Image Segmentation

DD2423 Image Analysis and Computer Vision

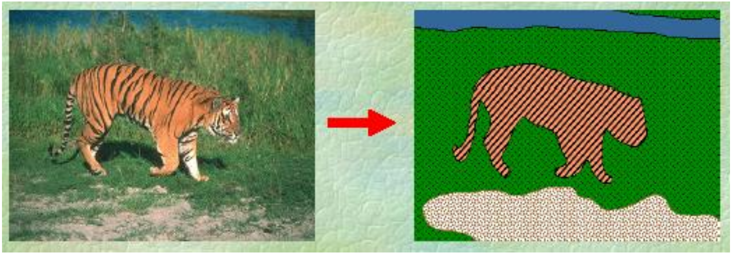
Mårten Björkman

Robotics, Perception and Learning Division
School of Electrical Engineering and Computer Science

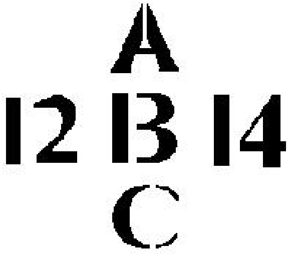
November 24, 2021

From images to objects

- Image segmentation: Dividing images into semantically meaningful regions.
- Reliable segmentation is possible with prior information, but such information is often not available.



Segmentation of usually ambigious



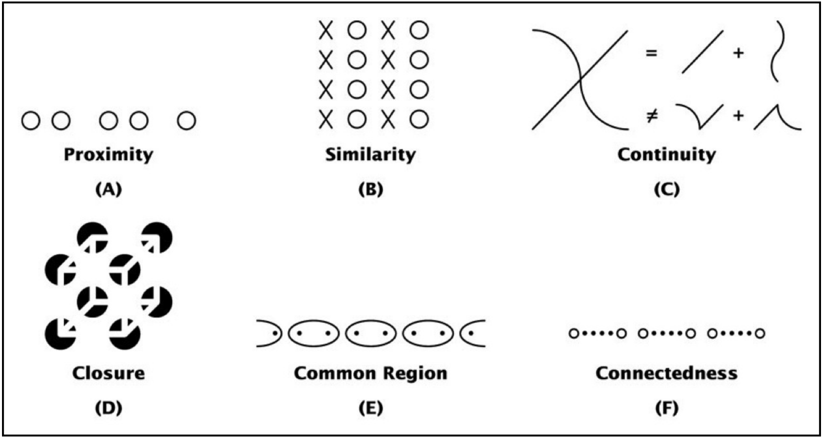
Spot the 'X'

OBCPGDOPDDGQQPCPOCG
PRGPOCBGQRQSSUOPCSR
QCDBPOSCUROOPCDBPOD
POQXGOPQCBBCGPOQDUO
OPQDCBGSOSPQSRCBDOP

Spot the 'X'

KLEFIZKNMLMVKWIYLKMNI
IKLWNMVKAILKHNMVTEFNL
MKLNVKWAVNMKLIYZFENM
NMKLNHKVEYIFKLXNVIWTY
KVNMKLIYWTNMILKMFWEN

Segmentation in humans (Gestalt Theory)



What is this?

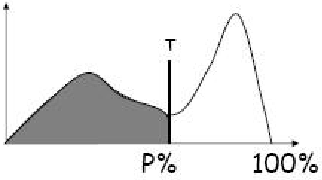
Segmentation techniques

- Figure-ground segmentation: divide image in foreground and background regions. Methods:
  - Thresholding *
  - Level-set methods
  - Energy minimization with graphs
  - Instance segmentation with CNNs
- Image segmentation: divide image in regions with pixels of similar qualities. Methods:
  - K-means clustering *
  - Watershedding *
  - Mean-shift segmentation *
  - Normalized cuts *
  - Semantic segmentation
- Methods mentioned today are used for grouping of data, not just image data, but nowadays most of them are rarely used directly for image segmentation.

Histogram based segmentation

- Threshold the grey-level values to create a binary image.
- How to automatically find a good threshold?
- Common problems: measurement noise, non-uniform illumination, non-constant intensity of objects, unknown size of objects, ...

1. P-tile method
   - Use the a priori knowledge about the size of the object: assume an object with size P as fraction of the whole.
   - Choose a threshold such that P% of the overall histogram is determined.



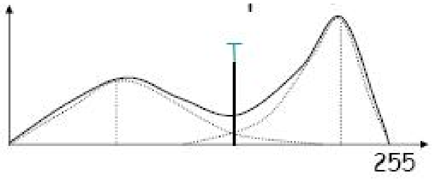
Rather limited use... but can be used as starting point.

2. Mode method

- Find the peaks and valleys of the histogram.
- Set threshold to the pixel value of the valley.
- Non-trivial to find peaks/valleys:

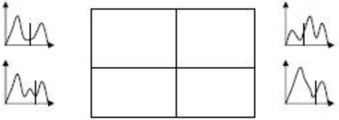
  Ignore small peaks, find largest peaks, find valley between these peaks.

  Maximize 'peakness' (difference between peaks and valleys) to find the threshold as valley.



Automatic thresholding

3. Iterative thresholding
   - Start with an approximate threshold and refine it iteratively, taking into account some goodness measure e.g. $T = (r_1 + r_2)/2$ where r_i is the mean gray value of previous segmented region i.
4. Adaptive thresholding
   - In case of uneven illumination, global threshold has no use.
   - One approach is to divide an image into $m \times m$ subimages and determine a threshold for each subimage.
   - Extension: fit local thresholds to a smooth function.



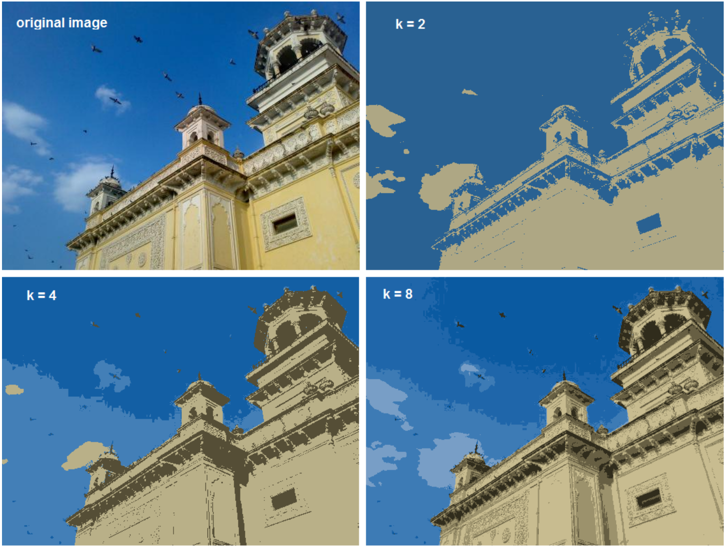
K-means clustering

- Group pixels based on similarity in colours (or any other measure).
- K-means (or ISODATA) algorithm:
  1) Choose K initial mean values (possibly randomly).
  2) Assign each pixel to the mean that is closest.
  3) Update means as the average of pixels assigned to each mean.
  4) Iterate until there is no change in mean values.
- Problem: segments can be splitted up into pieces.

K-means clustering

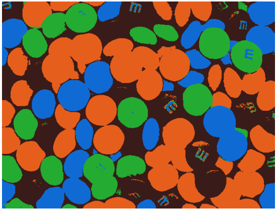
Two iterative steps: 1) associate points to closest cluster centers, 2) recompute cluster centers as mean of points associated to them.

K-means clustering



Original

4 clusters



8 clusters

Hard to know how many clusters you should have for best result.

Gaussian mixture models (GMM)

- Assume that a pixel has a combination of colours from multiple clusters, instead of just coming from one particular cluster.
- Then the colour distribution of pixel i might be written as

$$P(c_i) = \sum_k P(z_i = k) P(c_i | z_i = k) = \sum_k \pi_k \, \mathcal{N}(c_i; \mu_k, \Sigma_k),$$

  where $\pi_k = P(z_i = k)$ is the probability that pixel i belongs to cluster k (assumed the same for all pixels) and $\mathcal{N}(c_i; \mu_k, \Sigma_k)$ is a Gaussian distribution with mean μ_k and variance Σ_k.
- Goal: Find the maximum likelihood estimate of model parameters.

Gaussian mixture models (GMM)

Model parameters can be found with Expectation-Maximization.

1. Expectation step (update membership probabilities):

$$T_{i,k} \leftarrow P(z_i = k | c_i, \{\pi_k, \mu_k, \Sigma_k\}) = \frac{\pi_k\, \mathcal{N}(c_i; \mu_k, \Sigma_k)}{\sum_j \pi_j\, \mathcal{N}(c_i; \mu_j, \Sigma_j)}$$

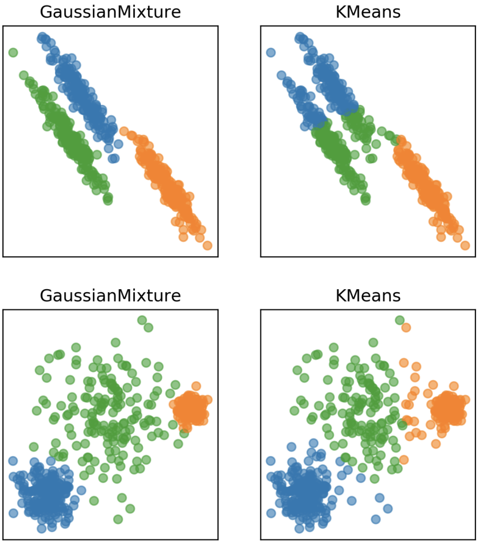
2. Maximization step (update model parameters):

$$\pi_k \leftarrow \frac{1}{N} \sum_i^N T_{i,k}, \quad \mu_k \leftarrow \frac{\sum_i^N T_{i,k} c_i}{\sum_i^N T_{i,k}}$$

$$\Sigma_k \leftarrow \frac{\sum_i^N T_{i,k} (c_i - \mu_k)(c_i - \mu_k)^T}{\sum_i^N T_{i,k}}$$

3. Iterate until convergence.

Gaussian mixture models (GMM)



K-means assumes round clusters, GMM allows them to be elliptic.

Spatial coherence

- Methods mentioned so far neglect the dependency between neighboring pixels.
- Neglecting dependency may cause segments to be splitted up into different pieces.
- If spatial coherence between pixels is taken into account, this can be avoided.
- Dependency between neighboring pixels or regions could be represented in various ways.

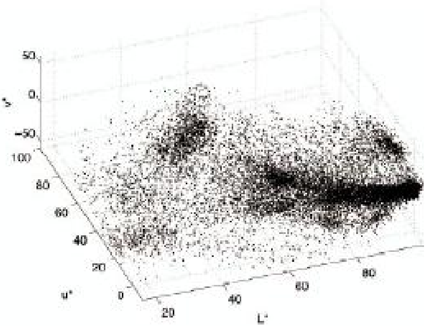
Mean-shift segmentation

- Mean shift is a common method for kernel density estimation.
- Group pixels both in terms of colours and positions.

$$\underset{\textit{each pixel}}{\Longrightarrow} \begin{bmatrix} x \\ y \\ R \\ G \\ B \end{bmatrix}$$

- Note: You could use (x, y) also for K-means and GMM. However, for Mean-shift it is more common to do so.

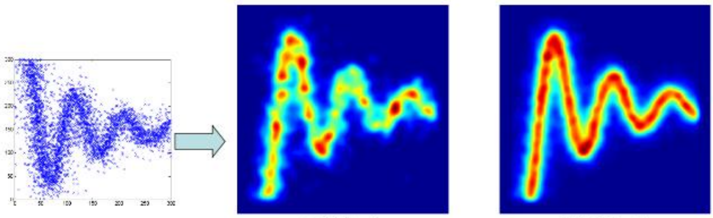
Mean-shift segmentation



Example distribution of colour values in Luv space
(L=luminance, uv=color)

Mean-shift segmentation

- Mean-shift tries to find points (in 5D space) with maximum density.
- Problem: We just have a bunch of samples.
- However, each sample can be assumed to be noisy.
- Solution: We place a small "ball" $K(x - x_i)$ around each sample, with maximum density in the center x_i.



(a) 2000 Samples (b) 20000 Samples

Formally: an iterative scheme

We want to find maxima of the total density function

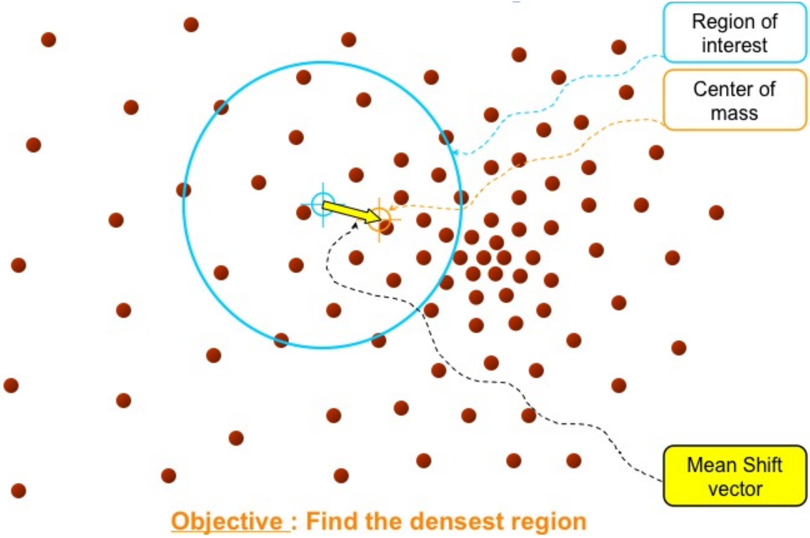
$$f(x) = \frac{1}{n}\sum_{i=1}^{n} K(x - x_i) \quad \text{where} \quad K(x) = C\, k(\|x\|^2)$$

Then the gradient should be

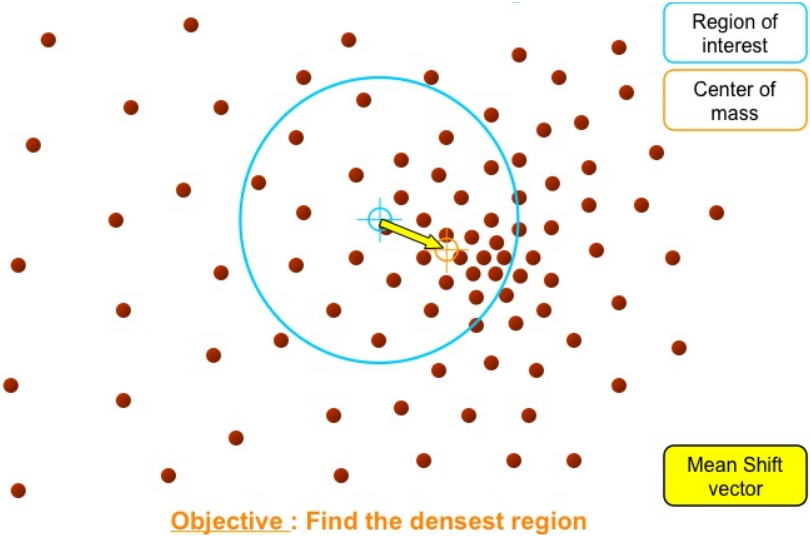
$$\nabla f(x) = \frac{2C}{n}\sum_{i=1}^{n}(x - x_i)k'(\|x - x_i\|^2) = 0 \;\Rightarrow\; x^{new} = \frac{\sum_{i=1}^{n} x_i\, k'(\|x - x_i\|^2)}{\sum_{i=1}^{n} k'(\|x - x_i\|^2)}$$

Most common kernel ('ball'): Gaussian kernel, for which the derivative also is a Gaussian.

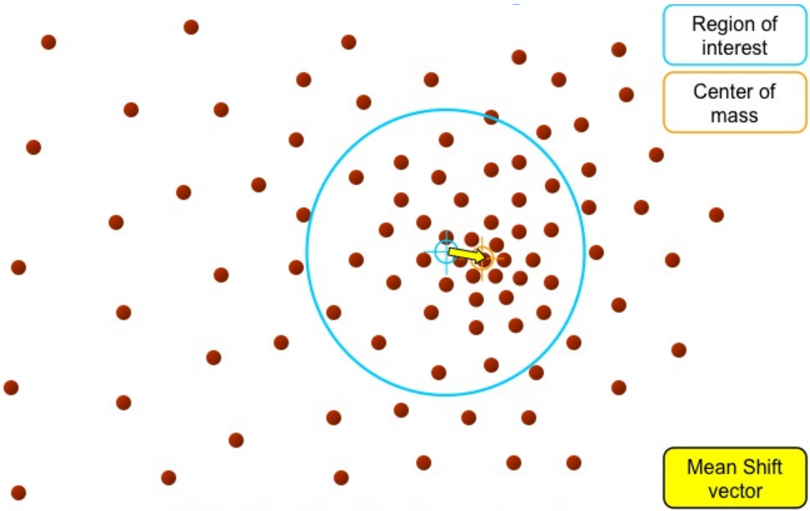
Mean-shift segmentation



Mean-shift segmentation

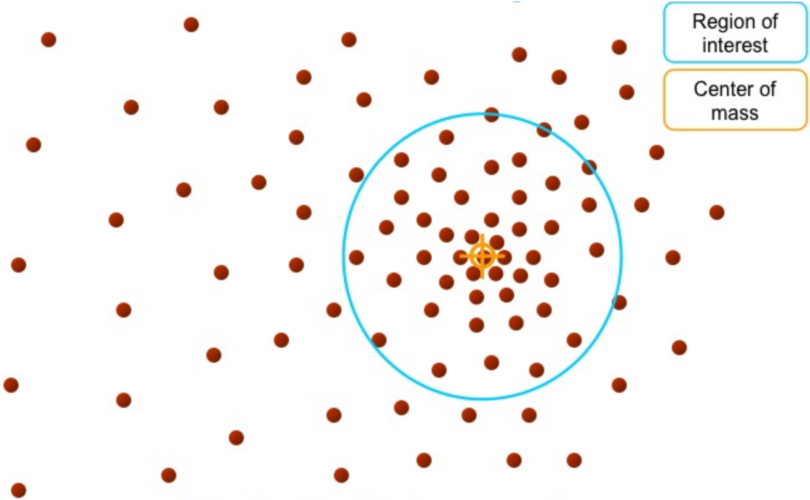


Mean-shift segmentation



Objective : Find the densest region

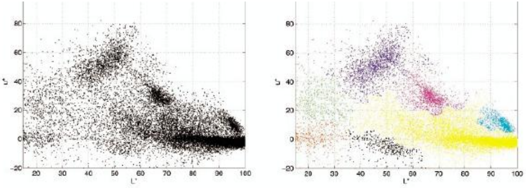
Mean-shift segmentation



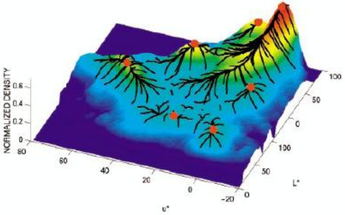
Objective : Find the densest region

Mean-shift segmentation

- The density function will have peaks (also called modes).
- If we start at each pixel and do gradient-ascent, we will converge to one of these modes.
- Then cluster the image based on the modes pixels converged to.

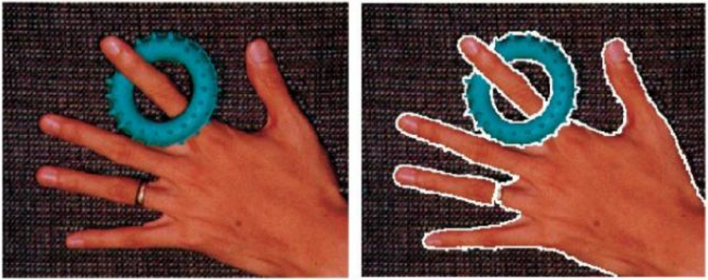


(a) (b)



Mean-shift segmentation example

- Get a segmentation by starting with the 5D value of each pixel, iterate and see which peak (mode) you end up with.



(a) (b)

Merging and splitting

The output of many segmentation methods can be improved by simply merging similar neighboring regions. Often you intentionally start with an oversegmentation and then merge regions.

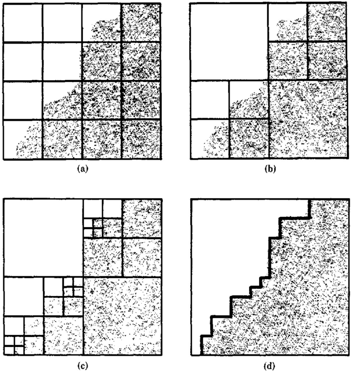
Region similarity can be measured by

- Comparing their mean intensities. Check against a predetermined threshold.
- Comparing their statistical distributions. Check whether a merge would represent 'observed' values better.
- Checking 'weakness' of the common boundary. Weak boundary: intensities on two sides differ less than a threshold.

Merging and Splitting

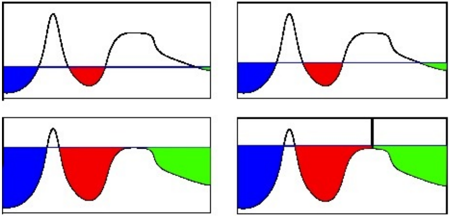
A split-and-merge algorithm can be very simple.

- First hierarchically split image regions until the variation in each region are small enough.
- Then merge neighbours as long as variations remain small.



Watershedding

- Create some topological map over image domain (using gradient magnitude, distance transform or similar).
- Gradually fill basins with 'water' from the deepest points upwards.
- When two basins meet, create an edge between two segments.
- End when all pixels are either filled or edge pixels.

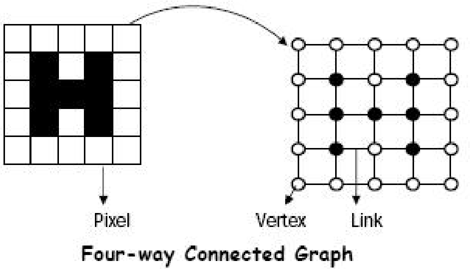


Watershedding

- Usually leads to over-segmentation, unless relevant image regions already have a close to uniform colour.
- However, efficient way to create superpixels (groups of similar pixels) that can be grouped using e.g. merging.

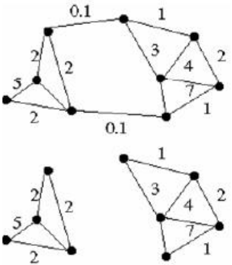
Graph theory in image segmentation

- Graph theory can be used to analyse and segment images.
- Each pixel (or superpixel) in the image corresponds to a node.
- Neighbouring pixels are connected by links.
- Nodes and links have weights.
  - Node weights are often based on the pixel colours, and
  - Link weights on similarities between neighbouring pixels.



Graph theoretic clustering

- In graph-theoretic clustering, the links in the graph represents similarities between the nodes.
- These can be summarized as a large affinity (similarity) matrix W.
- Problem: Split the graph in two (or more) pieces so that the cutted links have as low weights as possible.



Measuring affinity (similarity)

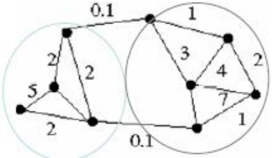
- The affinity matrix can for example have the following elements:
  - Intensity $W(x,y) = \exp\left\{-\|I(x) - I(y)\|^2/(2\sigma^2)\right\}$
  - Position $W(x,y) = \exp\left\{-\|x - y\|^2/(2\sigma^2)\right\}$
  - Colour $W(x,y) = \exp\left\{-\|c(x) - c(y)\|^2/(2\sigma^2)\right\}$
- Here σ represents a scale factor that can be thought of how different two pixels can be and we still regard them as similar.
- The best methods combine many possible similarity measures.
- Nowadays, common to use features learned with deep networks.

Normalized cuts

- Goal: Maximize sum of within cluster similarities, while minimizing sum of across cluster similarities.
- Minimize Normalized Cut

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

  - A and B are two disjoint sets of vertices and $V = A \cup B$.
  - $cut(A,B)$ – sum of links between vertices in A and B.
  - $assoc(A,V)$ – sum of links connected to any vertex in A.
- Segmentation found by solving a generalized eigenvalue problem.



Normalized cuts

- Let W be affinity matrix and D diagonal matrix with $D_{ii} = \sum_j W_{ij}$.
- Minimizing $Ncut(A, B)$ is equivalent to solving

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

where elements in y indicate whether nodes belong to A or B.

- Equivalent to solving generalized eigenvalue problem

$$(D - W)y = \lambda Dy$$

or after normalization

$$(I - D^{-1/2} W D^{-1/2})z = \lambda z, \quad \text{where } z = D^{1/2} y.$$

Normalized cuts

Summary of good questions

- What is the purpose of image segmentation?
- What is Gestalt Theory?
- How to select a threshold for histogram based segmentation?
- How does K-means work?
- Why is spatial coherence important for segmentation?
- What does a mean-shift algorithm try to do and how?
- How does split-and-merge work?
- What is an affinity matrix?
- What does Normalized Cuts try to optimize?

Readings

- Gonzalez and Woods: Chapter 10.3-10.7
- Szeliski: Chapter 5.2.1-5.2.2 and 7.5