Mean Shift Segmentation

In order to use the mean shift algorithm for segmenting images, we should consider each pixel as a 3D point (row, column, intensity). Regions form clusters in this 3D space. To apply mean shift clustering, we can use a sphere instead of a circle. Let us consider a one-dimensional image (or a horizontal cut through a 2D image) to illustrate this.

Better results can be obtained by giving greater weight to points near the center of the sphere than to those further away from it. We can use Gaussian functions for computing a weighted mean. Also, we may want to give different amounts of tolerance to the spatial and intensity dimensions.
Shape Representation

Practical Considerations

- Large, homogeneous regions do not have a single point of convergence.
- In such cases, the mean shift clustering can be followed by region merging and split-and-merge.
- Unlike split-and-merge, mean shift does not always generate contiguous regions.
- Thus, after mean shift segmentation, we should remove all regions that are smaller than (i.e., contain fewer pixels than) a certain threshold $\theta$.
- These regions should be merged with their neighbor that is most similar to it in its average intensity.
- (This is also useful for split-and-merge.)

Examples

Input image (512x512 pixels)

Image after mean shift clustering ($h_1 = 16, h_2 = 8, \theta = 100$)

Region Representation

Depending on the purpose of our system, we may want to represent the regions that we detected in a specific way.

There are three different basic classes of representation:

- array representation,
- hierarchical representation, and
- symbolic representation.

We will now discuss these different approaches.

Array Representation

The simplest way to represent regions is to use a two-dimensional array of the same size as the original image.

Each entry indicates the label of the region to which the pixel belongs.

This technique is similar to the result of component labeling in binary images.

If we have overlapping regions, we can provide a mask for each region.

A mask is a binary array in which 1-pixels belong to the region in question and 0-pixels do not.

Hierarchical Representation

Hierarchical representation of images allows us to represent the visual information at multiple resolutions.

This is useful for a variety of tasks in computer vision. For example, the presence and number of certain objects can be more efficiently detected at low resolution.

Then object identification can be performed at high resolution.

We will look at two techniques: pyramids and quad trees.
Pyramids

A pyramid is a set of pixel arrays, each of which shows the same image at a different resolution. The level 0 image at the top of the pyramid consists of only 1 pixel. Each successive level has twice as many pixels in x and y direction as the level above it. Therefore, an $n \times n$ image is represented by levels 0 to $\log_2 n$. The following slide shows the pyramid representation of a 512×512 image.

Quad Trees

Quad trees are a method for hierarchically representing binary images. The root of a quad tree stands for the entire image, and its 4 children represent the 4 quadrants of the image. If a quadrant is completely white or completely black, the corresponding node is marked white or black, respectively. Otherwise, the node is expanded, and its children represent the quadrants within the original quadrant. This process is repeated recursively until all pixels are represented.

Symbolic Representation

For each region, represent information such as:
- enclosing rectangle
- moments
- compactness
- mean and variance of intensity
- etc.

In other words, symbolic representation of a region means that we describe its characteristics using a set of (typically numerical or categorical) parameters.

Data Structures

How can we represent regions in computer programs? We will look at two different approaches: Region adjacency graphs, picture trees, and super grids. Region adjacency graphs are just undirected graphs, where each vertex represents a region. The edges in the graph indicate which regions are adjacent to each other.
Region Adjacency Graphs

Picture Trees

Picture trees are a hierarchical structure for storing regions. The rule for representation is the "is-included-in" relationship:
All child regions are included in their parent region.

Super Grids

If we want to represent region boundaries in an array, it is unclear to which region the pixels on the boundary belong (left: image; center: boundary). If we expand our grid (right image), the boundary can be indicated without ambiguity.

Chain Codes

An interesting way of describing a contour is using chain codes. A chain code is an ordered list of local orientations (directions) of a contour. These local directions are given through the locations of neighboring pixels, so there are only eight different possibilities. We assign each of these directions a code, e.g.:

Then we start at the first edge in the list and go clockwise around the contour. We add the code for each edge to a list, which becomes our chain code.

What happens if in our chain code for a given contour we replace every code \( n \) with \((n \mod 8) + 1\)? The contour will be (approximately) rotated clockwise by 45 degrees.
We can also compute the derivative of a chain code, also referred to as difference code.
Difference codes are rotation-invariant descriptors of contours. Some features of regions, such as their corners or areas, can be directly computed from chain or difference codes.
Slope Representation

The slope representation of a contour (also called the Ψ-s plot) is like a chain code for continuous data. Along the contour, we plot the tangent Ψ versus arc length s.

Then horizontal line segments in the Ψ-s plot correspond to straight line segments in the contour. Straight line segments at other orientations in the Ψ-s plot correspond to circular arcs in the contour. Other segments in the Ψ-s plot correspond to other curve primitives in the contour.

Note that in this plot not the actual arc length is used, but its horizontal projection: