Plan

• What is Clustering? Challenges in Clustering
• Clustering (for Segmentation)
  – K-Means
  – GMMs (and Expectation-Maximization)
  – Mean-Shift
• Other uses of clustering in vision
  – Texture and Textons
  – Quantization
  – Bag of Words
What is Clustering?

• What is clustering?
  – Grouping of “objects” into meaningful categories
  – Given a representation of N objects, find k clusters based on a suitable measure of similarity.

• Data Clustering is useful in and beyond Computer Vision
  – Segmentation as clustering (today)
  – Texture modeling
  – Quantization
  – Beyond
    • Data exploration
    • Compression
    • Natural classification

• Evidently important: Google Scholar tells us that more than 1500 papers get published on clustering a year!

Feature Space

• Every token is identified by a set of salient visual characteristics. For example:
  – Position
  – Color
  – Texture
  – Motion vector
  – Size, orientation (if token is larger than a pixel)
Feature Space

Source: Savarese slides.

Source: K. Grauman
*Feature space:* each token is represented by a point

Source: Savarese slides.
Token similarity is thus measured by distance between points ("feature vectors") in feature space.

\[ \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}. \]

Source: Savarese slides.
Cluster together tokens with high similarity
Source: Savarese slides.
Example: Topic Discovery

- 800,000 scientific papers clustered into 776 topics based on how often the papers were cited together by authors of other papers.

Formal Definition of Clustering

• Given a set of N data samples $D = x_1, x_2, \ldots, x_N$ in a d-dimensional feature space, $D$ is partitioned into a number of disjoint subsets $D_j$:

$$D = \bigcup_{j=1}^{k} D_j \quad \text{where} \quad D_i \cup D_j = \emptyset \quad \forall i \neq j$$

where the points in each subset are similar to each other according to the given similarity function.

• A partition is denoted by

$$\pi = (D_1, D_2, \ldots, D_k)$$

and clustering is then formulated as

$$\pi^* = \arg \min_{\pi} f(\pi)$$

for $f(\cdot)$ that captures the desired cluster properties.
Plan

• What is Clustering? Challenges in Clustering
• Clustering (for Segmentation)
  – K-Means
  – GMMs (and Expectation-Maximization)
  – Mean-Shift
• Other uses of clustering in vision
  – Texture and Textons
  – Quantization
  – Bag of Words
K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in $\mu_i$:
   
   (a) Classify $N$ samples according to nearest $\mu_i$
   (b) Recompute $\mu_i$


Data Point
K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in $\mu_i$:

   (a) Classify $N$ samples according to nearest $\mu_i$

   (b) Recompute $\mu_i$

K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_C$

2. Repeat until no change in $\mu_i$:
   (a) Classify $N$ samples according to nearest $\mu_i$
   (b) Recompute $\mu_i$

K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in $\mu_i$:
   (a) Classify $N$ samples according to nearest $\mu_i$
   (b) Recompute $\mu_i$

K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in $\mu_i$:
   (a) Classify $N$ samples according to nearest $\mu_i$
   (b) Recompute $\mu_i$
K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in $\mu_i$:
   
   (a) Classify $N$ samples according to nearest $\mu_i$
   
   (b) Recompute $\mu_i$

K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in $\mu_i$:
   
   (a) Classify $N$ samples according to nearest $\mu_i$
   
   (b) Recompute $\mu_i$

K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in $\mu_i$:

   (a) Classify $N$ samples according to nearest $\mu_i$

   (b) Recompute $\mu_i$

K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in $\mu_i$:
   
   (a) Classify $N$ samples according to nearest $\mu_i$
   (b) Recompute $\mu_i$

Points already assigned to nearest centers: Algorithm ends

K-Means++ Clustering

- Choose starting centers iteratively.

- Let $D(x)$ be the distance from $x$ to the nearest existing center, take $x$ as new center with probability $\propto D(x)^2$.

- Repeat until no change in $\mu_i$:
  - Classify $N$ samples according to nearest $\mu_i$
  - Recompute $\mu_i$
Source: Savarese slides.
Source: Savarese slides.
Source: Savarese slides.
K-Means pros and cons

• **Pros**
  – Simple and fast
  – (Always) converges to a local minimum of the error function
  – Available implementations (e.g., in Matlab)

\[
\sum_{\text{clusters } i} \sum_{\text{points } p \text{ in cluster } i} \| p - c_i \|^2
\]

• **Cons**
  – Need to pick K
  – Sensitive to initialization
  – Only finds “spherical” clusters
  – Sensitive to outliers

Source: Savarese slides.
Choosing Exemplars (Medoids)

• like k-means, but means must be data points

• Algorithms:
  – greedy k-means
  – affinity propagation (Frey & Dueck 2007)
  – medoid shift (Sheikh et al. 2007)

• Scene Summarization
Plan

• What is Clustering? **Challenges in Clustering**
• Clustering (for Segmentation)
  – K-Means
  – GMMs (and Expectation-Maximization)
  – Mean-Shift
• Other uses of clustering in vision
  – Texture and Textons
  – Quantization
  – Bag of Words
User’s Dilemma

1. What is a cluster?
2. How to define pair-wise similarity?
3. Which features? Which normalizations scheme?
4. How many clusters?
5. Which clustering method?
6. Are the discovered clusters and partitioning valid?
7. Does the data have any clustering tendency?

Cluster Similarity?

- **Compact Clusters**
  - Within-cluster distance < between-cluster connectivity

- **Connected Clusters**
  - Within-cluster connectivity > between-cluster connectivity

- **Ideal cluster:** compact and isolated.

Representation; what features?

- There is no universal representation.

Good Representations

- A *good* representation leads to compact and isolated clusters.

How should the features be weighted?

- Two different meaningful groupings produced by different weighting schemes.

How do we decide on the number of clusters?

- These samples are generated by 6 independent classes.

Cluster Validity

- Clustering algorithms find clusters, even if there are no natural clusters in the data!

Choosing a Clustering Method

• Which is best?

15 Data points
MST
FORGY
ISODATA
WISH
CLUSTER
Complete Link
JP

Choosing a Clustering Method

- Depends on problem/data.
- Each algorithm imposes some structure.

Plan

• What is Clustering? Challenges in Clustering
• Clustering (for Segmentation)
  – K-Means
  – GMMs (and Expectation-Maximization)
  – Mean-Shift
• Other uses of clustering in vision
  – Texture and Textons
  – Quantization
  – Bag of Words
Probabilistic clustering

• Basic questions
  – what’s the probability that a point \( x \) is in cluster \( m \)?
  – what’s the shape of each cluster?

• K-means doesn’t answer these questions

• Basic idea
  – instead of treating the data as a bunch of points, assume that they are all generated by sampling a continuous function
  – This function is called a **generative model**
    • defined by a vector of parameters \( \theta \)

Source: Seitz slides.
Gaussian Mixture Models

• Recall the Gaussian distribution

\[ p(x) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right] \]

• It forms the basis for the mixture of Gaussians density

• The Gaussian mixture is a linear superposition of Gaussians:

\[ p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k) \]

• The \( \pi_k \) are non-negative scalars called mixing coefficients and they govern the relative importance between the various Gaussians in the mixture density. \( \sum_k \pi_k = 1 \)
GMM: Introducing Latent Variables

• Define a K-dimensional binary random variable $z$
• $z$ has a 1-of-K representation such that a particular element $z_k$ is 1 and all of the others are zero. Hence:

$$z_k \in \{0, 1\}$$

$$\sum_{k} z_k = 1$$

• The marginal distribution over $z$ is specified in terms of the mixing coefficients:

$$p(z_k = 1) = \pi_k$$

And recall that $0 \leq \pi_k \leq 1$ and $\sum_{k} \pi_k = 1$
GMM: Introducing Latent Variables

• Since $z$ has a 1-of-$K$ representation, we can also write the distribution as

$$p(z) = \prod_{k=1}^{K} \pi_k^{z_k}$$

• The conditional distribution of $x$ given $z$ is a Gaussian:

$$p(x|z_k = 1) = \mathcal{N}(x|\mu_k, \Sigma_k)$$

$$p(x|z) = \prod_{k=1}^{K} \mathcal{N}(x|\mu_k, \Sigma)^{z_k}$$
GMM: Introducing Latent Variables

- We are interested in the marginal distribution of $x$

$$p(x) = \sum_z p(x, z)$$

$$= \sum_z p(z)p(x|z)$$

$$= \sum_k \prod_{k=1}^K \pi_k^z \mathcal{N}(x|\mu_k, \Sigma_k)^z_k$$

$$= \sum_{k=1}^K \pi_k \mathcal{N}(x|\mu_k, \Sigma_k)$$

- So, given our latent variable $z$, the marginal distribution of $x$ is a Gaussian mixture.

- If we have $N$ observations, $x_1, \ldots, x_N$, then because of our chosen representation, if follows that we have a latent variable $z_n$ for each observed data point $x_n$. 
Component Responsibility Term

- We need to also express the conditional probability of $z$ given $x$.
- Denote this conditional $p(z_k = 1|x)$ as $\gamma(z_k)$.
- Via Bayes’ theorem:

$$
\gamma(z_k) = \frac{p(z_k = 1)p(x|z_k = 1)}{\sum_{j=1}^{K} p(z_j = 1)p(x|z_j = 1)} = \frac{\pi_k \mathcal{N}(x|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x|\mu_j, \Sigma_j)}
$$

- View $\pi_k$ as the prior probability of $z_k = 1$ and the quantity $\gamma(z_k)$ as the corresponding posterior probability after observing $x$.
- $\gamma(z_k)$ is also the responsibility that component $k$ takes for $x$. 


Sampling from the GMM

• To sample from the GMM, we can first generate a value for $z$ from the marginal distribution $p(z)$. Denote this sample $\hat{z}$.
• Then, sample from the conditional distribution $p(x|\hat{z})$.
• The figure below-left shows samples from a three-mixture and colors the samples based on the component $(z)$. The figure below-middle shows samples from the marginal $p(x)$ and ignores $z$. On the right, we show the $\gamma(z_k)$ for each sampled point, colored accordingly.
Maximum-Likelihood Fitting

Suppose we have a set of $N$ observations $\{x_1, \ldots, x_N\}$ that we wish to model with a GMM.

Consider this data set as an $N \times d$ matrix $X$ in which the $n^{th}$ row is given by $x_n^T$.

Similarly, the corresponding latent variables define an $N \times K$ matrix $Z$ with rows $z_n^T$.

The log-likelihood of the corresponding GMM is given by

$$\ln p(X|\pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left[ \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \right].$$

Ultimately, we want to find the values of the parameters $\pi, \mu, \Sigma$ that maximize this function.

However, maximizing the log-likelihood terms for GMMs is much more complicated than for the case of a single Gaussian. Why?

The difficulty arises from the sum over $k$ inside of the log-term. The log function no longer acts directly on the Gaussian, and no closed-form solution is available.
Singularities with GMM Fitting

There is a significant problem when we apply MLE to estimate GMM parameters.

Consider simply covariances defined by \( \Sigma_k = \sigma_k^2 I \).

Suppose that one of the components of the mixture model, \( j \), has its mean \( \mu_j \) exactly equal to one of the data points so that \( \mu_j = x_n \) for some \( n \).

This term contributes

\[
\mathcal{N}(x_n | x_n, \sigma_j^2 I) = \frac{1}{(2\pi)^{1/2} \sigma_j}
\]

Consider the limit \( \sigma_j \to 0 \) to see that this term goes to infinity and hence the log-likelihood will also go to infinity.

Thus, the maximization of the log-likelihood function is not a well posed problem because such a singularity will occur whenever one of the components collapses to a single, specific data point.
Singularities with GMM Fitting

\[ p(x) \]
Expectation-Maximization

- External PDF slides
- em_fitting_gmm.pdf
Problems with EM

1. Local minima
   k-means is NP-hard even with $k=2$

2. Need to know number of segments
   solutions: AIC, BIC, Dirichlet process mixture

3. Need to choose generative model

Source: Seitz slides.
Applications of EM

• Turns out this is useful for all sorts of problems
  – any clustering problem
  – any model estimation problem
  – missing data problems
  – finding outliers
  – segmentation problems
    • segmentation based on color
    • segmentation based on motion
    • foreground/background separation
  – ...

Source: Seitz slides.
EM demo

Plan

• What is Clustering? Challenges in Clustering
• Clustering (for Segmentation)
  – K-Means
  – GMMs (and Expectation-Maximization)
  – Mean-Shift
• Other uses of clustering in vision
  – Texture and Textons
  – Quantization
  – Bag of Words
Finding Modes in a Histogram

• How Many Modes Are There?
  – Easy to see, hard to compute

Source: Seitz slides.
Mean Shift [Comaniciu & Meer]


- **Iterative Mode Search**
  1. Initialize random seed, and window W
  2. Calculate center of gravity (the “mean”) of W: \( \sum_{x \in W} x H(x) \)
  3. Translate the search window to the mean
  4. Repeat Step 2 until convergence

Source: Seitz slides.
Mean-shift for image segmentation

- Useful to take into account spatial information
  - instead of (R, G, B), run in (R, G, B, x, y) space
  - D. Comaniciu, P. Meer, Mean shift analysis and applications, *7th International Conference on Computer Vision*, Kerkyra, Greece, September 1999, 1197-1203.

More Examples:  [http://www.caip.rutgers.edu/~comanici/segm_images.html](http://www.caip.rutgers.edu/~comanici/segm_images.html)

Source: Seitz slides.
Mean shift algorithm


- The mean shift algorithm seeks a *mode* or local maximum of density of a given distribution
  - Choose a search window (width and location)
  - Compute the mean of the data in the search window
  - Center the search window at the new mean location
  - Repeat until convergence
Mean Shift

Slide by Y. Ukrainitz & B. Sarel
Mean Shift
Mean Shift

Window
Center of mass
Mean Shift vector

Slide by Y. Ukrainitz & B. Sarel
Mean Shift

- Window
- Center of mass

Mean Shift vector

Slide by Y. Ukrainitz & B. Sarel
Mean Shift

window

Center of mass

Mean Shift vector

Slide by Y. Ukrainitz & B. Sarel
Mean Shift

Window

Center of mass

Slide by Y. Ukrainitz & B. Sarel
Computing The Mean Shift

Simple Mean Shift procedure:
- Compute mean shift vector
- Translate the Kernel window by $m(x)$

\[
m(x) = \frac{\sum_{i=1}^{n} x_{i} g \left( \frac{\|x - x_{i}\|^{2}}{h} \right)}{\sum_{i=1}^{n} g \left( \frac{\|x - x_{i}\|^{2}}{h} \right)} - x
\]

Source: Savarese slides.
Multimodal distributions
Real Modality Analysis

- Tessellate the space with windows
- Merge windows that end up near the same “peak” or model

Source: Savarese slides.
**Attraction basin**

- **Attraction basin**: the region for which all trajectories lead to the same mode
- **Cluster**: all data points in the attraction basin of a mode
Attraction basin

Slide by Y. Ukrainitz & B. Sarel
Segmentation by Mean Shift

- Find features (color, gradients, texture, etc)
- Initialize windows at individual pixel locations
- Perform mean shift for each window until convergence
- Merge windows that end up near the same “peak” or mode

Source: Savarese slides.
Mean shift segmentation results

http://www.caip.rutgers.edu/~comanici/MSPAMI/msPamiResults.html

Source: Savarese slides.
Mean shift pros and cons

• Pros
  – Does not assume spherical clusters
  – Just a single parameter (window size)
  – Finds variable number of modes
  – Robust to outliers

• Cons
  – Output depends on window size
  – Computationally expensive
  – Does not scale well with dimension of feature space

Source: Savarese slides.
Plan

• What is Clustering? Challenges in Clustering
• Clustering (for Segmentation)
  – K-Means
  – GMMs (and Expectation-Maximization)
  – Mean-Shift
• Other uses of clustering in vision
  – Texture and Textons
  – Quantization
  – Bag of Words
Texture Feature

- Texture Gradient $TG(x,y,r,\theta)$
  - $\chi^2$ difference of texton histograms
  - Textons are vector-quantized filter outputs
### $P_b$ Images I

<table>
<thead>
<tr>
<th>Image</th>
<th>Canny</th>
<th>2MM</th>
<th>Us</th>
<th>Human</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="Image" /></td>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="Canny" /></td>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="2MM" /></td>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="Us" /></td>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="Human" /></td>
</tr>
<tr>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="Image" /></td>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="Canny" /></td>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="2MM" /></td>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="Us" /></td>
<td><img src="http://www.cs.berkeley.edu/projects/vision" alt="Human" /></td>
</tr>
</tbody>
</table>

Source: Martin, Fowlkes, Malik NIPS 2002 slides.
\( P_b \) Images II

<table>
<thead>
<tr>
<th>Image</th>
<th>Canny</th>
<th>2MM</th>
<th>Us</th>
<th>Human</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="red_flowers.jpg" alt="Image" /></td>
<td><img src="red_flowers_canny.jpg" alt="Canny" /></td>
<td><img src="red_flowers_2mm.jpg" alt="2MM" /></td>
<td><img src="red_flowers_us.jpg" alt="Us" /></td>
<td><img src="red_flowers_human.jpg" alt="Human" /></td>
</tr>
<tr>
<td><img src="building.jpg" alt="Image" /></td>
<td><img src="building_canny.jpg" alt="Canny" /></td>
<td><img src="building_2mm.jpg" alt="2MM" /></td>
<td><img src="building_us.jpg" alt="Us" /></td>
<td><img src="building_human.jpg" alt="Human" /></td>
</tr>
<tr>
<td><img src="boat.jpg" alt="Image" /></td>
<td><img src="boat_canny.jpg" alt="Canny" /></td>
<td><img src="boat_2mm.jpg" alt="2MM" /></td>
<td><img src="boat_us.jpg" alt="Us" /></td>
<td><img src="boat_human.jpg" alt="Human" /></td>
</tr>
</tbody>
</table>

Source: Martin, Fowlkes, Malik NIPS 2002 slides.
P_b Images III

![Image](http://www.cs.berkeley.edu/projects/vision)

Source: Martin, Fowlkes, Malik NIPS 2002 slides.
The (Very Common) Bag-of-Features Pipeline

Examples include Schüldt et al. ICPR 2004, Niebles et al. IJCV 2008, and many works building on this basic idea.
Next Lecture: Model-Fitting and Contours

• Readings: FP 10; SZ 4.3, 5.1