Segmentation
What is Segmentation?

- Underlying goal of image segmentation is to partition an image into multiple groups/regions

Source: The MathWorks
Why segment?

- Segmentation allows objects and regions to be analysed in a more meaningful manner.
- Some applications of segmentation:
  - Object tracking (e.g., people tracking for surveillance purposes)
  - Medical Image Analysis (e.g., tumor growth analysis)
  - Remote Sensing Analysis (e.g., determine the ratio of different types of sea-ice within a region)
  - Face recognition (e.g., partition face into individual parts for component recognition)
Some of the most common groups of segmentation algorithms are:
- Histogram based segmentation
- Clustering based segmentation
- Region growing segmentation
- Active contour based segmentation
- Watershed based segmentation
- Morphology based segmentation
Histogram based Segmentation

- One of the simplest and most efficient form of segmentation
- Steps:
  - Compute the histogram of the image
Steps:
- Analyse peaks of the histogram and determine an appropriate point of thresholding
Histogram based Segmentation

- Steps:
  - Determine an appropriate point of thresholding based on what we want (e.g., segment foreground and background)
Histogram based Segmentation

- Steps:
  - Set all pixels to a set of fixed values based on threshold
Otsu's Method

- **Goal:**
  - Find threshold that minimizes intra-class variance
- **What is intra-class variance?**

\[
\sigma^2_{\text{intra}}(T) = \sum_{i=0}^{T-1} p(i)\sigma^1_1(T) + \sum_{i=T}^{N-1} p(i)\sigma^2_2(T)
\]

- **Problem:** Expensive to compute!
Otsu's Method

- Solution: Minimizing intra-class variance equivalent to maximizing inter-class variance
- What is inter-class variance?

\[
\sigma^2_{\text{inter}}(T) = \sigma^2_{\text{total}}(T) - \sigma^2_{\text{inter}}(T)
\]

\[
\sigma^2_{\text{inter}}(T) = \sum_{i=0}^{T-1} p(i) [\mu_1(T) - \mu]^2 + \sum_{i=T}^{N-1} p(i) [\mu_2(T) - \mu]^2
\]
Otsu's Method

Since

\[ \mu(T) = \sum_{i=0}^{T-1} p(i)\mu_1(T) + \sum_{i=T}^{N-1} p(i)\mu_2(T) \]

- The inter-variance can be expressed as

\[ \sigma^2_{inter}(T) = \sum_{i=0}^{T-1} p(i) \sum_{i=T}^{N-1} p(i)[\mu_1(T) - \mu_2(T)]^2 \]
Otsu's Method

**Steps:**
- For each possible threshold $T$
  - 1. Determine the values within each of the clusters formed by $T$
  - 2. Find the mean of the clusters ($u_1$ and $u_2$)
  - 3. Compute the squared difference between the means
  - 4. Multiply result by the cumulative probability in cluster 1
  - 5. Multiply result by the cumulative probability in cluster 2
Otsu's Method

Steps:
- For each possible threshold $T$
  - 1. Determine the values within each of the clusters formed by $T$
  - 2. Find the mean of the clusters ($u_1$ and $u_2$)
  - 3. Compute the squared difference between the means
  - 4. Multiply result by the cumulative probability in cluster 1
  - 5. Multiply result by the cumulative probability in cluster 2
 Finite Mixture Models

- Suppose we know that there are $m$ clusters/classes in the image
- Suppose that the probability distribution of each cluster/class can be modeled using a parametric model (e.g., Gaussian, Gamma, Cauchy, etc.)
- Idea: We can model the probability distribution of the image as a mixture of $m$ different probability distributions, one for each cluster/class
Finite Mixture Models

• Formulation:

\[ f_X(x) = \sum_{i=1}^{m} a_i f_{Y_i}(x) \]

• Goal: Determine this set of \( m \) distributions and determine which pixel values belong to each cluster/class based on which of these distributions give the highest probability
Finite Mixture Models

- Example

![Graph showing two clusters](image-url)
Finite Mixture Models

- **Steps:**
  - Learn the parameters of the distribution models for the $m$ clusters
    - e.g., for Gaussian, learn the mean and standard deviation
  - For each possible pixel
    - 1. Determine the probability that its pixel intensity belongs to each of the $m$ clusters based on the distribution models
    - 2. Assign the pixel's cluster label to the cluster that provides the highest probability
  - Therefore, thresholds between clusters coincide with at points of equal probabilities
Finite Mixture Models

- Example: Suppose we have two classes, modeled by two Gaussians ($u_1=3$, $\sigma_1=1$, $u_2=5$, $\sigma_2=1$)
- What is the threshold between these two classes?
  - Set up distribution equations for each class
    
    \[ f_{Y_1}(x) = \exp\left[-\frac{(x - 3)^2}{2(1)^2}\right] \]
    
    \[ f_{Y_2}(x) = \exp\left[-\frac{(x - 5)^2}{2(1)^2}\right] \]
Finite Mixture Models

- Since threshold is at point of equal probabilities:

\[ f_{Y_1}(x) = f_{Y_2}(x) \]
\[ \exp\left[-\frac{(x - 3)^2}{2(1)^2}\right] = \exp\left[-\frac{(x - 5)^2}{2(1)^2}\right] \]
\[ -(x - 3)^2 = -(x - 5)^2 \]
\[ x^2 - 6x + 9 = x^2 - 10x + 25 \]
\[ 4x = 16 \]
\[ x = 4 \]
Histogram based Segmentation

• Advantages:
  – Efficient (usually requires only one pass for simple segmentation cases)

• Disadvantages:
  – Often difficult to determine proper peaks in the histogram
  – Difficult in situations where intensity is not sufficient to distinguish between two partitions (e.g., textured regions containing different mixes of intensities)
Clustering based Segmentation

- Create partitions by grouping pixels into clusters
- **Steps (for K-means clustering):**
  - Pick k pixels in the image to act as the initial centers of the k clusters
  - For each pixel, find the cluster that minimizes your distance metric
    - Distance metric can be the differences in:
      - Pixel intensity
      - Location
      - Variance
      - Weighted combination of these differences
Clustering based Segmentation

- **Steps:**
  - For each cluster, recalculate the cluster center based on the pixel locations within the cluster
  - Re-do all previous steps until convergence
- **Advantages:**
  - Good for segmentation where there are multiple distinct partitions
- **Disadvantages:**
  - Performs poorly when the regions have irregular shapes that mix into each other
Region Growing Segmentation

- Create partitions by continuously growing smaller regions until all pixels are accounted for.
- Steps:
  - Pick initial seeds that mark the individual regions who wish to segment
Region Growing Segmentation

- **Steps:**
  - For all neighboring pixels to the seed, compare the similarity between the seed and the pixel.
  - The pixel with the highest similarity is added to the seed to form a small region.
Region Growing Segmentation

• Steps:
  – For all neighboring pixels to the new regions, compare the similarity between the seed and the pixel
  – The pixel with the highest similarity is added to the small region to form a bigger region
  – Continue until all pixels in the image belong to a region
Region Growing Segmentation

- Advantages:
  - Good for segmentation with irregular shapes
- Disadvantages:
  - Slow
Active Contour Segmentation

- Create partitions by forming rough boundaries around regions/objects of interest and refining the boundaries until it matches the actual boundaries of the objects
- Steps:
  - Create rough boundary around the object

Source: http://www.markschulze.net/snakes/
Active Contour Segmentation

- Steps:
  - Calculate energy gradient between the current location of the boundary and its neighboring pixels
  - Expand or contract the boundary based on the gradient

Source: http://www.markschulze.net/snakes/
Active Contour Segmentation

- **Advantages:**
  - Good for segmenting and tracking objects with deformable motion and clear boundaries
  - Motion information can be extracted from the contour

- **Disadvantage:**
  - Relatively slow
  - Ill-suited for situations where your regions do not have clearly defined boundaries
Watershed Segmentation

- What is a watershed?
  - A ridge that divides different areas that different rivers and streams drain into
- What is a catchment basin?
  - An area that collects water within an area and drains into a body of water (e.g., sea, ocean, river, etc.)
- What does this have to do with segmentation?!
Watershed Segmentation

- Idea:
  - Let's treat bright areas as having high elevation and dark areas as having low elevation.
  - Let's treat boundaries between regions as watersheds and the local minimas of the regions as catchment basins.

Source: The MathWorks
Watershed Segmentation

• Idea:
  – Suppose we took a water hose and pour water into the catchment basins until they are full of water
  – Each of these filled catchment basins become the individual regions that you wish to segment!
  – Small regions can be merged into larger regions by continuing to pour water into the basins and overflow the smaller basins until they form a larger basin
Watershed Segmentation

- **Advantages:**
  - Efficient
  - Simple conceptually

- **Disadvantages:**
  - How much water do I pour?
    - Too little water leads to over-segmentation (too many remaining regions)
    - Too much water leads to under-segmentation (keep pouring water and the whole image becomes one big region)