Lecture 12 - Image Segmentation (BASICS): Thresholding, Region Growing, Clustering

Dr. Ulas Bagci

bagci@ucf.edu
Image Segmentation

- **Aim:** to partition an image into a collection of sets of pixels
  - Meaningful regions (coherent objects)
  - Linear structures (line, curve, ...)
  - Shapes (circles, ellipses, ...)

![Image Segmentation Examples](image.png)
Image Segmentation

• **Aim:** to partition an image into a collection of set of pixels
  – Meaningful regions (coherent objects)
  – Linear structures (line, curve, ...)
  – Shapes (circles, ellipses, ...)

• Content based image retrieval
• Machine Vision
• Medical Imaging applications (tumor delineation,..)
• Object detection (face detection,..)
• 3D Reconstruction
• Object/Motion Tracking
• Object-based measurements such as size and shape
• Object recognition (face recognition,..)
• Fingerprint recognition,
• Video surveillance

...
Image Segmentation

• In computer vision, image segmentation is one of the oldest and most widely studied problems
  – Early techniques -> region splitting or merging
  – More recent techniques -> Energy minimization, hybrid methods, and deep learning
Image Segmentation Methods

- Thresholding
- Region based methods (region growing,..)
- Clustering (k-means, mean shift,..)
- Machine Learning based methods
- Energy minimization methods (MRF,..)
- Shape based methods (level set, active contours)
- Graph-based methods (graph-cut, random walk,..)

Energy minimization methods
Basics of Image Segmentation

- **Definition**: *Image segmentation* partitions an image into regions called *segments*.
Basics of Image Segmentation

- **Definition**: *Image segmentation* partitions an image into regions called **segments**.

Image segmentation creates segments of connected pixels by analyzing some **similarity criteria**: intensity, color, texture, histogram, features, ...
Image Binarization

- Image binarization applies often just one global threshold $T$ for mapping a scalar image $I$ into a binary image.
Image Binarization

• Image binarization applies often just one global threshold $T$ for mapping a scalar image $I$ into a binary image

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

- Image binarization applies often just one global threshold $T$ for mapping a scalar image $I$ into a binary image

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

- The **global threshold** can be identified by an optimization strategy aiming at creating “large” connected regions and at reducing the number of small-sized regions, called *artifacts*. 
Image Binarization

- **Thresholding**: Most frequently employed method for determining threshold is based on histogram analysis of intensity levels.

**DIFFICULTIES**
1. The valley may be so broad that it is difficult to locate a significant minimum
2. Number of minima due to type of details in the image
3. Noise
4. No visible valley
5. Histogram may be multi-modal
Thresholding Example

Original Image

Thresholded Image
Thresholding Example 2

Threshold Too Low

Threshold Too High
Thresholding Example 3
Thresholding Example-4
Otsu Thresholding

• **Definition:** The method uses the grey-value histogram of the given image $I$ as input and aims at providing the best threshold in the sense that the “overlap” between two classes, set of object and background pixels, is minimized (i.e., by finding the best balance).
Otsu Thresholding

**Definition:** The method uses the grey-value histogram of the given image I as input and aims at providing the best threshold in the sense that the “overlap” between two classes, set of object and background pixels, is minimized (i.e., by finding the best balance).

Otsu’s algorithm selects a threshold that maximizes the between-class variance $\sigma_b^2$. In the case of two classes,

$$\sigma_b^2 = P_1(\mu_1 - \mu)^2 + P_2(\mu_2 - \mu)^2 = P_1 P_2 (\mu_1 - \mu_2)^2$$

where $P_1$ and $P_2$ denote class probabilities, and $\mu_i$ the means of object and background classes.
## Otsu Thresholding

- **Definition:** The method uses the grey-value histogram of the given image \( I \) as input and aims at providing the best threshold in the sense that the “overlap” between two classes, set of object and background pixels, is minimized (i.e., by finding the best balance).

- Otsu’s algorithm selects a threshold that maximizes the between-class variance \( \sigma_b^2 \). In the case of two classes,

\[
\sigma_b^2 = P_1 (\mu_1 - \mu)^2 + P_2 (\mu_2 - \mu)^2 = P_1 P_2 (\mu_1 - \mu_2)^2
\]

where \( P_1 \) and \( P_2 \) denote class probabilities, and \( \mu_i \) the means of object and background classes.

- Let \( C_I \) be the relative cumulative histogram of an image \( I \), then \( P_1 \) and \( P_2 \) are approximated by \( c_I(u) \) and \( 1 - c_I(u) \) respectively.
- \( u \) is assumed to be the chosen threshold.
Otsu Thresholding Algorithm

1: Compute histogram $H_I$ for $u = 0, \ldots, G_{\text{max}}$;
2: Let $T_0$ be the increment for potential thresholds; $u = T_0$; $T = u$; and $S_{\text{max}} = 0$;
3: while $u < G_{\text{max}}$ do
4: Compute $c_I(u)$ and $\mu_i(u)$ for $i = 1, 2$;
5: Compute $\sigma_b^2(u) = c_I(u)[1 - c_I(u)][\mu_i(u) - \mu_2(u)]^2$;
6: if $\sigma_b^2(u) > S_{\text{max}}$ then
7: $S_{\text{max}} = \sigma_b^2(u)$ and $T = u$;
8: end if
9: Set $u = u + T_0$
10: end while

\[
P_1 = \sum_{i=0}^{u} p(i) \quad \mu_1 = \sum_{i=0}^{u} ip(i) / P_1
\]
\[
P_2 = \sum_{i=u+1}^{G_{\text{max}}} p(i) \quad \mu_2 = \sum_{i=u+1}^{G_{\text{max}}} ip(i) / P_2
\]

probabilities \hspace{1cm} Class means

(a) Two distinct modes \hspace{1cm} (b) Overlapped modes
Example: Otsu Thresholding
Region Based Segmentation
Region Based Segmentation-Basics

**Region:**
A group of connected pixels with similar properties

Closed boundaries

Computation of regions is based on similarity

Regions may correspond to Objects in a scene or parts of objects

Spatial proximity + similarity
Region Growing

- For segment generation in grey-level or color images, we may start at one seed pixel \((x,y,I(x,y))\) and add recursively adjacent pixels that satisfy a “similarity criterion” with pixels contained in the so-far grown region around the seed pixel.
Region Growing

• For segment generation in grey-level or color images, we may start at one seed pixel \((x, y, I(x, y))\) and add recursively adjacent pixels that satisfy a “similarity criterion” with pixels contained in the so-far grown region around the seed pixel.

• Defining similarity criteria alone is not an effective basis for segmentation
• It is necessary to consider the adjacency spatial relationship between pixels
Region Growing

- For segment generation in grey-level or color images, we may start at one seed pixel \((x,y,I(x,y))\) and add recursively adjacent pixels that satisfy a “similarity criterion” with pixels contained in the so-far grown region around the seed pixel.

- Defining similarity criteria alone is not an effective basis for segmentation
- It is necessary to consider the adjacency spatial relationship between pixels

Algorithm

1. The absolute intensity difference between candidate pixel and the seed pixel must lie within a specified range.
2. The absolute intensity difference between a candidate pixel and the running average intensity of the growing region must lie within a specified range;
3. The difference between the standard deviation in intensity over a specified local neighborhood of the candidate pixel and that over a local neighborhood of the candidate pixel must (or must not) exceed a certain threshold.
Seeded Segmentation (Region Growing)

1. Choose the seed pixel
Seeded Segmentation (Region Growing)

1. Choose the seed pixel
2. Check the neighboring pixels and add them to the region if they are similar to the seed
Seeded Segmentation (Region Growing)

1. Choose the seed pixel
2. Check the neighboring pixels and add them to the region if they are similar to the seed
3. Repeat step 2 for each of the newly added pixels; stop if no more pixels can be added

\[ |\text{neighboring pixels} - \text{seed}| < \text{Threshold} \]
Ex: Muscle/Bone Segmentation in CT Scans
Ex: Muscle/Bone Segmentation in CT Scans
Ex: Muscle/Bone Segmentation in CT Scans
Ex: Muscle/Bone Segmentation in CT Scans
Ex: Muscle/Bone Segmentation in CT Scans
Region Growing Implementation

**growRegion:** red nodes are the “active_front” (queue or stack)

- Add seed into active_front
- Stop when active_front is empty

Object with small intensity variation within the image

**Algorithm:**
Remove pixel \( p \) from active_front and mark it as \( \text{region}[p] = 1 \).
Add all neighbors \( q \) such that:
\[
\text{region}[q] = 0, \quad |I_p - I_q| < T
\]
and set \( \text{region}[q] = -1 \).
Limitations of Region Growing

Note that a complete segmentation of an image must satisfy a number of criteria:

1) All pixels must be assigned to regions
2) Each pixel must belong to a single region only
3) Each region must be a connected set of pixels
4) Each region must be uniform
5) Any merged pair of adjacent regions must be non-uniform
Comparison of Thresholding and Region Growing

threshold $T \geq 10$

threshold $T \geq 11$

threshold $T \geq 12$

region growing with variance of 2 in respect to value 11 with reference to threshold $T \geq 11$
Region splitting and Merging Segmentation

• Region splitting:
  – Unlike region growing, which starts from a set of seed points, region splitting starts with the whole image as a single region and subdivides it into subsidiary regions recursively while a condition of homogeneity is not satisfied.
Region splitting and Merging Segmentation

• Region splitting:
  – Unlike region growing, which starts from a set of seed points, region splitting starts with the whole image as a single region and subdivides it into subsidiary regions recursively while a condition of homogeneity is not satisfied.

• Region merging:
  – Region merging is the opposite of splitting, and works as a way of avoiding over-segmentation
Region splitting and Merging Segmentation

• Region splitting:
  – Unlike region growing, which starts from a set of seed points, region splitting starts with the whole image as a single region and subdivides it into subsidiary regions recursively while a condition of homogeneity is not satisfied.

• Region merging:
  – Region merging is the opposite of splitting, and works as a way of avoiding over-segmentation
  – Start with small regions (2x2 or 4x4 regions) and merge the regions that have similar characteristics (such as gray level, variance).
Region splitting and Merging Segmentation

- RAG: region adjacency graph
- Quadtree for splitting (top-down) procedure

*RAG with adjacency relations (in red) for big black region.*
Region splitting and Merging Segmentation

**Algorithm:**

- If a region $R$ is inhomogeneous ($P(R)=$FALSE), then $R$ is split into four sub-regions.

- If two adjacent regions $R_i, R_j$ are homogeneous ($P(R_i \cup R_j)=$TRUE), they are then merged.

- The algorithm stops when no further splitting or merging is possible.
Region splitting and Merging Segmentation
Clustering Based Segmentation Methods
What is Clustering?

• Organizing data into classes such that:
  – High intra-class similarity
  – Low inter-class similarity

• Finding the class labels and the number of classes directly from the data (as opposed to *classification tasks*)
What is a natural grouping?
What is a natural grouping?

Clustering is subjective

Simpson’s Family  School Employees  Females  Males
High Intra-Class
High Intra-Class

Cluster by features

- Color
- Intensity
- Location
- Texture
- ....
Distance metrics

0.23
3
342.7
K-means Clustering
K-means Clustering
K-means Clustering
K-means Clustering
K-means Clustering
K-means Clustering
K-means Clustering
K-means Clustering
Questions