Outline

- Thresholding
- Region-based Segmentation
- Segmentation Using the Watershed Algorithm

Thresholding

- Thresholding is an operation used to group pixels in a gray level image based on similarities between gray levels
- The goal is to partition the image into several groups, where similarity in each group is at a high level and similarity between groups is low
- When the image is to be partitioned into two groups, a single threshold T is sufficient, then the thresholding of an image $f(x, y)$ results in a binary image $g(x, y)$

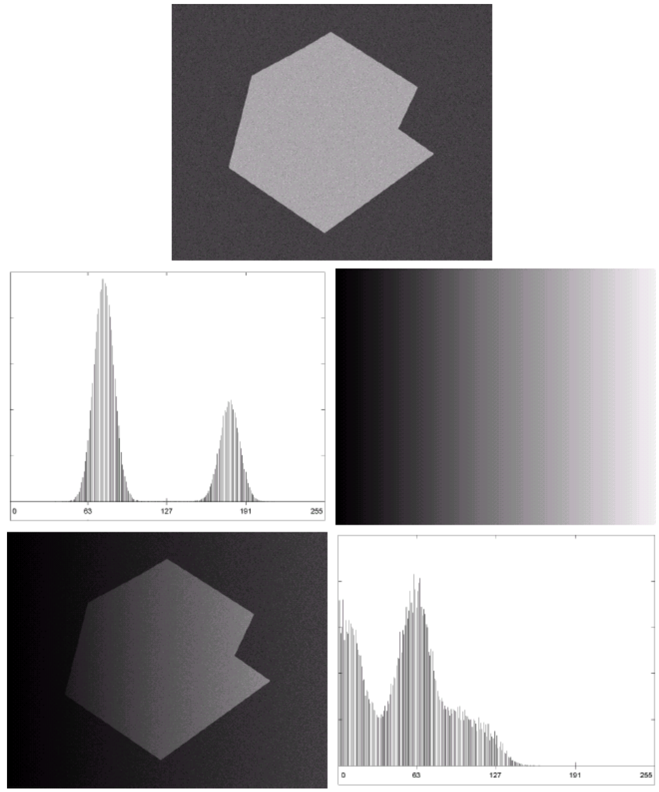
$$g(x,y) = \begin{cases} 1 & f(x,y) > T \\ 0 & \text{otherwise} \end{cases}$$

- If there is a single T value for the whole image then the thresholding is global
- If T is a function of local characteristics then the thresholding is local
- If T is allowed to change continuously with the spatial coordinates then the thresholding is adaptive

Thresholding (Cont.)

- Although thresholding looks like a simple task when the underlying image is actually composed of two groups, it becomes complicated because of noise and the image acquisition
- For example a non-uniform illumination would transform an image with two separable groups into a more complicated image which is difficult to threshold

Thresholding (Cont.)



- One solution would be to process the image to undo the effects of changes because of the image acquisition, in this example illumination, and then perform thresholding

A Simple Global Thresholding

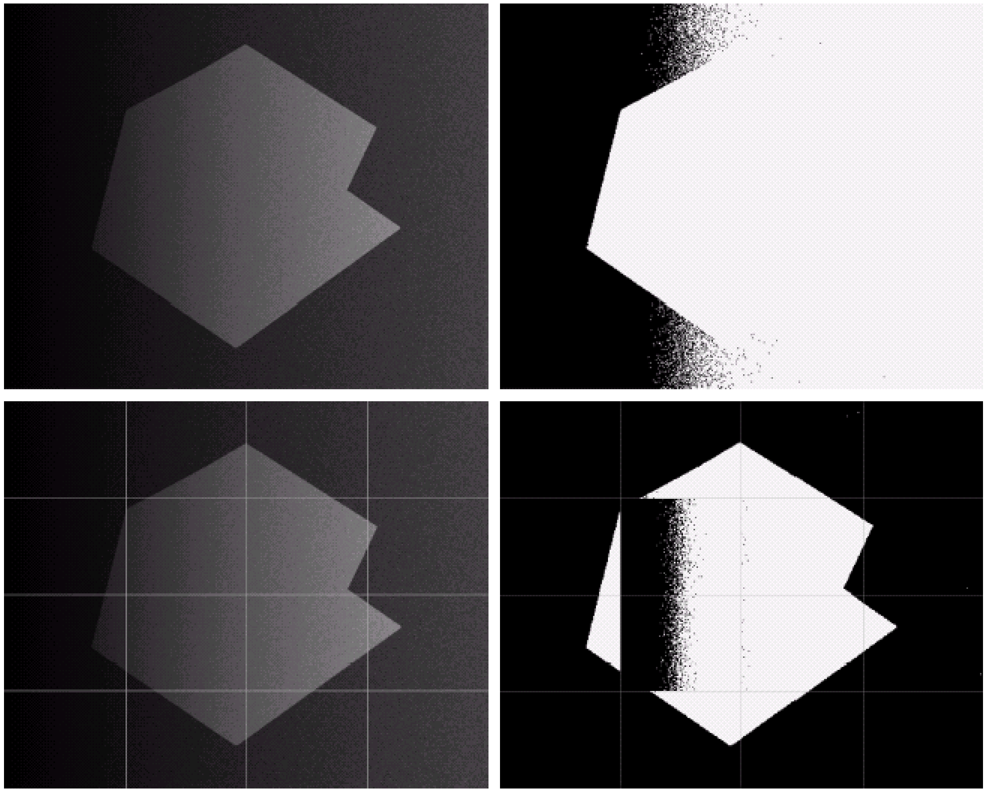
- In global thresholding, the only design parameter is the threshold T
- Then we need a "good" method to determined this value
- A simple approach would be
  - Select an initial T
  - Group the image into two using the current T value
  - Update your T value as the average of the means of two groups
  - Stop if change in T is small

Global Thresholding Using Otsu's Method

- There are hundreds of algorithms other than the previous one
- One of the classical methods is the Otsu's method
  - For each of the candidate T value, calculate the two group's variances
  - Select the T value that minimizes the sum of the variances

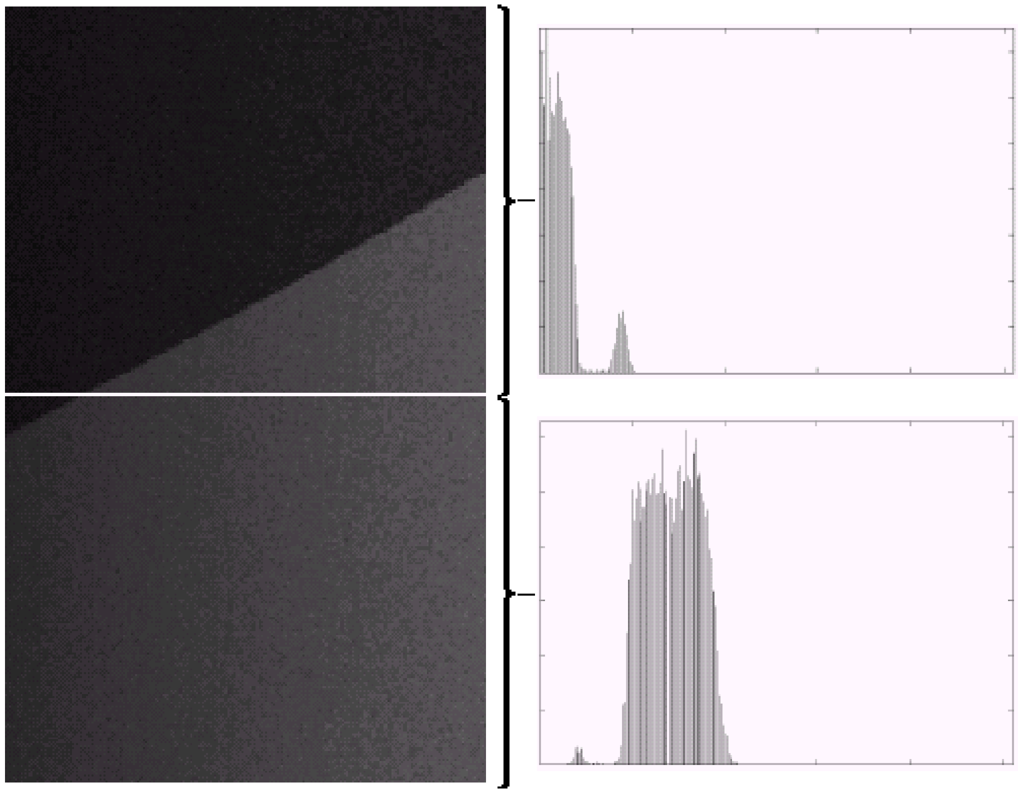
Adaptive Thresholding

- We can divide the image into subimages and then perform thresholding for each of the subimages with approximately uniform illimunation



Adaptive Thresholding (Cont.)

- Examples of subimages that are thresholded properly and not properly



- The problematic subimages are the ones that do not have bimodal histogram causing problems in thresholding

Optimum Thresholding Methods

- The methods discussed until now, are ad-hoc thresholding methods with no solid mathematical foundation
- We now describe thresholding methods that are optimum in a predefined sense
- Assume that the image is composed of two groups of pixels with histograms $p_1(z)$ and $p_2(z)$, where z denotes the gray levels
- The image histogram is a combination of these two groups

$$p(z) = P_1 p_1(z) + P_2 p_2(z)$$

where $P_1 + P_2$ is one

Optimum Thresholding Methods (Cont.)

- We will select the T value that minimizes the average error
- Two types of error is possible: (i) a pixel is selected as group 1, although in fact it belonged to group 2, (ii) a pixel is selected as group 2, although in fact it belonged to group 1
- We can calculate these probabilities as

$$E_1(T) = \int_{-\infty}^{T} p_2(z)\mathrm{d}z$$

$$E_2(T) = \int_{T}^{\infty} p_1(z)\mathrm{d}z$$

- The average error is then

$$E(T) = P_2 E_1(T) + P_1 E_2(T)$$

Optimum Thresholding Methods (Cont.)

- The optimum T is the T value that minimizes $E(T)$
- Taking the derivative and equating to zero gives

$$P_1 p_1(T) = P_2 p_2(T)$$

- The value for T can be calculated given that the PDF's and weights are known
- In practice these are not known and we need to estimate them
- We can simplify this estimation of the PDF's by assuming that the PDF's are Gaussian

Optimum Thresholding Methods (Cont.)

- For Gaussian PDF's we have

$$p(z) = \frac{P_1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{(z-\mu_1)^2}{2\sigma_1^2}} + \frac{P_2}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{(z-\mu_2)^2}{2\sigma_2^2}}$$

- Optimum T can be found by minimizing the error for this special forms of PDF's
- The resulting equation is a qaudratic one with two optimum T's
- For the special case of equal variances and $P_1 = P_2 = 0.5$ we have

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

Optimum Thresholding Methods (Cont.)

- Another approach (instead of using Gaussian) would be to simply not assume a certain form of PDF but minimize the mean squared error (MSE) to find a fitting PDF

$$e = \frac{1}{n}\sum_{i=1}^{n}[p(z_i) - h(z_i)]^2$$

- Either without a PDF form, or with a PDF form (to estimate means and variances) require numerical methods to find the optimum PDF parameters or the PDF values

Local Thresholding Using Edges

- In this case we will use boundary information to perform thresholding
- We use the gradient and the laplacian to decide whether a pixel is on a boundary or not
- Then the thresholded image (consisting of three types of pixel) is

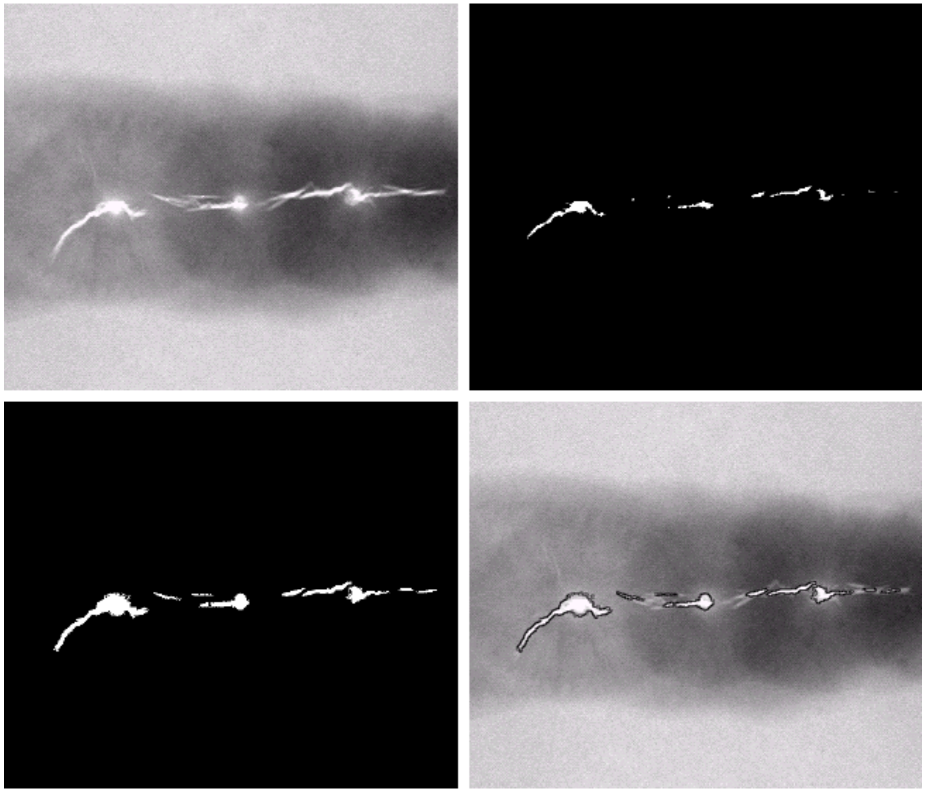
$$s(x,y) = \begin{cases} 0 & \nabla f < T \\ A & \nabla f \geq T \quad \nabla^2 f \geq 0 \\ B & \nabla f \geq T \quad \nabla^2 f < 0 \end{cases}$$

- Then the pixels are grouped as background+object, background to object edges, object to background edges
- The image is then analyzed to determine regions of object, and these are assigned the value 1, and the rest assigned the value 0

Region Growing Segmentation

- We now attempt to find the segments/regions of an image directly rather than using boundaries as in the previous discussion
- We usually start from a single point, and grow around it by adding the pixels that are similiar with respect to a property
- The seed points can be determined based on the prior information, or we can calculate the value of a cost function (in terms of the property of interest) for all pixels, and these cost function values will show clusters corresponding to segments
- The important factors to be determined in a region-growing segmentation procedure are
  - Seed points
  - Property that determines similarity
  - Stopping rule

Region Growing Segmentation: An Example



Region Growing Segmentation: An Example

- In this x-ray segmentation, our goal is to extract faulty welding
- We select brightest points as seeds
- Property of growing is that: the gray value of a candidate pixel and the seed pixel is less than a certain value
- The candidate pixel must be connected (4 or 8) to at least one pixel in the segment
- Stop when no pixels can be found satisfying similarity conditions

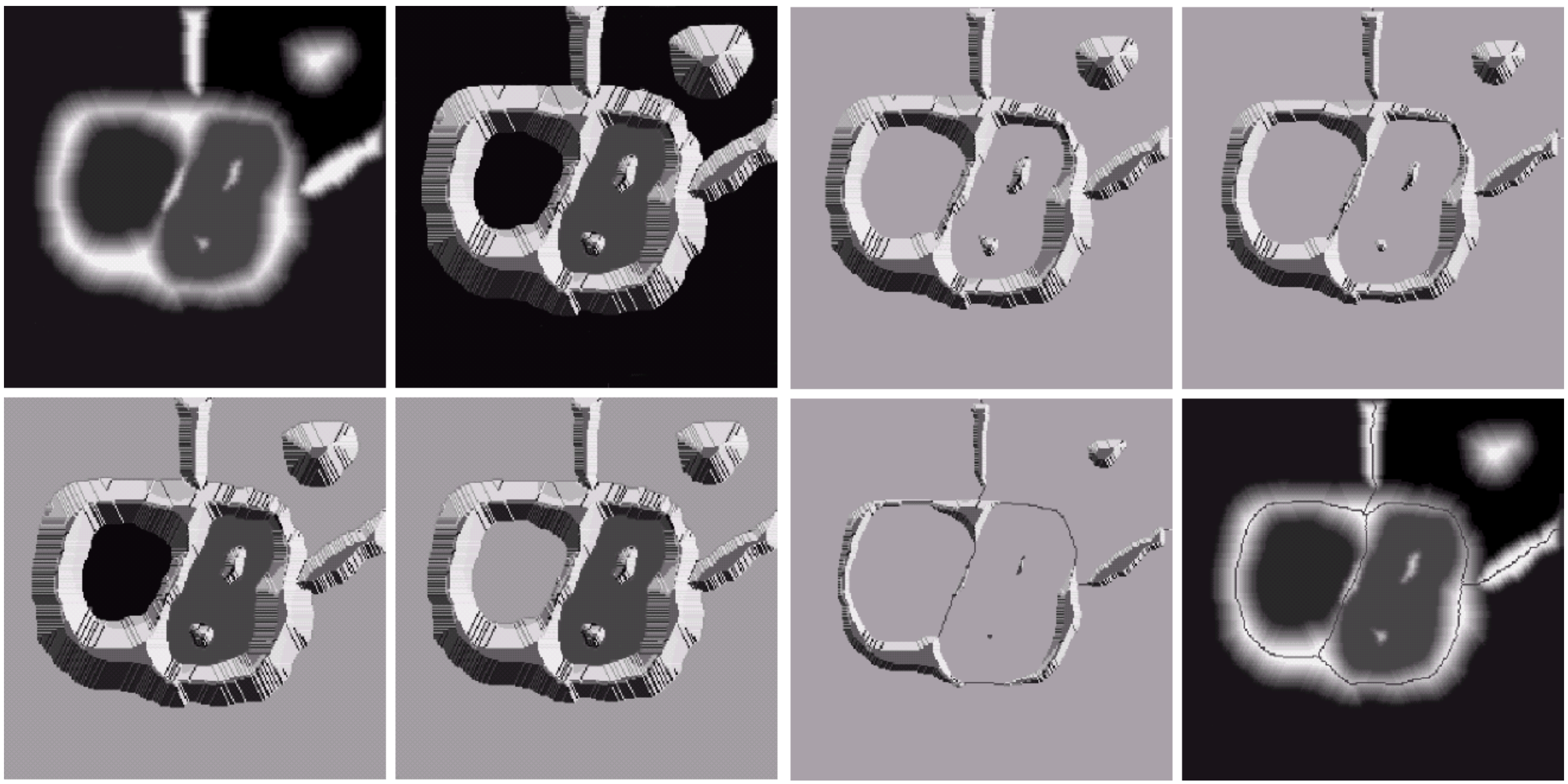
Segmentation Using Region Splitting and Merging

- Alternatively to region growing, we can choose not to start from single pixels called seeds
- We then partition the image into arbitrary regions, and then merge regions that are similar wrt a certain property, and split the ones that are different wrt a certain property
- The steps of the algorithm is as follows
  - Divide into 4 and then check if the regions are similar in property
  - For the regions of which all pixels are not similar in property, continue to divide
  - After all divisions are performed, check adjacent segments, if two are similar in property, merge them

Segmentation Using the Watershed Algorithm

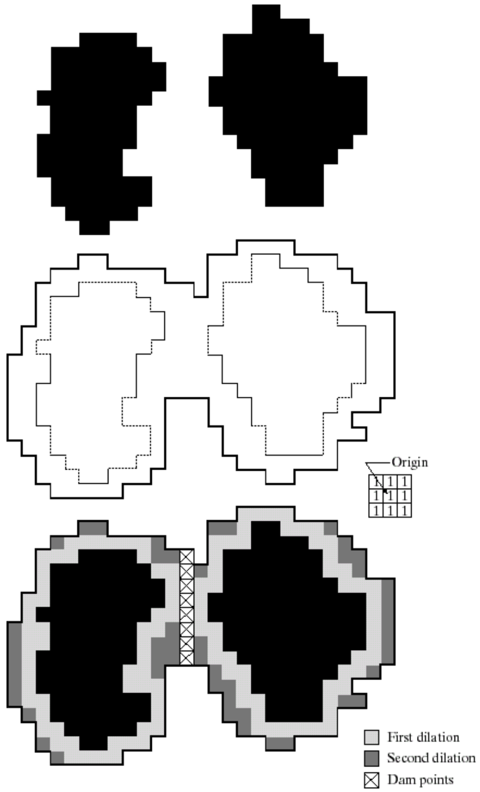
- This approach is fundamentally different from the previous methods
- We view the image in three dimensions, the spatial coordinates, and the gray levels
- Then all pixels in an image are part of three groups: (i) regional minimums, (ii) watersheds, where we know exactly where a drop of water would go when droppted, (iii) crest line where a drop of water could go towards more than one minimum
- Watershed algorithm can be thought as follows:
  - imagine the whole (3-D with gray levels) image being a bucket
  - We fill this bucket from below using the minimums
  - Water rises and at certain times, these waters merge together
  - When that happens we know what there is a boundary at those points, and we build dams to prevent merging

Segmentation Using the Watershed Algorithm - Illustration



Segmentation Using the Watershed Algorithm - Dam Construction

- We can use dilation to construct dams when a spilover occurs



Segmentation Using the Watershed Algorithm - Dam Construction

- Dilate the two segments (before merging) using the two conditions: (i) Locate the dilating structure's center on the segment, (ii) Do not dilate if the result causes two regions to merge
- The points that are part of the merged regions but not part of the dilated segments form the dam that we need