

CS 4391 Introduction to Computer Vision

Homework 1

Professor Yu Xiang

January 26, 2024

Download the [homework1_programming.zip](#) file from eLearning, Assignments, Homework 1. Finish the following programming problems and submit your scripts to eLearning. You can zip all the data and files for submission. TA will run your scripts to verify them.

Install the Python packages needed by

- `pip install -r requirement.txt`

Here are some useful resources:

- Python basics <https://pythonbasics.org/>
- Numpy <https://numpy.org/doc/stable/user/basics.html>
- OpenCV https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html

Problem 1

(5 points) Image gradient using central difference.

Complete the `main()` function in [image_gradient.py](#).

This script first reads the “cracker_box.jpg” image, and then performs central difference to compute the gradient of the image. The equations of the central difference are

$$\frac{\partial I(x, y)}{\partial x} = \frac{I(x + 1, y) - I(x - 1, y)}{2}, \quad (1.1)$$

$$\frac{\partial I(x, y)}{\partial y} = \frac{I(x, y + 1) - I(x, y - 1)}{2}, \quad (1.2)$$

where $I(x, y)$ denotes the image. **Note: in this implementation, you cannot directly use OpenCV functions to compute gradient.**

After your implementation, run the [image_gradient.py](#) in Python to verify it. Figure 1 shows an example of my implementation.

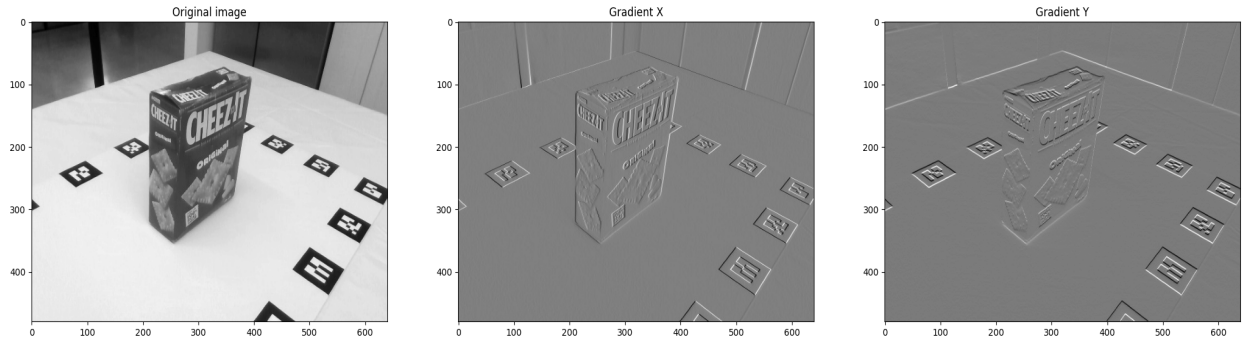


Figure 1: Image gradient.

Problem 2

(5 points) Image filtering.

Implement the `main()` function in `image_filtering.py`.

This script first reads the “cracker_box.jpg” image, and then performs image filtering using the following 3×3 kernel:

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (2.1)$$

Note: in this implementation, you cannot directly use OpenCV functions to do the filtering.

After your implementation, run the `image_filtering.py` in Python to verify it. Figure 2 shows an example of my implementation.

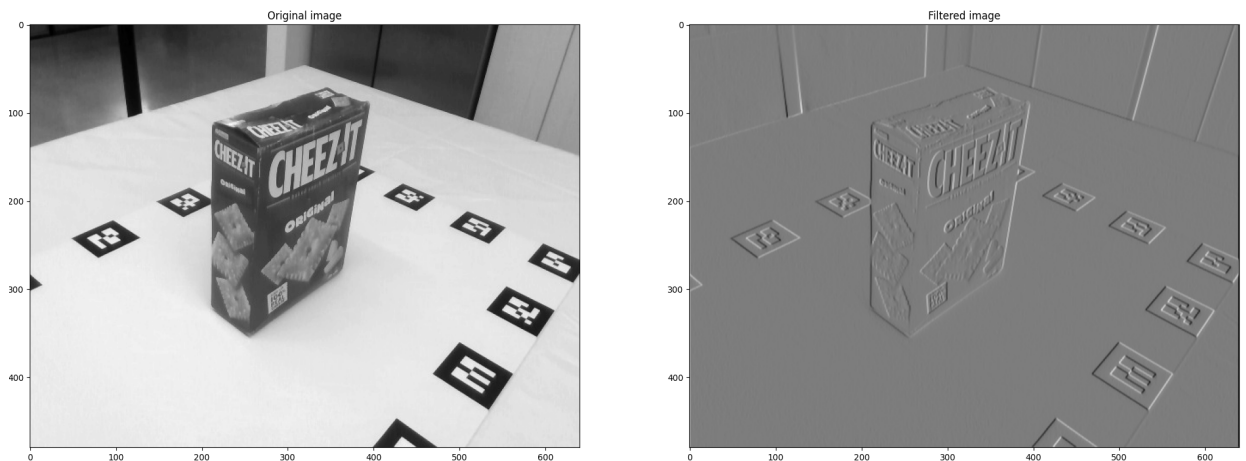


Figure 2: Image filtering.