Computer Vision - Lecture 6

Segmentation

09.11.2016

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

Course Outline

- Image Processing Basics
  - Structure Extraction
- Segmentation
  - Segmentation as Clustering
  - Graph-theoretic Segmentation
- Recognition
  - Global Representations
  - Subspace representations
- Local Features & Matching
- Object Categorization
- 3D Reconstruction

Recap: Chamfer Matching

- Chamfer Distance
  - Average distance to nearest feature
    \[ D_{chamfer}(T, I) = \frac{1}{|T|} \sum_{t \in T} d(t) \]
  - This can be computed efficiently by correlating the edge template with the distance-transformed image.

Recap: Hough Transform

- How can we use this to find the most likely parameters \((m, b)\) for the most prominent line in the image space?
  - Let each edge point in image space vote for a set of possible parameters in Hough space.
  - Accumulate votes in discrete set of bins; parameters with the most votes indicate line in image space.

Recap: Hough Transform for Circles

- Circle: center \((a, b)\) and radius \(r\)
  \[ (x - a)^2 + (y - b)^2 = r^2 \]
- For an unknown radius \(r\), unknown gradient direction
Generalized Hough Transform

- What if we want to detect arbitrary shapes defined by boundary points and a reference point?

At each boundary point, compute displacement vector: \( r = a - p_i \).

For a given model shape: store these vectors in a table indexed by gradient orientation \( \theta \).

[Dana H. Ballard, Generalizing the Hough Transform to Detect Arbitrary Shapes, 1980]

Example: Generalized Hough Transform

Say we’ve already stored a table of displacement vectors as a function of edge orientation for this model shape.

Now we want to look at some edge points detected in a new image, and vote on the position of that shape.

To detect the model shape in a new image:

- For each edge point
  - Index into table with its gradient orientation \( \theta \)
  - Use retrieved \( r \) vectors to vote for position of reference point
- Peak in this Hough space is reference point with most supporting edges

Assuming translation is the only transformation here, i.e., orientation and scale are fixed.
Example: Generalized Hough Transform

Slide credit: Svetlana Lazebnik

Votes for points with $\theta = \theta_1$

Displacement vectors for model points

Range of voting locations for test point

Votes for points with $\theta = \theta_2$

Application in Recognition

- Instead of indexing displacements by gradient orientation, index by “visual codeword”.

Training image

Visual codeword with displacement vectors

Test image

- We’ll hear more about this in later lectures...

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

Examples of Grouping in Vision

- Determining image regions
- Grouping video frames into shots
- What things should be grouped?
- What cues indicate groups?
- Object-level grouping

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

- Illusory/subjective contours
- Occlusion
- Familiar configuration

http://en.wikipedia.org/wiki/Gestalt_psychology

http://psy.ed.asu.edu/~classics/Wertheimer/Forms/forms.htm

“...at the window and see a house, trees, sky. Theoretically I might say there are 327 brightnesses and nuances of colour. Do I have ‘327’? No, I have sky, house, and trees.”

Max Wertheimer
(1880-1943)

Untersuchungen zur Lehre von der Gestalt, Psychologische Forschung, Vol. 4, pp. 301-350, 1923
http://psy.ed.asu.edu/~classics/Wertheimer/Forms/forms.htm
Continuity through Occlusion Cues

Continuity, explanation by occlusion

The Ultimate Gestalt?

Image Segmentation

• Goal: identify groups of pixels that go together

The Goals of Segmentation

• Separate image into coherent “objects”
**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

**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?

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

K-Means++

- Can we prevent arbitrarily bad local minima?
  1. Randomly choose first center.
  2. Pick new center with prob. proportional to \( ||p - c_i||^2 \)
    - (Contribution of \( p \) to total error)
  3. Repeat until \( k \) centers.
- Expected error = \( O(\log k) \) * optimal

Feature Space

- Depending on what we choose as the feature space, we can group pixels in different ways.

  Grouping pixels based on color similarity

  Feature space: color value (3D)

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?

  ![Image: Original vs. Labeled by cluster center's intensity]

Segmentation as Clustering

- Depending on what we choose as the feature space, we can group pixels in different ways.

  - Grouping pixels based on intensity+position similarity

  ![Image: Simple way to encode both similarity and proximity.]

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 $\mu_j$, cov. matrices $\Sigma_j$, dim. $D$
  - $p(x|\theta) = \sum_{j=1}^{K} \pi_j p(x|\theta_j)$
  - Blob $j$ is selected with probability $\pi_j$
  - The likelihood of observing $x$ is a weighted mixture of Gaussians

$$p(x|\theta) = \sum_{j=1}^{K} \pi_j p(x|\theta_j)$$
Expectation Maximization (EM)

- **Goal**
  - Find blob parameters $\theta$ that maximize the likelihood function:
  
  $$p(\text{data}|\theta) = \prod_{n=1}^{N} p(x_n|\theta)$$

- **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 Algorithm

- **E-step**: softly assign samples to mixture components
  
  $$\gamma_j(x_n) = \frac{\pi_j N(x_n; \mu_j, \Sigma_j)}{\sum_{k=1}^{K} \pi_k N(x_n; \mu_k, \Sigma_k)}$$

- **M-step**: re-estimate the parameters (separately for each mixture component) based on the soft assignments
  
  $$\hat{N}_j = \sum_{n=1}^{N} \gamma_j(x_n)$$
  
  $$\hat{\mu}_j = \frac{1}{\hat{N}_j} \sum_{n=1}^{N} \gamma_j(x_n) x_n$$
  
  $$\hat{\Sigma}_j = \frac{1}{\hat{N}_j} \sum_{n=1}^{N} \gamma_j(x_n)(x_n - \hat{\mu}_j)(x_n - \hat{\mu}_j)^T$$

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
  - ...

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
  - Mixture 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: \( \sum_{x \in W} xH(x) \)
  3. Shift the search window to the mean
  4. Repeat Step 2 until convergence

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

Slide credit: Steve Seitz

Slide credit: 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 vector

Mean-Shift

Region of interest
Center of mass
Mean Shift vector

Real Modality Analysis

Tessellate the space with windows
Run the procedure in parallel

Real Modality Analysis

The blue data points were traversed by the windows towards the mode.
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

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

Mean-Shift Segmentation Results

More Results

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.

B. Leibe

### Speedups

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

B. Leibe

### 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

B. Leibe

### 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 can be found in Chapter 14 of
- More on the EM algorithm can be found in Chapter 16.1.2.