COLLEGE OF ENGINEERING
ELECTRICAL ENGINEERING & COMPUTER SCIENCE
UNIVERSITY OF MICHIGAN

Clustering in Computer Vision

EECS 598-08 Fall 2014
Foundations of Computer Vision

Instructor: Jason Corso (jjcorso)
web.eecs.umich.edu/~jjcorso/t/598F14

Readings: FP 6.2, 9; SZ 5.2-5.4

Date: 10/6/14

Plan

- What is Clustering? Challenges in Clustering
- Clustering (for Segmentation)
  - K-Means
  - GMMs (and Expectation-Maximization)
  - Mean-Shift
- Other uses of clustering in vision
  - Texture and Textons
  - Quantization
  - Bag of Words

What is Clustering?

- What is clustering?
  - Grouping of “objects” into meaningful categories
  - Given a representation of N objects, find k clusters based on a suitable measure of similarity.
- Data Clustering is useful in and beyond Computer Vision
  - Segmentation as clustering (today)
  - Texture modeling
  - Quantization
  - Beyond
    - Data exploration
    - Compression
    - Natural classification
- Evidently important: Google Scholar tells us that more than 1500 papers get published on clustering a year!

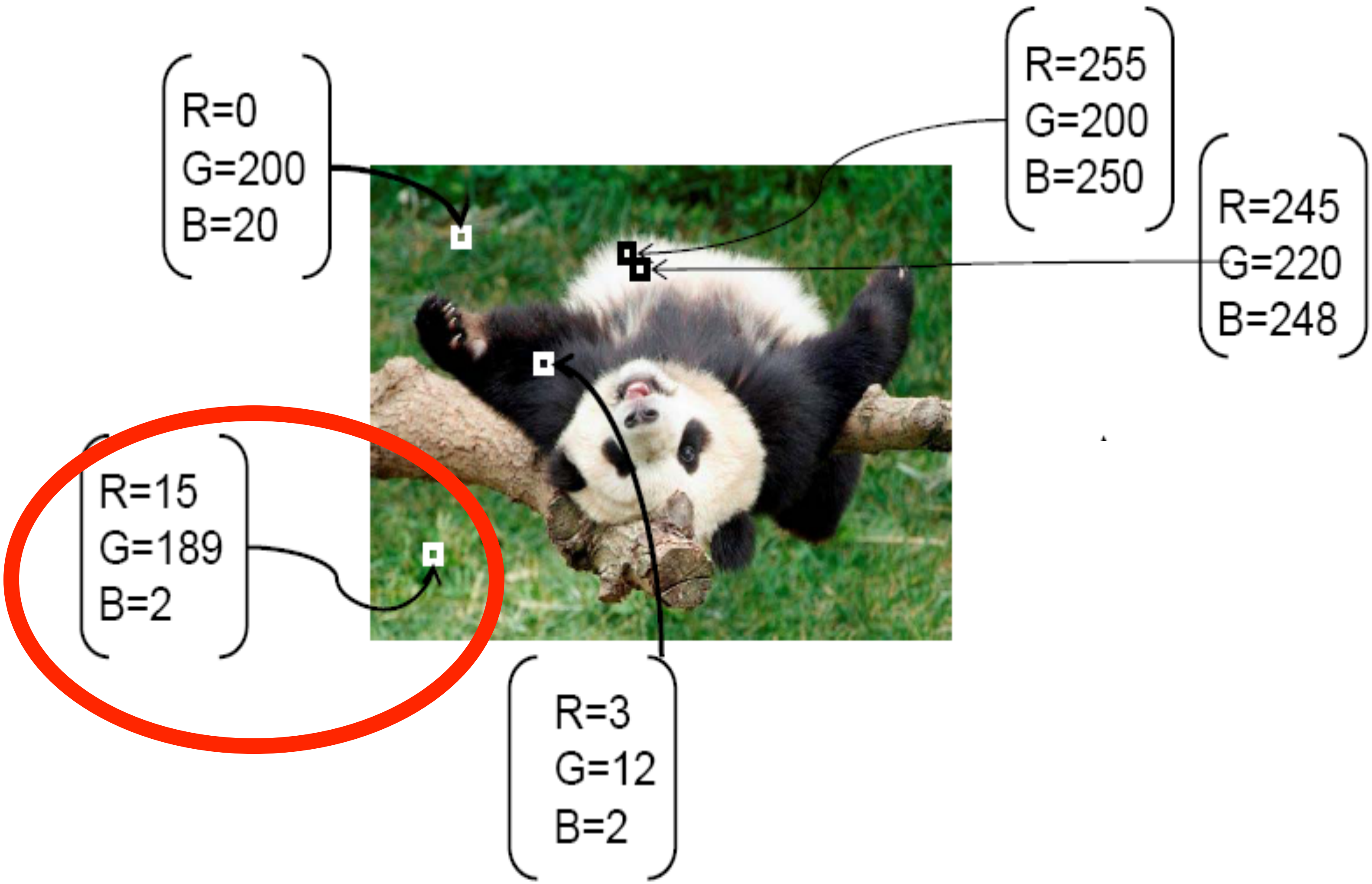
Source: A. K. Jain and R. C. Dubes. Alg. for Clustering Data, Prentice Hall, 1988.

Feature Space

- Every token is identified by a set of salient visual characteristics. For example:
  - Position
  - Color
  - Texture
  - Motion vector
  - Size, orientation (if token is larger than a pixel)

Slide credit: Christopher Rasmussen

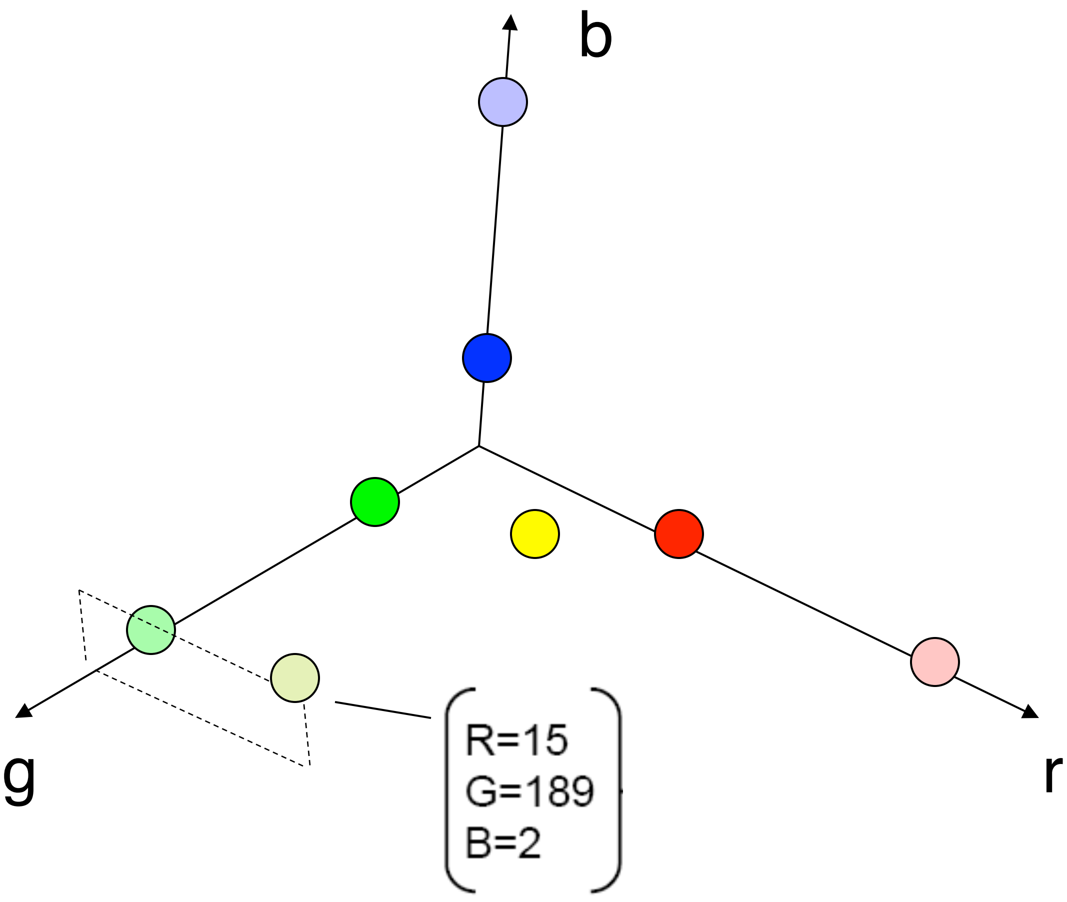
Feature Space



Source: Savarese slides.

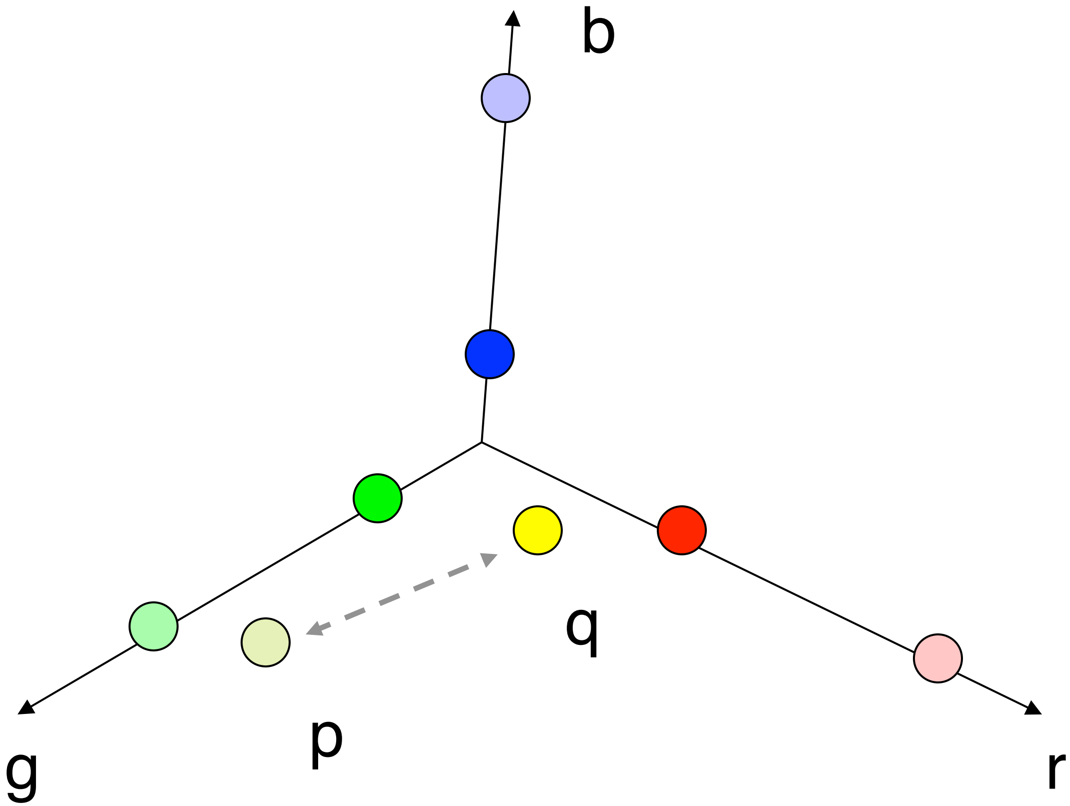
Source: K. Grauman

Feature space: each token is represented by a point



Source: Savarese slides.

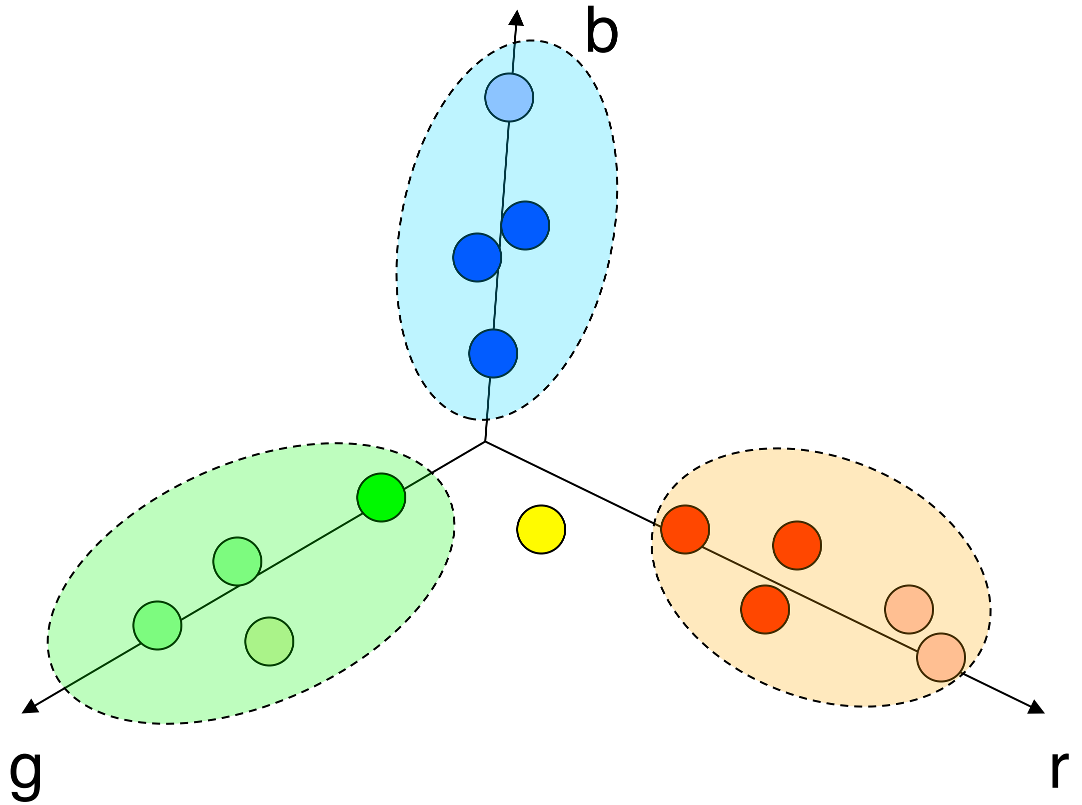
Token similarity is thus measured by distance between points ("feature vectors") in feature space



$$\sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}.$$

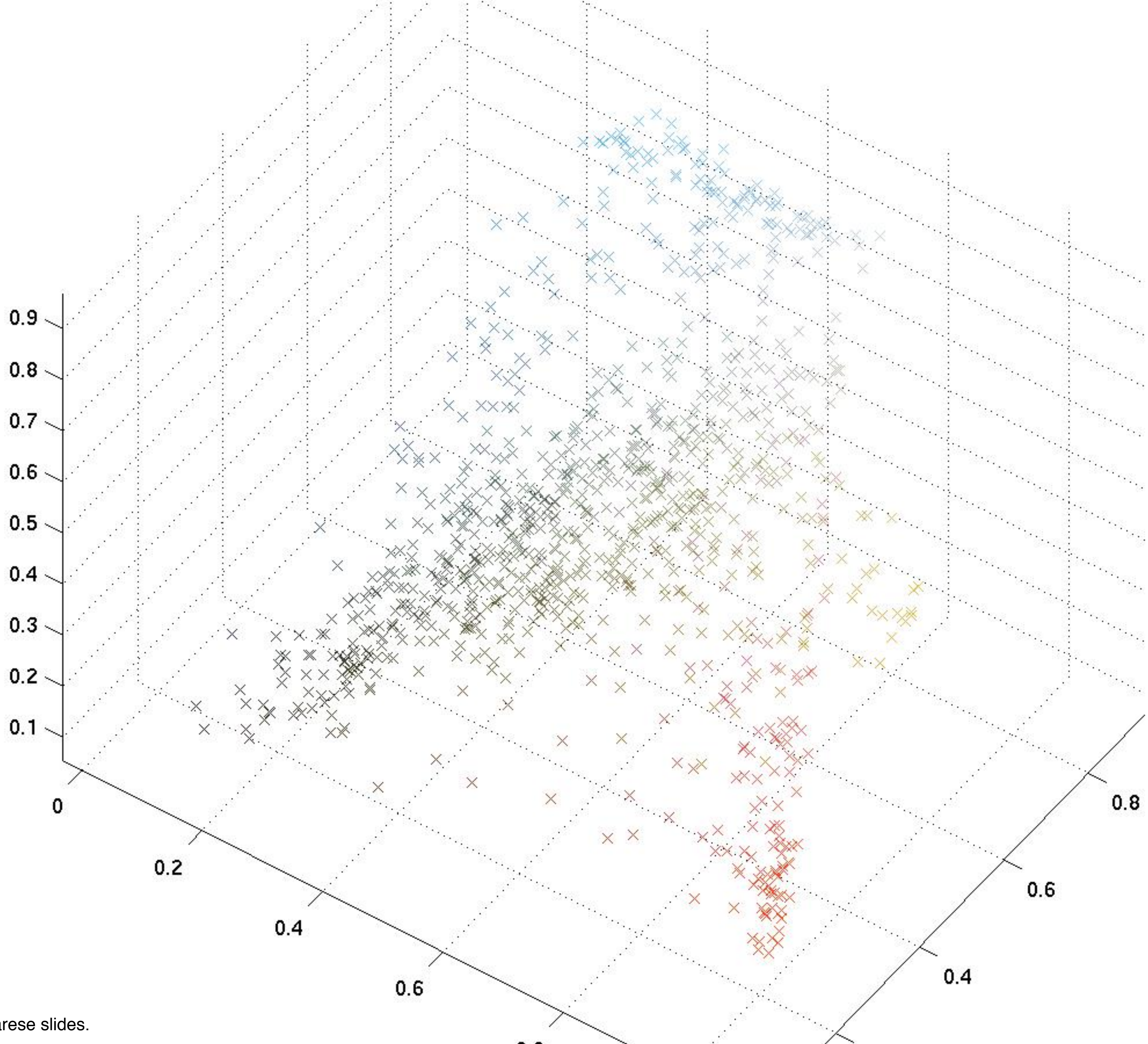
Source: Savarese slides.

Cluster together tokens with high similarity

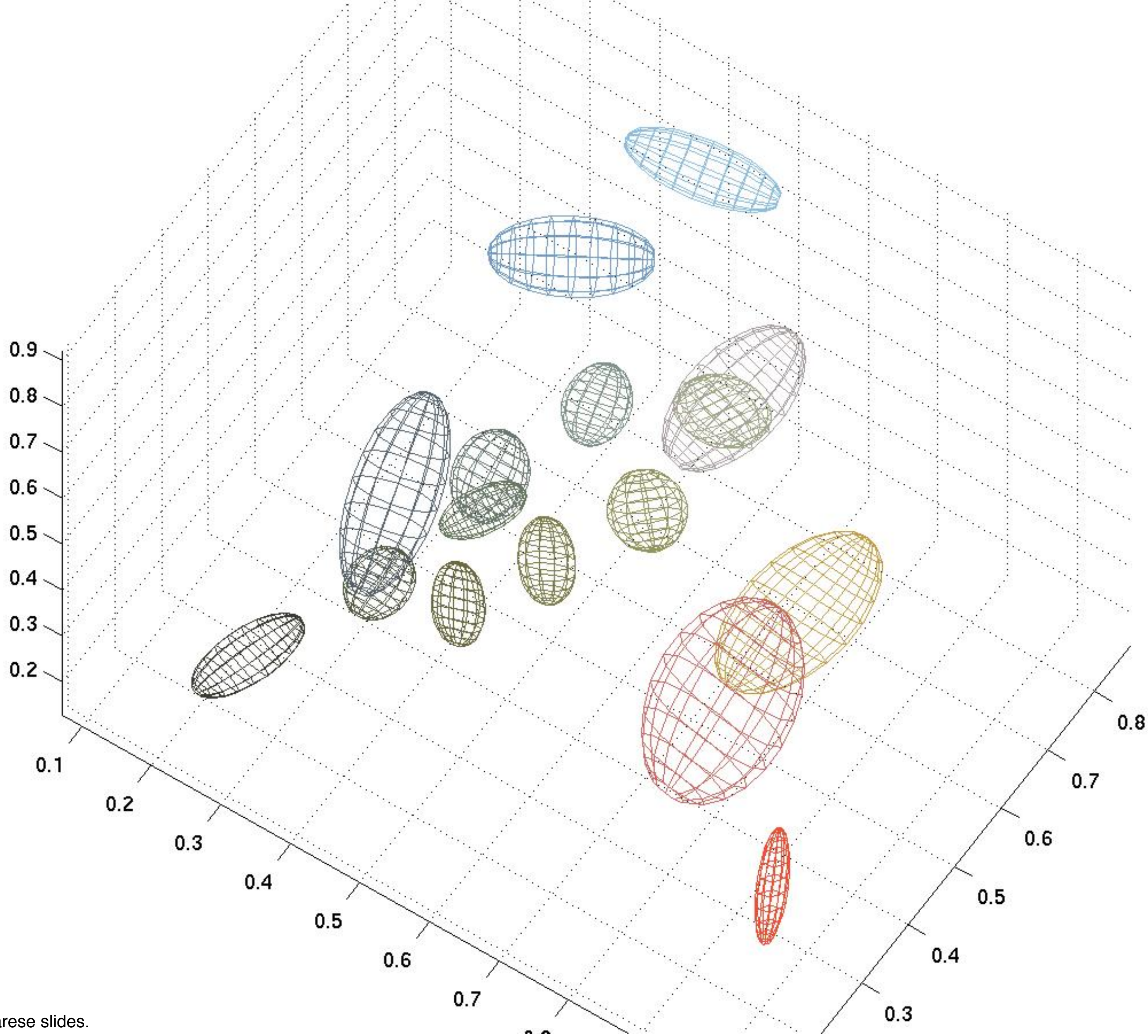


Source: Savarese slides.

Source: Savarese slides.



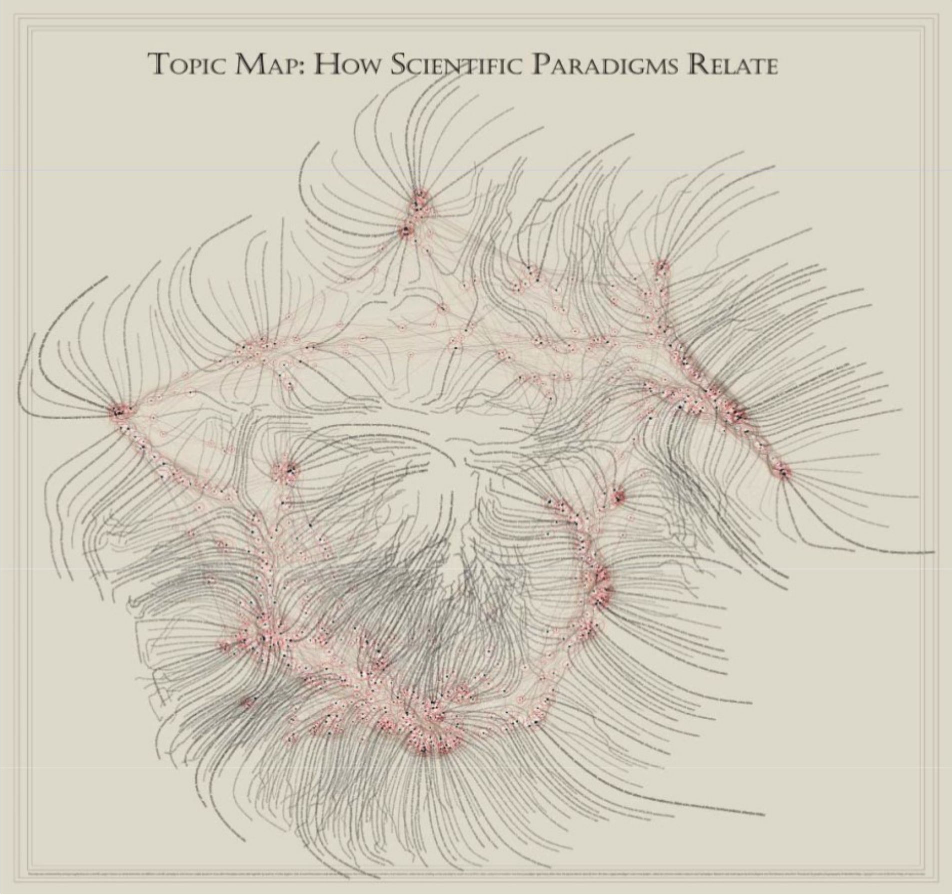
Source: Savarese slides.



Source: Savarese slides.

E.g.: Topic Discovery

- 800,000 scientific papers clustered into 776 topics based on how often the papers were cited together by authors of other papers



Source: Map of Science, Nature 2006.

Formal Definition of Clustering

- Given a set of N data samples $D = x_1, x_2, \ldots, x_N$ in a d-dimensional feature space, D is partitioned into a number of disjoint subsets D_j:

$$D = \bigcup_{j=1}^{k} D_j \quad \text{where} \quad D_i \cup D_j = \varnothing \quad \forall i \neq j$$

where the points in each subset are similar to each other according to the given similarity function.

- A partition is denoted by

$$\pi = (D_1, D_2, \ldots, D_k)$$

and clustering is then formulated as

$$\pi^* = \arg\min_{\pi} f(\pi)$$

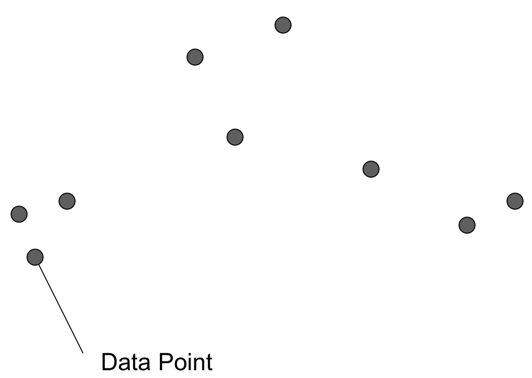
for $f(\cdot)$ that captures the desired cluster properties.

Plan

- What is Clustering? Challenges in Clustering
- Clustering (for Segmentation)
  - K-Means
  - GMMs (and Expectation-Maximization)
  - Mean-Shift
- Other uses of clustering in vision
  - Texture and Textons
  - Quantization
  - Bag of Words

K-Means Clustering

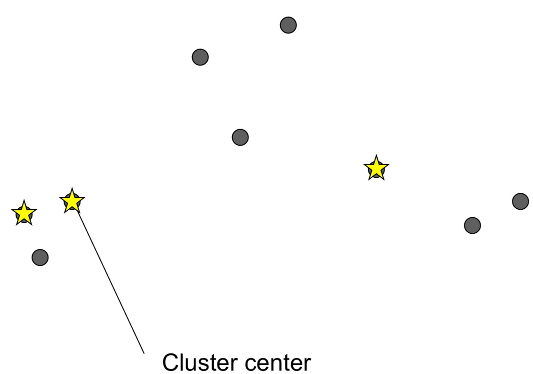
1. Randomly initialize $\mu_1, \mu_2, ..., \mu_c$
2. Repeat until no change in μ_i:
   (a) Classify N samples according to nearest μ_i
   (b) Recompute μ_i



Source: D. Aurthor, S. Vassilivitskii. “k-Means++: The Advantages of Careful Seeding”.

K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, ..., \mu_c$
2. Repeat until no change in μ_i:
   (a) Classify N samples according to nearest μ_i
   (b) Recompute μ_i



First choose *k* arbitrary centers

Source: D. Aurthor, S. Vassilivitskii. “k-Means++: The Advantages of Careful Seeding”.

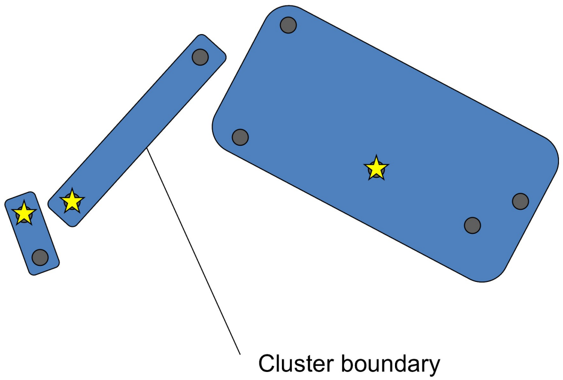
K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, ..., \mu_c$

2. Repeat until no change in μ_i:

   (a) Classify N samples according to nearest μ_i

   (b) Recompute μ_i

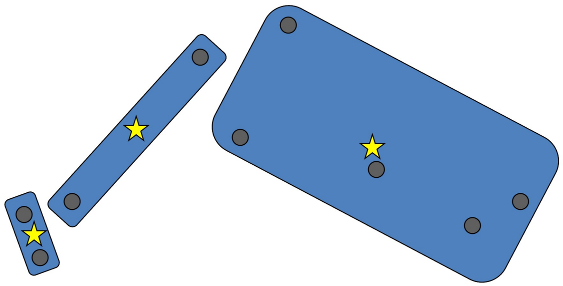


Assign points to closest centers

Source: D. Aurthor, S. Vassilivitskii. “k-Means++: The Advantages of Careful Seeding”.

K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, ..., \mu_c$
2. Repeat until no change in μ_i:
   (a) Classify N samples according to nearest μ_i
   (b) Recompute μ_i



Recompute centers

Source: D. Aurthor, S. Vassilivitskii. "k-Means++: The Advantages of Careful Seeding".

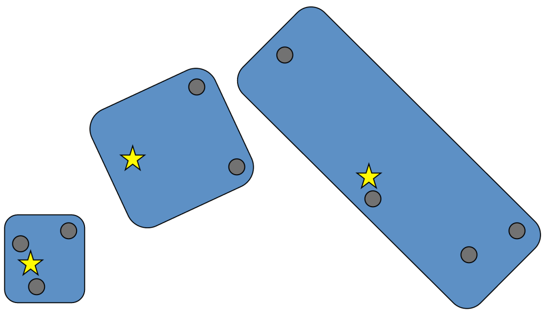
K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, ..., \mu_c$

2. Repeat until no change in μ_i:

   (a) Classify N samples according to nearest μ_i

   (b) Recompute μ_i



Assign points to closest centers

Source: D. Aurthor, S. Vassilivitskii. “k-Means++: The Advantages of Careful Seeding”.

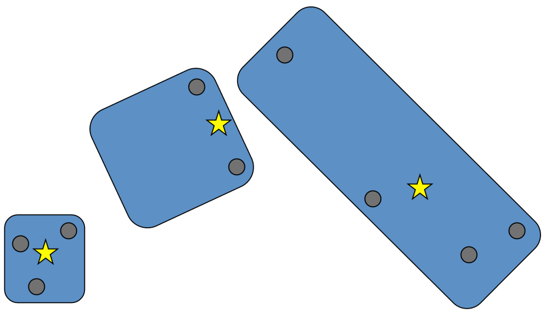
K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in μ_i:

   (a) Classify N samples according to nearest μ_i

   (b) Recompute μ_i

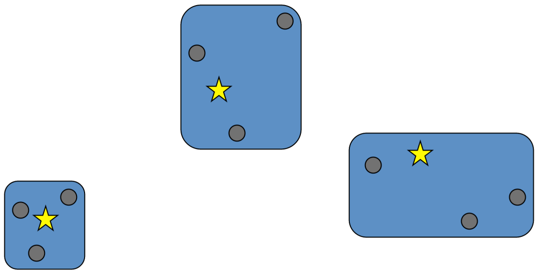


Recompute centers

Source: D. Aurthor, S. Vassilivitskii. "k-Means++: The Advantages of Careful Seeding".

K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$
2. Repeat until no change in μ_i:
   (a) Classify N samples according to nearest μ_i
   (b) Recompute μ_i



Assign points to closest centers

Source: D. Aurthor, S. Vassilivitskii. “k-Means++: The Advantages of Careful Seeding”.

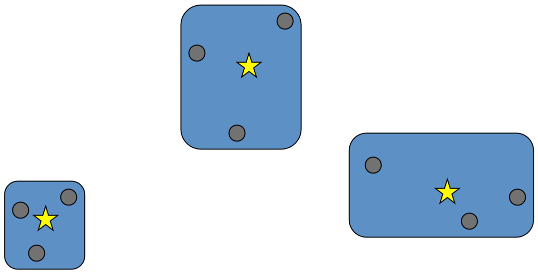
K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, ..., \mu_c$

2. Repeat until no change in μ_i:

   (a) Classify N samples according to nearest μ_i

   (b) Recompute μ_i



Recompute centers

Source: D. Aurthor, S. Vassilivitskii. “k-Means++: The Advantages of Careful Seeding”.

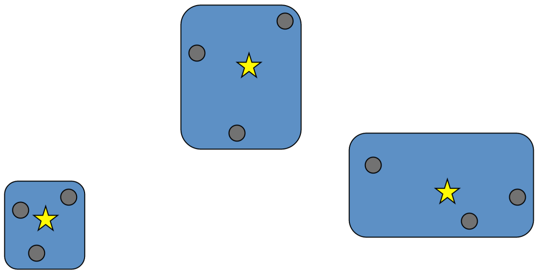
K-Means Clustering

1. Randomly initialize $\mu_1, \mu_2, \ldots, \mu_c$

2. Repeat until no change in μ_i:

   (a) Classify N samples according to nearest μ_i

   (b) Recompute μ_i



Points already assigned to nearest centers: Algorithm ends

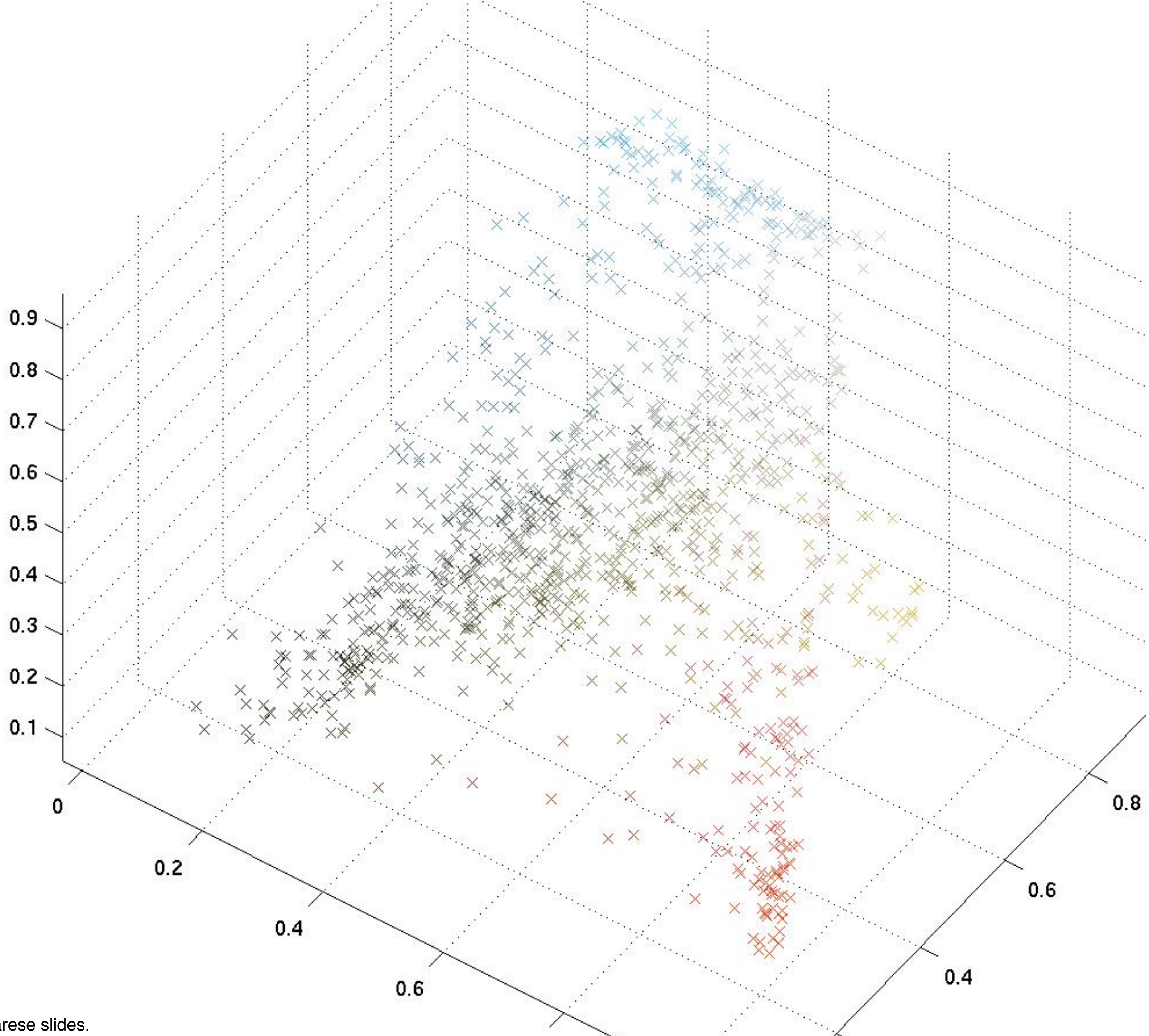
Source: D. Aurthor, S. Vassilivitskii. “k-Means++: The Advantages of Careful Seeding”.

K-Means++ Clustering

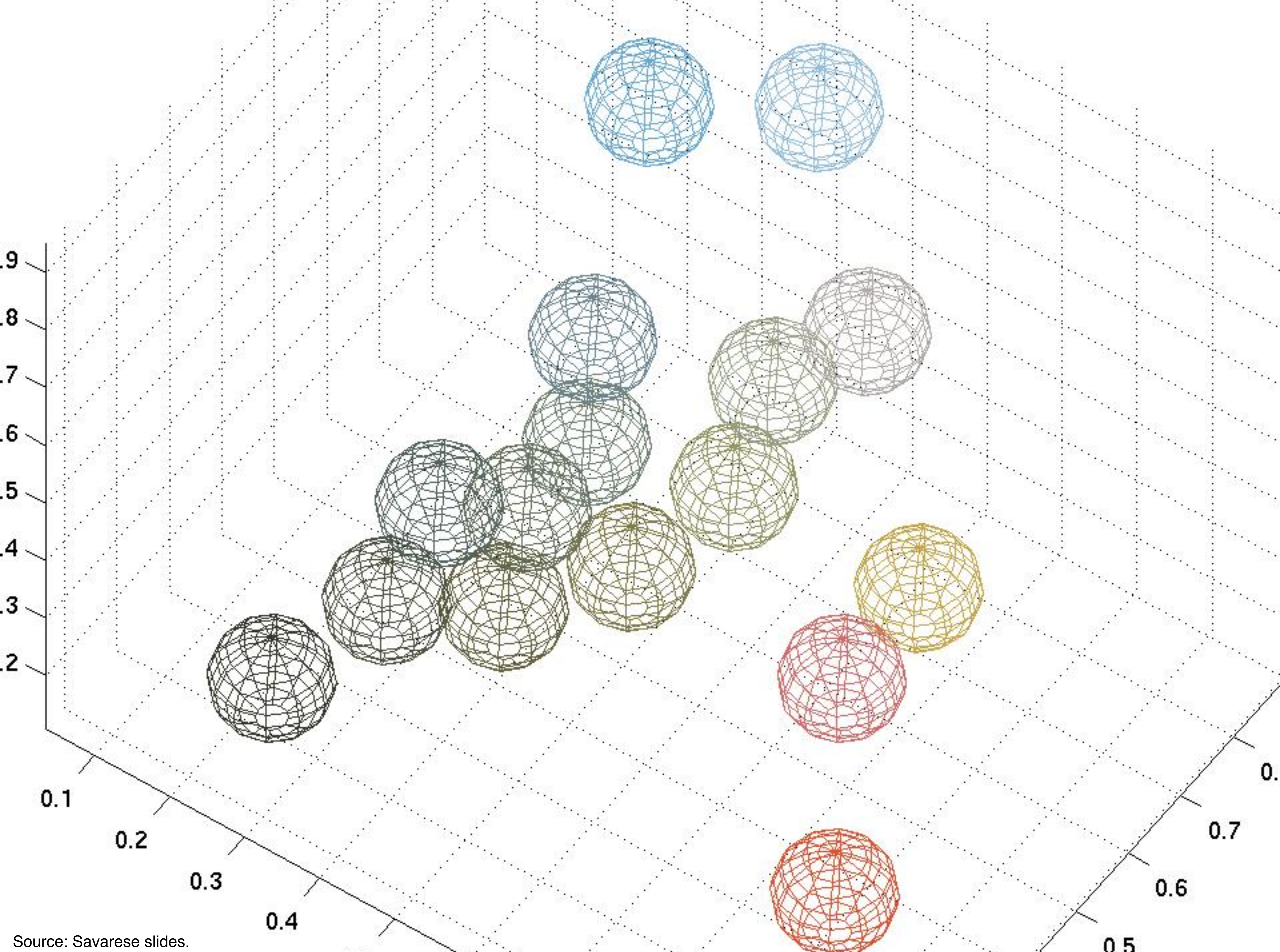
- Choose starting centers iteratively.
- Let $D(x)$ be the distance from x to the nearest existing center, take x as new center with probability $\propto D(x)^2$.
- Repeat until no change in μ_i:
  - Classify N samples according to nearest μ_i
  - Recompute μ_i

Source: D. Aurthor, S. Vassilivitskii. “k-Means++: The Advantages of Careful Seeding”.

Source: Savarese slides.



Source: Savarese slides.



Source: Savarese slides.

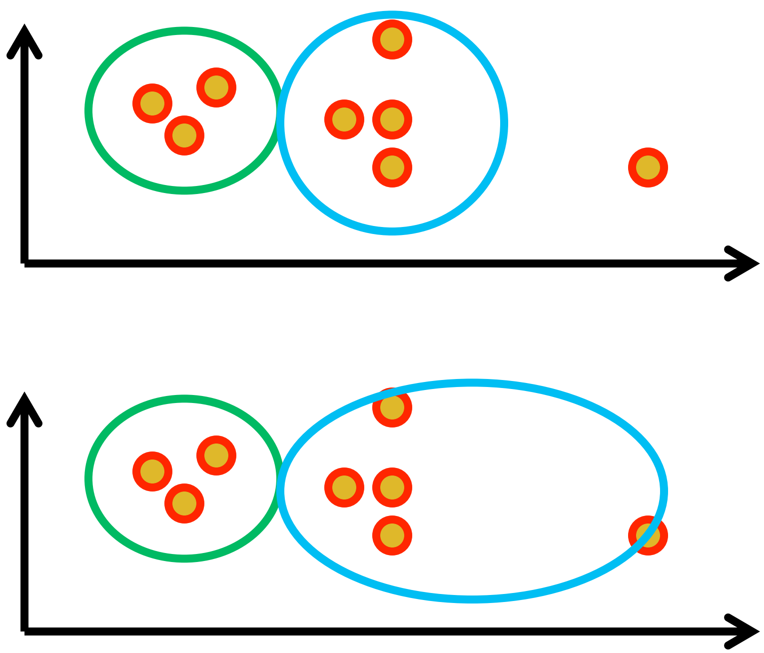
Source: Savarese slides.

K-Means pros and cons

- Pros
  - Simple and fast
  - (Always) converges to a local minimum of the error function
  - Available implementations (e.g., in Matlab)

$$\sum_{\text{clusters } i} \quad \sum_{\text{points p in cluster } i} ||p - c_i||^2$$

- Cons
  - Need to pick K
  - Sensitive to initialization
  - Only finds "spherical" clusters
  - Sensitive to outliers



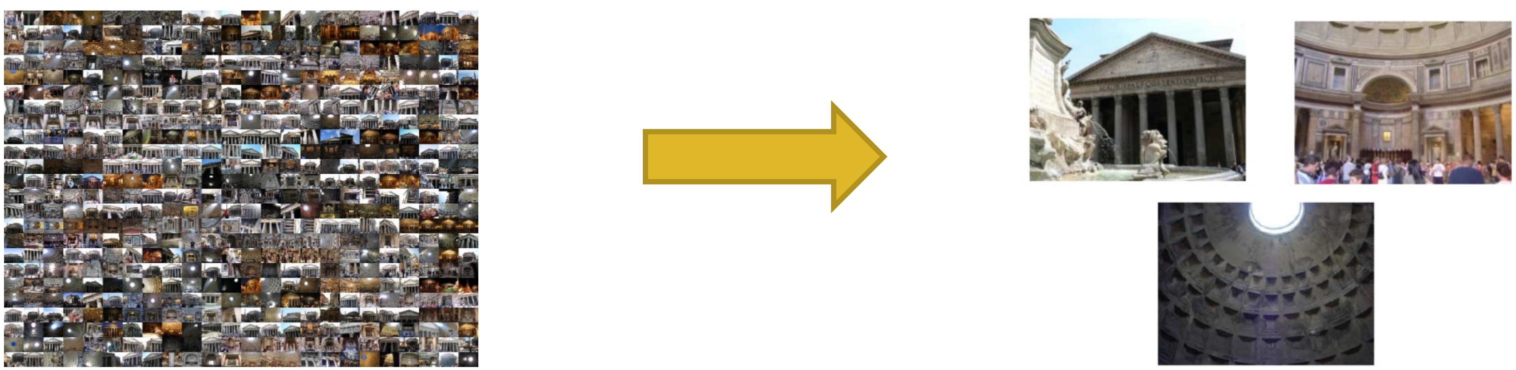
Source: Savarese slides.

Choosing Exemplars (Medoids)

- like k-means, but means must be data points

- Algorithms:
  - greedy k-means
  - affinity propagation (Frey & Dueck 2007)
  - medoid shift (Sheikh et al. 2007)

- Scene Summarization



Source: Seitz slides.

Plan

- What is Clustering? Challenges in Clustering
- Clustering (for Segmentation)
  - K-Means
  - GMMs (and Expectation-Maximization)
  - Mean-Shift
- Other uses of clustering in vision
  - Texture and Textons
  - Quantization
  - Bag of Words

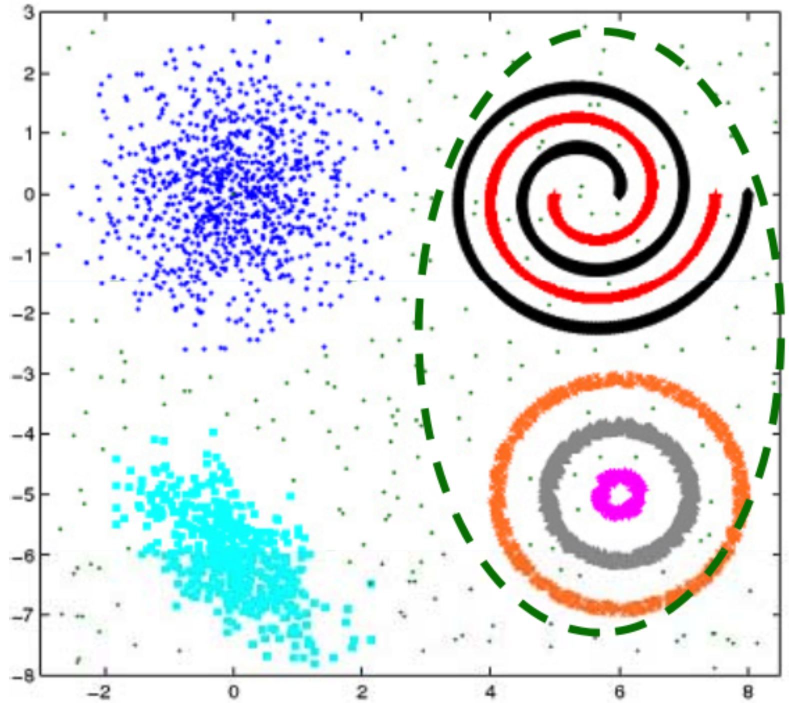
User's Dilemma

1. What is a cluster?
2. How to define pair-wise similarity?
3. Which features? Which normalizations scheme?
4. How many clusters?
5. Which clustering method?
6. Are the discovered clusters and partitioning valid?
7. Does the data have any clustering tendency?

Source: R. Dubes and A. K. Jain. "Clustering Techniques: User's Dilemma" PR 1976.

Cluster Similarity?

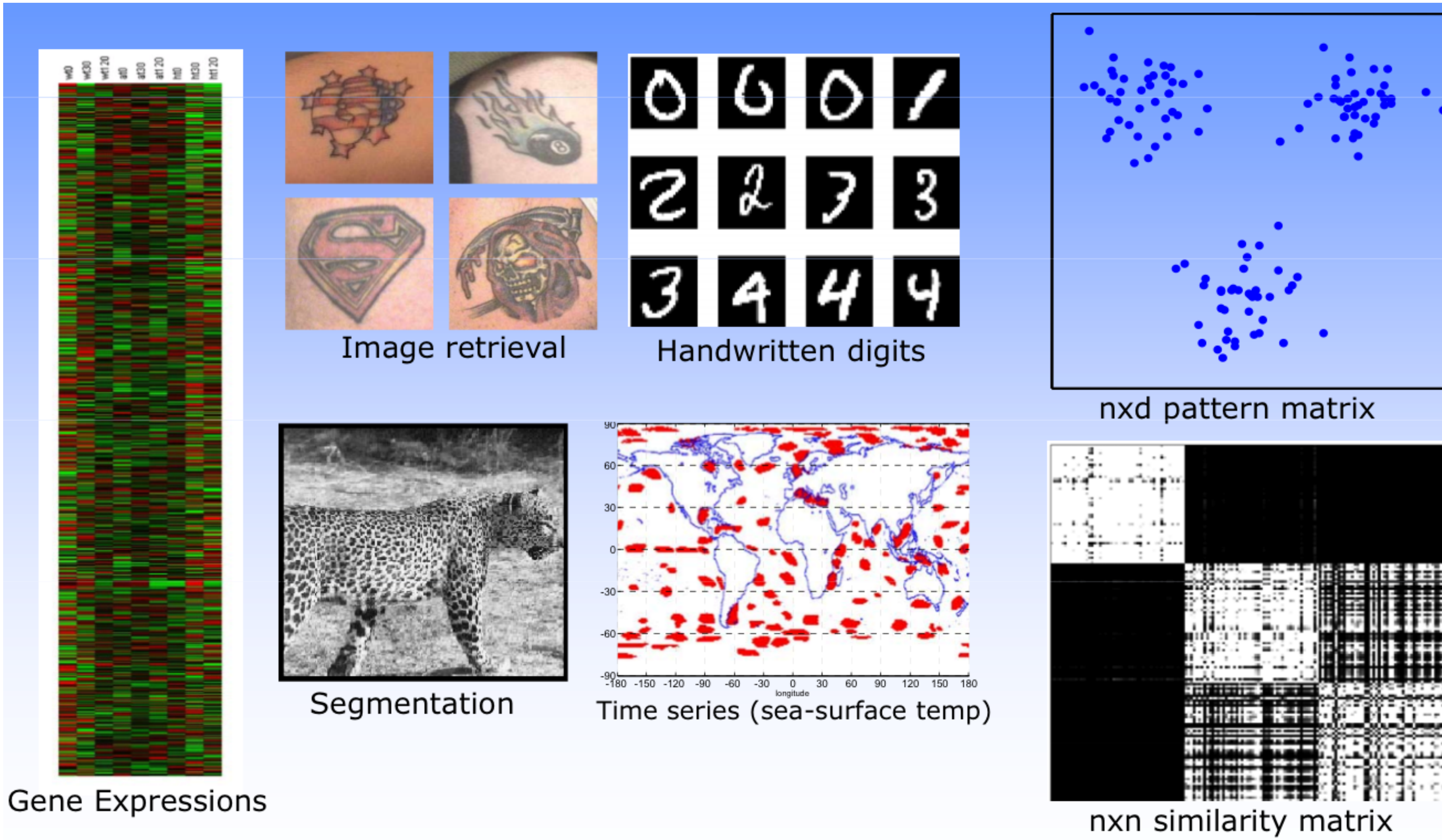
- Compact Clusters
  - Within-cluster distance < between-cluster connectivity
- Connected Clusters
  - Within-cluster connectivity > between-cluster connectivity
- Ideal cluster: **compact** and **isolated**.



Source: R. Dubes and A. K. Jain. “Clustering Techniques: User’s Dilemma” PR 1976.

Representation; what features?

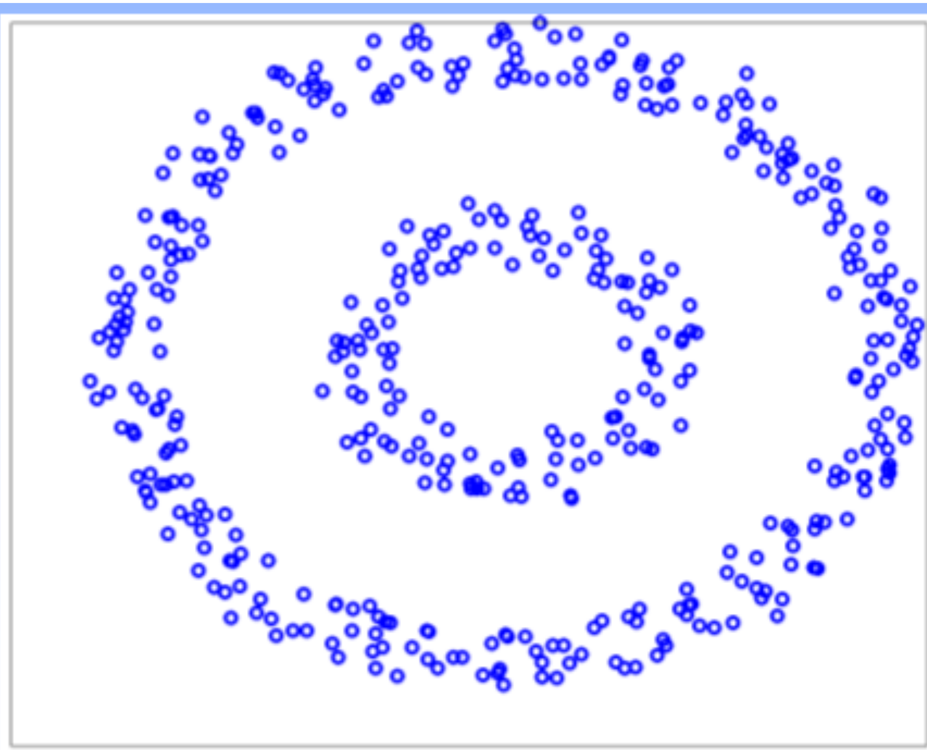
- There is no universal representation.



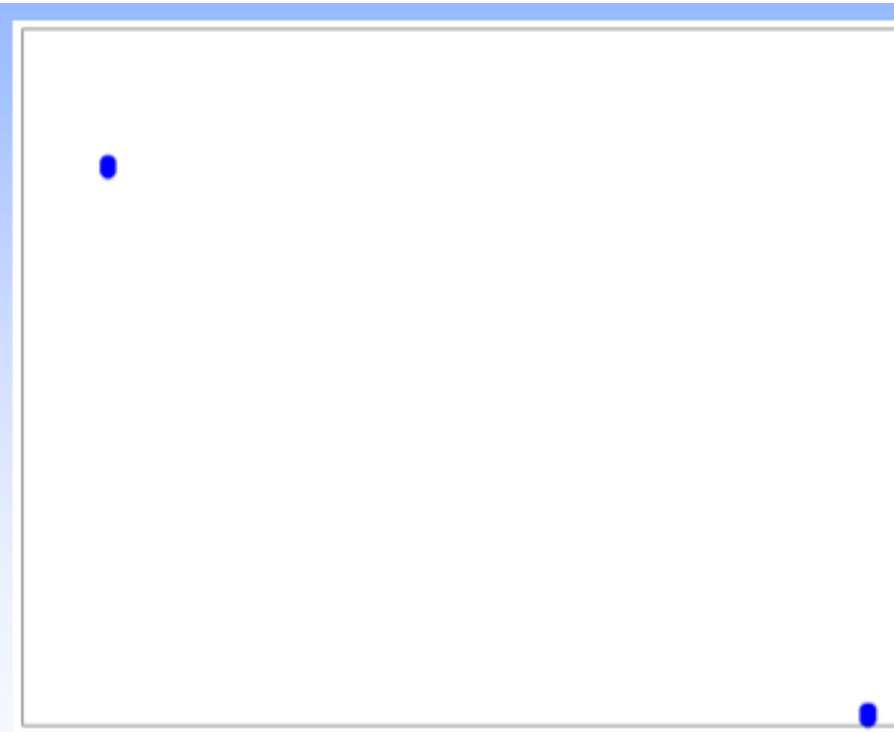
Source: R. Dubes and A. K. Jain. “Clustering Techniques: User’s Dilemma” PR 1976.

Good Representations

- A *good* representation leads to compact and isolated clusters.



Points in given 2D space

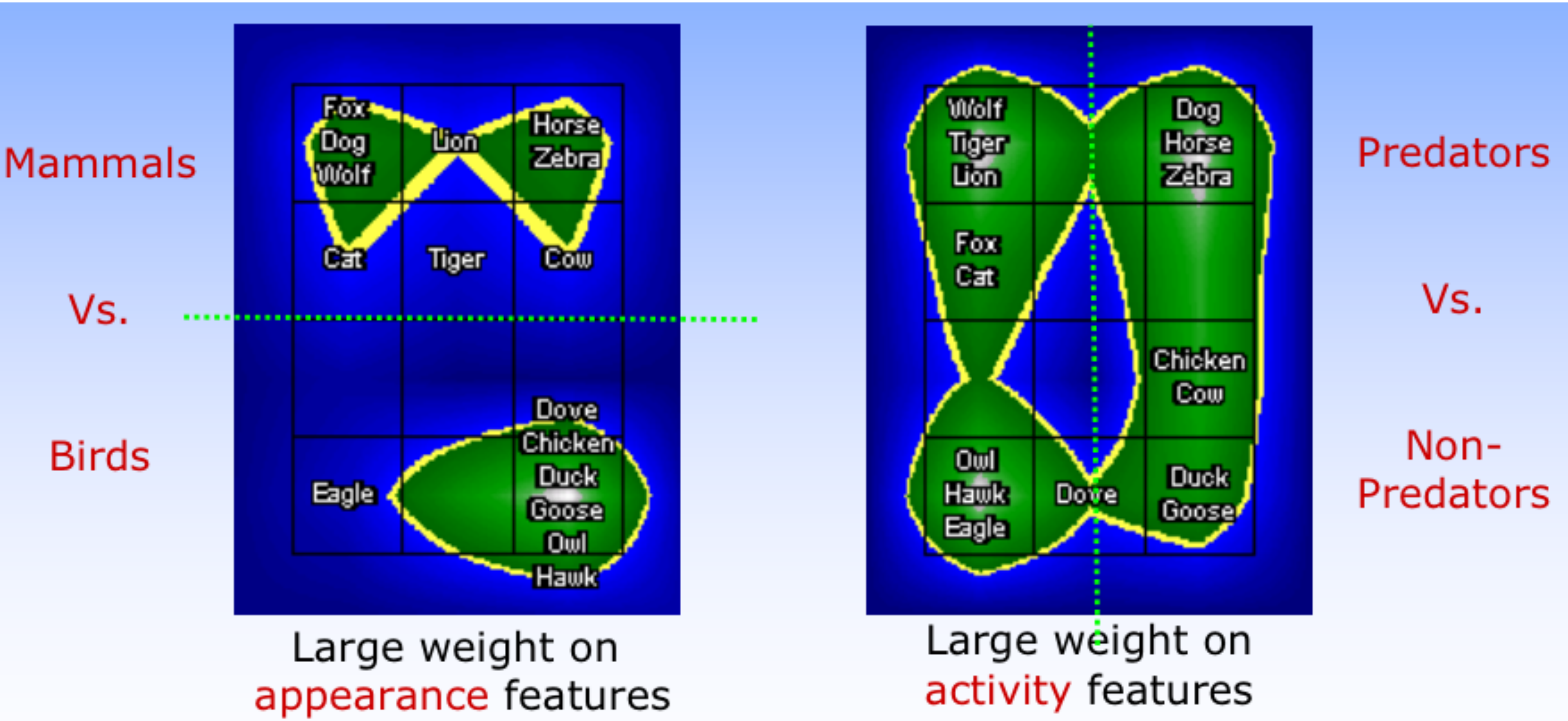


Representation based on eigenvectors of RBF kernel

Source: R. Dubes and A. K. Jain. "Clustering Techniques: User's Dilemma" PR 1976.

How should the features be weighted?

- Two different meaningful groupings produced by different weighting schemes.



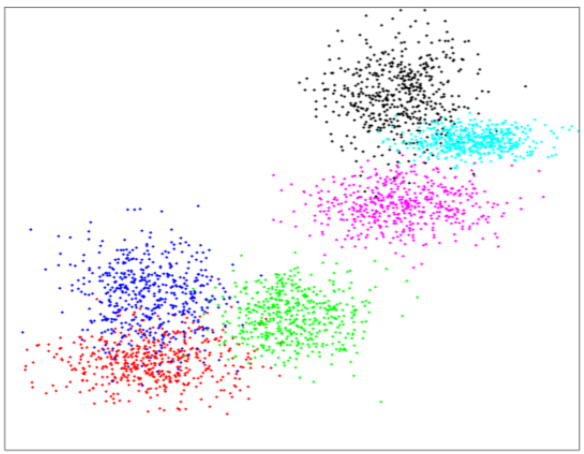
http://www.ofai.at/~elias.pampalk/kdd03/animals/

Source: R. Dubes and A. K. Jain. “Clustering Techniques: User’s Dilemma” PR 1976.

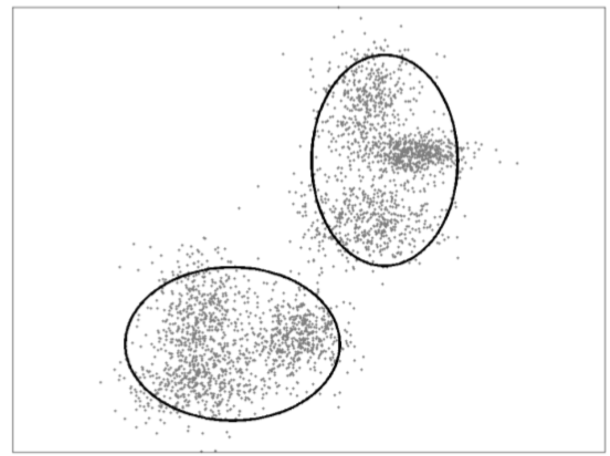
How do we decide on the number of clusters?

- These samples are generated by 6 independent classes.

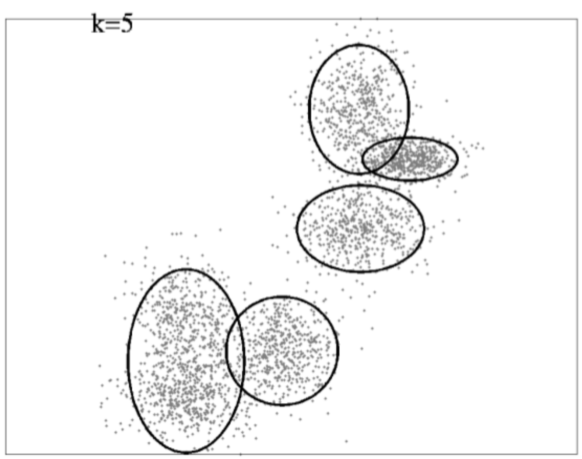
Input



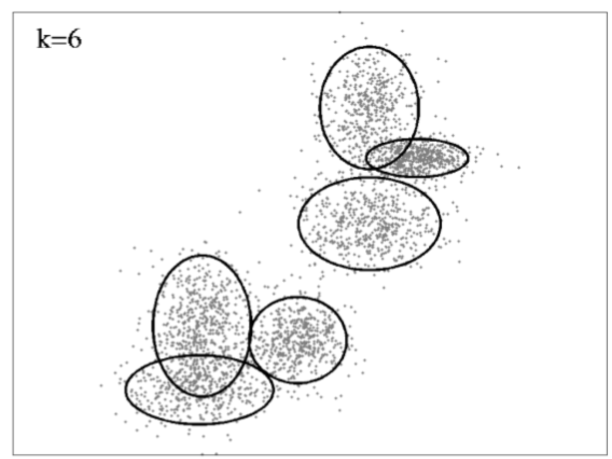
K=2



K=5



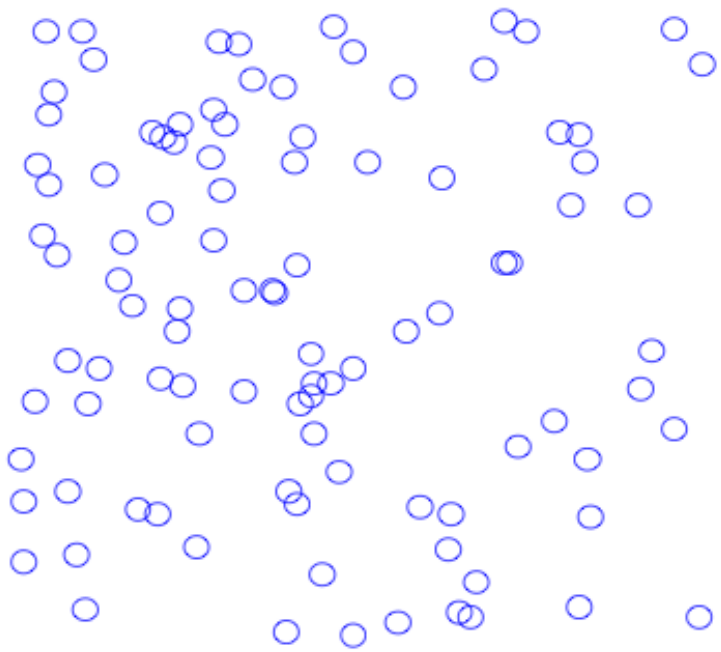
K=6



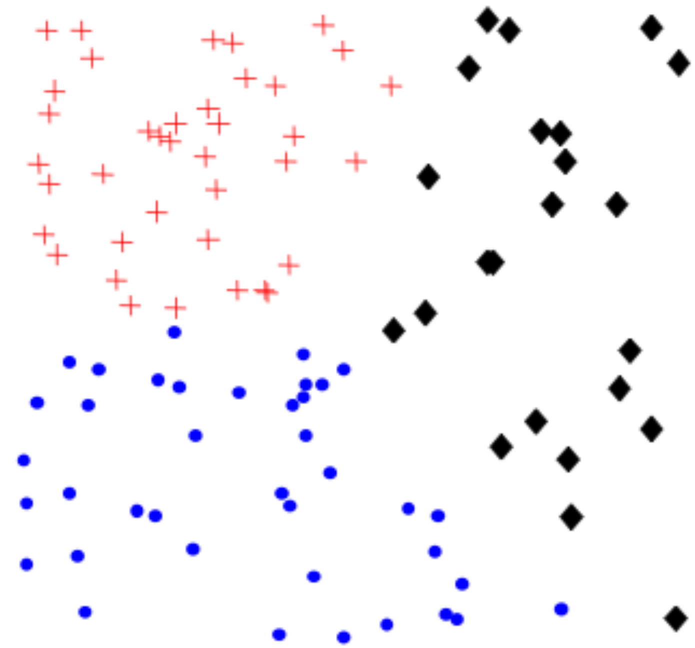
Source: R. Dubes and A. K. Jain. “Clustering Techniques: User’s Dilemma” PR 1976.

Cluster Validity

- Clustering algorithms find clusters, even if there are no natural clusters in the data!



100 2D Uniform Data Points

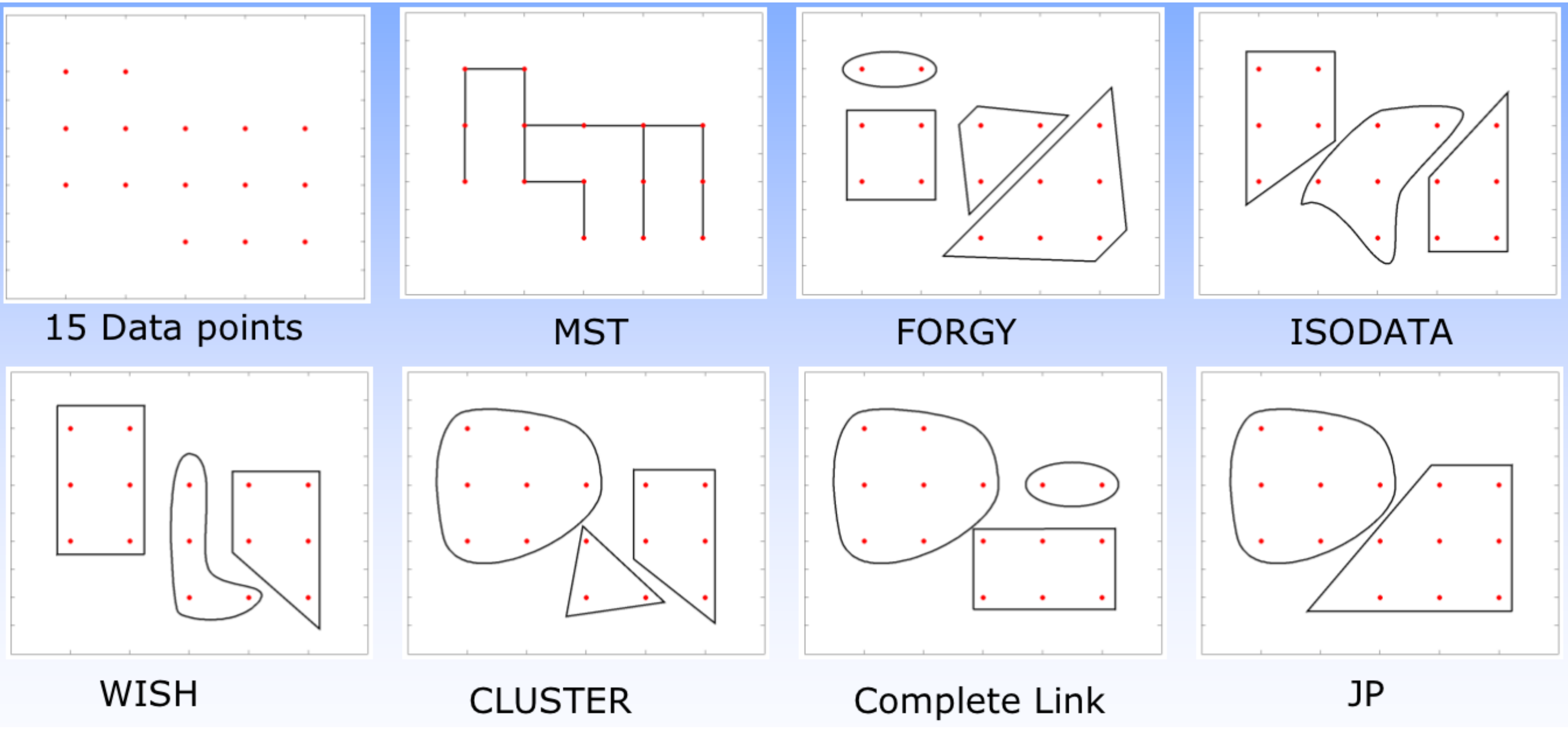


K-means with K=3

Source: R. Dubes and A. K. Jain. “Clustering Techniques: User’s Dilemma” PR 1976.

Choosing a Clustering Method

- Which is best?



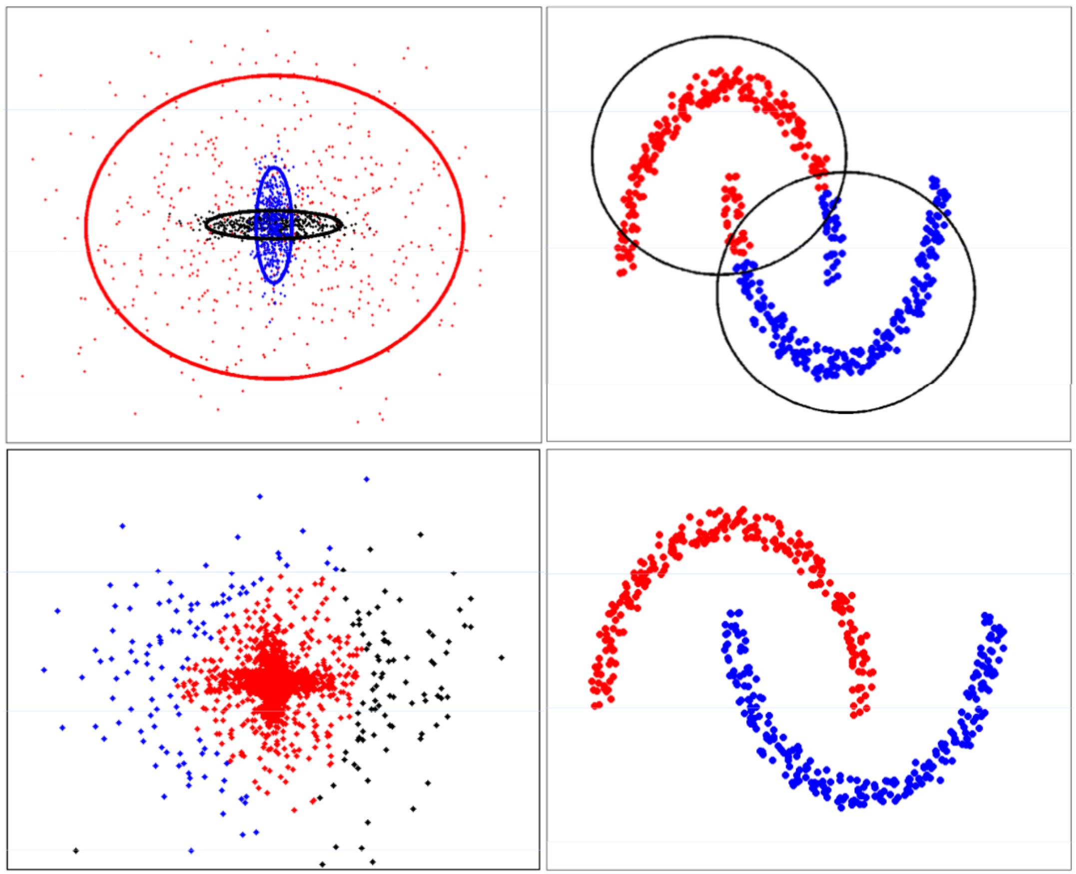
Source: R. Dubes and A. K. Jain. "Clustering Techniques: User's Dilemma" PR 1976.

Choosing a Clustering Method

- Depends on problem/data.
- Each algorithm imposes some structure.

GMM: k=3

GMM: k=2



Spectral: k=3

Spectral: k=2

Source: R. Dubes and A. K. Jain. “Clustering Techniques: User’s Dilemma” PR 1976.

Plan

- What is Clustering? Challenges in Clustering
- Clustering (for Segmentation)
  - K-Means
  - GMMs (and Expectation-Maximization)
  - Mean-Shift
- Other uses of clustering in vision
  - Texture and Textons
  - Quantization
  - Bag of Words

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 $\boldsymbol{\theta}$

Source: Seitz slides.

Gaussian Mixture Models

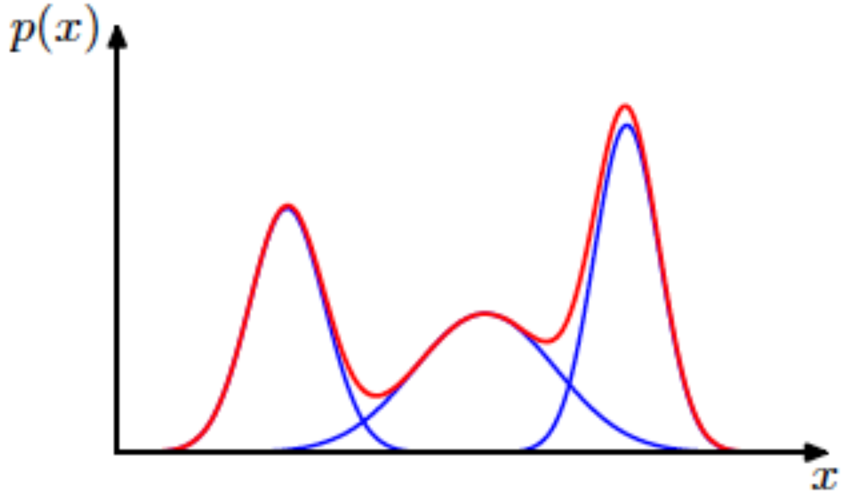
- Recall the Gaussian distribution

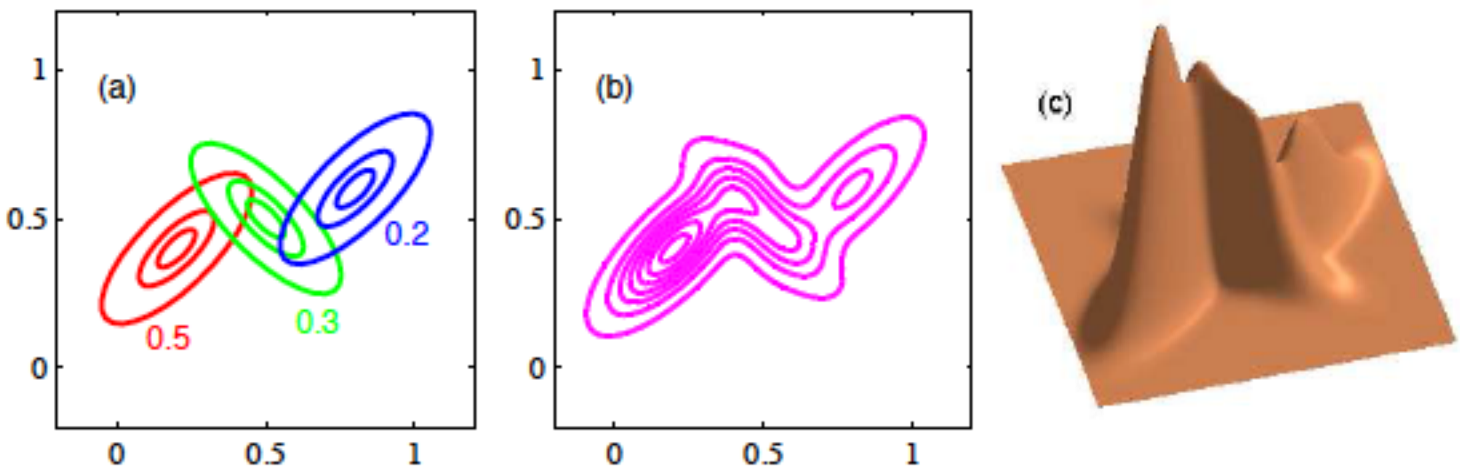
$$\mathcal{N}(x|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$

- It forms the basis for the mixture of Gaussians density
- The Gaussian mixture is linear superposition of Gaussians:

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- The π_k are non-negative scalars called mixing coefficients and they govern the relative importance between the various Gaussians in the mixture density. $\sum_k \pi_k = 1$


p(x)
x


(a)
0.5
0.3
0.2
(b)
(c)
1
0.5
0
0
0.5
1

GMM: Introducing Latent Variables

- Define a K-dimensional binary random variable $\mathbf{z}$
- $\mathbf{z}$ has a 1-of-K representation such that a particular element z_k is 1 and all of the others are zero. Hence:

$$z_k \in \{0, 1\}$$

$$\sum_k z_k = 1$$

- The marginal distribution over $\mathbf{z}$ is specified in terms of the mixing coefficients:

$$p(z_k = 1) = \pi_k$$

And recall that $0 \leq \pi_k \leq 1$ and $\sum_k \pi_k = 1$

GMM: Introducing Latent Variables

- Since $\mathbf{z}$ has a 1-of-K representation, we can also write the distribution as

$$p(\mathbf{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$$

- The conditional distribution of $\mathbf{x}$ given $\mathbf{z}$ is a Gaussian:

$$p(\mathbf{x}|z_k = 1) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

$$p(\mathbf{x}|\mathbf{z}) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma})^{z_k}$$

GMM: Introducing Latent Variables

- We are interested in the marginal distribution of $\mathbf{x}$

$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{x}, \mathbf{z})$$
$$= \sum_{\mathbf{z}} p(\mathbf{z}) p(\mathbf{x}|\mathbf{z})$$
$$= \sum_{\mathbf{z}} \prod_{k=1}^{K} \pi_k^{z_k} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_k}$$
$$= \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- So, given our latent variable $\mathbf{z}$, the marginal distribution of $\mathbf{x}$ is a Gaussian mixture.
- If we have N observations, $\mathbf{x}_1, \ldots, \mathbf{x}_N$, then because of our chosen representation, if follows that we have a latent variable $\mathbf{z}_n$ for each observed data point $\mathbf{x}_n$.

Component Responsibility Term

- We need to also express the conditional probability of $\mathbf{z}$ given $\mathbf{x}$.
- Denote this conditional $p(z_k = 1|\mathbf{x})$ as $\gamma(z_k)$
- Via Bayes' theorem:

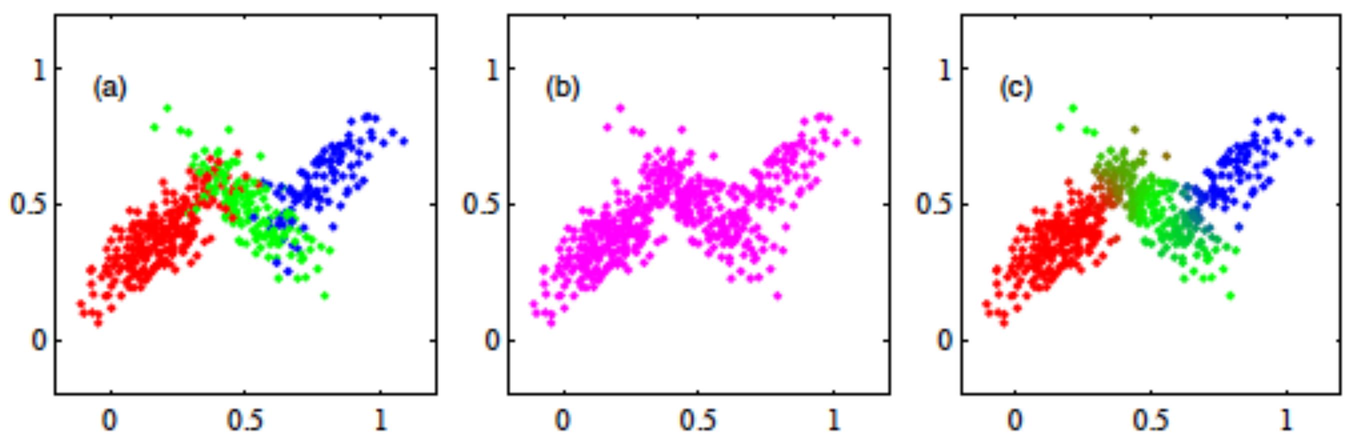
$$\gamma(z_k) = \frac{p(z_k = 1)p(\mathbf{x}|z_k = 1)}{\sum_{j=1}^{K} p(z_j = 1)p(\mathbf{x}|z_j = 1)}$$

$$= \frac{\pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

- View π_k as the prior probability of $z_k = 1$ and the quantity $\gamma(z_k)$ as the corresponding posterior probability after observing $\mathbf{x}$.
- $\gamma(z_k)$ is also the *responsibility* that component k takes for $\mathbf{x}$

Sampling from the GMM

- To sample from the GMM, we can first generate a value for $\mathbf{z}$ from the marginal distribution $p(\mathbf{z})$. Denote this sample $\hat{\mathbf{z}}$.
- Then, sample from the conditional distribution $p(\mathbf{x}|\hat{\mathbf{z}})$.
- The figure below-left shows samples from a three-mixture and colors the samples based on the component ($\mathbf{z}$). The figure below-middle shows samples from the marginal $p(\mathbf{x})$ and ignores $\mathbf{z}$. On the right, we show the $\gamma(z_k)$ for each sampled point, colored accordingly.



Maximum-Likelihood Fitting

Suppose we have a set of N observations $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ that we wish to model with a GMM.

Consider this data set as an $N \times d$ matrix $\mathbf{X}$ in which the n^{th} row is given by $\mathbf{x}_n^\mathsf{T}$.

Similarly, the corresponding latent variables define an $N \times K$ matrix $\mathbf{Z}$ with rows $\mathbf{z}_n^\mathsf{T}$.

The log-likelihood of the corresponding GMM is given by

$$\ln p(\mathbf{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \ln \left[\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right] \quad .$$

Ultimately, we want to find the values of the parameters $\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$ that maximize this function.

However, maximizing the log-likelihood terms for GMMs is much more complicated than for the case of a single Gaussian. Why?

The difficulty arises from the sum over k inside of the log-term. The log function no longer acts directly on the Gaussian, and no closed-form solution is available.

Singularities with GMM Fitting

There is a significant problem when we apply MLE to estimate GMM parameters.

Consider simply covariances defined by $\boldsymbol{\Sigma}_k = \sigma_k^2 \mathbf{I}$.

Suppose that one of the components of the mixture model, j, has its mean $\boldsymbol{\mu}_j$ exactly equal to one of the data points so that $\boldsymbol{\mu}_j = \mathbf{x}_n$ for some n.

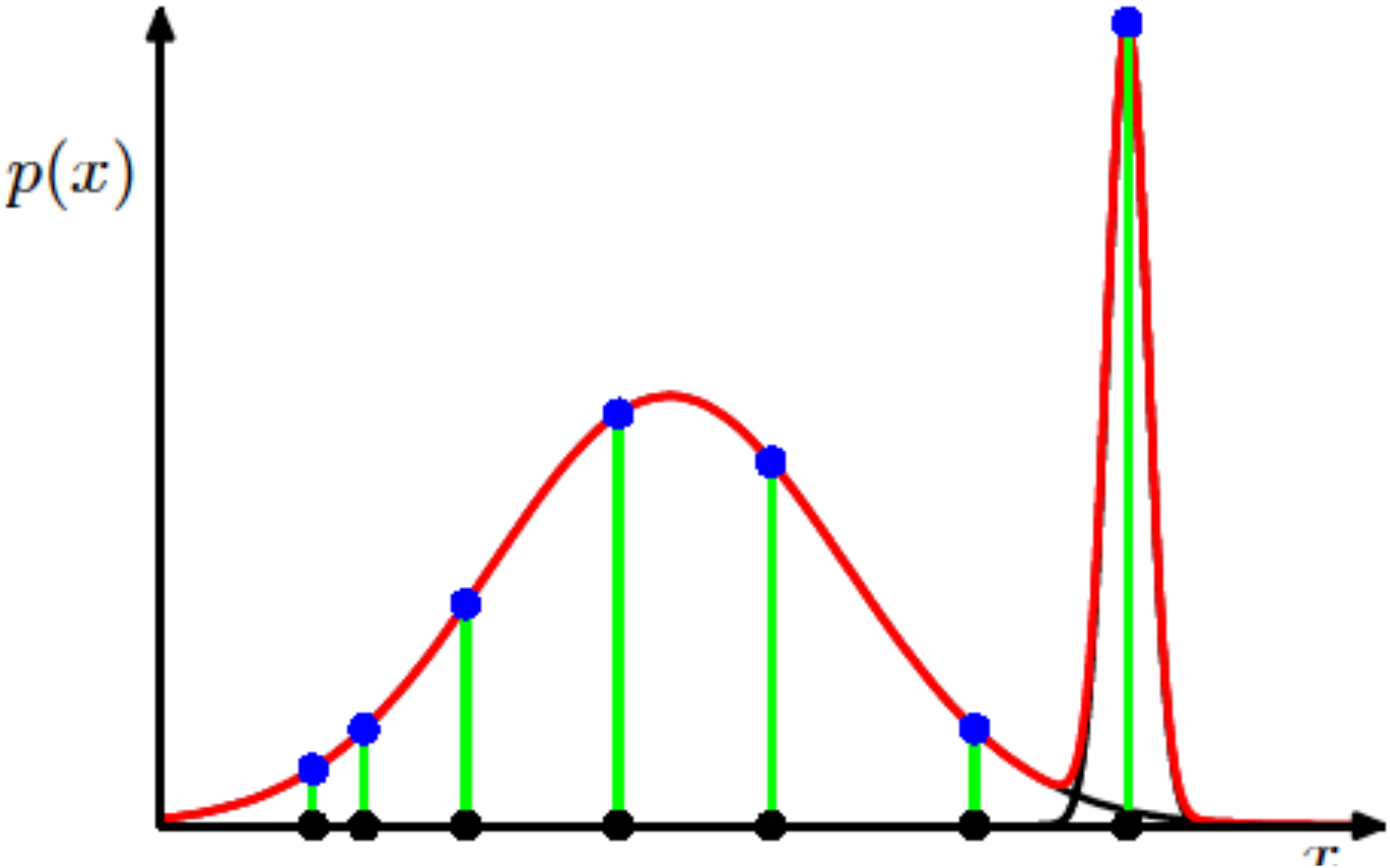
This term contributes

$$\mathcal{N}(\mathbf{x}_n | \mathbf{x}_n, \sigma_j^2 \mathbf{I}) = \frac{1}{(2\pi)^{(1/2)} \sigma_j}$$

Consider the limit $\sigma_j \to 0$ to see that this term goes to infinity and hence the log-likelihood will also go to infinity.

Thus, the maximization of the log-likelihood function is not a well posed problem because such a singularity will occur whenever one of the components collapses to a single, specific data point.

Singularities with GMM Fitting



Expectation-Maximization

- External PDF slides
- em_fitting_gmm.pdf

Problems with EM

1. Local minima

   k-means is NP-hard even with k=2

2. Need to know number of segments

   solutions: AIC, BIC, Dirichlet process mixture

3. Need to choose generative model

Source: Seitz slides.

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
  - ...

Source: Seitz slides.

EM demo

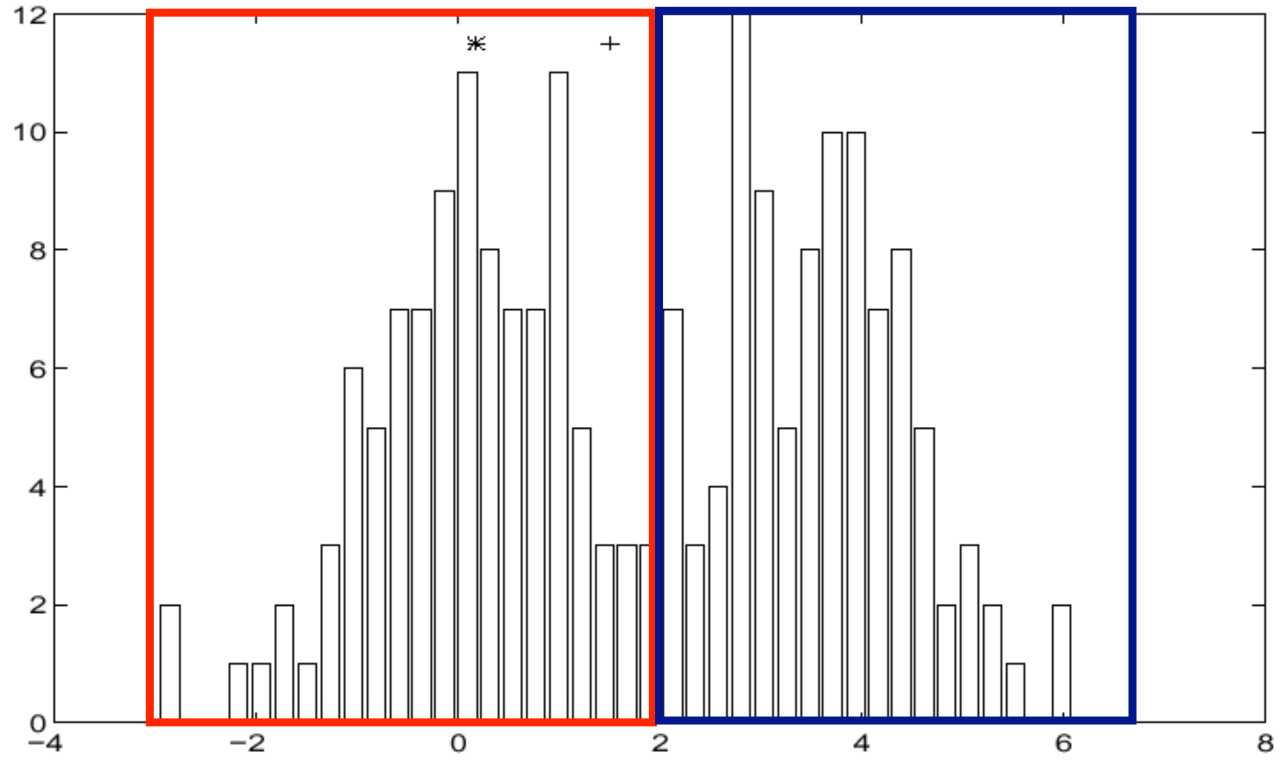
- http://lcn.epfl.ch/tutorial/english/gaussian/html/index.html

Source: Seitz slides.

Plan

- What is Clustering? Challenges in Clustering
- Clustering (for Segmentation)
  - K-Means
  - GMMs (and Expectation-Maximization)
  - Mean-Shift
- Other uses of clustering in vision
  - Texture and Textons
  - Quantization
  - Bag of Words

Finding Modes in a Histogram

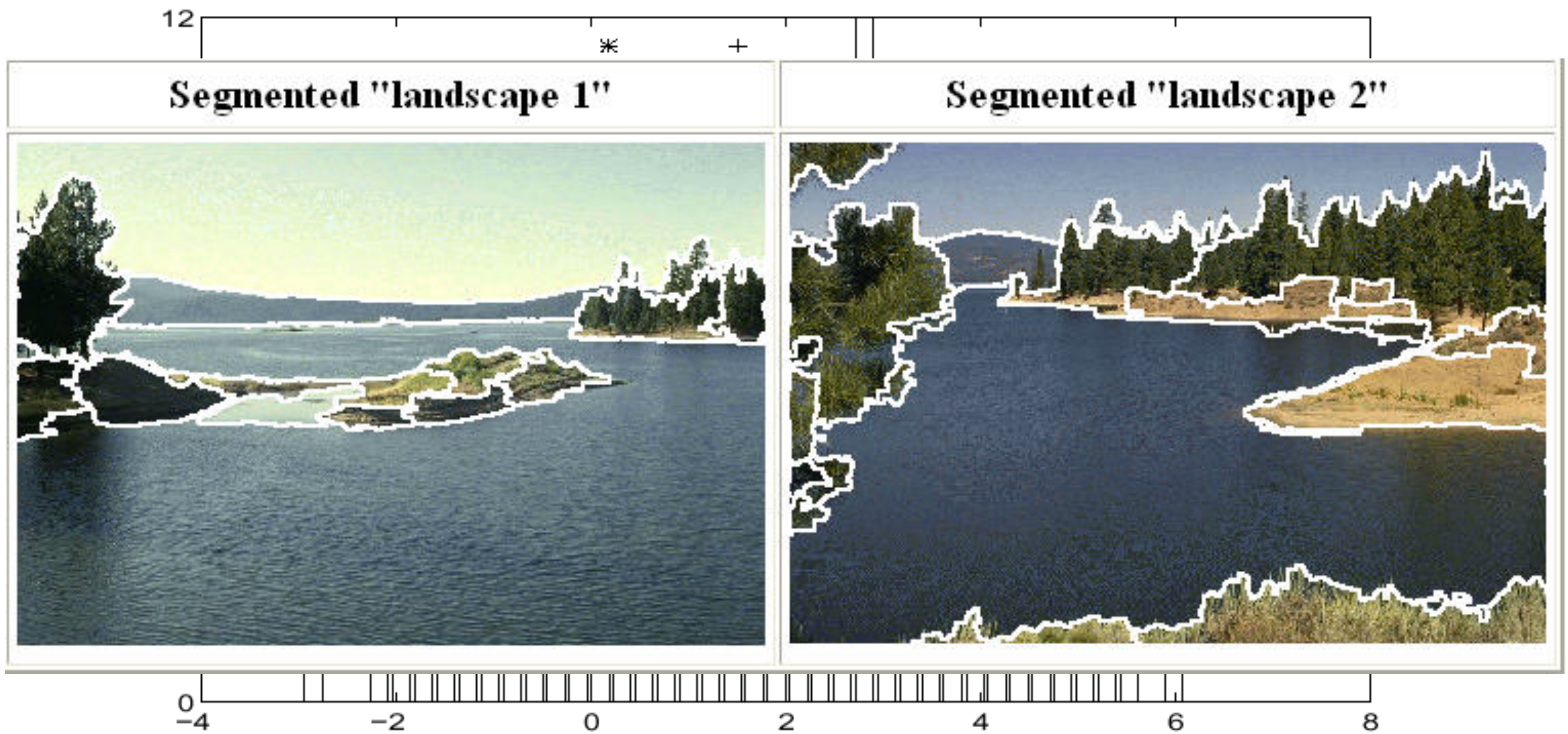


- How Many Modes Are There?
  - Easy to see, hard to compute

Source: Seitz slides.

Mean Shift [Comaniciu & Meer]

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

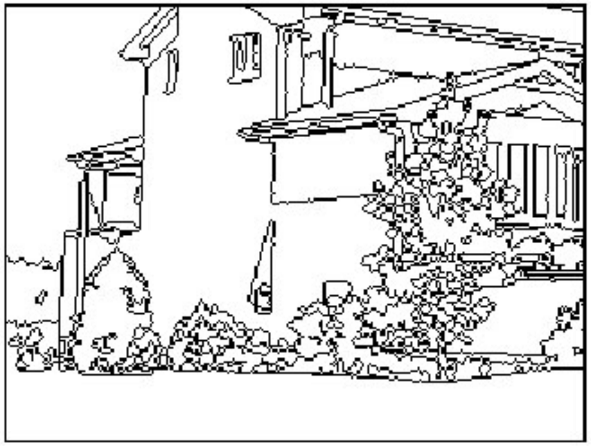


- Iterative Mode Search
  1. Initialize random seed, and window W
  2. Calculate center of gravity (the "mean") of W: $\sum_{x \in W} x H(x)$
  3. Translate the search window to the mean
  4. Repeat Step 2 until convergence

Source: Seitz slides.

Mean-shift for image segmentation

- Useful to take into account spatial information
  - instead of (R, G, B), run in (R, G, B, x, y) space
  - D. Comaniciu, P. Meer, Mean shift analysis and applications, *7th International Conference on Computer Vision*, Kerkyra, Greece, September 1999, 1197-1203.
    - http://www.caip.rutgers.edu/riul/research/papers/pdf/spatmsft.pdf



More Examples: http://www.caip.rutgers.edu/~comanici/segm_images.html

Source: Seitz slides.

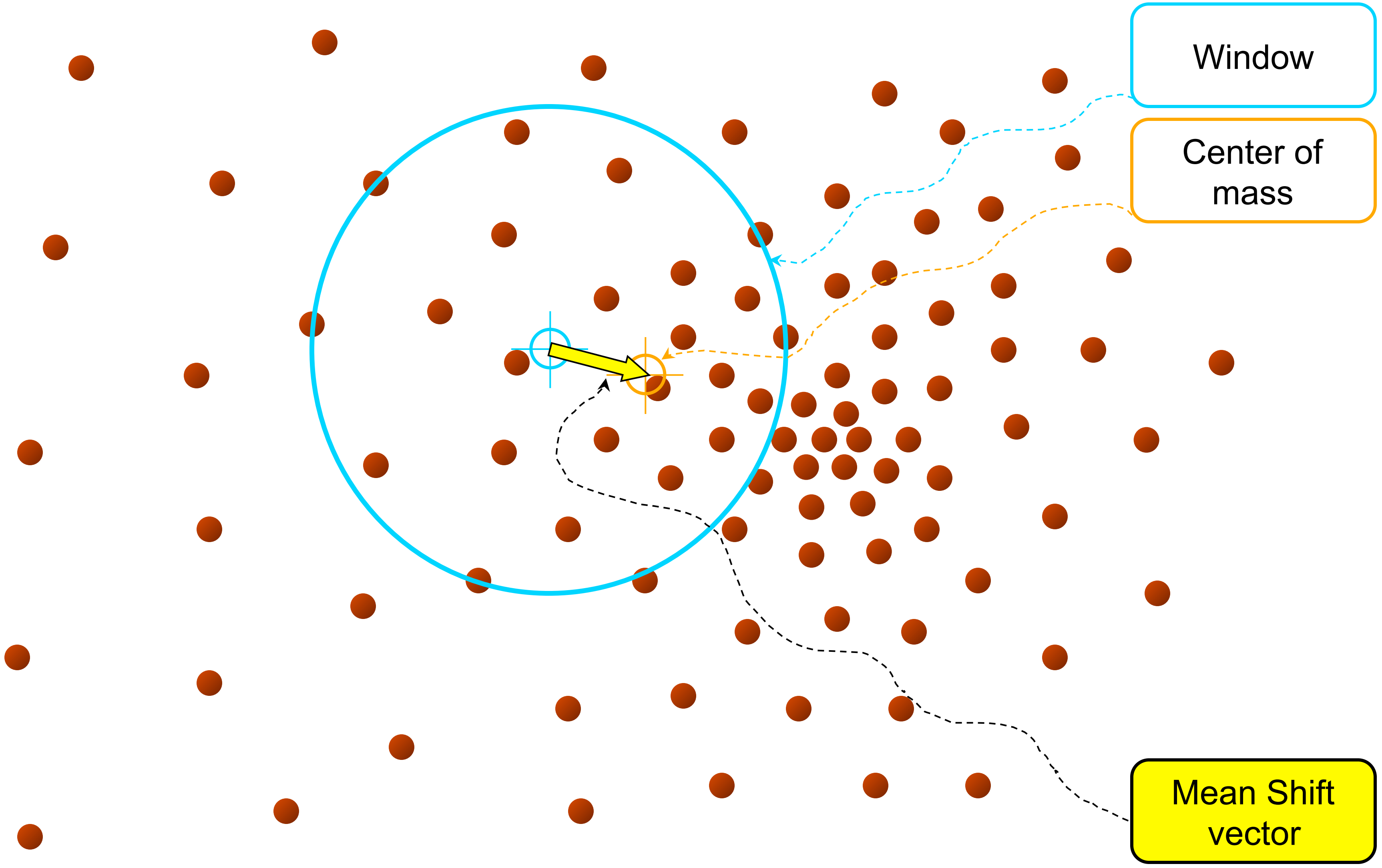
Mean shift algorithm

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 a *mode* or local maximum of density of a given distribution
  - Choose a search window (width and location)
  - Compute the mean of the data in the search window
  - Center the search window at the new mean location
  - Repeat until convergence

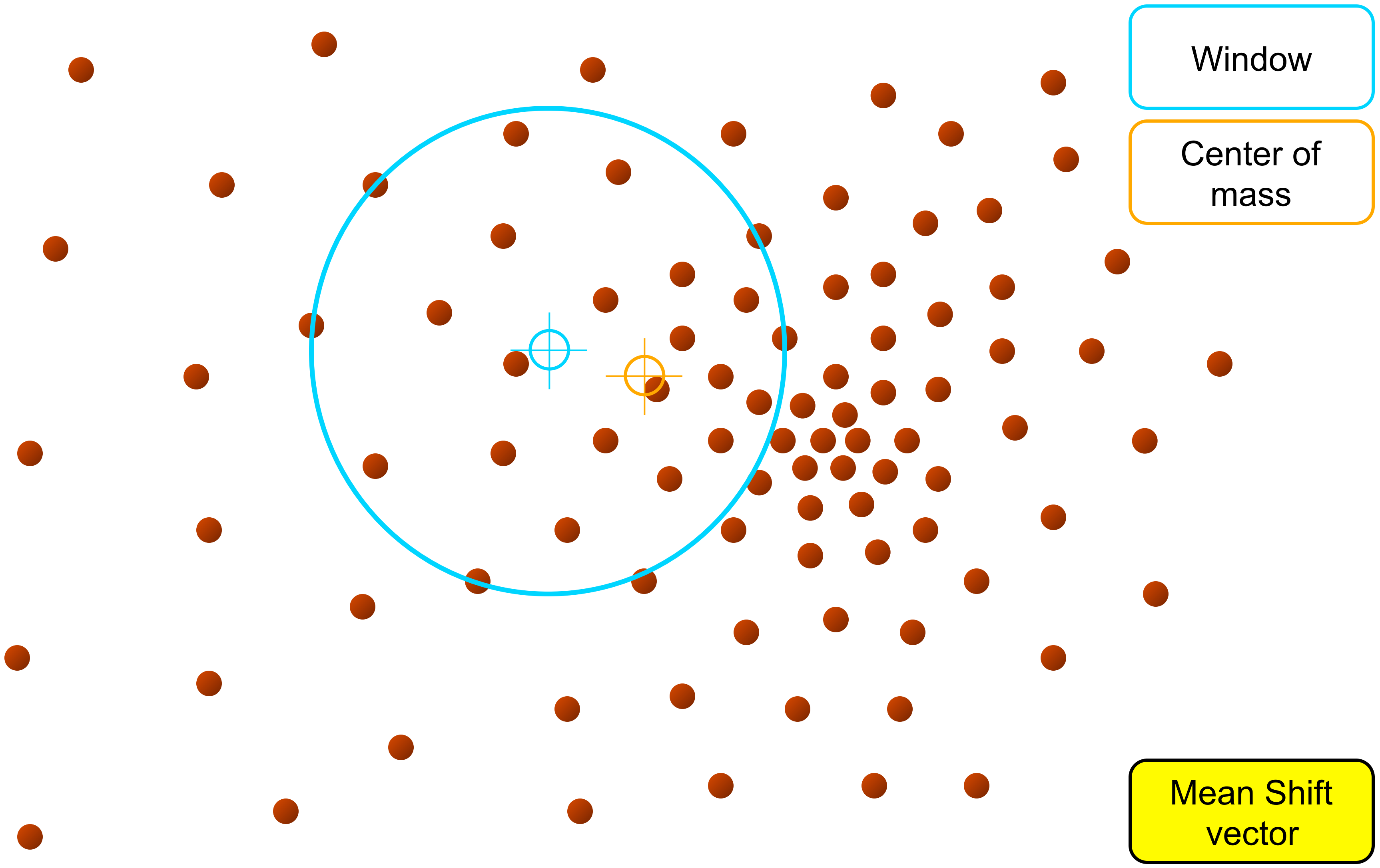
Source: Savarese slides.

Mean Shift



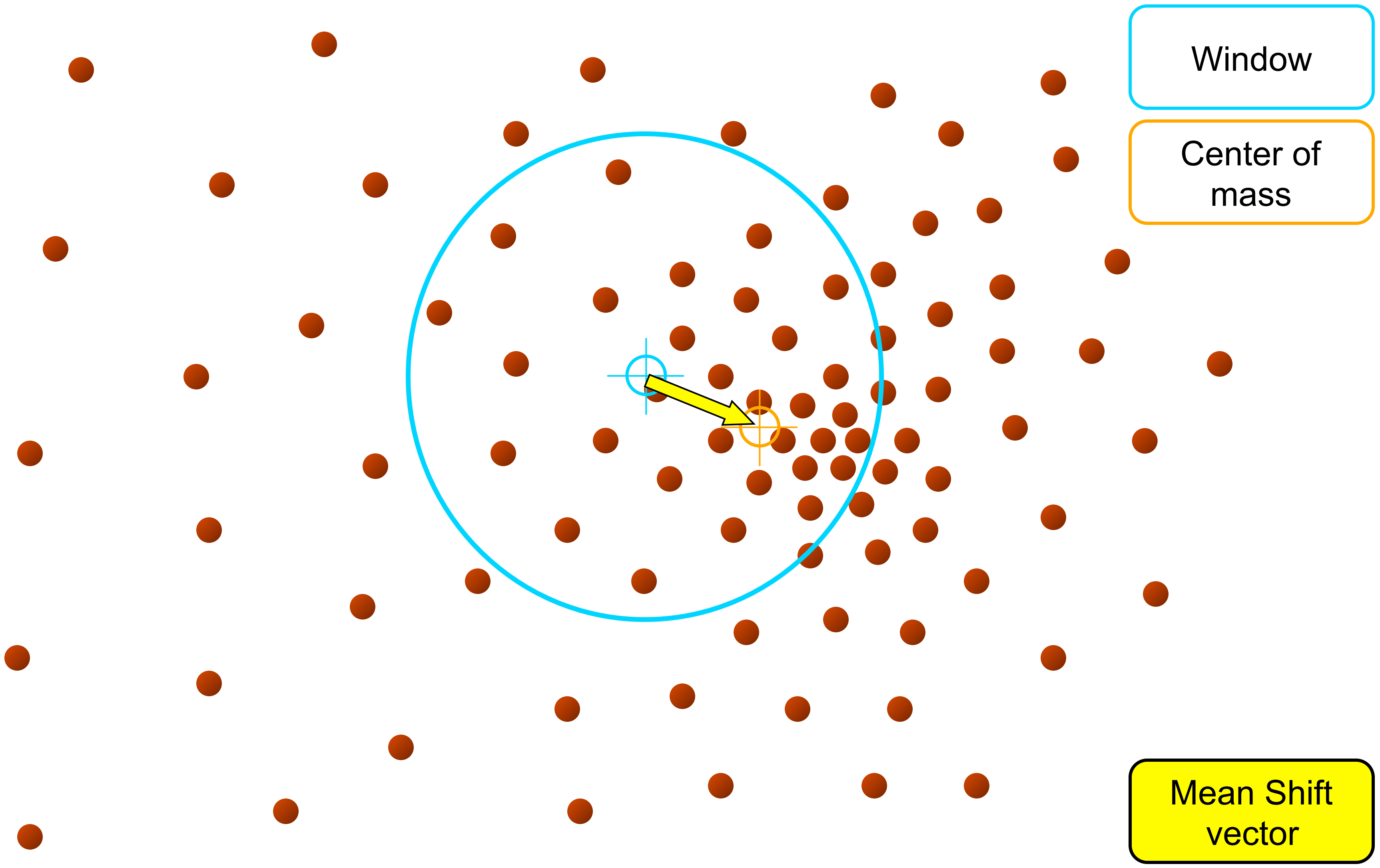
Slide by Y. Ukrainitz & B. Sarel

Mean Shift



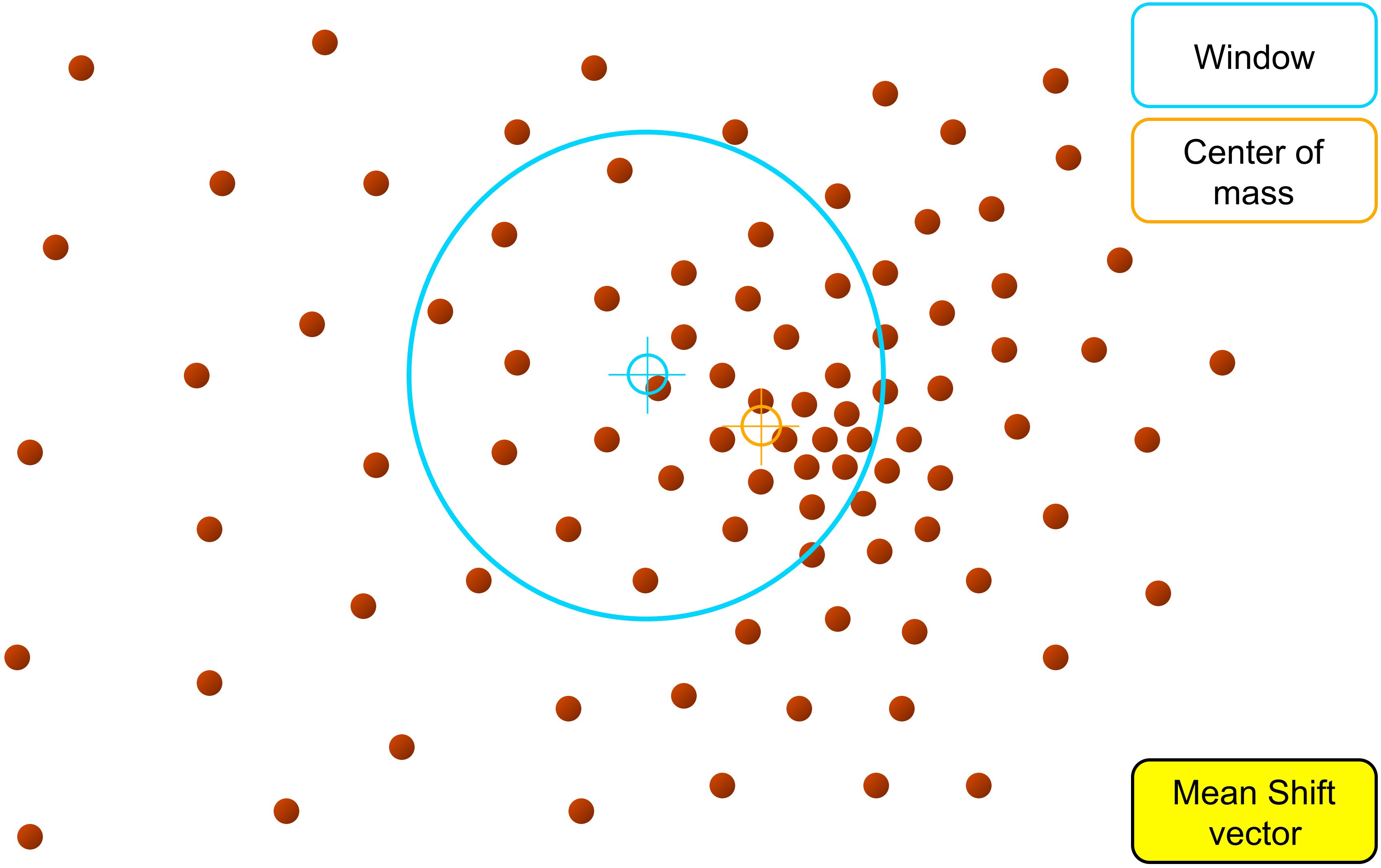
Slide by Y. Ukrainitz & B. Sarel

Mean Shift



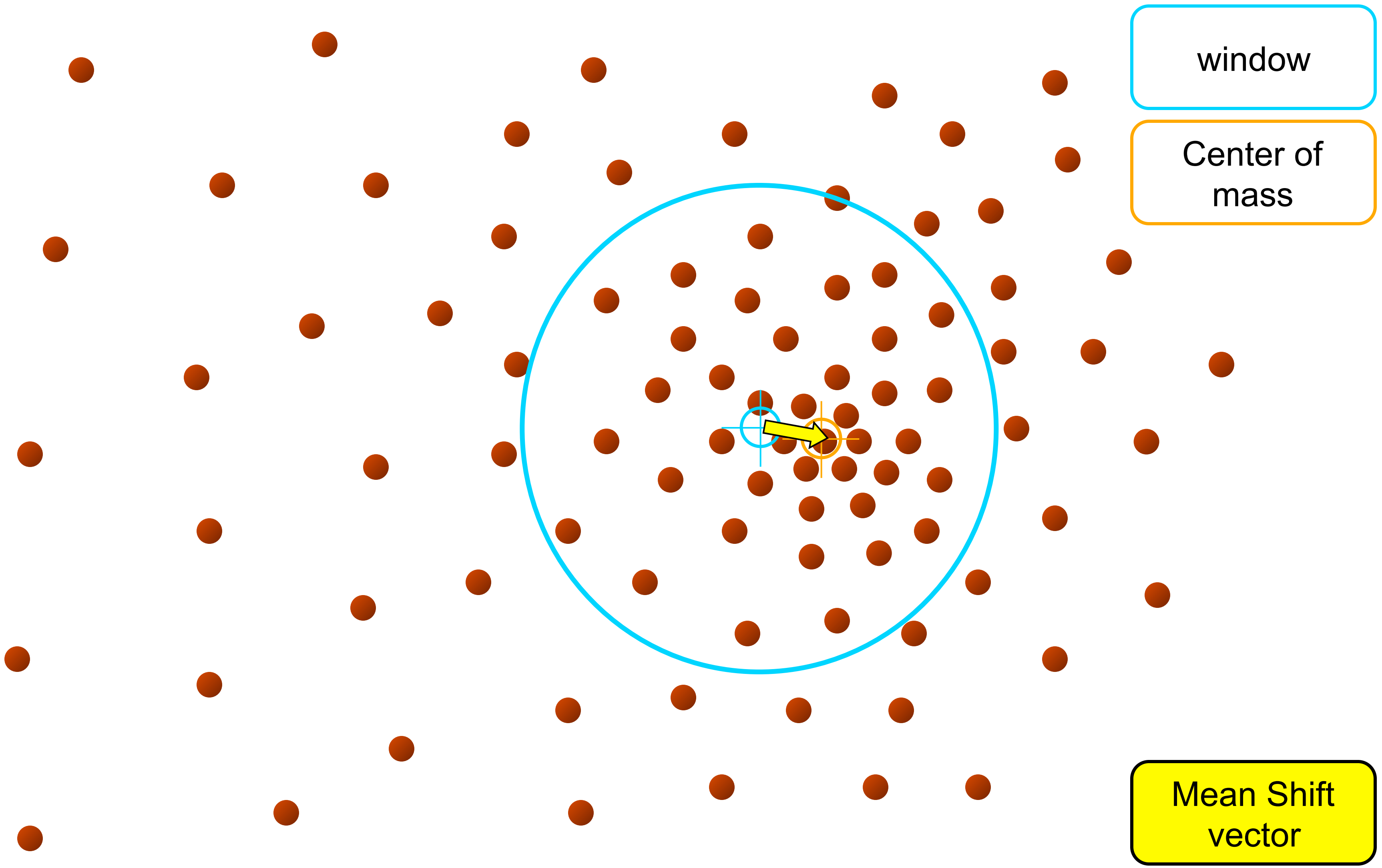
Slide by Y. Ukrainitz & B. Sarel

Mean Shift



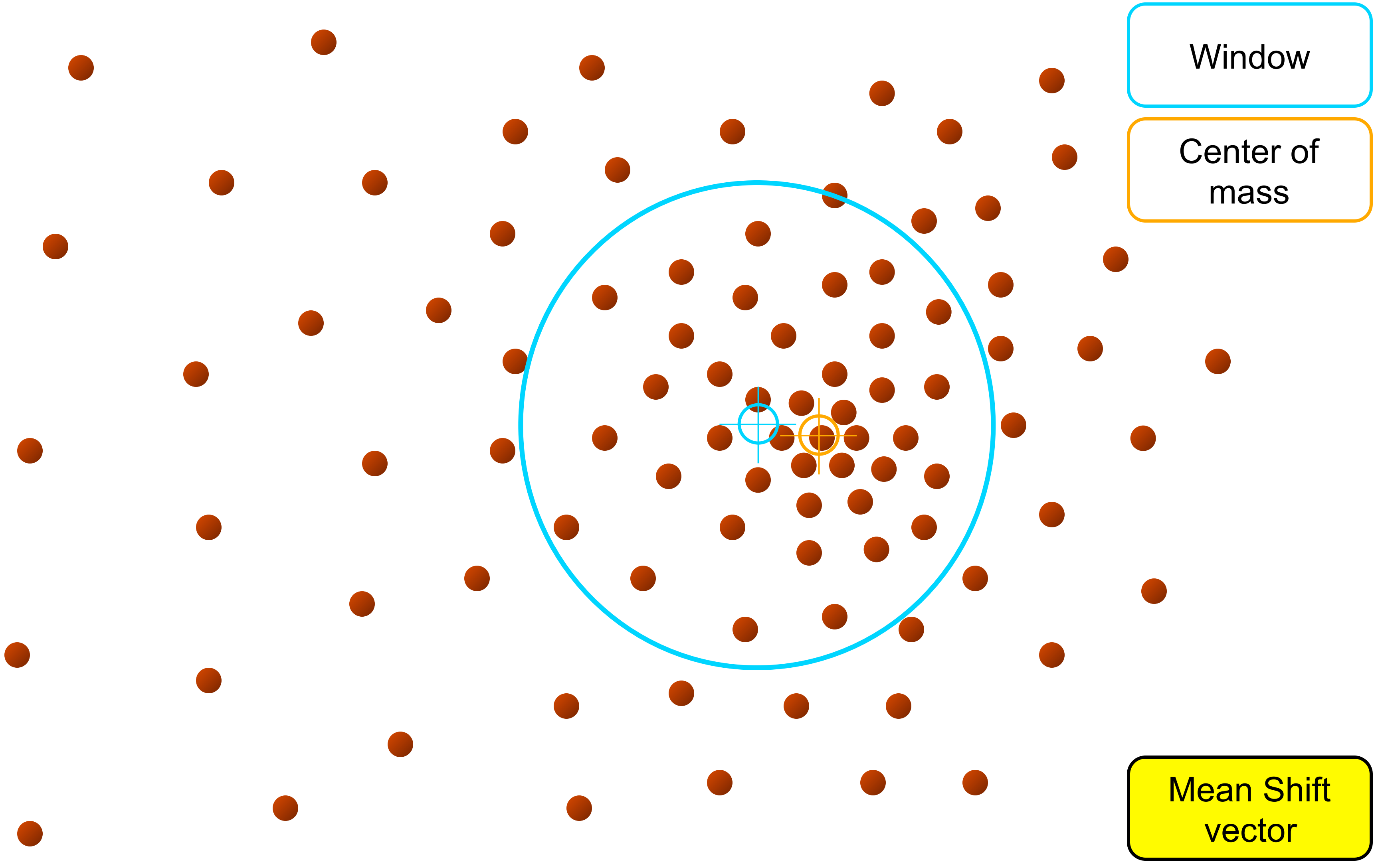
Slide by Y. Ukrainitz & B. Sarel

Mean Shift



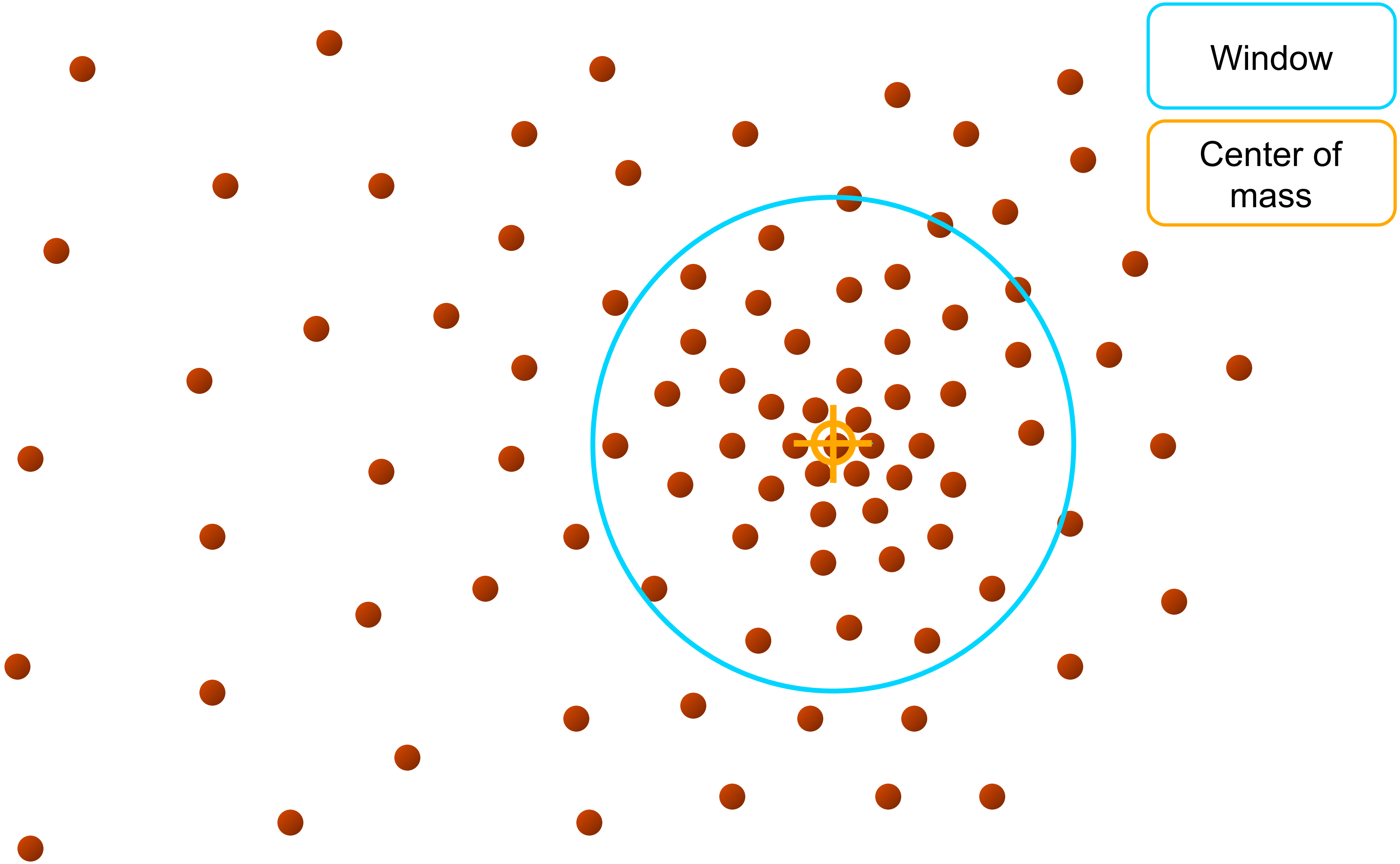
Slide by Y. Ukrainitz & B. Sarel

Mean Shift



Slide by Y. Ukrainitz & B. Sarel

Mean Shift



Slide by Y. Ukrainitz & B. Sarel

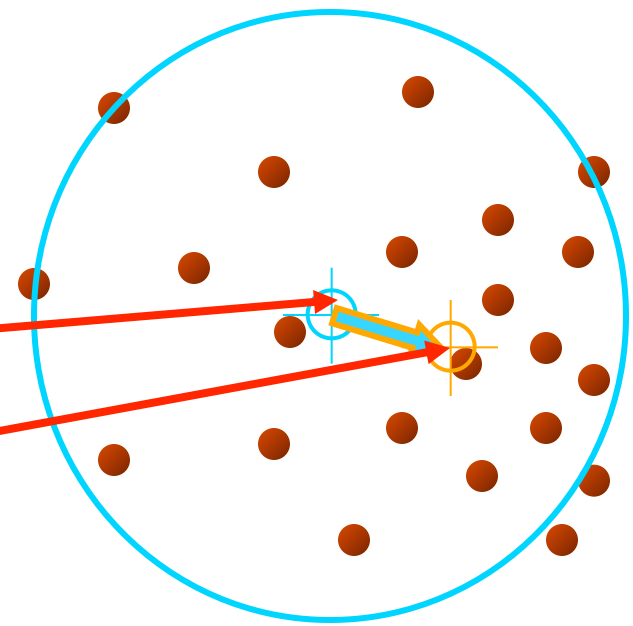
Computing The Mean Shift

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

- Compute mean shift vector

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

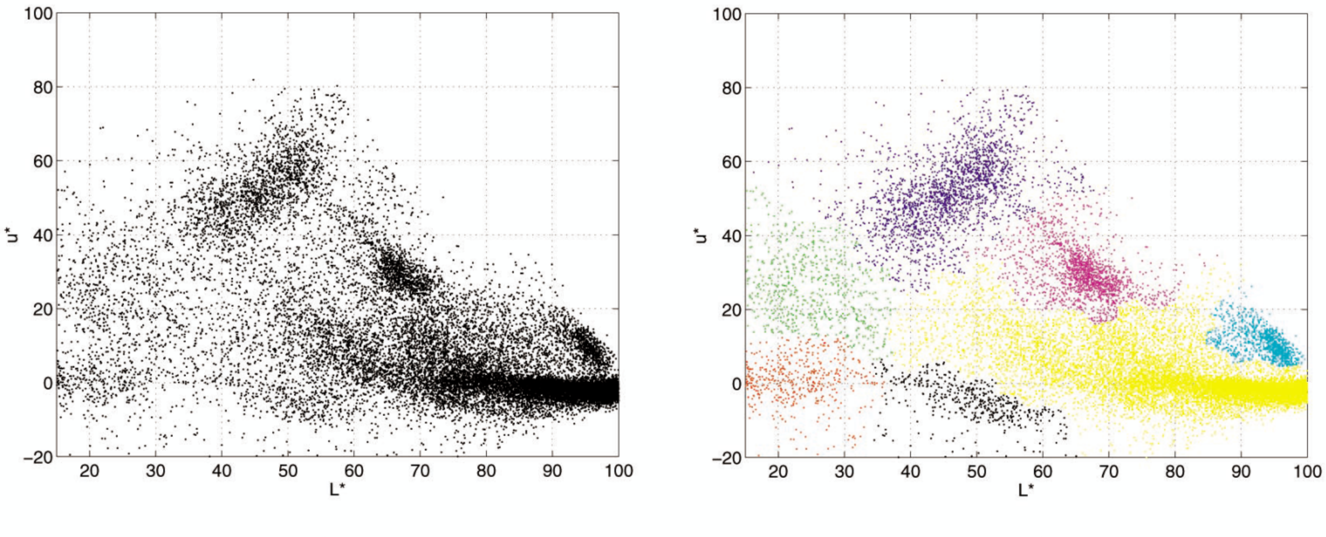
$$\mathbf{m}(\mathbf{x}) = \left[\frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\frac{\left\| \mathbf{x} - \mathbf{x}_i \right\|^2}{h} \right)}{\sum_{i=1}^{n} g\left(\frac{\left\| \mathbf{x} - \mathbf{x}_i \right\|^2}{h} \right)} - \mathbf{x} \right]$$



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

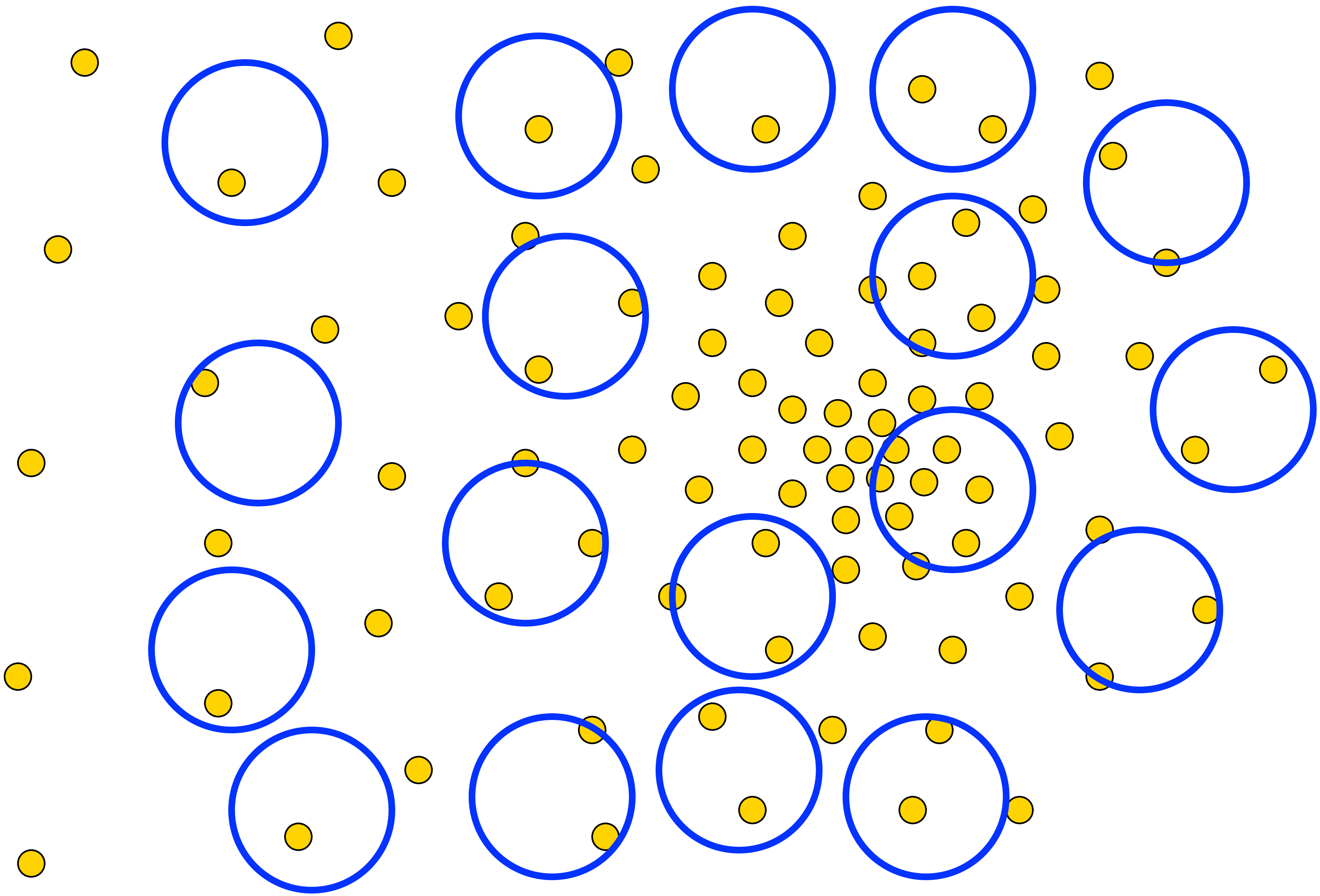
Source: Savarese slides.

Multimodal distributions



Slide by Y. Ukrainitz & B. Sarel

Real Modality Analysis



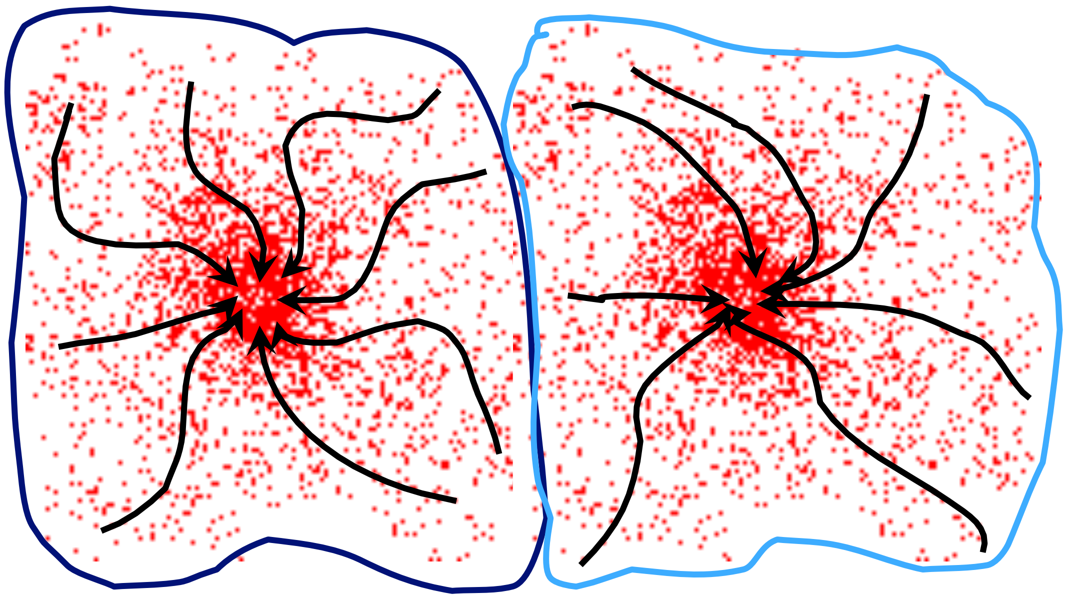
- **Tessellate the space with windows**

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

Source: Savarese slides.

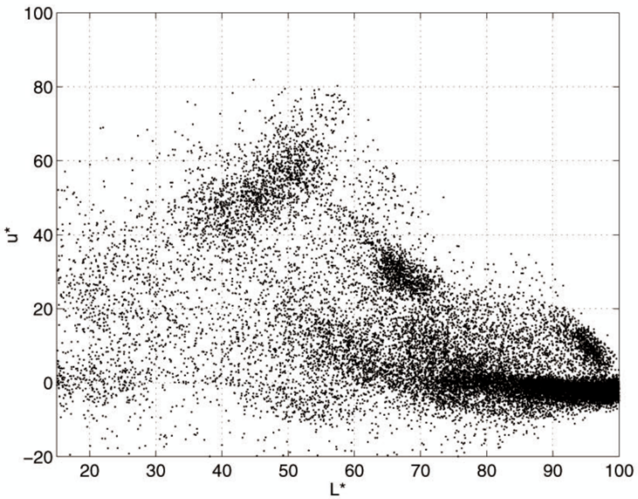
Attraction basin

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

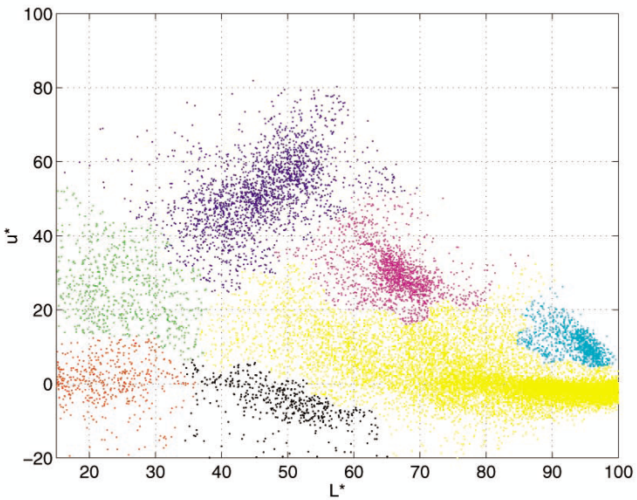


Slide by Y. Ukrainitz & B. Sarel

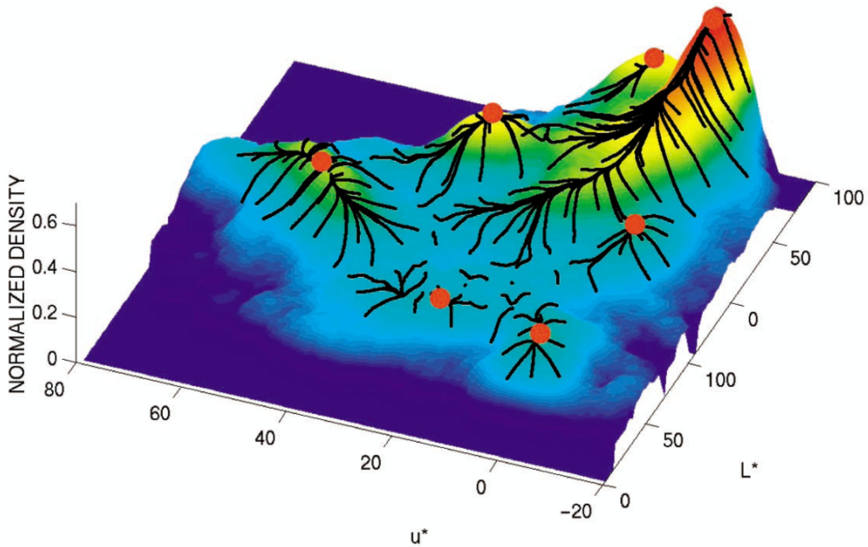
Attraction basin



(a)



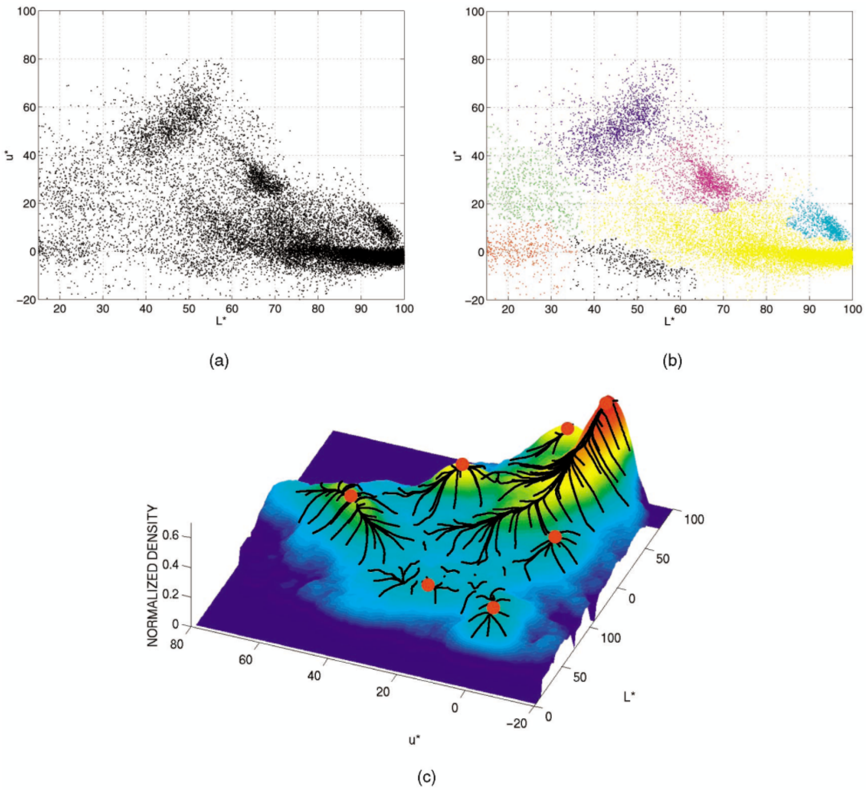
(b)



Slide by Y. Ukrainitz & B. Sarel

Segmentation by Mean Shift

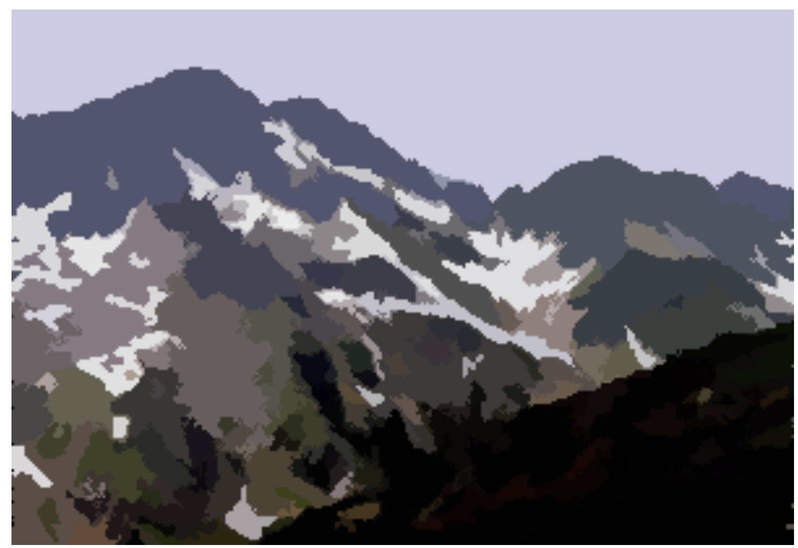
- Find features (color, gradients, texture, etc)
- Initialize windows at individual pixel locations
- Perform mean shift for each window until convergence
- Merge windows that end up near the same “peak” or mode



Source: Savarese slides.

Mean shift segmentation results



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

Source: Savarese slides.

Source: Savarese slides.

Mean shift pros and cons

- Pros
  - Does not assume spherical clusters
  - Just a single parameter (window size)
  - 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

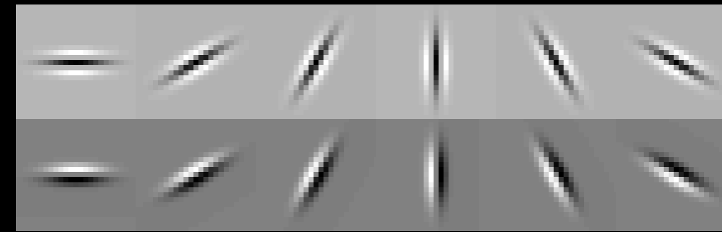
Source: Savarese slides.

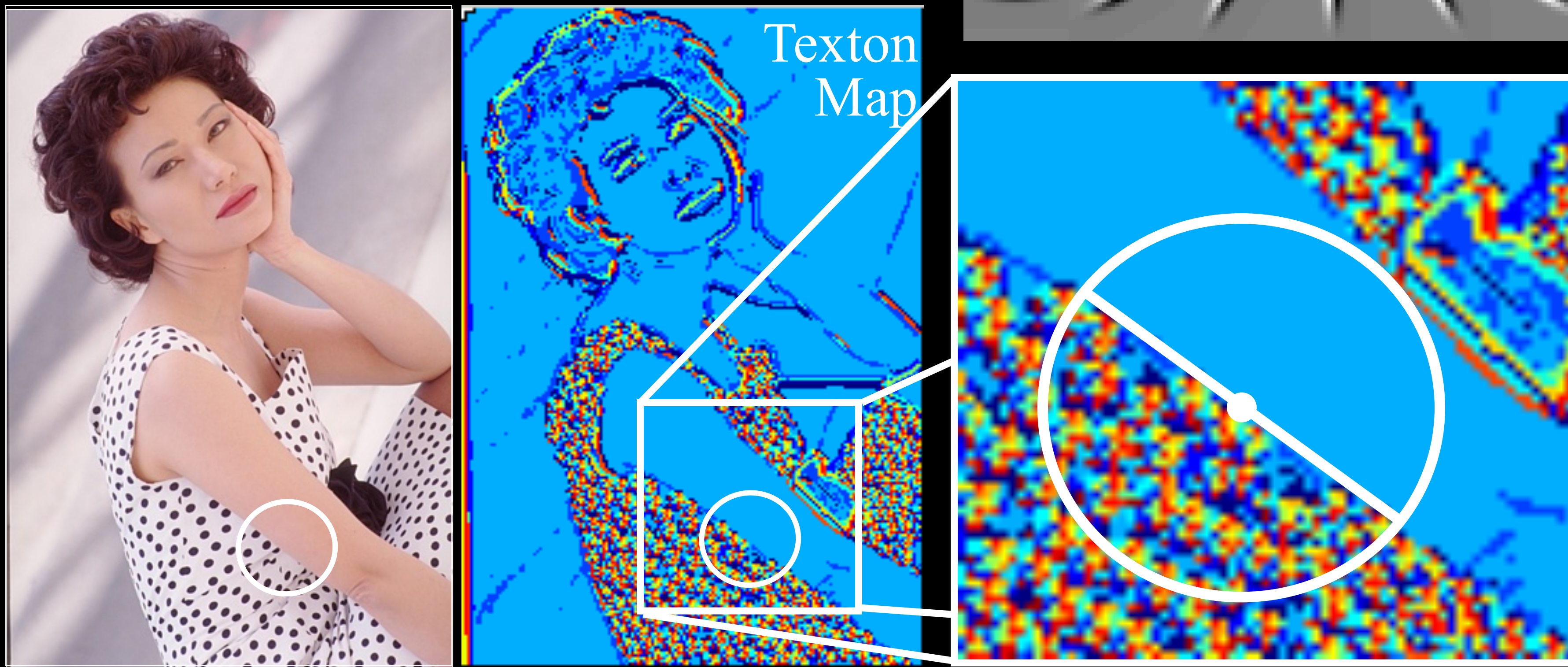
Plan

- What is Clustering? Challenges in Clustering
- Clustering (for Segmentation)
  - K-Means
  - GMMs (and Expectation-Maximization)
  - Mean-Shift
- Other uses of clustering in vision
  - Texture and Textons
  - Quantization
  - Bag of Words

Source: Martin, Fowlkes, Malik NIPS 2002 slides.

Texture Feature

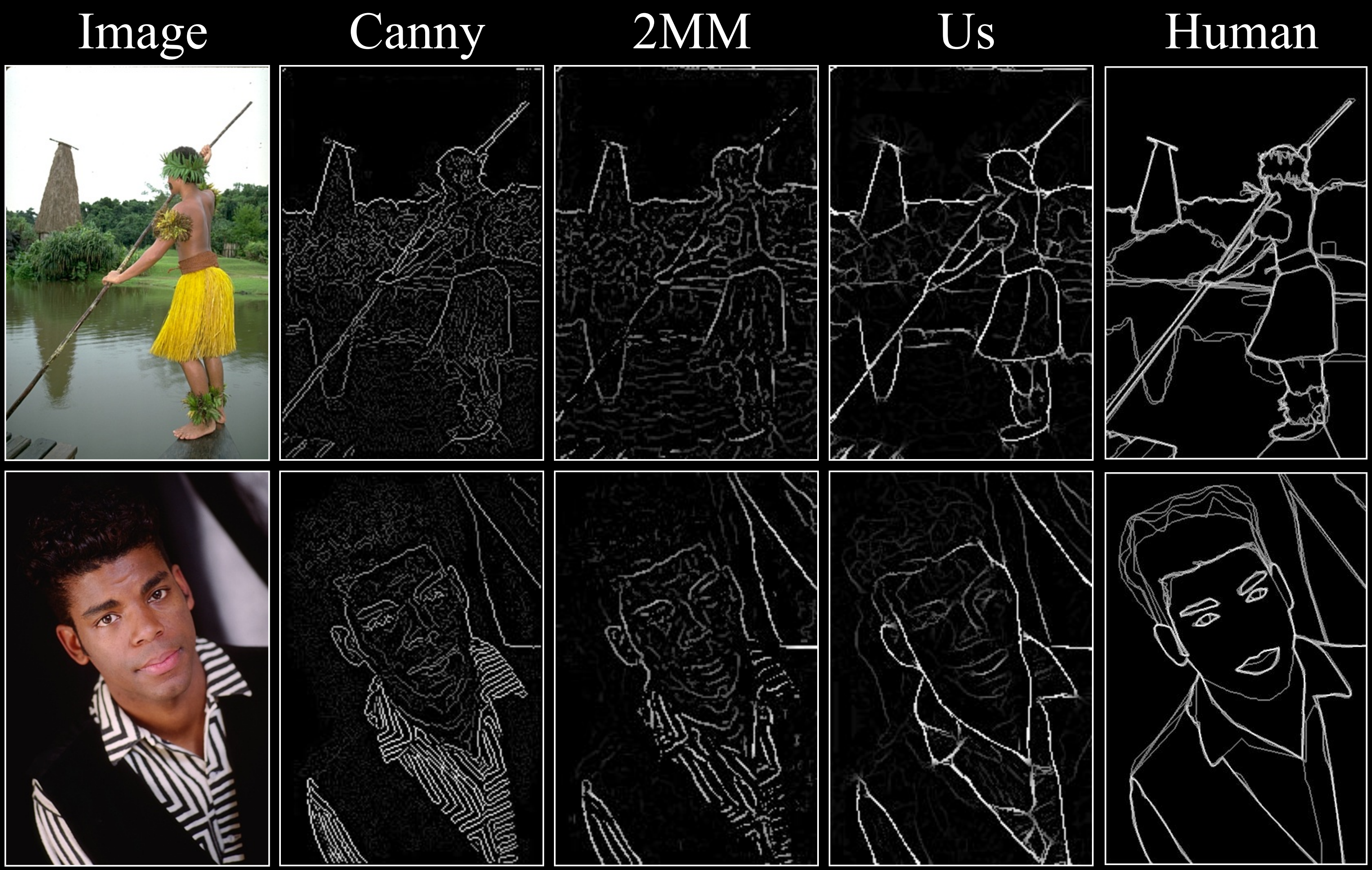




- Texture Gradient TG(x,y,r,θ)
  - χ^2 difference of texton histograms
  - Textons are vector-quantized filter outputs

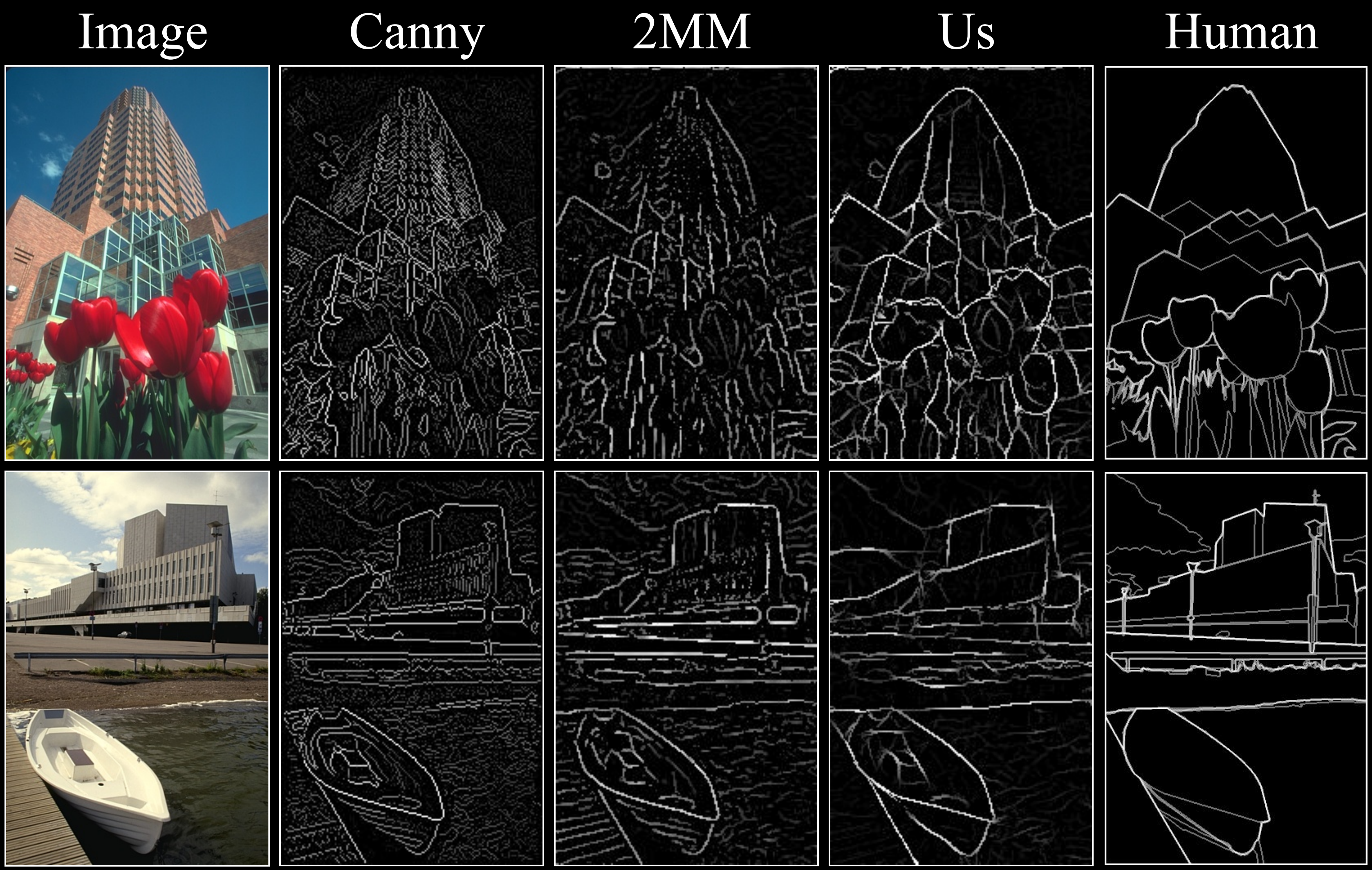
Source: Martin, Fowlkes, Malik NIPS 2002 slides.

P_b Images I



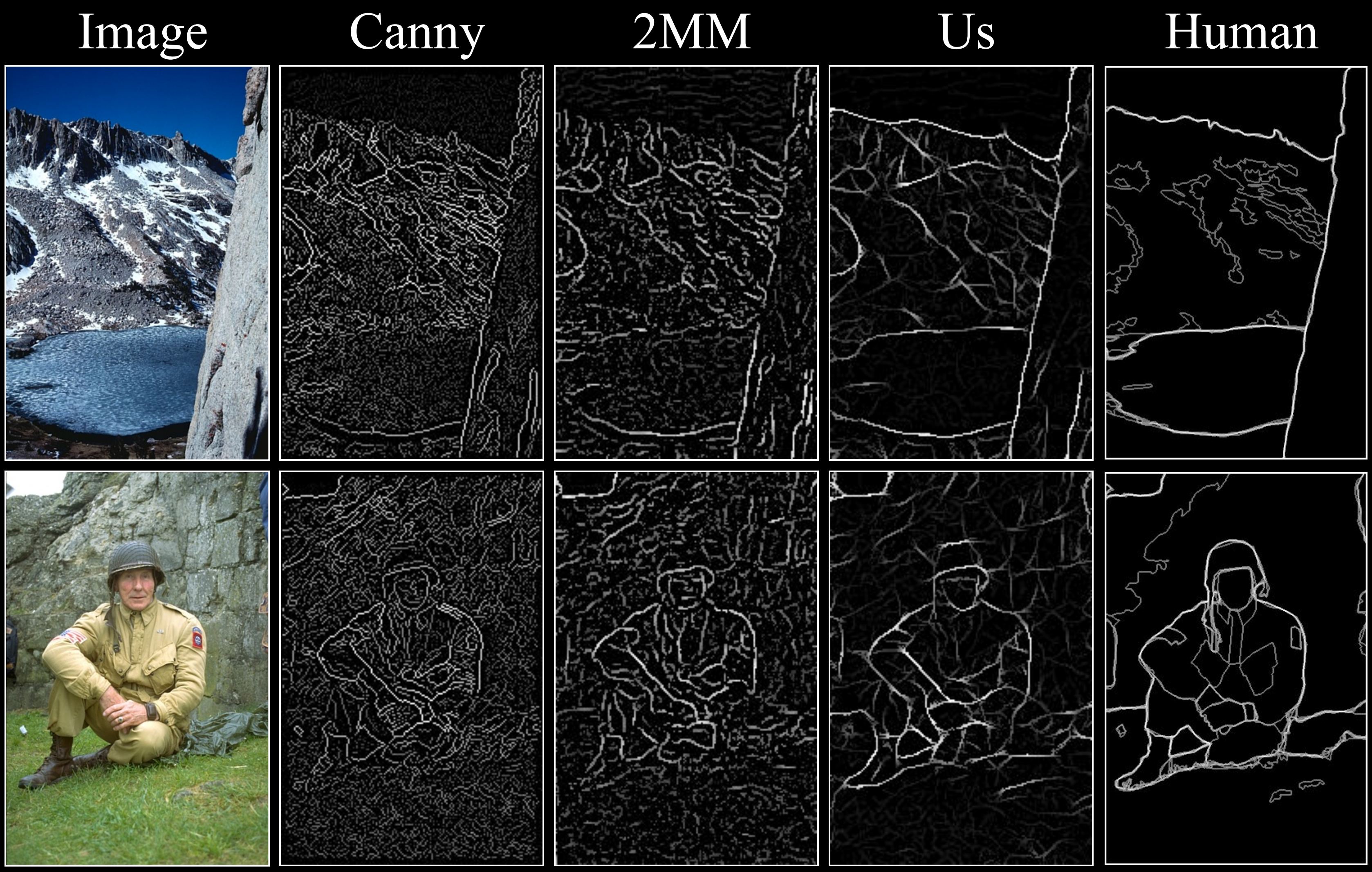
Source: Martin, Fowlkes, Malik NIPS 2002 slides.

P_b Images II



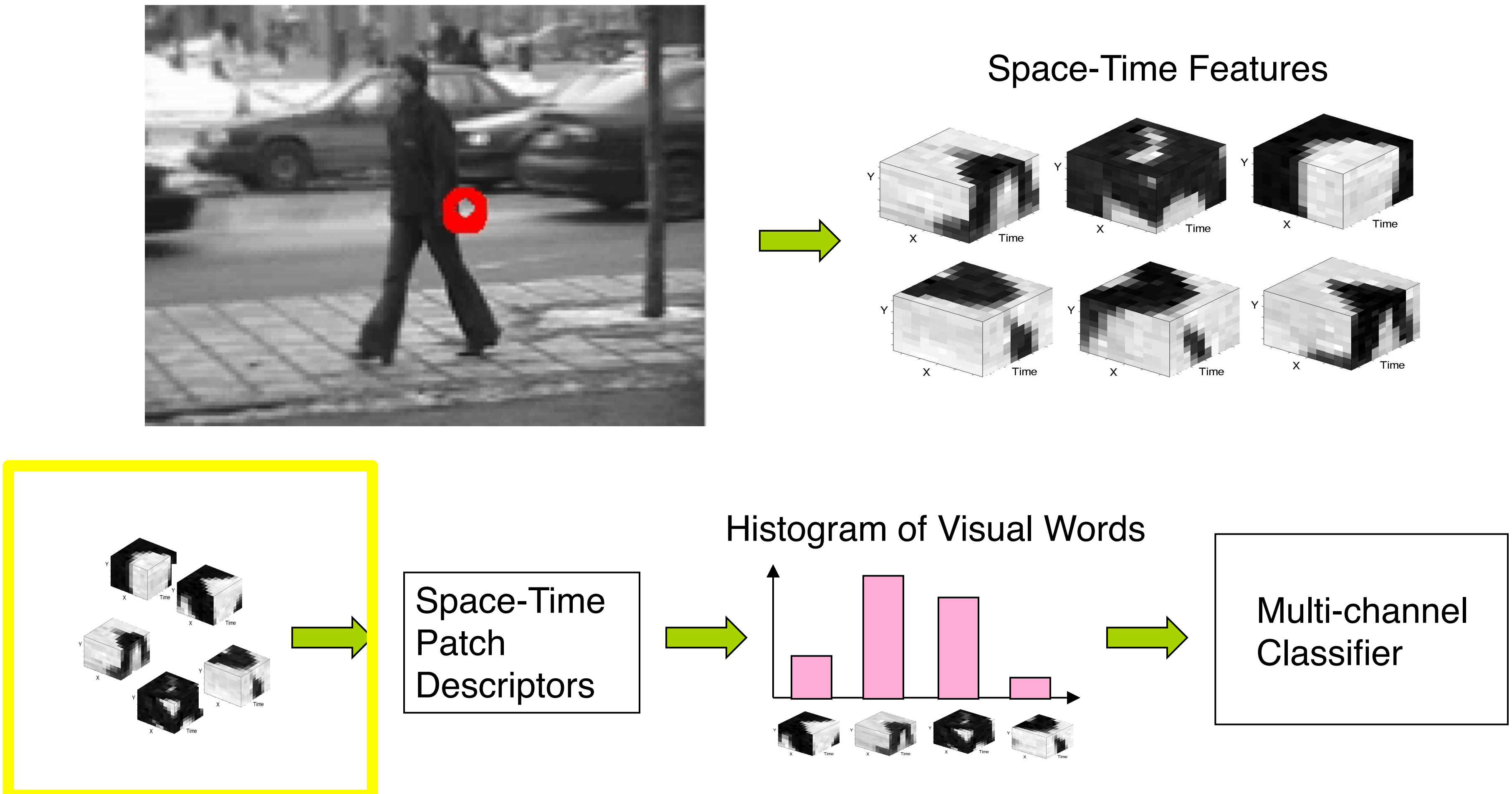
Source: Martin, Fowlkes, Malik NIPS 2002 slides.

P_b Images III



The (Very Common) Bag-of-Features Pipeline

Source: materials adapted from Laptev's CVPR 2008 slides.



- Examples include Schüldt et al. ICPR 2004, Niebles et al. IJCV 2008, and many works building on this basic idea.

Next Lecture: Model-Fitting and Contours

- Readings: FP 10; SZ 4.3, 5.1