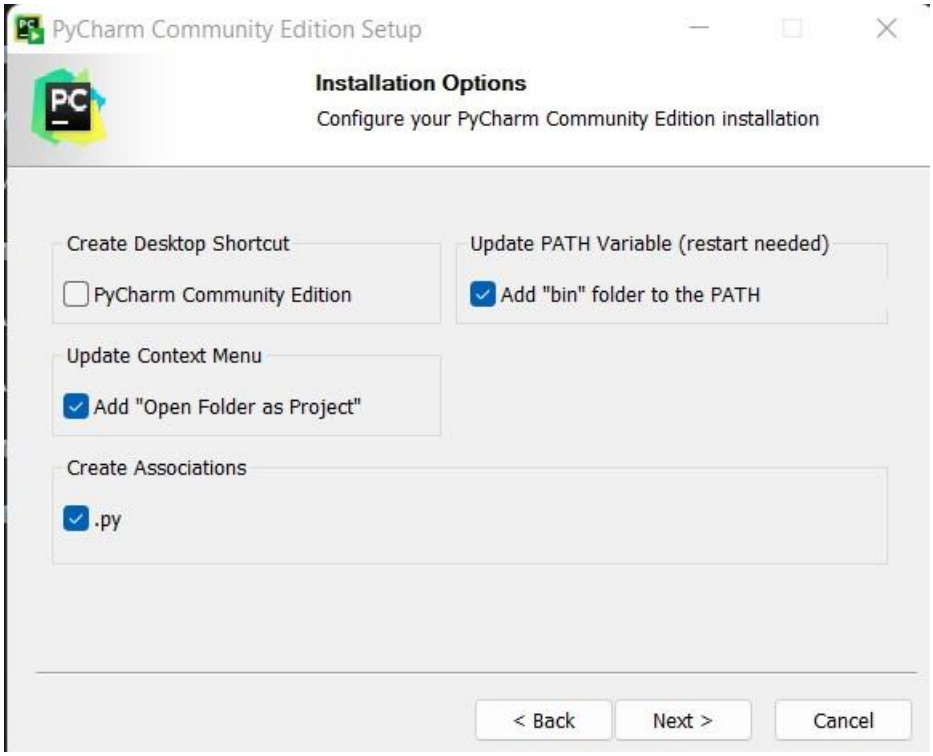
Lab 1

In this lab, we will try to use Opencv library for some image processing operations.

1. Fisrtly, we will need to install python environment on the machine from the following site:
   - https://www.python.org/downloads/

Click install now after checking the boxes

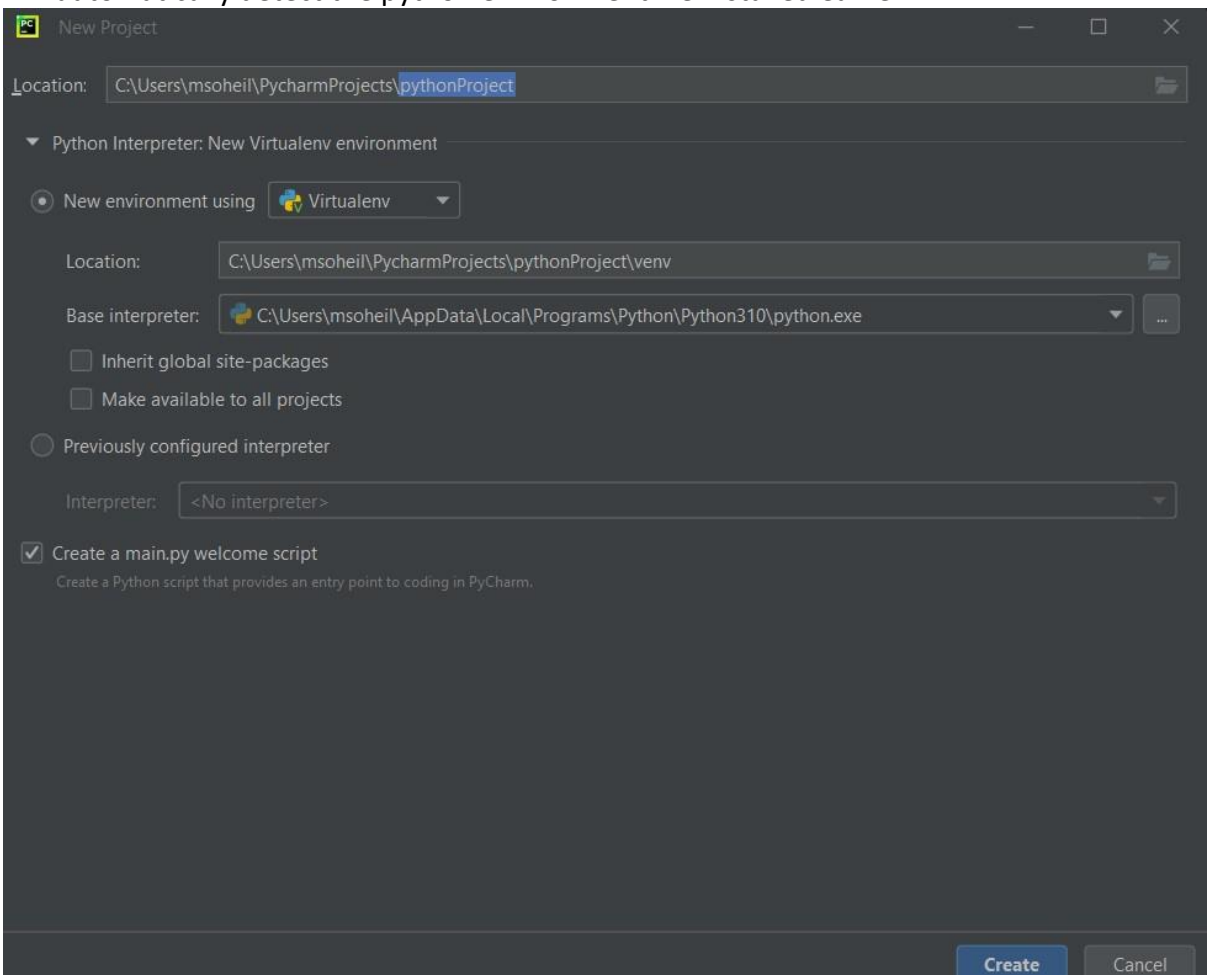
2. Secondly, we will need to download pycharm community from the following link:
   - https://www.jetbrains.com/pycharm/download/#section=windows



Keep pressing next and check the boxes in the previous figure then install then reboot now.

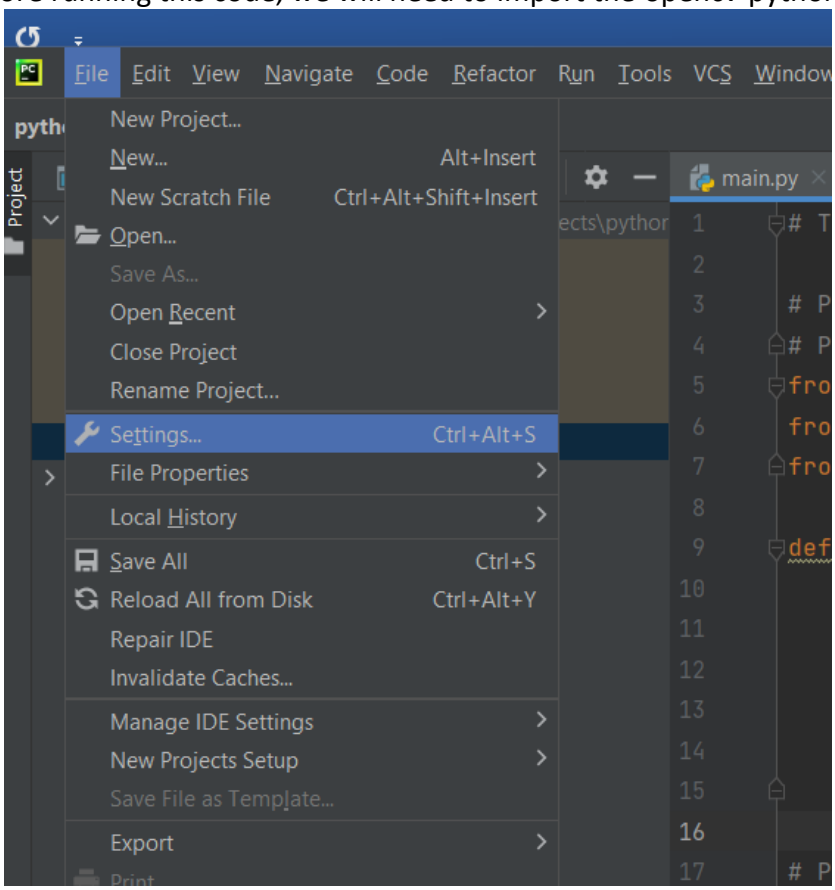
Note:
You can use any python IDE like anaconda or codelab
https://www.anaconda.com/

3. Open now pycharm and click don't import settings. Then create new project, It will automatically detect the python environment we installed earlier

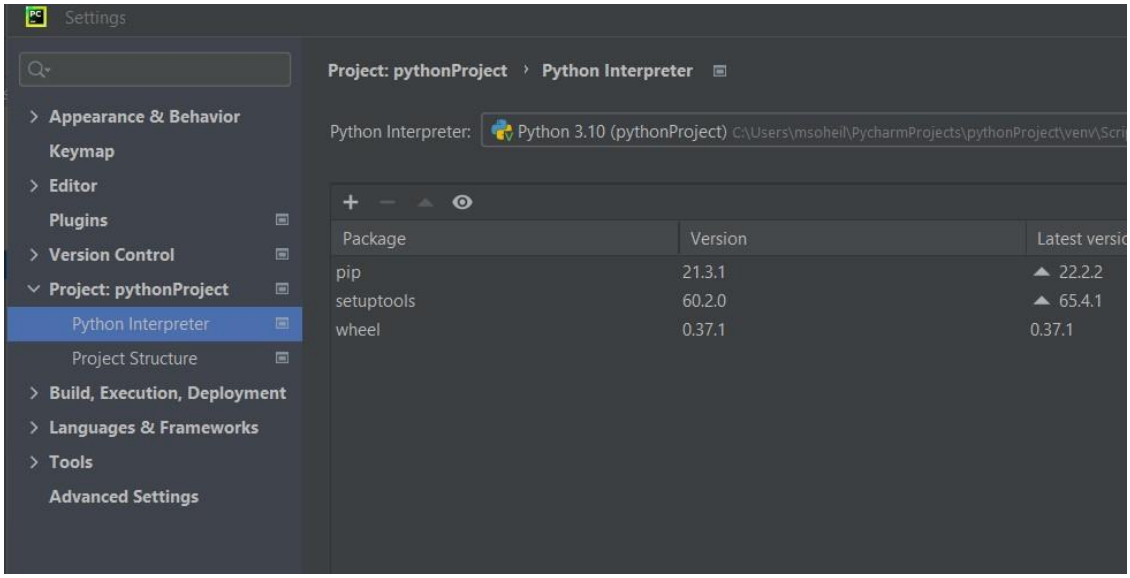


We will just check that the environment is correct and click create

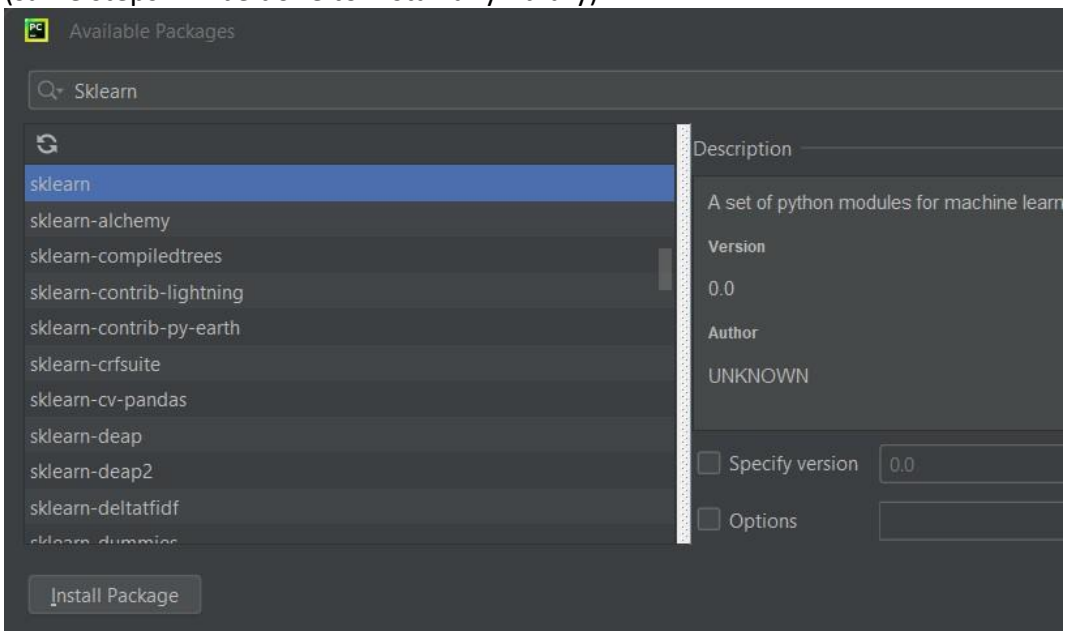
4. Before running this code, we will need to import the opencv-python library



From settings, click on the project name, then project interpreter, then the plus sign



Then search for opencv-python and click install package
(same steps will be done to install any library)



Sample:

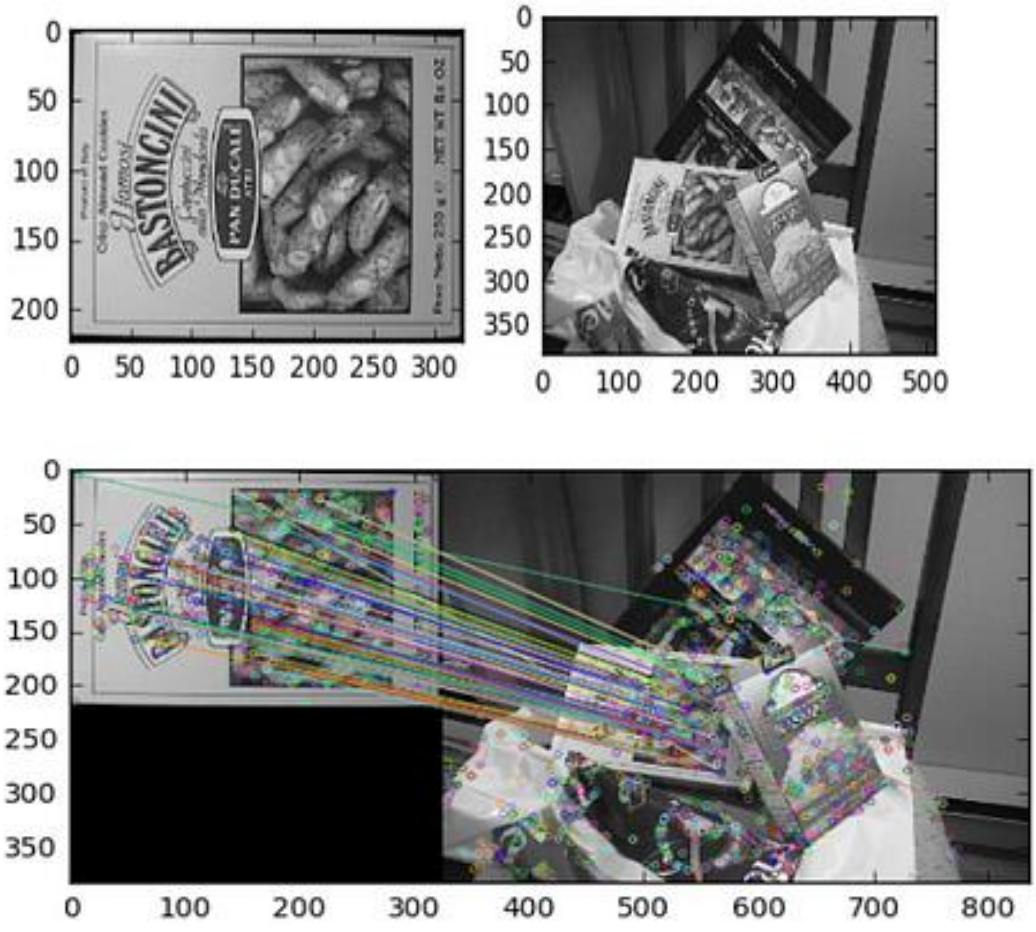
```
import cv2
import numpy as np
from matplotlib import pyplot as plt


# Press the green button in the gutter to run the script.
if __name__ == '__main__':
    img = cv2.imread('monkey.jpg', 1)
    cv2.imshow('monkey',img)
    k = cv2.waitKey(0)
    if k == 27 or k == ord('q'):
        cv2.destroyAllWindows()
    h, w, c = img.shape
    print("Dimensions of the image is:\nHeight:", h, "pixels\nWidth:", w,
"pixels\nNumber of Channels:", c)

    print(img)

    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    cv2.imwrite('Mandrill_grey.jpg', gray)

    kernel = np.ones((5, 5), np.float32) / 25
    dst = cv2.filter2D(img, -1, kernel)
    plt.subplot(121), plt.imshow(img), plt.title('Original')
    plt.xticks([]), plt.yticks([])
    plt.subplot(122), plt.imshow(dst), plt.title('Averaging')
    plt.xticks([]), plt.yticks([])
    plt.show()

    kernel2 = np.array([[-1, -1, -1],
                        [-1, 8, -1],
                        [-1, -1, -1]])

    # Applying the filter2D() function
    dst2 = cv2.filter2D(src=img, ddepth=-1, kernel=kernel2)
    plt.subplot(121), plt.imshow(img), plt.title('Original')
    plt.xticks([]), plt.yticks([])
    plt.subplot(122), plt.imshow(dst2), plt.title('Edge detection')
    plt.xticks([]), plt.yticks([])
    plt.show()
```

Required Task:

Develop a code like the previous code [use other Kernels or some Open CV functions] to perform the following on any image from your choice:

1. Feature Extraction using SIFT
2. Feature Matching
3. Transformations (scaling, rotation, translation)

Example for Feature Matching:



Deliverables:

1. Report containing steps used in the code and screenshots of the run.
2. Python code developed.