Grouping and Image Segmentation

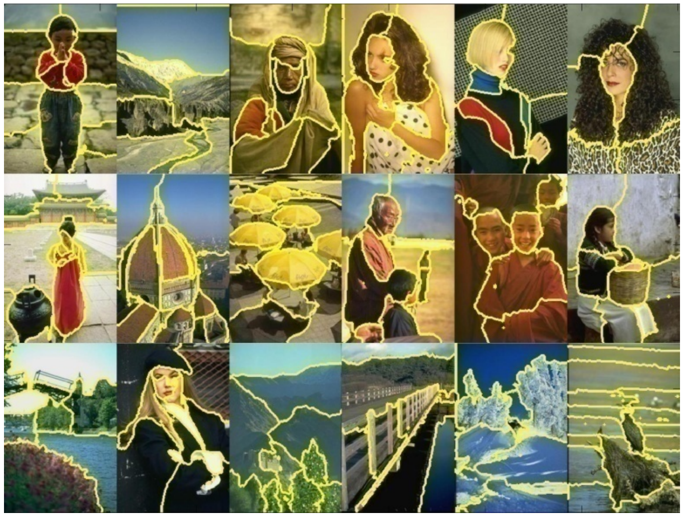
Slides adapted from James Hays, Kristen Grauman, Robert Pless, Khurram Hassan-Shafique, Marc Pollefeys

Fitting & Grouping

- Last Class: Fitting
  - Find model that best represents features
  - Find features that best fit a model

- Today: Grouping in Vision
  - Gather features that belong together
  - Obtain an intermediate representation that compactly describes key image or video parts

Examples of grouping in vision



[Figure by J. Shi]

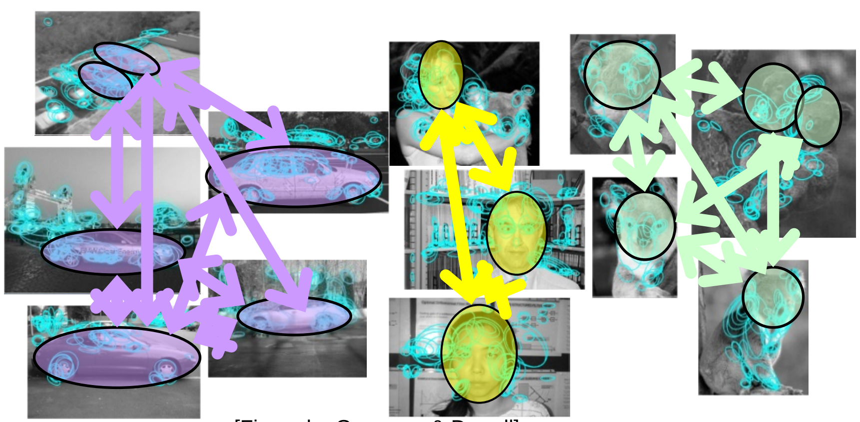
Determine image regions

[http://poseidon.csd.auth.gr/LAB_RESEARCH/Latest/imgs/SpeakDepVidIndex_img2.jpg]

Group video frames into shots

[Figure by Wang & Suter]

Figure-ground



[Figure by Grauman & Darrell]

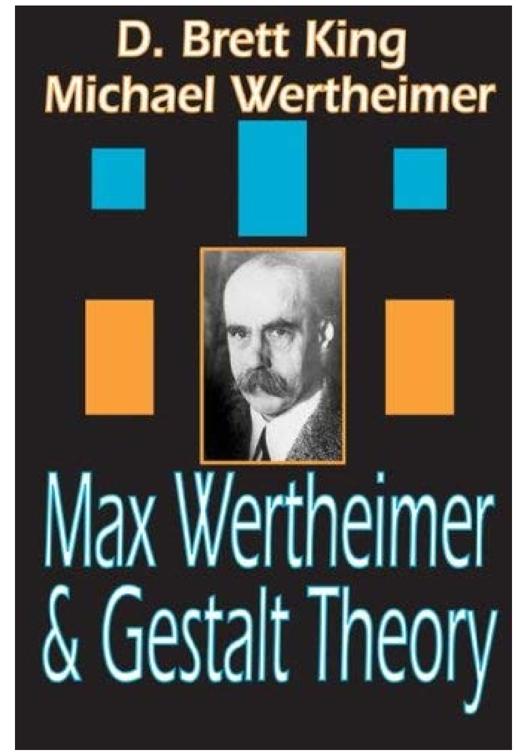
Object-level grouping

What things should be grouped?

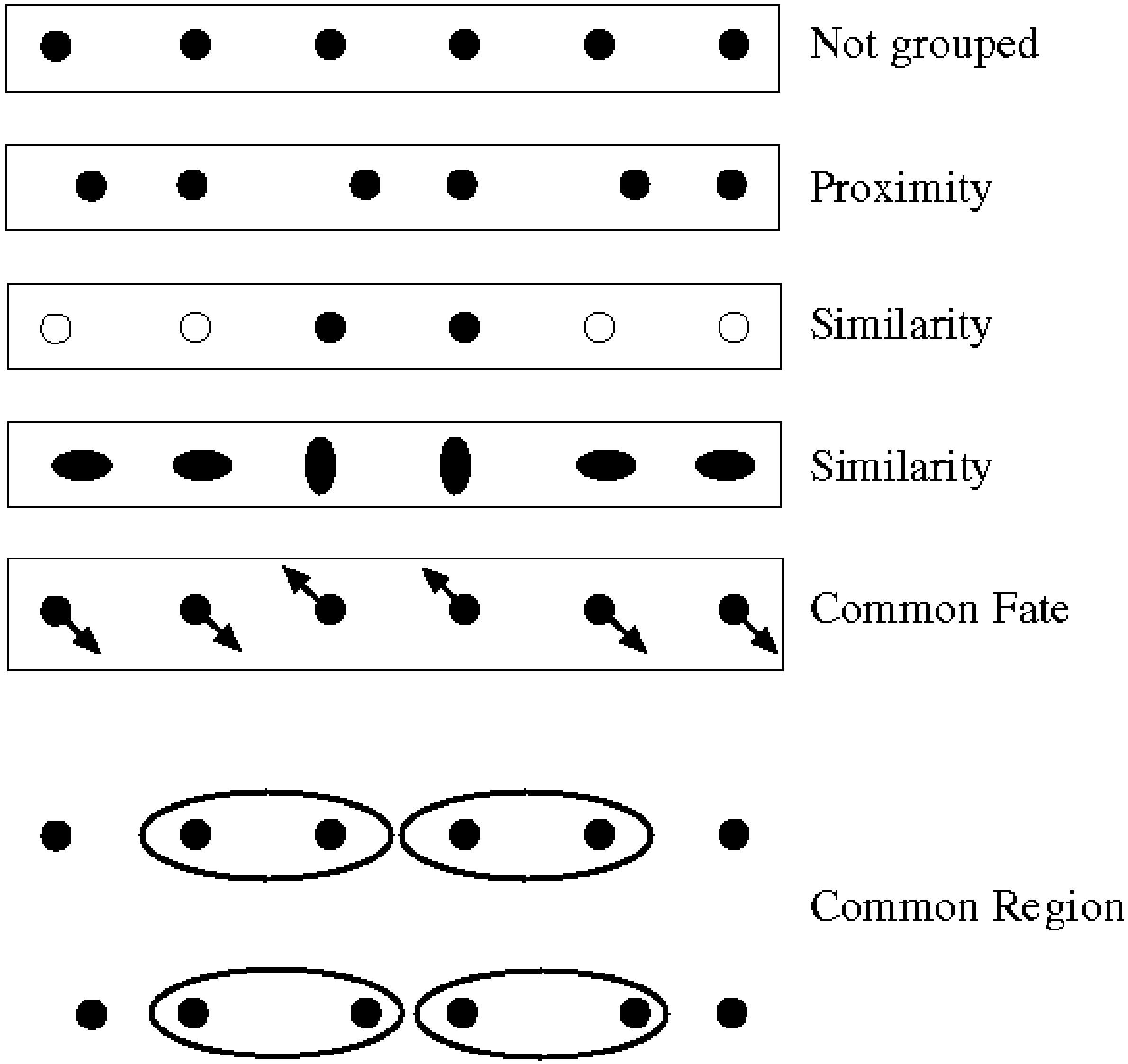
What cues indicate groups?

Gestalt psychology or **Gestaltism**

- German: *Gestalt* - "form" or "whole”
- Berlin School, early 20th century
  – Kurt Koffka, Max Wertheimer, and Wolfgang Köhler
- 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)



Principles of perceptual organization

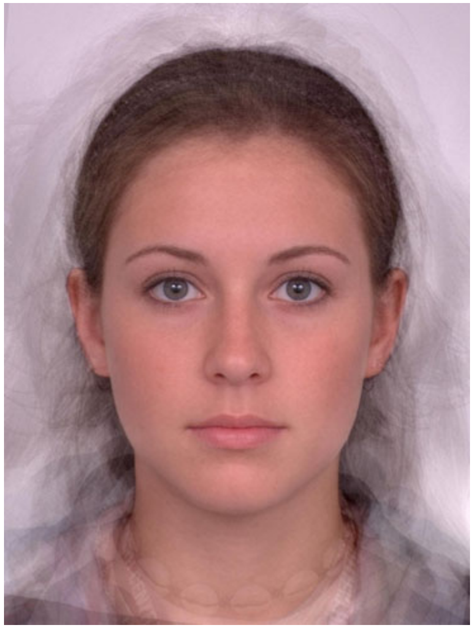


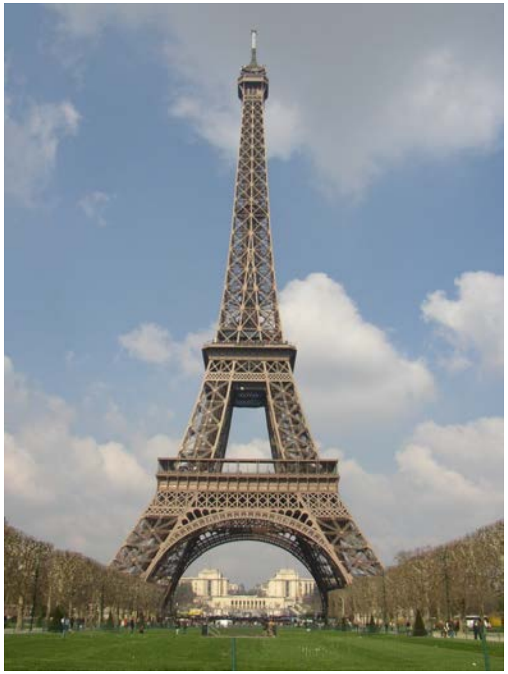
From Steve Lehar: The Constructive Aspect of Visual Perception

Similarity

http://chicagoist.com/attachments/chicagoist_alicia/GEESE.jpg, http://wwwdelivery.superstock.com/WI/223/1532/PreviewComp/SuperStock_1532R-0831.jpg

Symmetry





http://seedmagazine.com/news/2006/10/beauty_is_in_the_processingtim.php

Common fate

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

Proximity

http://www.capital.edu/Resources/Images/outside6_035.jpg

Slide: Kristin Grauman

Gestalt cues

- Good intuition and basic principles for grouping
- Basis for many ideas in segmentation and occlusion reasoning
- Some (e.g., symmetry) are difficult to implement in practice

Image Segmentation

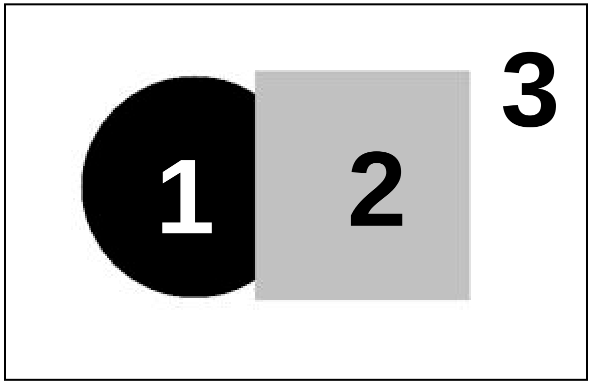
- The process of partitioning an image into multiple regions
  - Sets of pixels

- Goal is to simplify the representation of an image
  - Makes it more meaningful and easier to analyze

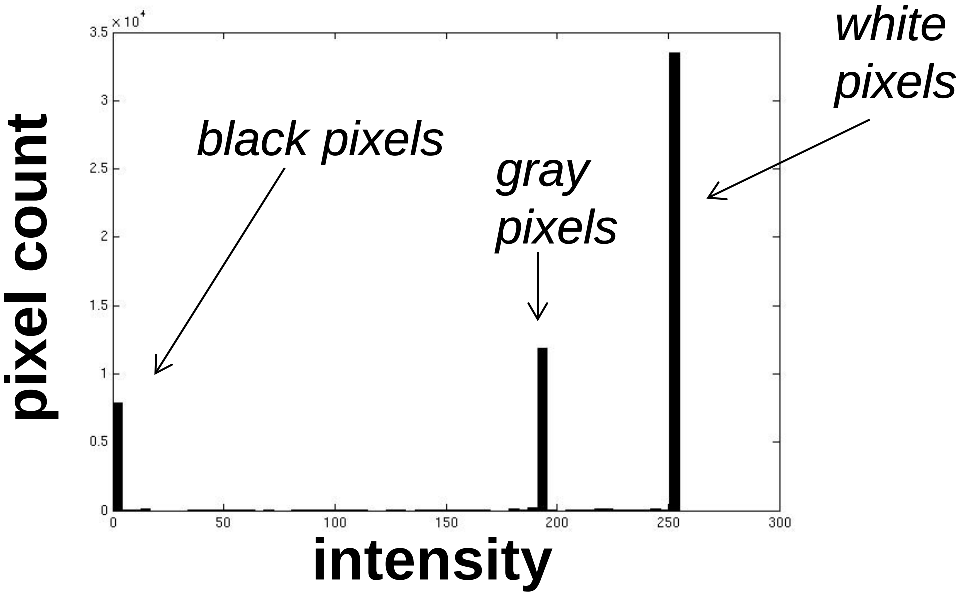
- Helpful for detecting objects and boundaries

http://www.seas.upenn.edu/~timothee/papers/images/imageResult.jpg

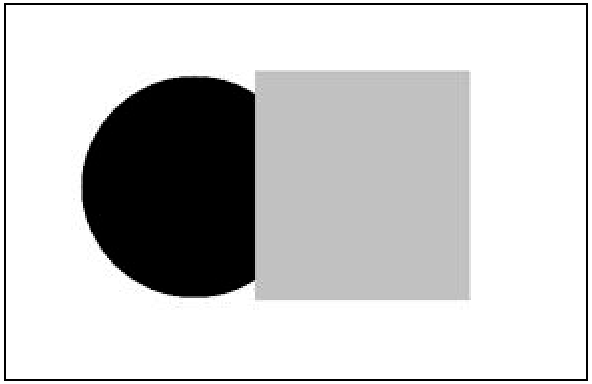
Image Segmentation: Toy Example



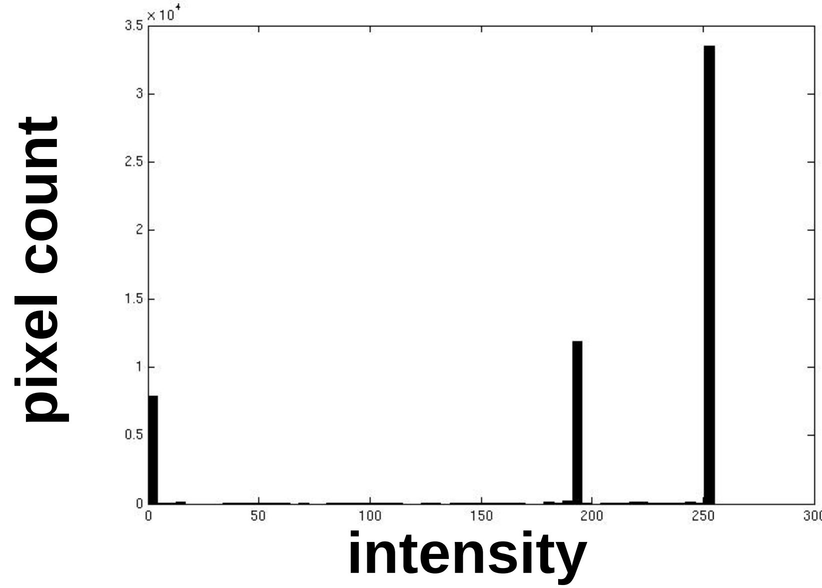
input image

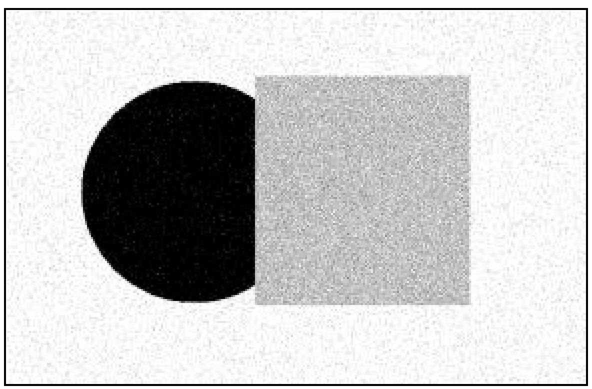


- These intensities define the three groups.
- We could label every pixel in the image according to which of these primary intensities it is.
  - i.e., *segment* the image based on the intensity feature.
- What if the image isn't quite so simple?

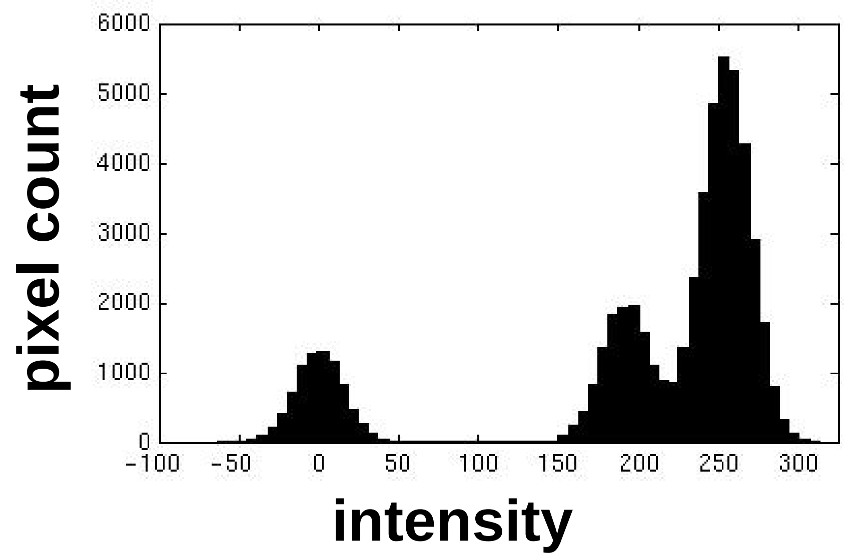


input image





input image



- How do we determine the 3 main intensities that define our groups?

Segmentation as Clustering

- Cluster together pixels that belong together
- Agglomerative clustering
  - attach pixel to cluster it is closest to
  - repeat
- Divisive clustering
  - split cluster along best boundary
  - repeat

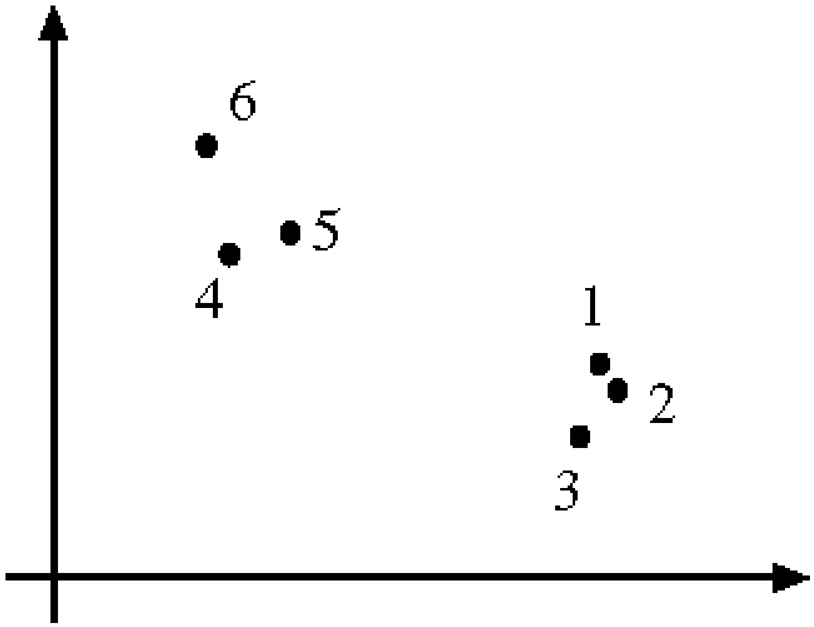
Simple clustering algorithms

Algorithm 15.3: Agglomerative clustering, or clustering by merging

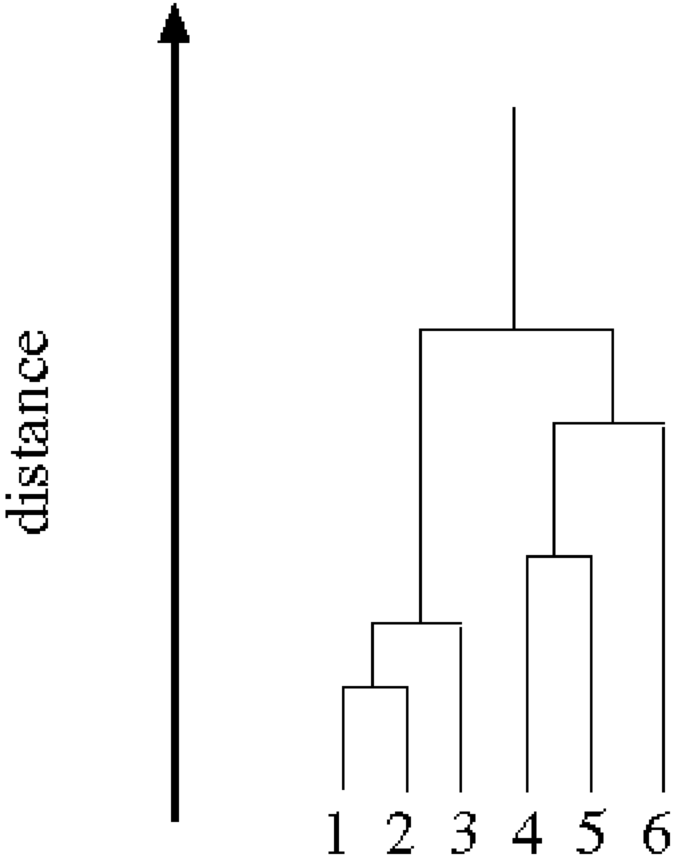
Make each point a separate cluster
Until the clustering is satisfactory
    Merge the two clusters with the
      smallest inter-cluster distance
end

Algorithm 15.4: Divisive clustering, or clustering by splitting

Construct a single cluster containing all points
Until the clustering is satisfactory
    Split the cluster that yields the two
      components with the largest inter-cluster distance
end



Feature Space



Dendogram

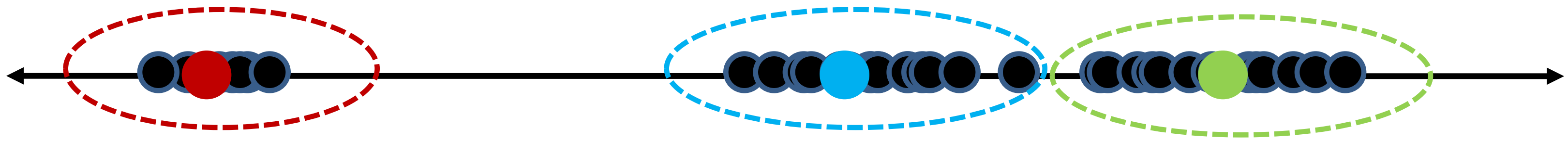
Segmentation Algorithms

- *Top down* vs. *bottom up* segmentation
  - Top down: pixels belong together because they are from the same object
  - Bottom up: pixels belong together because they look similar

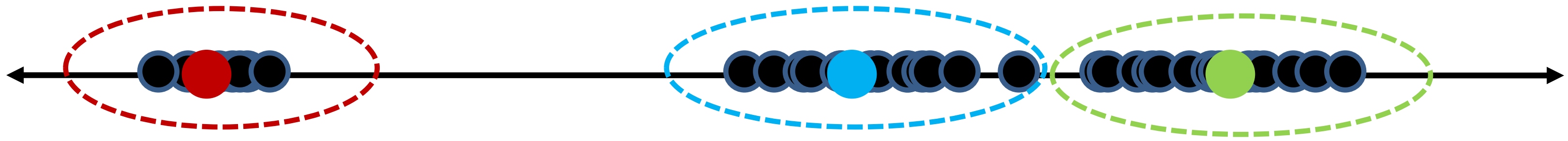
- Common methods:
  - Group similar pixels
    - K-means
  - Separate groups of pixels
    - Graph cuts

Grouping & Cluster Centers

- A "chicken and egg" problem:
  - If we knew the **cluster centers**, we could allocate points to groups by assigning each to its closest center.

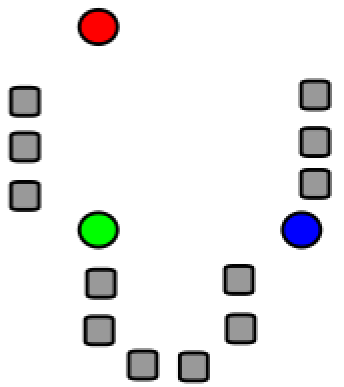


  - If we knew the **group memberships**, we could get the centers by computing the mean per group.

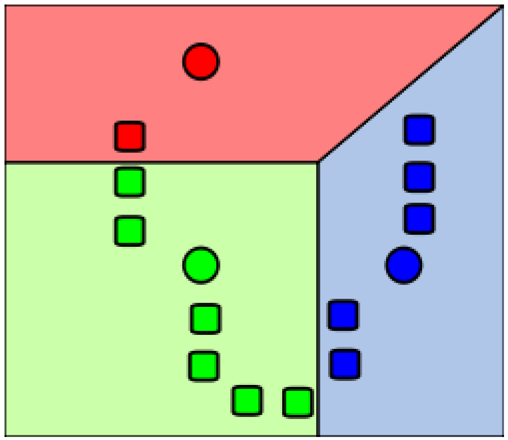


Kristen Grauman

K-Means

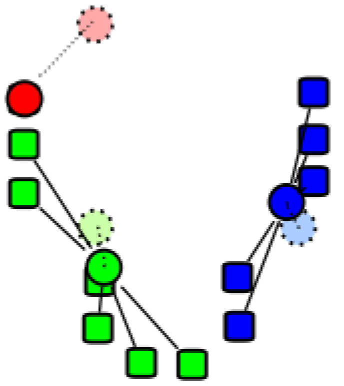


- Basic unsupervised learning (clustering) algorithm
- Randomly place *k* centres in the pattern space
- For each training example, calculate the distances to all *k* centres and choose the closest
  - This is the initial classification
  - All examples will be assigned a class from 1 to *k*



K-Means (cont'd)

- For all examples in the same class, calculate a new mean and move the centre to this location
  - We now have *k* new centres.
- Repeat the process
  - Measure distance between the centres and each example
  - Re-classify examples
  - Until there is no further change (convergence)
    - i.e. the sum of the distances monitored and training halts when the total distance no longer falls



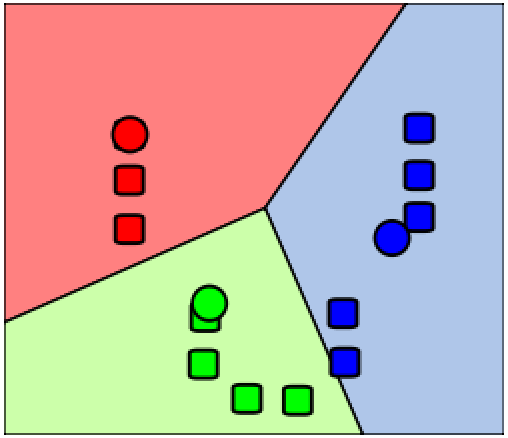
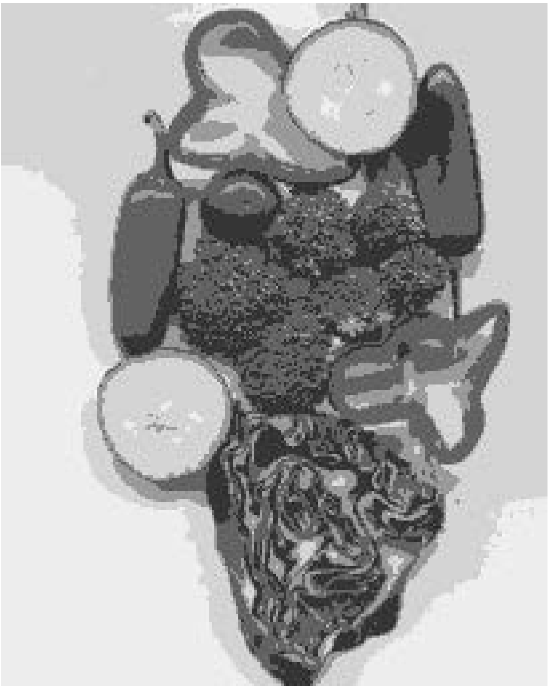


Image Segmentation by K-Means

- Select a value of K
- Select a feature vector for every pixel
  - e.g., color, texture, position, or combination
- Define a similarity measure between feature vectors
  - Usually Euclidean Distance
- Apply K-Means Algorithm
- Post-process (“clean up”) results

Results of K-Means Clustering:



Image

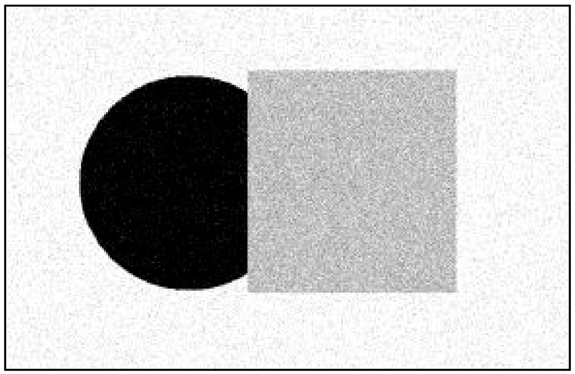
Clusters on intensity

Clusters on color

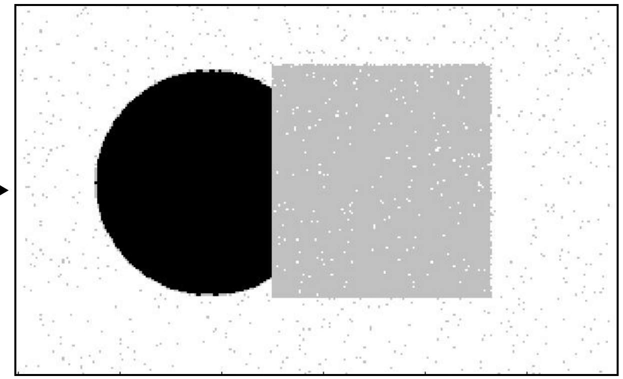
K-means clustering using intensity alone and color alone

Smoothing out cluster assignments

- Assigning a cluster label per pixel may yield outliers:



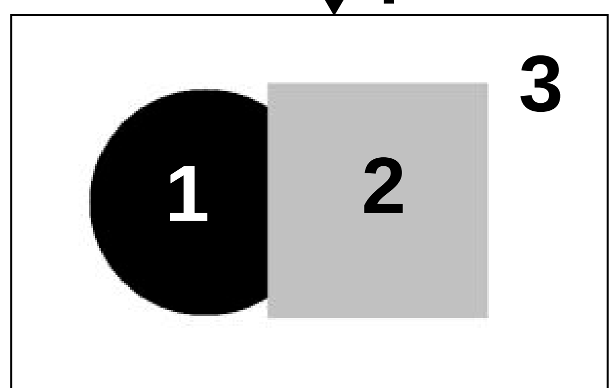
original



labeled by cluster center’s intensity

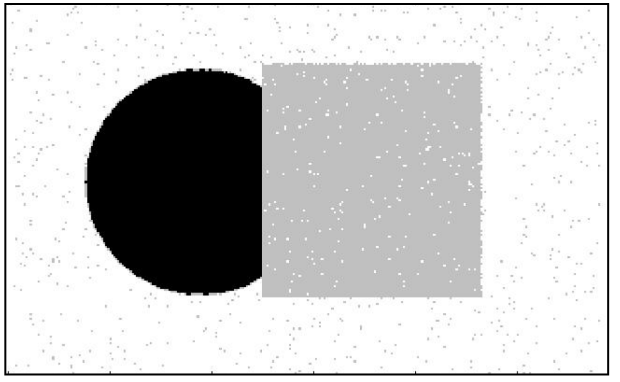
?

- How to ensure they are spatially smooth?



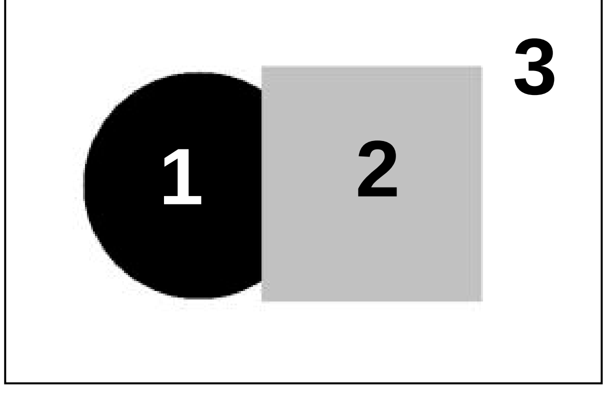
Spatial Smoothness

- How to get spatially smooth clusters?
- Post-Processing
  - Smoothing
  - Connected Components
- Regularization
  - Dependencies between pixels
  - Cluster membership is affected by neighboring pixels
  - Many approaches
- Separate groups of pixels



labeled by cluster center's intensity

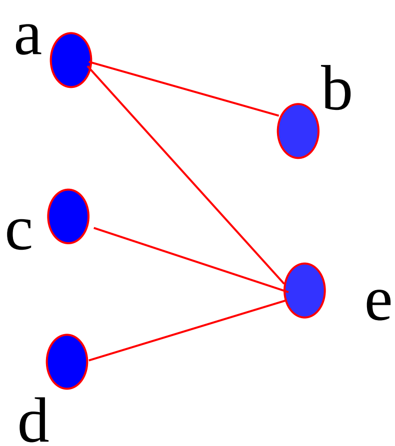
↓?



Graph theoretic clustering

- Represent image using a weighted graph
  - Pixels are vertices and connected to neighboring pixels
  - affinity (similarity) matrix
  - (p_i *same as* p_j → affinity of 1)
- Cut up this graph to get subgraphs with strong interior links and weaker exterior links

Graphs Representations

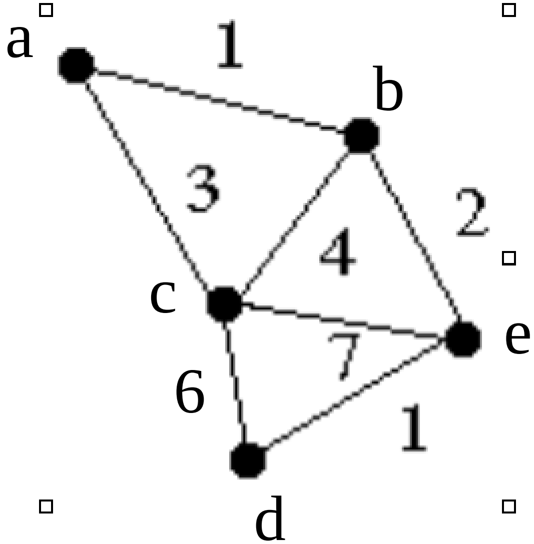


$$\begin{array}{c|ccccc} & a & b & c & d & e \\ \hline a & \\ b & \\ c & \\ d & \\ e & \end{array}$$

$$\begin{array}{cc} & \begin{array}{ccccc} a & b & c & d & e \end{array} \\ \begin{array}{c} a \\ b \\ c \\ d \\ e \end{array} & \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

Adjacency Matrix: W

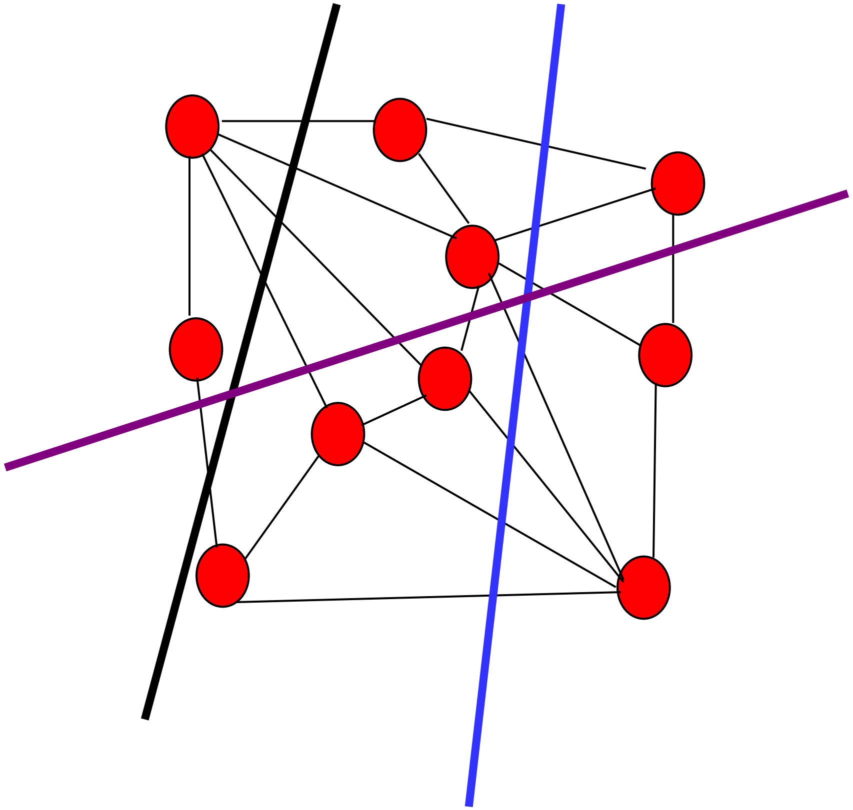
Weighted Graphs and Their Representations



$$
\begin{array}{c} \\ a \\ b \\ c \\ d \\ e \end{array}
\begin{array}{c}
\begin{array}{ccccc} a & b & c & d & e \end{array} \\
\begin{bmatrix}
0 & 1 & 3 & \infty & \infty \\
1 & 0 & 4 & \infty & 2 \\
3 & 4 & 0 & 6 & 7 \\
\infty & \infty & 6 & 0 & 1 \\
\infty & 2 & 7 & 1 & 0
\end{bmatrix}
\end{array}
$$

Weight Matrix: W

Minimum Cut

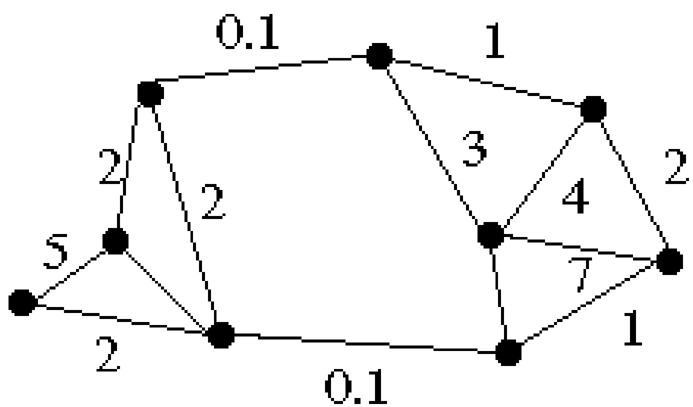


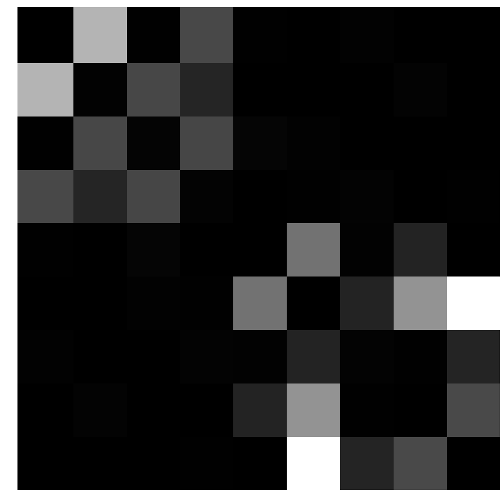
A cut of a graph G is the set of edges S such that removal of S from G disconnects G.

Minimum cut is the cut of minimum weight, where weight of cut <A,B> is given as

$$w(\langle A, B \rangle) = \sum_{x \in A, y \in B} w(x, y)$$

Minimum Cut and Clustering





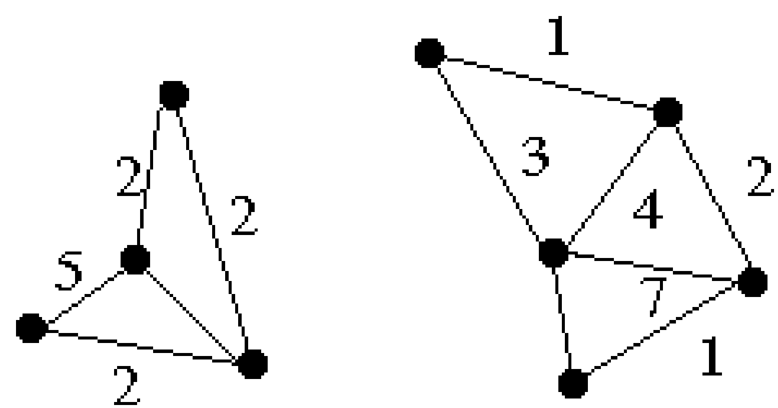
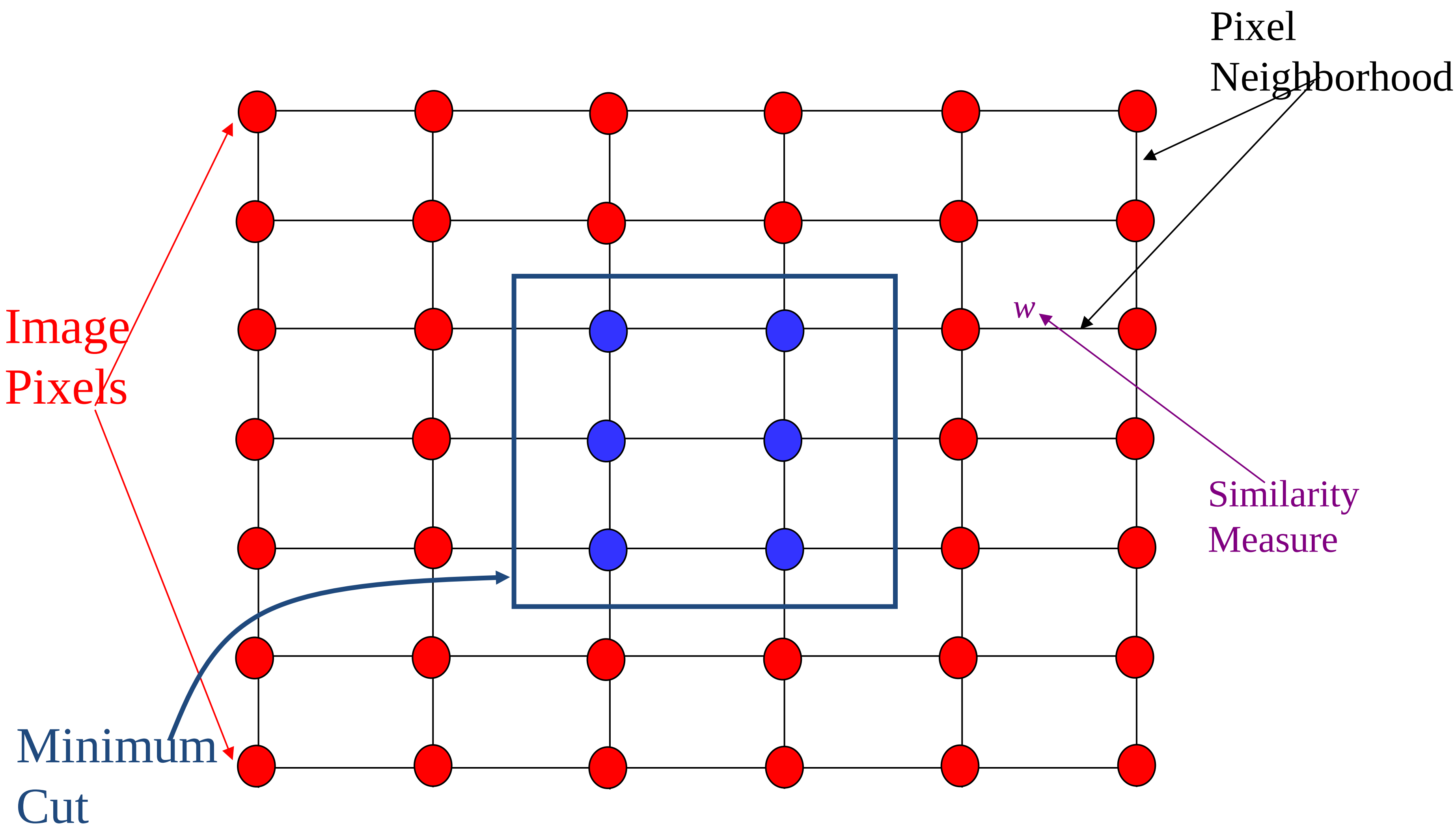
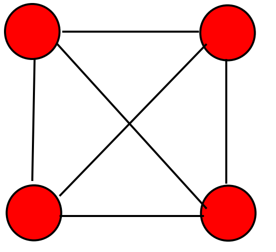


Image Segmentation & Minimum Cut



Minimum Cut

- There can be more than one minimum cut in a given graph



- All minimum cuts of a graph can be found in polynomial time[1].

[1]H. Nagamochi, K. Nishimura and T. Ibaraki, “Computing all small cuts in an undirected network. SIAM J. Discrete Math. 10 (1997) 469-481.

Measuring Affinity

Intensity

$$aff(x,y) = \exp\left\{ -\left(\frac{1}{2\sigma_i^2} \right) \left(\left\| I(x) - I(y) \right\|^2 \right) \right\}$$

Distance

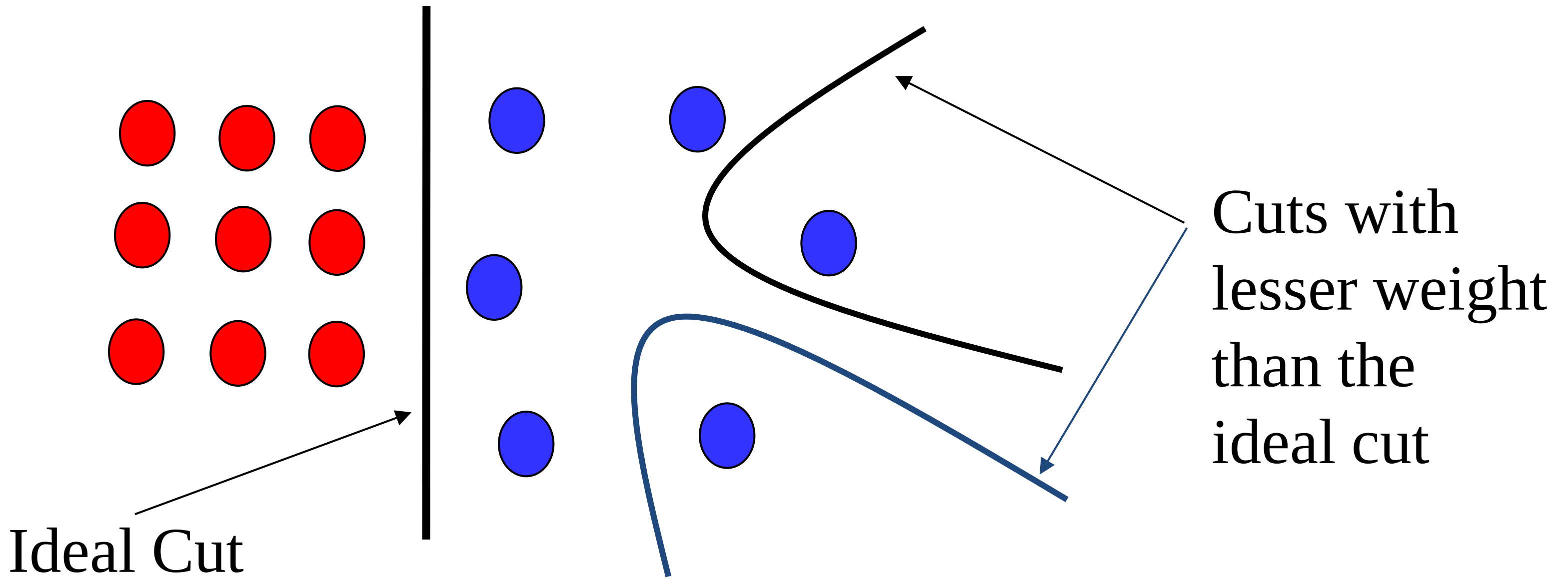
$$aff(x,y) = \exp\left\{ -\left(\frac{1}{2\sigma_d^2} \right) \left(\left\| x - y \right\|^2 \right) \right\}$$

Texture

$$aff(x,y) = \exp\left\{ -\left(\frac{1}{2\sigma_t^2} \right) \left(\left\| c(x) - c(y) \right\|^2 \right) \right\}$$

Drawbacks of Minimum Cut

- Weight of cut is directly proportional to the number of edges in the cut.



- How could we deal with this?

Normalized Cuts

- Normalized cut is defined as

$$
\begin{aligned}
Ncut(A,B) &= \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)} \\
&= \frac{assoc(A,V) - assoc(A,A)}{assoc(A,V)} \\
&\quad + \frac{assoc(B,V) - assoc(B,B)}{assoc(B,V)} \\
&= 2 - \left(\frac{assoc(A,A)}{assoc(A,V)} + \frac{assoc(B,B)}{assoc(B,V)} \right)
\end{aligned}
$$

- Minimizing $N_{cut}(A,B)$ maximizes a measure of similarity within the sets A and B

J. Shi and J. Malik, "Normalized Cuts & Image Segmentation," IEEE Trans. of PAMI, Aug 2000. *As of Aug. 2016, this paper has been cited ~10000 times.*

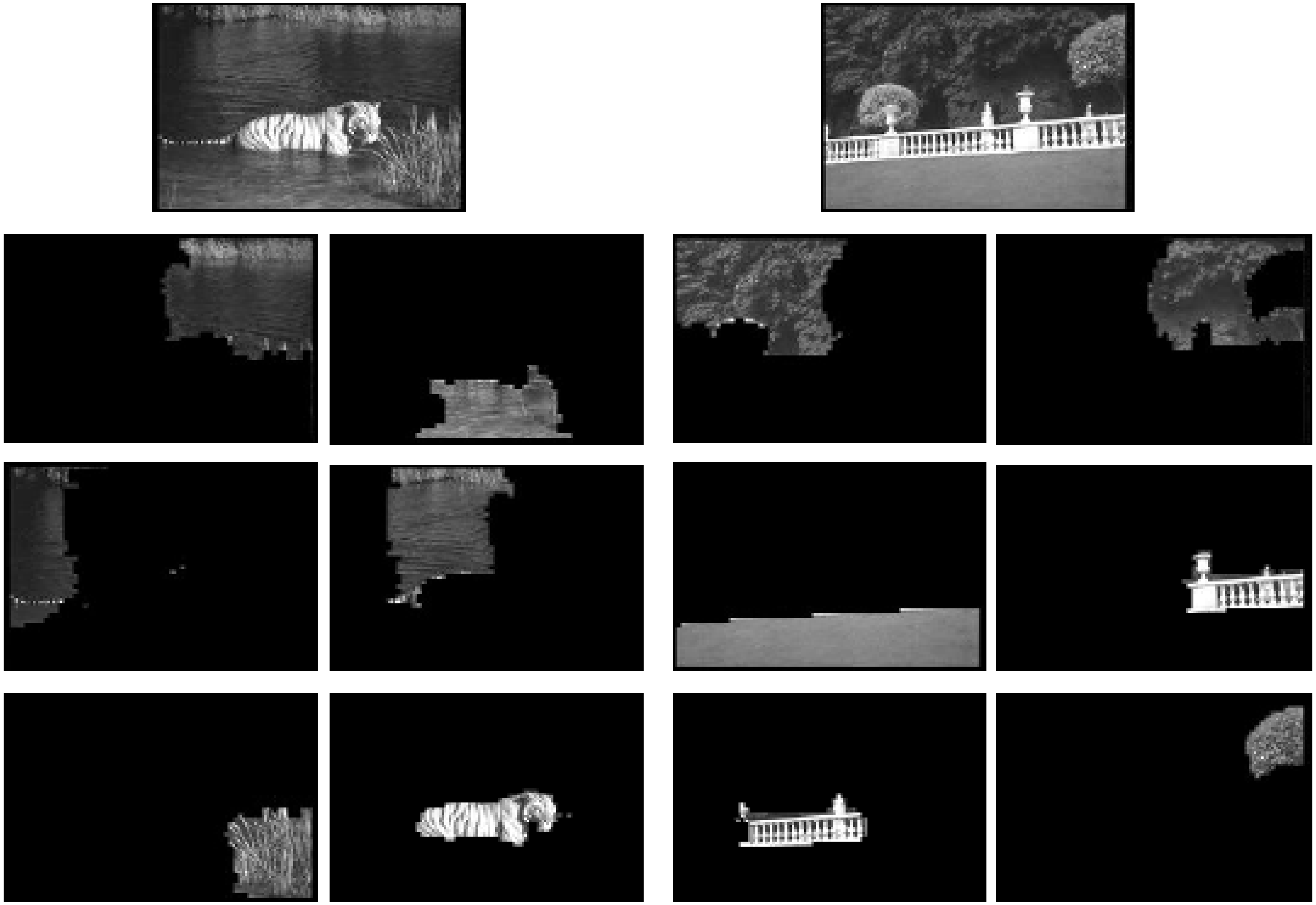
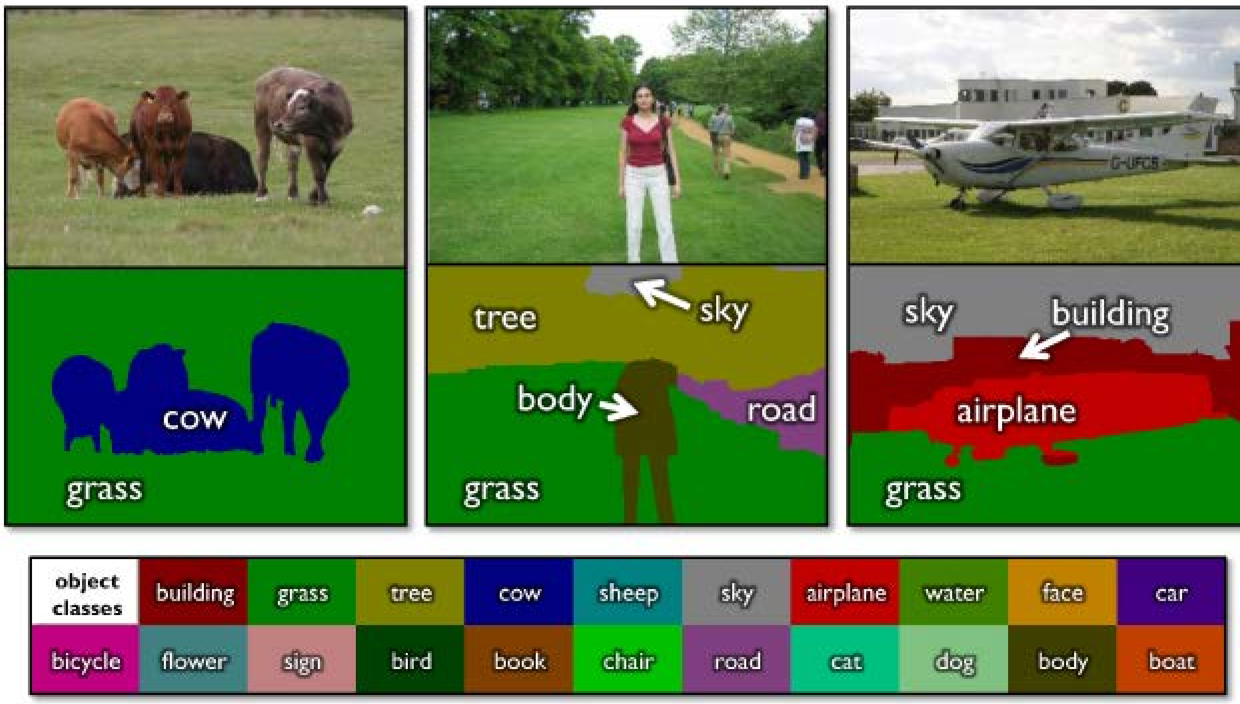


Figure from “Image and video segmentation: the normalised cut framework”, by Shi and Malik, 1998

Using Graph Cuts for Recognition

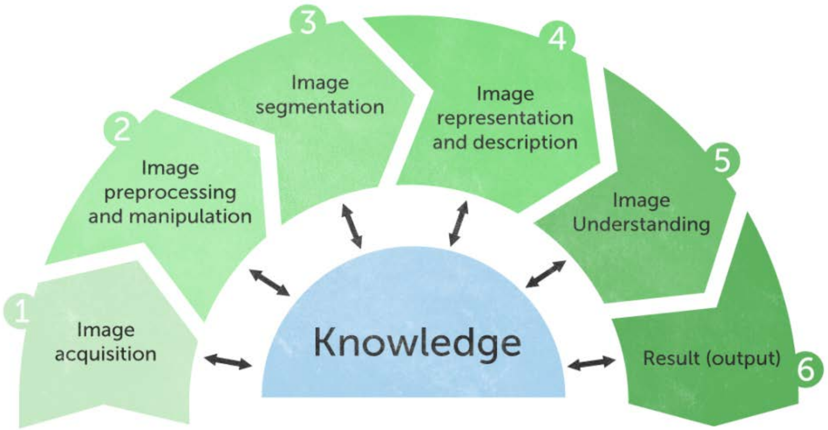


TextonBoost (Shotton et al. 2009 IJCV)

Drawbacks of Minimum Normalized Cut

- Huge storage requirement and time complexity
  - Finding the Minimum Normalized-Cut is NP-Hard
  - Polynomial approximations are generally used for segmentation
- Bias towards partitioning into equal segments
- Problems with textured backgrounds

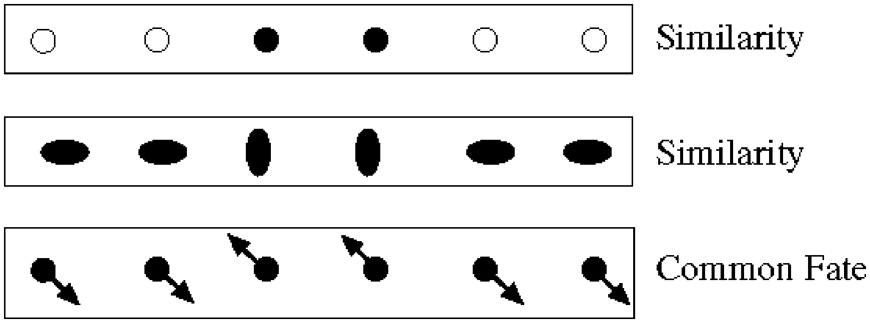
Segmentation



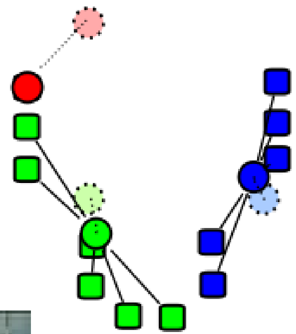
- Important CV Problem
  - Gets its own step (#3)
  - First step for many other problems: recognition, stereo matching, image labeling
- Lots more approaches
  - Watershed, mean-shift, level set, etc.

Summary

- Grouping
  - Gestalt properties



- K-means
  - Iterative "chicken-and-egg" solver



- Graph cut
  - Dividing image into regions