Content-based Image and Video Retrieval

Vorlesung, SS 2010

Image Segmentation

3.05.2009
What is in this Image?
Introduction

- One of the key problems in computer vision
- Identification of homogenous region in the image
- Partition an image into meaningful regions with respect to a particular application
- The segmentation is based on measurements taken from the image and might be grey-level, colour, texture, depth or motion
Different Examples

- Search in image collections
  - Find representations that make sense to the user and is related to picture content
- Video summarization / shot boundary detection
  - Find similar frames, represent subsequences by key frame
- Finding people
  - Specific detectors, part-based detectors
- Finding buildings
- Finding machine parts
- Background subtraction
Motivation

- Before high-level reasoning on image, it can be break down into its major structural components
- Necessary for extracting reasonable local features (color, texture, etc.)
- Simplify or change image representation into more meaningful one for ease of analysis
Difficulties

- What is “correct” segmentation?
  - No single correct answer
  - Interpretation depends on prior world knowledge
  - World knowledge is difficult to represent
“Correct” Segmentation
“Correct” Segmentation

- Typical assumptions (Low level vision)
  - Intensity / color coherence
  - Texture coherence
  - Motion coherence
Image Segmentation

- Categories:
  - Pixel-based Segmentation
  - Region-based Segmentation
  - Edge-based Segmentation
  - (Graph-based Segmentation)
Pixel-based Segmentation

- Thresholding
- Clustering
Thresholding

- Determine the best threshold given a histogram of intensities
- Automatic thresholding
  - P-tile method
  - Mode method
  - Local adaptive method
  - Double threshold method
- Limitation of thresholding
  - Use *global* information
  - Ignore *spatial* relationships among pixels
Thresholding

- Determine the best threshold given a histogram of intensities
- Simplest way to segment an image: separate light and dark regions

\[ g(x, y) = \begin{cases} 1 & \text{if } f(x, y) > T \\ 0 & \text{otherwise} \end{cases} \]
**P-tile method**

- Use the a priori knowledge about the size of the object: assume an object with size $p$
- Choose the threshold such that $p\%$ of the overall histogram is determined

$\implies$ Obviously limited use
Mode-Method

- Find the peaks (modes) of the histogram and the local minimum between them
- Set threshold to the pixel value of the local minimum
- Not trivial to find peaks and local minimum on a noisy histogram
  - Ignore local peaks
  - Maximize “peakiness”
Adaptive-Method

- One single global threshold does not work for uneven illumination
- Local adaptive method
  - Divide an image into $mxm$ subimages and determine a threshold for each subimage
Double Thresholding-Method

- Also called hysteresis thresholding
- Starting from a conservative initial threshold $T_1$, determine the “core” parts of the object
- Continuing from this core part, grow this object by including neighboring pixels which are between $T_1$ and $T_2$
Clustering

- Process of partitioning a set of “patterns” into clusters – find subsets of points which are close together

- Cluster pixels based on
  - Intensity values
  - Color properties
  - Motion/optical flow properties
  - Texture measurements etc.

- Input: set of measurements
- Output: set of clusters and their centers $X_1, X_2, \ldots, X_m$
Simple Clustering Approaches

- Agglomerative Clustering (Merging)
  1. Make each point (pixel) a separate cluster
  2. Merge clusters with smallest inter-cluster distance until clustering is satisfactory

- Divisive Clustering (Splitting)
  1. Construct a single cluster using all points
  2. Split clusters with largest inter-cluster distance until clustering is satisfactory

- Difficulties:
  - Choice of inter-cluster distances
  - Stopping criterion (how many clusters are there?)
Segmentation by k-means

- Simple clustering methods use greedy approaches
- Alternative:
  - Formulate an objective function that should be optimized
  - Assuming that we know that there should be k-clusters, a good objective function would be

\[
\Phi(\text{clusters, data}) = \sum_{i \in \text{clusters}} \left\{ \sum_{j \in \text{ith cluster}} (x_j - c_i)^T (x_j - c_i) \right\}
\]

- Where \( x_j \) is a point coordinate, \( c_j \) is a cluster center
- If allocation of points to clusters where known, centers could be easily computed
  - But this is not the case
k-means algorithm

- Define iterative algorithm:
  - Assume the cluster centers are known and allocate each point to closest cluster
  - Assume allocation is known and choose new set of cluster centers. Each center is the mean of the points allocated to the that cluster

- Algorithm:
  - Choose initial mean values for $k$ regions
  - Classify $n$ pixels by assigning them to “closest” mean
  - Recompute the means as the average of samples in their (new) classes
  - Continue till there is no change in mean values
Color Clustering Examples

- Clustering in RGB space

Original images

Segmented images

9 clusters 5 clusters 4 clusters
Region-based Segmentation

- The main idea in region-based segmentation techniques is to identify different regions in an image that have similar features (gray level, colour, texture, etc.).
- There are two main region-based image segmentation techniques:
  - Region merging
  - Region splitting
Formulation of Region-based Segmentation

Let $R$ represent the entire image. Segmentation can be viewed as a process that partitions $R$ into $N$ disjoint regions, $R_1, R_2, ..., R_N$, such that:

a) $\sum_{i=1}^{N} R_i = R$

b) $R_i$ is a connected region, $i = 1, 2, ..., N$

c) $R_i \cap R_j = \emptyset$ for all $l$ and $j$, $l \neq j$,

d) $P(R_i) = TRUE$ for $i = 1, 2, ..., N$

e) $P(R_i \cup R_j) = FALSE$ for $i \neq j$

where $P(R_i)$ is a logical predicate over points in set $R_i$ and $\emptyset$ is the empty set.
Region Merging

- Merge two adjacent regions if they have “similar” properties according to some criterion.

- What does “similar” mean?
  - Examples:
    - “similar” average values: $|\mu_l - \mu_f| < T$
    - “small” spread of gray values: $|g_{max} - g_{min}| < T$
      - $g_{max} = \max \{g(x,y) | (x,y) \in R_l \cup R_f\}$
      - $g_{min} = \min \{g(x,y) | (x,y) \in R_l \cup R_f\}$
  - Note: non-transitive
    - A similar to B, and B similar to C does not imply that A is similar to C.
Region Merging

- Start with an initial segmentation
  - e.g. By thresholding
- Form the Region Adjacency Graph (RAG)
  - Regions are the nodes
  - Adjacency relations are the links
Region Merging

- For each region in the image do:
  - Consider its adjacent regions and test if they are similar
  - If they are similar, merge them and update the RAG
- Repeat the merging steps until there are no more merges.

![Diagram of region merging process](image-url)
Region Splitting

- Quad-tree decomposition:
  - Subdivide the entire image successively into smaller and smaller quadrant regions until for any region $R_i$, $P(R_i) = \text{TRUE}$

- The subdivision is represented with quad tree
The Quadtree Representation

- **Quadtrees:**
  - Trees where nodes have 4 children

- **Build quadtree:**
  - Nodes represent regions
  - Every time a region is split, its node gives birth to 4 children
  - Leaves are nodes for uniform regions

![Quadtree Diagram]

**Image segmentation and descriptors**
Region Splitting & Merging

- Splitting only results in adjacent regions with identical properties
- The final result can be obtained through merging the quadtree
  - Siblings that are “similar” can be merged