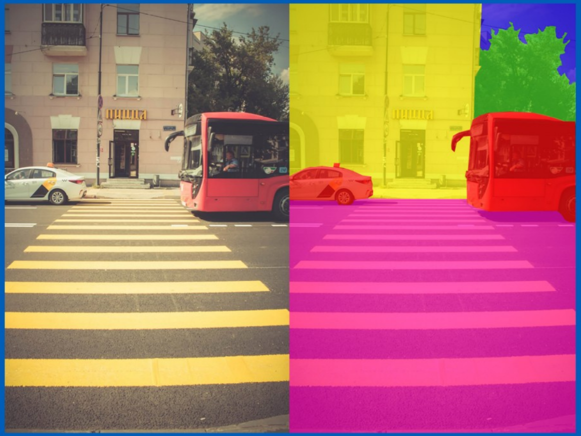
UNIVERSITÀ DI PARMA

# Image Segmentation



# Summary

UNIVERSITÀ DI PARMA

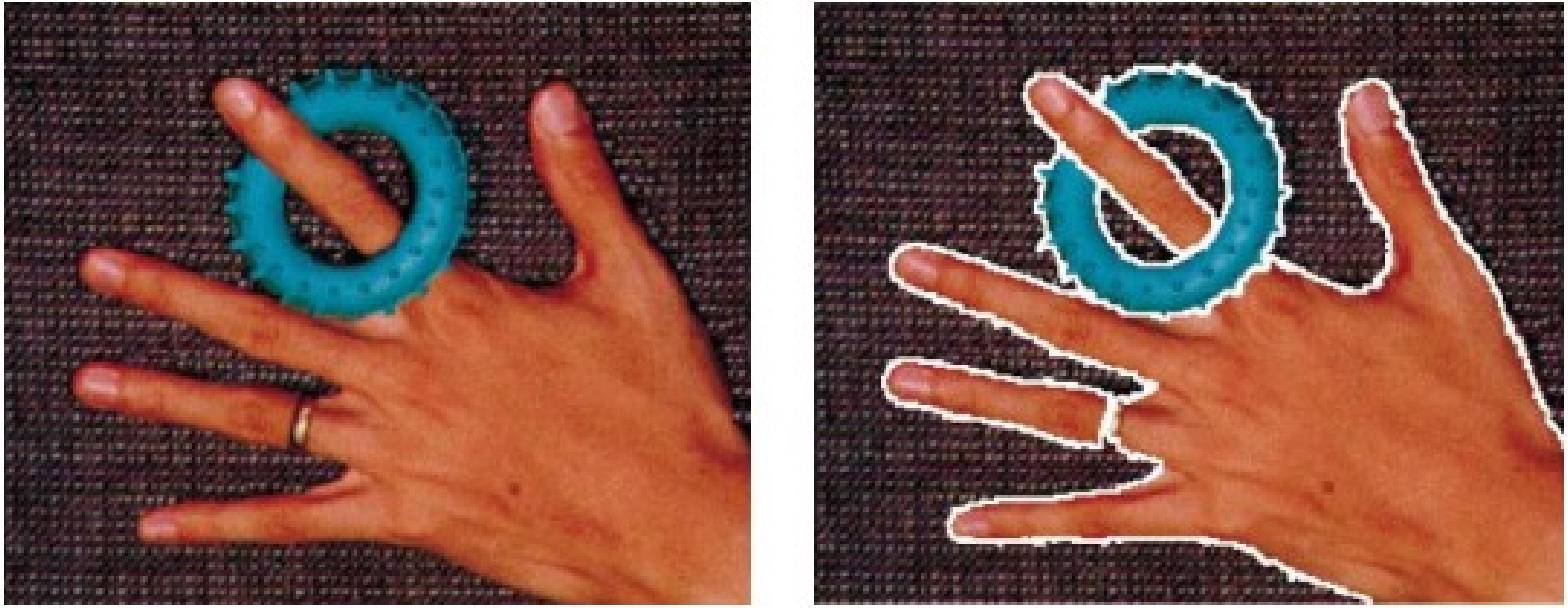
- What does “segmentation” means?
- K-Means
- Mean-shift

# References

- [FP] D. A. Forsyth and J. Ponce. Computer Vision: A Modern Approach (2nd Edition). Prentice Hall, 2011.
- CS231A · Computer Vision: from 3D reconstruction to recognition
  - Prof. Silvio Savarese – Stanford University
- CS131 · Computer Vision: Foundations and Applications
  - Prof. Fei-Fei Li – Stanford University
- Elementi di Analisi per Visione Artificiale
  - Paolo Medici http://www.ce.unipr.it/people/medici

# What does segmentation means?

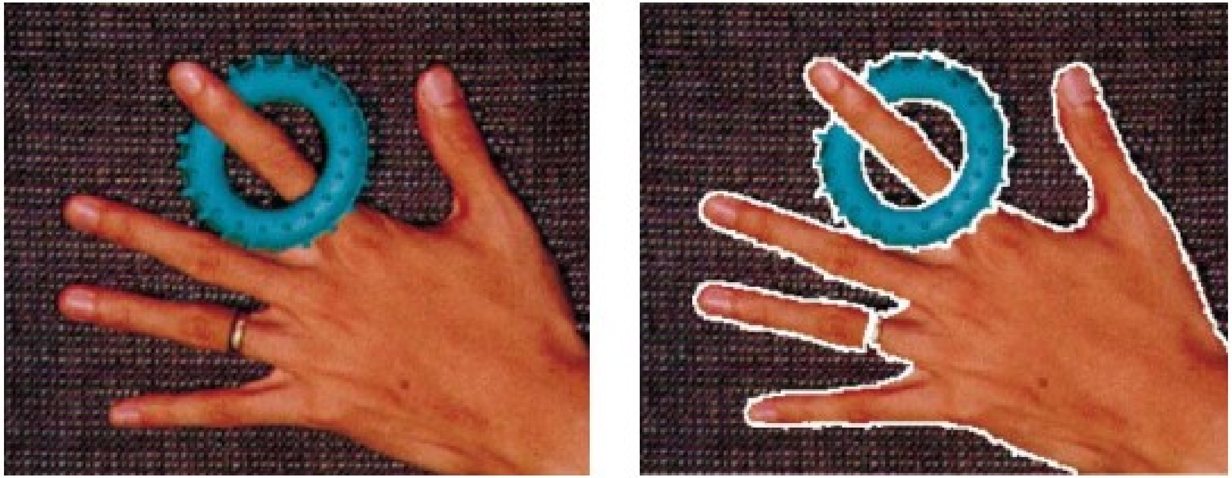
- Extraction of homogeneous parts of an image → **components**
- Usually a more compact representation of an image



Source: Silvio Savarese

# Why we use segmentation?

- From image to description
  - Low level $\longrightarrow$ medium level
- Components are often blocks for other algorithms



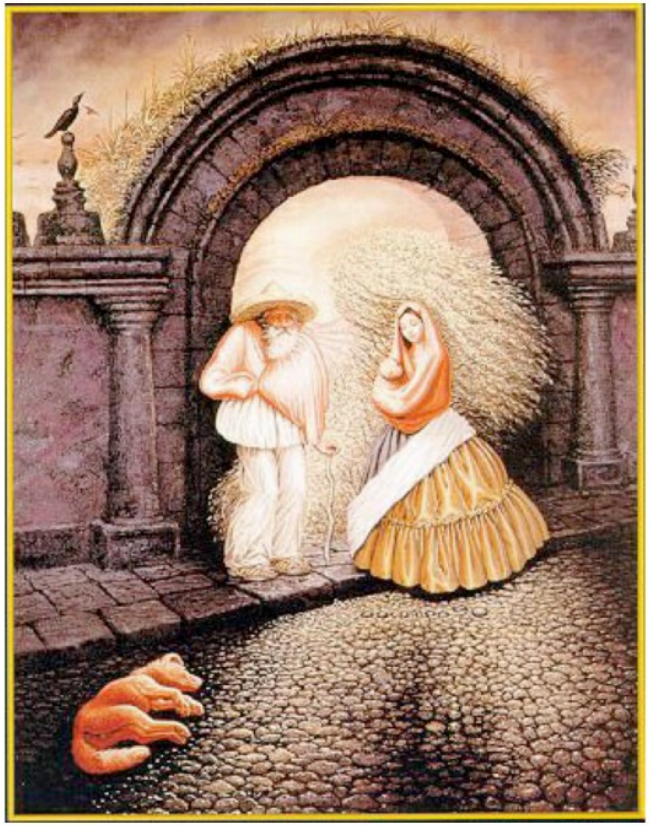
# What does segmentation means?

UNIVERSITÀ DI PARMA

- Extraction of homogeneous parts of an image $\longrightarrow$ **components**
- Usually a more compact representation of an image
- Components share common properties
- Different levels of abstraction for properties

Source: Silvio Savarese

# What does segmentation means?



Source: Silvio Savarese

# What does segmentation means?



Douillard, et al. ICRA 2011

http://www-rech.telecom-lille1.eu/shrec2012-segmentation/

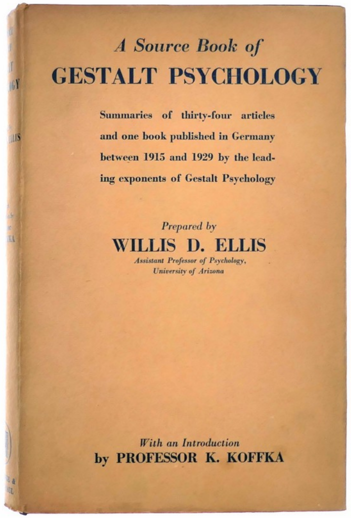
Source: Silvio Savarese

# How do humans segment?

# How do humans segment?

# Gestalt

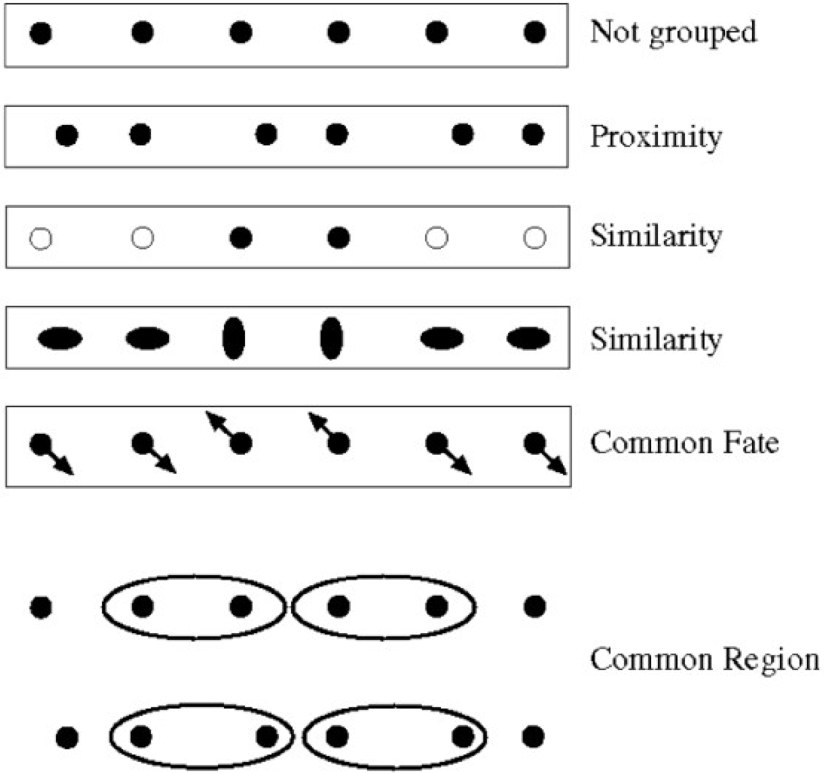
- German word for "shape", "structure"
- The brain is holistic and parallel
  - Namely no "simple steps" as for algorithms
- Whole is greater than the sum of its parts
  - We perceive objects in their entirety
  - Only after that we perceive details
- What are the principles that allow us to do so?



# Gestalt



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




Source: Silvio Savarese

# Gestalt



Grouping by occlusions

Source: Silvio Savarese

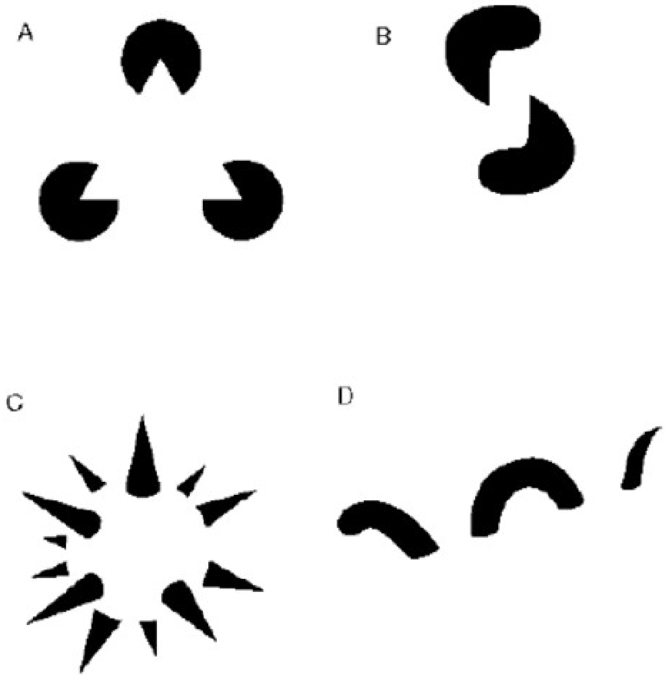
# Gestalt

Grouping
by occlusions

Source: Silvio Savarese

# Gestalt

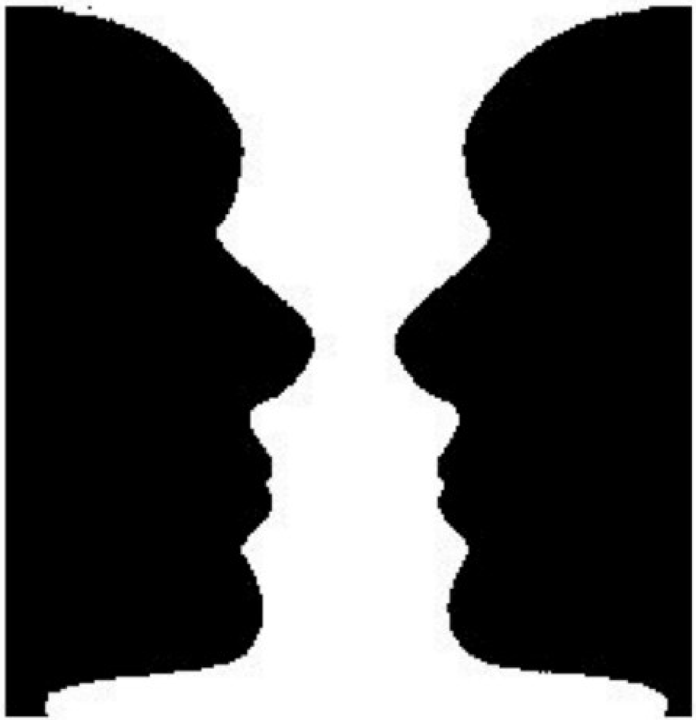
Grouping
by invisible
completions

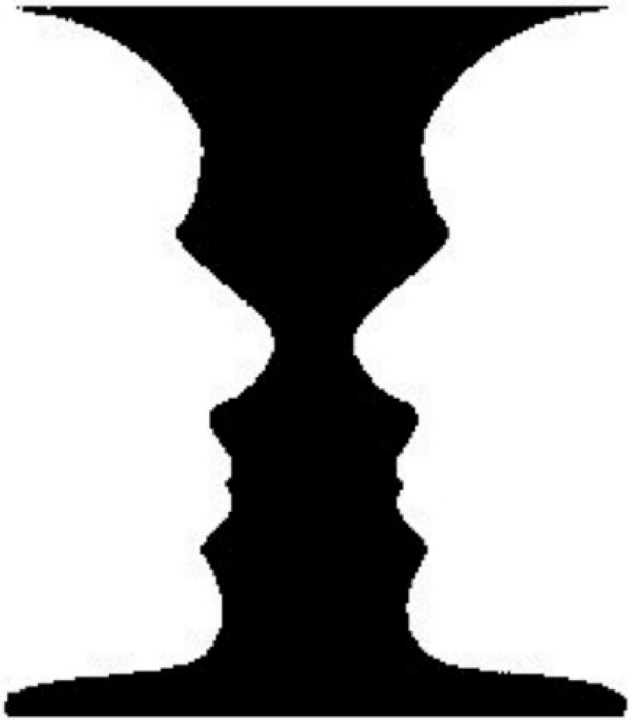



Source: Silvio Savarese

# Gestalt





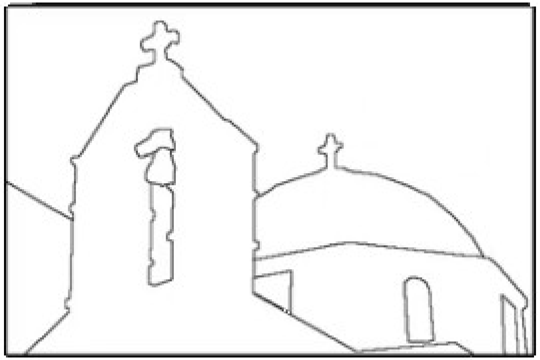


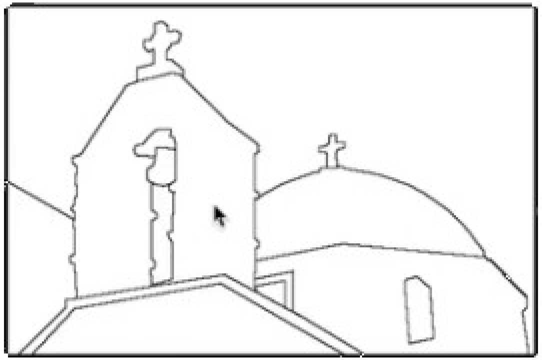
Source: Silvio Savarese

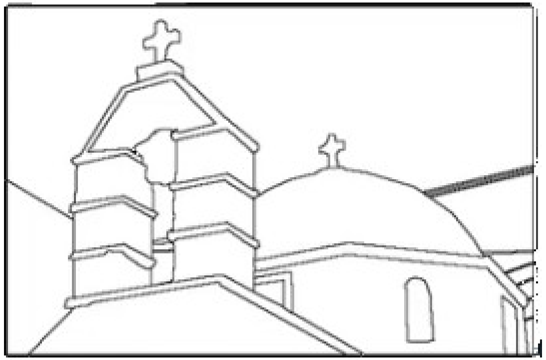
# Gestalt

If we ask to different users to segmentate, different results are obtained


**Segmentation is HIGHLY Subjective!**







Source: Shree Nayar

# Gestalt

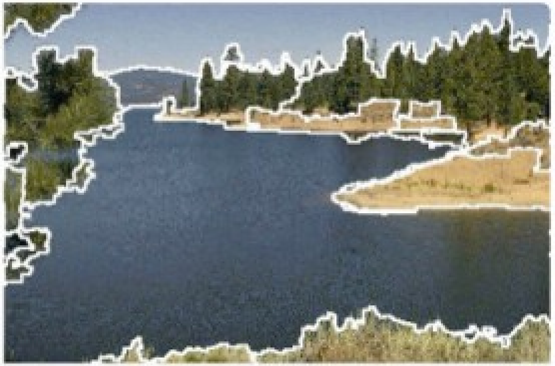


- Segmentation is intuitive for us
  - Even if highly subjective
- Very hard to translate it to an algorithm!
  - If we do not know how it works, how we can write an algorithm?
- Anyway two basic approaches

# Strategies

UNIVERSITÀ DI PARMA

- Bottom-up: components are locally coherent

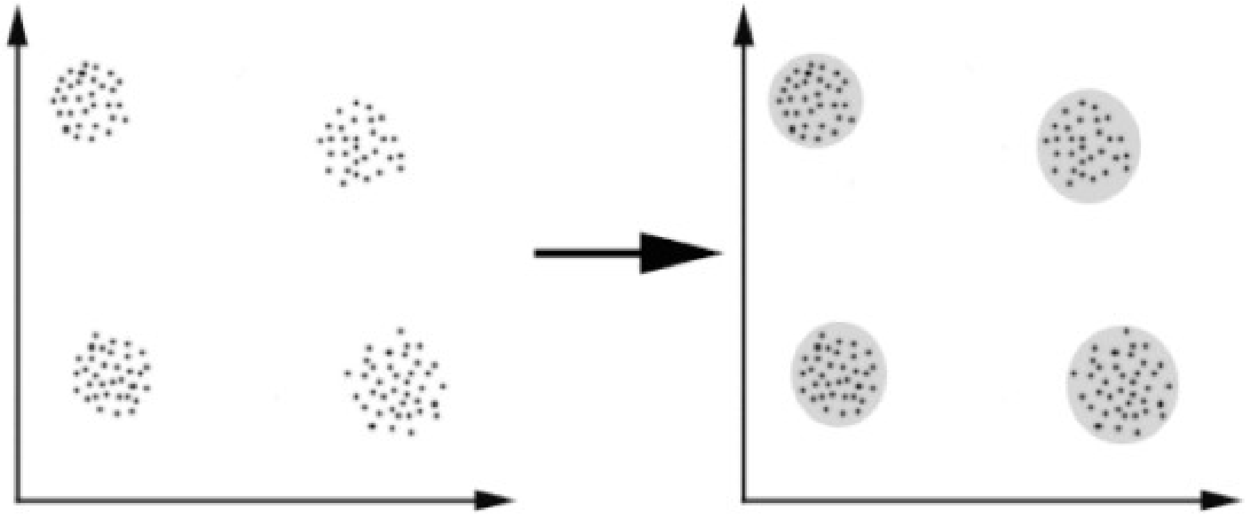


- Top-down: components belong to same entity (object, scene…)


Source: Silvio Savarese

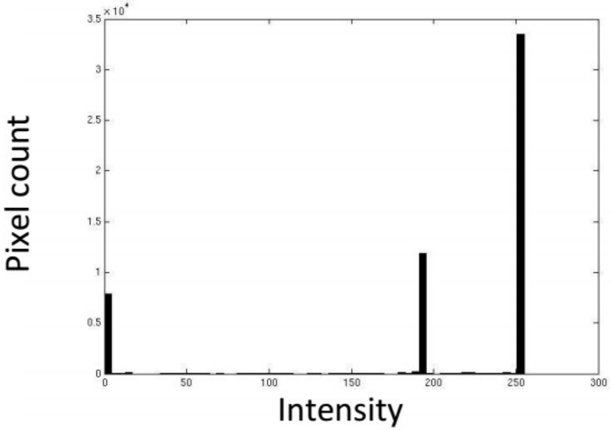
# Clustering

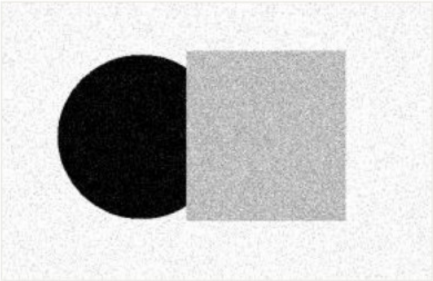
- Each point belongs to a specific “cluster”
- It depends on points visual characteristics
  - A “vector”



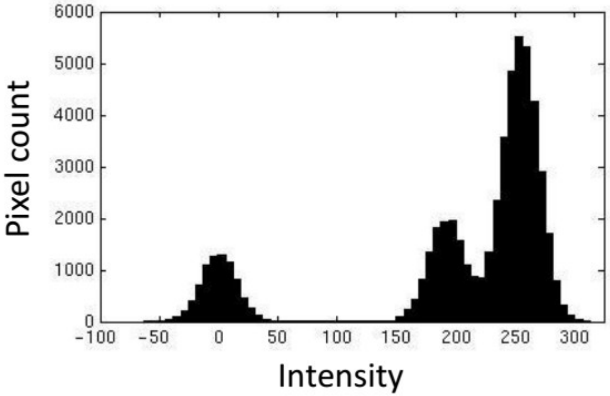
# Clustering

Input image



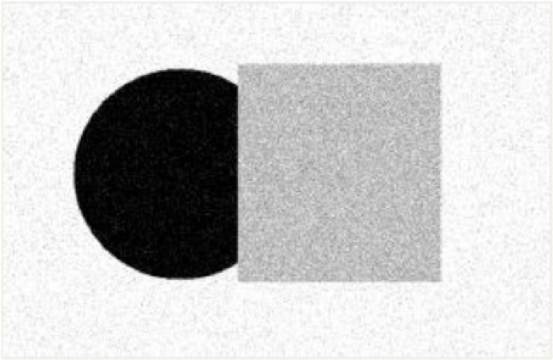
Input image



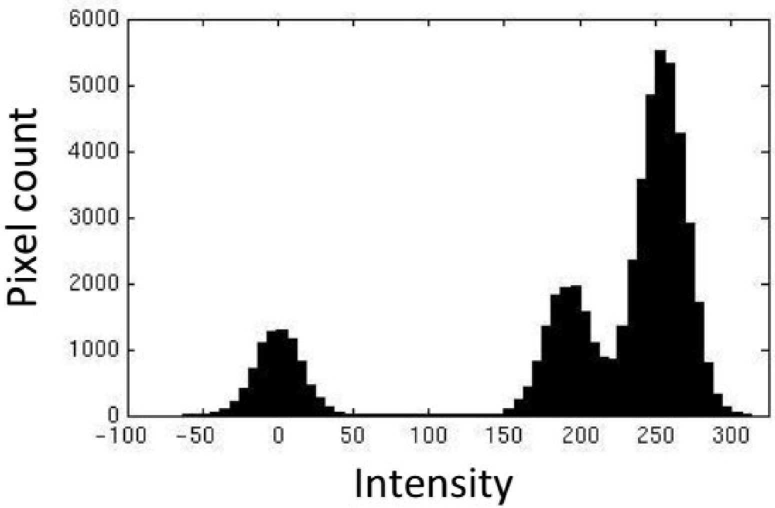

Slide credit: Kristen Grauman

# Clustering

- Clustering → identify different luminance levels

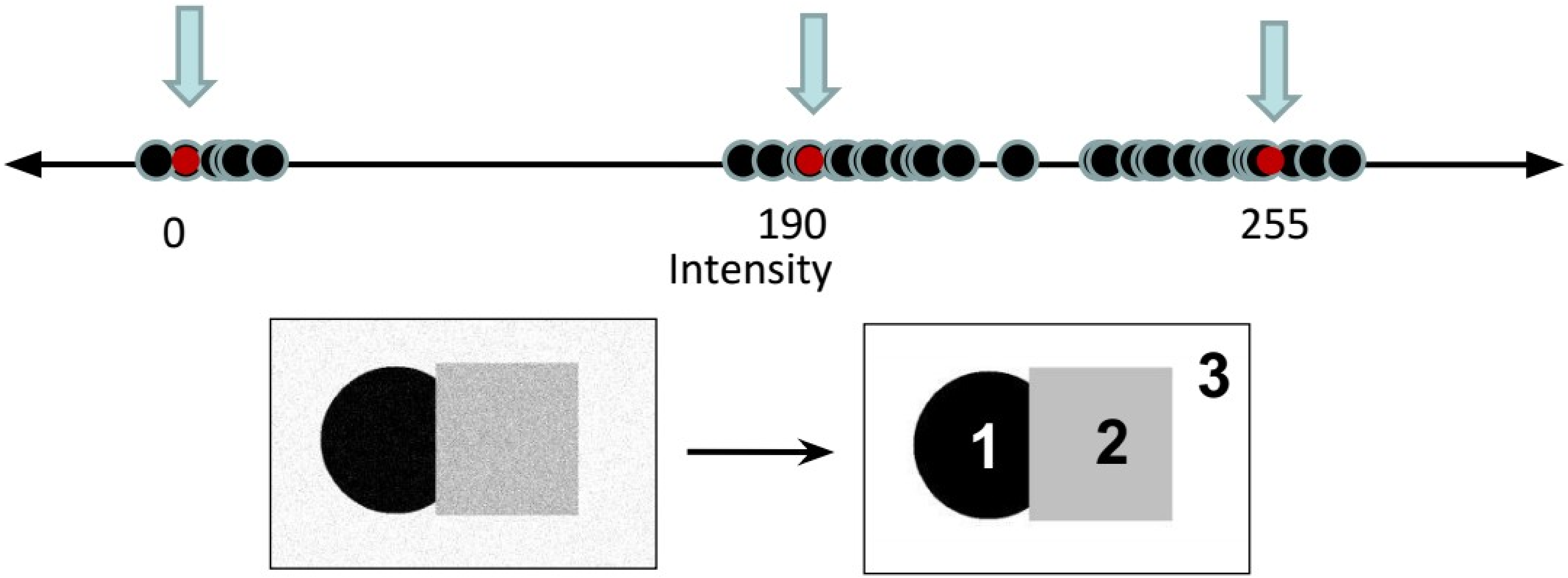


Input image



# Clustering

- Clustering → identify different luminance levels
  - Associate each pixel to a specific cluster



# Clustering

UNIVERSITÀ DI PARMA

- Underliying idea: minimize "distance" of each "pixel" from cluster center
  - Given N data & k clusters:

Is $x_j$ belonging to $c_i$?

$$c^*, \delta^* = \underset{c,\delta}{arg\ \ min} \frac{1}{N} \sum_j^N \sum_i^k \delta_{ij} (c_i - x_j)^2$$

Cluster center

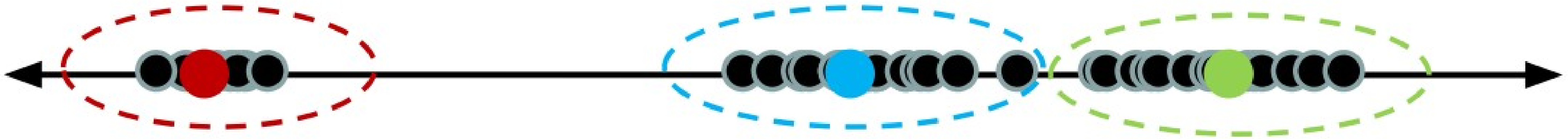
Data

Source: Silvio Savarese

# Clustering

UNIVERSITÀ DI PARMA

- Not well defined...
  - If we know cluster centers $\longrightarrow$ easy points assignement
  - If we know groups $\longrightarrow$ easy to compute centroids



# Clustering techniques

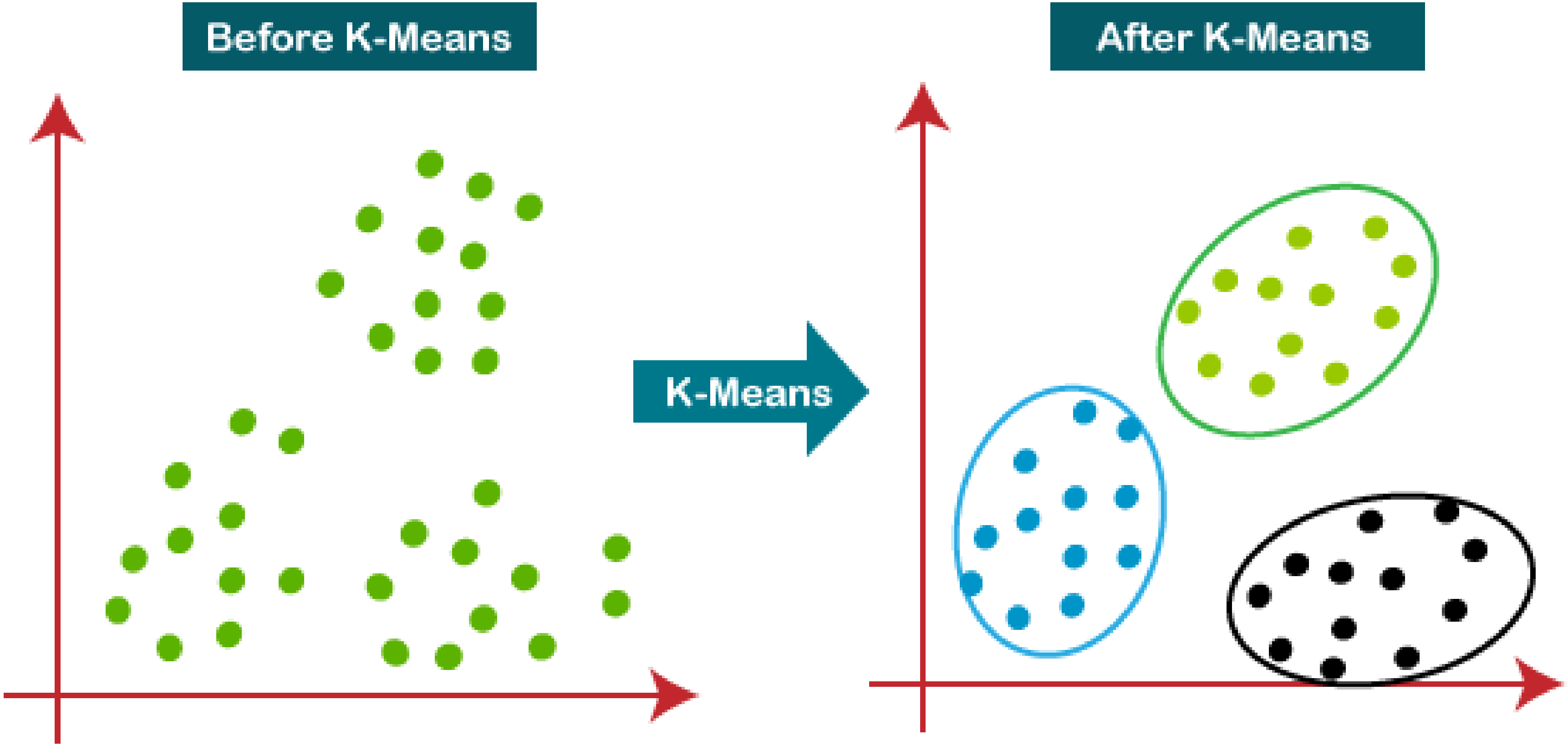
- K-Means
  - Predefined number of clusters
  - “Spherical” clusters
- Mean-shift
  - Variable number of clusters
  - No a priori shape assumptions
  - Slow

UNIVERSITÀ DI PARMA

# K-means clustering

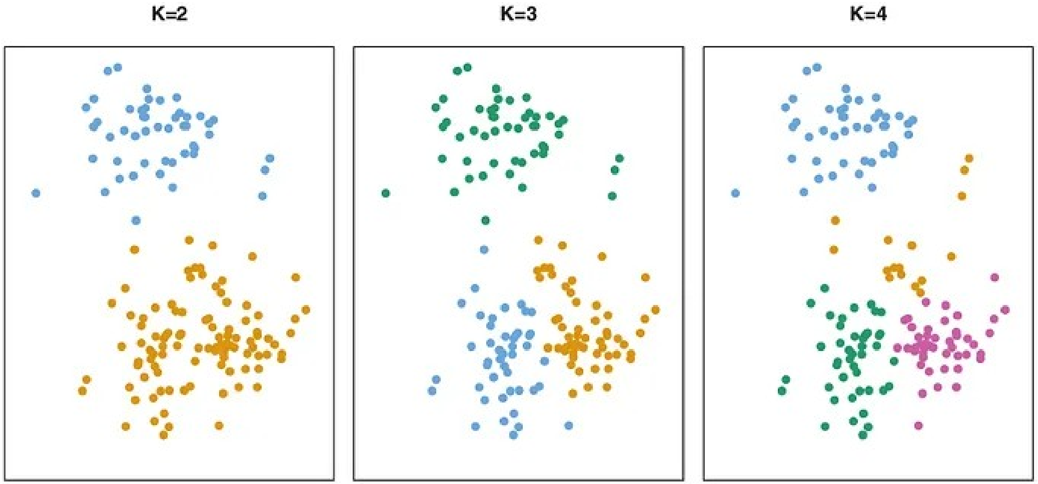
# K-Means clustering

UNIVERSITÀ DI PARMA



# K-Means clustering

- The underlying idea under k-means clustering is to guess the number of clusters
  - Namely the "k" in k-means




Source: Introduction to Statistical Learning with Applications in R page 387

# K-Means clustering

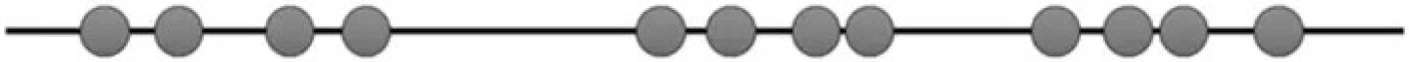
UNIVERSITÀ DI PARMA

- Initialization
  - Choose $k$ cluster centers
- Repeat
  - Assignement step:
    - For each point find the closest center
  - Update step
    - Update every center as the mean of its points
- Until
  - A maximum number of iterations is reached
  - No or little changes during Assignement step

Source: Pranshu Sharma

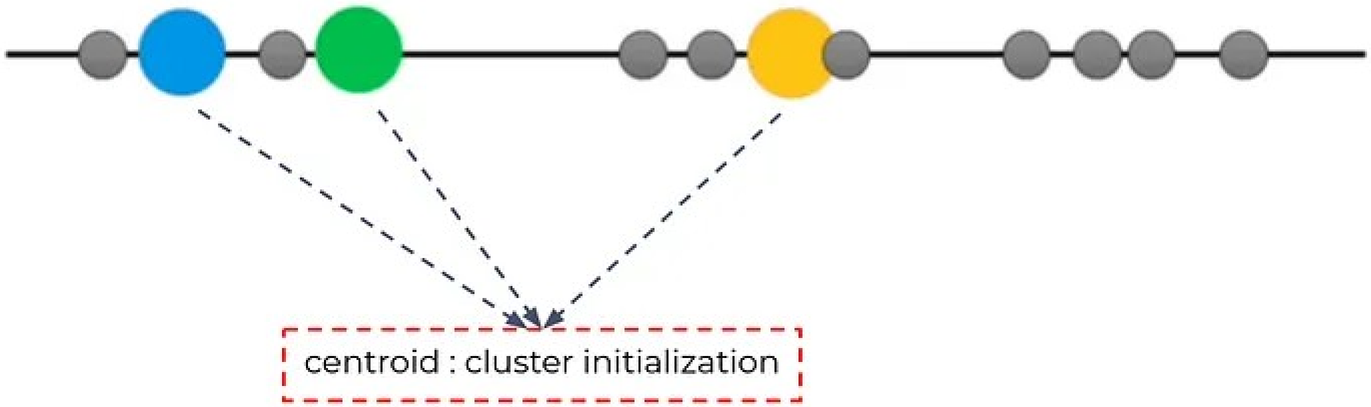
# K-Means clustering

- Initialization
  - Choose $k$ cluster centers
  - The value can be critical, since it can lead to completely different results
- How to select them?
  - randomly
  - $k$ points of the set



# K-Means clustering

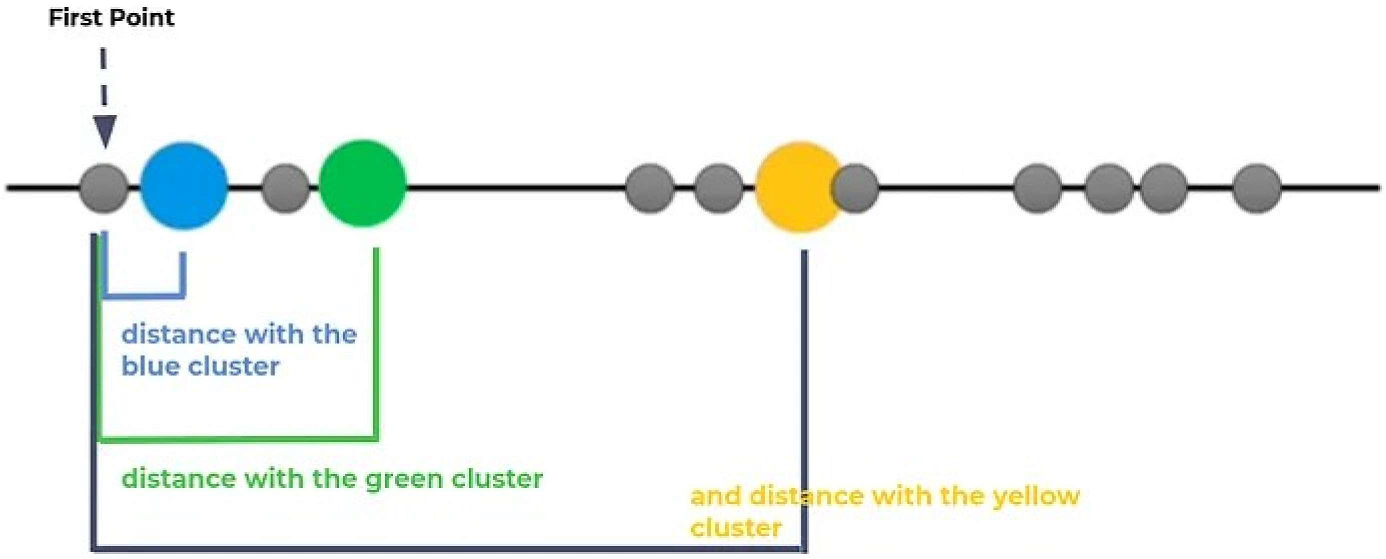
- In our example
  - $k = 3$
  - 3 points are selected as center of clusters



# K-Means clustering

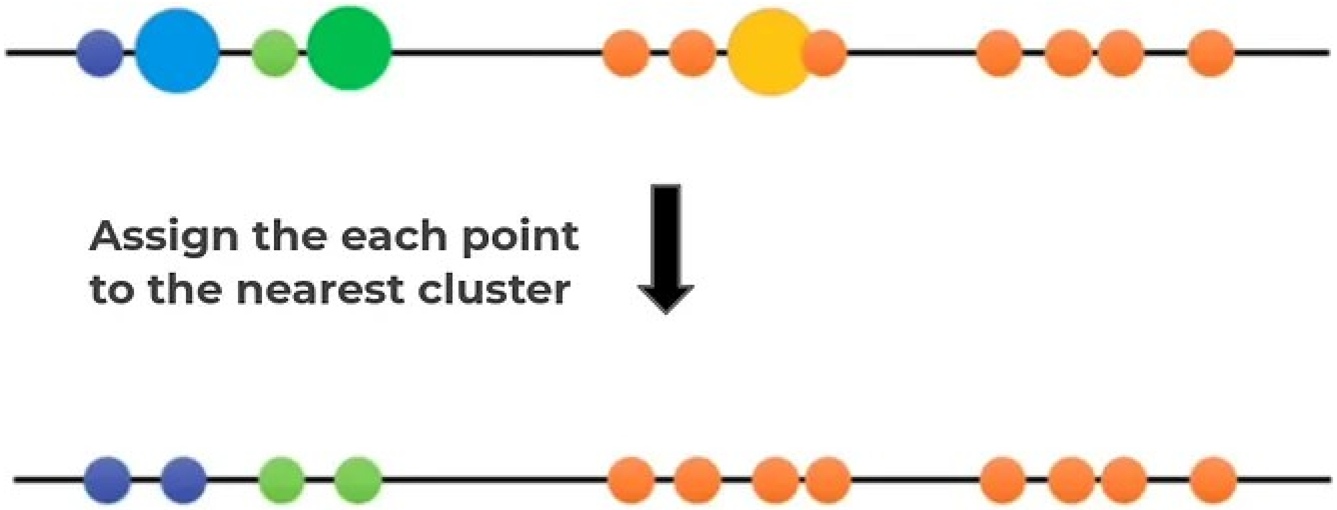


- Assignement
  - For each point find the closest center
  - In our case euclidean distance is used



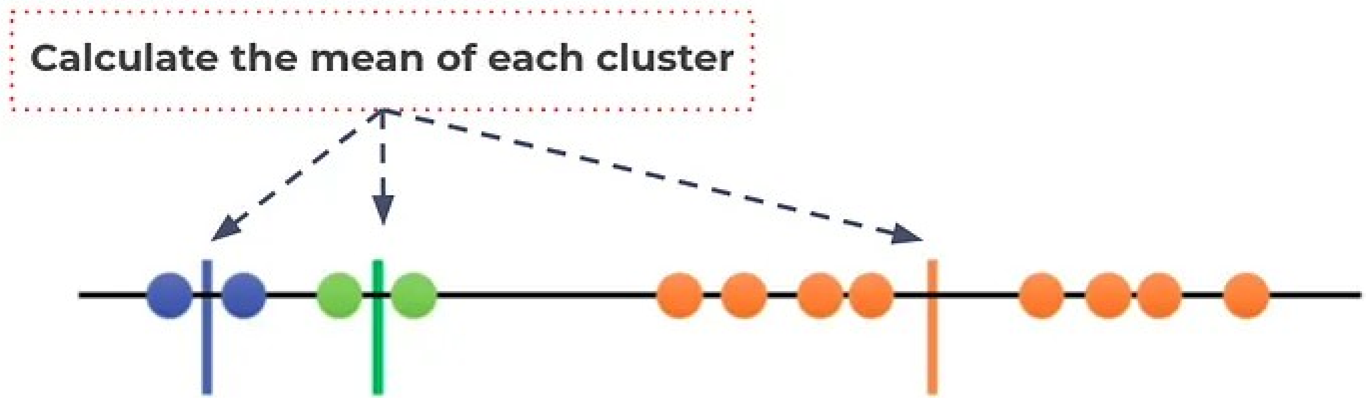
# K-Means clustering

- Assignement
  - For each point find the closest center
  - In our case euclidean distance is used



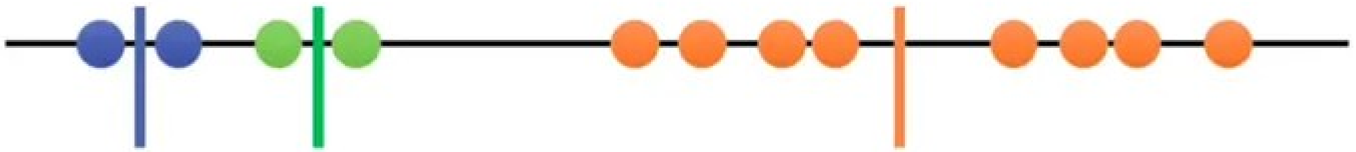
# K-Means clustering

- Update step
  - Update every center as the mean of its points



# K-Means clustering

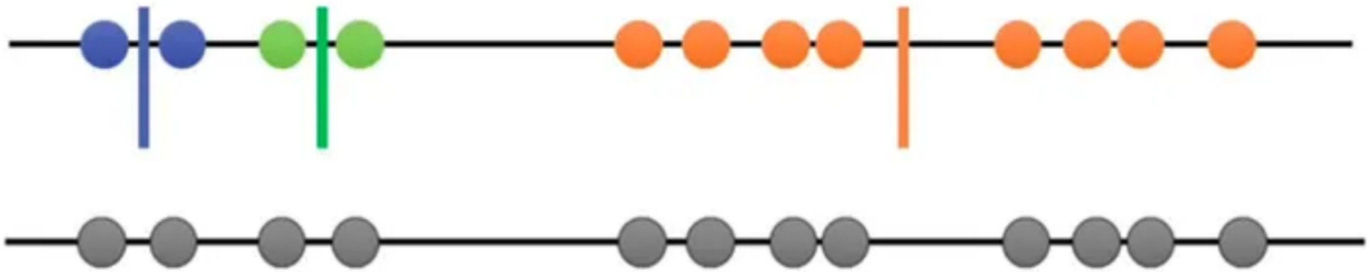
- Repeat
  - Assignement & Update
  - Until no or little changes or a fixed amount of iterations



# K-Means clustering



- K-means is extremely sensitive to initialization
- Bad initialization ⟶ bad overall clustering
- Potential solutions:
  - K "spread out" points (k-means++)
  - Try multiple inizializations



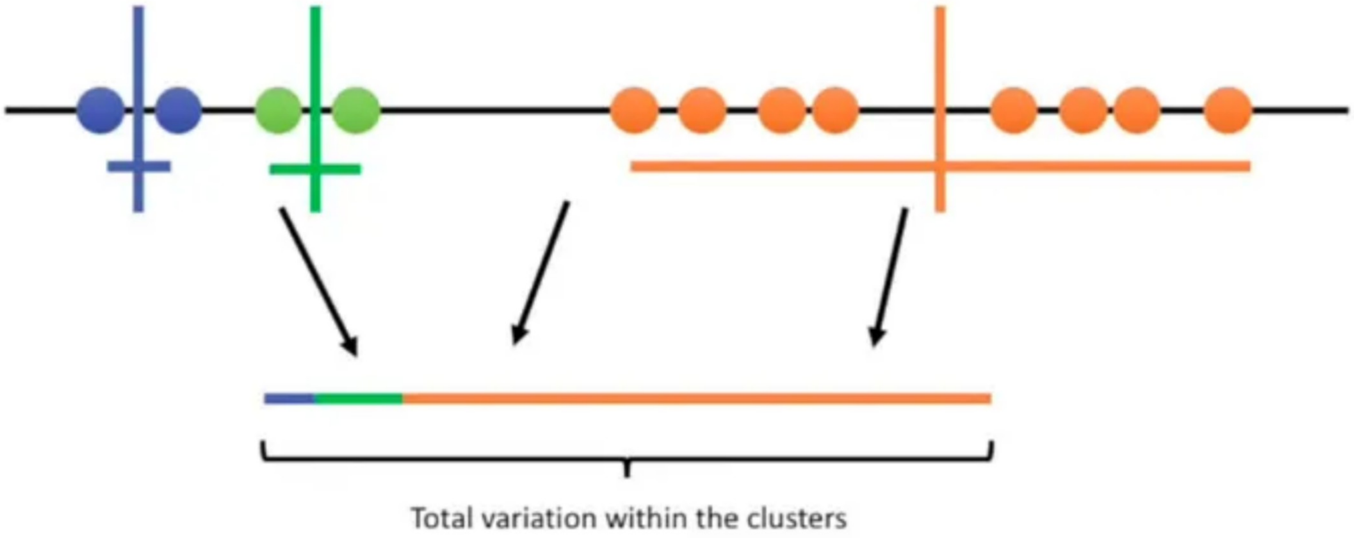
# K-Means clustering



- K-means++ approach for initialization
  - Randomly select first center
  - Pick other centers using a probability proportional to $\Sigma_i(x - c_i)^2$
  - Expected error ~ log(k)

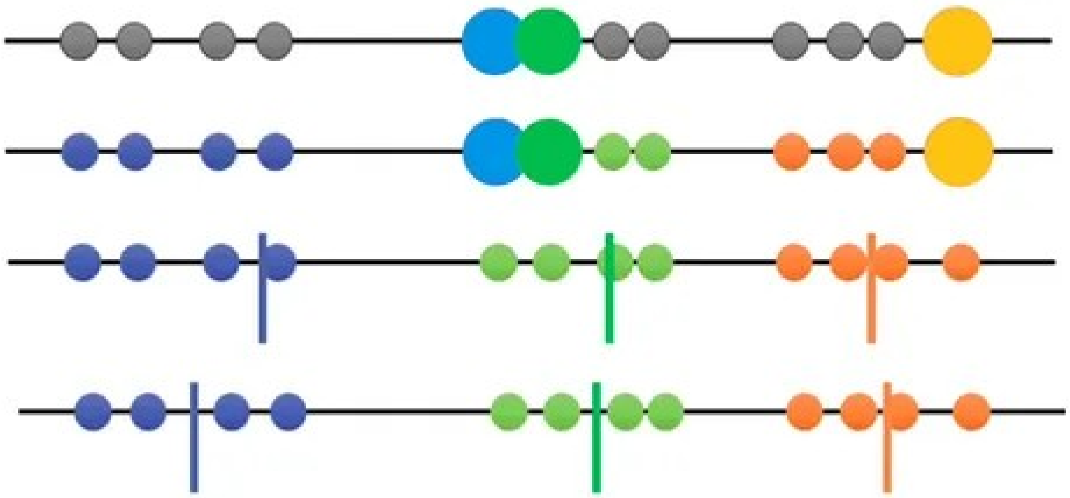
# K-Means clustering

- Multiple initializations
  - Given a first try, asses the quality of the result
  - Compute the variance of each cluster



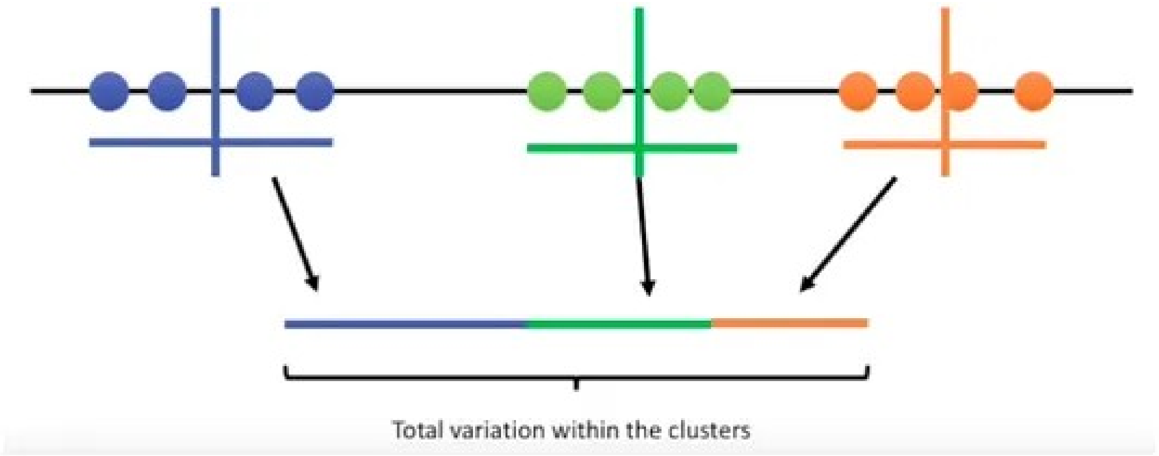
# K-Means clustering

- Repeat again "until" we obtain the minimum variance



# K-Means clustering

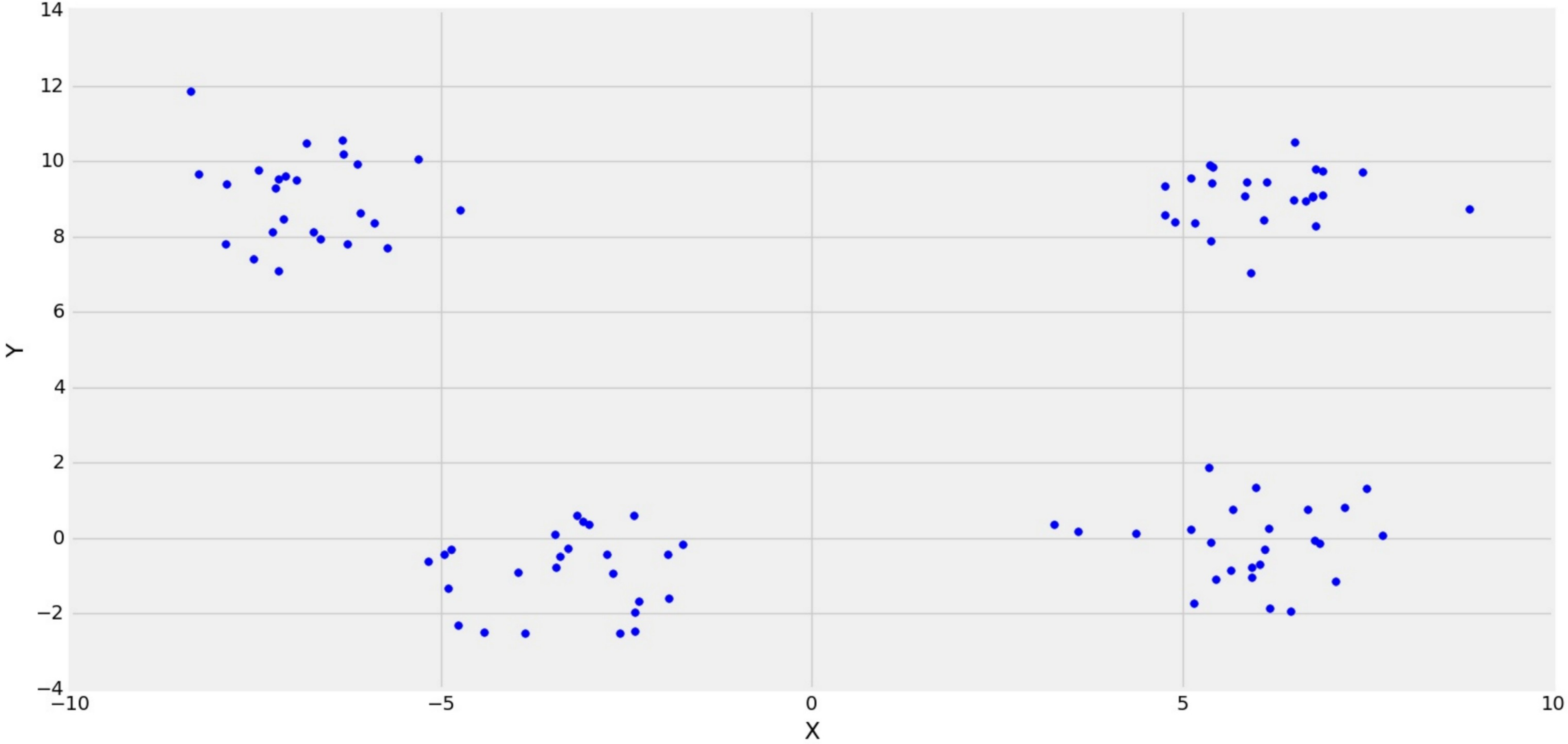
- Repeat again until we obtain the minimum variance



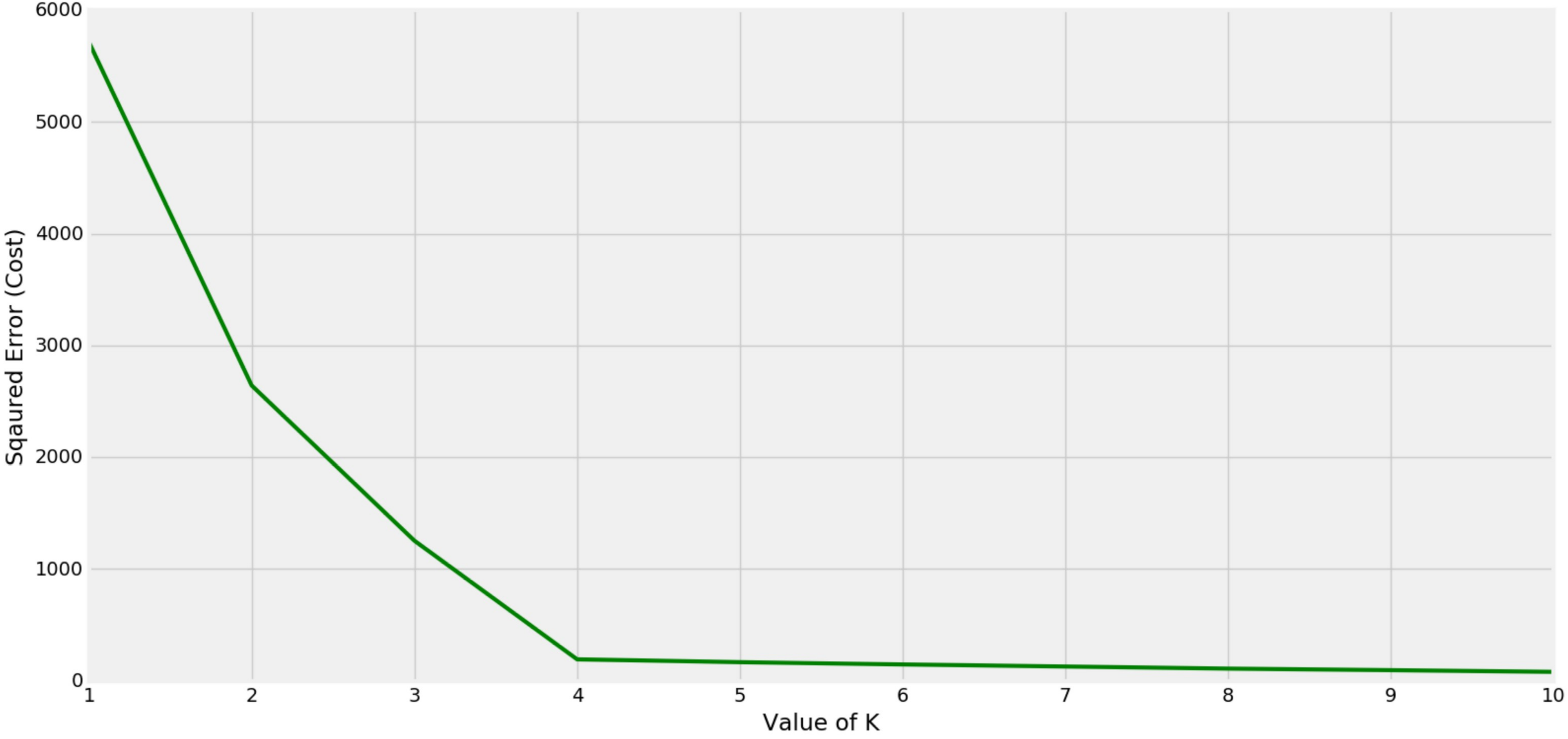
# K-Means clustering

UNIVERSITÀ DI PARMA

- How to choose a proper value for $k$?
- Test different $k$s
  - When k increases $\longrightarrow$ fewer elements in each cluster
    - Also average distortion declines
  - The k value where distortion declines the most is the “elbow” point



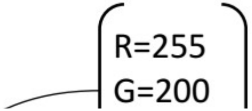
14
12
10
8
6
4
2
0
−2
−4
Y
−10
−5
0
5
10
X

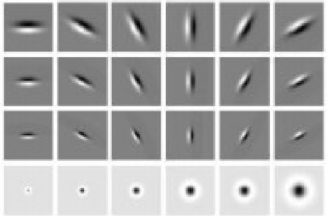


6000
5000
4000
3000
2000
1000
0
Sqaured Error (Cost)
1
2
3
4
5
6
7
8
9
10
Value of K

# Features

- In previous slides we used Euclidean distance as a cost function
- We can select other approaches
  - Intensity clustering
  - Color clustering
  - Texture clustering
  - …



Filter bank of 24 filters

# Features



- Not always image space coherence
- As example when using color clustering we have a vector clustering

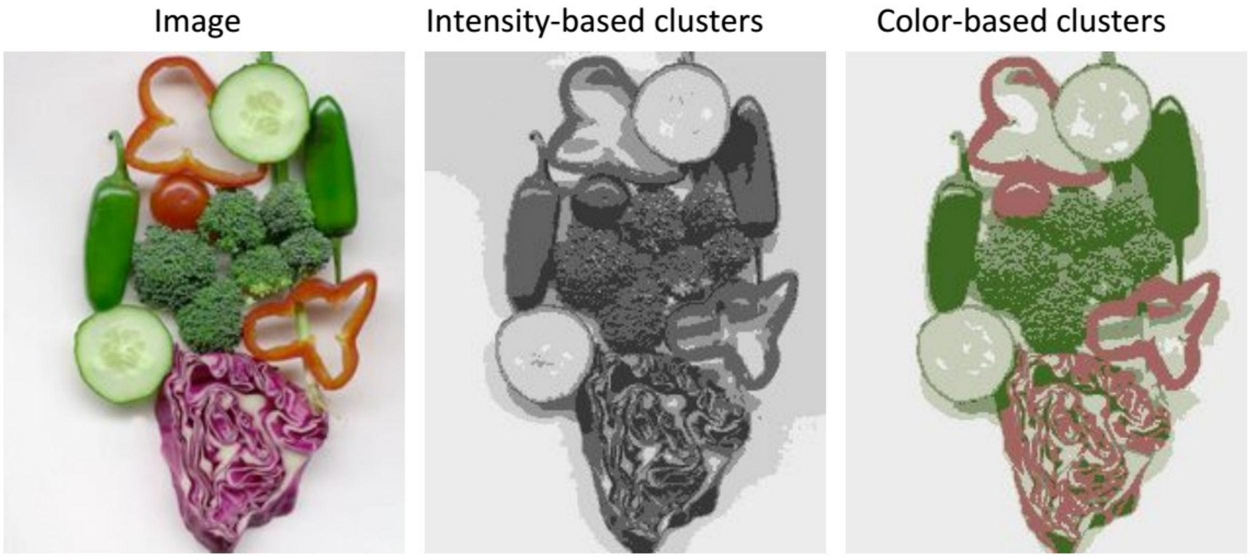
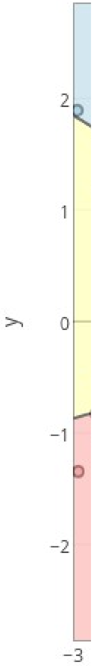



Image source: Forsyth & Ponce

# K-means limits

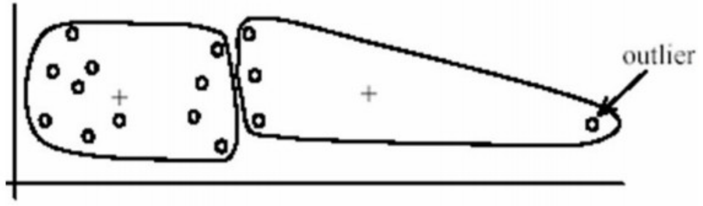


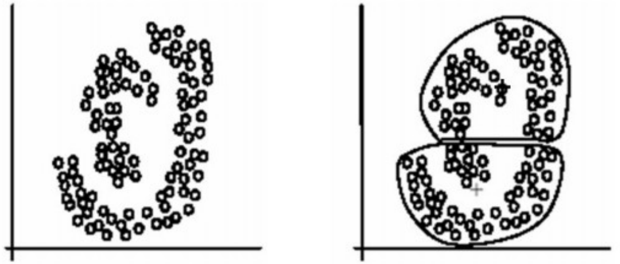
- Each point is assigned to a given cluster
  - No probability function
- Very sensitive to outliers
  - Centroids are affected
  - K-Medians is a K-Means variant to cope with this
- Good with “round shaped” clusters
- Bad with convex distributions
  - Voronoid space





(B): Ideal clusters



(A): Two natural clusters (B): *k*-means clusters

# K-means steps

UNIVERSITÀ DI PARMA

- Images can not be a suitable data for clustering
- We need to extract relevant data $\longrightarrow$ vector
- The output of k-means us then
  - A list of centroids
  - Labelled data
  - Compactness measure (quality of the result)

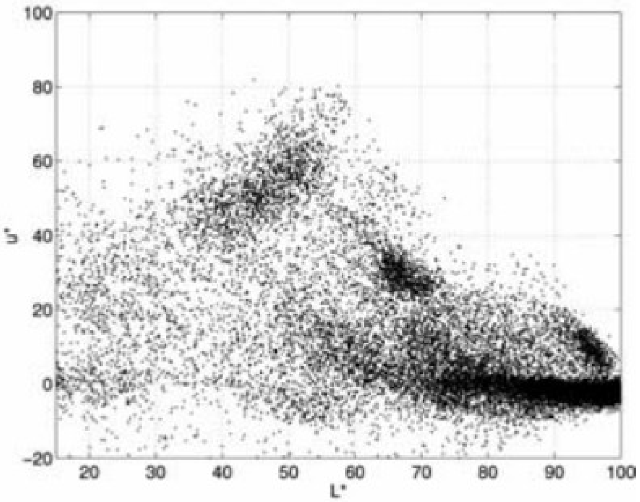
# K-means OpenCV

```
double cv::kmeans  (
     InputArray data,                              // vector of data
       int K,                                      // the K in K-means
       InputOutputArray bestLabels,                // labels (output/input)
       TermCriteria criteria,                      // termination criteria
       int attempts,                               // number of tries
       int flags,                                  // flags
       OutputArray centers = noArray()             // centroids list
    )
```

# Mean-shift clustering

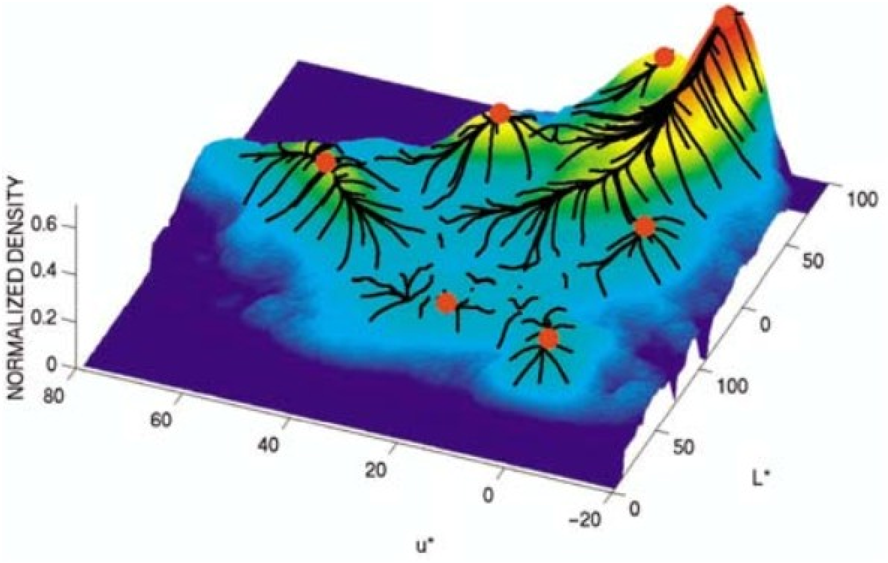
# Mean shift

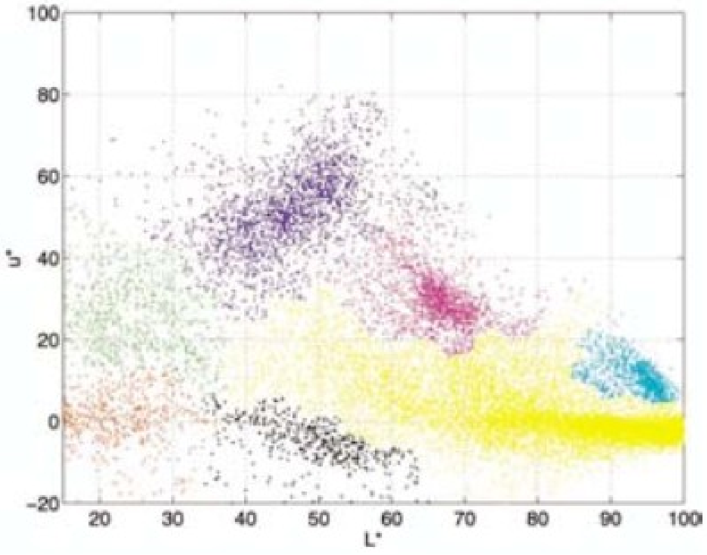
- As for k-means we work on data, not directly on images
  - For simplicity we assume a 2D data
  - Higher dimensionality is anyway possible



# Mean shift

- Looking at data we can imagine a potential clustering
  - Density based
  - Consider density as a third "dimension"

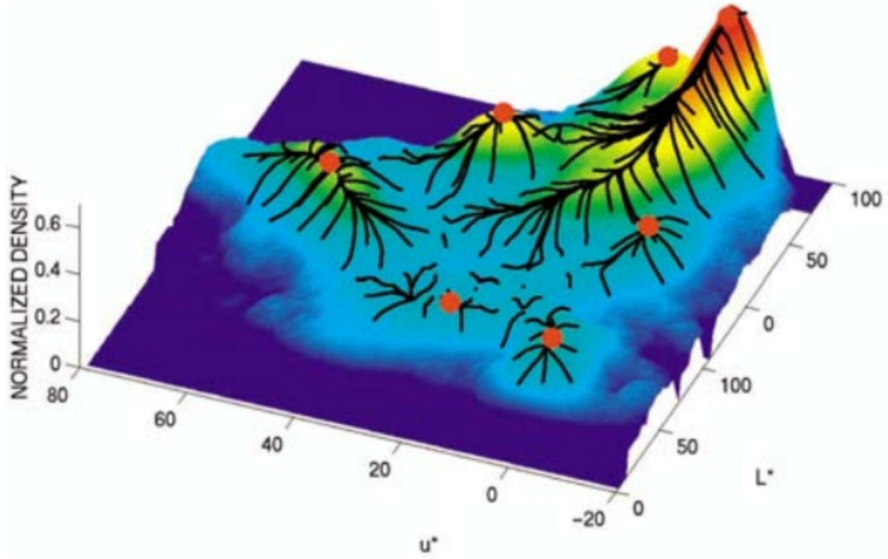


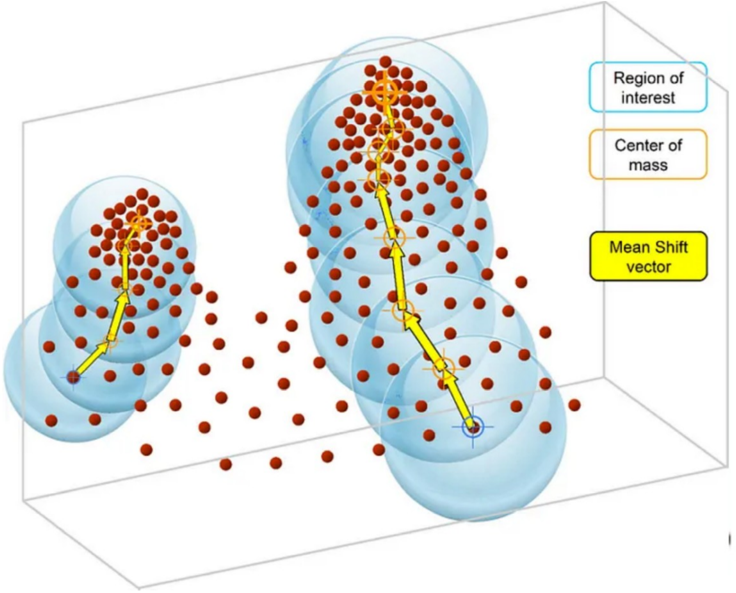



# Mean shift



- Iteratively we can have each datum to "climb" the hill
  - Expensive when a large or complex dataset is used
  - The use a reduced set of "seeds" is possible



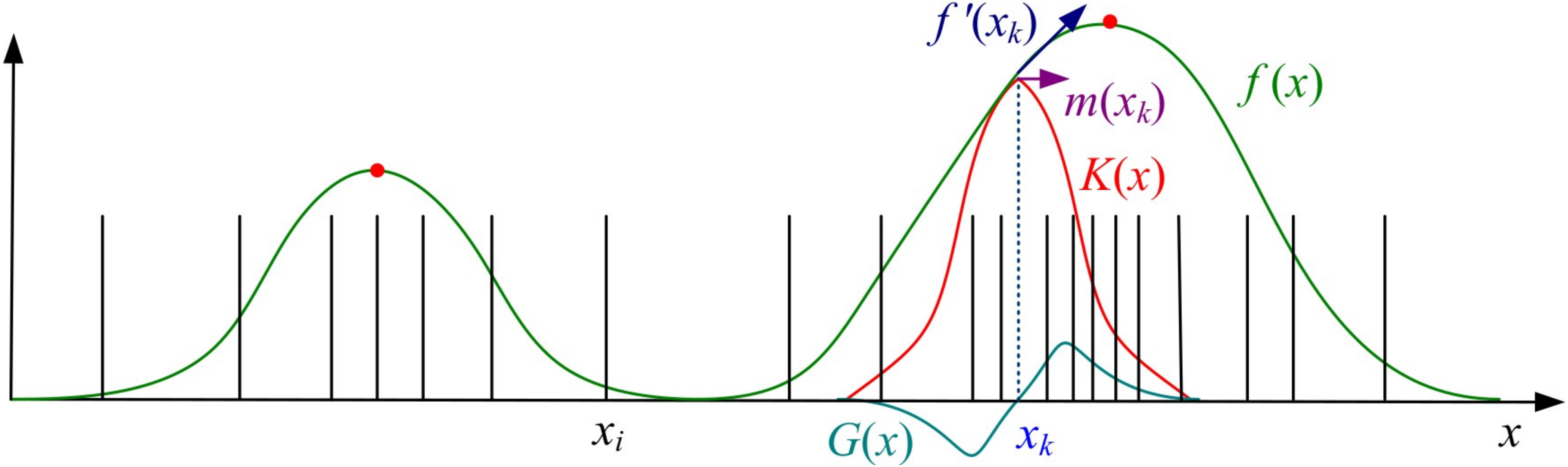
# Mean shift

UNIVERSITÀ DI PARMA

1) Initialize random seed and a search window W around it
2) Compute center of gravity (the "mean") of W using a kernel density estimation function K()
   - Typically a Gaussian
3) Shift W to the mean
4) Repeat (2) until convergence

# Mean shift

- Mean shift (m(x)) of “sampled” data f(x) using a K() window

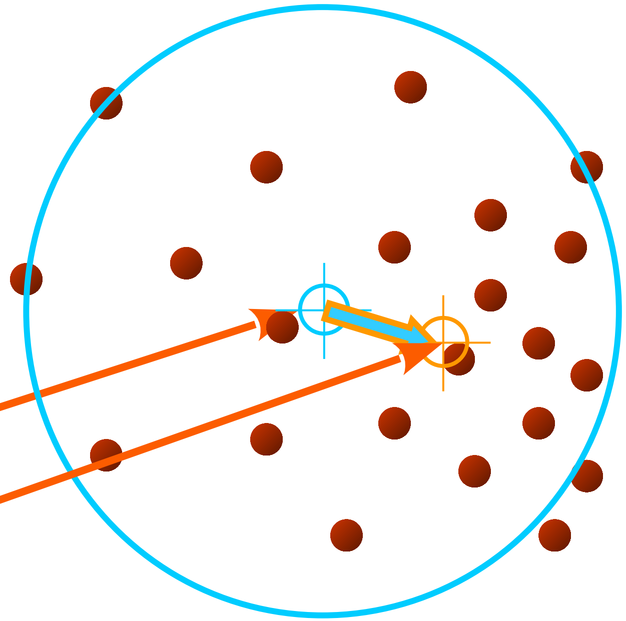


# Mean shift

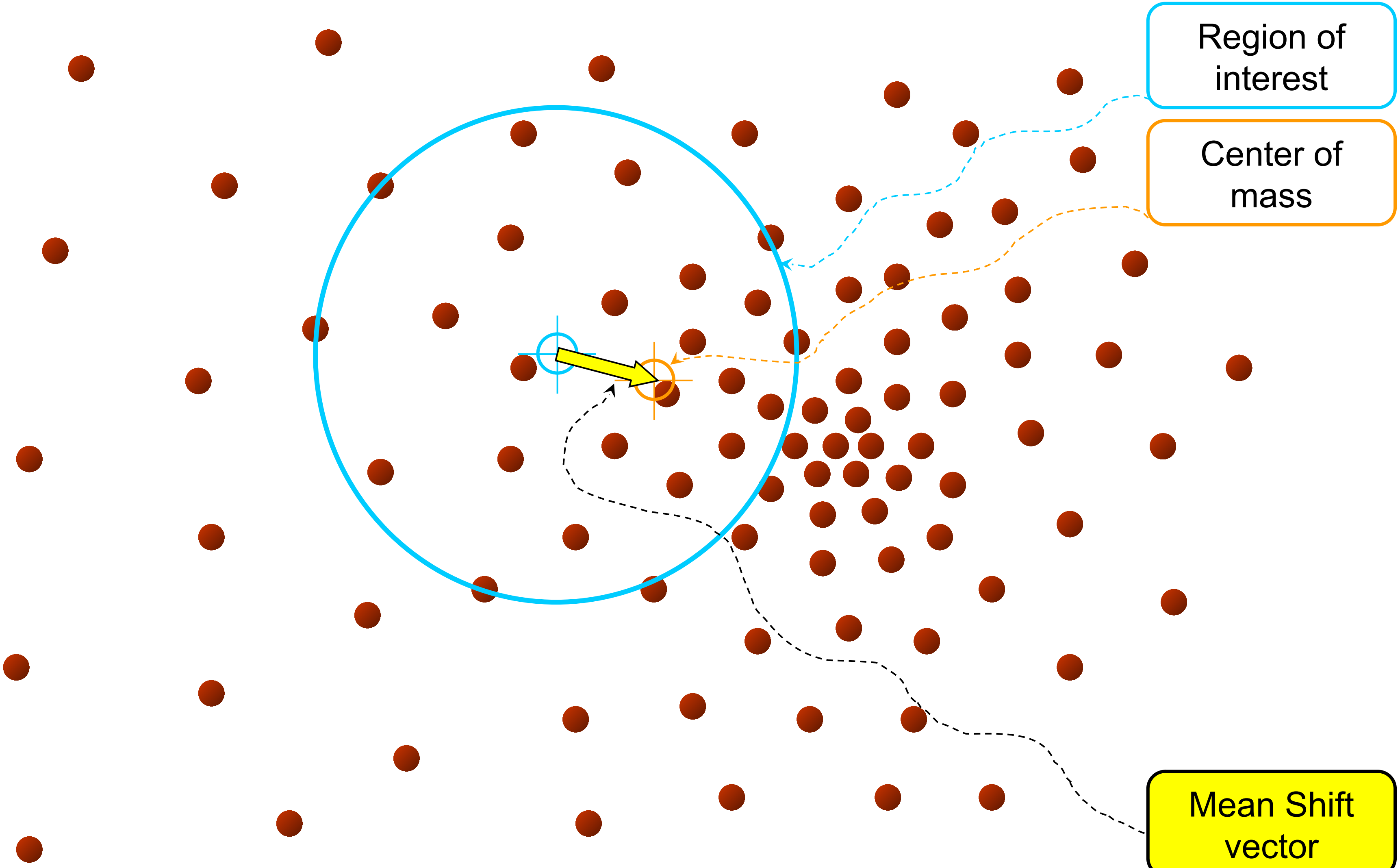
- Compute mean shift
  - Usually a Gaussian approach is used
- Translate W by m(x)

$$G(x) = -K'(x)$$

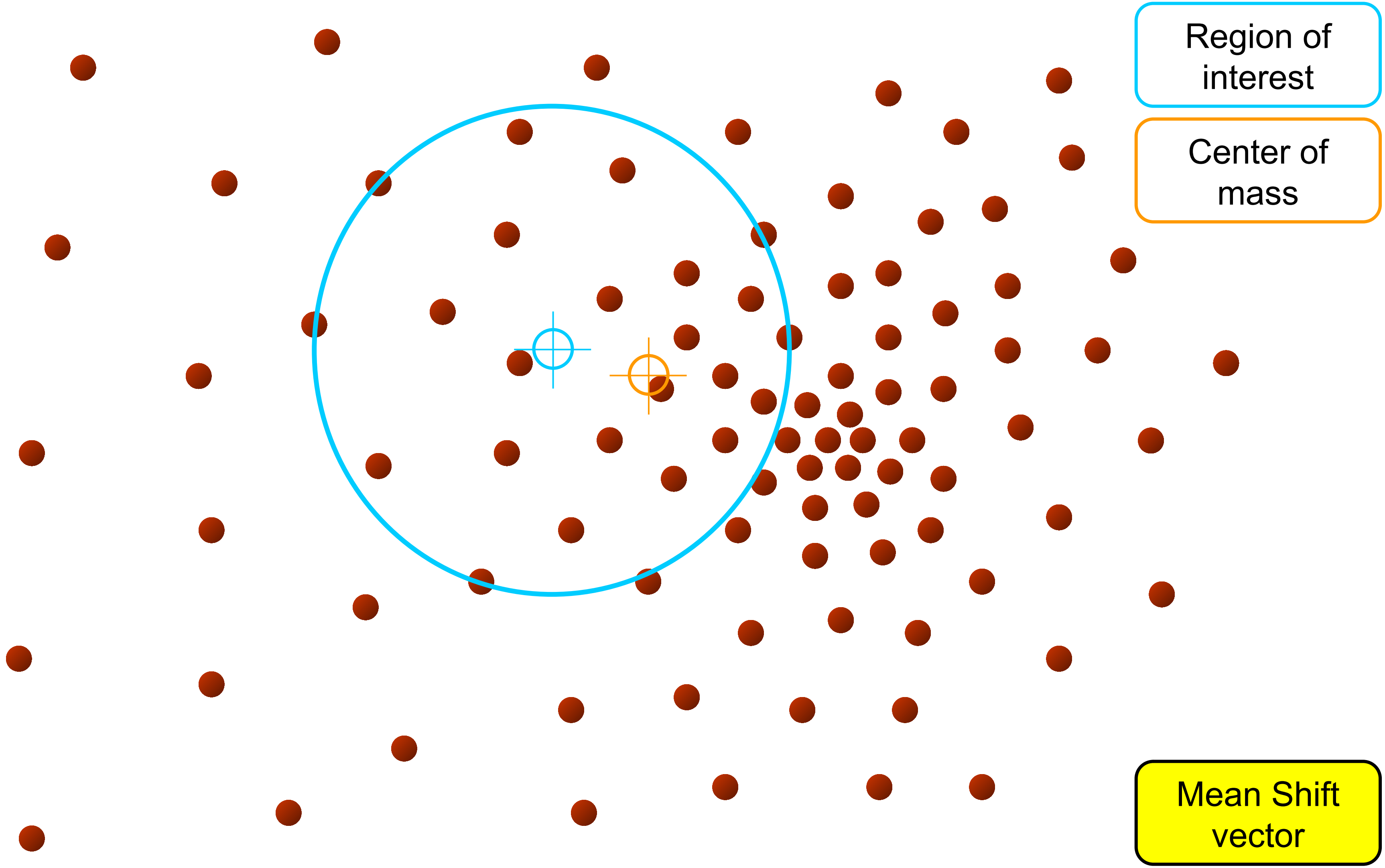
$$m(x) = \left[ \boxed{\frac{\sum_{i=1}^{n} x_i G\left(\frac{\|x - x_i\|^2}{h}\right)}{\sum_{i=1}^{n} G\left(\frac{\|x - x_i\|^2}{h}\right)}} - \boxed{x} \right]$$



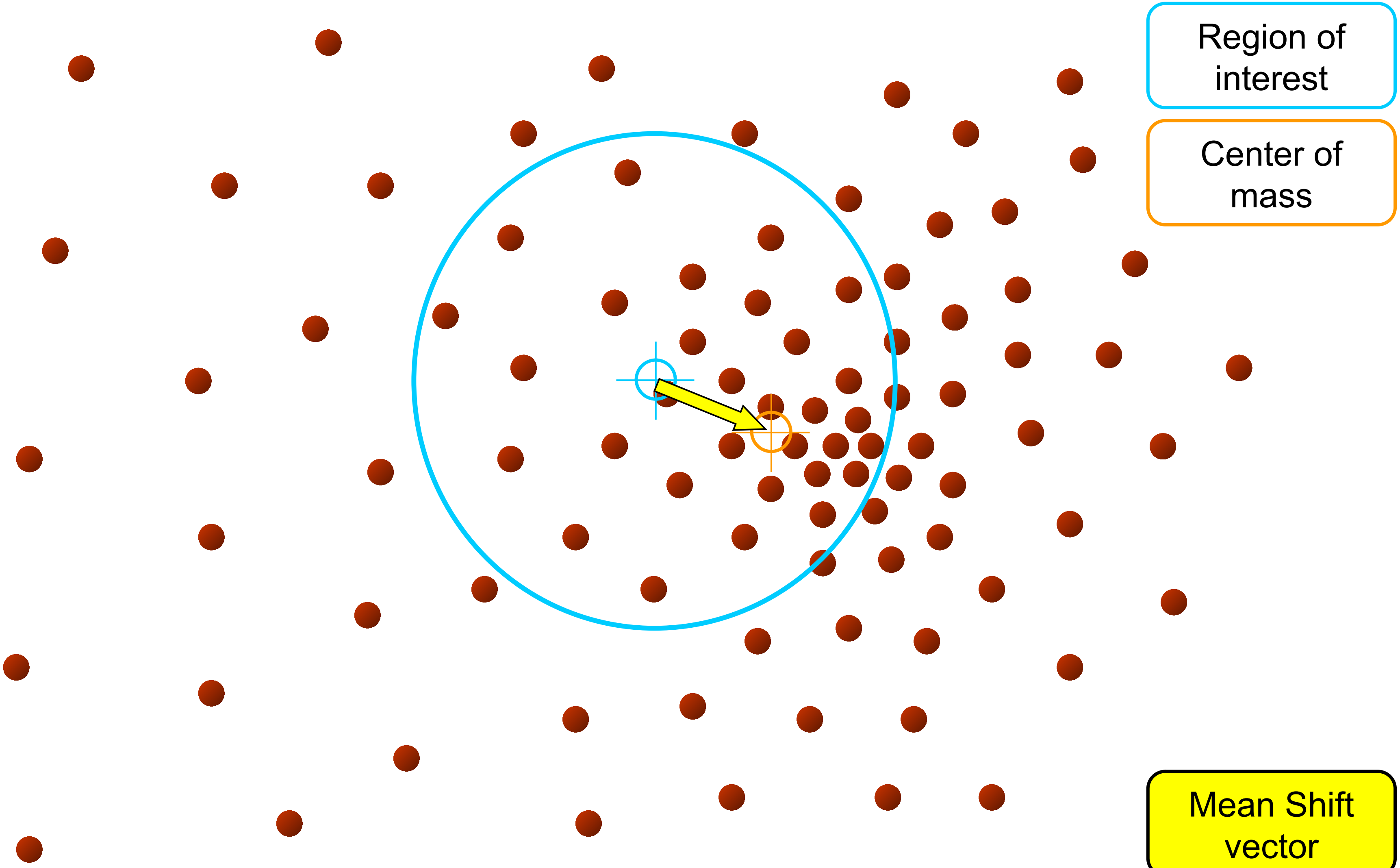
h → bandwidth (~size of W)

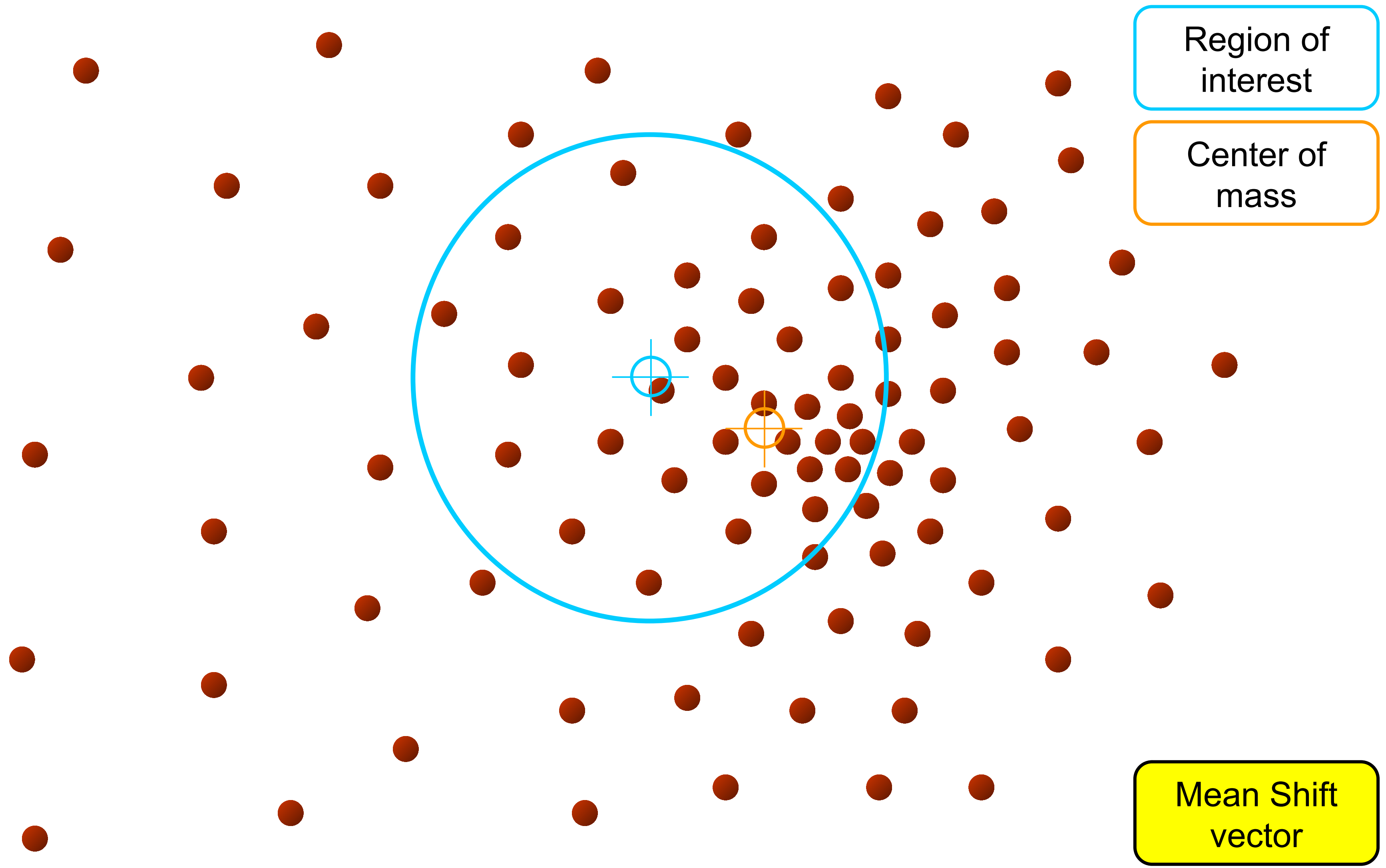



Slide by Y. Ukrainitz & B. Sarel

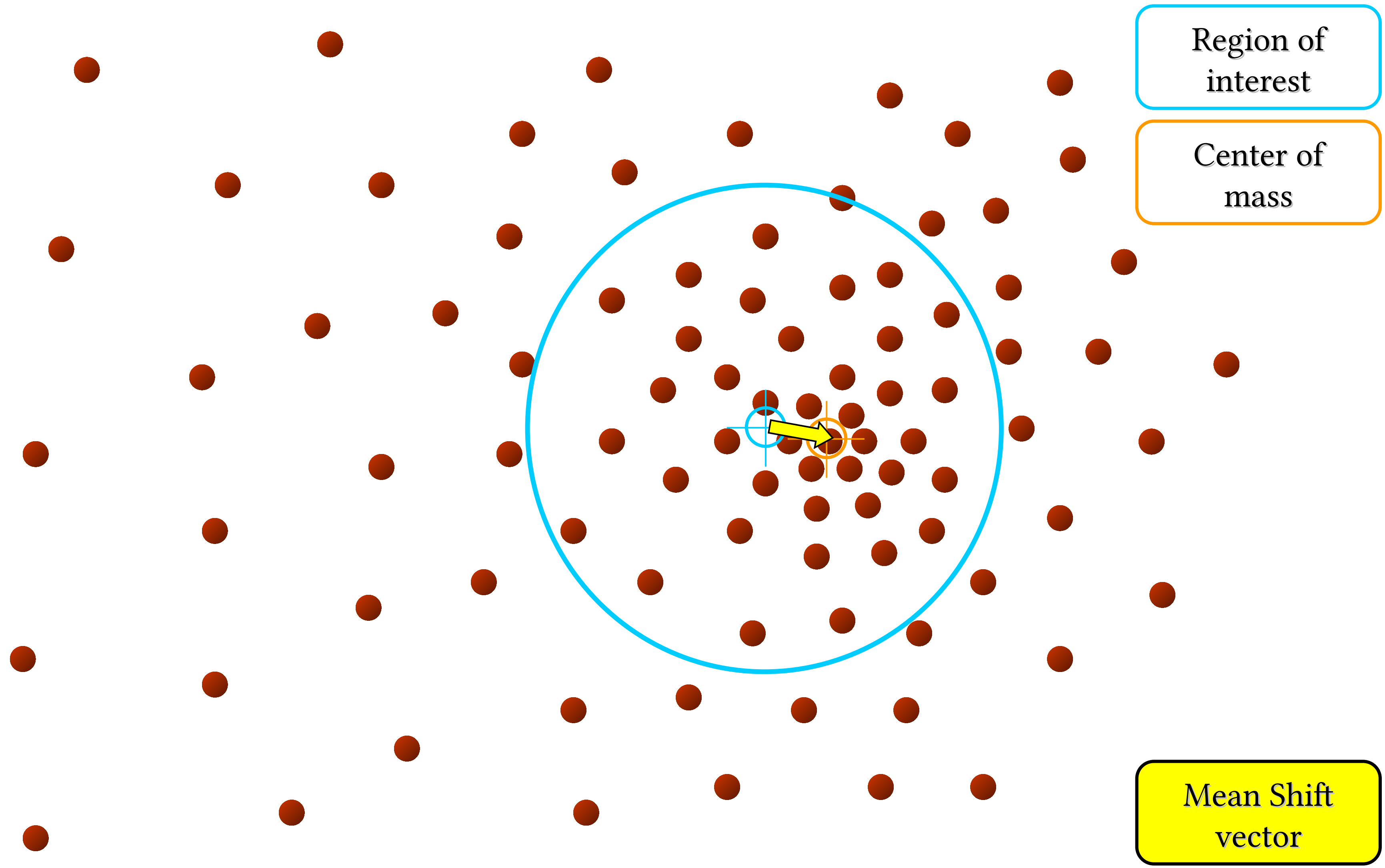



Slide by Y. Ukrainitz & B. Sarel

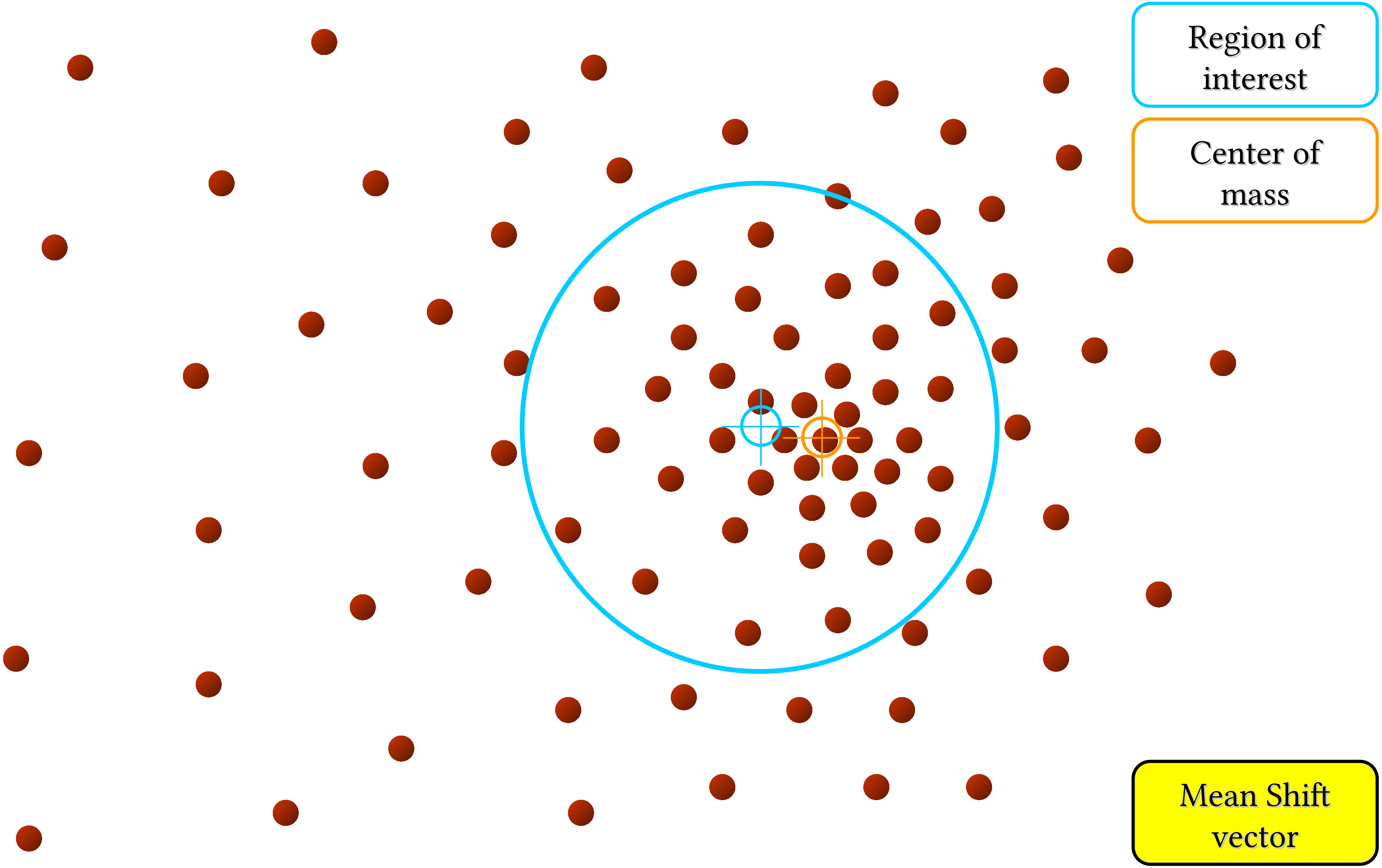

Region of interest
Center of mass
Mean Shift vector



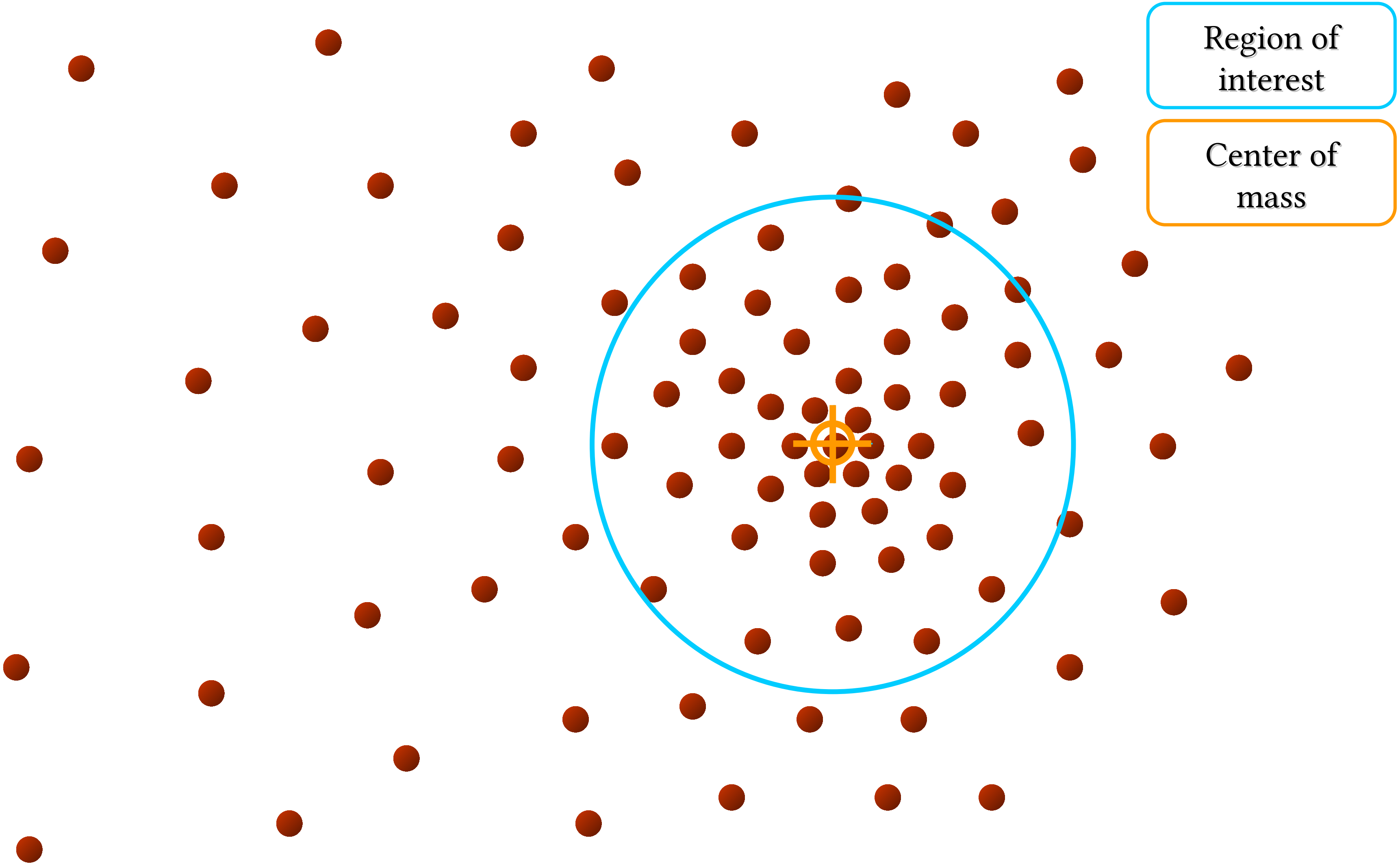

Slide by Y. Ukrainitz & B. Sarel



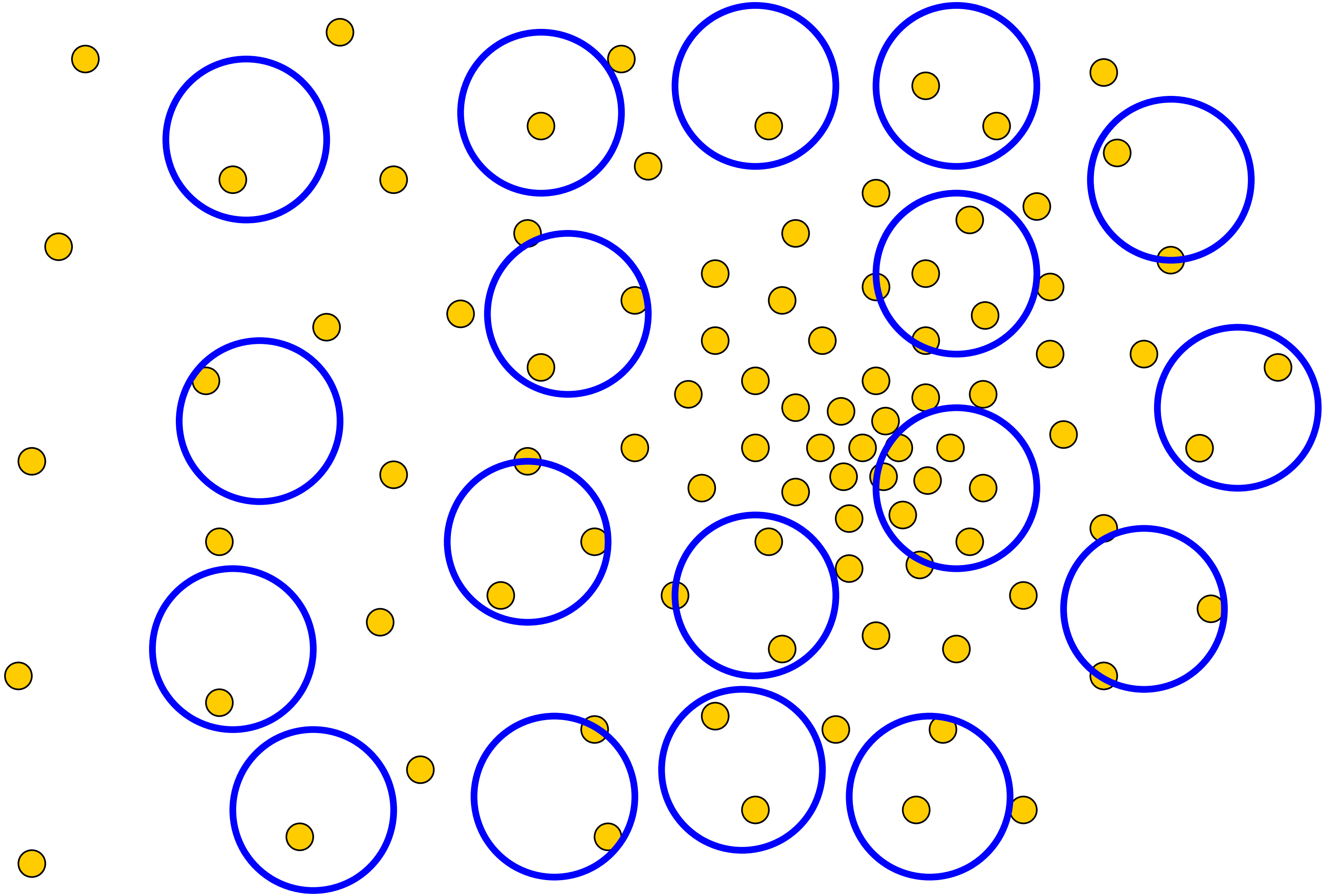

Slide by Y. Ukrainitz & B. Sarel




Slide by Y. Ukrainitz & B. Sarel



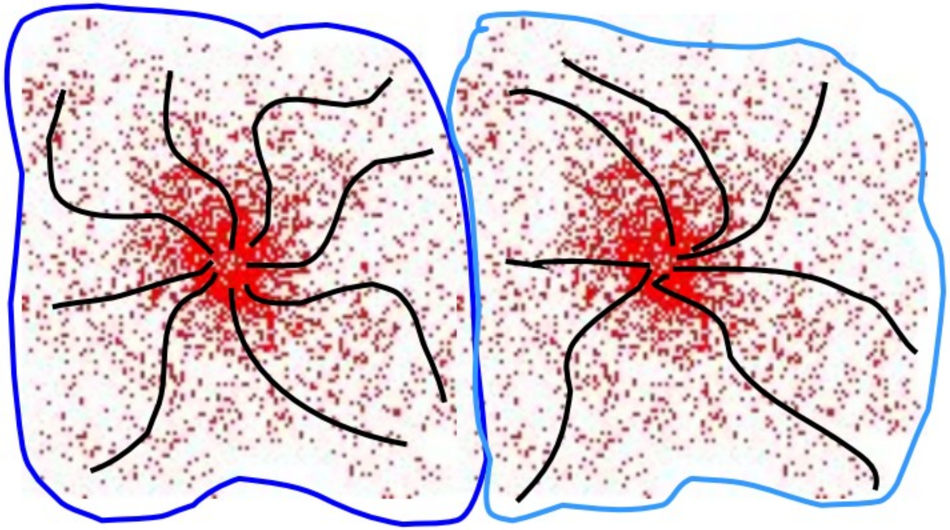

Slide by Y. Ukrainitz & B. Sarel



# Mean Shift



- Attraction basins → region where all trajectories lead to the same node
- Cluster → all points in the attraction basin of each node



# Mean Shift examples



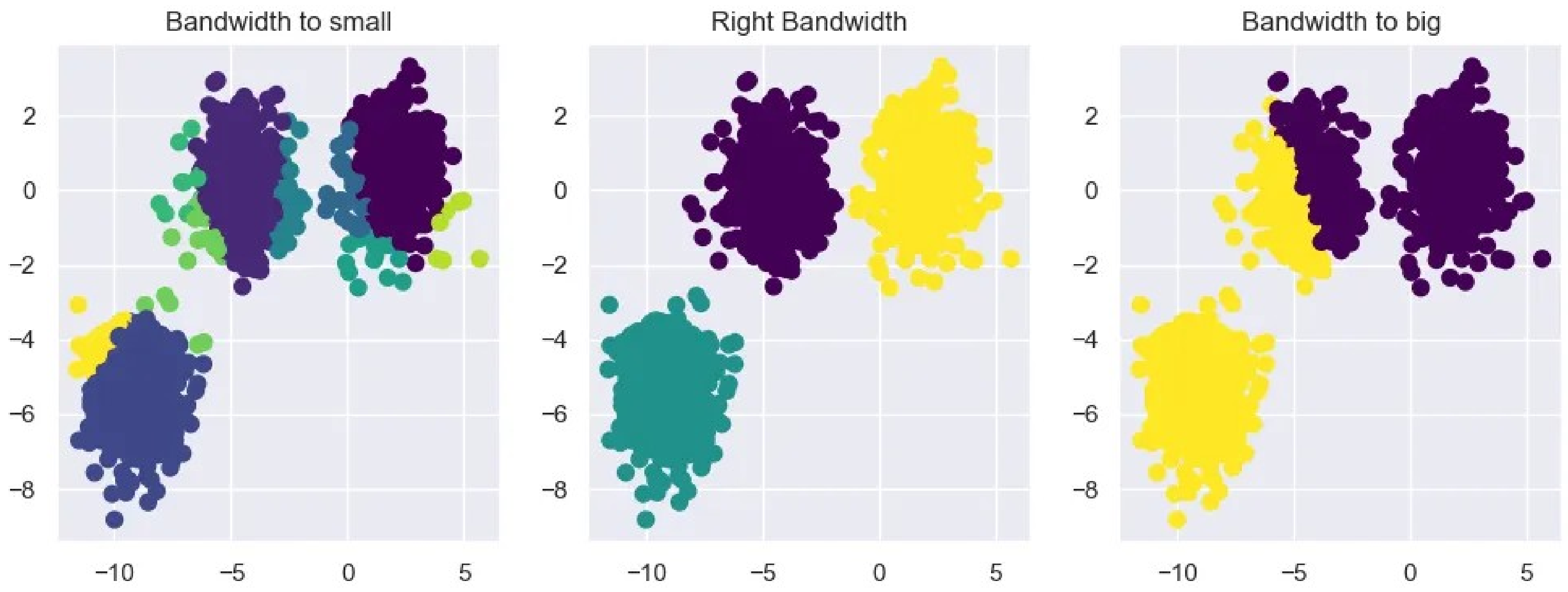



# Mean Shift: summary



- Pros
  - Good results
  - Flexible
  - Robust to outliers
- Cons
  - We have to choose kernel size
  - Not suitable for high dimensional features
  - Slow

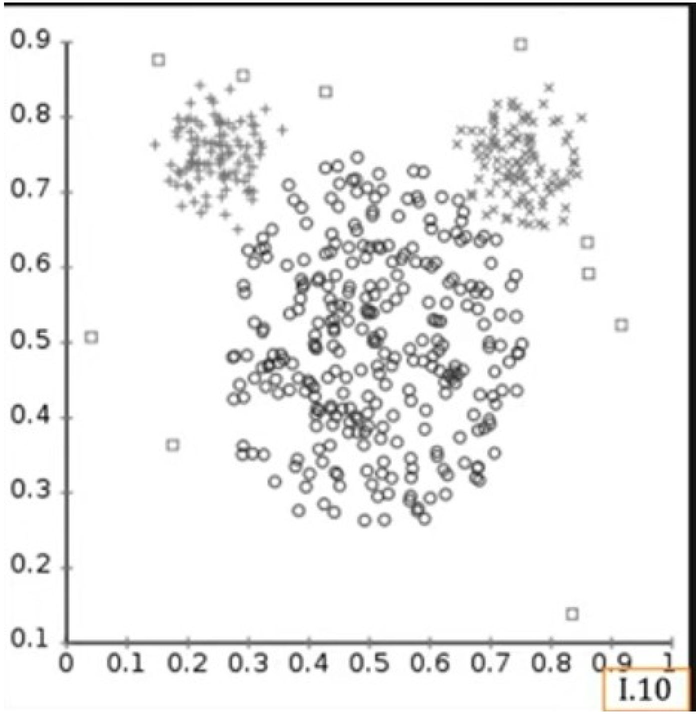
# Mean shift: bandwidth selection



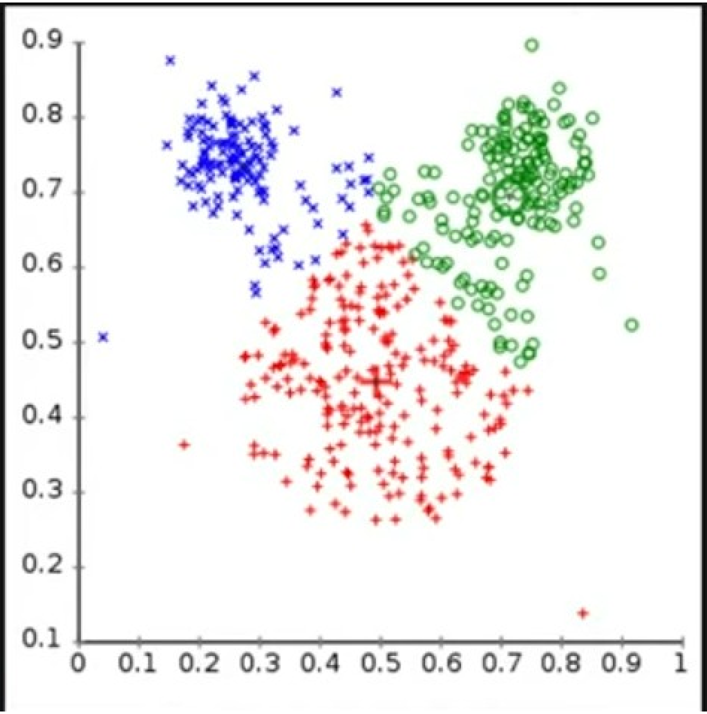
UNIVERSITÀ DI PARMA

# K-means vs mean shift

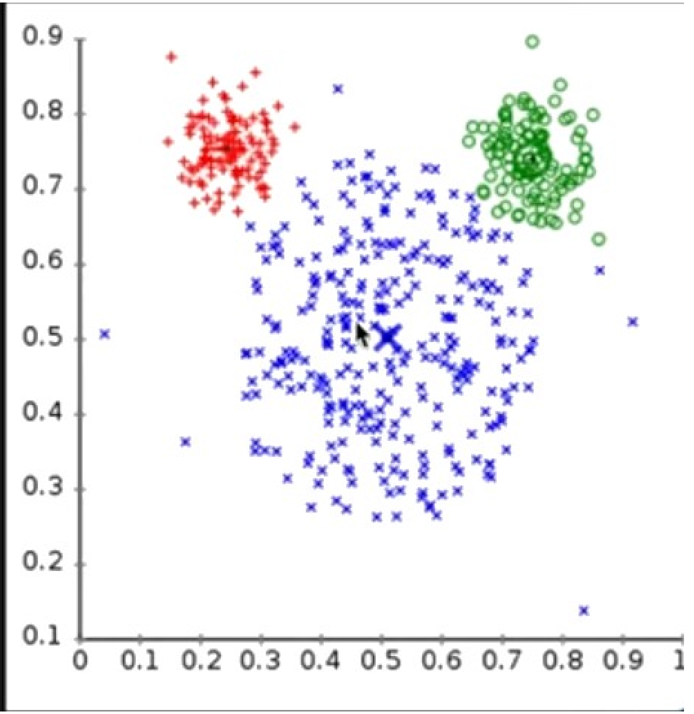
# K-means vs mean shift




Original Data




K-means (k=3)




Mean shift

# K-means vs mean shift

Original Data

K-means (k=16)

Mean shift

UNIVERSITÀ DI PARMA

# Other clustering approaches

# Other clustering approaches

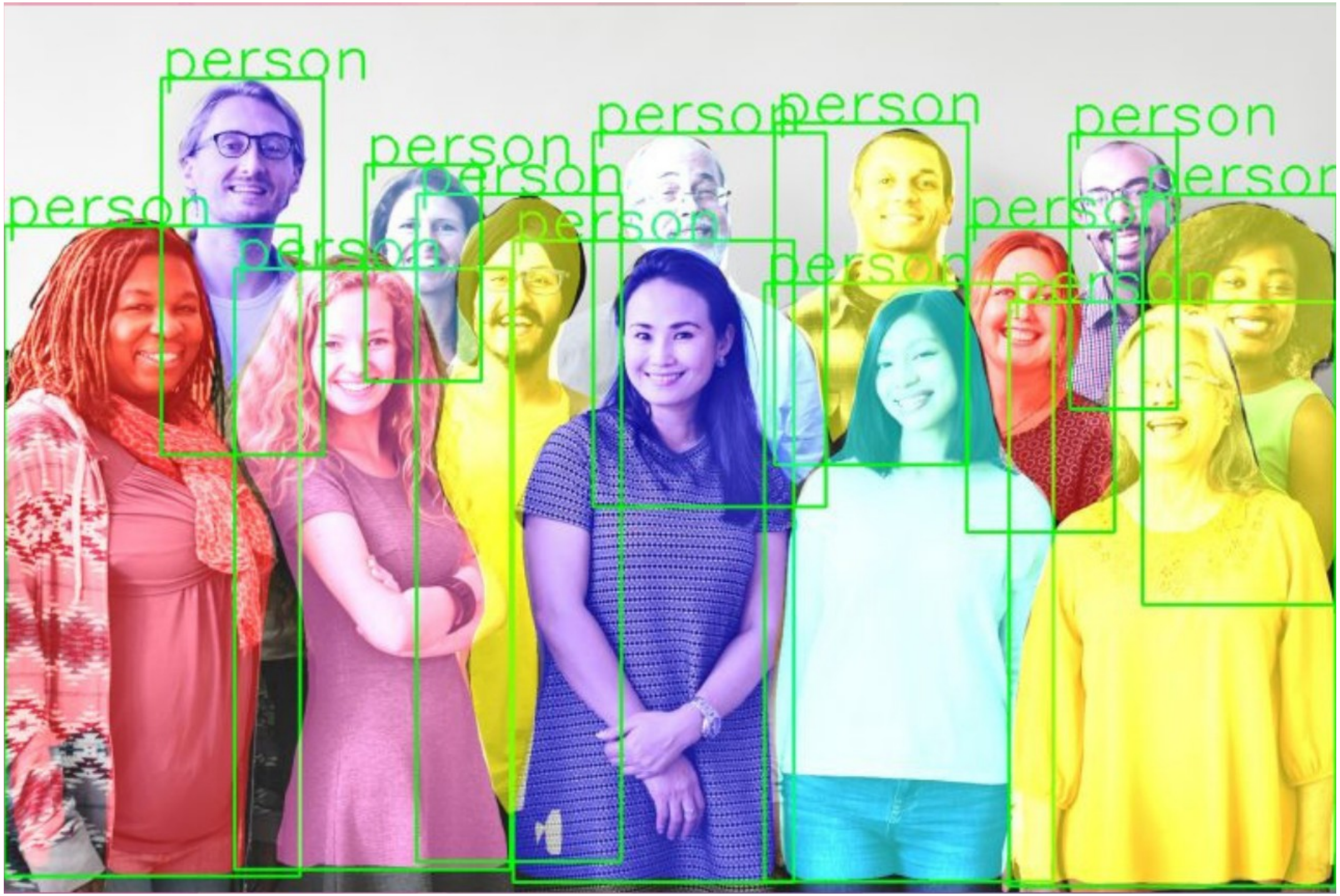


- Spectral clustering
- Agglomerative clustering
- Graph cut
- …
- All bottom-up approaches!

# Semantic segmentation (DL)

# Instance segmentation (DL)

UNIVERSITÀ DI PARMA



UNIVERSITÀ DI PARMA

# Image Segmentation

## Question time!