University of Kurdistan

Digital Image Processing (DIP)

Lecture 4: Image Segmentation

Instructor:

Kaveh Mollazade, Ph.D.

Department of Biosystems Engineering, Faculty of Agriculture, University of Kurdistan, Sanandaj, IRAN.

Contents

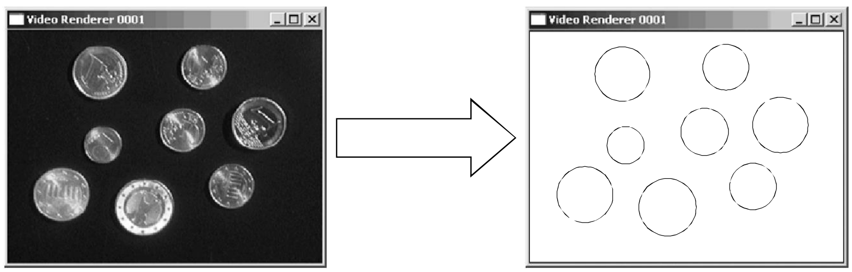
• So far we have been considering image processing techniques used to transform images for human interpretation. In this lecture we will begin looking at automated image analysis by examining the thorny issue of image segmentation:

– The segmentation problem
– Finding points, lines, and edges
– What is thresholding?
– Simple thresholding algorithms
– Adaptive thresholding

The segmentation problem

• Segmentation attempts to partition the pixels of an image into groups that strongly correlate with the objects in an image.

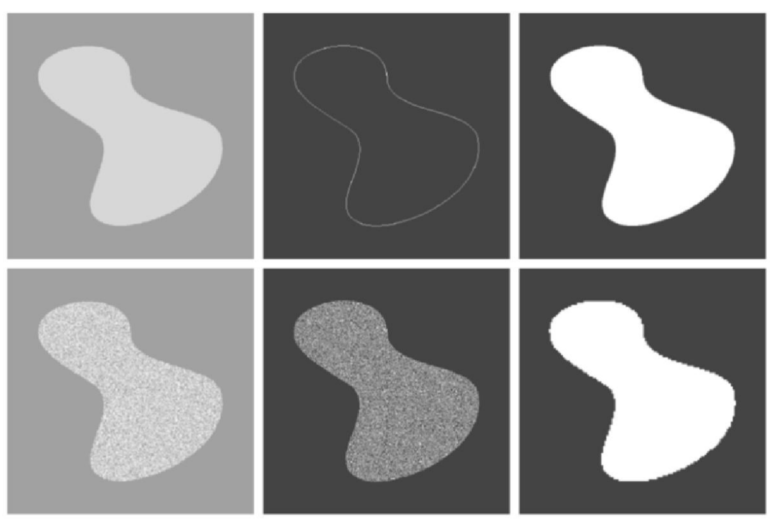
• Typically the first step in any automated computer vision application.



Gray-scale image segmentation techniques

Discontinuity-based

Similarity-based



Detection Of discontinuities

• There are three basic types of grey level discontinuities that we tend to look for in digital images:

- Points
- Lines
- Edges

• We typically find discontinuities using masks and correlation.

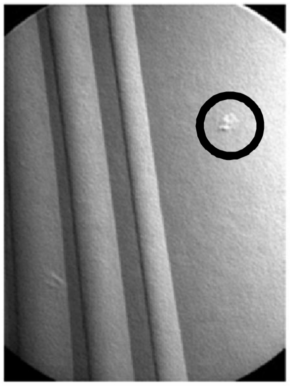
Point detection

• Point detection can be achieved simply using the mask below:

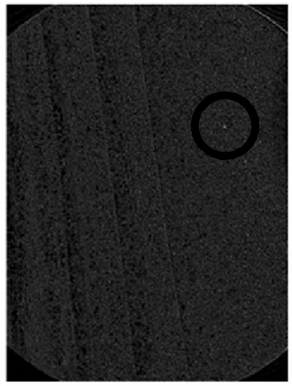
1	1	1
1	−8	1
1	1	1

• Points are detected at those pixels in the subsequent filtered image that are above a set threshold.

Point detection (cont…)



X-ray image of a turbine blade



Result of point detection

Result of thresholding

Line detection

- The next level of complexity is to try to detect lines.

- The masks below will extract lines that are one pixel thick and running in a particular direction.

−1	−1	−1
2	2	2
−1	−1	−1

Horizontal

−1	−1	2
−1	2	−1
2	−1	−1

+45°

−1	2	−1
−1	2	−1
−1	2	−1

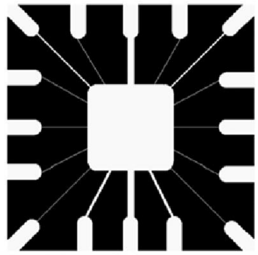
Vertical

2	−1	−1
−1	2	−1
−1	−1	2

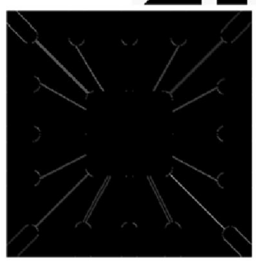
−45°

Line detection (cont…)

Binary image of a wire bond mask



After processing with -45° line detector

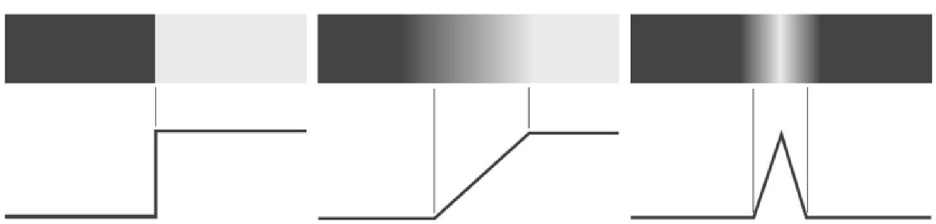


Result of thresholding filtering result

Edge models

An edge is a set of connected pixels that lie on the boundary between two regions.

Step (ideal) edge Ramp edge Roof edge



Edge models example

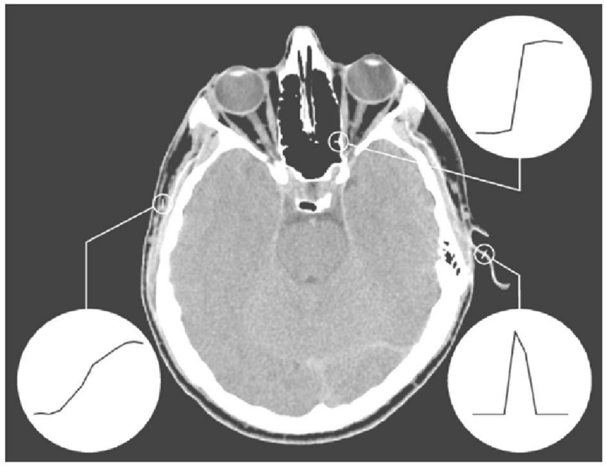
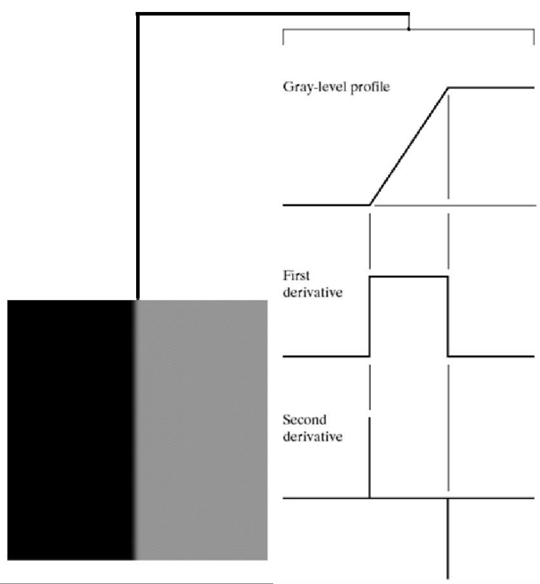


FIGURE 10.9 A 1508 × 1970 image showing (zoomed) actual ramp (bottom, left), step (top, right), and roof edge profiles. The profiles are from dark to light, in the areas indicated by the short line segments shown in the small circles. The ramp and "step" profiles span 9 pixels and 2 pixels, respectively. The base of the roof edge is 3 pixels. (Original image courtesy of Dr. David R. Pickens, Vanderbilt University.)

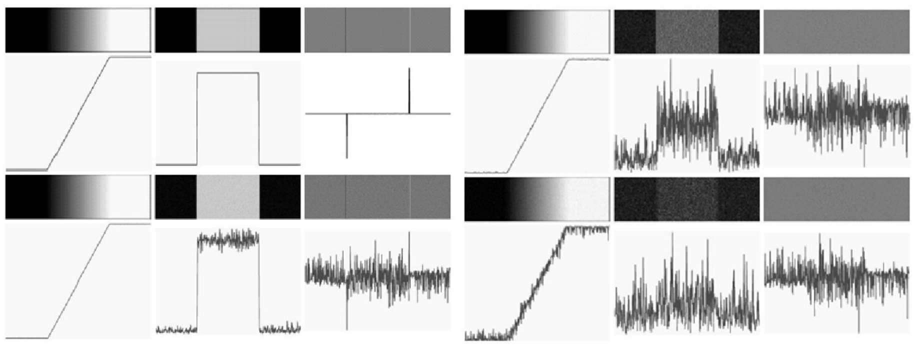
Edges & derivatives

- We have already spoken about how derivatives are used to find discontinuities.
- 1st derivative tells us where an edge is.
- 2nd derivative can be used to show edge direction.



Derivatives & noise

Derivative based edge detectors are extremely sensitive to noise. We need to keep this in mind.



Common edge detectors

Given a 3*3 region of an image the following edge detection filters can be used.

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	−1	−1		−1	0	1
0	0	0		−1	0	1
1	1	1		−1	0	1

Prewitt

−1	0		0	−1
0	1		1	0

Roberts

−1	−2	−1		−1	0	1
0	0	0		−2	0	2
1	2	1		−1	0	1

Sobel

Prewitt and Sobel masks for detecting diagonal edges

0	1	1	−1	−1	0
−1	0	1	−1	0	1
−1	−1	0	0	1	1

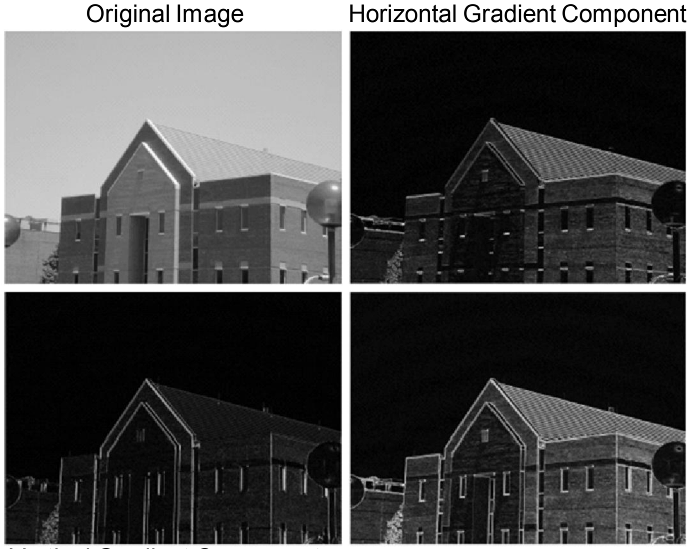
Prewitt

0	1	2	−2	−1	0
−1	0	1	−1	0	1
−2	−1	0	0	1	2

Sobel

Edge Detection Example

Original Image Horizontal Gradient Component



Vertical Gradient Component Combined Edge Image

Edge detection problems

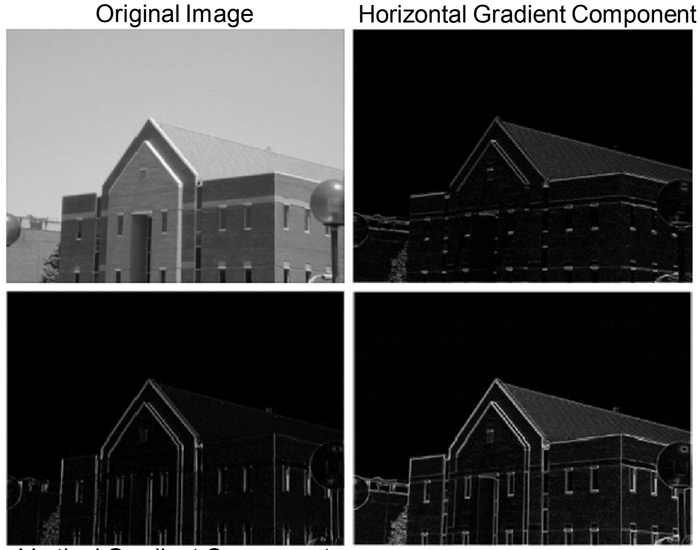
• Often, problems arise in edge detection in that there are is too much detail.

• For example, the brickwork in the previous example.

• One way to overcome this is to smooth images prior to edge detection.

Edge detection example with smoothing

Original Image | Horizontal Gradient Component



Vertical Gradient Component | Combined Edge Image

Laplacian edge detection

• We encountered the 2nd-order derivative based Laplacian filter already.

0	−1	0
−1	4	−1
0	−1	0

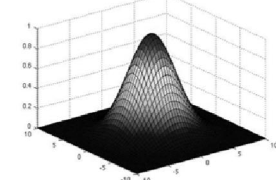
−1	−1	−1
−1	8	−1
−1	−1	−1

• The Laplacian is typically not used by itself as it is too sensitive to noise.

• Usually when used for edge detection the Laplacian is combined with a smoothing Gaussian filter.

Laplacian of Gaussian

$$G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$



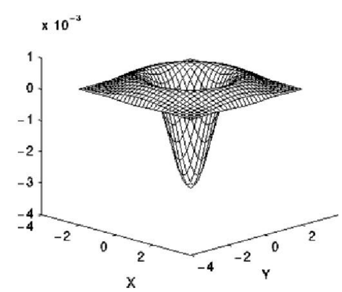
• The Laplacian of Gaussian (or Mexican hat) filter uses the Gaussian for noise removal and the Laplacian for edge detection.

$$\nabla^2 G(x, y) = \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2}$$

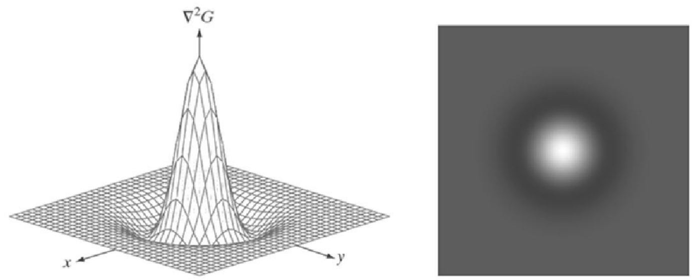
$$= \frac{\partial}{\partial x}\left[\frac{-x}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}\right] + \frac{\partial}{\partial y}\left[\frac{-y}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}\right]$$

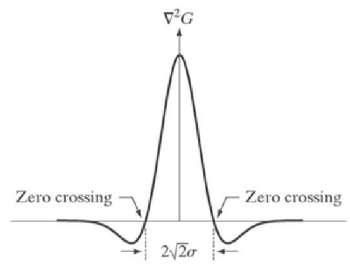
$$= \left[\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}} + \left[\frac{y^2}{\sigma^4} - \frac{1}{\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$\text{LoG} = \nabla^2 G(x, y) = \left[\frac{x^2 + y^2 - 2\sigma^2}{\sigma^4}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$



Laplacian of Gaussian (cont …)



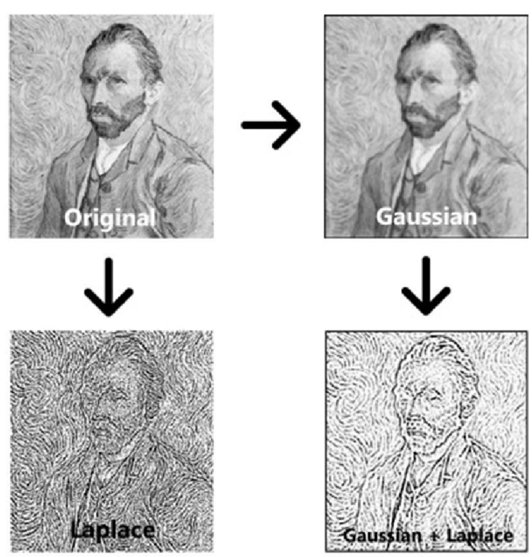


0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

a b
c d

FIGURE 10.21 (a) Three-dimensional plot of the *negative* of the LoG. (b) Negative of the LoG displayed as an image. (c) Cross section of (a) showing zero crossings. (d) 5×5 mask approximation to the shape in (a). The negative of this mask would be used in practice.

Laplacian of Gaussian example



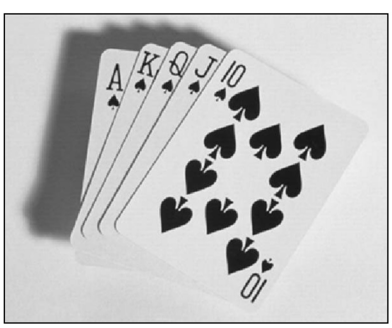
Thresholding

- Thresholding is usually the first step in any segmentation approach.
- We have talked about simple single value thresholding already.
- Single value thresholding can be given mathematically as follows:

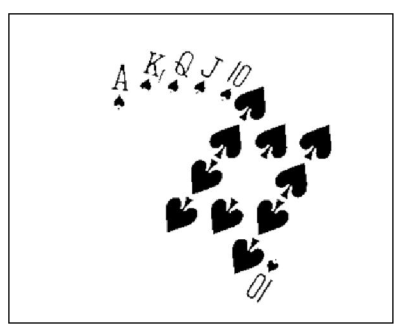
$$g(x,y) = \begin{cases} 1 \; if \;\; f(x,y) > T \\ 0 \; if \;\; f(x,y) \le T \end{cases}$$

Thresholding example

- Imagine a poker playing robot that needs to visually interpret the cards in its hand.



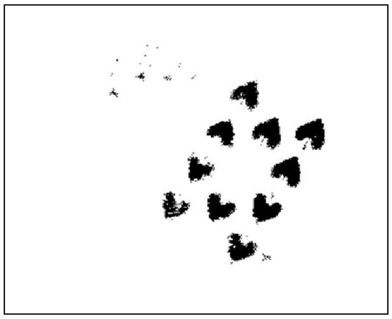
Original Image



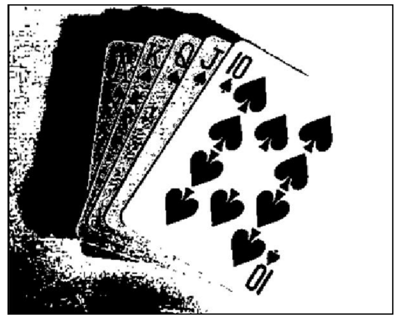
Thresholded Image

But be careful

- If you get the threshold wrong the results can be disastrous.



Threshold Too Low



Threshold Too High

Basic global thresholding

- Based on the histogram of an image.

- Partition the image histogram using a single global threshold.

- The success of this technique very strongly depends on how well the histogram can be partitioned.

Basic global thresholding algorithm

The basic global threshold, T, is calculated as follows:

1. Select an initial estimate for T (typically the average grey level in the image).
2. Segment the image using T to produce two groups of pixels: G_1 consisting of pixels with grey levels >T and G_2 consisting pixels with grey levels ≤ T.
3. Compute the average grey levels of pixels in G_1 to give μ_1 and G_2 to give μ_2.

Basic global thresholding algorithm (cont …)

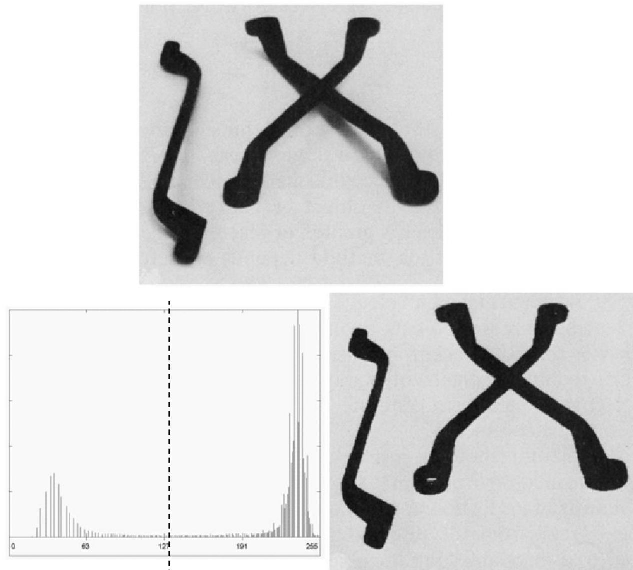
4. Compute a new threshold value:

$$T = \frac{\mu_1 + \mu_2}{2}$$

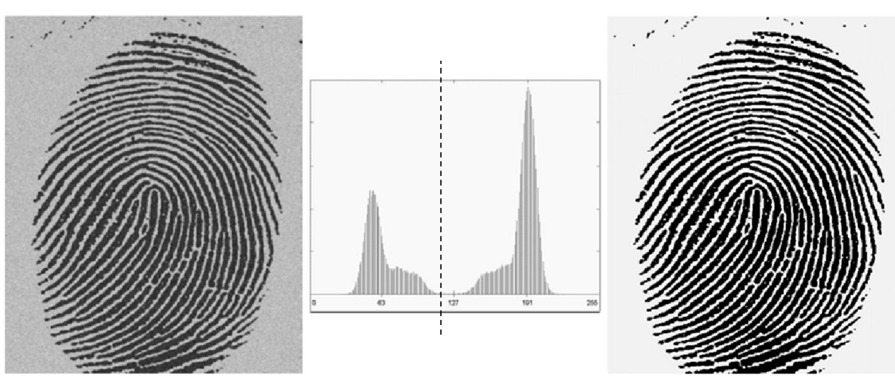
5. Repeat steps 2 – 4 until the difference in T in successive iterations is less than a predefined limit ΔT.

This algorithm works very well for finding thresholds when the histogram is suitable.

Thresholding example 1



Thresholding example 2



Optimized general thresholding using Otsu's method

• Segmentation is based on "region homogeneity".

• Region homogeneity can be measured using variance (i.e., regions with high homogeneity will have low variance).

• Otsu's method selects the threshold by maximizig the *within-class variance*.

Algorithm of Optimized general thresholding sing Otsu's method

The Otsu's thrsholding method finds the optimum threshold, k, as follows:

1. Compute the normalized histogram of input image. Show the histogram components as pi, where i=0, 2, …, L-1.

2. Compute the cumulative sets $P_1(k)$ for k=0, 1, …, L-1 using the following formula:

$$P_1(k) = \sum_{i=0}^{k} p_i$$

Algorithm of Optimized general thresholding sing Otsu’s method (cont …)

3. Compute the cumulative means (Intensity means) m(k) for k=0, 1, …, L-1 using the following formula:

$$m(k) = \sum_{i=0}^{k} i p_i$$

4. Compute the global intensity mean (whole image intensity mean) m_G as:

$$m_G = \sum_{i=0}^{L-1} i p_i$$

5. Compute the within-class variance $\sigma_B^2(k)$ for k=0, 1, …, L-1 by:

$$\sigma_B^2(k) = \frac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]}$$

Algorithm of Optimized general thresholding sing Otsu’s method (cont …)

6. Compute the Otsu’s threshold k*, where k* is a measure of k in which $\sigma_B^2(k)$ is maximum.

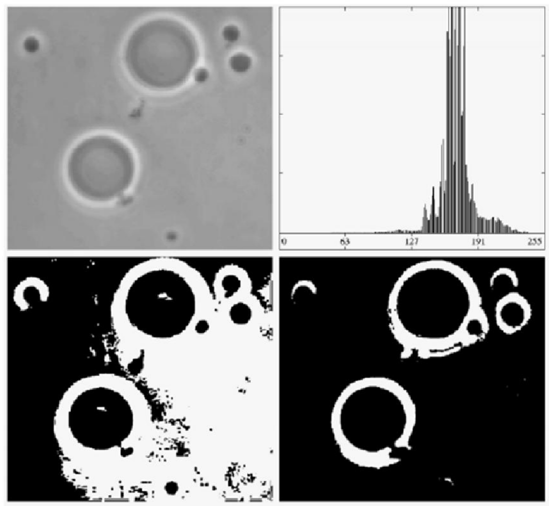
6. Compute the segregate value η* by investigating the following formula in k=k*:

$$\eta(k) = \frac{\sigma_B^2(k)}{\sigma_G^2}$$

where σ_G^2 is the global variance of whole image intensity values:

$$\sigma_G^2 = \sum_{i=0}^{L-1} (i - m_G)^2 p_i$$

Otsu's thresholding example

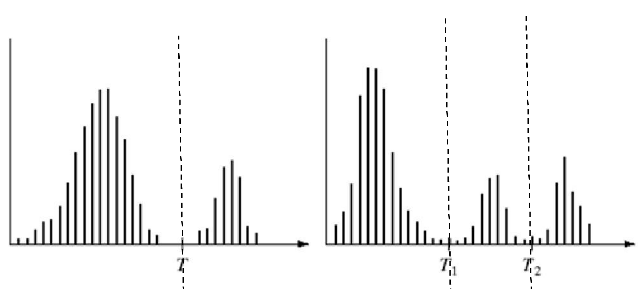


a b
c d

FIGURE 10.39 (a) Original image. (b) Histogram (high peaks were clipped to highlight details in the lower values). (c) Segmentation result using the basic global algorithm from Section 10.3.2. (d) Result obtained using Otsu's method. (Original image courtesy of Professor Daniel A. Hammer, the University of Pennsylvania.)

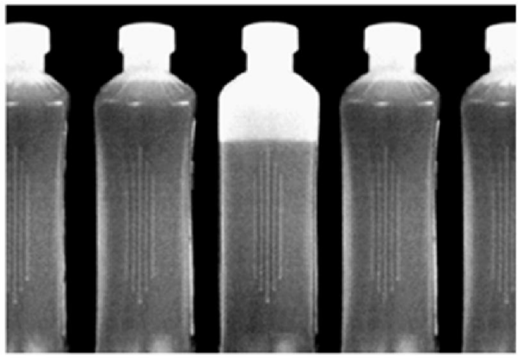
Problems with single value thresholding

- Single value thresholding only works for bimodal histograms.
- Images with other kinds of histograms need more than a single threshold.

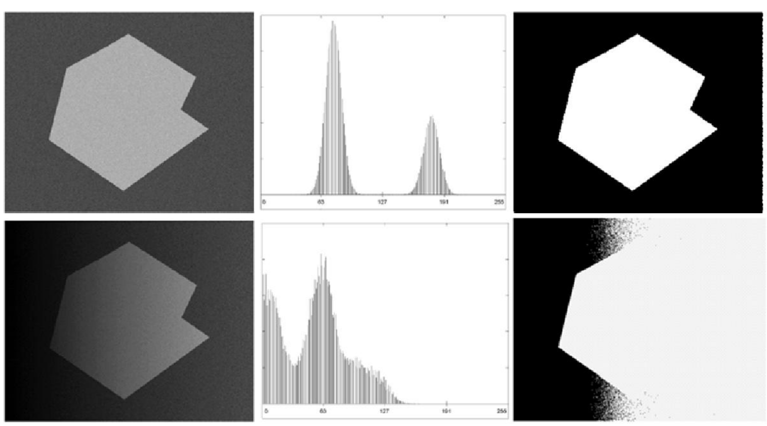


Problems with single value thresholding (cont…)

- Let's say we want to isolate the contents of the bottles.
- Think about what the histogram for this image would look like.
- What would happen if we used a single threshold value?



Single value thresholding and illumination



Uneven illumination can really upset a single valued thresholding scheme.

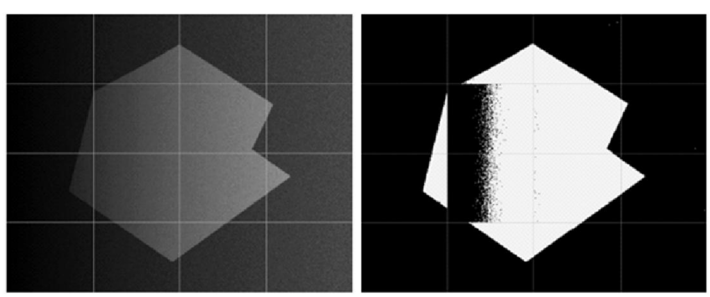
Basic adaptive thresholding

• An approach to handling situations in which single value thresholding will not work is to divide an image into sub images and threshold these individually.

• Since the threshold for each pixel depends on its location within an image this technique is said to *adaptive*.

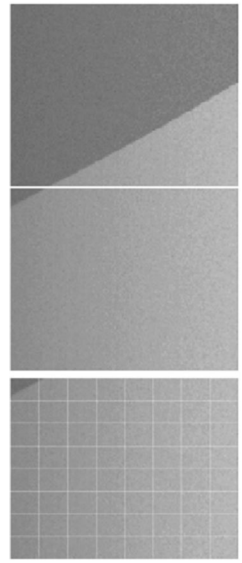
Basic adaptive thresholding example

• The image below shows an example of using adaptive thresholding with the image shown previously.



As can be seen success is mixed. But, we can further subdivide the troublesome sub images for more success.

Basic adaptive thresholding example (cont…)



• These images show the troublesome parts of the previous problem further subdivided.

• After this sub division successful thresholding can be achieved.

Summary

• In this lecture we have begun looking at segmentation, and in particular edge detection and thresholding.

• Edge detection is massively important as it is in many cases the first step to object recognition.

• We saw the basic global thresholding and Osu's thresholding algorithms and their shortcomings. We also saw a simple way to overcome some of these limitations using adaptive thresholding.