Signal Estimation from Mixture in Images

Department of Electrical engineering
IIT Jodhpur

J Sandeep Narayan(B17EE035)
Devraj Meena(B17EE023)
Sabyasachi Pradhan(B17EE048)

Image segmentation

Subdivides a image into its constituent parts or multiple segments. (say, sets of pixels). Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images

- Discontinuity Based Approach: Edge detection
- Similarity Based Approach: Thresholding, Region growing

Introduction

Segmentation is based on two basic properties of gray scale levels

1. Discontinuity - that is, partition the image based on abrupt changes in intensity (like edges)
2. Similarity - that is, to partition the image into similar region (like thresholding, region growing, splitting, merging, etc)

Region growing

It is a procedure that group pixels or subregions into larger regions based on predefined criteria for growth.

The basic idea is to set a seed points from where these grow regions by appending to each neighbouring seed with similar pixel properties.

Schematics

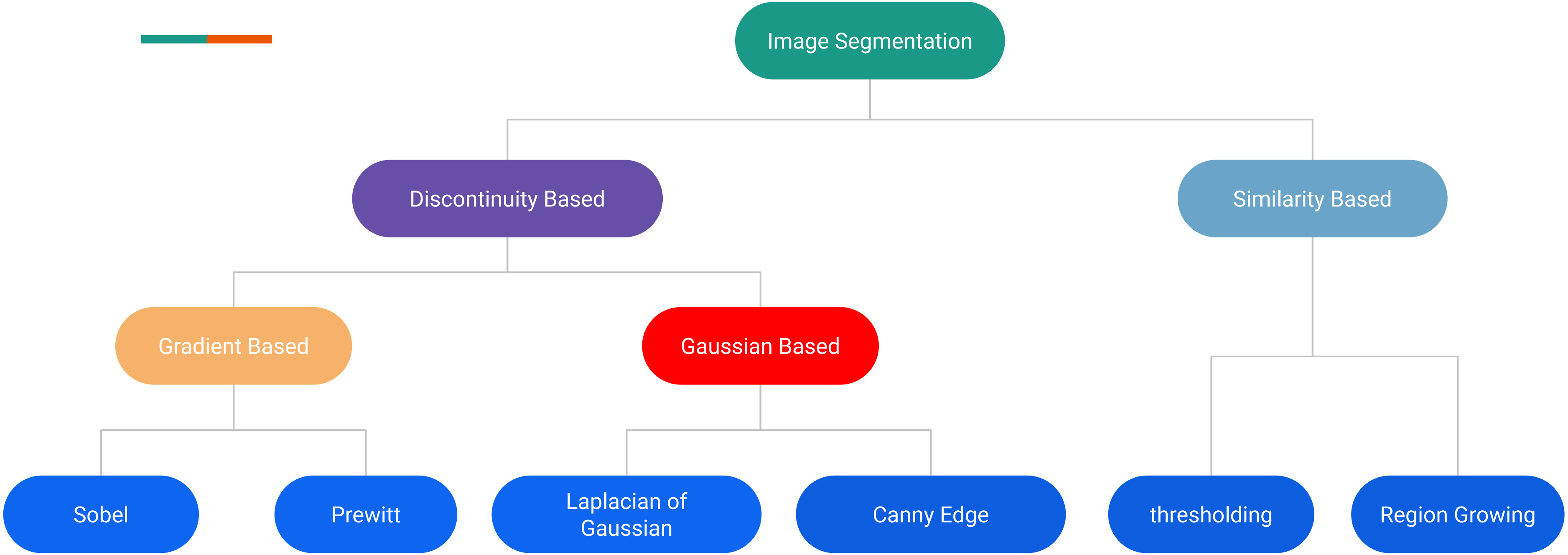


Image Segmentation
Discontinuity Based
Similarity Based
Gradient Based
Gaussian Based
Sobel
Prewitt
Laplacian of Gaussian
Canny Edge
thresholding
Region Growing

Image Edge Detection

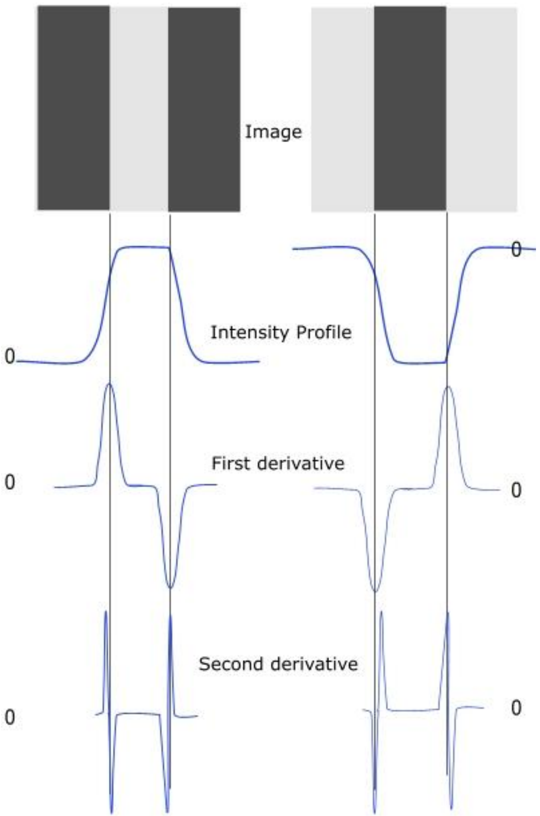
In images, edge detection refers to identifying regions where a rapid change in intensity occurs. Such regions generally act as boundaries between different features.

It is used for identifying and picking out the different features in an image, essentially separating images within an image.

Mathematical Rationale

As edge detection in images is based on detecting sudden discontinuities in the values of intensities, they generally deploy derivatives to identify the edges.

Either the maxima/minima of first derivative or the zero-crossing of second derivative is used to identify these discontinuities.



Pre-Processing

Most edge detection algorithms are sensitive to noise, or differences in illumination in the image.

To remove noise, generally noise filtering masks are used to smoothen out or blur the image.

To deal with uneven illumination, homomorphic filtering is used, where in the image is translated into a different domain by non-linear mapping. Linear filters can be used in the different domain, and the result is them mapped back to the original domain.

Mean filter or averaging filter

The **mean filter** is a simple sliding-window spatial **filter** that replaces the center value in the window with the **average** (**mean**) of all the pixel values in the window. The window, or kernel, is usually square but can be any shape

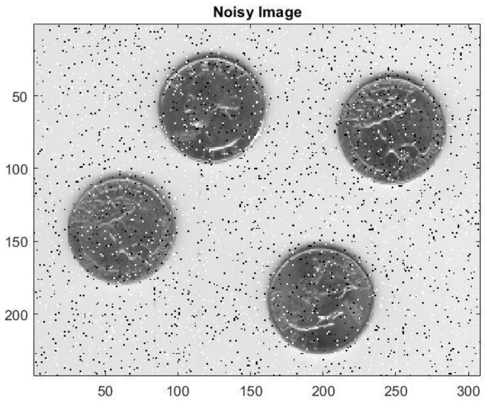
The main idea here is to each image pixel is replace by the average of its neighbouring pixels.

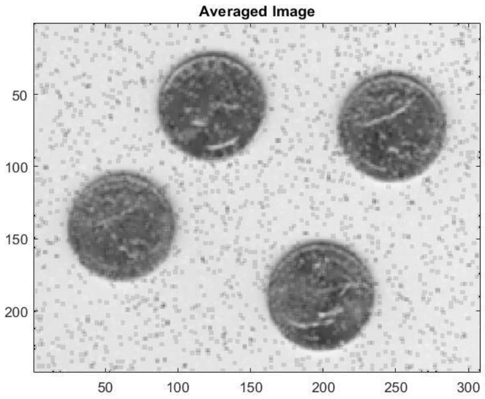
Median Filter

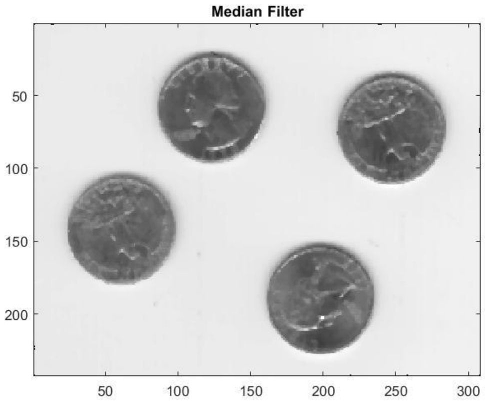
It is a non-linear digital filtering technique, which is used to remove noise from image or signal.

The main idea here is that each pixel of the image is replaced by the median of its neighbouring pixels, which is called window.

Simulation for Denoising images







Simulation for Homomorphic Filtering

Gradient-based Edge Detection

When just the first derivative is used to detect the edges, then it is referred to as Gradient-based.

Examples include Sobel-operator and Prewitt operator.

It works by detecting maximas or minimas in the derivative.

Sobel Operator

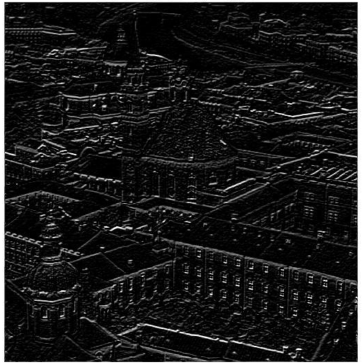
It involves convolving a simple 3x3 kernel in along the X-axis and Y-axis of the image independently.

It provides quite rough and inaccurate information, but simultaneously, is computationally inexpensive.

Simulation of Sobel Operator

Sobel - Vertical

Sobel - Secondary Diagonal



Sobel - Horizontal

Sobel - Edge Detected

Sobel - Principal Diagonal

Prewitt Operator

Similar in operation to Sobel Operator, but it doesn't place a focus on the pixels closer to the centre.

Quite rough and inaccurate, but again, it is also less computationally expensive.

-1	0	+1
-1	0	+1
-1	0	+1

G_x

+1	+1	+1
0	0	0
-1	-1	-1

G_y

Simulation of Prewitt Operator

Prewitt - Horizontal

Prewitt - Principal Diagonal

Prewitt - Vertical

Prewitt - Edge Detected

Prewitt - Secondary Diagonal

Drawbacks of Gradient Edge Detection

Gradient edge detection does not provide every accurate results, and instead gives thick, often smudged edges or broken edges.

Both the Sobel and Prewitt Operators are sensitive to noise.

Laplacian of Gaussian (LoG) Operator

The LoG operator uses two functions, the Gaussian Function and the Laplacian function together.

Gaussian Function: $$G(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp - \left(\frac{x^2 + y^2}{2\sigma^2} \right)$$

Laplacian Function: $$L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

Laplacian of Gaussian

The combined function:

$$LoG = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2 + y^2}{2\sigma^2}\right]e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Advantages:

➔ Easy to detect edges and their various orientations
➔ There is fixed characteristics in all directions

Disadvantages:

➔ Very sensitive to noise
➔ The localization error may be severe at curved edges
➔ It generates noisy responses that do not correspond to edges, so-called “false edges”

Laplacian of Gaussian

The Gaussian function used here is a two dimensional function, so only one kernel is generated.

Example of LoG Kernel with sigma= 1.4

0	1	1	2	2	2	1	1	0
1	2	4	5	5	5	4	2	1
1	4	5	3	0	3	5	4	1
2	5	3	-12	-24	-12	3	5	2
2	5	0	-24	-40	-24	0	5	2
2	5	3	-12	-24	-12	3	5	2
1	4	5	3	0	3	5	4	1
1	2	4	5	5	5	4	2	1
0	1	1	2	2	2	1	1	0

Laplacian of Gaussian (Contd.)

It is bigger and more complicated than the Sobel or Prewitt operator, and has a more provides more accurate results, but is simultaneously more complex computationally.

It is also very sensitive to noise, and also adds a lot of artificial noise to the final, sharpened image due to the double derivative it amplifies the noise present.

Simulation of LoG Edge Detection

Original Image

Normalized LoG with Edges

Final Image

Canny Edge Detection

The additional noise that LoG introduced to the image meant that a better algorithm for edge detection was needed.

Hence a multi-step algorithm was developed by John F. Canny in 1986.

Canny Edge Detection was lesser sensitive to noise than any of the traditional methods, and it was also far more accurate than the earlier discussed methods, but simultaneously, it was also much more complex a task computationally.

Algorithm

Canny Edge Detection works in 5 steps:

1. **Smoothing**: Blurring of the images to remove noise.
2. **Finding Gradients**: Marking the rough edges using gradient edge detection.
3. **Non-Maximum Suppression**:Eliminating those edges detected that are not local maximas.
4. **Double Thresholding**: Potential edges are then determined by thresholding the image.
5. **Edge tracking by Hysteresis**: The final edges are determined by suppressing those edges which are not connected to a very certain/strong edge.

Smoothing

Smoothing, or blurring, is done to suppress the noise in the image.

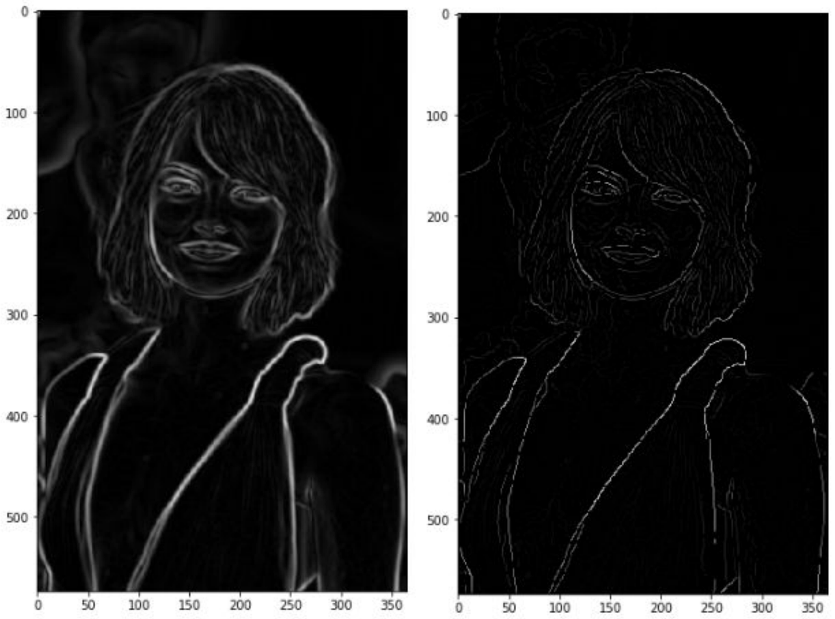
A gaussian filter is generally used for this, and smaller the kernel used, less obvious the blurring is.

Gradient Edge Detection

Usine a gradient edge detector like the Sobel operator, the rough edges are marked out.

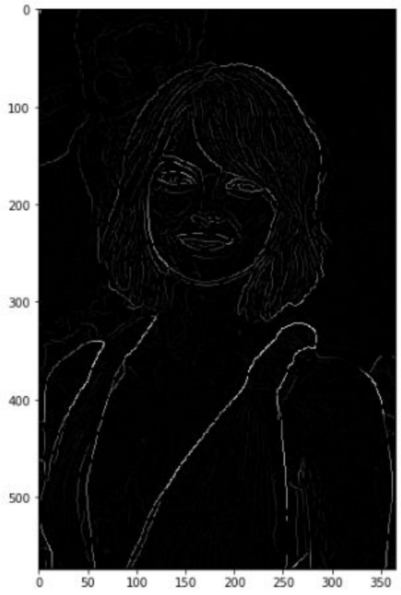
Non-Maximum Suppression

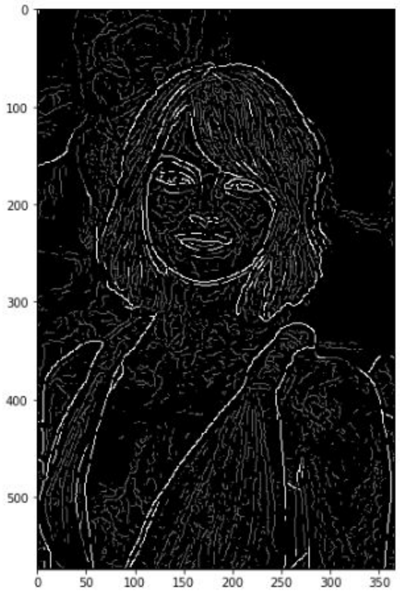
The thick edges that result from Gradient Edge Detection are thinned out by taking up the local maxima values only. That is, the maximum pixel value in the local region is retained and the rest are dropped.



Double Thresholding

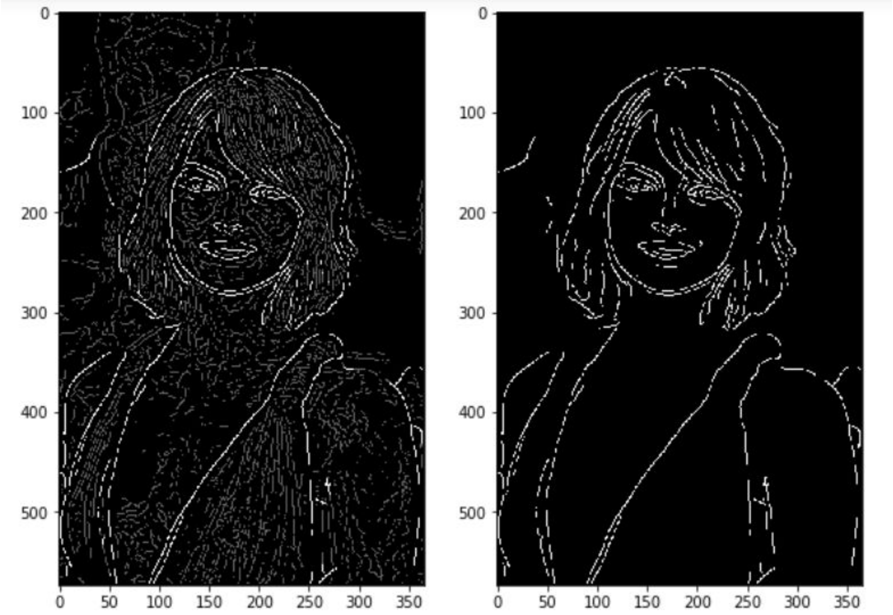
In thresholding, those pixel above a certain value are dropped. This specific process is called double thresholding because at least two thresholds are used, to separate out the weak and strong edges at the lower and higher thresholds respectively.



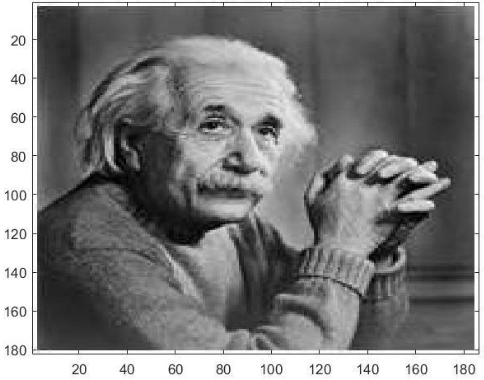


Edge Tracking by Hysteresis

In the final step, those weak edges which are connected to one strong edge are converted into strong edges, and the rest are dropped to give the final edges.

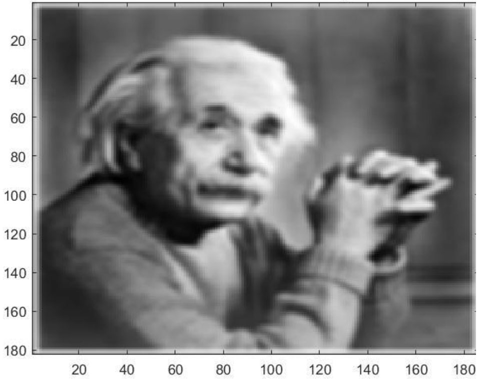


Original Image

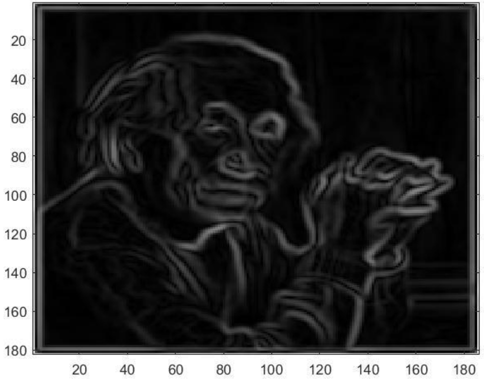


Sobel - Edge Detected

Gausian Smoothened Image



Gradient Magnitude



Non-Maximum Suppression

Double Thresholding

Edge Detection

Comparative Example

Clockwise from top left:

1. Original, gray-scale image
2. Gradient Edge Detection
3. LoG Operator
4. Canny Edge Detection

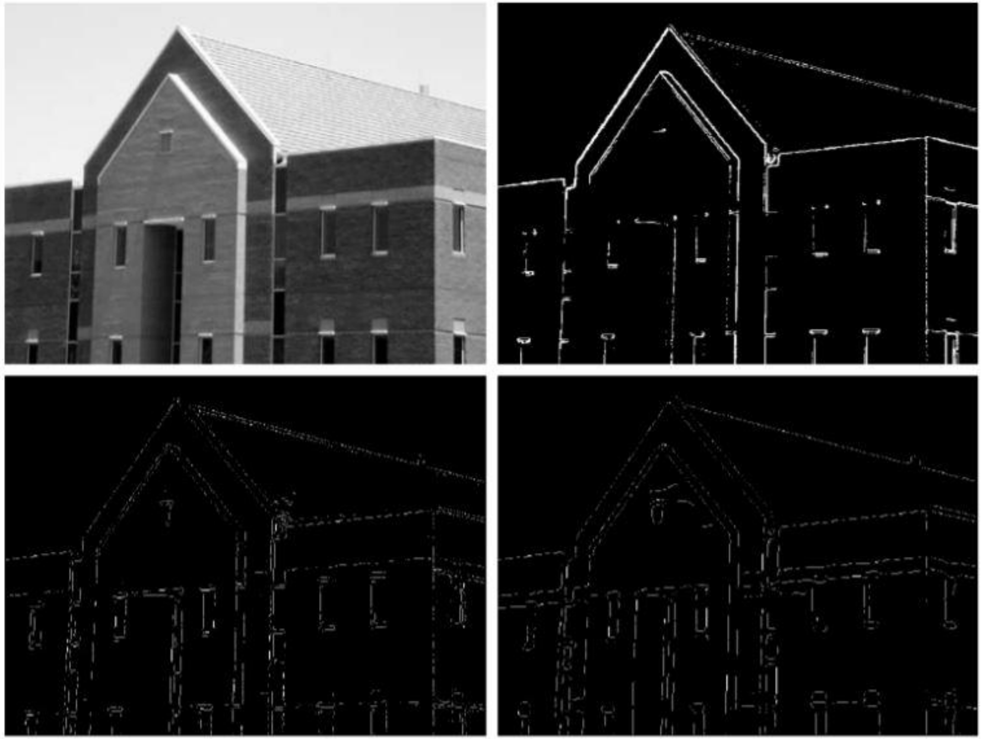


Image thresholding

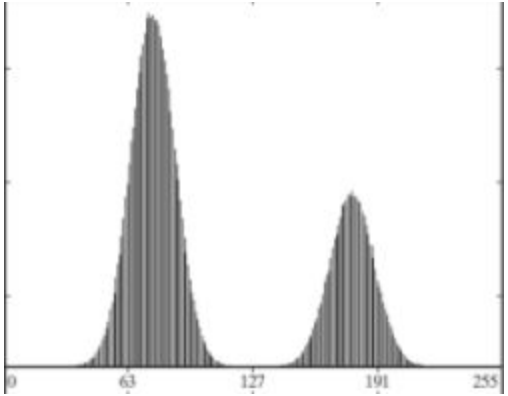
Image thresholding is used to convert the image pixels into binary levels.
The input to such a thresholding algorithm is usually a grayscale image and a threshold. The output is a binary image.

$$J(x, y) = \begin{cases} 0 & \text{if } I(x, y) < T \\ 1 & \text{otherwise.} \end{cases}$$

Global Thresholding

Global thresholding consists of setting an intensity value (threshold) such that all pixels having intensity value below the threshold belong to one phase, the remainder belong to the other.

Global thresholding is as good as the degree of intensity separation between the two peaks in the image.



Global thresholding through Otsu’s method

It is a method which involves iterating through the all possible threshold values and calculate a measure of spread for pixel level in each side of threshold

I. First it creates a histogram of the input image
II. Within class variance - Then it divides the histogram levels based on the given threshold into 2 levels
III. Between class variance - it takes into account variance difference of 2 clases

Global Threshold through Otsu Method

Morphological Process

- ➢ The shape analysis is easy in case of binary image.
- ➢ Approach for processing digital image based on its shape.
- ➢ The main aim is to extract useful information or features from the shape. Pixel location describe the shape.
- ➢ Set theory based process.

The components extracted are useful in the representation and description of region shape like,

- Image pre-processing : noise filtering, shape simplification
- Enhancing object structure : skeletonizing, thinning, thickening, convex hull,object marking
- Segmenting objects from background
- Quantitative description of objects : area, perimeter.

Primary Operations

Dilation : it is enlargement operation, which combines two sets using vector addition of set elements.

Let A and B be two substrate in 2D space

A- image input

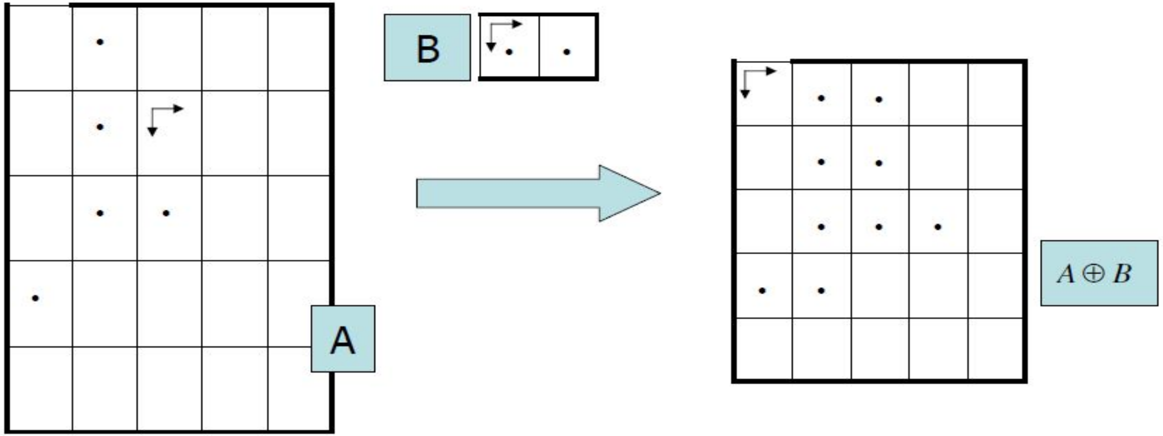
B- structuring element

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

Dilation

$A = \{(0,1), (1,1), (2,1), (2,2), (3,0)\}$

$B = \{(0,0), (0,1)\}$



$$A \oplus B = \{(0,1), (0,2), (1,1), (1,2), (2,1), (2,2), (2,3), (3,0), (3,1)\}$$

Erosion : it is reverse of dilation. It combines two sets using vector subtraction of set elements.

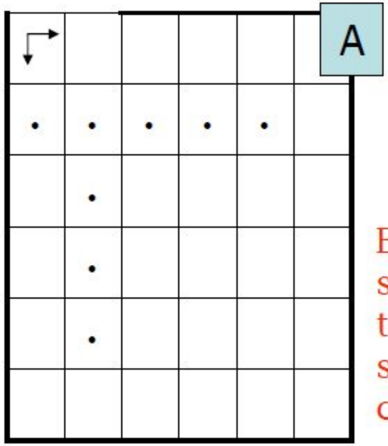
Erosion is given by -

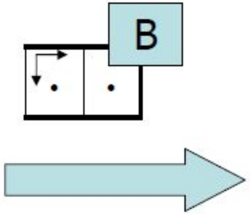
$$A \Theta B = \{x \in Z^2 \mid \text{ for every } b \in B, \text{ exist an } a \in A \text{ s.t. } x = a - b\}$$
$$= \{x \in Z^2 \mid x + b \in A \text{ for every } b \in B)$$

Erosion

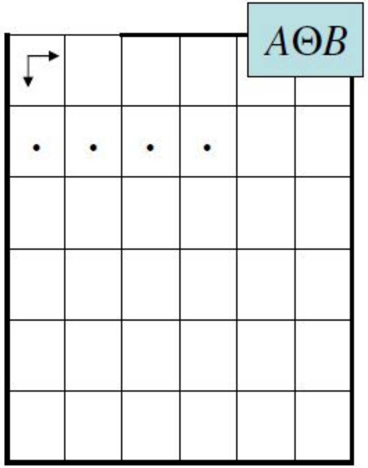
$A = \{(1,0), (1,1), (1,2), (1,3), (1,4), (2,1), (3,1), (4,1)\}$

$B = \{(0,0), (0,1)\}$





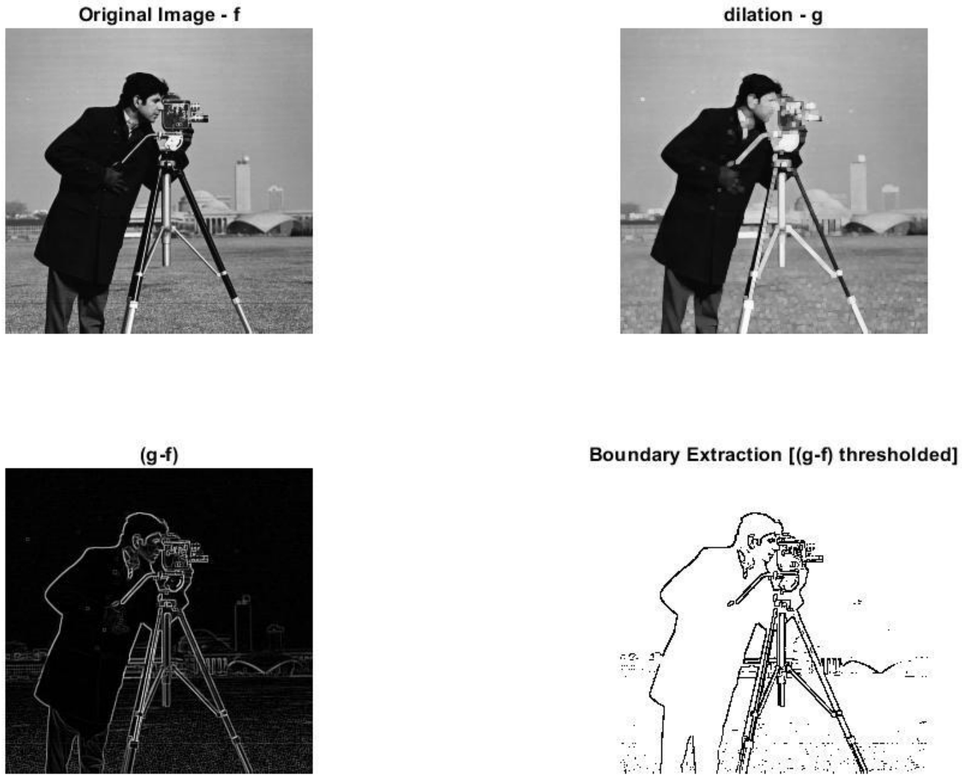
Erosion of A by B is the set of all points such that B, translated by set points of A is contained in A



$A\Theta B = \{(1,0), (1,1), (1,2), (1,3)\}$

Erosion with 3x3 element will disappear single pixel wide lines

Simulation for Morphological Process

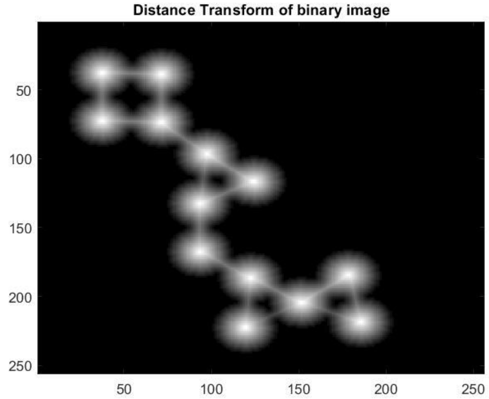


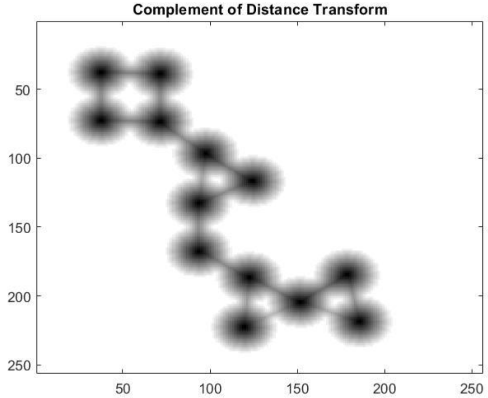
Watershed Algorithm

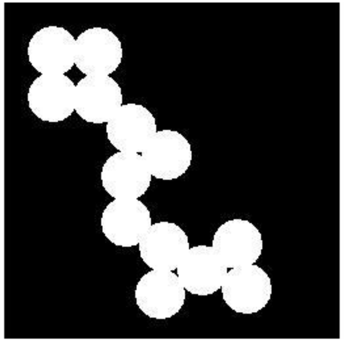
It is a algorithm for segmentation of different objects of an image.

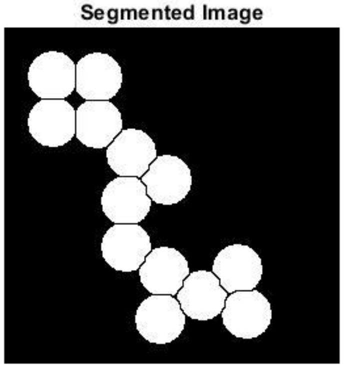
The watershed algorithm treats all the pixels values as a local topography . The algorithm floods basins from the markers until basins attributed to different markers meet on watershed lines. In many cases, markers are chosen as local minima of the image, from which basins are flooded.

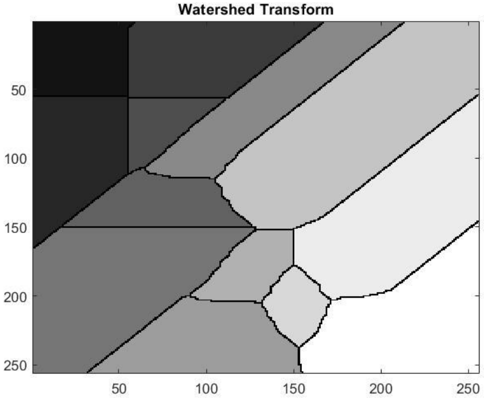
Simulation for Watershed











Conclusion

Hence, we have analysed different techniques of image segmentation.

Thank You