Computer Vision - Lecture 6
Segmentation and Grouping
10.11.2011

Bastian Leibe
RWTH Aachen
http://www.mmp.rwth-aachen.de
leibe@umic.rwth-aachen.de

Examples of Grouping in Vision

What things should be grouped?
What cues indicate groups?

Object-level grouping

Slide credit: Kristen Grauman
B. Leibe

Similarity

Slide credit: Kristen Grauman
B. Leibe

Symmetry

Slide credit: Kristen Grauman
B. Leibe

Course Outline

- Image Processing Basics
- Segmentation
  - Segmentation and Grouping
  - Graph-theoretic Segmentation
- Recognition
  - Global Representations
  - Subspace representations
- Local Features & Matching
- Object Categorization
- 3D Reconstruction
- Motion and Tracking

Slide credit: Kristen Grauman
B. Leibe

Common Fate

Image credit: Arthus-Bertrand (via F. Durand)

Slide credit: Kristen Grauman
B. Leibe
The Gestalt School

- Grouping is key to visual perception
- Elements in a collection can have properties that result from relationships
  - “The whole is greater than the sum of its parts”

Gestalt Factors

- These factors make intuitive sense, but are very difficult to translate into algorithms.

Gestalt Theory

- Gestalt: whole or group
  - Whole is greater than sum of its parts
  - Relationships among parts can yield new properties/features
- Psychologists identified series of factors that predispose set of elements to be grouped (by human visual system)

Continuity through Occlusion Cues
Continuity through Occlusion Cues

Continuity, explanation by occlusion

Figure-Ground Discrimination

The Ultimate Gestalt?

Image Segmentation

- Goal: identify groups of pixels that go together
The Goals of Segmentation

• Separate image into coherent “objects”

Image

Human segmentation

The Goals of Segmentation

• Separate image into coherent “objects”

Group together similar-looking pixels for efficiency of further processing

“superpixels”

Topics of This Lecture

• Segmentation and grouping
  - Gestalt principles
  - Image Segmentation

• Segmentation as clustering
  - k-Means
  - Feature spaces

• Probabilistic clustering
  - Mixtures of Gaussians, EM

• Model-free clustering
  - Mean-Shift clustering

Image Segmentation: Toy Example

• These intensities define the three groups.
• We could label every pixel in the image according to which of these primary intensities it is.
  - i.e., segment the image based on the intensity feature.
• What if the image isn’t quite so simple?

• Now how to determine the three main intensities that define our groups?
• We need to cluster.
Clustering
• With this objective, it is a "chicken and egg" problem:
  - If we knew the cluster centers, we could allocate points to
groups by assigning each to its closest center.
  - If we knew the group memberships, we could get the centers by
  computing the mean per group.

K-Means Clustering
• Basic idea: randomly initialize the \( k \) cluster centers, and
iterate between the two steps we just saw.
  1. Randomly initialize the cluster centers, \( c_1, \ldots, c_k \).
  2. Given cluster centers, determine points in each cluster
     - For each point \( p \), find the closest \( c_i \). Put \( p \) into cluster \( i \)
  3. Given points in each cluster, solve for \( c_i \)
     - Set \( c_i \) to be the mean of points in cluster \( i \)
  4. If \( c_i \) have changed, repeat Step 2

Properties
• Will always converge to some solution
• Can be a "local minimum"
  - Does not always find the global minimum of objective function:
    \[
    \sum_{i} \sum_{p \text{ in cluster } i} \| p - c_i \|^2
    \]

Java demo:
http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/AppletKM.html

Segmentation as Clustering
\( K=2 \)
\( K=3 \)

K-Means++
• Can we prevent arbitrarily bad local minima?
  1. Randomly choose first center.
  2. Pick new center with prob. proportional to
     \[ \frac{1}{\sum_{i} \sum_{p \text{ in cluster } i} \| p - c_i \|^2} \]
  3. Repeat until \( k \) centers.

• Expected error \( = O(\log k) \) * optimal

Arthur & Vassilvitskii 2007
Feature Space

- Depending on what we choose as the feature space, we can group pixels in different ways.
- Grouping pixels based on intensity similarity
- Feature space: intensity value (1D)

Segmentation as Clustering

- Depending on what we choose as the feature space, we can group pixels in different ways.
- Grouping pixels based on texture similarity
- Feature space: filter bank responses (e.g., 24D)

Smoothing Out Cluster Assignments

- Assigning a cluster label per pixel may yield outliers:
- How can we ensure they are spatially smooth?

K-Means Clustering Results

- K-means clustering based on intensity or color is essentially vector quantization of the image attributes:
  - Clusters don’t have to be spatially coherent

Perceptual and Sensory Augmented Computing
Computer Vision WS 11/12

Slide credit: Kristen Grauman
K-Means Clustering Results

- K-means clustering based on intensity or color is essentially vector quantization of the image attributes
  - Clusters don’t have to be spatially coherent
- Clustering based on (r,g,b,x,y) values enforces more spatial coherence

Summary K-Means

- Pros
  - Simple, fast to compute
  - Converges to local minimum of within-cluster squared error
- Cons/Issues
  - Setting k?
  - Sensitive to initial centers
  - Sensitive to outliers
  - Detects spherical clusters only
  - Assuming means can be computed

Topics of This Lecture

- Segmentation and grouping
  - Gestalt principles
  - Image Segmentation
- Segmentation as clustering
  - K-means
  - Feature spaces
- Probabilistic clustering
  - Mixture of Gaussians, EM
- Model-free clustering
  - Mean-shift clustering

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 \( \theta \)

Mixture of Gaussians

- One generative model is a mixture of Gaussians (MoG)
  - K Gaussian blobs with means \( \mu_b \), covariance matrices \( V_b \), dimension d
  - Blob b defined by:
    \[
    P(x|\theta_b) = \frac{1}{\sqrt{(2\pi)^d|V_b|}} e^{-\frac{1}{2}(x-\mu_b)^T V_b^{-1}(x-\mu_b)}
    \]
  - Blob b is selected with probability \( x_b \)
  - The likelihood of observing x is a weighted mixture of Gaussians
    \[
    P(x|\theta) = \sum_{b=1}^{K} x_b P(x|\theta_b) \quad \theta = [\mu_1, \ldots, \mu_K, V_1, \ldots, V_K]
    \]

Expectation Maximization (EM)

- Goal
  - Find blob parameters \( \theta \) that maximize the likelihood function:
    \[
    P(\text{data}|\theta) = \prod_{x} P(x|\theta)
    \]
- Approach:
  1. E-step: given current guess of blobs, compute ownership of each point
  2. M-step: given ownership probabilities, update blobs to maximize likelihood function
  3. Repeat until convergence
EM Details

• **E-step**
  - Compute probability that point \( x \) is in blob \( b \), given current guess of \( \theta \)
  \[
P(b|x|\theta) = \frac{\alpha_b P(x|\mu_b, \Sigma_b)}{\sum_{i=1}^{K} \alpha_i P(x|\mu_i, \Sigma_i)}
\]

• **M-step**
  - Compute probability that blob \( b \) is selected
  \[
  \alpha_b^{(n+1)} = \frac{1}{N} \sum_{i=1}^{N} P(\theta|x_i, \mu_b, \Sigma_b)
  \]
  - Mean of blob \( b \)
  \[
  \mu_b^{(n+1)} = \frac{\sum_{i=1}^{N} x_i P(\theta|x_i, \mu_b, \Sigma_b)}{\sum_{i=1}^{N} P(\theta|x_i, \mu_b, \Sigma_b)}
  \]
  - Covariance of blob \( b \)
  \[
  \Sigma_b^{(n+1)} = \frac{\sum_{i=1}^{N} (x_i - \mu_b^{(n+1)})(x_i - \mu_b^{(n+1)})^T P(\theta|x_i, \mu_b, \Sigma_b)}{\sum_{i=1}^{N} P(\theta|x_i, \mu_b, \Sigma_b)}
  \]

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
    - ...

• EM demo

Segmentation with EM

Original image

EM segmentation results

Summary: Mixtures of Gaussians, EM

• **Pros**
  - Probabilistic interpretation
  - Soft assignments between data points and clusters
  - Generative model, can predict novel data points
  - Relatively compact storage

• **Cons**
  - Local minima
    - k-means is NP-hard even with \( k=2 \)
  - Initialization
    - Often a good idea to start with some k-means iterations.
  - Need to know number of components
    - Solutions: model selection (AIC, BIC), Dirichlet process mixture
  - Need to choose generative model
  - Numerical problems are often a nuisance

Topics of This Lecture

• Segmentation and grouping
  - Gestalt principles
  - Image segmentation

• Segmentation as clustering
  - k-means
  - Feature spaces
  - Probabilistic clustering
    - Mixtures of Gaussians, EM

• Model-free clustering
  - Mean-Shift clustering

Finding Modes in a Histogram

• How many modes are there?
  - Mode = local maximum of the density of a given distribution
  - Easy to see, hard to compute
Mean-Shift Segmentation

- An advanced and versatile technique for clustering-based segmentation

Mean-Shift Algorithm

- Iterative Mode Search
  1. Initialize random seed, and window $W$
  2. Calculate center of gravity (the “mean”) of $W$
  3. Shift the search window to the mean
  4. Repeat Step 2 until convergence

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


Slide credits: Svetlana Lazebnik
Mean-Shift

- Region of interest
- Center of mass
- Mean Shift vector

Mean-Shift

- Region of interest
- Center of mass
- Mean Shift vector

Mean-Shift

- Region of interest
- Center of mass

Mean-Shift Clustering

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

Real Modality Analysis

- Tessellate the space with windows
- Run the procedure in parallel

The blue data points were traversed by the windows towards the mode.
Mean-Shift Clustering/Segmentation

- 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

Mean-Shift Segmentation Results

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

Problem: Computational Complexity

- Need to shift many windows...
- Many computations will be redundant.

Speedups: Basin of Attraction

1. Assign all points within radius r of end point to the mode.

Slide credit: Svetlana Lazebnik
Speedups

1. Assign all points within radius $r/c$ of the search path to the mode.

Summary Mean-Shift

- **Pros**
  - General, application-independent tool
  - Model-free, does not assume any prior shape (spherical, elliptical, etc.) on data clusters
  - Just a single parameter (window size $h$)
  - $h$ has a physical meaning (unlike k-means)
  - Finds variable number of modes
  - Robust to outliers

- **Cons**
  - Output depends on window size
  - Window size (bandwidth) selection is not trivial
  - Computationally (relatively) expensive (~2s/image)
  - Does not scale well with dimension of feature space

Segmentation: Caveats

- We’ve looked at **bottom-up** ways to segment an image into regions, yet finding meaningful segments is intertwined with the recognition problem.
- Often want to avoid making hard decisions too soon
- Difficult to evaluate; when is a segmentation successful?

Generic Clustering

- We have focused on ways to group pixels into image segments based on their appearance
  - Find groups; “quantize” feature space
- In general, we can use clustering techniques to find groups of similar “tokens”, provided we know how to compare the tokens.
  - E.g., segment an image into the types of motions present
  - E.g., segment a video into the types of scenes (shots) present

References and Further Reading

- Background information on segmentation by clustering and on Normalized Cuts can be found in Chapter 14 of
- More on the EM algorithm can be found in Chapter 16.1.2.
- Read Max Wertheimer’s classic thoughts on Gestalt
  - http://psy.ed.asu.edu/~classics/Wertheimer/Forms/forms.htm
- Try the k-means and EM demos at
  - http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/AppletKM.html