Region Segmentation

CSC 249/449
Sensory Motor Systems
Randal C. Nelson

Region Segmentation

Topics:

- Introduction
- Connected components in binary (discretely labeled) images
- Histogram-based labeling
- Pixel by pixel region growing.
- Splitting and merging techniques
- Watershed approaches
- Postprocessing
- Semantic labeling approaches

Introduction

Basic problem is to find contiguous groups of pixels that “belong” together. What we are generally after is semantically meaningful image regions that are stably extractable under expected variation.

- All pixels that compose (the projection of) a given object (probably not feasible bottom up).
- Stable parts of the same physical object.
- Areas composed of the same physical material (e.g. water, skin, grass, cornfields).
- Regions sharing some other property (e.g. shadow, common depth, or common motion).

Complements boundary-detection methods. Regions can be converted to boundaries, and closed boundaries to regions, but algorithms take different general approaches. For boundaries, we are looking for places where “things change” For regions, we are looking for “things that are the same”

Division is not absolute, of course. “Boundary” evidence is used in generation of regions, and “region” evidence in the generation of boundaries.

Range of representations not as broad as for boundaries - generally collections of pixels. A few other possibilities exist, but are generally produced secondarily, and are not useful as part of a growth process.

- Collection of (connected) pixels
- Quadtree representation
- Closed contours
- Skeleton (distance transform)

Range of prior information is also not as varied

- High-level information about scene content (e.g. output of an object recognizer, with locations, extents, and model information.
- Approximate location (size, shape) of regions
- Previously extracted, but incomplete extended boundaries
- Local "edgel" images
- Statistical measures of within-class variance (textures)

Connected Components

Start with binary (or discretly value) image, and produce a list of connected sets of same-valued pixels. This is not really a segmentation algorithm, since the image pixels of the image have already been classified, but it is sometimes applied to produce connected sets after some other process has classified pixels on the basis of local information.

- Two pixels are connected if they have the same label, and are (4 or 8) neighbors in the image plane.
- Connected sets can be represented compactly, though not very usefully, as a labelled image, with one label for each connected set.
- Connected components are more frequently represented by more complex data structures containing a list of component pixels, and frequently auxiliary descriptive information such as area, centroid, bounding box, etc.

Row-at-a-time (actually 2-row-at-a-time) algorithms were formerly important, as images were too large to fit in memory. Still useful for very large images, or when every bit of speed is needed.

- Label propagation step: scan image left to right, top to bottom, and propagate labels down and to the right, and issue binary equivalences.
- Use general transitive closure graph algorithm (union-find) to propagate binary equivalences and produce equivalence classes of labels.

Common current practice is to write simpler “flood fill” algorithms that systematically explore neighbors directly in the image plane to produce components.

- These are reasonably efficient when entire image can be placed in memory, and can be written to interact well with the cache.
- Natural algorithm structure is recursive, but recursion should not be used explicitly as the call stack can get very deep (on the order of the number of pixels in the image).

Digital topology consistency issues are complicated if there are more than two values in the image (and even with binary, one value must be 4-connected and the other 8-connected). In most applications, this sort of topological consistency is not a requirement.

Histogram-Based Segmentation

Probably the simplest method of attempting to classify pixels into groups. Basic idea is to produce a histogram of pixel values, either in monchrome, or multi-dimensional color space, and then find peaks in the histogram, and use them to suggest likely classes.

- In monochrome images valleys between peaks can be used as segmentation “thresholds”
- In multi-dimensional images, a “valley” is not something that is described by a single number, so classes are usually associated more directly with neighborhoods of histogram peaks.
- Multi-dimensional regions can be simple spheres defined using a Euclidian metric; rectangular, axis-aligned boxes; axis-aligned ellipsoids defined by weighting dimensions differently under a Euclidean metric; or more general ellipsoids defined by a Mahalanobis metric.

The Mahalanobis Distance

The Mahalanobis norm can be viewed as multiplying a vector by a positive definite matrix $\mathbf{A}$ before taking the Euclidean norm. The effect of this is equivalent to performing a non-uniform scaling along the axes of a rotated orthogonal coordinate system. The surfaces of constant distance are n-dimensional ellipsoids.

The matrix $\mathbf{A}$ can be related to the covariance matrix $\mathbf{S}$ of a distribution in the same way that the standard deviation can be related to the variance in a one-dimensional distribution: namely $\mathbf{A} = \sqrt{\mathbf{S}}$. This in itself is not sufficient, since matrices in general can have many square roots. However if $\mathbf{S}$ is a covariance matrix it is positive definite, and there is only one square root that is also positive definite. That is the one we are after.

This square root can be computed by summing up the outer products of the normalized eigenvectors each multiplied by the square root of the corresponding eigenvalue. In practice, the square root matrix does not have to be computed explicitly, and the squared norm of a vector $\mathbf{x}$ is computed as $\mathbf{x}^T\mathbf{S}\mathbf{x}$ This is because if $\mathbf{A}$ is symmetric and $\mathbf{A}^2 = \mathbf{S}$, then

$$Mnorm^2(\mathbf{x}) = (\mathbf{A}\mathbf{x}) \cdot (\mathbf{A}\mathbf{x}) = (\mathbf{A}\mathbf{x})^T(\mathbf{A}\mathbf{x}) = \mathbf{x}^T\mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{x}^T\mathbf{S}\mathbf{x}$$

Pixel by pixel region growing

Basically a modification of the discrete flood fill algorithm to handle images that are not discretely labelled.

- Start with a seed region (e.g. small, nearly uniform neighborhoods)
- Add neighboring pixels as long as they meet some membership criterion (e.g. they are close in value to neighboring pixels already in the region, and not too far from the starting value)
- Method can "leak" through small breaks in an otherwise clear boundary.
- Controlling leaks with a maximum variation threshold can produce "artificial" boundaries.
- Problems are due to local character of algorithm.
- More complex control strategies can sometimes improve performance. For example, changing the allowed deviation from the starting value only if some connected percentage of the current boundary can add nearby neighbors if the standard is relaxed. This starts to resemble boundary-based methods.

Splitting and merging techniques

Idea is that instead of adding a pixel at a time, we consider larger regions, that may be divided if they appear to contain two populations, or combined if the merged region seems uniform. Generally only adjacent regions are considered candidates for merging.

- An important concept is a *homogeneity condition.* This is some function of a (usually connected) set of pixels that conceptually evaluates to true if the region is “uniform” and thus should not be divided, and evaluates to false otherwise. An example might be some histogram bimodality test, or even something as simple as a threshold on the variance.
- A second important concept is that of boundary strength. This applies to a linear structure, generally representing a possible dividing line, and attempts to evaluate the evidence that the linear structure corresponds with a region boundary along part or most of its length. Simple examples are summed gradient magnitude, or fraction of pixels with magnitude above a threshold and direction aligned with purported boundary.
- Algorithms in this general group have a great deal of variation. There are pure merging techniques that start with small regions and attempt to merge them recursively. Others start with large areas and try to split them until uniform regions are obtained. Still others use both splitting and merging operations in alternating fashion, or in conjuction with a multiple scale approach.
- Methods of this type can become arbitrarily complex; they can also be made quite sophisticated mathematically . A number fairly successful approaches have been based on attempts to create notions of homogeneity, or boundary evaluation that are statistically well founded, or even based on physical models of the image generation process.

Watershed approaches

Idea is that the image can be divided into "basins" separated by "hills" or "mountain ranges" of high gradient magnitude. At the lowest level, every local minimum has an associated attraction neighborhood within which all points can reach the minimum along a path that goes only downhill.

- Definition is attractive because it is mathematically clean, and has certain nice properties.
- In practice, the initial regions tend to be extremely small, necessitating some grouping or multi-resolution process in order to obtain the desired regions.
- Regions resulting from such processes can be somewhat unstable, with large changes in shape resulting from small changes in the underlying pixel values.

Postprocessing

Region segmentation algorithms, especially low-level ones based on thresholds or simple growth processes, often produce small, isolated regions, small holes in large regions, and ragged boundaries. If such artifacts are a problem, they can sometimes be eliminated through simple post-processing techniques.

- Morphological operations such as erosion and dilation, or the derived operations of uniform opening and closing are particularly helpful in this respect.
- An opening operation (erosion followed by dilation), will eliminate small, isolated regions, while closing (dilation followed by erosion) will eliminate small holes.
- More complex methods looking at region shape or color characteristics, often in the presence of application-specific information are frequently employed. Such methods often blur the line between where a general segmentation algorithm ends and a specific application begins.

Semantic labeling approaches

Basic idea is that semantic labels such as “sky” “water”, “foliage”, or “pavement” can be associated with probabilistic color (or texture) measurements, and adjacency information. These prior probabilities can be derived from a hand segmented set of training images. The problem of segmenting an unknown image into labeled regions can then be cast as an optimization problem that attempts to find the labeling with maximum posterior probability (MPP techniques).

- Solving the associated optimization problem exactly is generally intractable, so various approximation schemes are employed, including relaxation methods, and simulated annealing.
- In domains that have only a few classes and for which adequate training data is available (e.g. land-use categorization from multi-spectral satellite photos), this approach can be quite effective.
- Effectively a problem-specific method, and not very useful for processing a broad range of images.