Lab session of Image Analysis

BE 4. Mathematical Morphology

Data Duration: 4h.
The material of this lab session can be found on **Chamilo**.

Instructions Submit a **report** for each group (binome) of the session in a unique **pdf** , name it **`LabX_Name1_Name2`**, with **`X`** denoting the number of the lab session and **`Name1,2`** your surnames. Upload it in the folder corresponding to your group and lab in **Chamilo**

Deadline submission The material report should be submitted within a week from the lab work. The preparation has to be done individually and will be collected at the **beginning** of the lab.

Note you can use the command `addpath('images')` to add the folder `'images'` to the search paths Matlab looks into when calling `imread` and other such commands.

Objectives The objectives of this lab work is:
- Use Mathematical Morphology to extract features of an image

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0
0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0
0	0	0	1	1	1	1	0	1	0	0	1	1	1	0	0
0	0	1	1	1	1	0	1	1	1	1	1	1	0	0	0
0	1	1	1	1	0	0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	1	0	1	1	0	0	0	0	0	0
0	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	1	1	0	0	0	0	1	1	1	0	1	0	0	0	0
0	1	1	1	0	0	0	1	1	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 1: .

Preparation

- Draw the result of a dilatation, an erosion, an opening, and a closing of Fig.1 by a structuring element of 3 by 3

1 Familiarize yourself with the tools (max 45min)

Many operators defined in mathematical morphology framework are neighborhood operations: the result of an operation over one pixel depends on the pixel values in its neighborhood. The neighborhood is defined by a mask of given shape and size, called structuring element (SE).

1.1 Structuring element

- Type `doc strel` and have a look at the syntax for defining a SE and the different shapes available.
- Define a few structuring elements and see the masks produced. You can visualize a SE s using `imshow(getnhood(s))`.

1.2 Morphological operators

Some basic operators are listed below. B is a structuring element and $\overline{B}$ is the reflected SE (i.e., rotation of B of 180 degrees around its origin). Load the cameraman image. Apply each operator to the image and see the effects.

- **Erosion** $\epsilon_B(f)$ with `imerode`.
- **Dilation** $\delta_B(f)$ with `imdilate`.
- **Opening** $\gamma_B(f) = \delta_{\overline{B}}\left[\varepsilon_B(f)\right]$ with `imopen`.
- **Closing** $\phi_B(f) = \varepsilon_{\overline{B}}\left[\delta_B(f)\right]$ with `imclose`.
- **White Top Hat** (Residues of an opening) $WTH(f) = f - \gamma_B(f)$.
- **Black Top Hat** (Residues of a closing) $BTH(f) = \phi_B(f) - f$.

1.3 Other useful operators

Apply this operations to the cameraman image (Note: some functions may be more effectively applied in cascade to appropriate morphological operators)

- **Image thresholding**: this can be done with the instruction `B=I>t`. (Note: As this is a dynamical indicization, the output variable type will be a `logical`, which has to be properly converted).
- **Regional extrema**: finds the local minima or maxima of a grayscale image, with `imregionalmax` and `imregionalmin`.
- **Median filter**, with `medfilt2`.
- **Regions Labeling** with `bwlabel` (Note: This instruction works on binary images; specifically it assigns a different label to connected regions identified by 1s).

2 Image processing

2.1 Extracting the tablets

The goal is to design an algorithm for extracting the tablets (*tablets.png*) in order to detect the missing one. For this, we want to generate a binary image with ones at the pixels belonging to a tablet.

- What are the characteristics of the objects of interest?
- **Approach 1** Define a sequence of operations based on image thresholding (the threshold value can be defined manually) to isolate the tablets and post-processing (with which filter?) to remove spurious regions. Describe which operator you choose.
- **Approach 2** Try now to approach the same problem but by applying an appropriate morphological operation (which should "flatten" the regions of the tablets) followed by a regional max. (Hint: use a structuring element that approaches the size and shape of the tablets.)

2.2 Broken cookies

We would like to detect the broken cookies (*cookies_broken.png*).

- Propose a solution with the tools at your disposal and test it. (Hint: exploit the fact that the broken cookies are slightly smaller than the others.)

2.3 Area of chocolates

The goal is to design an algorithm for counting and estimating the area (in number of pixel) of the chocolates (*chocos.jpg*).

- (a) Let us first extract the chocolates. Similarly as before, we can process the image in order to derive a binary image showing the objects of interest.
- (a-bis) How do you deal with the grid around the chocolates?
- (b) Once you have a satisfactory binary image, use `bwlabel` to label the regions. The result of this operation is an image where the background is set to zero and each region has a different value.
- (b-bis) How can you exploit this image to compute the average area of the chocolates? (Hint: use `hist`)

2.4 Caps of bottles

The goal is to design an algorithm for extracting in *bottles.png* the caps of the bottles (i.e., the result should be again a binary image with ones at the pixels belonging to the caps).
Let us have a look at the image. Unfortunately, the caps have similar graylevels and similar size to the box.

- How can we separate them? (Hint: exploit the fact that the caps and box have different shapes)

2.5 Retina

Load the image *retina.png*.

- Extract the vessels.
- Extract the optical nerve (the bright blob at the upper right corner of the image).
- Enhance the original image by improving its contrast since the parts of the image close to the borders are darker. To this end, first estimate the background of the image and you will then subtract it to the original image. (In order to avoid saturation, recast variables in double format before subtraction, e.g., $I_{enhanced} = double(I_{original}) - double(I_{background})$)

2.6 Aligned cookies

Load the image *cookies_lines_misal.png*.

- During production, the cookies are aligned in columns. Define an algorithm for detecting the columns in which the cookies are not aligned.