RIM. Reconocimiento de Imágenes
Image Recognition
Image Segmentation and Graphs

Francesc J. Ferri

Dept. d'Informàtica.
Universitat de València

november 2009

Image Segmentation

Image segmentation deals with separating a given image into its constituent parts.

This implies understanding the image to some extent.

The problem in its full generality is by far too difficult. Many different restrictions or particularizations need to be applied.

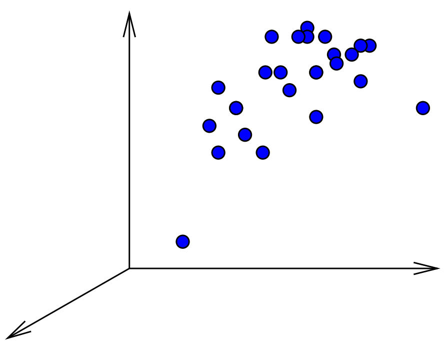
Image segmentation approaches

- Greedy and local approaches (split & merge and variants)
- Global criterion-based approaches
  - Markov Random Fields
  - Variational approaches
- Classification-based approaches
  - (Extended) Pixel classification
  - Clustering-based approaches
  - Semi-supervised methods
- **Graph-based methods**

Basic cluster-based segmentation

Local representations

If an appropriate description, $\mathbf{x}$, is assigned to each image location, (x, y), then the whole image can be seen as a cloud of points in a representation space.



Close points in this space should represent points corresponding to the same regions. So, **grouping** will lead to segmentation.

Problems with cluster-based segmentation

A good/appropriate local description is needed.

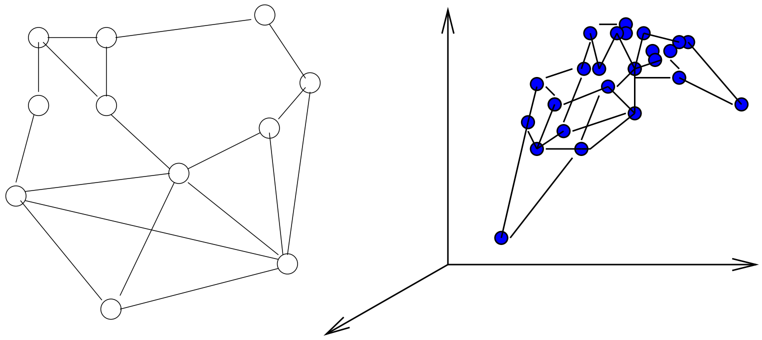
Some kind of spatial coherence/homogeneity needs to be imposed.

Lots of different kinds of extensions exists: postprocessing/filtering, extended representation, contextual classification, dual representations, co-clustering, etc.

Introduction

The graph-based approach

Represent the image and its contents as a graph

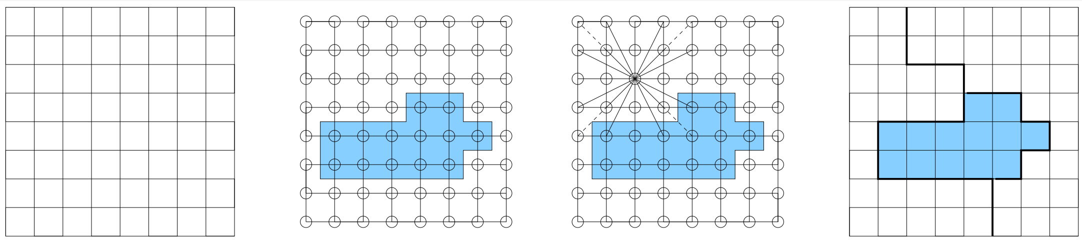


Nodes are image points or locations in a given representation space while edges represent similarities between nodes

image segmentation ⟶ graph partitioning

Graph representation of images

In its finest interpretation, pixels are nodes and edge weights represent pixel similarity or "affinity" with regard to belonging to the same region in terms of spatial and representation proximity.



Some measure of global affinity or homogeneity intra and inter regions (subgraphs) needs to be defined in order to define an appropriate partition criterion.

Graph clustering

Let $G = (V, E, w)$ a weighted undirected graph, $w : E \longrightarrow \mathbb{R}$.

$w(i, j)$ is a measure of similarity or proximity between nodes i and j representing two pixels in the image.

Let us consider bipartitions only. A (connected) graph is partitioned into two disjoint subgraphs A and B ($A \cup B = V$ and $A \cap B = \emptyset$) by removing from E any edges connecting nodes from A to B

$$\{e = (i, j) \in E : i \in A, j \in B\}$$

The total weight of such edges are referred to as the **cut**.

$$cut(A, B) = \sum_{i \in A, j \in B} w(i, j)$$

Minimum cut

Minimum cut

A good bipartition is the one that minimizes the cut. This is a well-studied problem in graph theory but tends to give too small subgraphs in the kind of graphs that come from images.

To measure the relative importance of the cut with regard to the subgraphs it produces the following measure is defined for any $A \subseteq V$:

$$assoc(A, V) = \sum_{i \in A, j \in V} w(i, j)$$

Normalized cuts

The normalized cut

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)}$$

The normalized association

$$Nassoc(A, B) = \frac{assoc(A, A)}{assoc(A, V)} + \frac{assoc(B, B)}{assoc(B, V)}$$

It is straightforward to show that $Ncut(A, B) = 2 - Nassoc(A, B)$.

The bipartition that minimizes Ncut (or maximizes Nassoc) gives maximum association and minimum disassociation between subgraphs.

Minimum normalized cut problem

The discrete minimum normalized cut problem is NP-hard but it can be efficiently solved in the continuous case.

$$min_y \frac{y^T(D-W)y}{y^T D y}$$

$$\text{subject to: } y^T D \mathbf{1} = 0$$

W : weight matrix, $W(i,j) = w(i,j)$.
D : (diagonal) degree matrix, $D(i,i) = \sum_j w(i,j)$.
$\mathbf{1}$: column vector of ones.
$L = D - W$: **laplacian matrix** (positive semidefinite)
y : indicator (bivalued) vector that represents the partition of V.
$y_i \in \{1, -b\}$

Eigensystem

The optimal partition is then the solution to the eigensystems:

$$Ly = \lambda D y$$

$$D^{-\frac{1}{2}} L D^{-\frac{1}{2}} z = \lambda z \text{ where } z = D^{\frac{1}{2}} y$$

$$D^{-1} L y = \lambda y$$

It can be shown that the first smallest eigenvector (whose eigenvalue is 0) is $\mathbf{1}$, and the second smallest one, y_2 is the (real-valued) solution to the minimum normalized cut problem.

Spectral clustering algorithm

Given $G = (V, E, W)$

1. Compute the second smallest eigenvector, y_2 of $D^{-1}(D - W)$.
2. Find an appropriate splitting value for y_2
3. Recursively partition each block until some appropriate stop condition

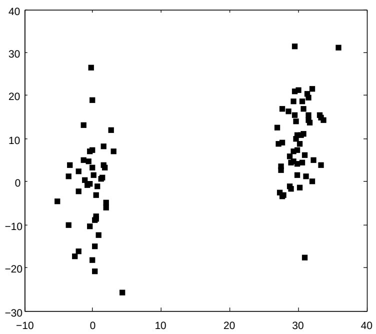
3'. Find appropriate splitting values for eigenvectors $y_3, y_4, \ldots$ until some appropriate stop condition.

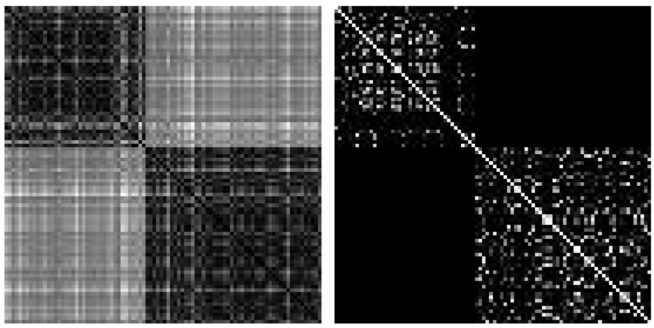
Example: data clustering

Consider points in a vector space and the Euclidean distance, d_{ij}.

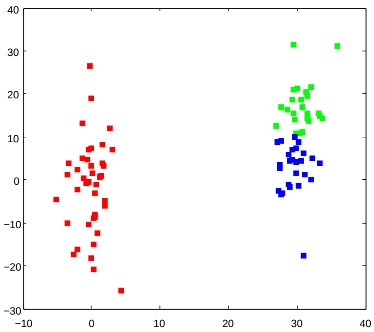
An appropriate **affinity** can be defined as

$$w_{ij} = e^{-\frac{d_{ij}}{\sigma}^2}$$





Clustering result



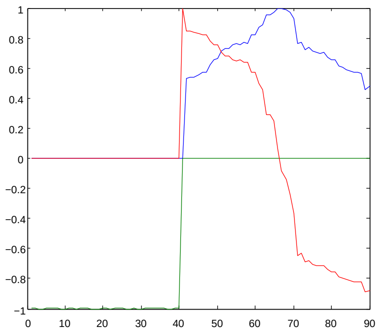


Image segmentation

Affinity matrix

is the matrix formed with the weights assigned to each pair of pixels (nodes).

The matrix must be sparse (few edges).

Edges must take into account both spatial proximity and representation similarity.

Affinity matrix for images

$$w_{ij} = e^{-\frac{{d_{ij}^F}^2}{\sigma_F}} \cdot \begin{cases} e^{-\frac{{d_{ij}^X}^2}{\sigma_X}} & \text{if } d_{ij}^X < r \\ 0 & \text{otherwise} \end{cases}$$

Example: image segmentation

Image segmentation: affinity design

Affinity matrices can be defined from a number of cues: color, texture, brightness, edges, motion, etc.

Many different possibilities exist to aggregate information and to combine it.

Advanced methods for image description and characterization, can be integrated into the approach.

Other (spectral) graph clustering algorithms

Other possibilities for analyzing and partition graphs exist, all based on the Laplacian matrix.

Alternatives to normalized cut

Cheeger conductance
ratio cut (cut divided by min of assocs)
average cut