Experiment 5

Morphological Operations

The word *Morphology* denotes a branch of biology that deals with the form and structure of animals and plants. Here, we use the same word in the context of Mathematical Morphology, which means as a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries, skeletons etc. We use Morphology for shape analysis & shape study. Mathematical Morphology is used to extract image components that are useful in the representation and description of region shape.

Mathematical morphology involves a convolution-like process using various shaped kernels, called structuring elements. Every Operation has two elements are present – Input Image (almost Binary) and Structuring element. The operation's results depend upon the structuring element that is chosen. The structuring elements are mostly symmetric: squares, rectangles, and circles. Most common morphological operations are – *Dilation* and *Erosion.* The operations can be applied iteratively in selected order to effect a powerful process - *Opening* and *Closing.*

Let A be the image undergoing analysis, B be the structuring element, then *Dilation* is described by:

$$A \oplus B = \{c \in Z^2 \mid c = a + b \quad for \quad some \quad a \in A, b \in B\}$$

Erosion is defined as

$$A \Theta B = \{x \in Z^2 \mid (B)_x \subseteq A)$$

Opening is defined as

$$A \circ B = (A \ \Theta \ B) \oplus B$$

Closing is defined as

$$A \bullet B = (A \oplus B) \ \Theta \ B$$

Problem Objective

Write C++/Image-J modular functions to perform the following operations on the given test image, *ricegrains_mono.bmp*. All functions must support binary images.

1. Make separate functions for erosion, dilation, opening, and closing of binary images
   a. **ErodeBinary, DilateBinary**
      Input: Binary image, structuring element
      Output: Eroded/dilated image
   b. **OpenBinary, CloseBinary**
      Input: Binary image, structuring element
      Output: Opened/closed image

Use structuring elements:

1	1

1	1	1
1	1	1
1	1	1

0	1	0
1	1	1
0	1	0

and 9 × 9, 15 × 15 kernels of grayvalue = 1 (reference point – centre pixel).

Note

1. Do not hardcode the filenames and/or image size into the code.
2. Show structuring element, input and morphed images together.
3. Use proper code commenting and documentation.
4. Use self explanatory identifiers for variables/functions etc.

References

1. Prof. P. K. Biswas, Lecture 33 - 36, *Video Lecture Series (NPTEL) on Digital Image Processing*
2. R. C. Gonzalez and R. Woods, *Digital Image Processing*, Reading, MA: Addison-Wesley, 1992.
3. http://www.mmorph.com/mmtutor1.0/html/mmtutor/mm030gray.html