**HW4 Report**

R10922067 林云雲

**Introduction**

This program applies binary morphology operations to a binary image, including dilation, erosion, opening, closing, and hit-and-miss.

**Usage**

Place the source image and main.py under the same directory. Run the following command in the terminal.

python3 main.py -s <source> -b <binary\_threshold>

Parameters

-s <source> : the file path of source image, default = lena.bmp

-b <binary\_threshold> : the binary threshold, default = 128

**Binary Morphology**

* 1. **Dilation**

For each *white* pixel in the input binary image, superimpose the octagonal kernel (3, 5, 5, 5, 3) with origin at (2, 2) on top of the binary image so that the origin coincides with the input pixel. For each kernel pixel with value 1, set the value of the corresponding pixel in the binary image to 1. The result image is saved as dilation.bmp as shown below.



* 1. **Erosion**

For each pixel in the input binary image, superimpose the octagonal kernel (3, 5, 5, 5, 3) with origin at (2, 2) on top of the binary image so that the origin coincides with the input pixel. If all kernel pixels with value 1 coincides with a white pixel in the binary image, i.e. the entire kernel fits into the white parts of the image, then preserve only the origin pixel. If it does not fit, the original value in the binary image remains. The result image is saved as erosion.bmp as shown below.



* 1. **Opening**

Apply dilation and then erosion with the octagonal kernel. The result image is saved as opening.bmp as shown below.



* 1. **Closing**

Apply erosion and then dilation with the octagonal kernel. The result image is saved as closing.bmp as shown below.



* 1. **Hit-and-Miss**

First, generate the complement of the binary image. Then, erode the binary image and its complement with L-shape kernel J and K with origin at (1, 1) respectively. Finally, intersect the two eroded images. The result image is saved as hit\_and\_miss.bmp as shown below.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| L-shape kernel J | | |  | L-shape kernel K | | |
| 0 | 0 | 0 |  | 0 | 1 | 1 |
| 1 | 1 | 0 |  | 0 | 0 | 1 |
| 0 | 1 | 0 |  | 0 | 0 | 0 |

