EM Algorithm and its Applications

Yi Li
Department of Computer Science and Engineering
University of Washington

Outline

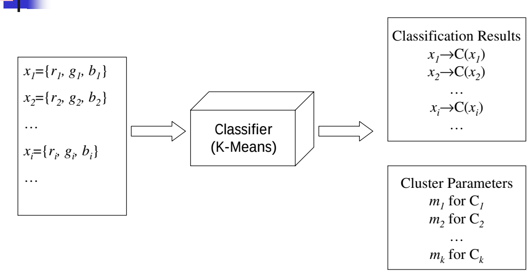
- Introduction of EM
  K-Means → EM
- EM Applications
  - Image Segmentation using EM
  - Object Class Recognition in CBIR

Color Clustering by K-means Algorithm

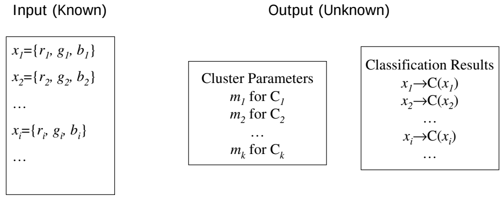
Form K-means clusters from a set of n-dimensional vectors

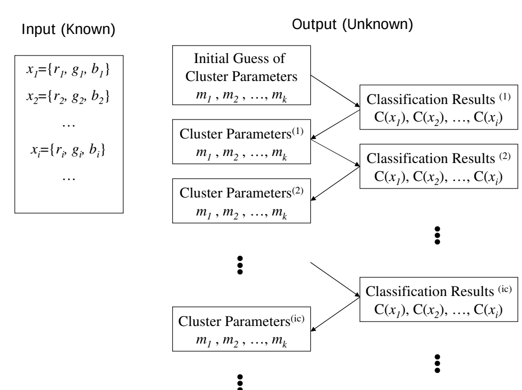
1. Set ic (iteration count) to 1
2. Choose randomly a set of K means $m_1(1), \ldots, m_K(1)$.
3. For each vector x_i, compute $D(x_i, m_k(ic))$, $k=1,\ldots K$ and assign x_i to the cluster C_j with nearest mean.
4. Increment ic by 1, update the means to get $m_1(ic),\ldots,m_K(ic)$.
5. Repeat steps 3 and 4 until $C_k(ic) = C_k(ic+1)$ for all k.

K-Means Classifier



K-Means Classifier (Cont.)

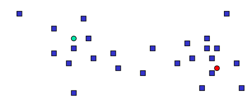




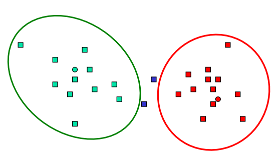
K-Means (Cont.)

- Boot Step:
  - Initialize K clusters: $C_1, \ldots, C_K$
    Each Cluster is represented by its mean m_j
- Iteration Step:
  - Estimate the cluster of each data
    $x_i \Longrightarrow C(x_i)$
  - Re-estimate the cluster parameters
    $m_j = mean\{x_i \mid x_i \in C_j\}$

K-Means Example



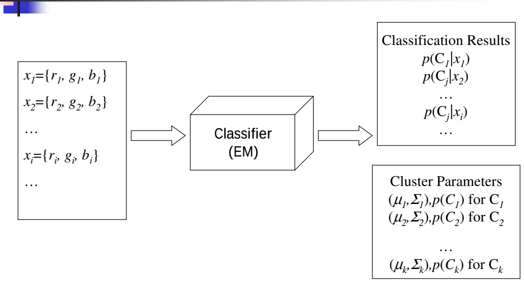
K-Means Example



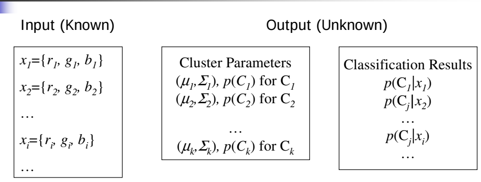
K-Means → EM

- Boot Step:
  - Initialize K clusters: $C_1, \ldots, C_K$
    (μ_j, Σ_j) and $P(C_j)$ for each cluster j.
- Iteration Step:
  - Estimate the cluster of each data ➡ Expectation
    $p(C_j \mid x_i)$
  - Re-estimate the cluster parameters ➡ Maximization
    $(\mu_j, \Sigma_j), p(C_j)$ For each cluster j

EM Classifier



EM Classifier (Cont.)



Expectation Step

Input (Known)		Input (Estimation)		Output
$x_1=\{r_1, g_1, b_1\}$ $x_2=\{r_2, g_2, b_2\}$ … $x_i=\{r_i, g_i, b_i\}$ …	+	Cluster Parameters (μ_1,Σ_1), $p(C_1)$ for C_1 (μ_2,Σ_2), $p(C_2)$ for C_2 … (μ_k,Σ_k), $p(C_k)$ for C_k	⇨	Classification Results $p(C_1\|x_1)$ $p(C_j\|x_2)$ … $p(C_j\|x_i)$ …

$$p(C_j | x_i) = \frac{p(x_i | C_j)\cdot p(C_j)}{p(x_i)} = \frac{p(x_i | C_j)\cdot p(C_j)}{\sum_j p(x_i | C_j)\cdot p(C_j)}$$

Maximization Step

Input (Known)		Input (Estimation)		Output
$x_1=\{r_1, g_1, b_1\}$ $x_2=\{r_2, g_2, b_2\}$ … $x_i=\{r_i, g_i, b_i\}$ …	+	Classification Results $p(C_1\|x_1)$ $p(C_j\|x_2)$ … $p(C_j\|x_i)$ …	⇨	Cluster Parameters (μ_1,Σ_1), $p(C_1)$ for C_1 (μ_2,Σ_2), $p(C_2)$ for C_2 … (μ_k,Σ_k), $p(C_k)$ for C_k

$$\mu_j = \frac{\sum_i p(C_j | x_i)\cdot x_i}{\sum_i p(C_j | x_i)} \quad \Sigma_j = \frac{\sum_i p(C_j | x_i)\cdot (x_i - \mu_j)\cdot (x_i - \mu_j)^T}{\sum_i p(C_j | x_i)} \quad p(C_j) = \frac{\sum_i p(C_j | x_i)}{N}$$

EM Algorithm

- Boot Step:
  - Initialize K clusters: $C_1, \ldots, C_K$
    (μ_j, Σ_j) and $P(C_j)$ for each cluster j.
- Iteration Step:
  - Expectation Step
    $$p(C_j | x_i) = \frac{p(x_i | C_j)\cdot p(C_j)}{p(x_i)} = \frac{p(x_i | C_j)\cdot p(C_j)}{\sum_j p(x_i | C_j)\cdot p(C_j)}$$
  - Maximization Step
    $$\mu_j = \frac{\sum_i p(C_j | x_i)\cdot x_i}{\sum_i p(C_j | x_i)} \quad \Sigma_j = \frac{\sum_i p(C_j | x_i)\cdot (x_i - \mu_j)\cdot (x_i - \mu_j)^T}{\sum_i p(C_j | x_i)} \quad p(C_j) = \frac{\sum_i p(C_j | x_i)}{N}$$

EM Demo

- Demo

  http://www.neurosci.aist.go.jp/~akaho/MixtureEM.html
- Example

  http://www-2.cs.cmu.edu/~awm/tutorials/gmm13.pdf

EM Applications

- Blobworld: Image Segmentation Using Expectation-Maximization and its Application to Image Querying

Image Segmentation using EM

- Step 1: Feature Extraction
- Step 2: Image Segmentation using EM

Symbols

- The feature vector for pixel i is called x_i.
- There are going to be K segments; K is given.
- The j-th segment has a Gaussian distribution with parameters $\theta_j=(\mu_j,\Sigma_j)$.
- α_j's are the weights (which sum to 1) of Gaussians. Θ is the collection of parameters: $\Theta = (\alpha_1, \ldots, \alpha_k, \theta_1, \ldots, \theta_k)$

Initialization

- Each of the K Gaussians will have parameters $\theta_j=(\mu_j,\Sigma_j)$, where
  - μ_j is the mean of the j-th Gaussian.
  - Σ_j is the covariance matrix of the j-th Gaussian.
- The covariance matrices are initialed to be the identity matrix.
- The means can be initialized by finding the average feature vectors in each of K windows in the image; this is data-driven initialization.

E-Step

$$p(j|x_i,\Theta)=\frac{\alpha_j f_j(x_i|\theta_j)}{\sum_{k=1}^{K}\alpha_k f_k(x_i|\theta_k)}$$

$$f_j(x|\theta_j)=\frac{1}{(2\pi)^{d/2}|\Sigma_j|^{1/2}}e^{-\frac{1}{2}(x-\mu_j)^T\Sigma_j^{-1}(x-\mu_j)}$$

M-Step

$$\mu_j^{new}=\frac{\sum_{i=1}^{N}x_i\, p(j|x_i,\Theta^{old})}{\sum_{i=1}^{N}p(j|x_i,\Theta^{old})}$$

$$\Sigma_j^{new}=\frac{\sum_{i=1}^{N}p(j|x_i,\Theta^{old})(x_i-\mu_j^{new})(x_i-\mu_j^{new})^T}{\sum_{i=1}^{N}p(j|x_i,\Theta^{old})}$$

$$\alpha_j^{new}=\frac{1}{N}\sum_{i=1}^{N}p(j|x_i,\Theta^{old})$$

Sample Results

Object Class Recognition in CBIR

- The Goal:

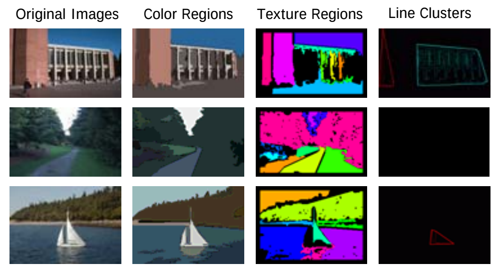
  Automatic image labeling (annotation) to enable object-based image retrieval

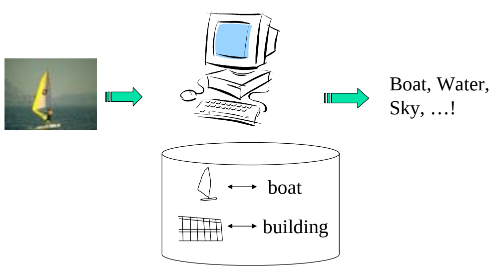
Problem Statement

Known: Some images and their corresponding descriptions

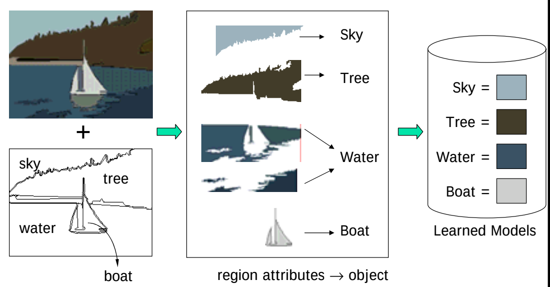
To solve: What object classes are present in new images

Abstract Regions

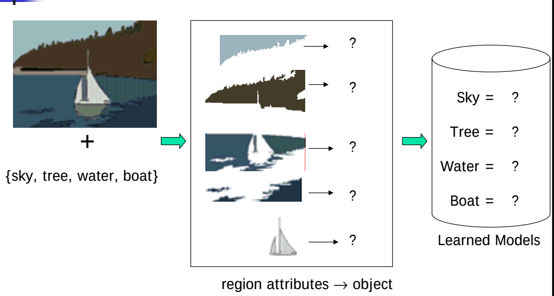




Object Model Learning (Ideal)

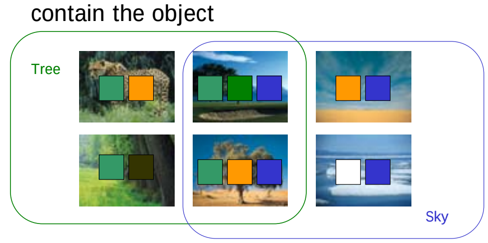


Object Model Learning (Real)

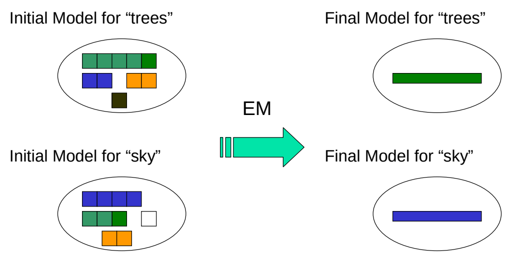


Model Initial Estimation

- Estimate the initial model of an object using all the region features from all images that contain the object



EM Variant

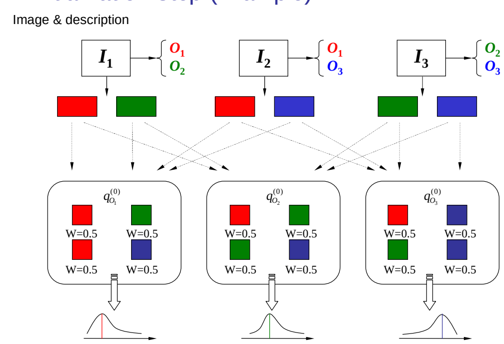


Object Model Learning

Assumptions

- The feature distribution of each object within a region is a Gaussian;
- Each image is a set of regions, each of which can be modeled as a mixture of multivariate Gaussian distributions.

1. Initialization Step (Example)



2. Iteration Step (Example)

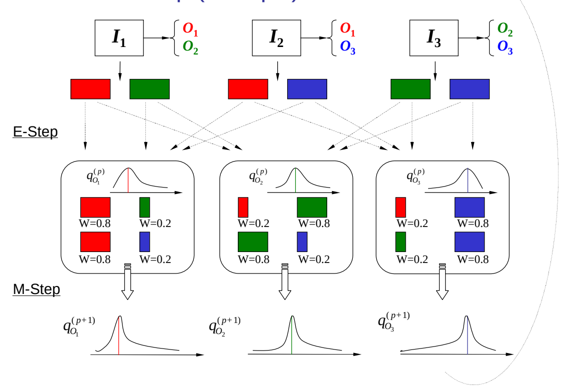
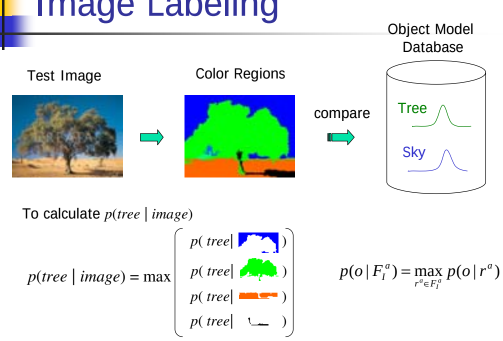


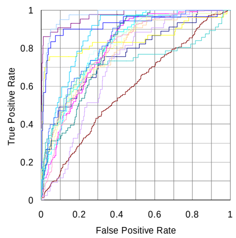
Image Labeling



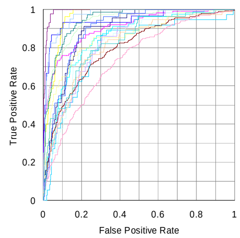
Experiments

- 860 images
- 18 keywords: mountains (30), orangutan (37), track (40), tree trunk (43), football field (43), beach (45), prairie grass (53), cherry tree (53), snow (54), zebra (56), polar bear (56), lion (71), water (76), chimpanzee (79), cheetah (112), sky (259), grass (272), tree (361).
- A set of cross-validation experiments (80% as the training set and the other 20% as the test set)

ROC Charts



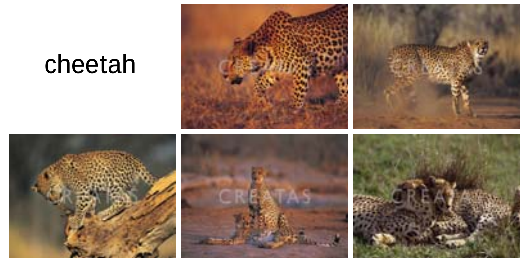
Independent Treatment of
Color and Texture



Using Intersection of
Color and Texture

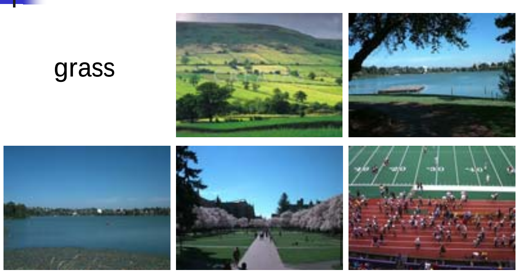
Sample Results

cheetah



Sample Results (Cont.)

grass



Sample Results (Cont.)

lion