

Homework 1

Image Manipulation

Submission deadline: 11:59 pm, 14/2

Instruction:

1. Coding: While you are free to use Google Colab or Jupyter Notebook to write code, I recommend you to start with an IDE like PyCharm as it will be easier for you to debug. Using local systems like PyCharm or Jupyter Notebook will also get you familiar with the environment setup.
2. Submission: After completing your programs, please include all of them in a Jupyter Notebook as it will be easier for me to check your programs. Write your answers to the questions in your Notebook when necessary. Put your Notebook and any related files such as input images in a ZIP file and submit it to Canvas.
3. Submission of Homework 1 is **Optional** (even though you are encouraged to complete and submit it)

Questions:

Input: Take a photo of a red object and use it as the input (image A).

Question 1: Color image

- a. Swap the red and blue pixels of image A.
Output: new image
- b. Save the blue channel of image A as a grayscale image (named Ablue).
Output: new image
- c. Save the red channel of image A as a grayscale image (named Ared).
Output: new image
- d. Which channel would you use if you want to detect the object?
Output: response

Question 2: Working with pixels

Take the inner square of 50x100 pixels (that's 50 rows by 100 columns – a horizontal image) of Ablue and insert them into the corresponding position of Ared.

Output: new image

Question 3: Arithmetic and Geometric operation

Convert image A to a grayscale image (called Agray) and complete the following tasks:

- a. What are the min and max of the pixel values of Agray? What is the mean? What is the standard deviation?
Output: response
- b. Subtract the mean from all the pixels, then divide by the standard deviation, then multiply by 10 (if your image is zero to 255) or by 0.05 (if your image ranges from 0.0 to 1.0). Now add the mean back in.
Output: new image
- c. Shift Agray to the left by 5 pixels. *Hint: you can use this OpenCV function: warpAffine*
Output: new image
- d. Subtract the shifted version of Agray from the original and make sure that the values are legal (*be aware of negative values*).
Output: new image