Lecture 5: Clustering and Segmentation – Part 1

Professor Fei-Fei Li

Stanford Vision Lab

What we will learn today

- Segmentation and grouping
  - Gestalt principles
- Segmentation as clustering
  - K-means
  - Feature space
- Probabilistic clustering (Problem Set 1 (Q3))
  - Mixture of Gaussians, EM

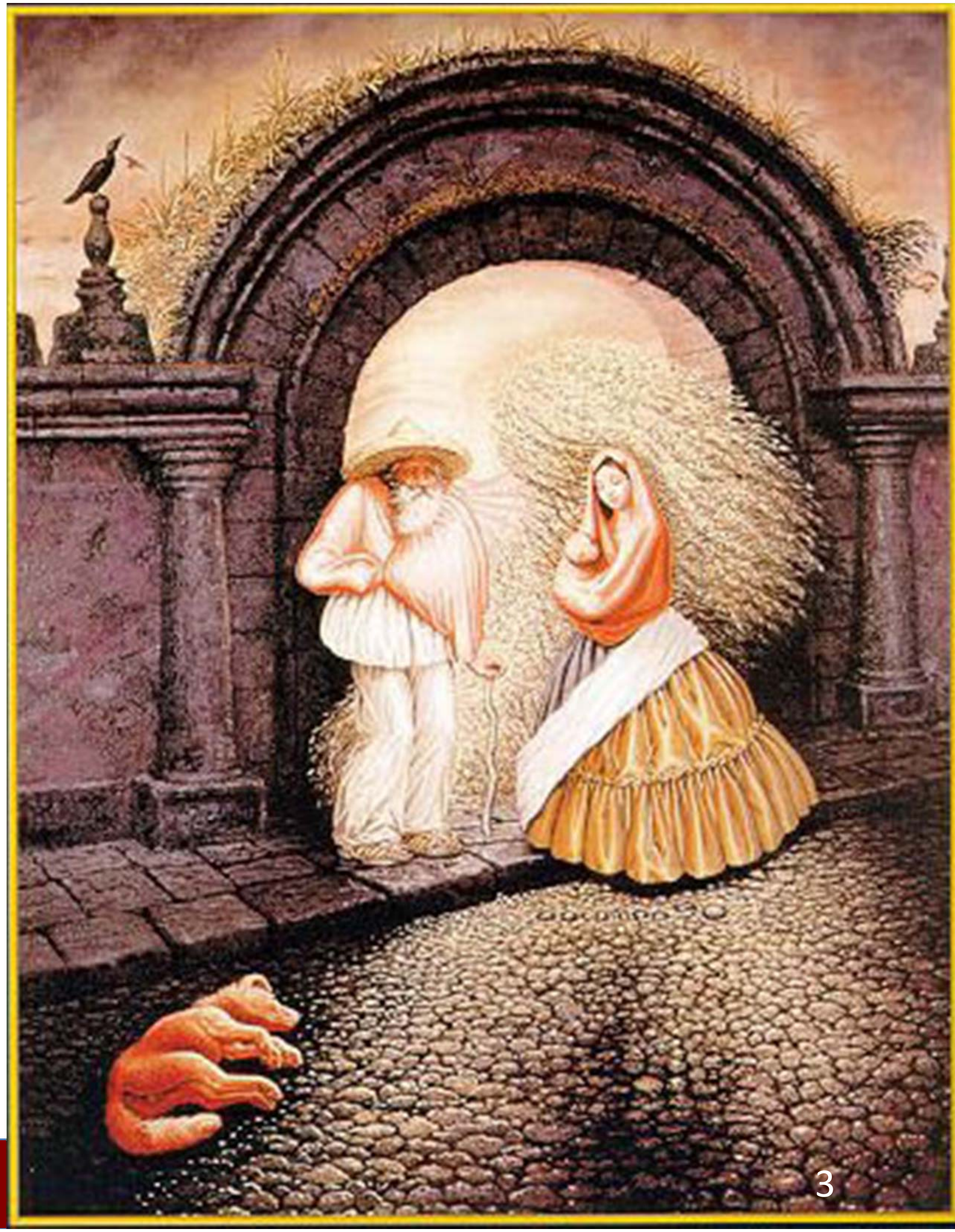
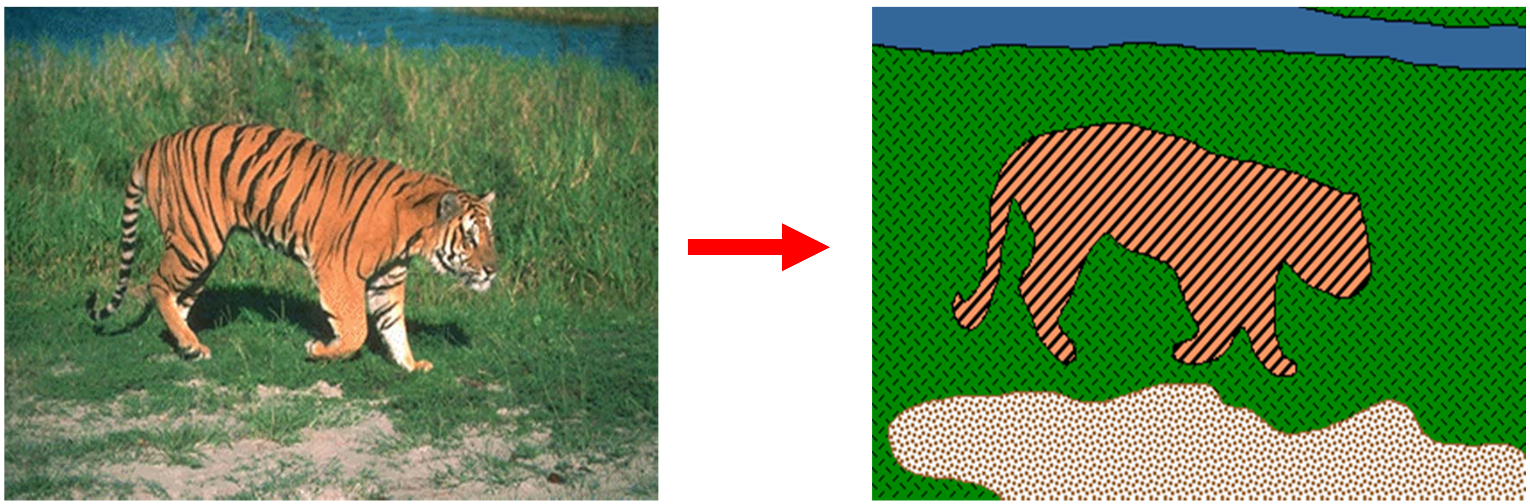


Image Segmentation

- Goal: identify groups of pixels that go together

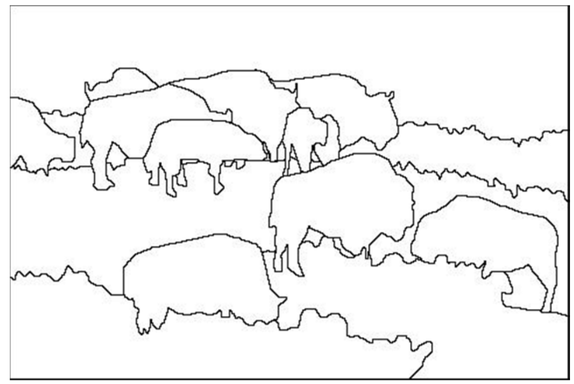


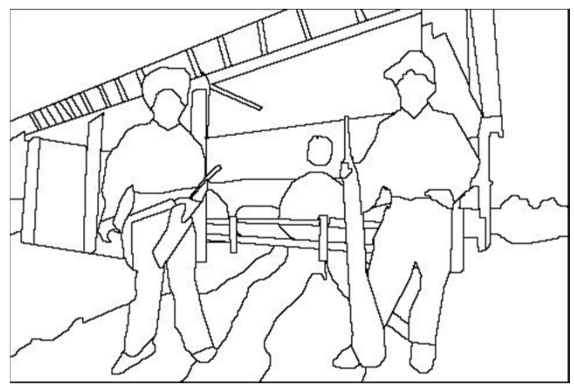
Slide credit: Steve Seitz, Kristen Grauman

The Goals of Segmentation

- Separate image into coherent "objects"

Image — Human segmentation





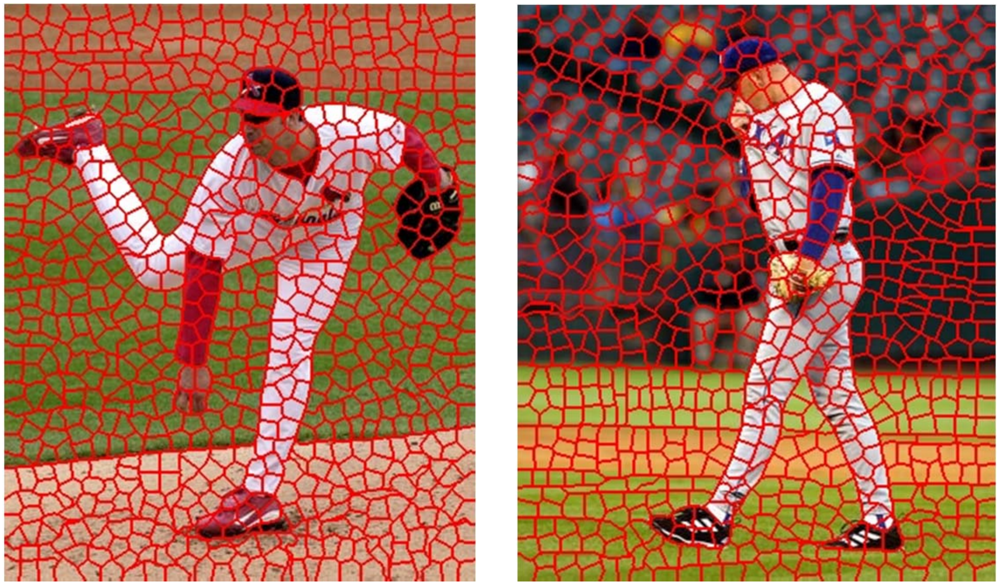
Slide credit: Svetlana Lazebnik

The Goals of Segmentation

- Separate image into coherent “objects”
- Group together similar-looking pixels for efficiency of further processing

“over-segmentation”

“superpixels”



X. Ren and J. Malik. Learning a classification model for segmentation. ICCV 2003.

Slide credit: Svetlana Lazebnik

Segmentation

- Compact representation for image data in terms of a set of components
- Components share "common" visual properties
- Properties can be defined at different level of abstractions

General ideas

This lecture (#5)

- Tokens "features"
  - whatever we need to group (pixels, points, surface elements, etc., etc.)
- Bottom up segmentation
  - tokens belong together because they are locally coherent
- Top down segmentation
  - tokens belong together because they lie on the same visual entity (object, scene…)

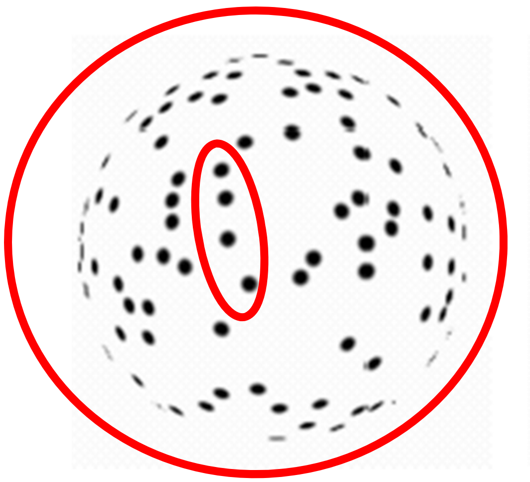
> These two are not mutually exclusive

What is Segmentation?

- **Clustering image elements that “belong together”**
  - **Partitioning**
    - Divide into regions/sequences with coherent internal properties
  - **Grouping**
    - Identify sets of coherent tokens in image

Slide credit: Christopher Rasmussen

What is Segmentation?

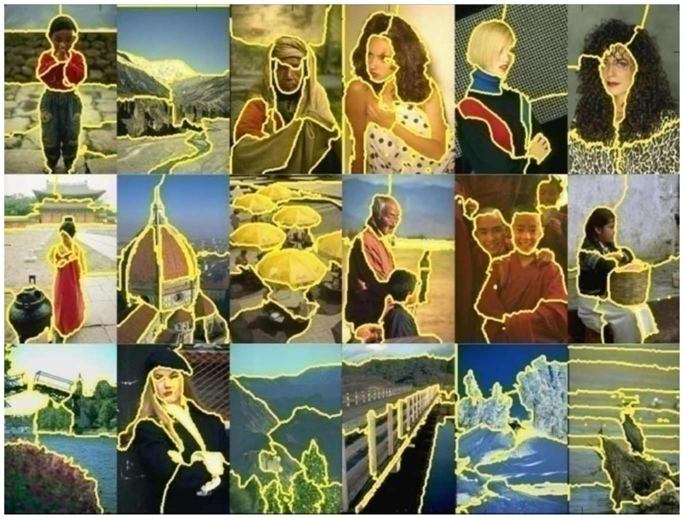


Why do these tokens belong together?

Basic ideas of grouping in human vision

- Gestalt properties
- Figure-ground discrimination

Examples of Grouping in Vision


Determining image regions

Grouping video frames into shots

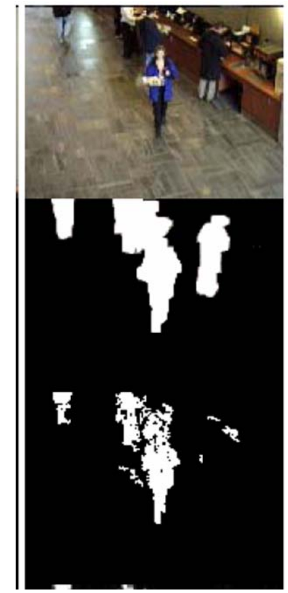
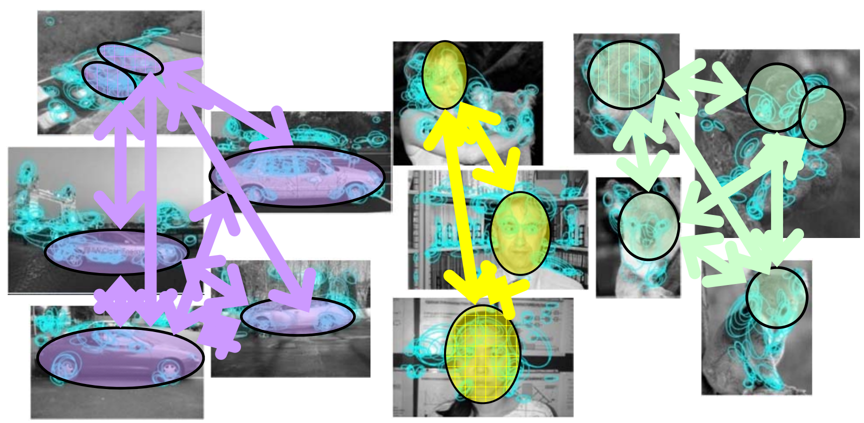


Figure-ground

What things should be grouped?

What cues indicate groups?

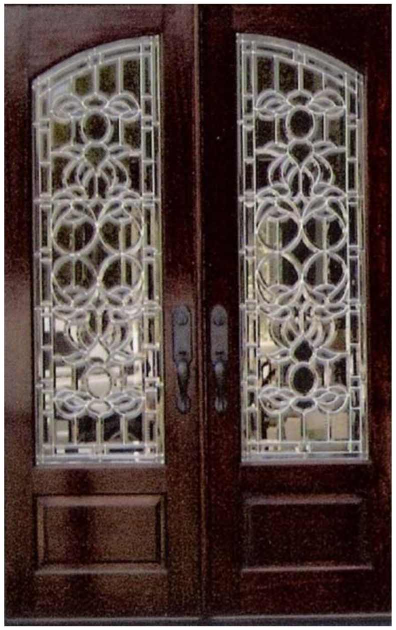

Object-level grouping

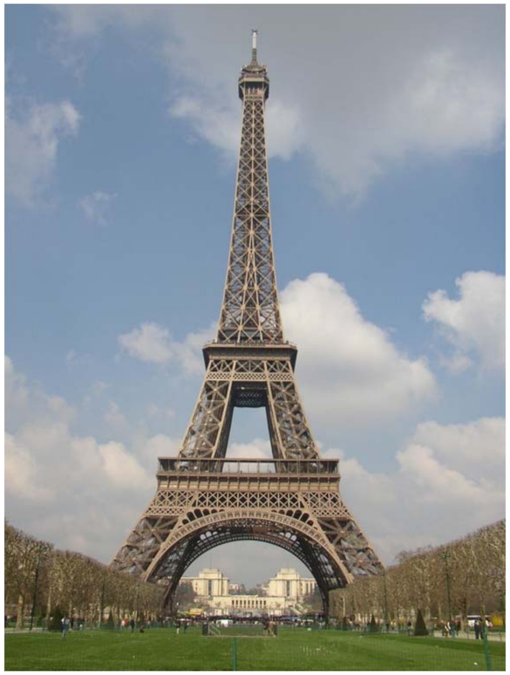
Slide credit: Kristen Grauman

Similarity

Slide credit: Kristen Grauman

Symmetry





Slide credit: Kristen Grauman

Common Fate

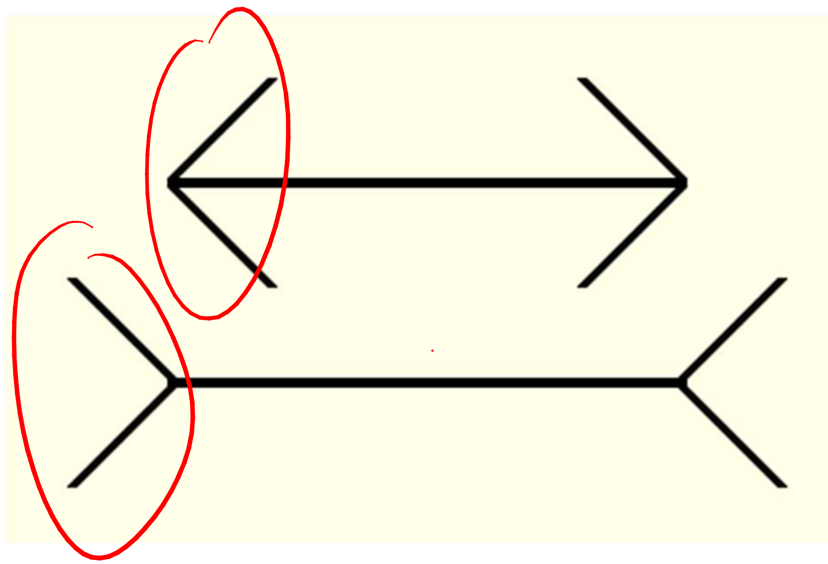
Image credit: Arthus-Bertrand (via F. Durand)

Slide credit: Kristen Grauman

Proximity

Slide credit: Kristen Grauman

Muller-Lyer Illusion

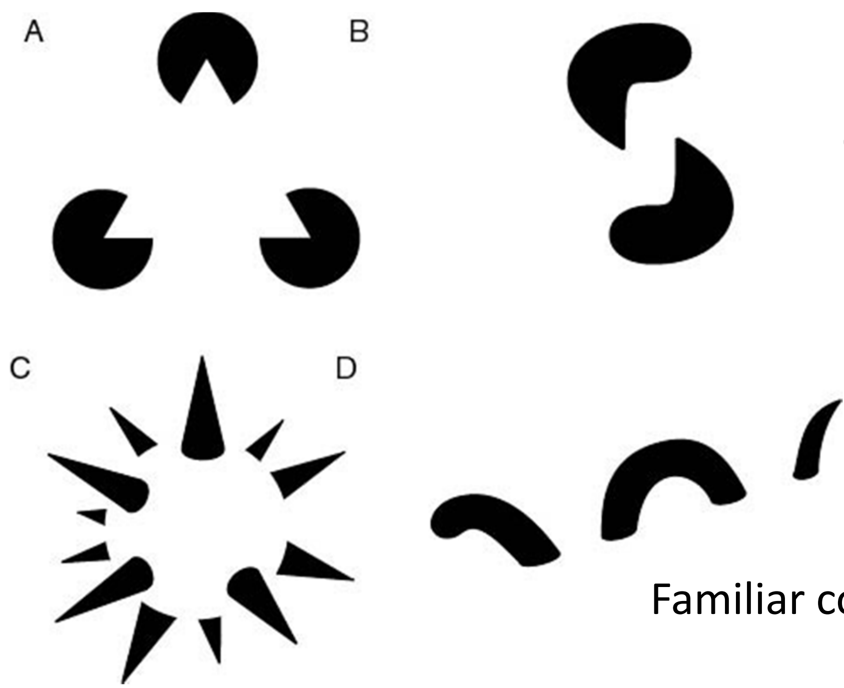


- Gestalt principle: grouping is key to visual perception.

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"

Illusory/subjective contours



Occlusion

Familiar configuration

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

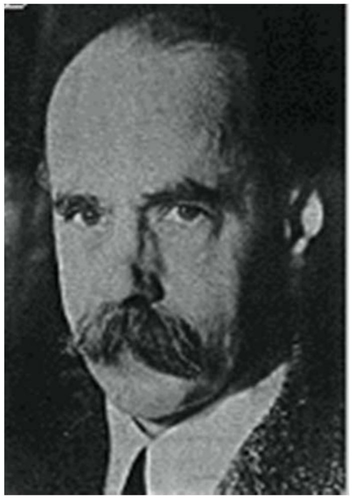
Slide credit: Svetlana Lazebnik

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)

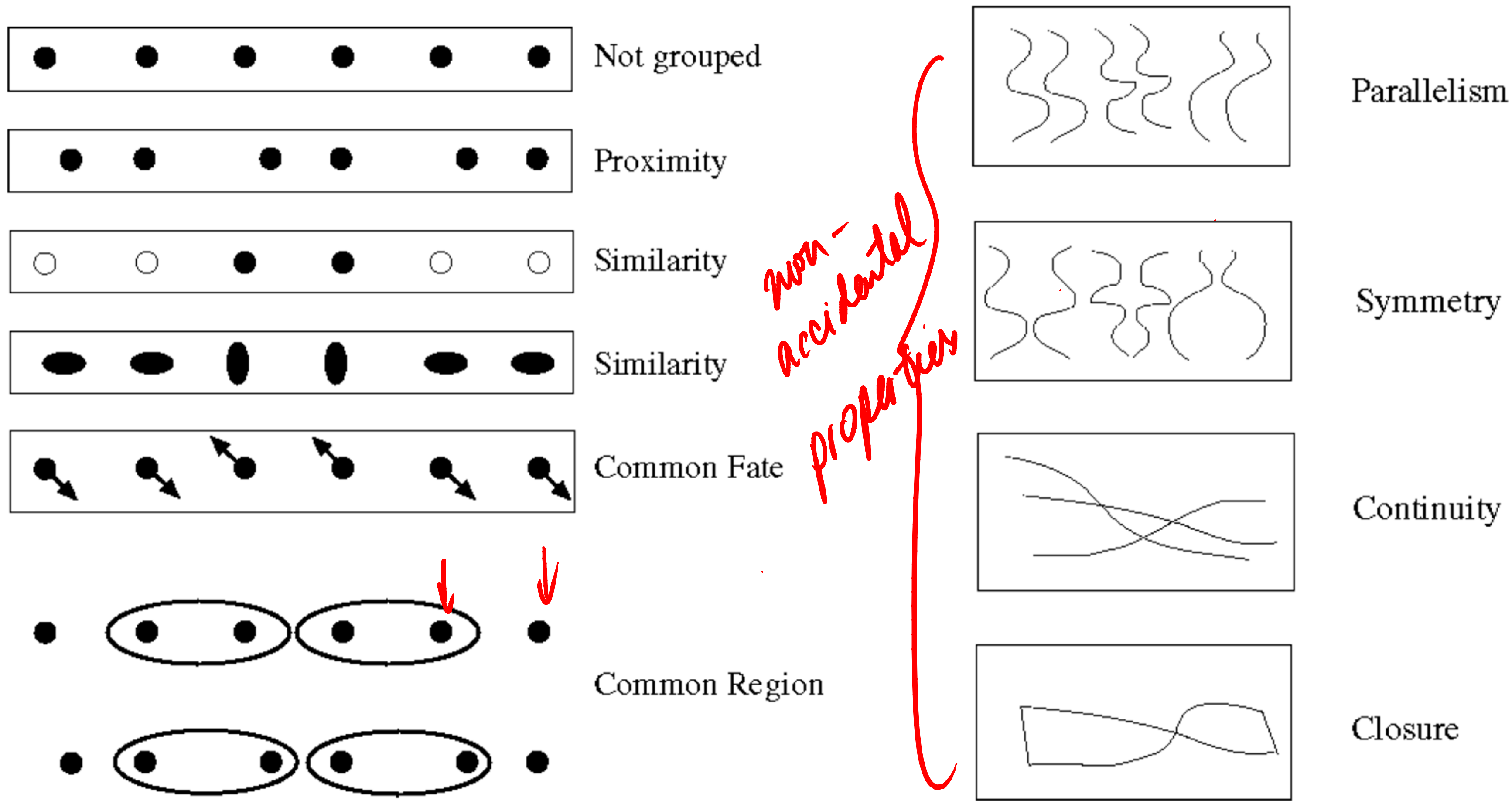
"I stand at the window and see a house, trees, sky. Theoretically I might say there were 327 brightnesses and nuances of colour. Do I have "327"? No. I have sky, house, and trees."

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

Gestalt Factors

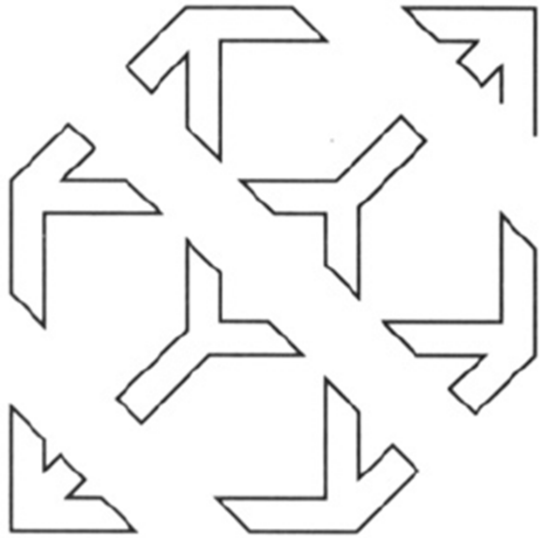


non-accidental properties

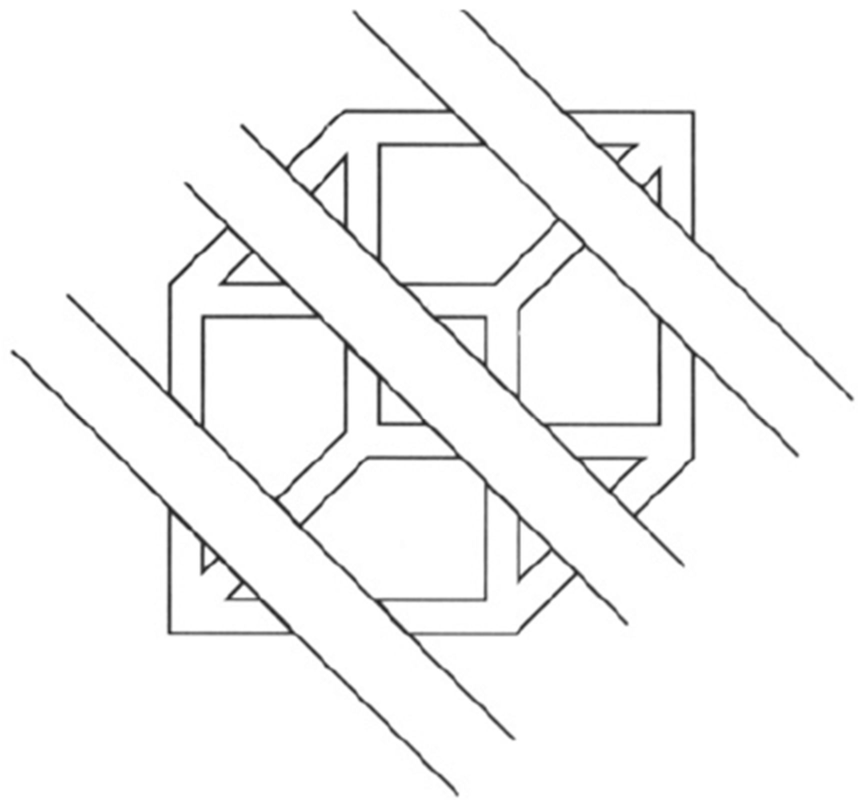
Image source: Forsyth & Ponce

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

Continuity through Occlusion Cues



Continuity through Occlusion Cues



Continuity, explanation by occlusion

Continuity through Occlusion Cues

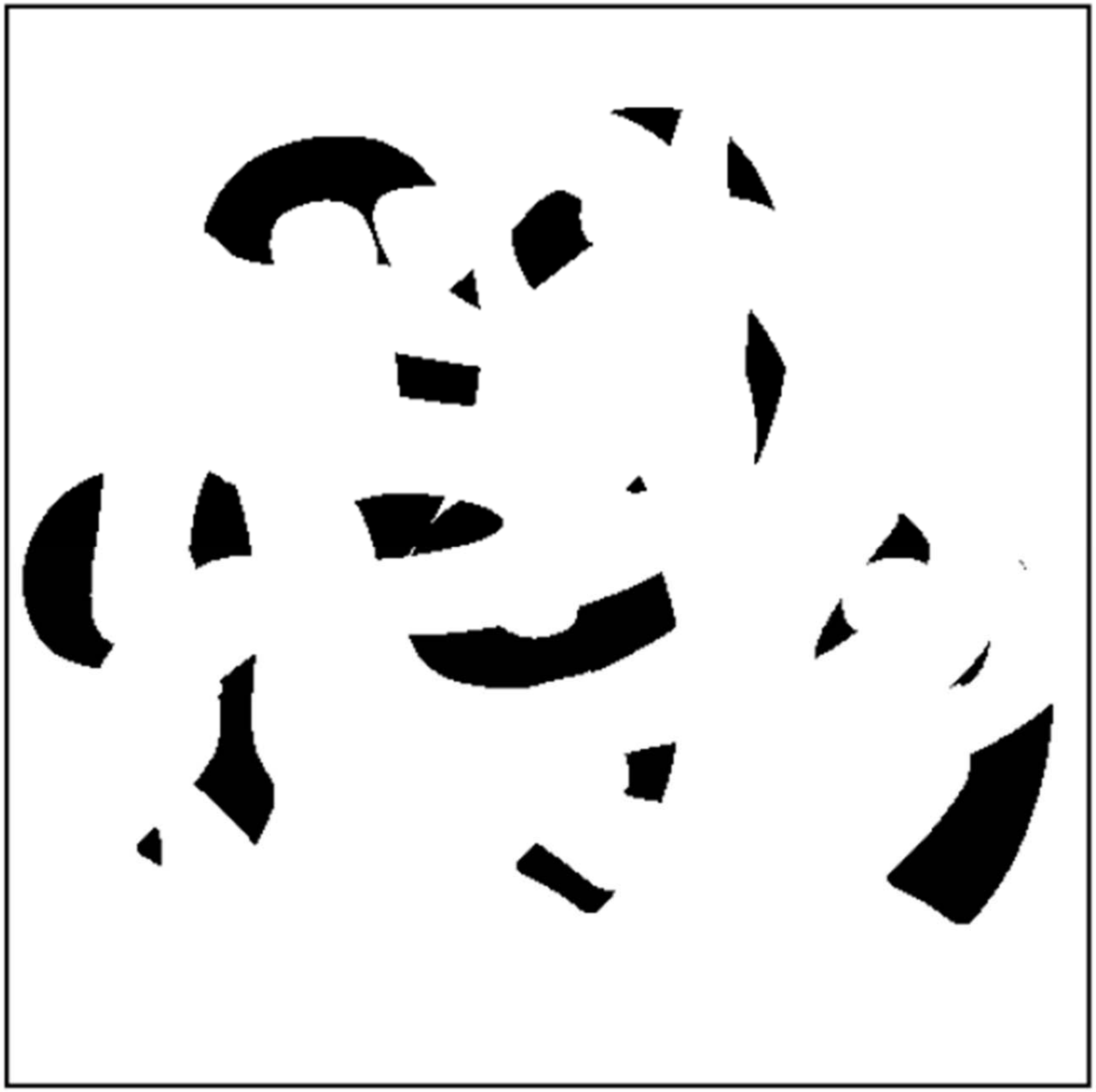
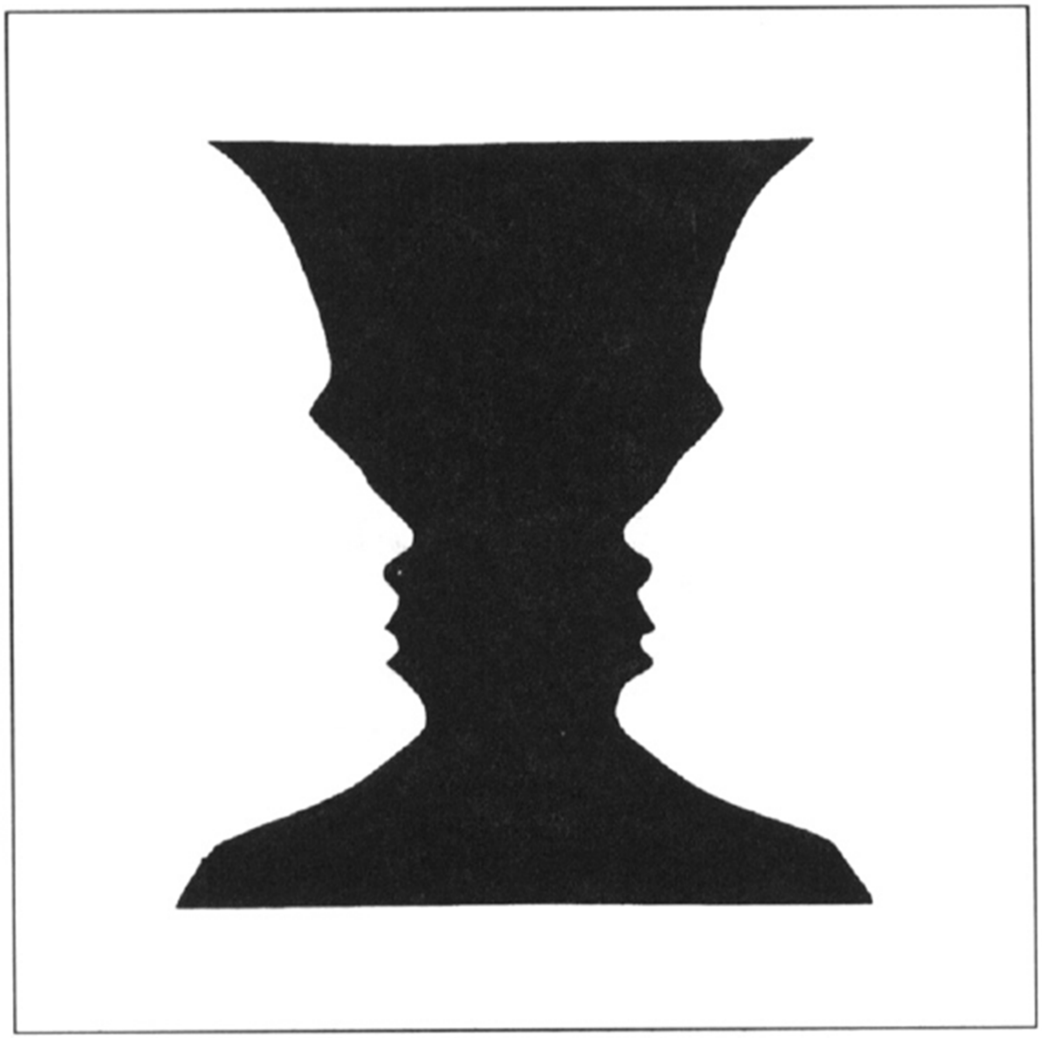


Image source: Forsyth & Ponce

Continuity through Occlusion Cues

Image source: Forsyth & Ponce

Figure-Ground Discrimination

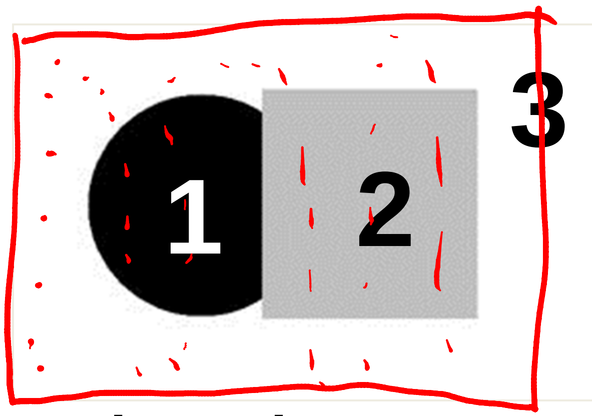


The Ultimate Gestalt?

What we will learn today

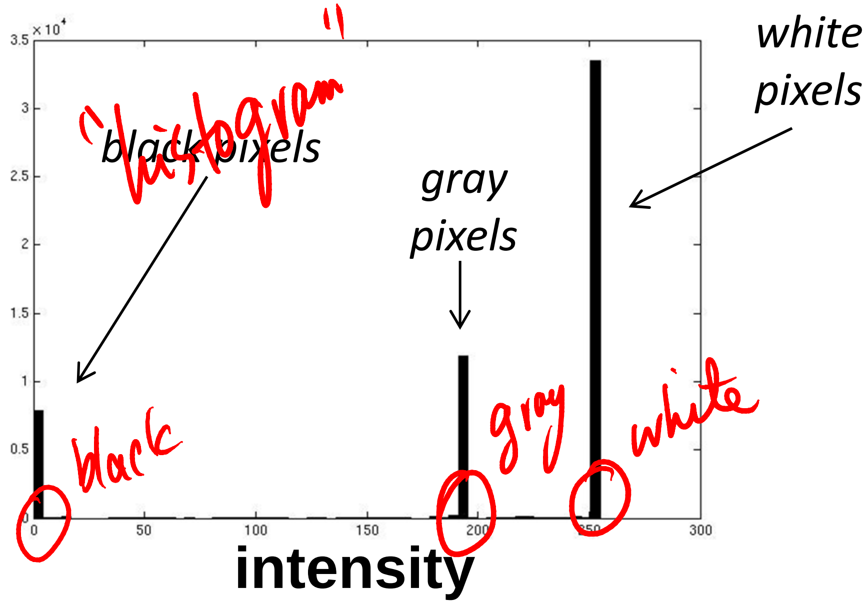
- Segmentation and grouping
  - Gestalt principles
- Segmentation as clustering
  - K-means
  - Feature space
- Probabilistic clustering
  - Mixture of Gaussians, EM
- Model-free clustering
  - Mean-shift

Image Segmentation: Toy Example



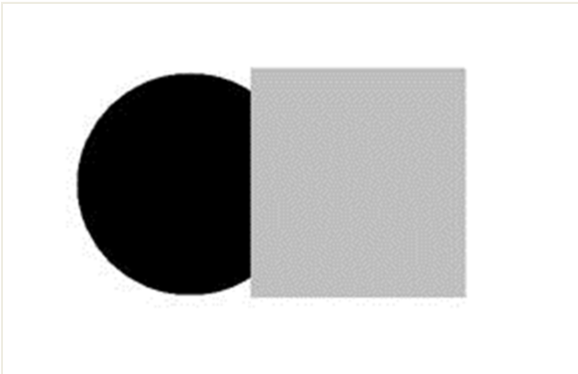
input image

%

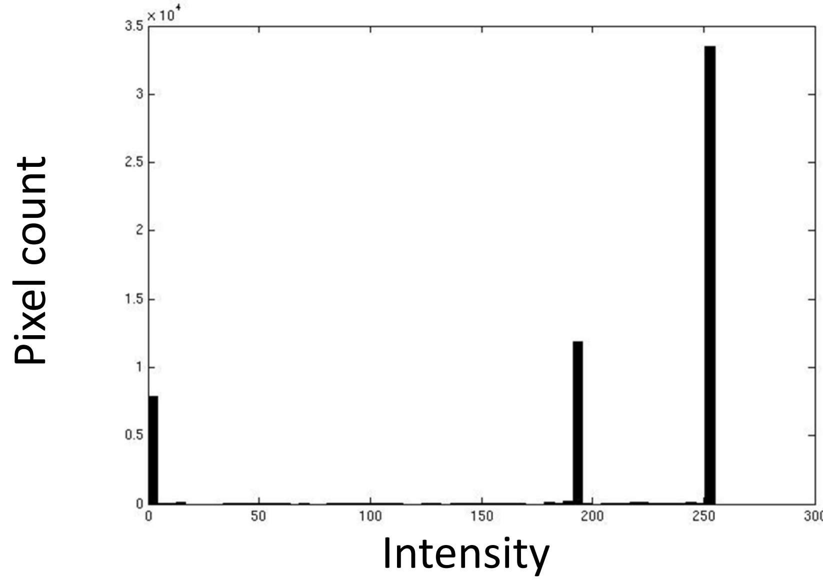


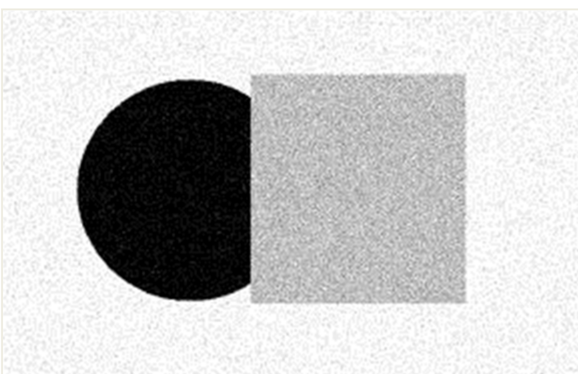
- These intensities define the three groups.
- We could label every pixel in the image according to which of these primary intensities it is.
  - i.e., segment the image based on the intensity feature.
- What if the image isn't quite so simple?

Slide credit: Kristen Grauman

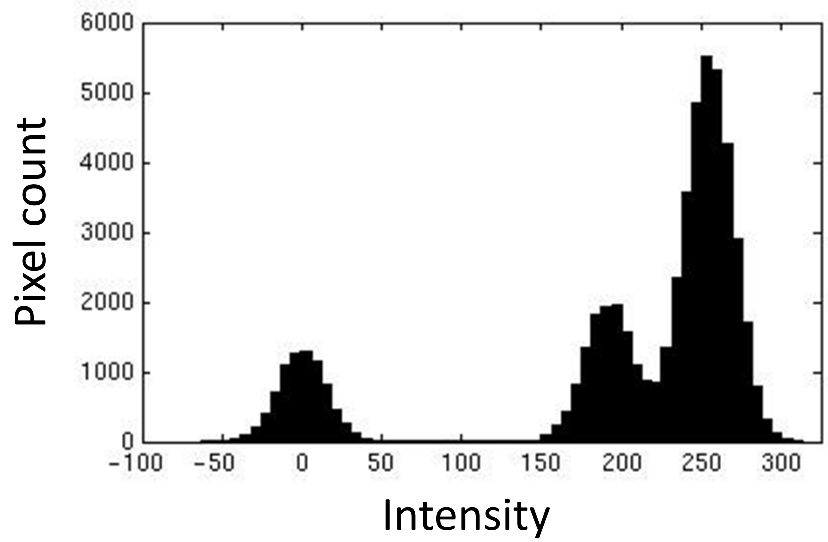


Input image

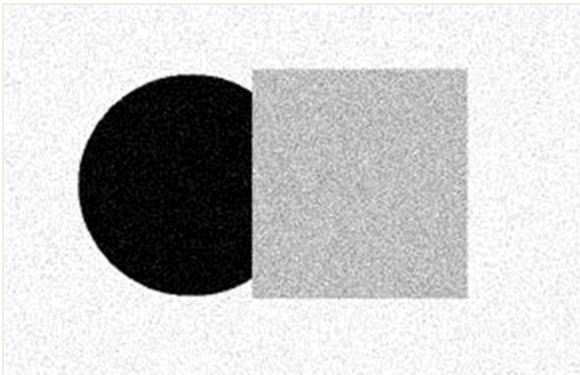




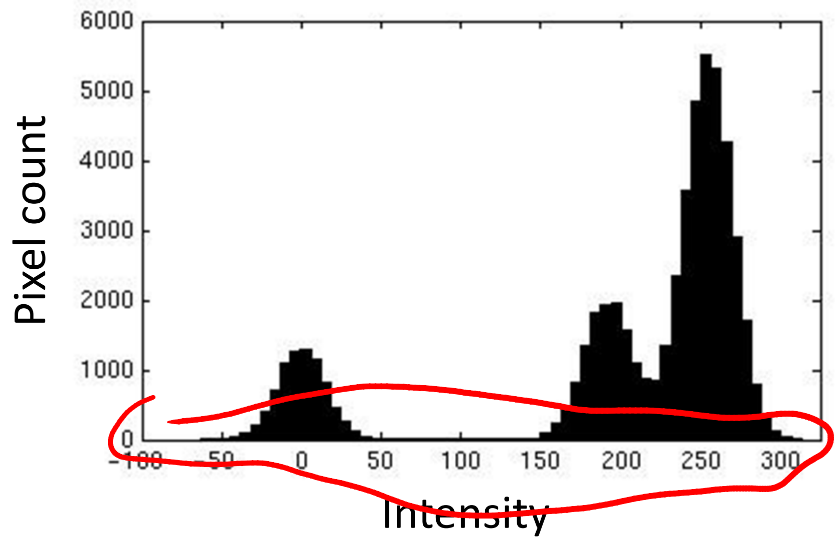
Input image



Slide credit: Kristen Grauman

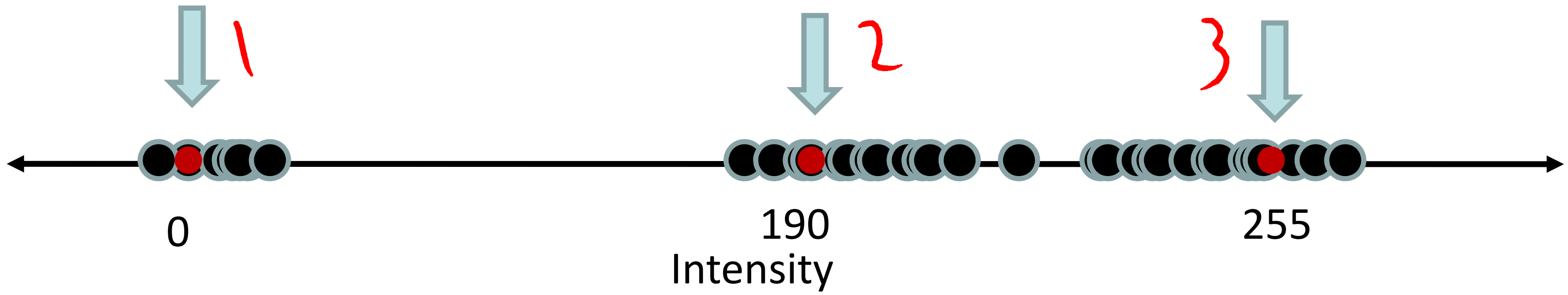


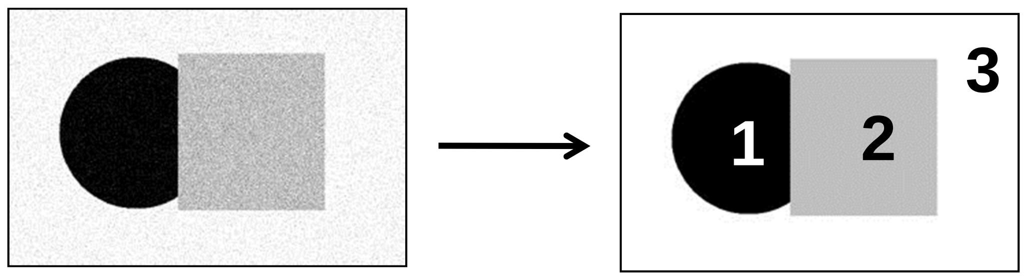
Input image



- Now how to determine the three main intensities that define our groups?
- We need to cluster.

Slide credit: Kristen Grauman





- Goal: choose three “centers” as the representative intensities, and label every pixel according to which of these centers it is nearest to.
- Best cluster centers are those that minimize Sum of Square Distance (SSD) between all points and their nearest cluster center c_i:

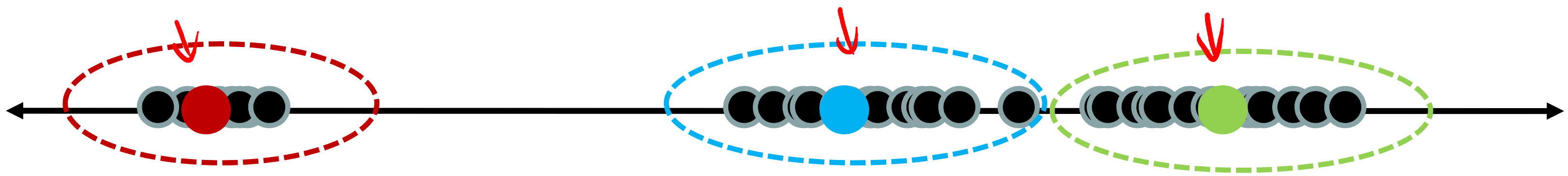
center

$$SSD = \sum_{clusters\ i} \sum_{p \in cluster\ i} \|p - c_i\|^2$$

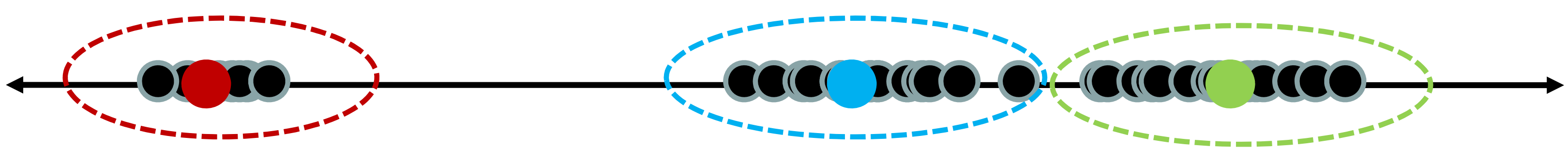
Slide credit: Kristen Grauman

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.



Slide credit: Kristen Grauman

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:

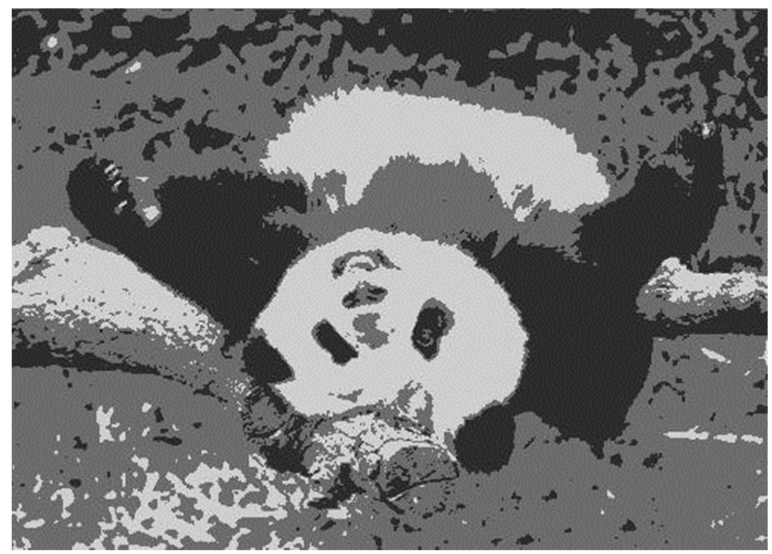
$$SSD = \sum_{clusters\ i} \sum_{p \in cluster\ i} \|p - c_i\|^2$$

Slide credit: Steve Seitz

Segmentation as Clustering

K=2

K=3



```
img_as_col = double(im(:));
cluster_membs = kmeans(img_as_col, K);

labelim = zeros(size(im));
for i=1:k
   inds = find(cluster_membs==i);
   meanval = mean(img_as_column(inds));
   labelim(inds) = meanval;
end
```

Slide credit: Kristen Grauman

K-Means Clustering

- Java demo:

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

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

Arthur & Vassilvitskii 2007

Slide credit: Steve Seitz

Feature Space

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

- Grouping pixels based on intensity similarity

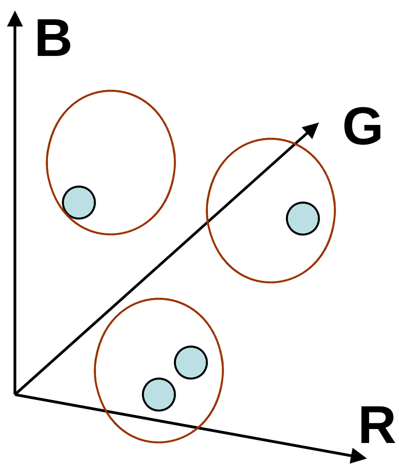
- Feature space: intensity value (1D)

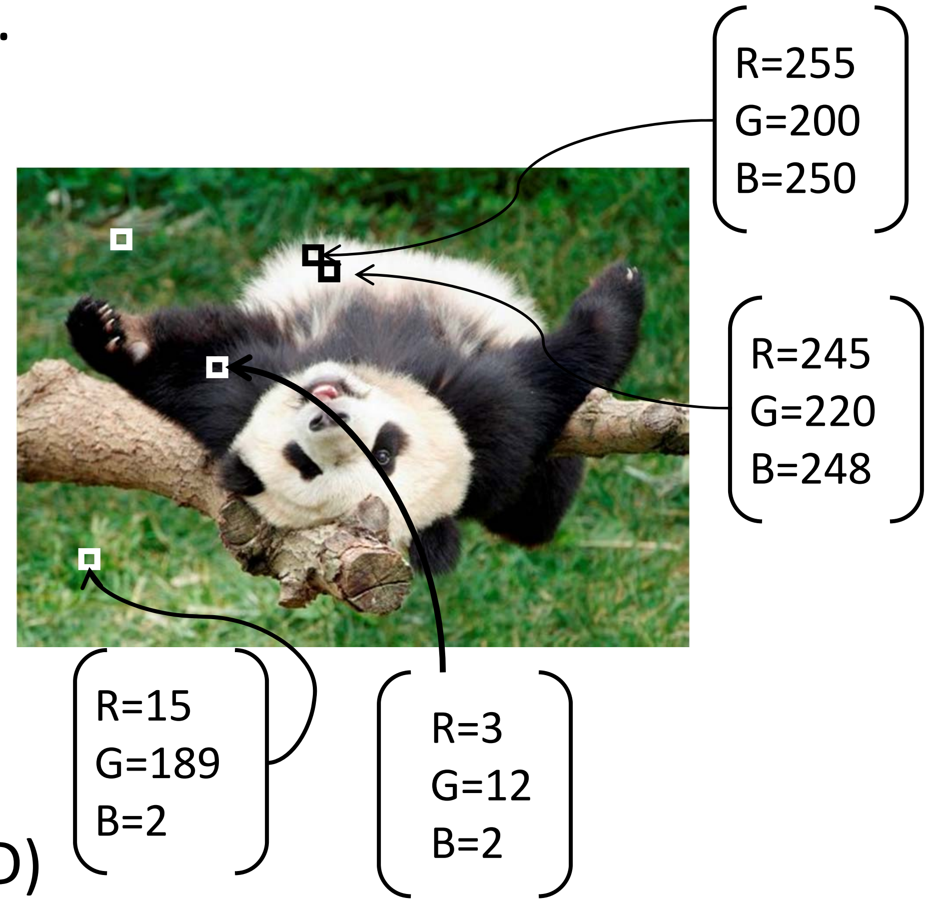
Slide credit: Kristen Grauman

Feature Space

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

- Grouping pixels based on color similarity



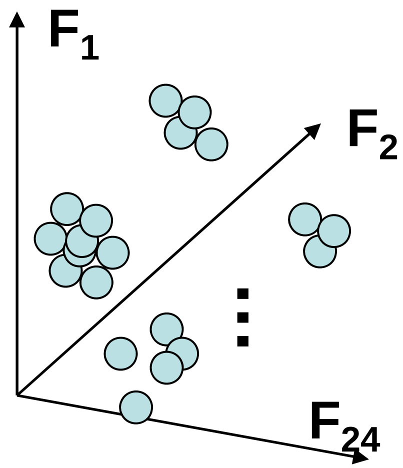


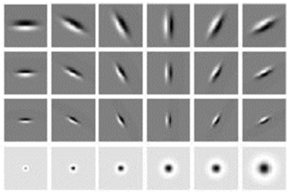
- Feature space: color value (3D)

Slide credit: Kristen Grauman

Feature Space

- Depending on what we choose as the *feature space*, we can group pixels in different ways.
- Grouping pixels based on texture similarity





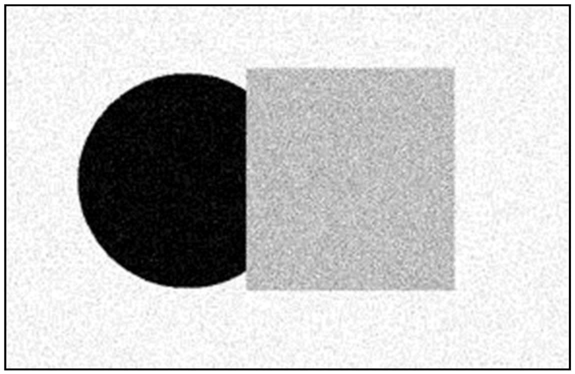
Filter bank of 24 filters

- Feature space: filter bank responses (e.g., 24D)

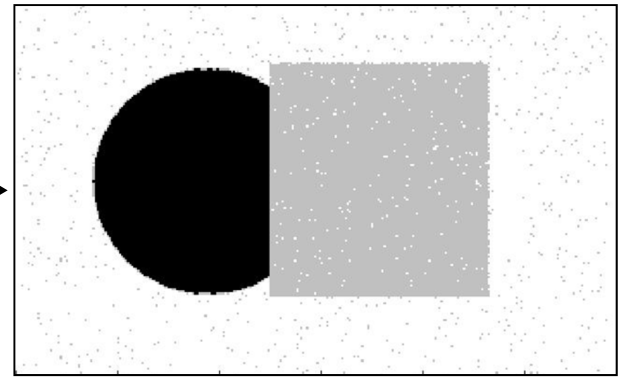
Slide credit: Kristen Grauman

Smoothing Out Cluster Assignments

- Assigning a cluster label per pixel may yield outliers:



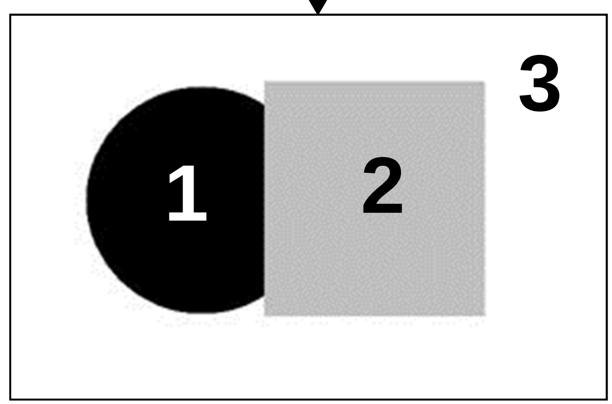
Original



Labeled by cluster center's intensity

?

- How can we ensure they are spatially smooth?

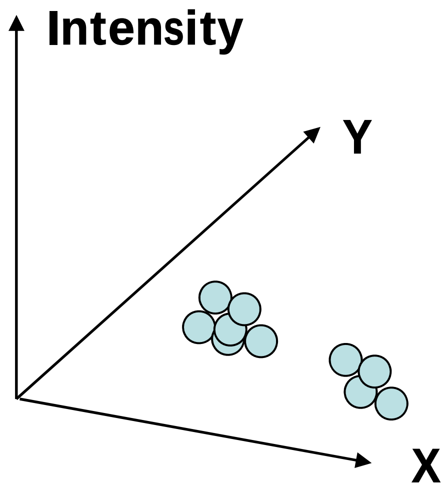


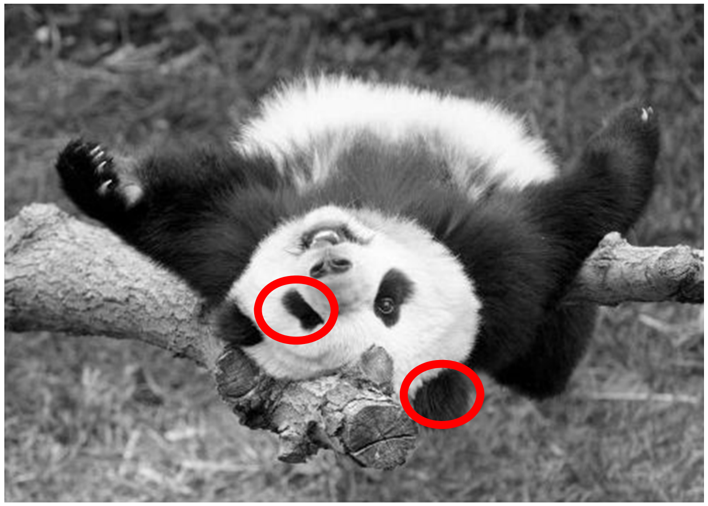
Slide credit: Kristen Grauman

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





⇒ Way to encode both *similarity* and *proximity.*

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

Image

Intensity-based clusters

Color-based clusters

Image source: Forsyth & Ponce

K-Means Clustering Results

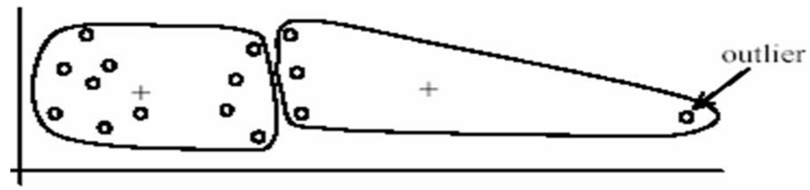
- K-means clustering based on intensity or color is essentially vector quantization of the image attributes
  - Clusters don't have to be spatially coherent
- Clustering based on (r,g,b,x,y) values enforces more spatial coherence

Image source: Forsyth & Ponce

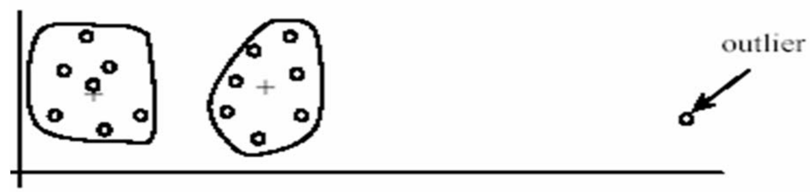
Summary K-Means

- Pros
  - Simple, fast to compute
  - Converges to local minimum of within-cluster squared error

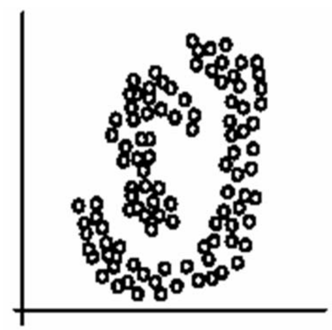
- Cons/issues
  - Setting k?
  - Sensitive to initial centers
  - Sensitive to outliers
  - Detects spherical clusters only
  - Assuming means can be computed

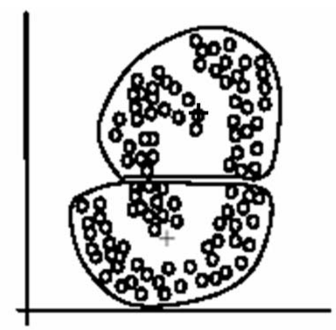


(A): Undesirable clusters



(B): Ideal clusters


(A): Two natural clusters


(B): *k*-means clusters

Slide credit: Kristen Grauman

What we will learn today

- Segmentation and grouping
  - Gestalt principles
- Segmentation as clustering
  - K-means
  - Feature space
- Probabilistic clustering (Problem Set 1 (Q3))
  - Mixture of Gaussians, EM

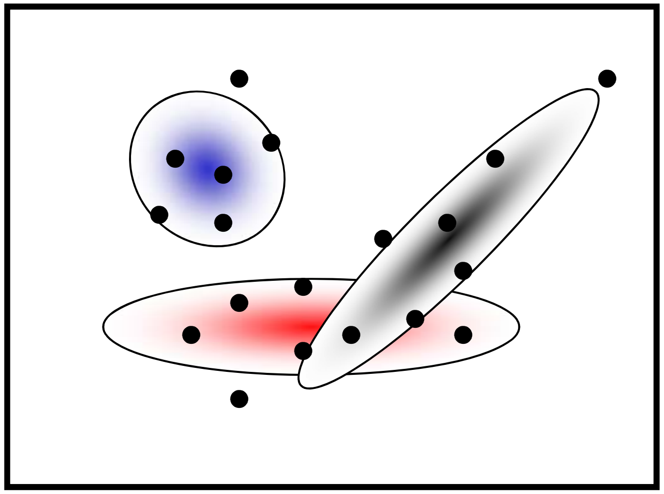
Probabilistic Clustering

- 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 θ

Slide credit: Steve Seitz

Mixture of Gaussians

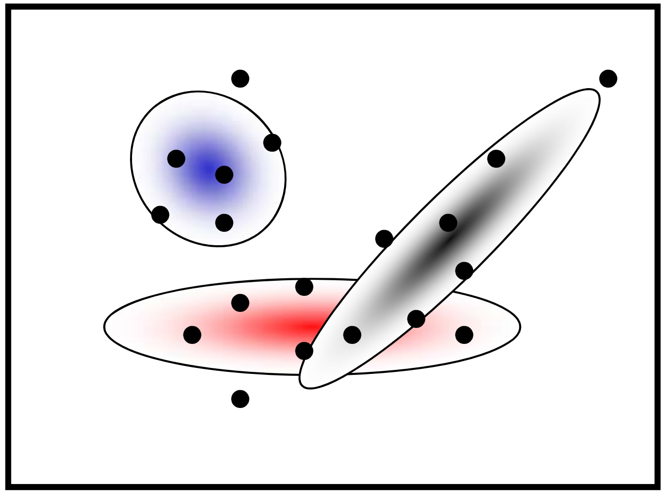


- One generative model is a mixture of Gaussians (MoG)
  - K Gaussian blobs with means μ_b covariance matrices V_b, dimension d
    - Blob *b* defined by: $P(x|\mu_b, V_b) = \frac{1}{\sqrt{(2\pi)^d |V_b|}} e^{-\frac{1}{2}(x-\mu_b)^T V_b^{-1}(x-\mu_b)}$
  - Blob *b* is selected with probability α_b
  - The likelihood of observing x is a weighted mixture of Gaussians

$$P(x|\theta) = \sum_{b=1}^{K} \alpha_b P(x|\theta_b) \;, \quad \theta = [\mu_1, \ldots, \mu_n, V_1, \ldots, V_n]$$

Slide credit: Steve Seitz

Expectation Maximization (EM)



- Goal
  - Find blob parameters θ that maximize the likelihood function:

$$P(data|\theta) = \prod_x P(x|\theta)$$

- 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

Slide credit: Steve Seitz

EM Details

See CS229 Lecture 8!

- E-step
  - Compute probability that point x is in blob *b*, given current guess of θ

$$P(b|x,\mu_b,V_b) = \frac{\alpha_b P(x|\mu_b,V_b)}{\sum_{i=1}^{K}\alpha_i P(x|\mu_i,V_i)}$$

- M-step
  - Compute probability that blob *b* is selected

$$\alpha_b^{new} = \frac{1}{N}\sum_{i=1}^{N} P(b|x_i,\mu_b,V_b)$$

(N data points)

  - Mean of blob *b*

$$\mu_b^{new} = \frac{\sum_{i=1}^{N} x_i P(b|x_i,\mu_b,V_b)}{\sum_{i=1}^{N} P(b|x_i,\mu_b,V_b)}$$

  - Covariance of blob *b*

$$V_b^{new} = \frac{\sum_{i=1}^{N}(x_i-\mu_b^{new})(x_i-\mu_b^{new})^T P(b|x_i,\mu_b,V_b)}{\sum_{i=1}^{N} P(b|x_i,\mu_b,V_b)}$$

Slide credit: Steve Seitz

Applications of EM

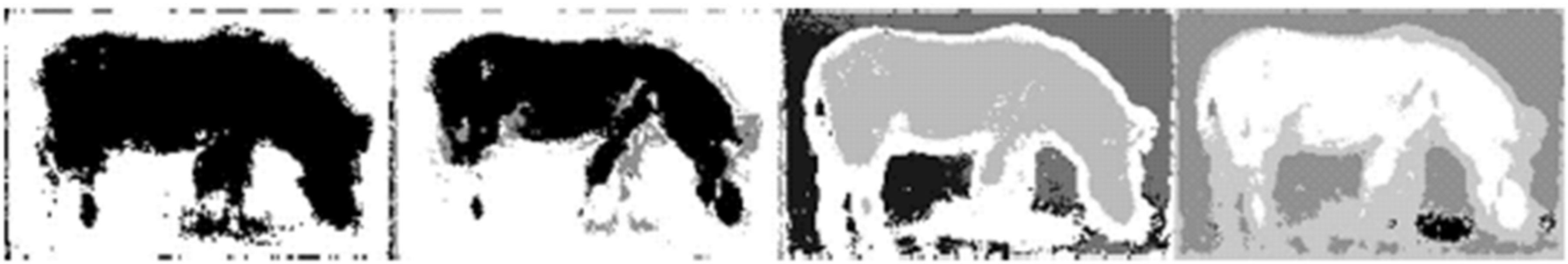
- Turns out this is useful for all sorts of problems
  - Any clustering problem
  - Any model estimation problem
  - Missing data problems
  - Finding outliers
  - Segmentation problems
    - Segmentation based on color
    - Segmentation based on motion
    - Foreground/background separation
  - ...
- EM demo
  - http://lcn.epfl.ch/tutorial/english/gaussian/html/index.html

Slide credit: Steve Seitz

Segmentation with EM

Original image

EM segmentation results



k=2 **k=3** **k=4** **k=5**

Image source: Serge Belongie

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
  - 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

What we have learned today

- Segmentation and grouping
  - Gestalt principles
- Segmentation as clustering
  - K-means
  - Feature space
- Probabilistic clustering (Problem Set 1 (Q3))
  - Mixture of Gaussians, EM