CSC 481 – Intro. To Image Processing
2024-25 – Winter Term
Instructor: Kenny Davila Castellanos

Weekly Assignment 6: Mathematical Morphology

Objective

The main objective of this assignment is to help student get familiar with multiple morphological operations that were studied in class.

Data

To complete this assignment, please pick 3 images of your liking. You should use:

1) A picture of one light object placed on a dark background. This can be a picture taken by you or from the internet. Please clearly indicate the source on your report.
2) A picture of one dark object placed on a light background. This can be a picture taken by you or from the internet. Please clearly indicate the source on your report.
3) A random picture of your liking different from the first two. Ideally, look for an image with an object of interest that has a contrasting background.

   You will have to run your code with each of these images, and then you must display and briefly discuss the results in your report.

Part 1 – Erosion and Dilation

For each image, do the following main steps:

a) Read the image, convert to **grayscale** and display it.
b) Convert the image to **binary** using a **mean-intensity** threshold.
c) Chose a fixed structuring element (SE). You can use anything that you would like to try.
d) Use the SE from **step c** to apply **twice** an **Erosion operation** on the **binary image** from **step b**.
e) Use the SE from **step c** to apply **twice** a **Dilation operation** on the **binary image** from **step b**.
f) Compute and display the **difference** between the images from **step d** and **step e**.
g) In the **report**, analyze the differences between these images.

For **every input image** (you have 3 inputs, see **Data**), you must show: the input image, the grayscale image, binary image, binary image after two erosions, binary image after two dilations, and difference image.

Part 2 – Opening and Closing

For each image, do the following main steps:

a) Read the image, convert to **grayscale** and display it.
b) Convert the image to **binary** using a **mean-intensity** threshold.
c) Chose a fixed structuring element (SE). You can use anything that you would like to try.
d) Use the SE from **step c** to apply **twice** an **Opening** **operation** on the **binary image** from **step b**.
e) Use the SE from **step c** to apply **twice** a **Closing** **operation** on the **binary image** from **step b**.
f) Compute and display the **difference** between the images from **step d** and **step e**.
g) In the **report**, analyze the differences between these images.

For **every input image** (you have 3 inputs, see **Data**), you must show: the input image, the grayscale image, binary image, binary image after two erosions, binary image after two dilations, and difference image.

Part 3 – Boundary Extraction

Boundary extraction of an image I, β(I), can be performed by eroding I with a structuring element B and then subtracting the eroded image from I: β(I) = I – (I ɵ B).

For each image, do the following main steps:

a) Read the image, convert to **grayscale** and display it.
b) Convert the image to **binary** using a **mean-intensity** threshold.
c) Create structuring element B using a disk (ellipse) of 5x5 pixels.
d) Use the structuring element to do boundary extraction as described above, and display your results.
e) Extract edges from the grayscale image using Canny and show your results. You might need to fine-tune the parameters of Canny for best results.
f) In the **report**, analyze the differences between these images.

For **every input image** (you have 3 inputs, see **Data**), you must show: the input image, the grayscale image, binary image, the eroded image, the boundary image, and the results for canny edge detection.

The Program

In this assignment, you will write a program that handles multiple operations. You can use any programming language of your preference, but Python is highly recommended. Your code should be properly organized and well-commented to clearly identify the correspondence between portions of the code and each part of this assignment. The code should compile and run on any properly configured programming environment without the need of making significant changes to it. In this sense, it is highly recommended to avoid the usage of absolute paths.

The report

You are asked to submit a written report showing the results for each part of the assignment. Use proper section headings and descriptions to clearly identify the results of each part. If I cannot easily find a match in the report for a required result, I might assume that this portion was not completed, and a score of 0 will be given by default.

For students working with Python, you are allowed to use Jupyter Notebooks. These allow you to combine code with the report in a single deliverable, which is perfectly acceptable for this course. Besides the original Jupyter notebook file, please also submit an export to PDF.

Delivery Instructions

You are given the freedom to use any programming language and IDE of your preference. You are also required to provide your own images, and to produce a written report. When done, **you should submit everything using a single Zip file**.

File names. The zip file that you submit should use "[Last Name(s)], [Given Name(s)].zip" as it appears in D2L. For example, "Kenny Davila Castellanos" (Davila Castellanos is two last names), would have to submit the homework with the name "Davila Castellanos, Kenny.zip". Another student named "Kenny Mauricio Davila" (Mauricio is a middle name), would have to submit the homework as "Davila, Kenny Mauricio.zip". Not following these instructions might lead to a penalty.

Policies

1. All general policies about Plagiarism and Cheating apply to this homework. If you plagiarize or receive code from other people, you will be caught and you will receive a score of 0, and a report of the academic integrity violation will be filed.
2. Please limit the usage of Chat-GPT and other code generators to ethical usage only. Submitting code that was directly generated by these tools is considered a form of plagiarism.
3. Do not post your solutions online and do not share them with anyone. It is your responsibility to safeguard your private data.
4. **Code that does not compile due to syntax and/or semantic errors will automatically receive a score of 0.** It is hard to assign partial credit when I cannot even run your code.
5. You can use any programming language of your choosing.
6. You must follow the delivery instructions.
7. Very late homework's without justification will receive a score of 0.
8. The homework description outlines very specific requirements. You are welcome to try other things and report your results. However, no amount of extra work can be used as a substitution of the actual requirements.
9. Do ask for help if anything is unclear, but do it in a timely manner (e.g., by e-mail, Discord or during the Office Hours).