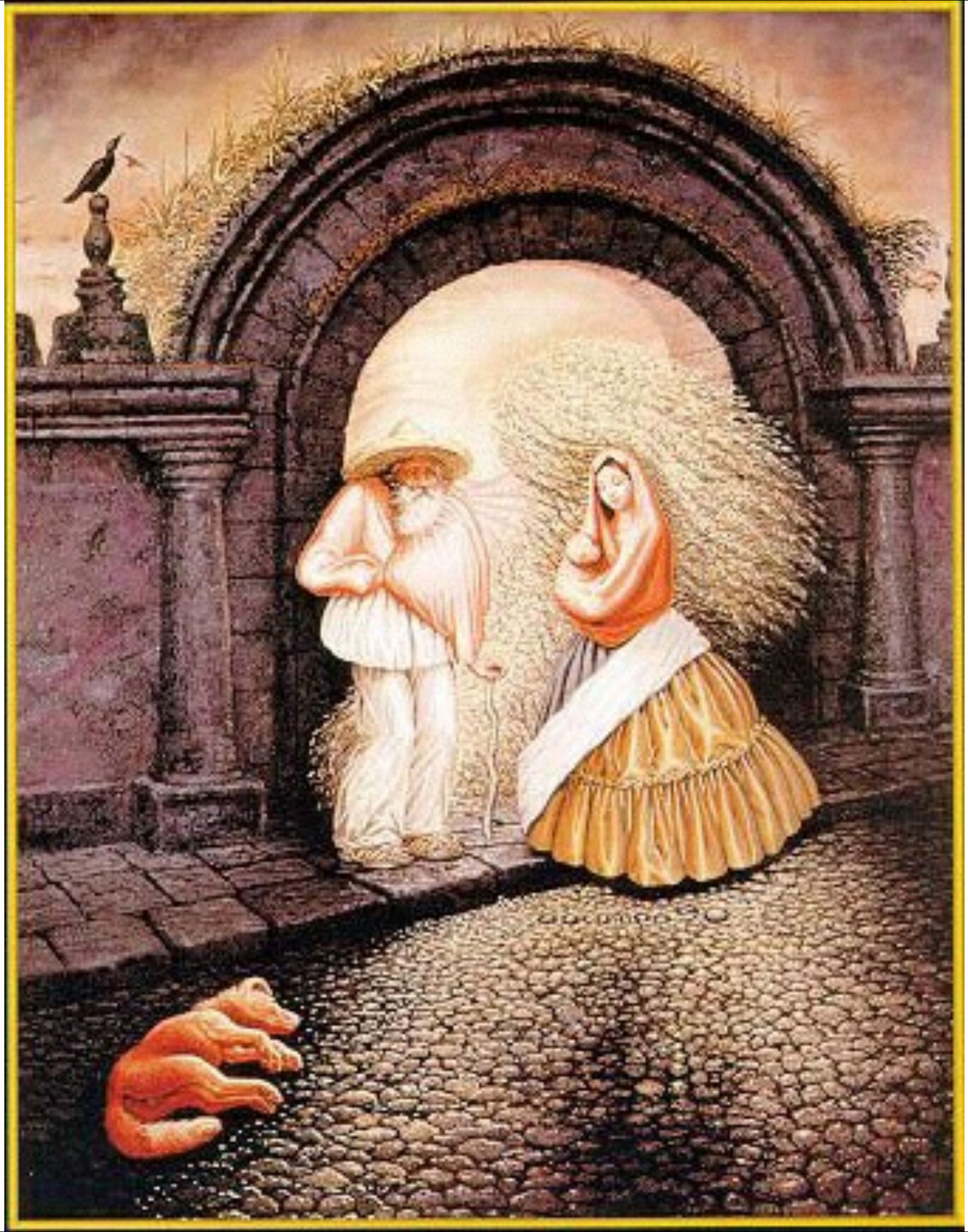
Lecture 16
Segmentation and Scene understanding

- Introduction
- Mean-shift
- Graph-based segmentation
- Top-down segmentation

Segmentation

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

- Tokens
  - 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

Basic ideas of grouping in human vision

- Gestalt properties
- Figure-ground discrimination
- Emergence

Gestalt psychology or **gestaltism**

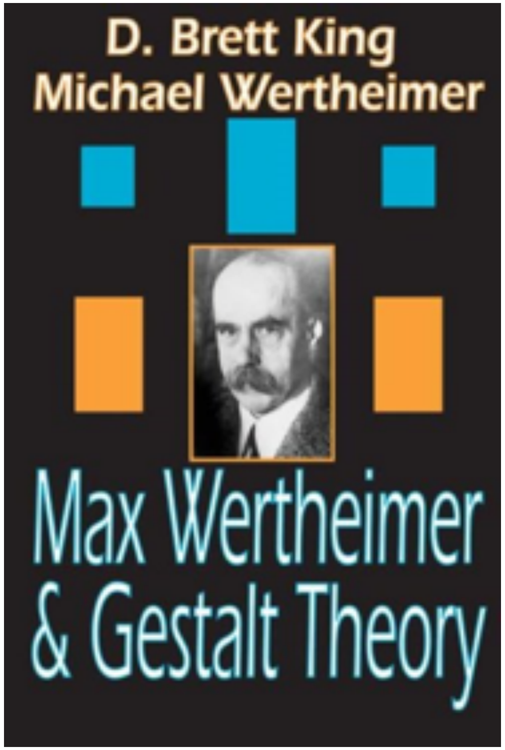
German: *Gestalt* - "form" or "whole”

Berlin School , early 20th century, Carl Stumpf

Kurt Koffka, Max Wertheimer, and Wolfgang Köhler

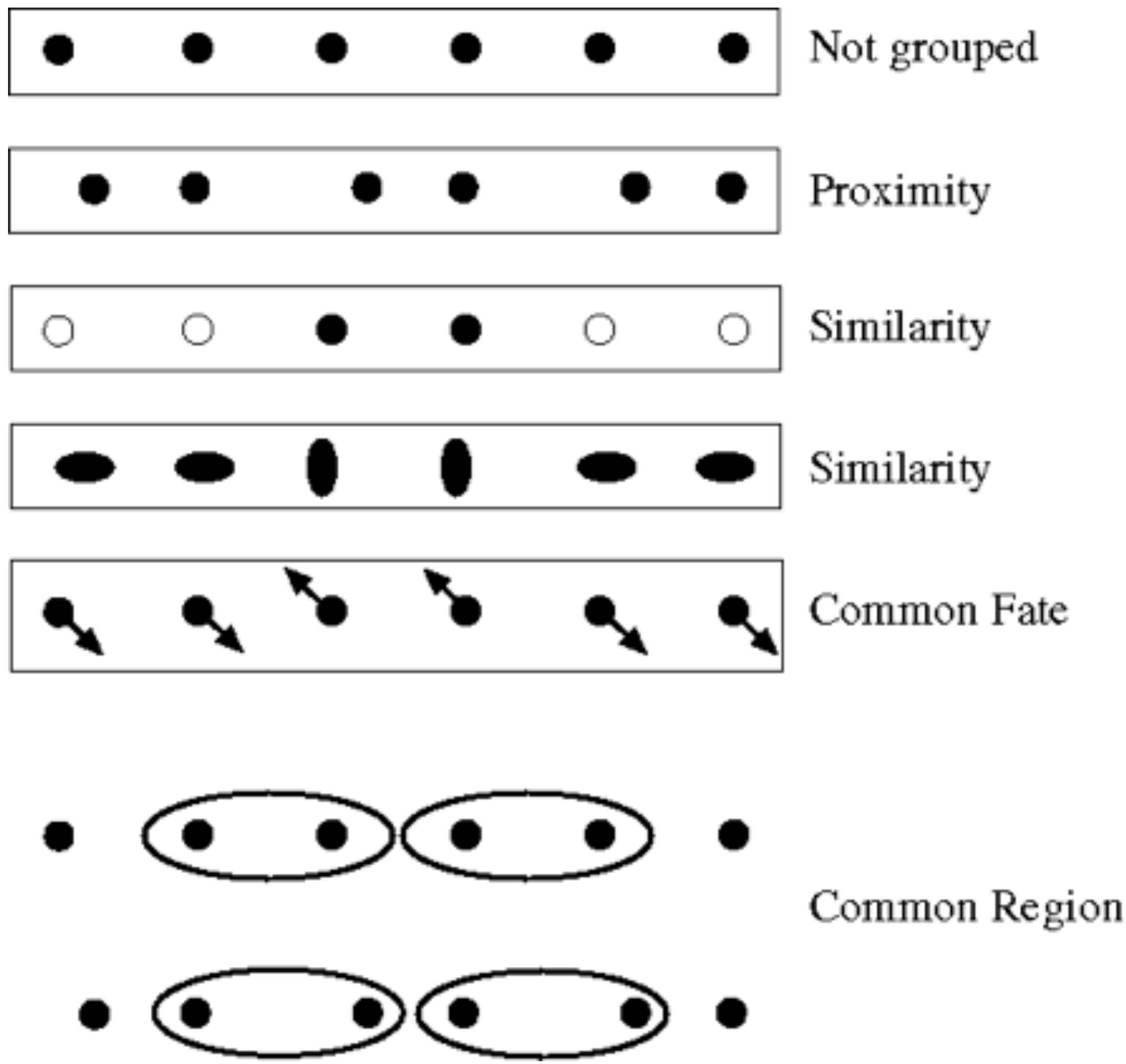
Advocate brain is holistic

Whole is greater than the sum of its parts.



Gestalt properties

–A series of factors affect whether elements should be grouped together

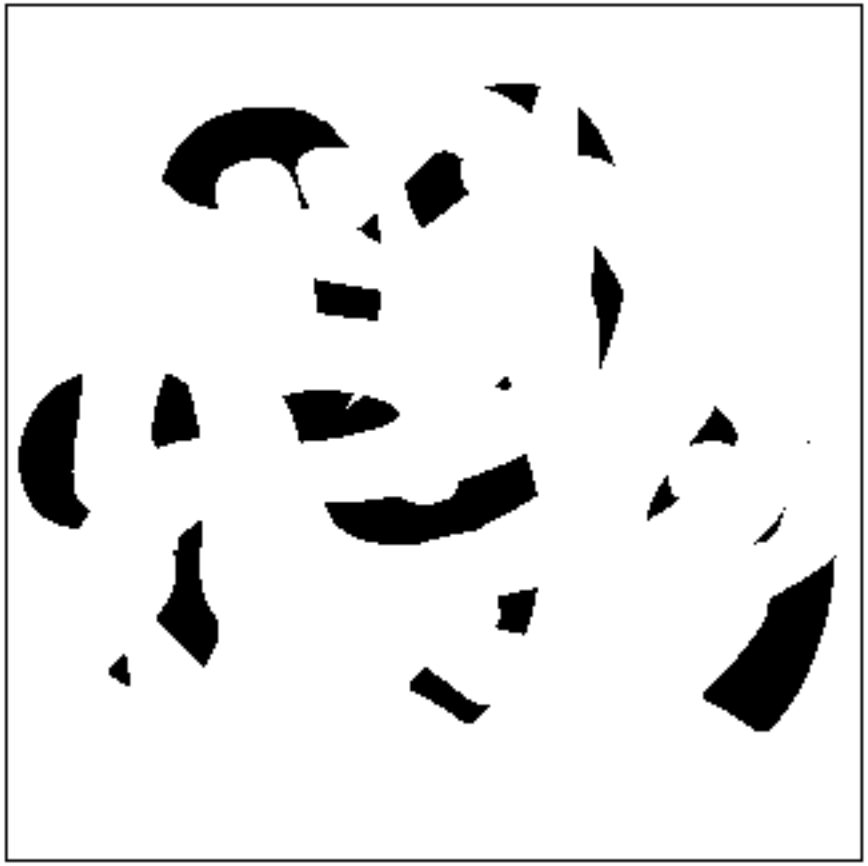


Gestalt properties

Grouping
by occlusions

Gestalt properties

Grouping
by occlusions



Gestalt properties

Grouping
by invisible
completions

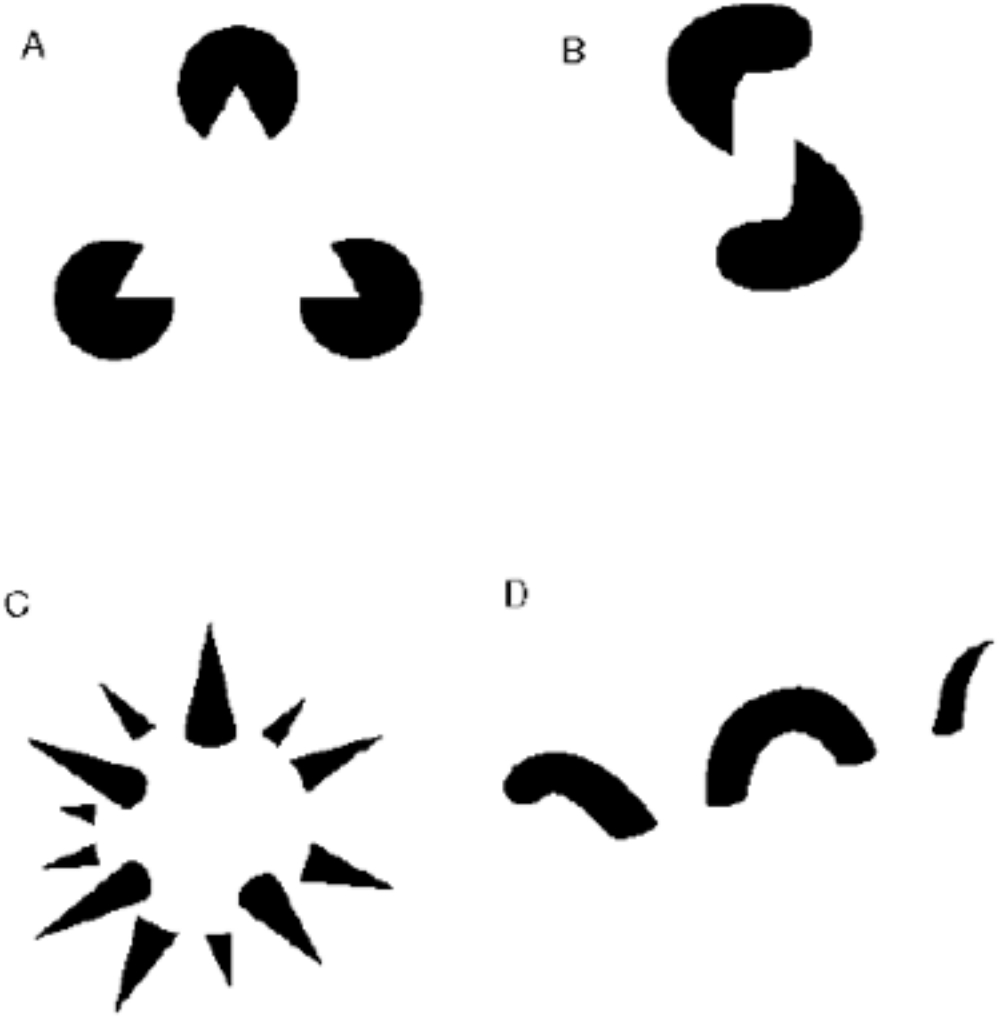
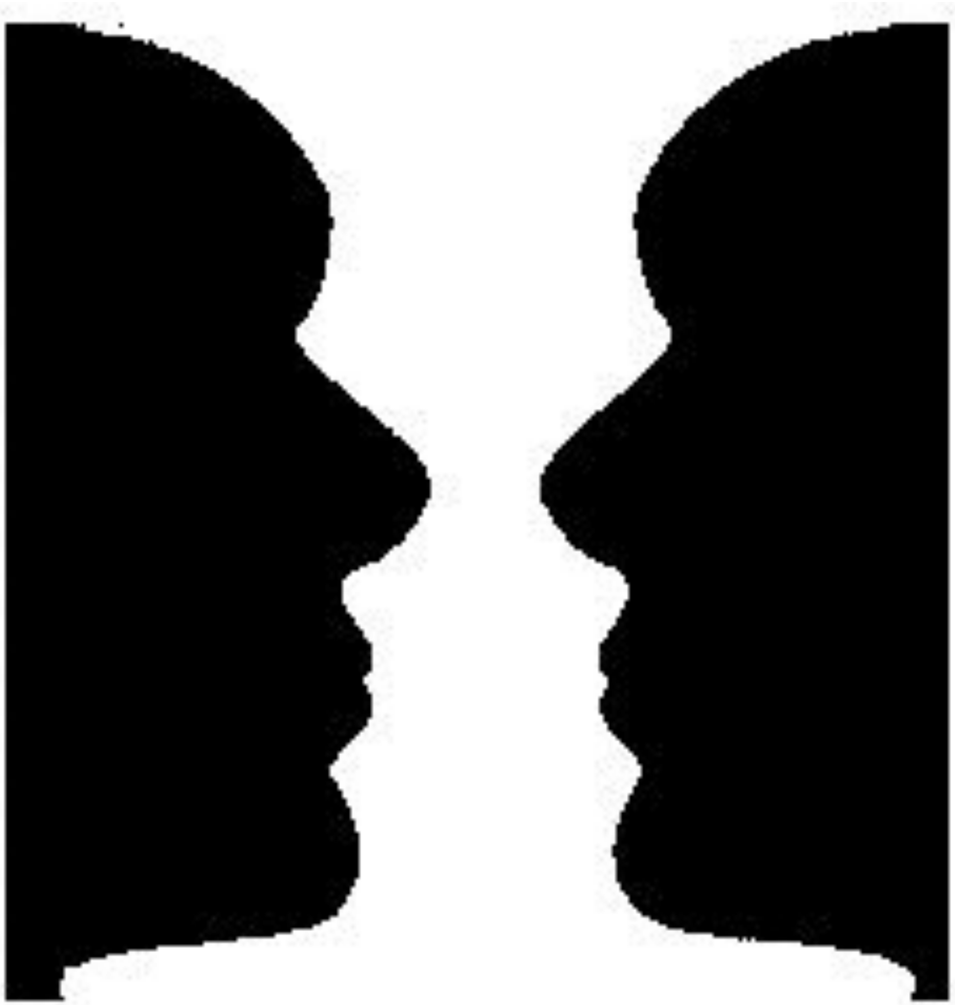


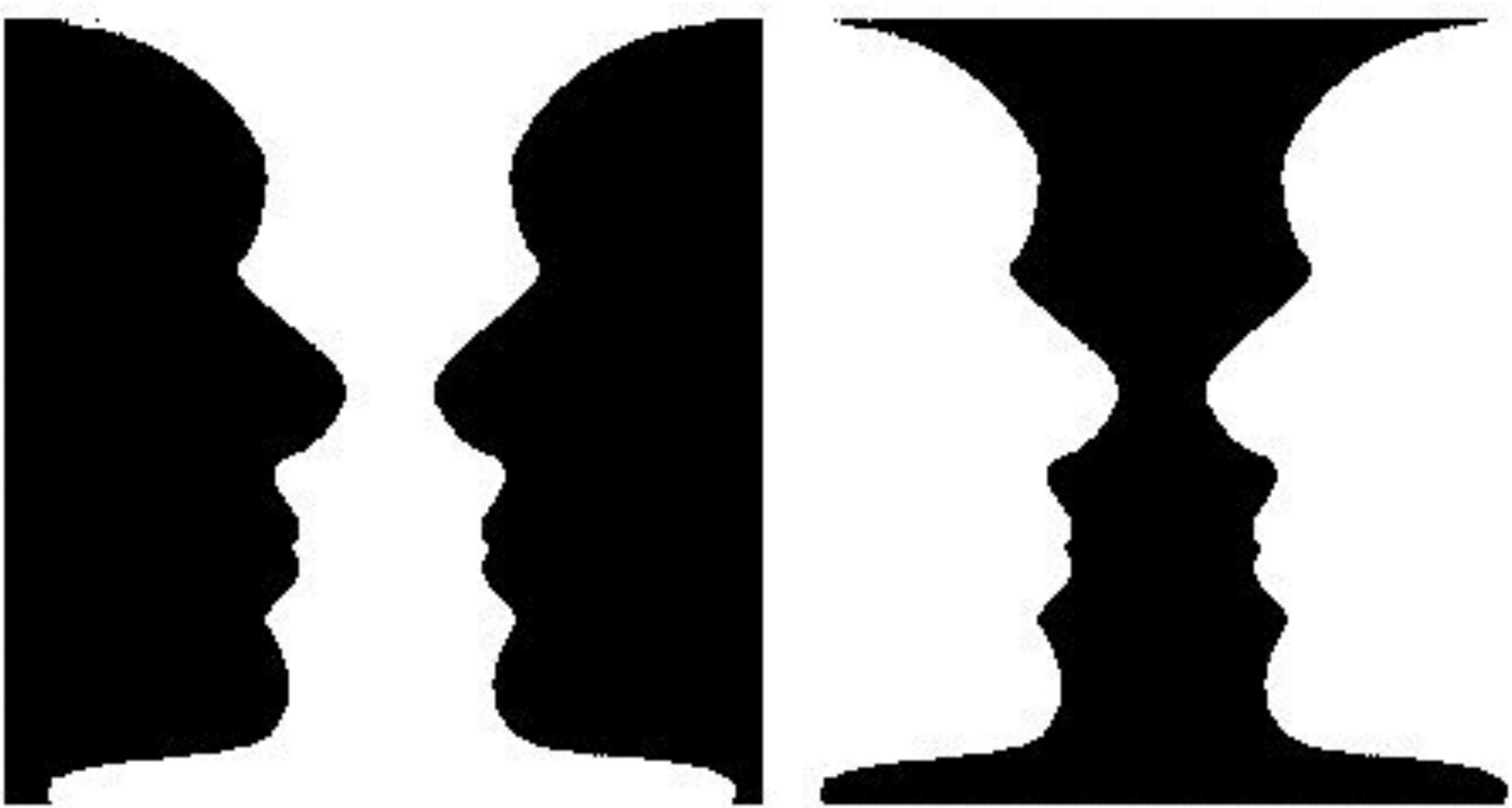
Figure-ground discrimination



–Grouping can be seen in terms of allocating some elements to a figure, some to ground

–Can be based on local bottom-up cues or high level recognition

Figure-ground discrimination



Emergence

Segmentation as clustering

Cluster together tokens that share similar visual characteristics

- **K-means** } See CS131A, CS 229
  - **Must specify number of clusters K in advance**
  - **Assumes clusters are spherical**

- **Mean-shift** } This lecture
  - **Discovers arbitrary number of clusters**
  - **No a priori assumptions about cluster shapes**

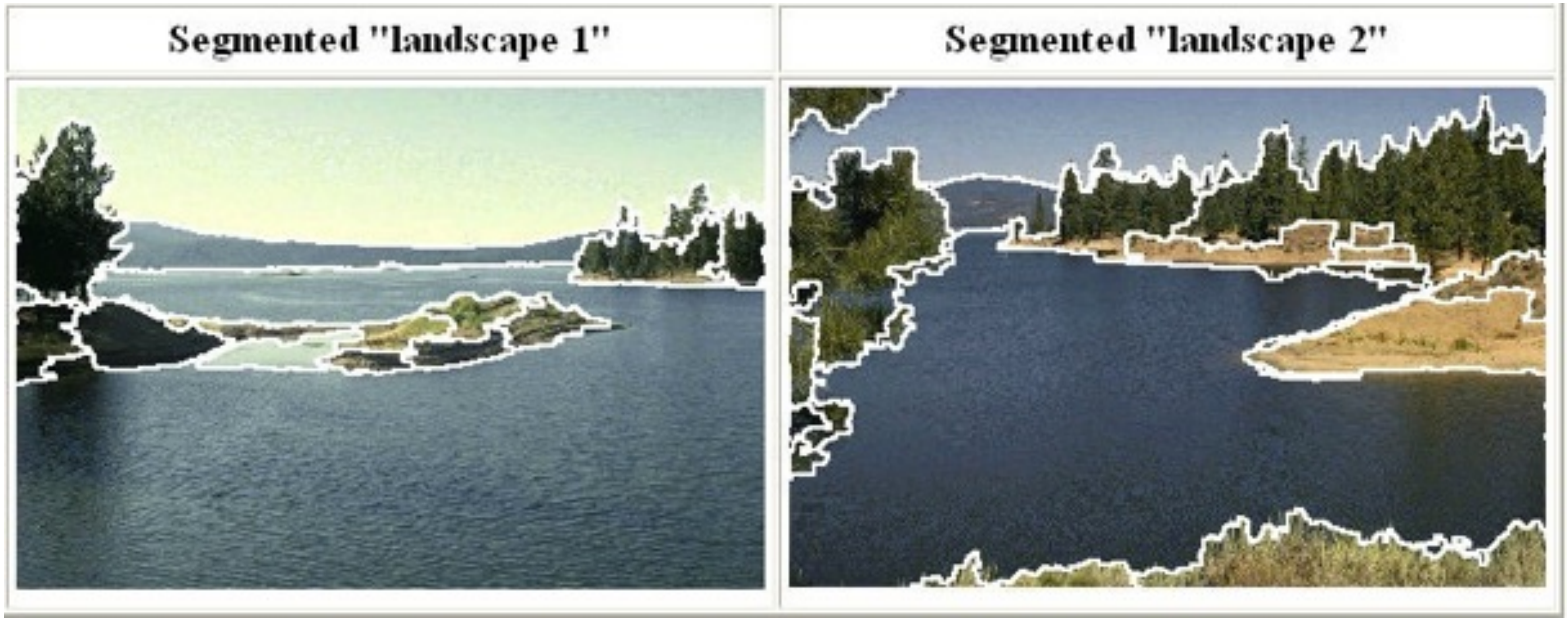
Lecture 16
Segmentation and Scene understanding

- Introduction
- Mean-shift
- Graph-based segmentation
- Top-down segmentation

Mean shift segmentation

D. Comaniciu and P. Meer, Mean Shift: A Robust Approach toward Feature Space Analysis, PAMI 2002.

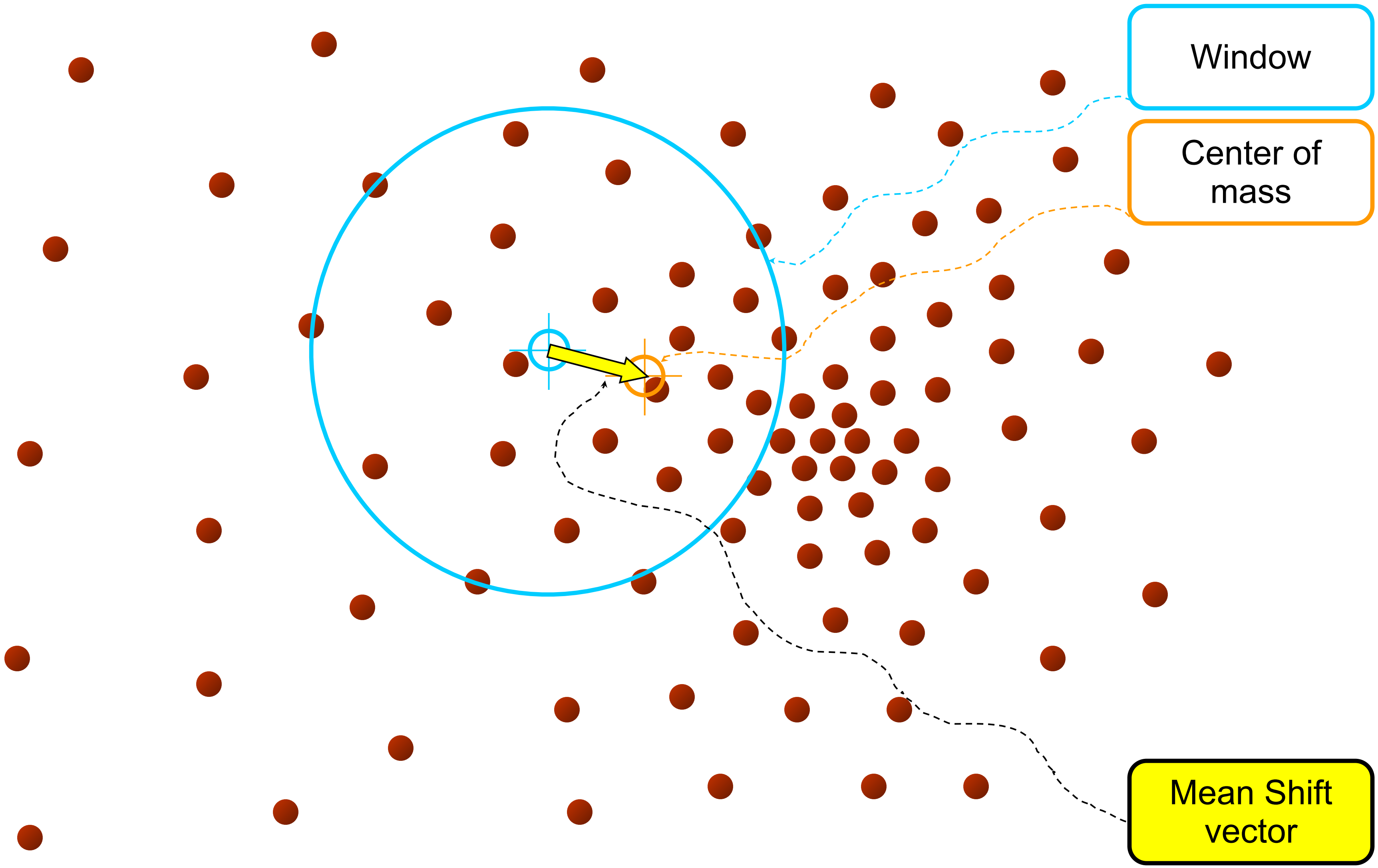
- A versatile technique for clustering-based segmentation



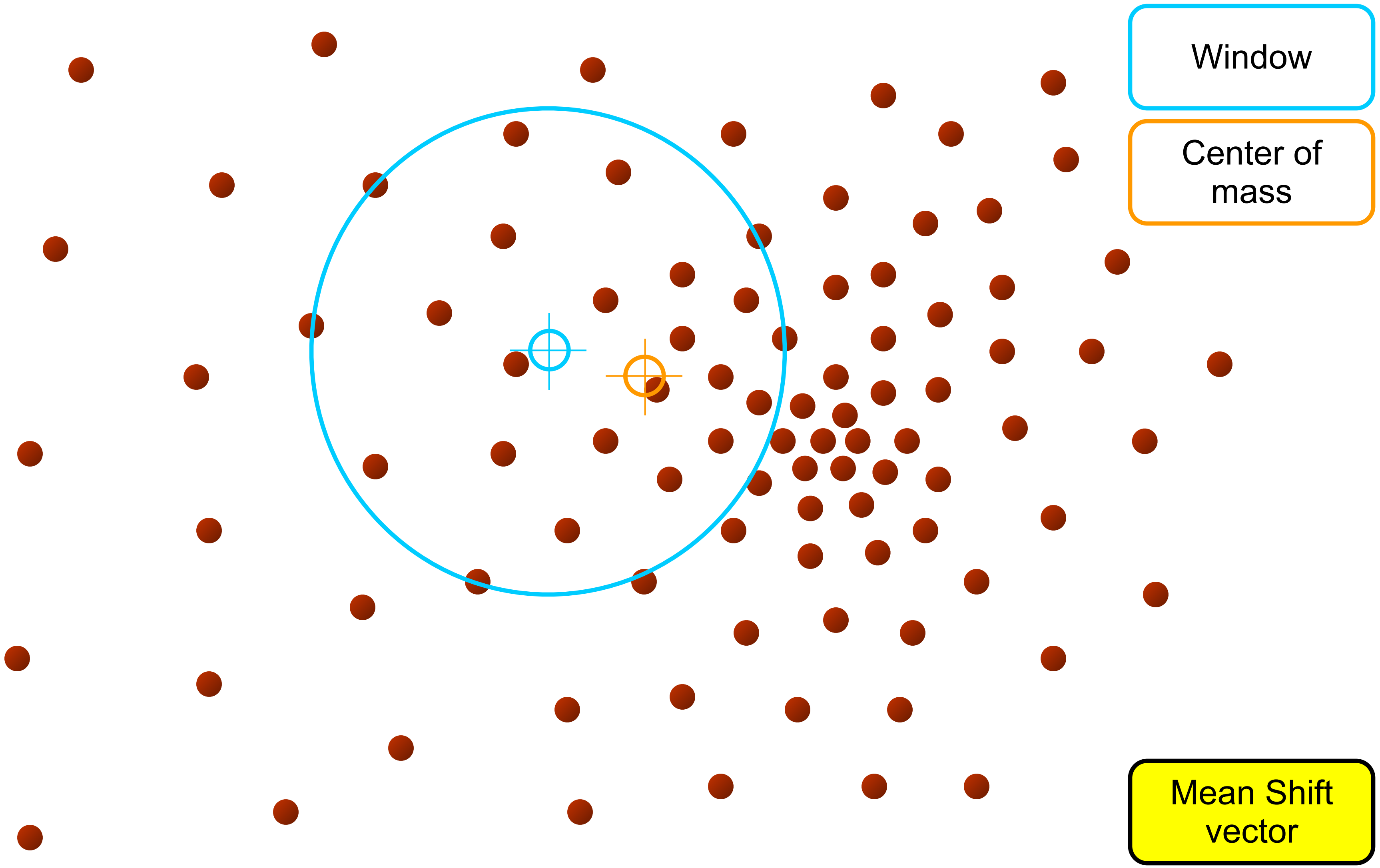
Mean shift algorithm

Fukunaga, Keinosuke; Larry D. Hostetler (January 1975). "The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition". *IEEE Transactions on Information Theory* (IEEE) **21** (1): 32–40

- The mean shift algorithm seeks the *modes* or local maximums of density of a given distribution
  - Choose a search window (size and location)
  - Compute the mean of the data in the search window
  - Center the search window at the new mean location
  - Repeat until convergence

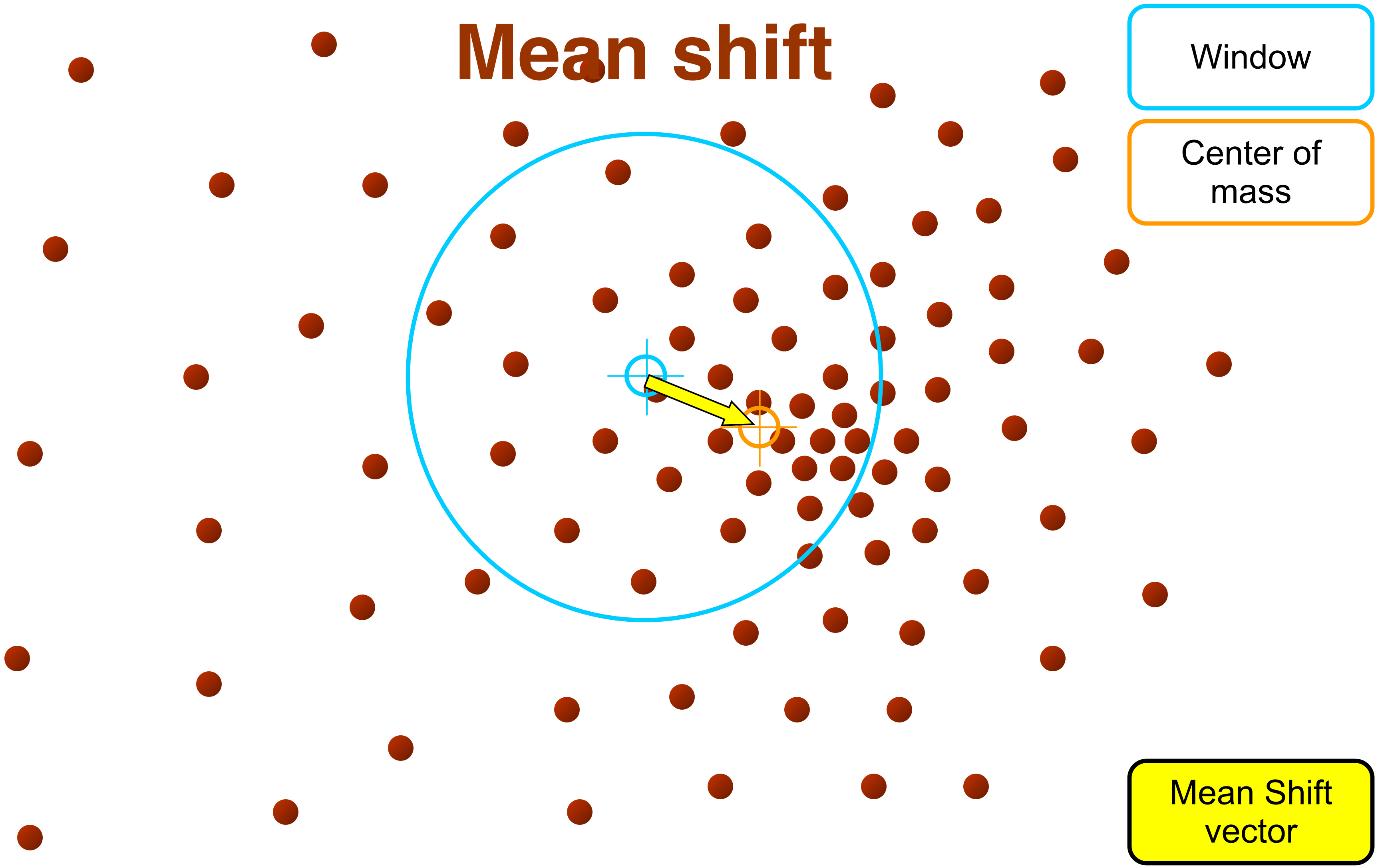


Slide by Y. Ukrainitz & B. Sarel



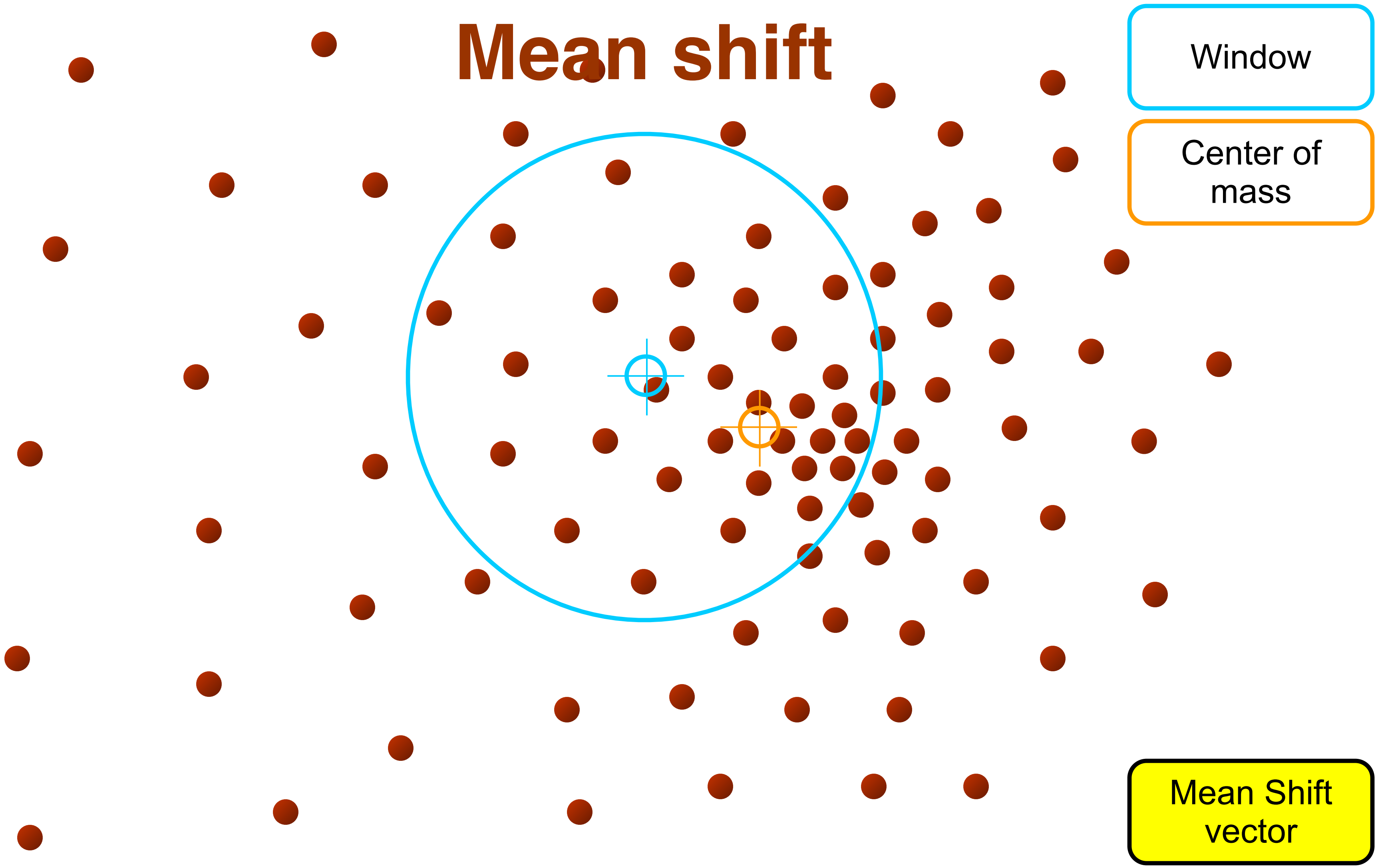
Slide by Y. Ukrainitz & B. Sarel

Mean shift

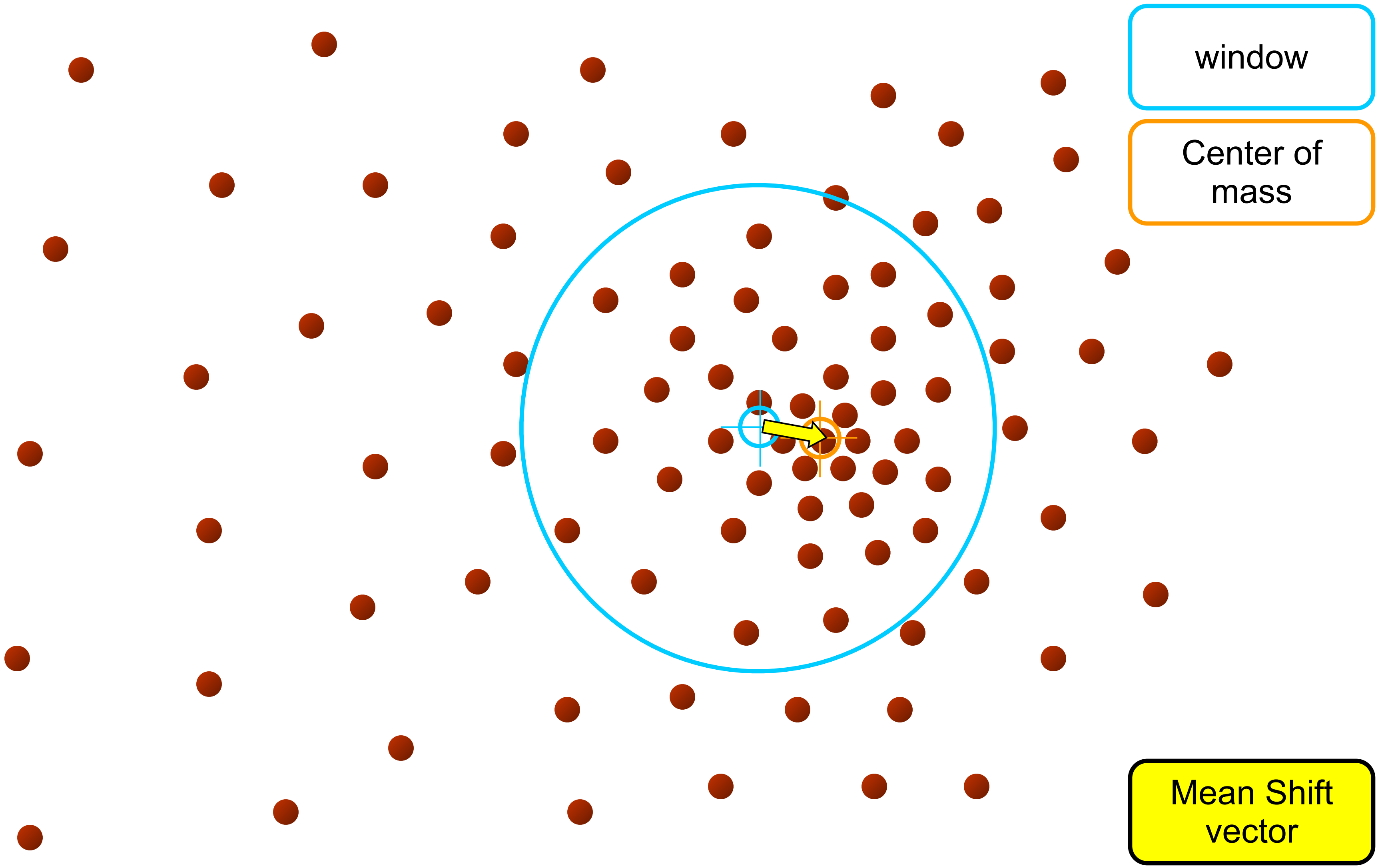


Slide by Y. Ukrainitz & B. Sarel

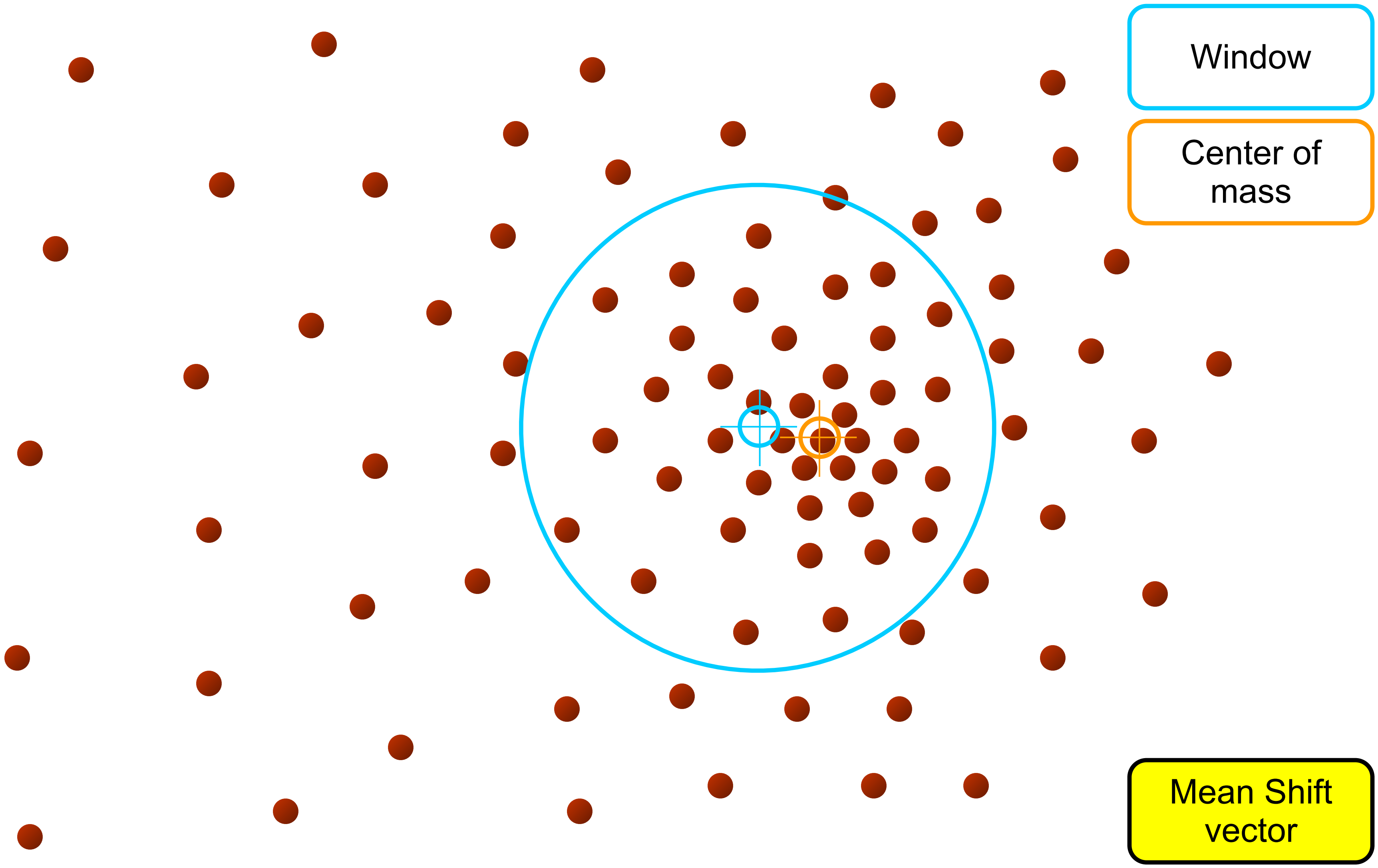
Mean shift



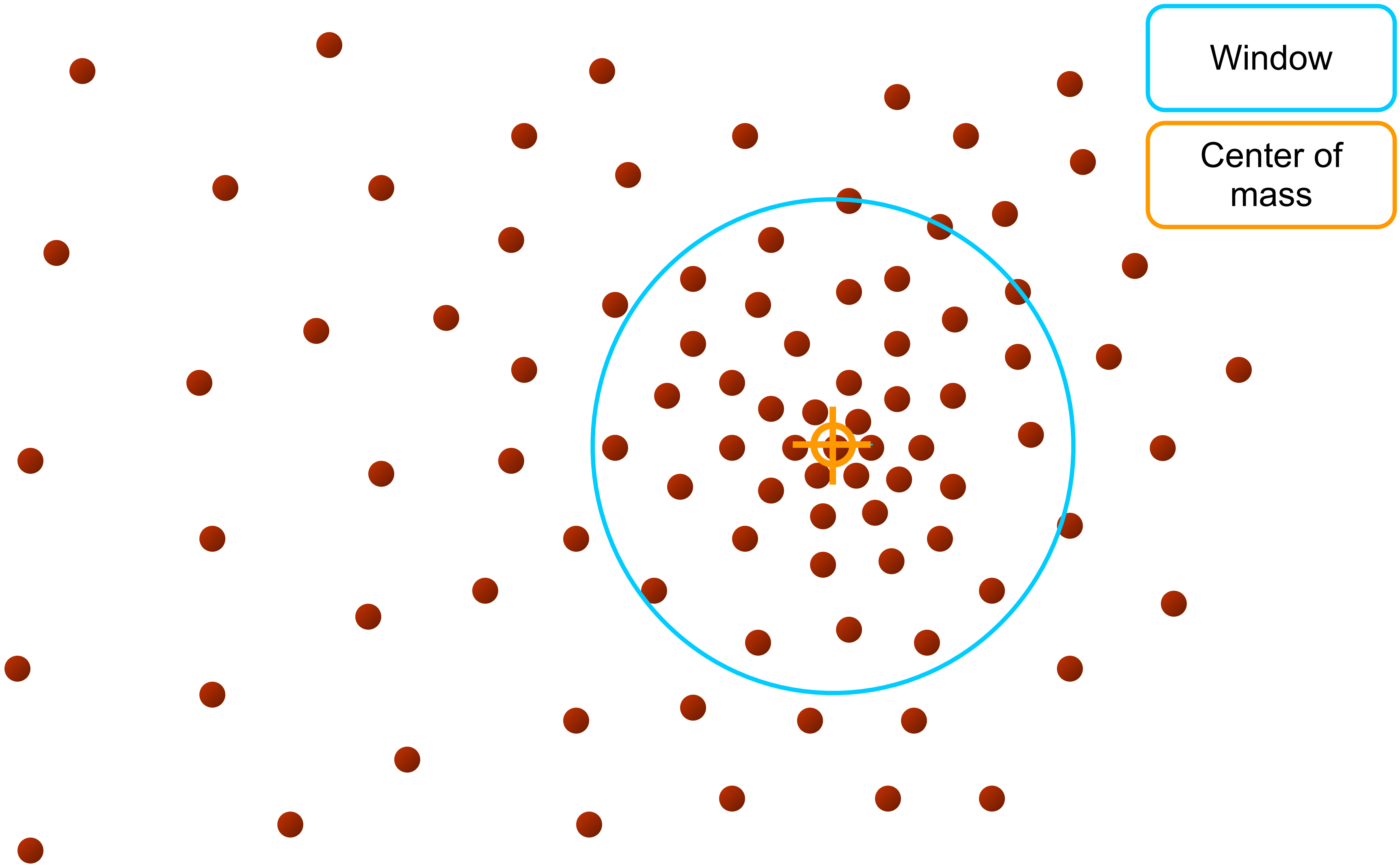
Slide by Y. Ukrainitz & B. Sarel



Slide by Y. Ukrainitz & B. Sarel



Slide by Y. Ukrainitz & B. Sarel



Slide by Y. Ukrainitz & B. Sarel

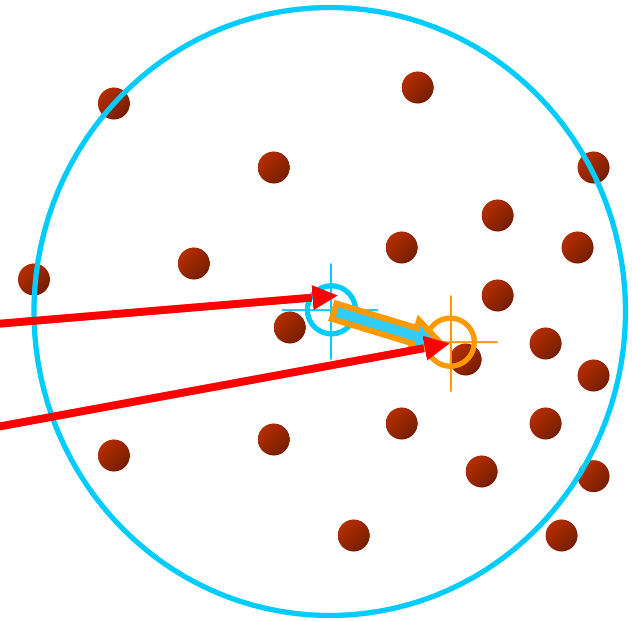
Computing The Mean Shift

<u>Simple Mean Shift procedure</u>:

- Compute mean shift vector

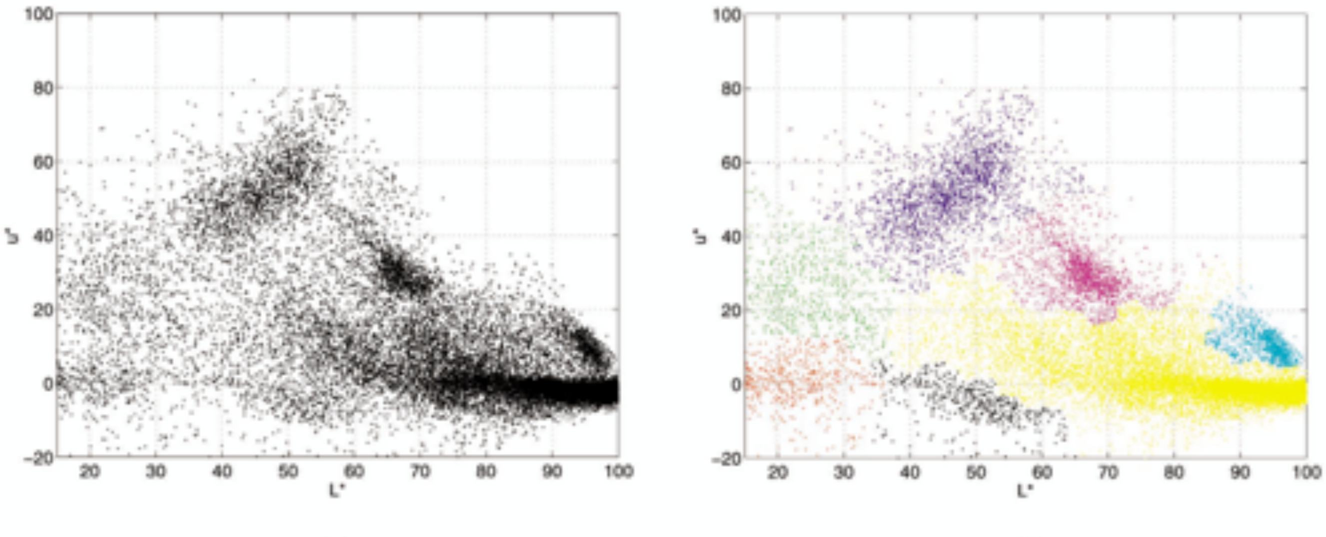
- Translate the Kernel window by **m(x)**

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



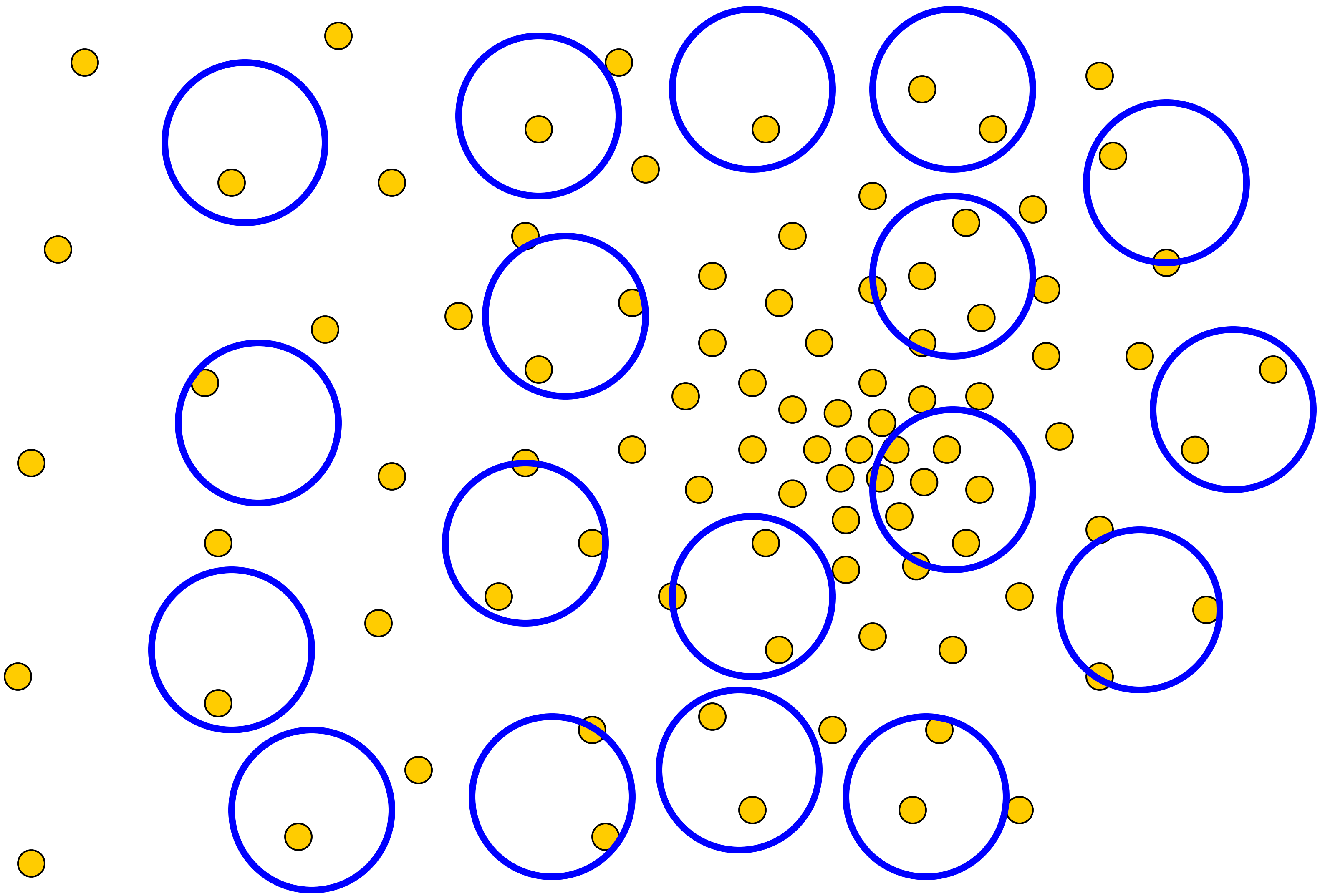
$g(\mathbf{x}) = -k'(\mathbf{x})$

Multimodal distributions



Slide by Y. Ukrainitz & B. Sarel

Real Modality Analysis

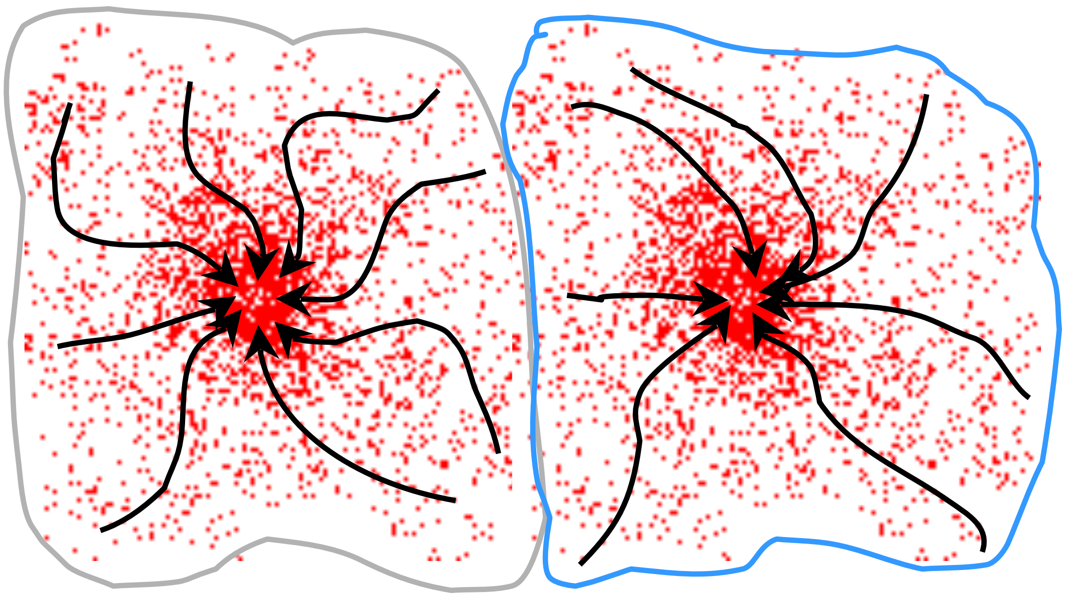


- **Tessellate the space with windows**

•Merge windows that end up near the same “peak” or model

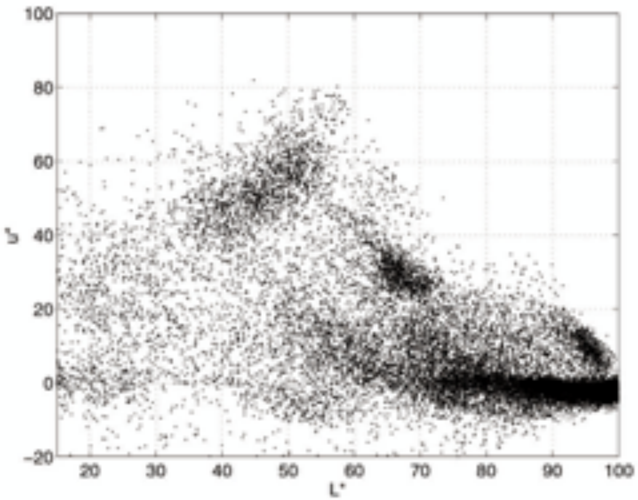
Attraction basin

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

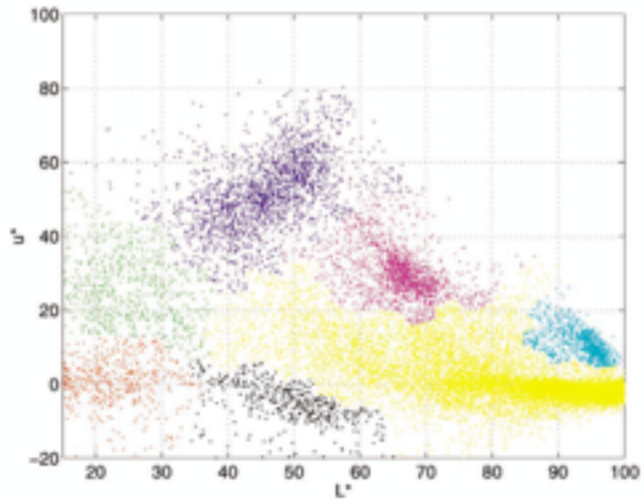


Slide by Y. Ukrainitz & B. Sarel

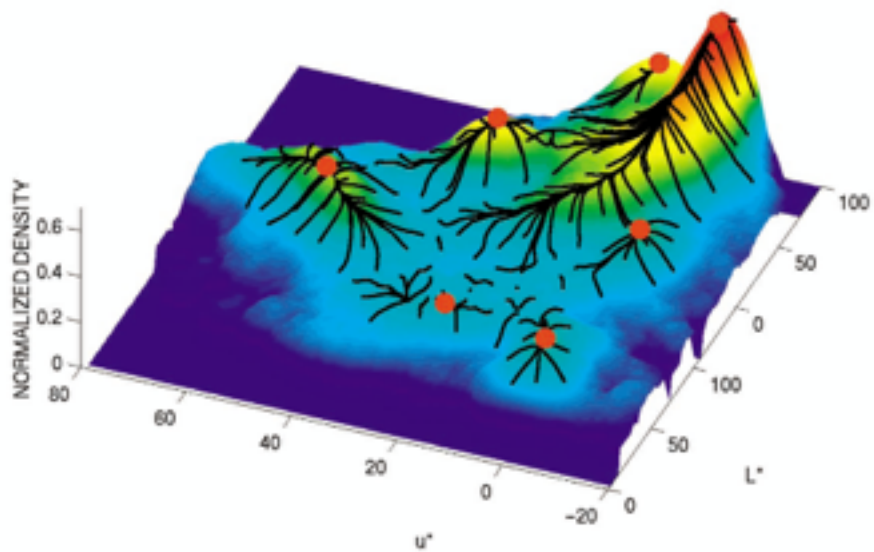
Attraction basin



(a)



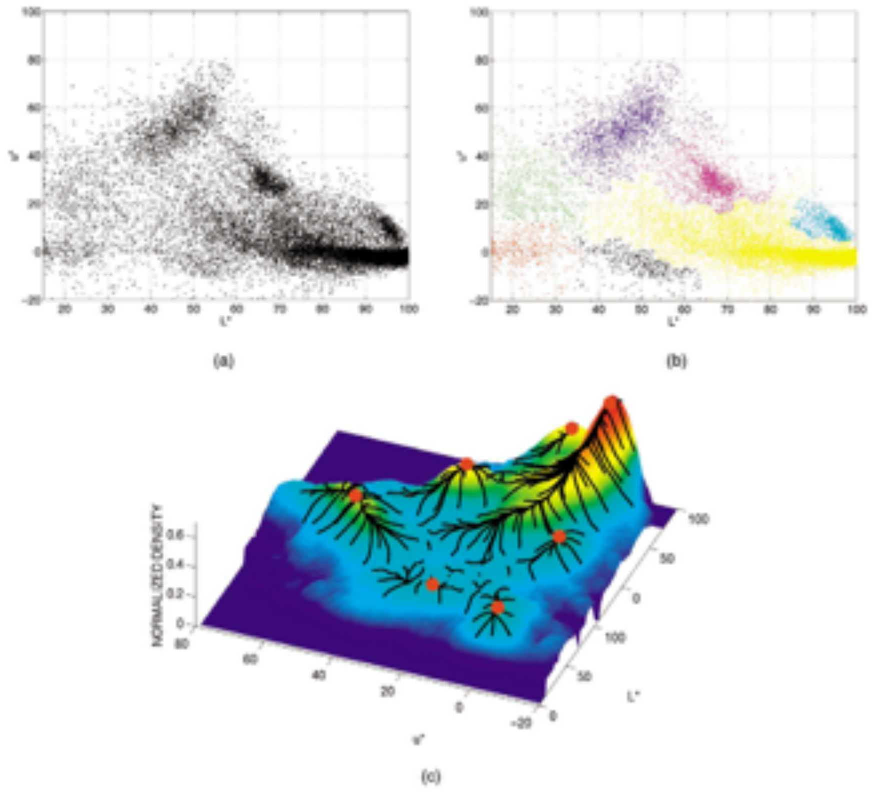
(b)



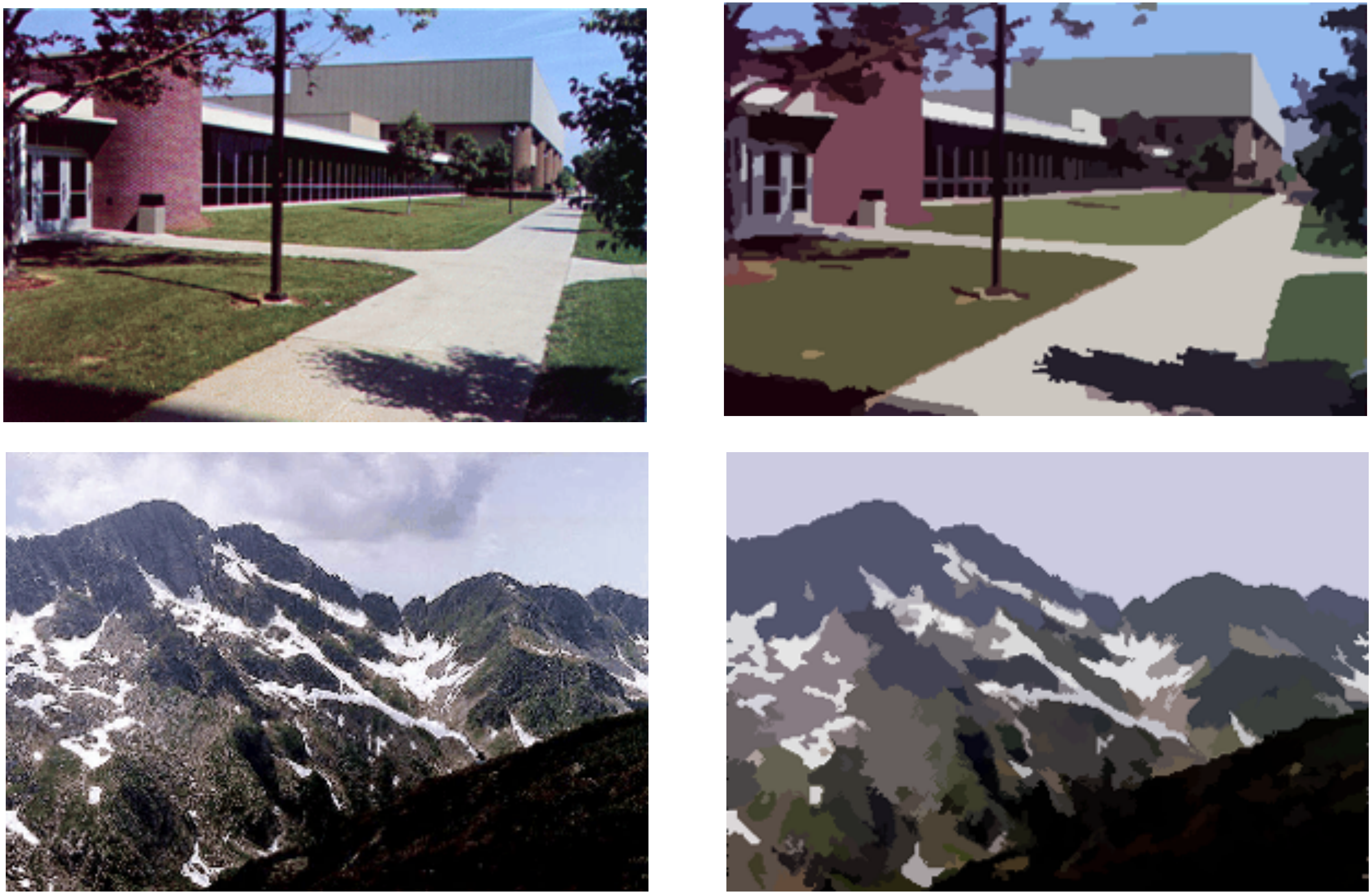
Slide by Y. Ukrainitz & B. Sarel

Segmentation by Mean Shift

- Find features (color, gradients, texture, etc)
- Plot points in a joint feature-spatial space, e.g. (u, v, R, G, B)
- 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




Mean shift pros and cons

- Pros
  - Arbitrary shape of the clusters (not necessarily spherical as in kmeans)
  - Just a small number of parameters (window size, kernel variance)
  - 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

Lecture 16
Segmentation and Scene understanding

- Introduction
- Mean-shift
- Graph-based segmentation
  - Normalized cut
  - Energy based
- Top-down segmentation

Graph-based segmentation

- Represent features and their relationships using a graph
- Cut the graph to get subgraphs with strong interior links and weaker exterior links
- Cuts correspond to segmentation boundaries

Images as graphs

- Node for every pixel
- Edge between every pair of pixels
- Each edge is weighted by the *affinity* or similarity of the two nodes

Source: S. Seitz

Measuring Affinity

Distance

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

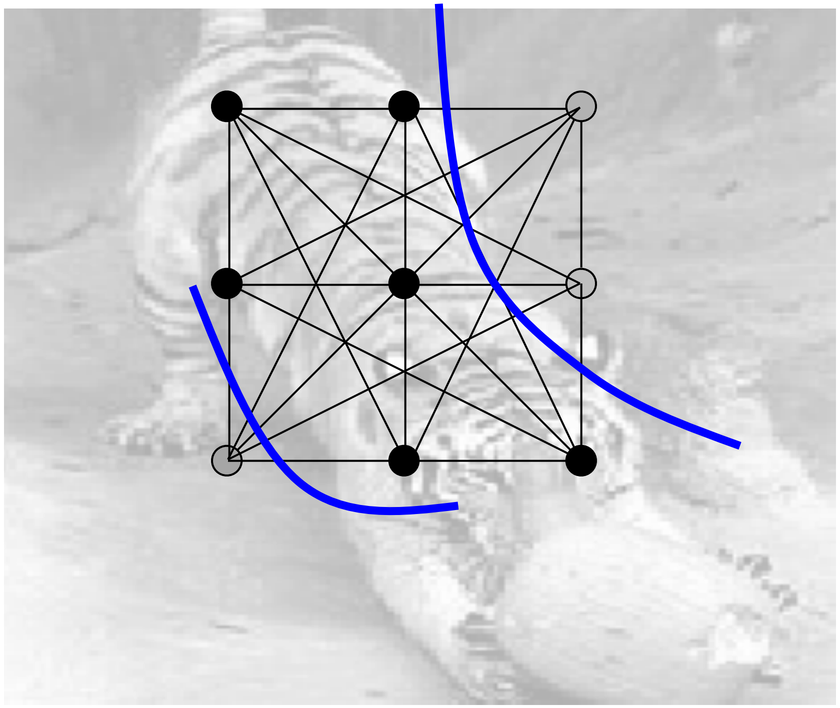
Intensity

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

Color

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

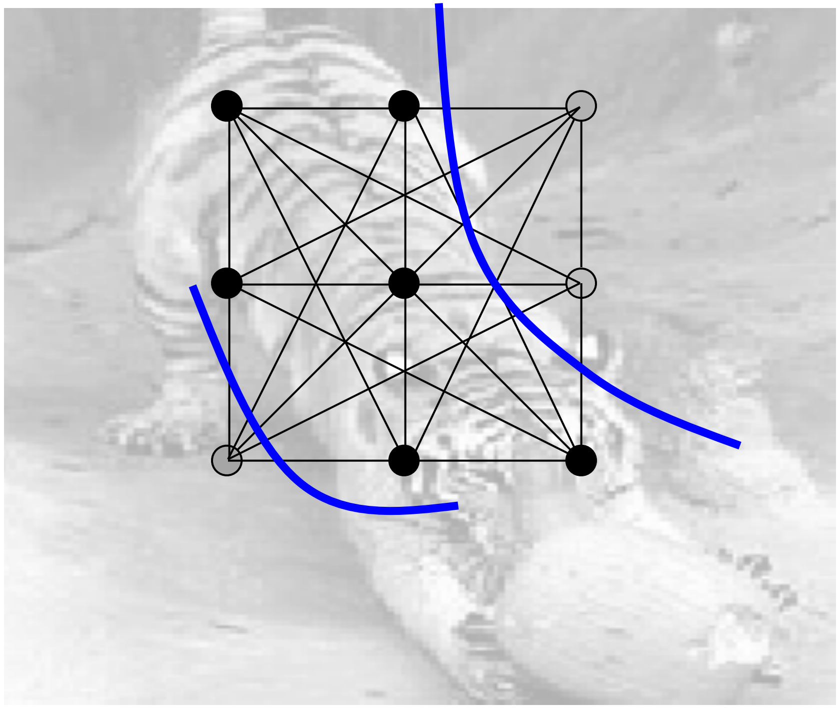
Segmentation by graph partitioning



- Break Graph into sub-graphs
  - Break links (cutting) that have low affinity
    - similar pixels should be in the same sub-graphs
    - dissimilar pixels should be in different sub-graphs

•Sub-graphs represents different image segments

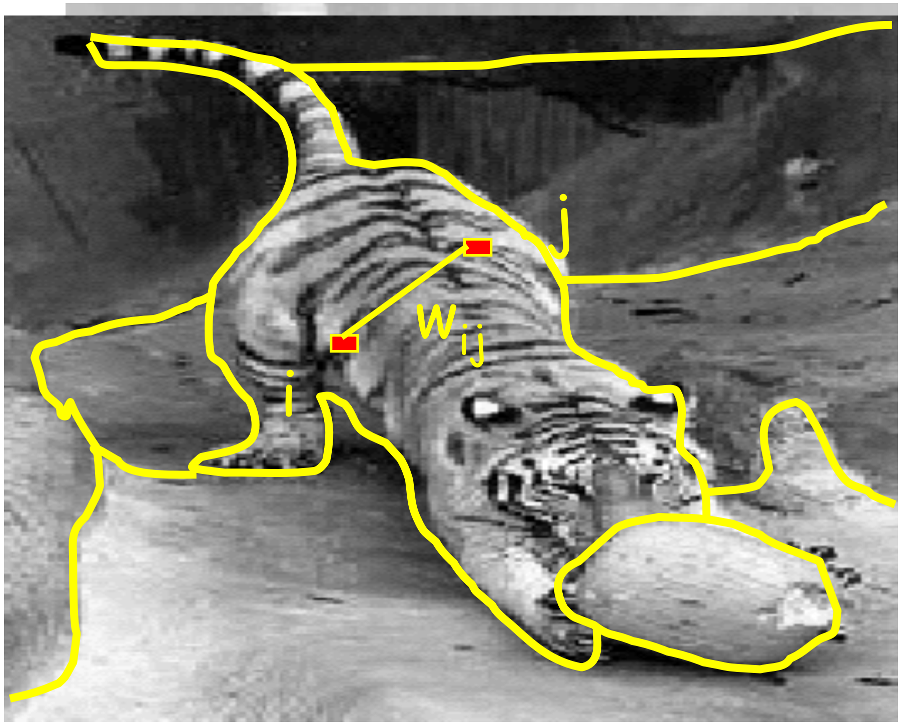
Segmentation by graph partitioning



- Break Graph into sub-graphs
  - Break links (cutting) that have low affinity
    - similar pixels should be in the same sub-graphs
    - dissimilar pixels should be in different sub-graphs

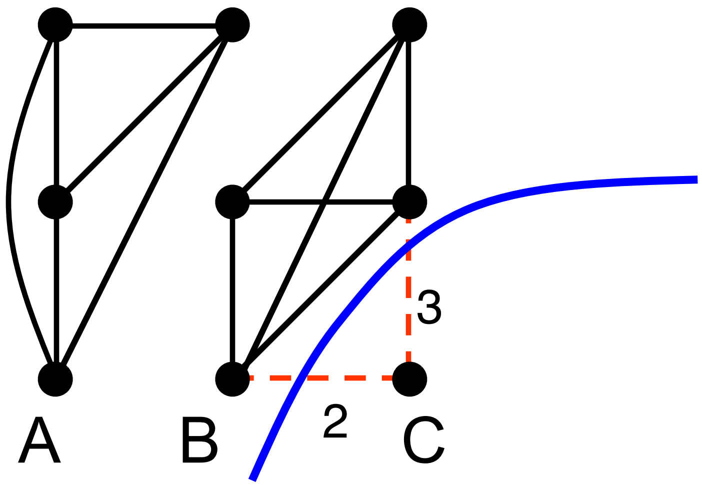
•Sub-graphs represents different image segments

Segmentation by graph partitioning



- Break Graph into sub-graphs
  - Break links (cutting) that have low affinity
    - similar pixels should be in the same sub-graphs
    - dissimilar pixels should be in different sub-graphs
- Sub-graphs represents different image segments
- Graph-cut: technique to cut a graph optimally

Segmentation by graph partitioning

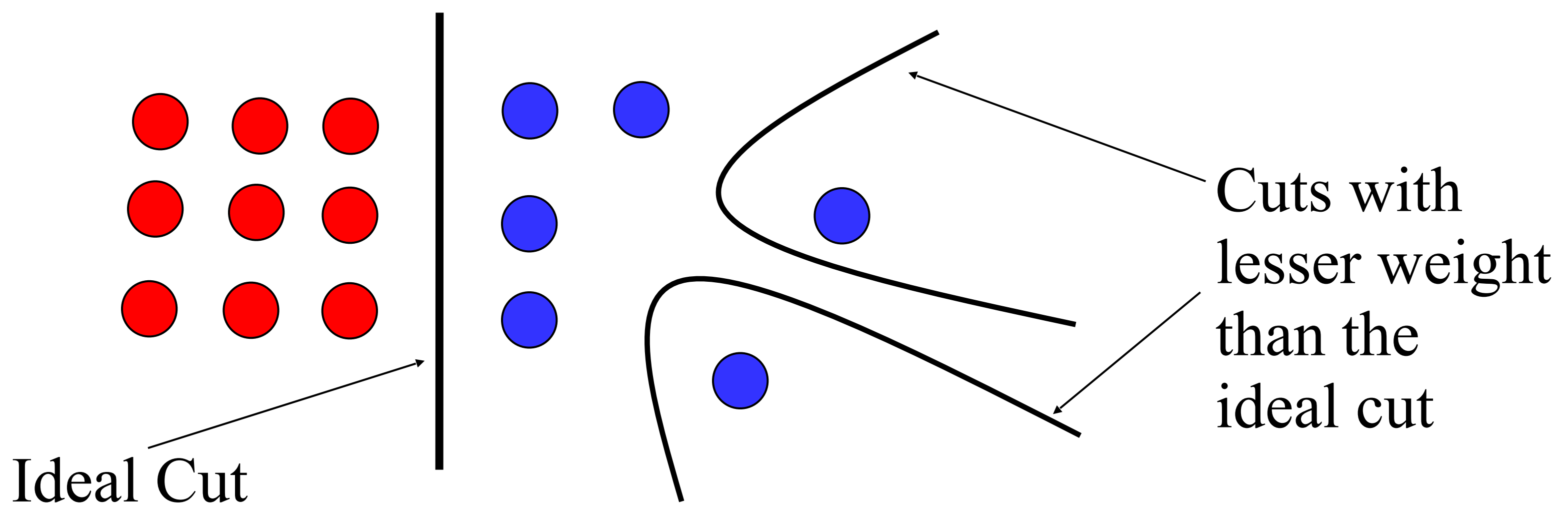


- CUT: Set of edges whose removal makes a graph disconnected
- Cost of a cut: sum of weights of cut edges
- Example: Cost of the blue cut?

Source: S. Seitz

Minimum cut

- We can do segmentation by finding the *minimum cut* in a graph (i.e. that associated to the min cost)
  - Efficient algorithms exist for doing this
- Drawback: minimum cut tends to cut off very small, isolated components



* Slide from Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Normalized cut

J. Shi and J. Malik. Normalized cuts and image segmentation. PAMI 2000

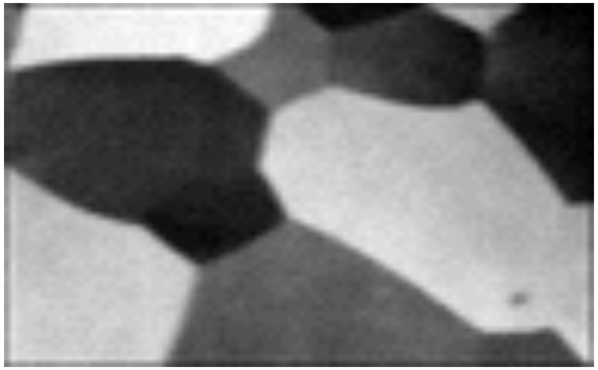
- IDEA: normalizing the cut by component size
- The *normalized cut* cost is:

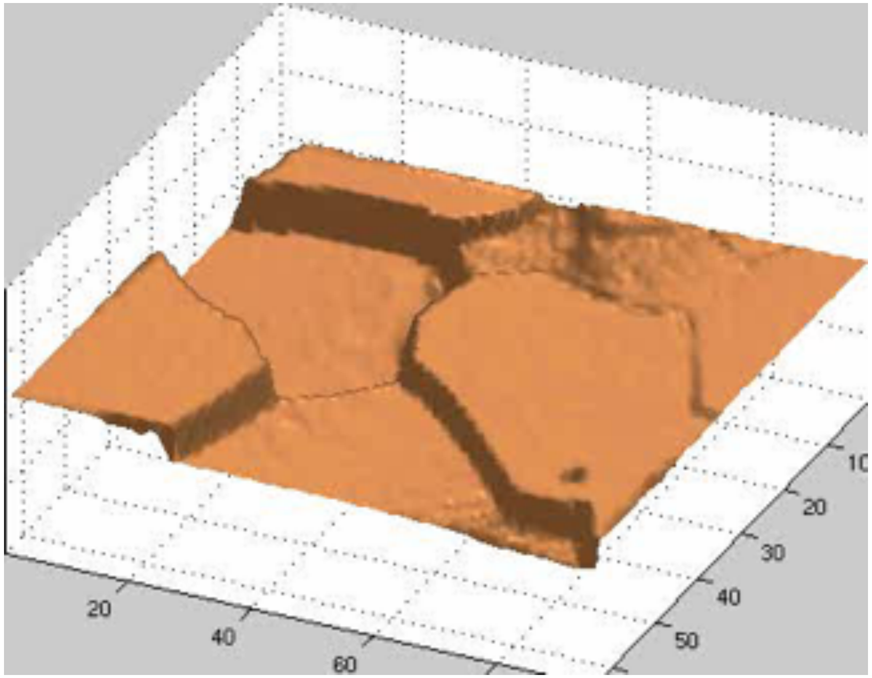
$$\frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}$$

assoc(*A*, *V*) = sum of weights of all edges in V that touch A

- The exact solution is NP-hard but an approximation can be computed

Interpretation as a Dynamical System



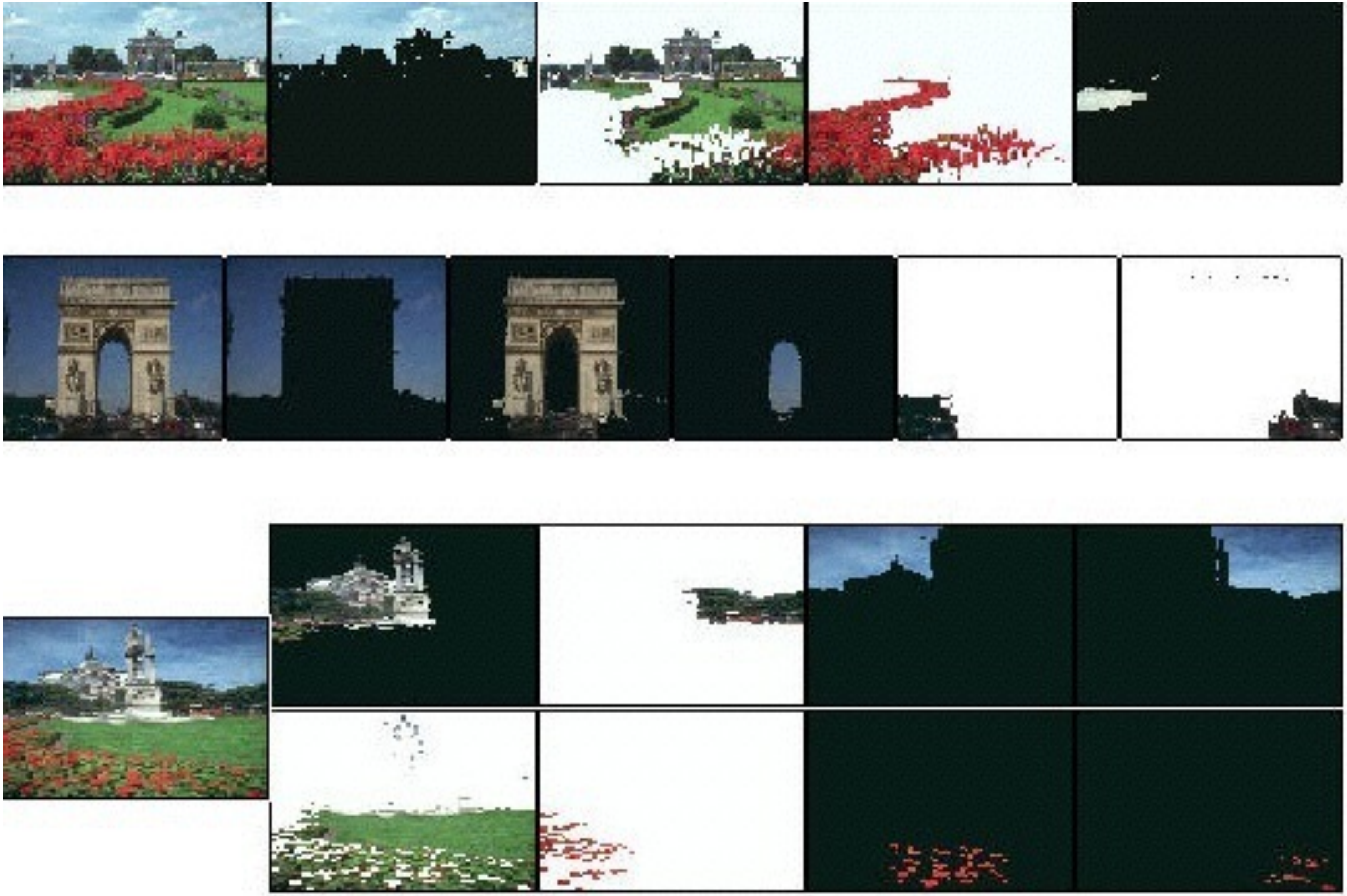


Treat the links as springs and shake the system

- elasticity proportional to cost
- vibration “modes” correspond to segments
  - can compute these by solving an eigenvector problem
  - http://www.cis.upenn.edu/~jshi/papers/pami_ncut.pdf

Courtesy of Noah Snavely

Color Image Segmentation



Normalized cuts: Pro and con

- Pros
  - Generic framework, can be used with many different features and affinity formulations
- Cons
  - High storage requirement and time complexity
  - Bias towards partitioning into equal segments

* Slide from Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Lecture 16

Segmentation and Scene understanding

- Introduction
- Mean-shift
- Graph-based segmentation
  - Graph cut
  - Energy based
- Top-down segmentation

Binary segmentation as energy minimization

- Suppose we want to segment an image into foreground and background


User sketches out a few strokes on foreground and background…

How do we classify the rest of the pixels?

Courtesy of Noah Snavely

Binary segmentation as energy minimization

- Define a labeling *L* as an assignment of each pixel with a 0-1 label (background or foreground)
- Problem statement: find the labeling *L* that minimizes

$$E(L) = \underbrace{E_d(L)}_{\text{match cost}} + \lambda \underbrace{E_s(L)}_{\text{smoothness cost}}$$

("how similar is each labeled pixel to the foreground / background?")

Courtesy of Noah Snavely

$$E(L) = E_d(L) + \lambda E_s(L)$$

$\tilde{L}(x,y)$

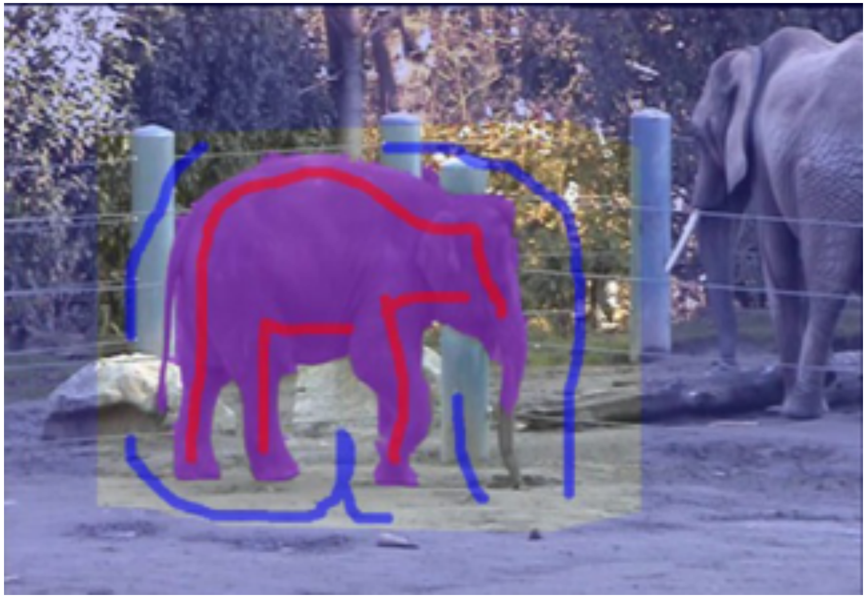
$$E_d(L) = \sum_{(x,y)} C(x,y,L(x,y))$$

$$C(x,y,L(x,y)) = \begin{cases} \infty & \text{if } L(x,y) \neq \tilde{L}(x,y) \\ C'(x,y,L(x,y)) & \text{otherwise} \end{cases}$$

$$\left.\begin{array}{ll} C'(x,y,0) & \text{: "distance" from pixel to background pixels} \\ C'(x,y,1) & \text{: "distance" from pixel to foreground pixels} \end{array}\right\}\ \text{usually computed by creating a color model from user-labeled pixels}$$

Courtesy of Noah Snavely

$$E(L) = E_d(L) + \lambda E_s(L)$$



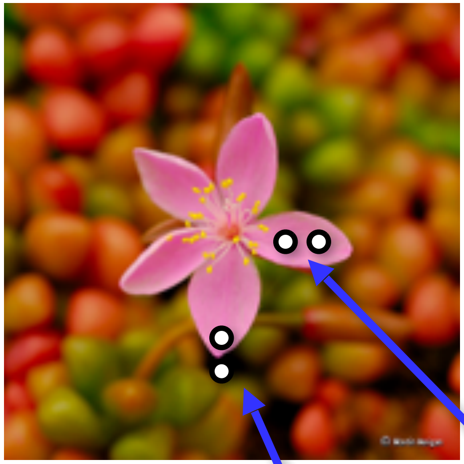
$C'(x, y, 0)$

$C'(x, y, 1)$

Courtesy of Noah Snavely

$$E(L) = E_d(L) + \lambda E_s(L)$$

- Neighboring pixels should generally have the same labels
  - Unless the pixels have very different intensities



$$E_s(L) = \sum_{\text{neighbors } (p,q)} w_{pq} |L(p) - L(q)|$$

w_{pq} : similarity in intensity of *p* and *q*

Courtesy of Noah Snavely

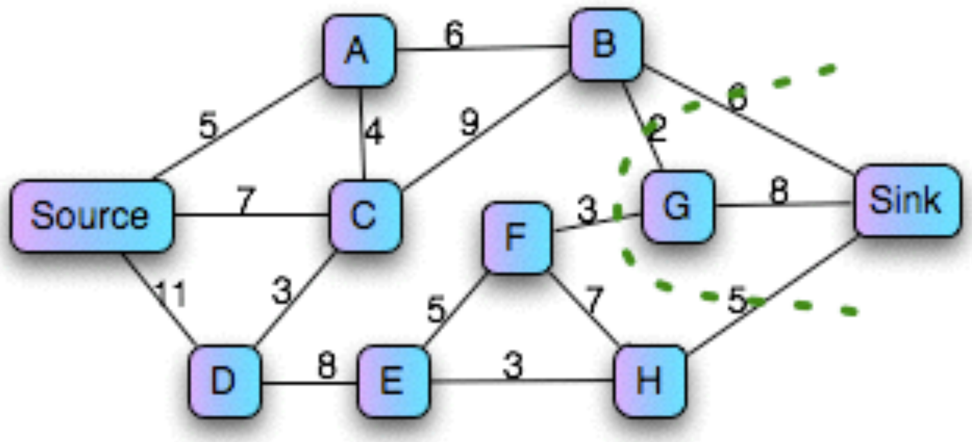
Binary segmentation as energy minimization

$$E(L) = E_d(L) + \lambda E_s(L)$$

- Require energies to be "submodular"
- For this problem, we can easily find the global minimum!
- Use max flow / min cut algorithm

Courtesy of Noah Snavely

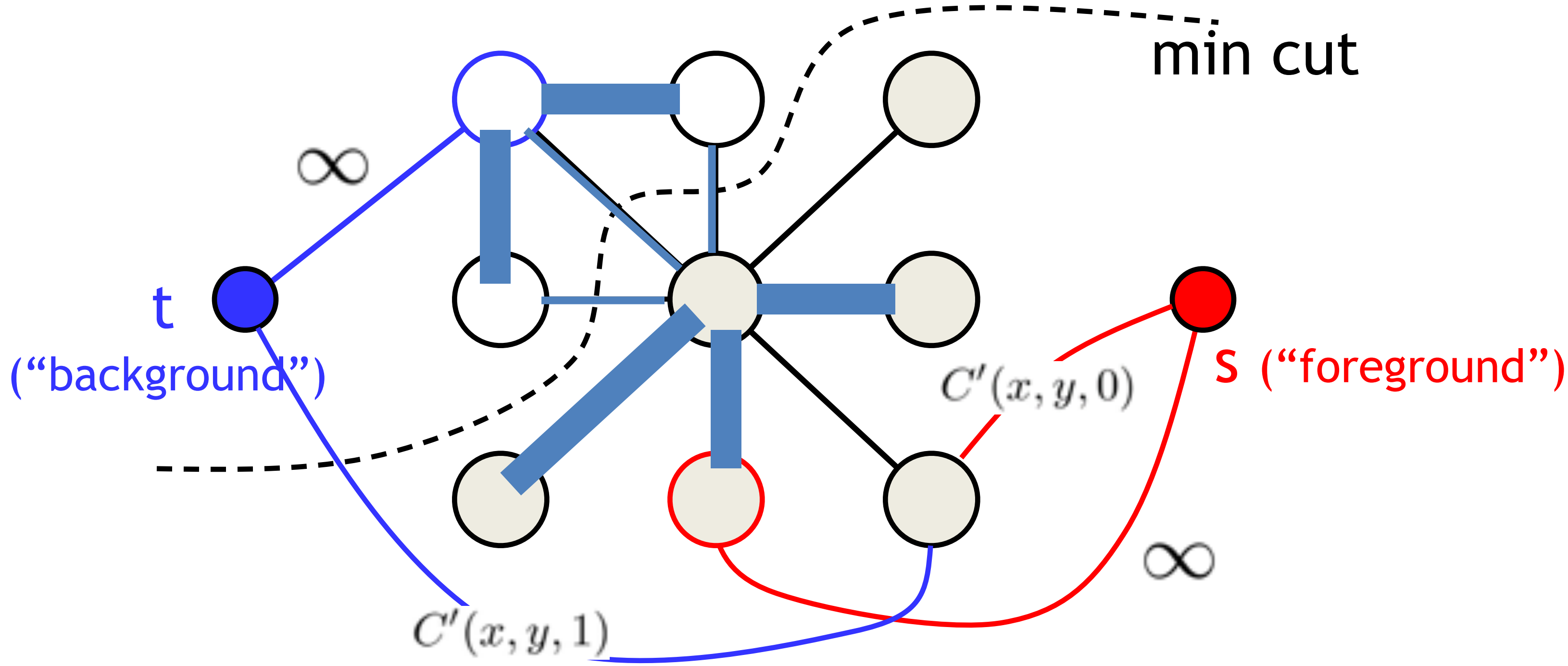
Graph min cut problem



- Given a weighted graph G with source and sink nodes (s and t), partition the nodes into two sets, S and T such that the sum of edge weights spanning the partition is minimized
  – and $s \in S$ and $t \in T$

Courtesy of Noah Snavely

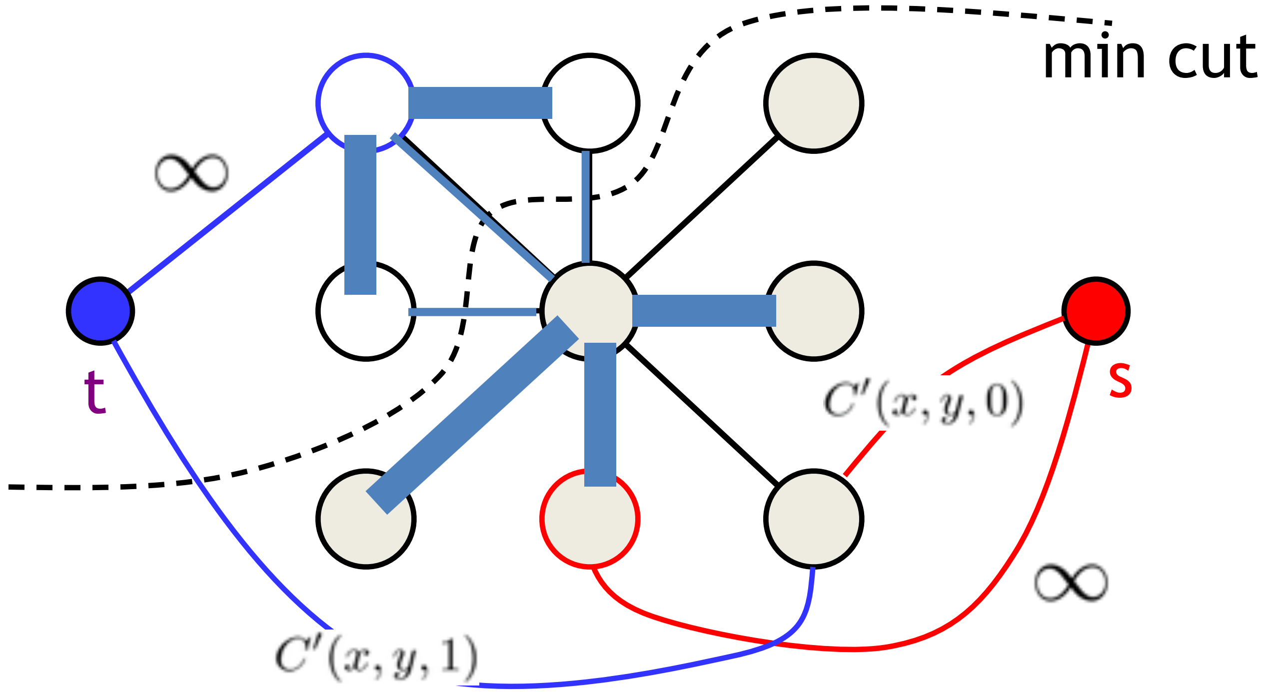
Segmentation by min cut



- Graph
  - node for each pixel, link between adjacent pixels
  - specify a few pixels as foreground and background
    - create an infinite cost link from each bg pixel to the *t* node
    - create an infinite cost link from each fg pixel to the *s* node
    - create finite cost links from *s* and *t* to each other node
  - compute min cut that separates s from t
    - The min-cut max-flow theorem [Ford and Fulkerson 1956]

Courtesy of Noah Snavely

Segmentation by min cut



- The partitions *S* and *T* formed by the min cut give the optimal foreground and background segmentation
- I.e., the resulting labels minimize

$$E(d) = E_d(d) + \lambda E_s(d)$$

Courtesy of Noah Snavely

GrabCut

Grabcut [Rother et al., SIGGRAPH 2004]

Courtesy of Noah Snavely

Graph Cuts: Pros and Cons

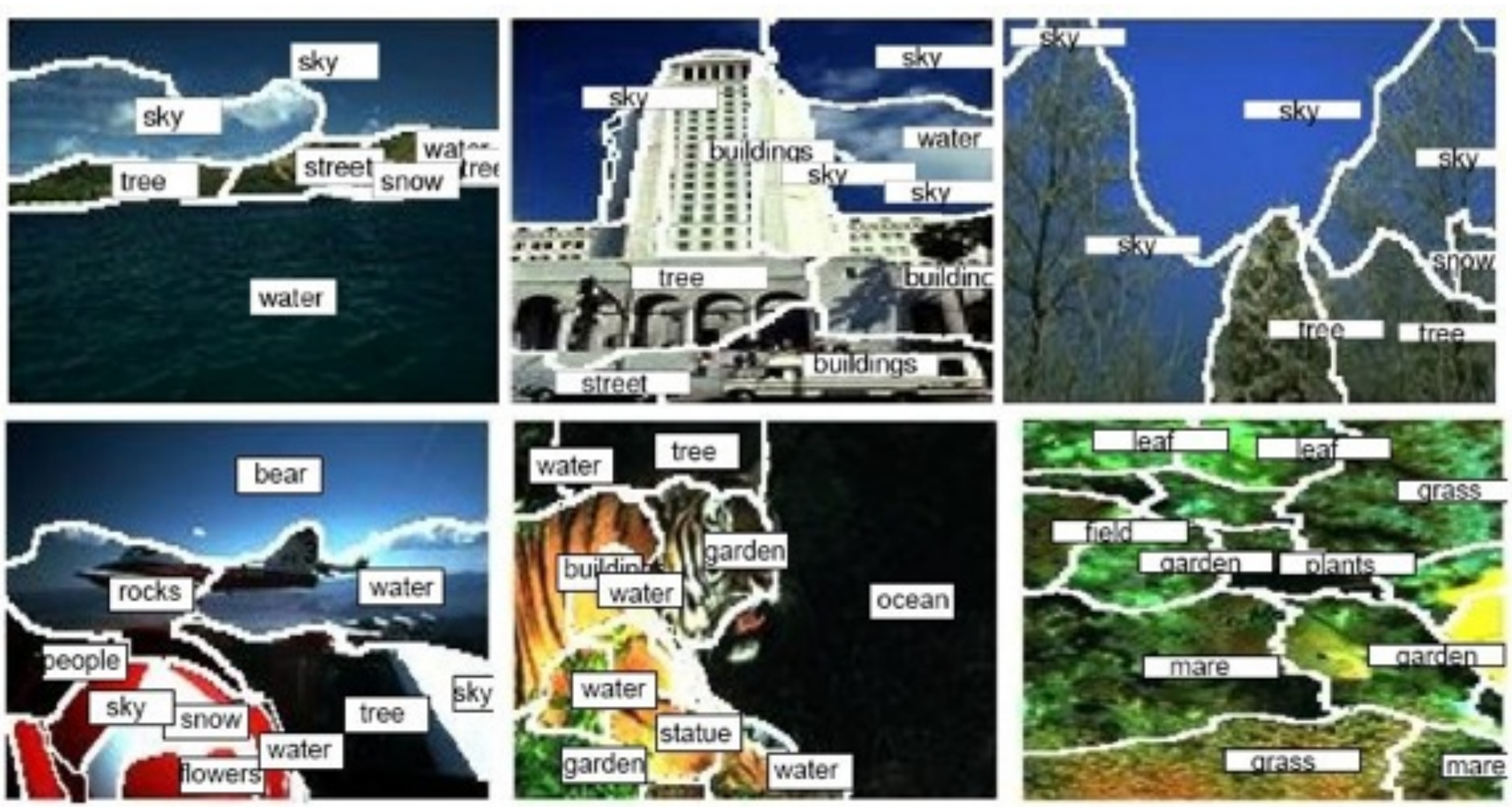
- Pros
  - Very fast inference
  - Applies to a wide range of problems (stereo, image labeling, recognition)
- Cons
  - Not always applicable (submodular energies only)
  - Need “seed” labels
- Use whenever applicable

Lecture 16
Segmentation and Scene understanding

- Introduction
- Mean-shift
- Graph-based segmentation
- Top-down segmentation

Semantic segmentation

Object Recognition as Machine Translation, Duygulu, Barnard, **de Freitas**, Forsyth, ECCV02



Semantic segmentation

Ladický, Russell, Kohli, Torr ICCV09

Next lecture

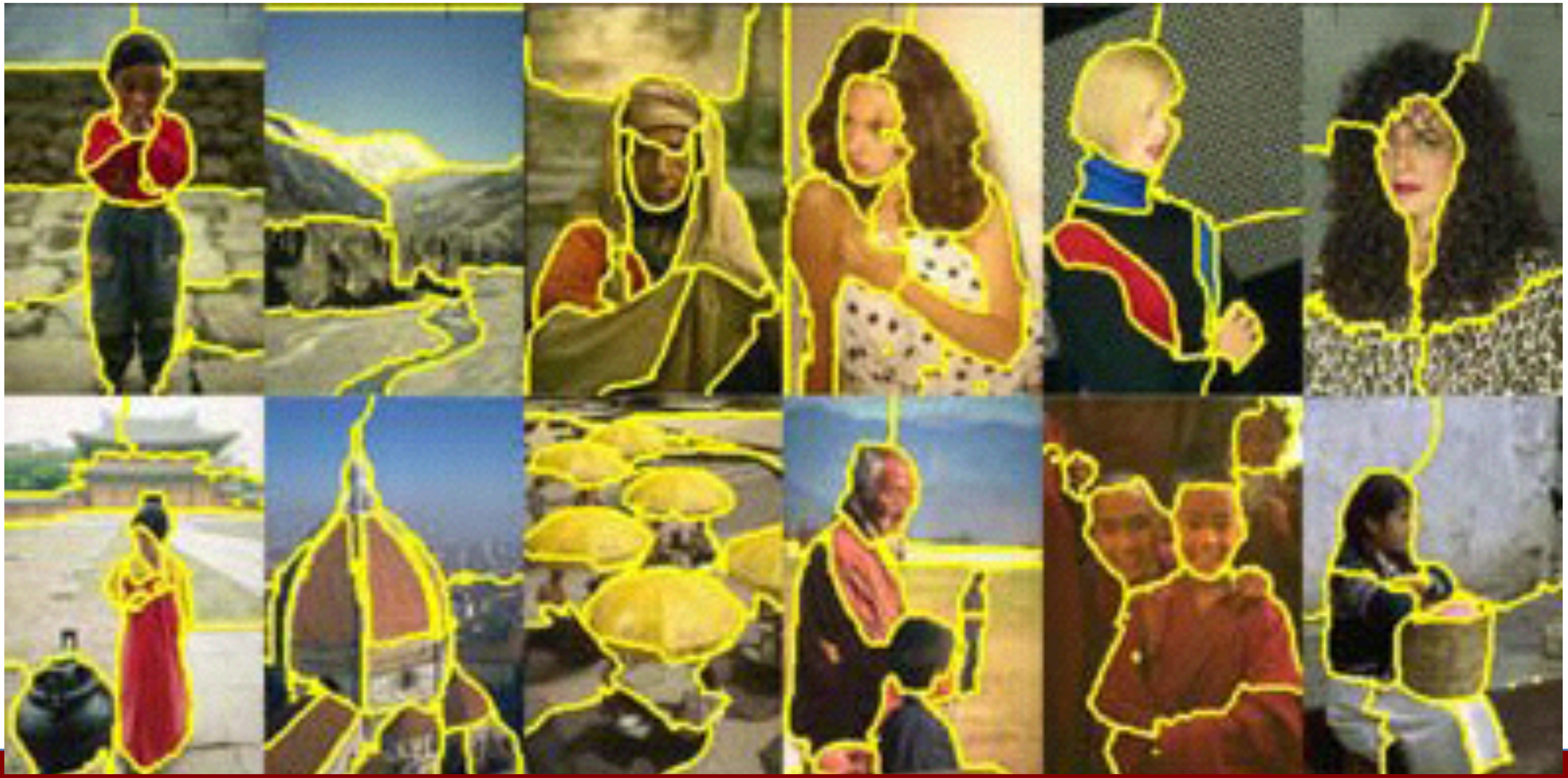
- 3D Scene Understanding

Extension to Soft Segmentation

- Each pixel is convex combination of segments.
  [Levin et al. 2006]
  - compute mattes by solving eigenvector problem

Contour and Texture Analysis for Image Segmentation

J. Malik, S. Belongie, T. Leung and J. Shi. "*Contour and Texture Analysis for Image Segmentation*". IJCV 43(1), 7-27,2001.



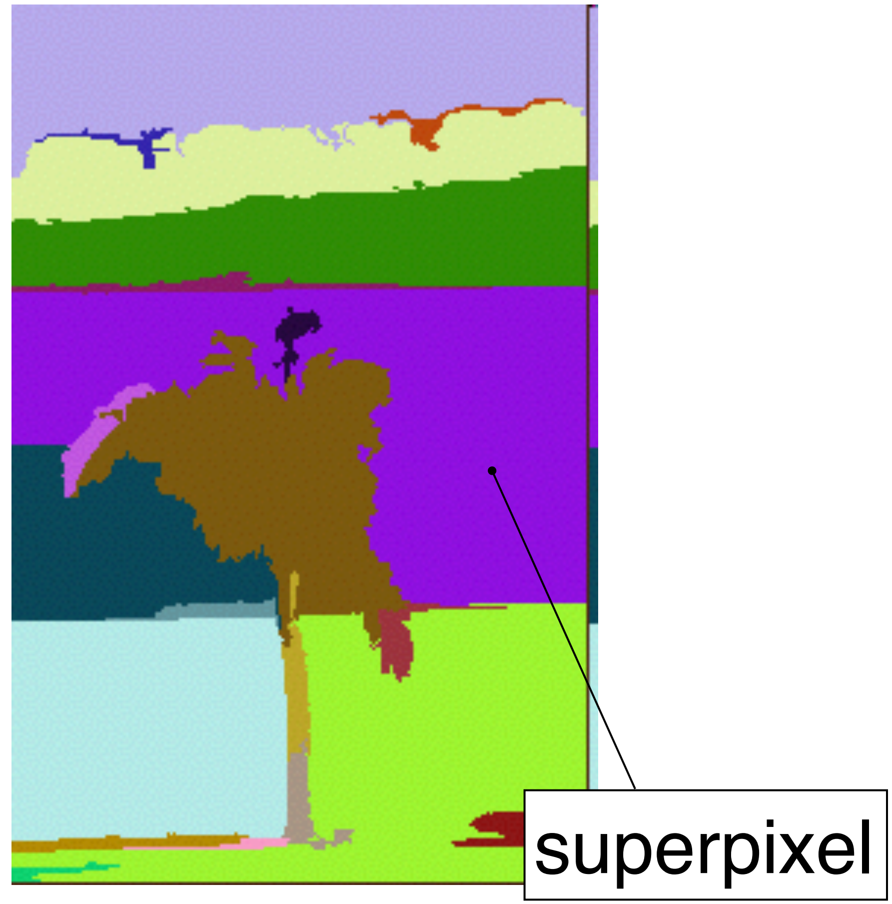
Contour and Texture Analysis for Image Segmentation

Using Contours to Detect and Localize Junctions in Natural Images"
M. Maire, P. Arbelaez, C. Fowlkes, and J. Malik. CVPR 2008

Now on CUDA

Efficient Graph-Based Image Segmentation

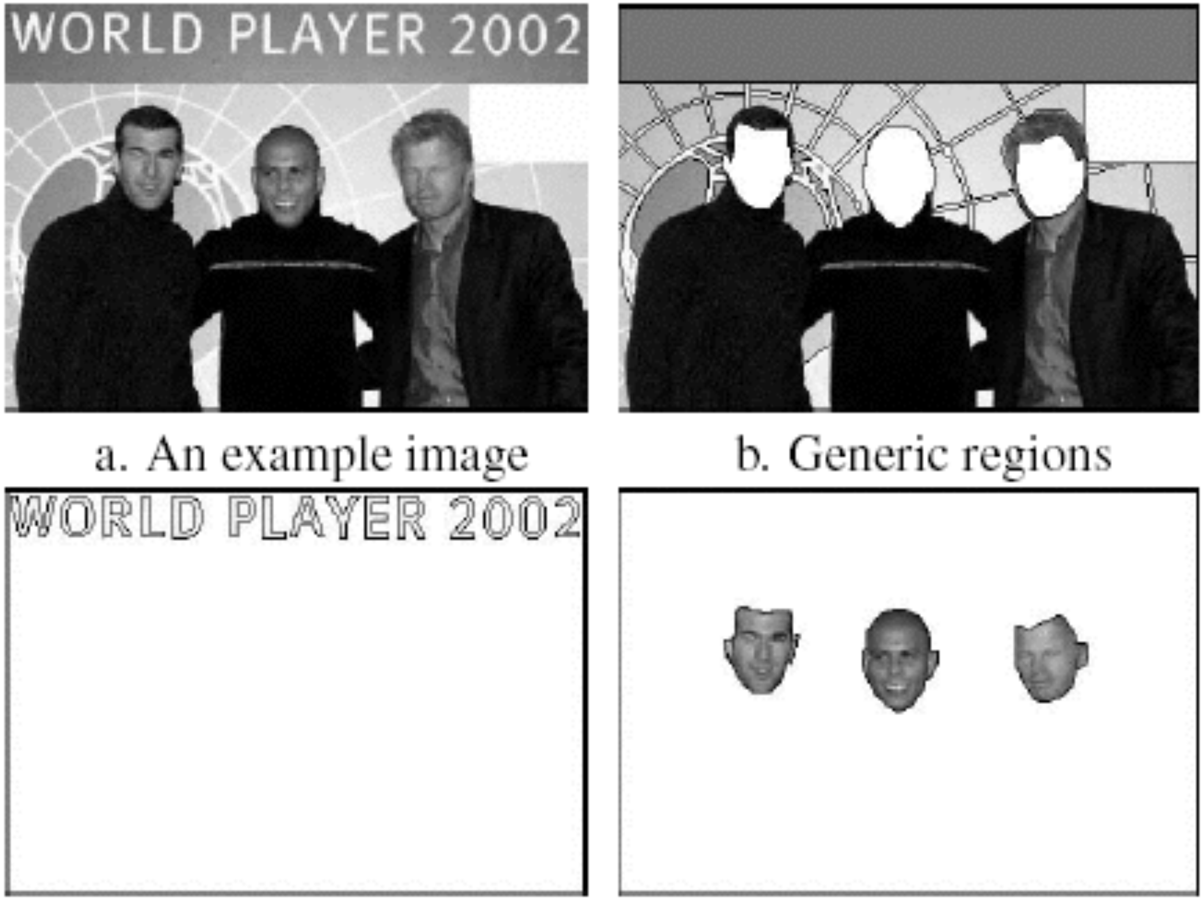
Efficient Graph-Based Image Segmentation Pedro F. Felzenszwalb and Daniel P. Huttenlocher
International Journal of Computer Vision, Volume 59, Number 2, September 2004



C++ implementation
http://people.cs.uchicago.edu/~pff/segment

Semantic segmentation

Z.W. Tu, X.R. Chen, A.L. Yuille, and S.C. Zhu. Image parsing: unifying segmentation, detection and recognition. IJCV 63(2), 113-140, 2005.



a. An example image b. Generic regions c. Text d. Faces