CAP5415-Computer Vision
Lecture 12-Image Segmentation (BASICS): Thresholding, Region Growing, Clustering

Dr. Ulas Bagci

bagci@ucf.edu

Image Segmentation

- Aim: to partition an image into a collection of set of pixels
  - Meaningful regions (coherent objects)
  - Linear structures (line, curve, …)
  - Shapes (circles, ellipses, …)

Image Segmentation

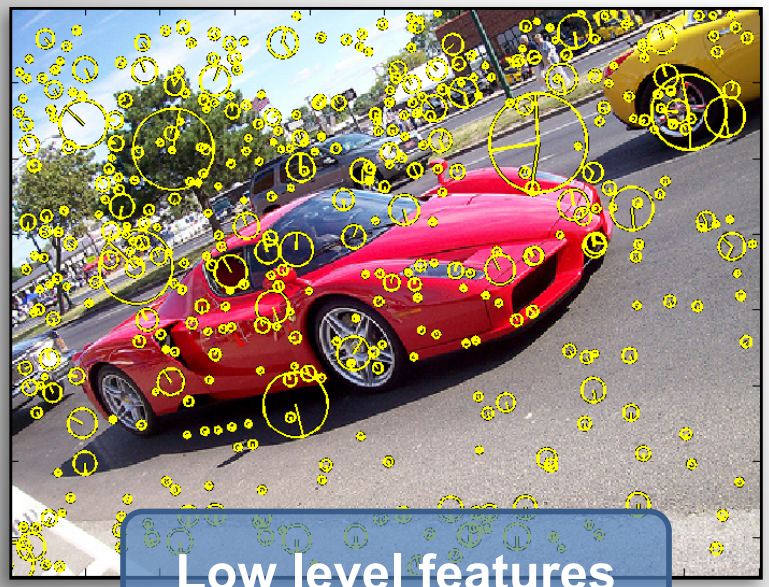
- Aim: to partition an image into a collection of set of pixels
  - Meaningful regions (coherent objects)
  - Linear structures (line, curve, …)
  - Shapes (circles, ellipses, …)

- Content based image retrieval
- Machine Vision
- Medical Imaging applications (tumor delineation,..)
- Object detection (face detection,…)
- 3D Reconstruction
- Object/Motion Tracking
- Object-based measurements such as size and shape
- Object recognition (face recognition,…)
- Fingerprint recognition,
- Video surveillance

…

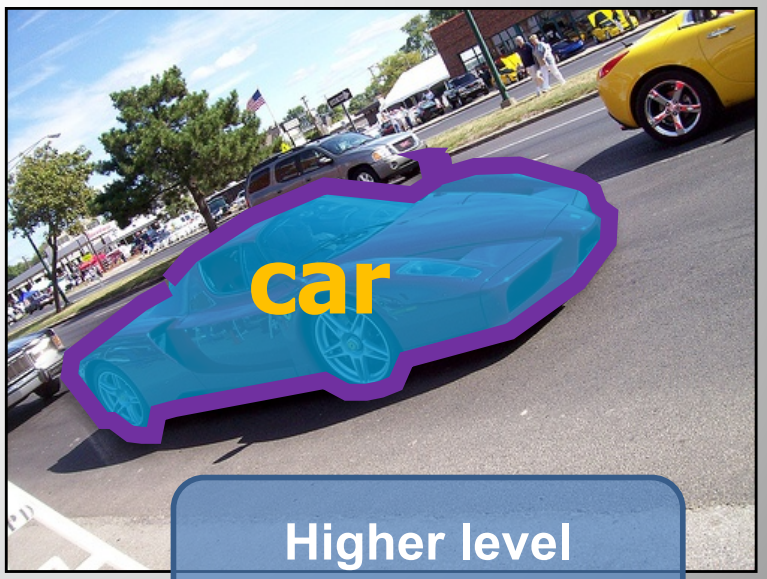
Image Segmentation

- In computer vision, image segmentation is one of the oldest and most widely studied problems
  - *Early techniques -> region splitting or merging*
  - *More recent techniques -> Energy minimization, hybrid methods, and deep learning*

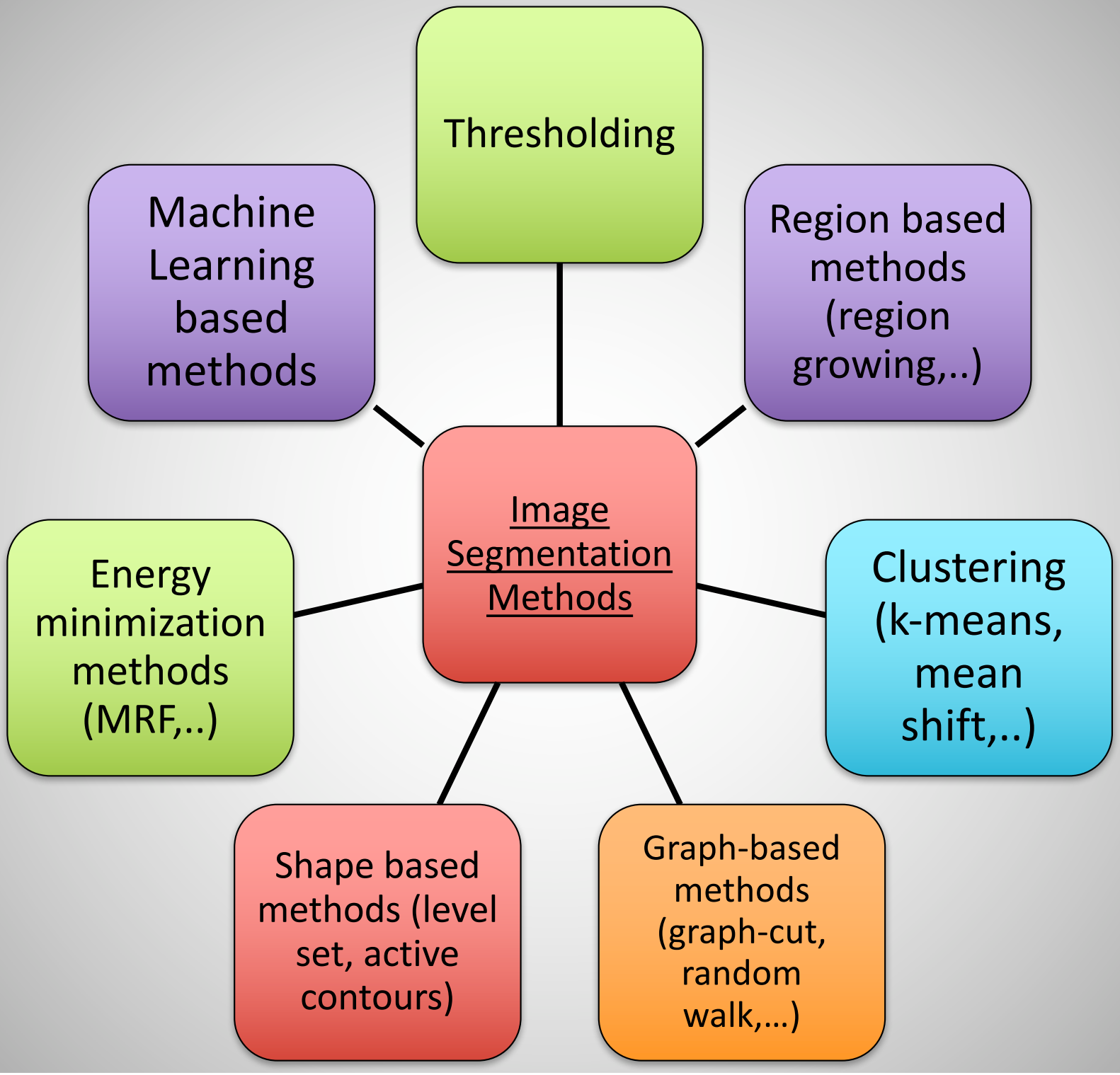


Low level features

Inference

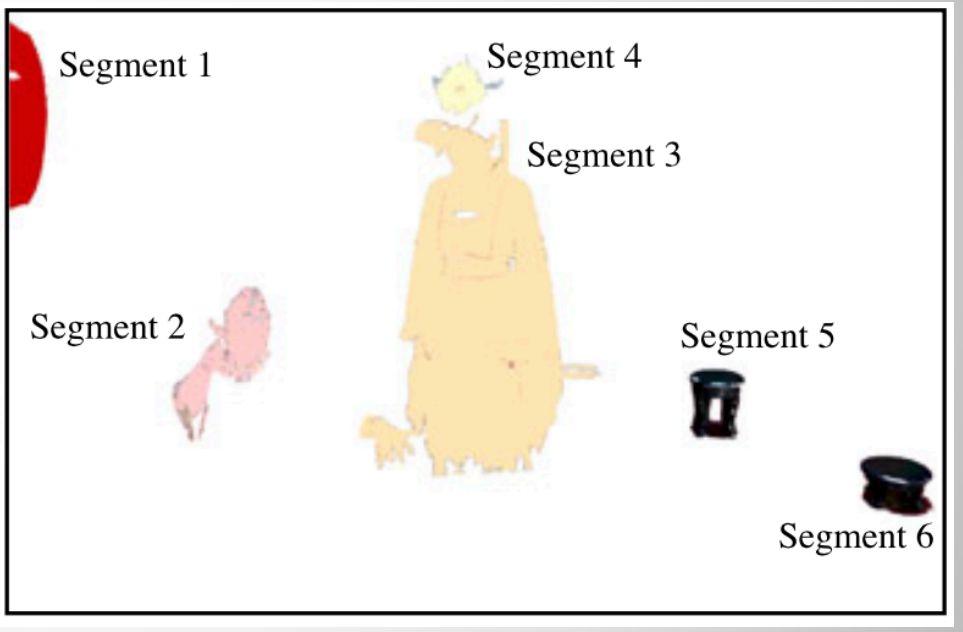


Higher level inference


Thresholding
Machine Learning based methods
Region based methods (region growing,..)
Image Segmentation Methods
Energy minimization methods (MRF,..)
Clustering (k-means, mean shift,..)
Shape based methods (level set, active contours)
Graph-based methods (graph-cut, random walk,…)

Basics of Image Segmentation

- Definition: *Image segmentation* partitions an image into regions called segments.



Basics of Image Segmentation

- Definition: *Image segmentation* partitions an image into regions called segments.

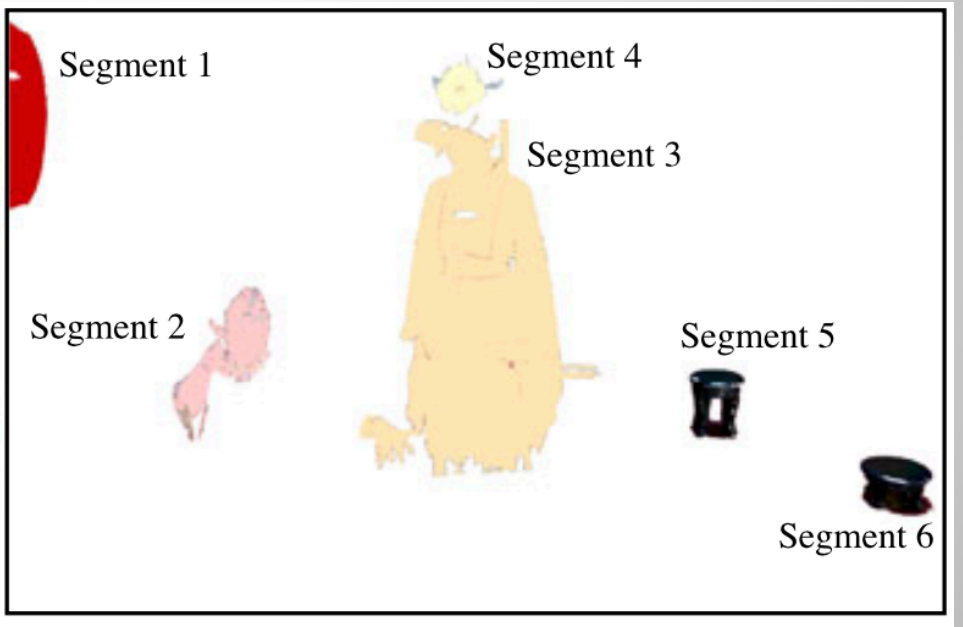


Image segmentation creates segments of connected pixels by analyzing some similarity criteria:

intensity, color, texture, histogram, features, …

Image Binarization

- Image binarization applies often just one global threshold T for mapping a scalar image I into a binary image

Image Binarization

- Image binarization applies often just one global threshold T for mapping a scalar image I into a binary image

$$J(x,y) = \begin{cases} 0 & \text{if } I(x,y) < T \\ 1 & \text{otherwise.} \end{cases}$$

Image Binarization

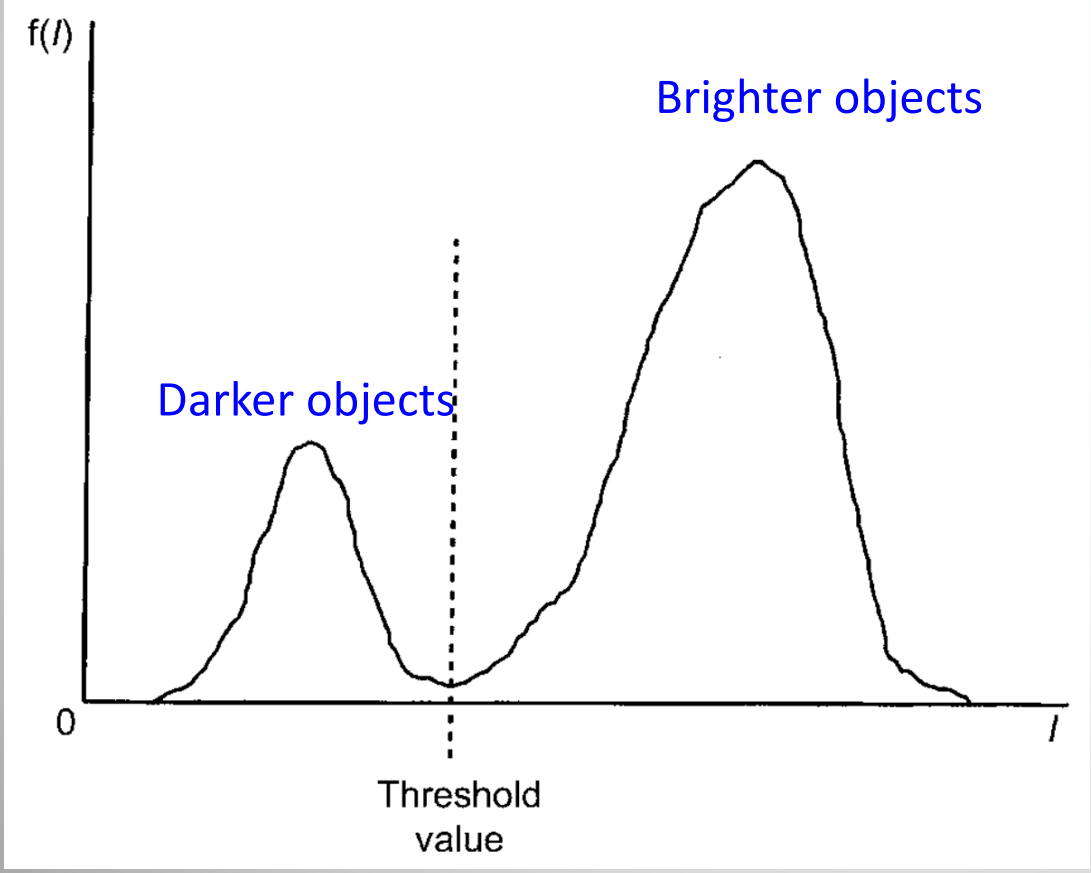
- Image binarization applies often just one global threshold T for mapping a scalar image I into a binary image

$$J(x,y) = \begin{cases} 0 & \text{if } I(x,y) < T \\ 1 & \text{otherwise.} \end{cases}$$

- The global threshold can be identified by an optimization strategy aiming at creating “large” connected regions and at reducing the number of small-sized regions, called *artifacts*.

Image Binarization

- Thresholding: Most frequently employed method for determining threshold is based on histogram analysis of intensity levels.



Peak on the left of the histogram corresponds to dark objects

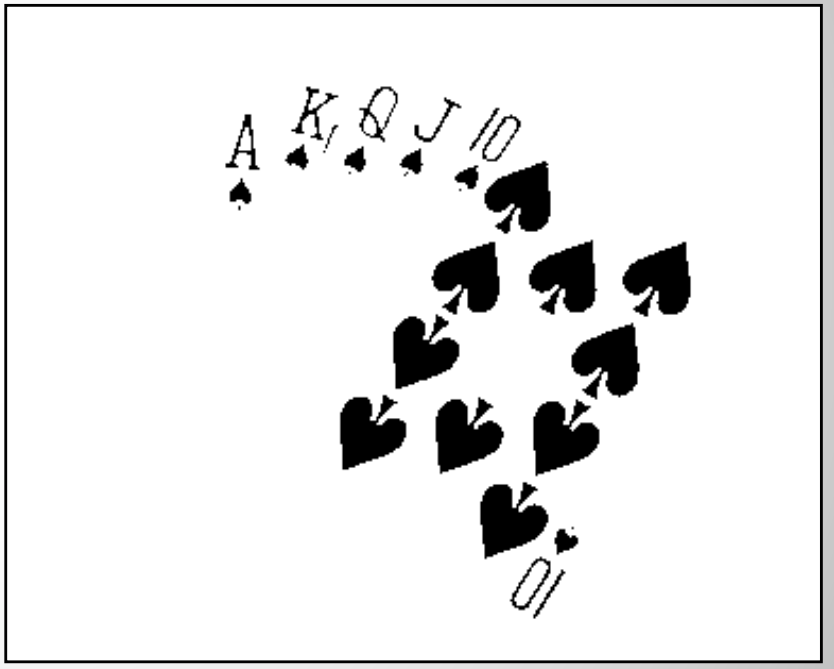
Peak on the right of the histogram corresponds to brighter objects

DIFFICULTIES

1. The valley may be so broad that it is difficult to locate a significant minimum
2. Number of minima due to type of details in the image
3. Noise
4. No visible valley
5. Histogram may be multi-modal

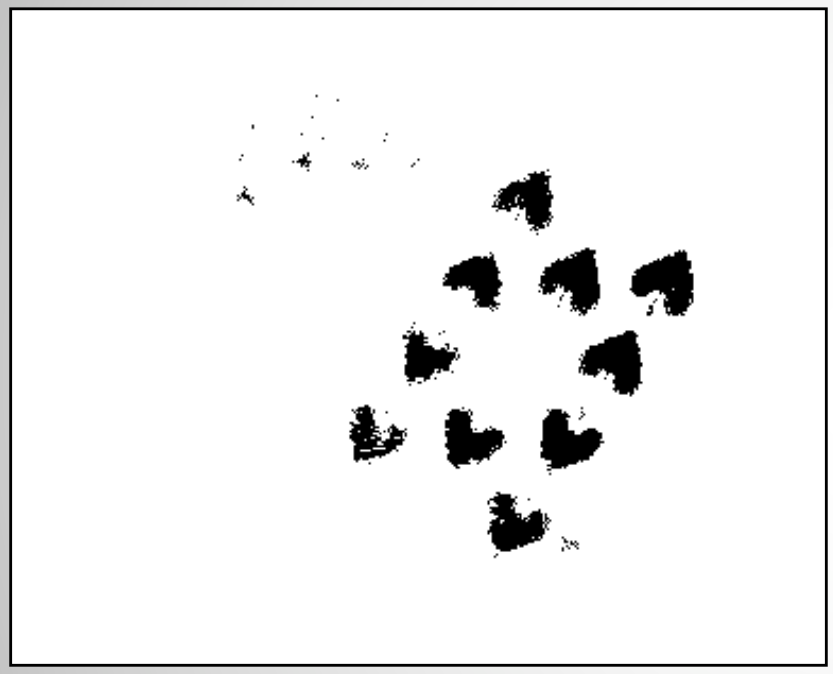
Thresholding Example

Original Image

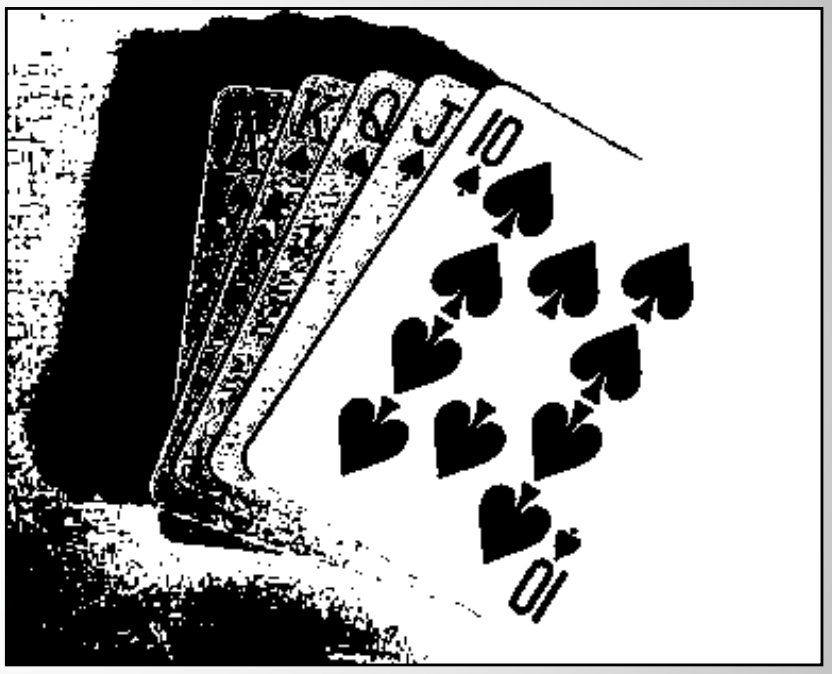


Thresholded Image

Thresholding Example 2

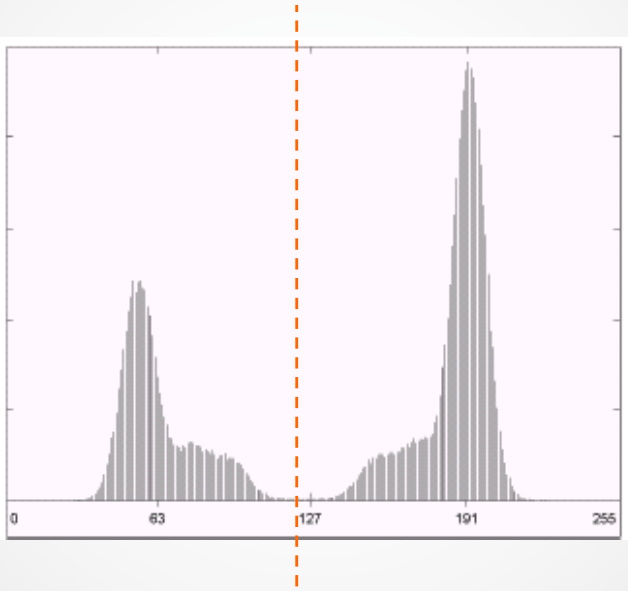


Threshold Too Low

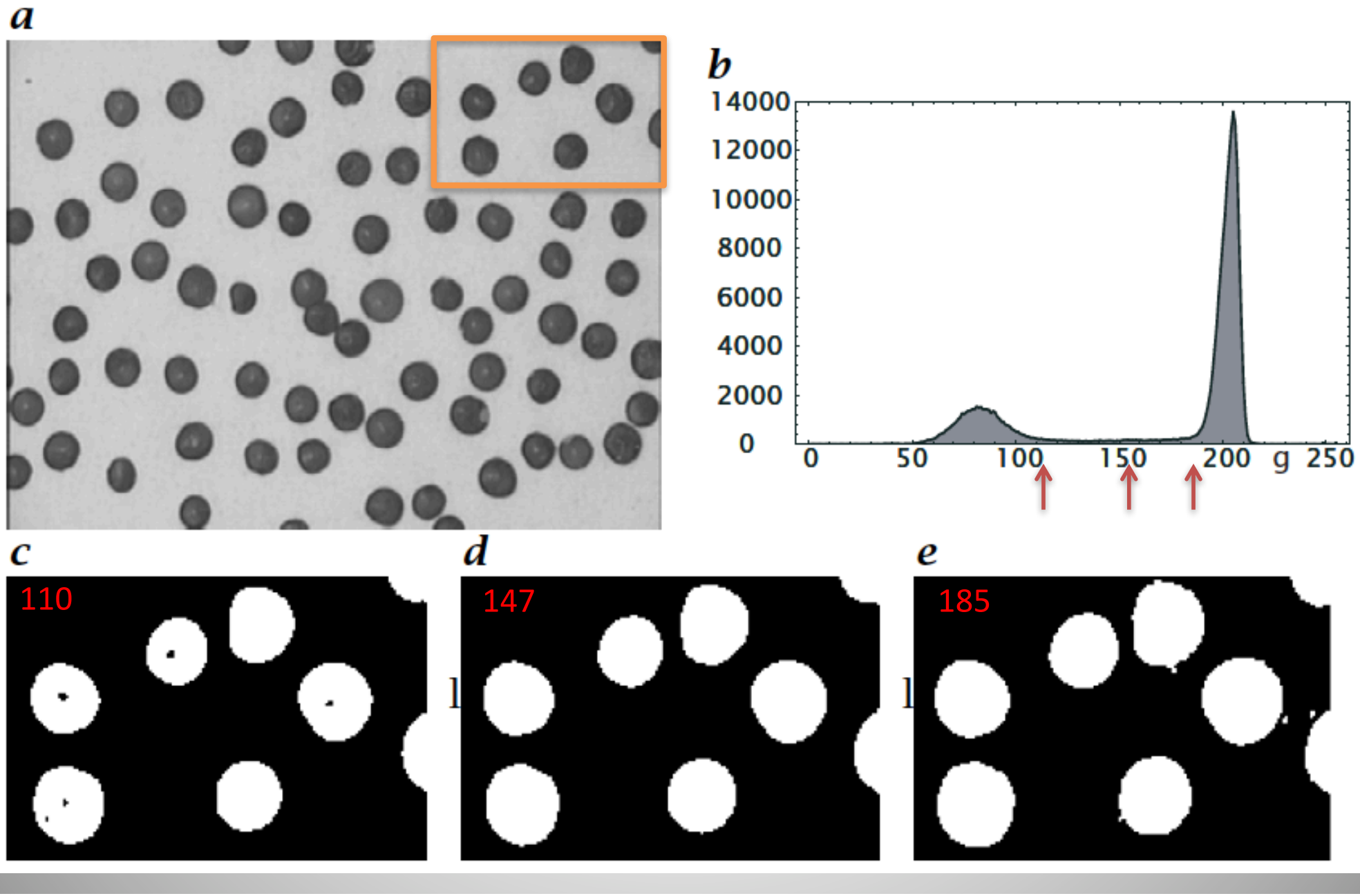


Threshold Too High

Thresholding Example 3



Thresholding Example-4



Otsu Thresholding

- **Definition:** The method uses the grey-value histogram of the given image I as input and aims at providing the best threshold in the sense that the “overlap” between two classes, set of object and background pixels, is minimized (i.e., by finding the best balance).

Otsu Thresholding

- **Definition:** The method uses the grey-value histogram of the given image I as input and aims at providing the best threshold in the sense that the "overlap" between two classes, set of object and background pixels, is minimized (i.e., by finding the best balance).

- Otsu's algorithm selects a threshold that maximizes the between-class variance σ_b^2. In the case of two classes,

$$\sigma_b^2 = P_1(\mu_1 - \mu)^2 + P_2(\mu_2 - \mu)^2 = P_1 P_2(\mu_1 - \mu_2)^2$$

where P_1 and P_2 denote class probabilities, and μ_i the means of object and background classes.

Otsu Thresholding

- **Definition:** The method uses the grey-value histogram of the given image I as input and aims at providing the best threshold in the sense that the "overlap" between two classes, set of object and background pixels, is minimized (i.e., by finding the best balance).

- Otsu's algorithm selects a threshold that maximizes the between-class variance σ_b^2. In the case of two classes,

$$\sigma_b^2 = P_1(\mu_1 - \mu)^2 + P_2(\mu_2 - \mu)^2 = P_1 P_2(\mu_1 - \mu_2)^2$$

where P_1 and P_2 denote class probabilities, and μ_i the means of object and background classes.

- Let c_I be the relative cumulative histogram of an image I, then P_1 and P_2 are approximated by $c_I(u)$ and $1 - c_I(u)$ respectively.
- u is assumed to be the chosen threshold.

Otsu Thresholding Algorithm

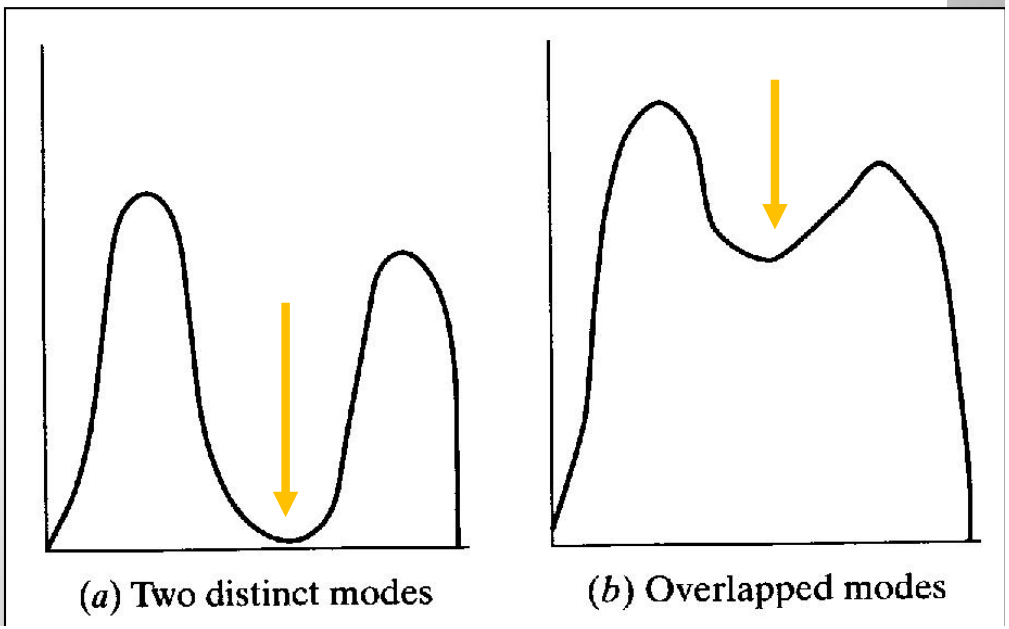
1: Compute histogram H_I for $u = 0, \ldots, G_{\max}$;
2: Let T_0 be the increment for potential thresholds; $u = T_0$; $T = u$; and $S_{\max} = 0$;
3: **while** $u < G_{\max}$ **do**
4: Compute $c_I(u)$ and $\mu_i(u)$ for $i = 1, 2$;
5: Compute $\sigma_b^2(u) = c_I(u)[1 - c_I(u)][\mu_i(u) - \mu_2(u)]^2$;
6: **if** $\sigma_b^2(u) > S_{\max}$ **then**
7: $S_{\max} = \sigma_b^2(u)$ and $T = u$;
8: **end if**
9: Set $u = u + T_0$
10: **end while**

$$P_1 = \sum_{i=0}^{u} p(i) \qquad \mu_1 = \sum_{i=0}^{u} ip(i)/P_1$$

$$P_2 = \sum_{i=u+1}^{G_{max}} p(i) \qquad \mu_2 = \sum_{i=u+1}^{G_{max}} ip(i)/P_2$$

probabilities

Class means



(a) Two distinct modes (b) Overlapped modes

Example: Otsu Thresholding

$T = 63$

$T = 92$

$T = 141$

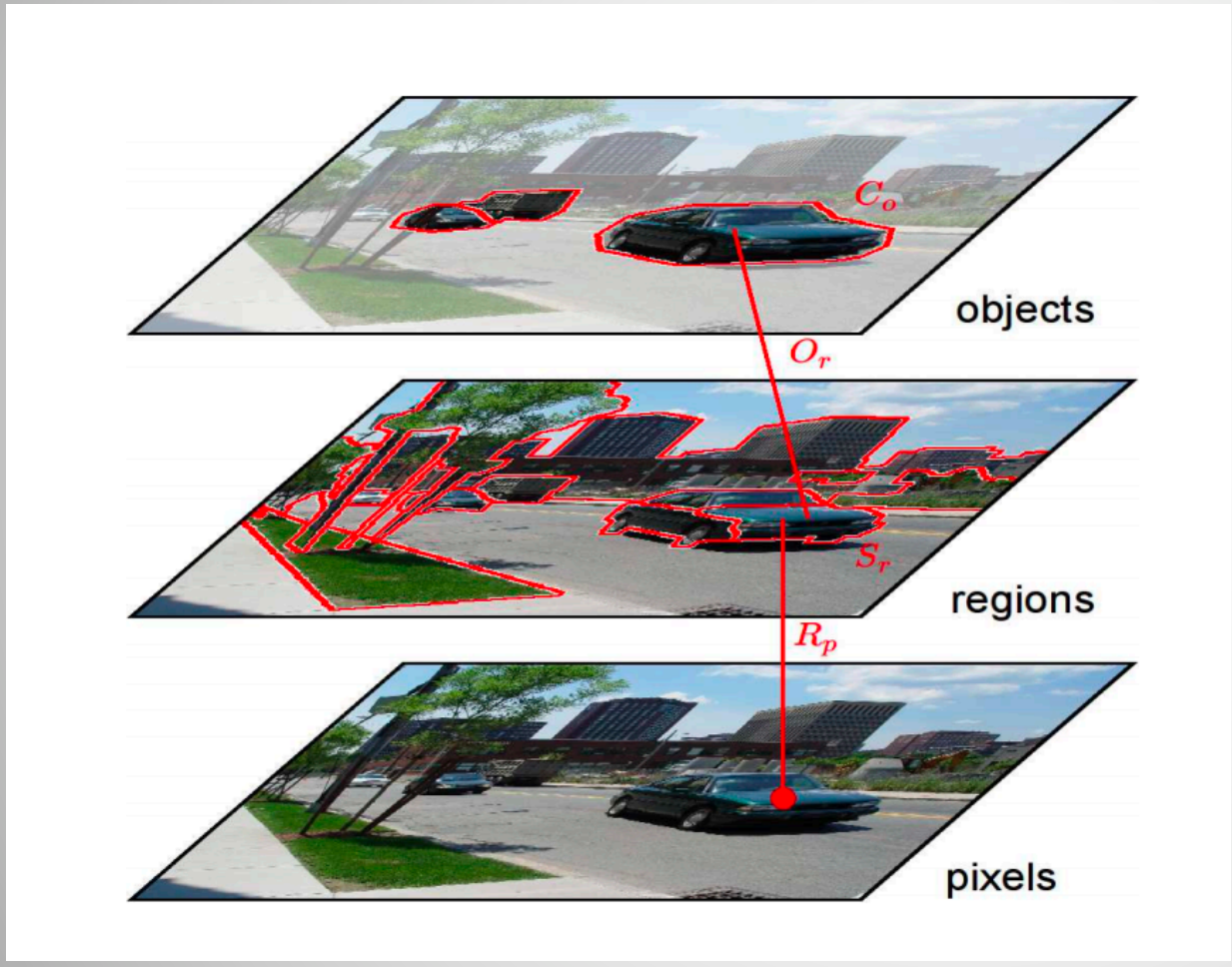
$T = 162$

$T = 187$

$T = 230$

Region Based Segmentation

Region Based Segmentation-Basics



Region:
A group of connected pixels with similar properties

Closed boundaries

Computation of regions is based on similarity

Regions may correspond to Objects in a scene or parts of objects

Spatial proximity + similarity

Region Growing

- For segment generation in grey-level or color images, we may start at one *seed pixel* (x,y,I(x,y)) and add recursively adjacent pixels that satisfy a “similarity criterion” with pixels contained in the so-far grown region around the *seed pixel.*

Region Growing

- For segment generation in grey-level or color images, we may start at one *seed pixel* (x,y,I(x,y)) and add recursively adjacent pixels that satisfy a "similarity criterion" with pixels contained in the so-far grown region around the *seed pixel.*

- Defining similarity criteria alone is not an effective basis for segmentation
- It is necessary to consider the adjacency spatial relationship between pixels

Region Growing

- For segment generation in grey-level or color images, we may start at one *seed pixel* (x,y,I(x,y)) and add recursively adjacent pixels that satisfy a "similarity criterion" with pixels contained in the so-far grown region around the *seed pixel.*

- Defining similarity criteria alone is not an effective basis for segmentation
- It is necessary to consider the adjacency spatial relationship between pixels

Algorithm

1. The absolute intensity difference between candidate pixel and the seed pixel must lie within a specified range
2. The absolute intensity difference between a candidate pixel and the running average intensity of the growing region must lie within a specified range;
3. The difference between the standard deviation in intensity over a specified local neighborhood of the candidate pixel and that over a local neighborhood of the candidate pixel must (or must not) exceed a certain threshold

Seeded Segmentation (Region Growing)

1. Choose the seed pixel

0	0	5	6	7
1	1	5	8	7
0	1	6	7	7
2	0	7	6	6
0	1	5	6	5

(a)

Seeded Segmentation (Region Growing)

1. Choose the seed pixel
2. Check the neighboring pixels and add them to the region if they are similar to the seed

0	0	5	6	7
1	1	5	8	7
0	1	6	7	7
2	0	7	6	6
0	1	5	6	5

(a)

a	a	b	b	b
a	a	b	b	b
a	a	b	b	b
a	a	b	b	b
a	a	b	b	b

(b)

Seeded Segmentation (Region Growing)

1. Choose the seed pixel
2. Check the neighboring pixels and add them to the region if they are similar to the seed
3. Repeat step 2 for each of the newly added pixels; stop if no more pixels can be added

0	0	5	6	7
1	1	5	8	7
0	1	6	7	7
2	0	7	6	6
0	1	5	6	5

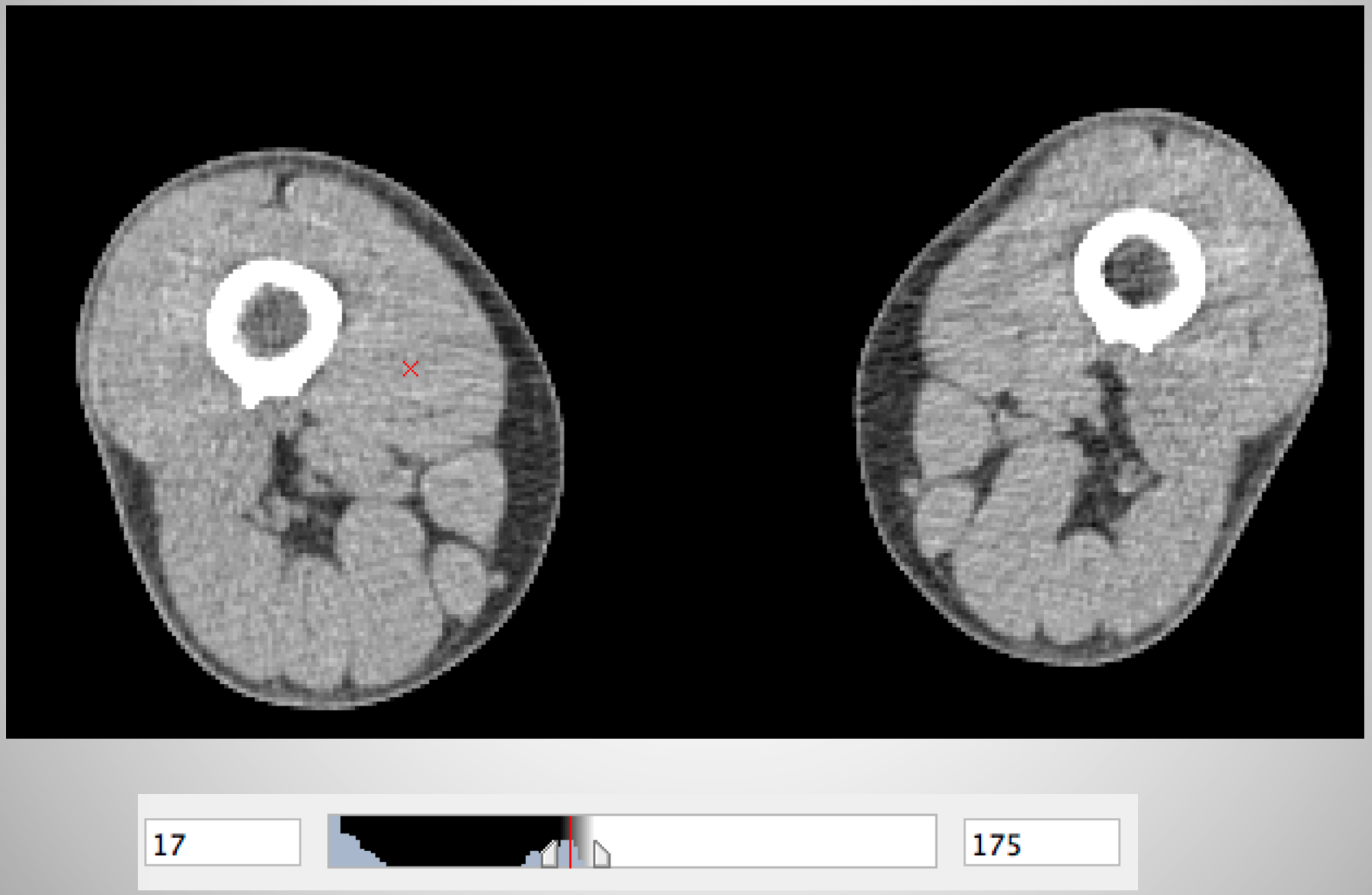
(a)

a	a	b	b	b
a	a	b	b	b
a	a	b	b	b
a	a	b	b	b
a	a	b	b	b

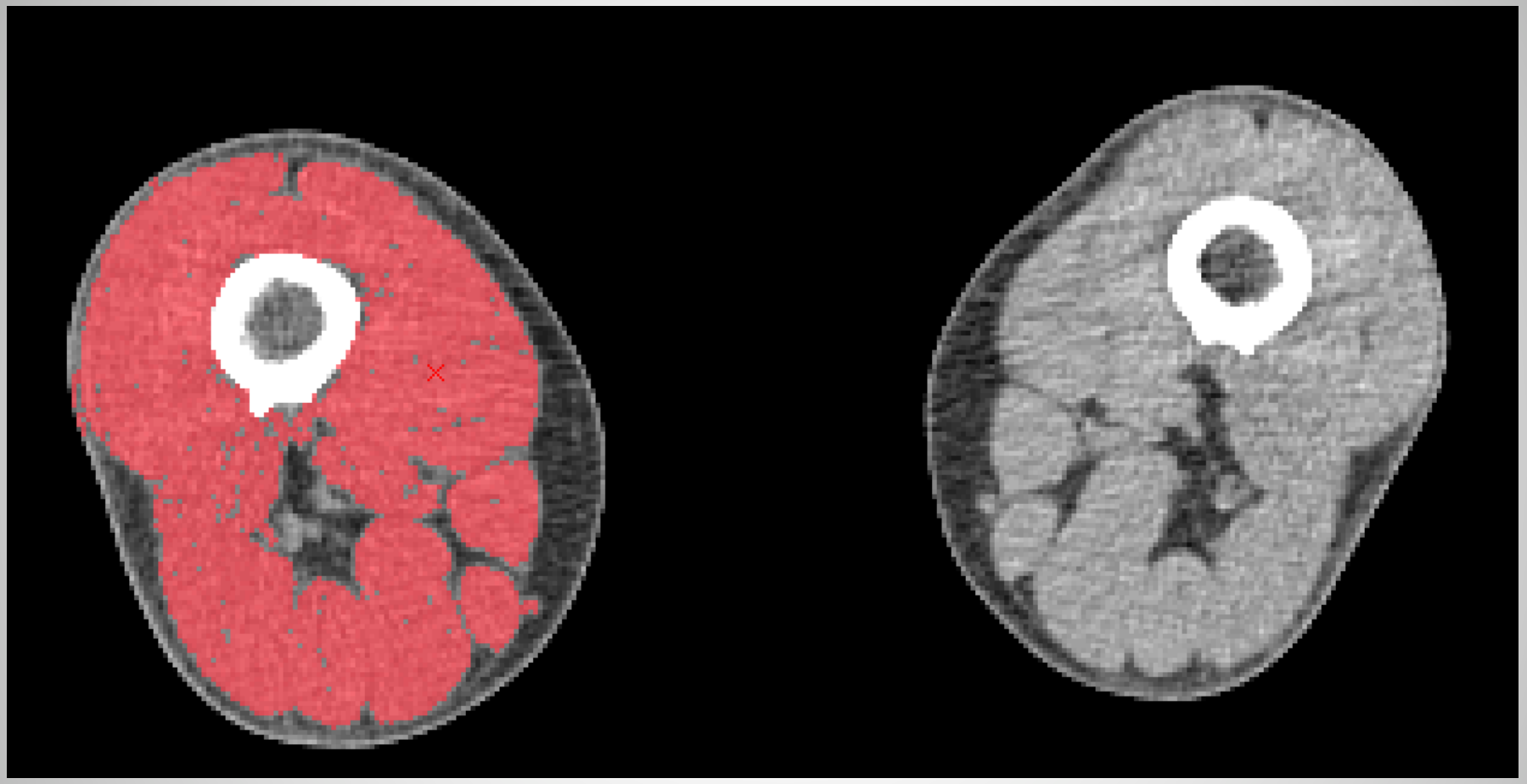
(b)

$$|\text{neighboring pixels} - \text{seed}| < Threshold$$

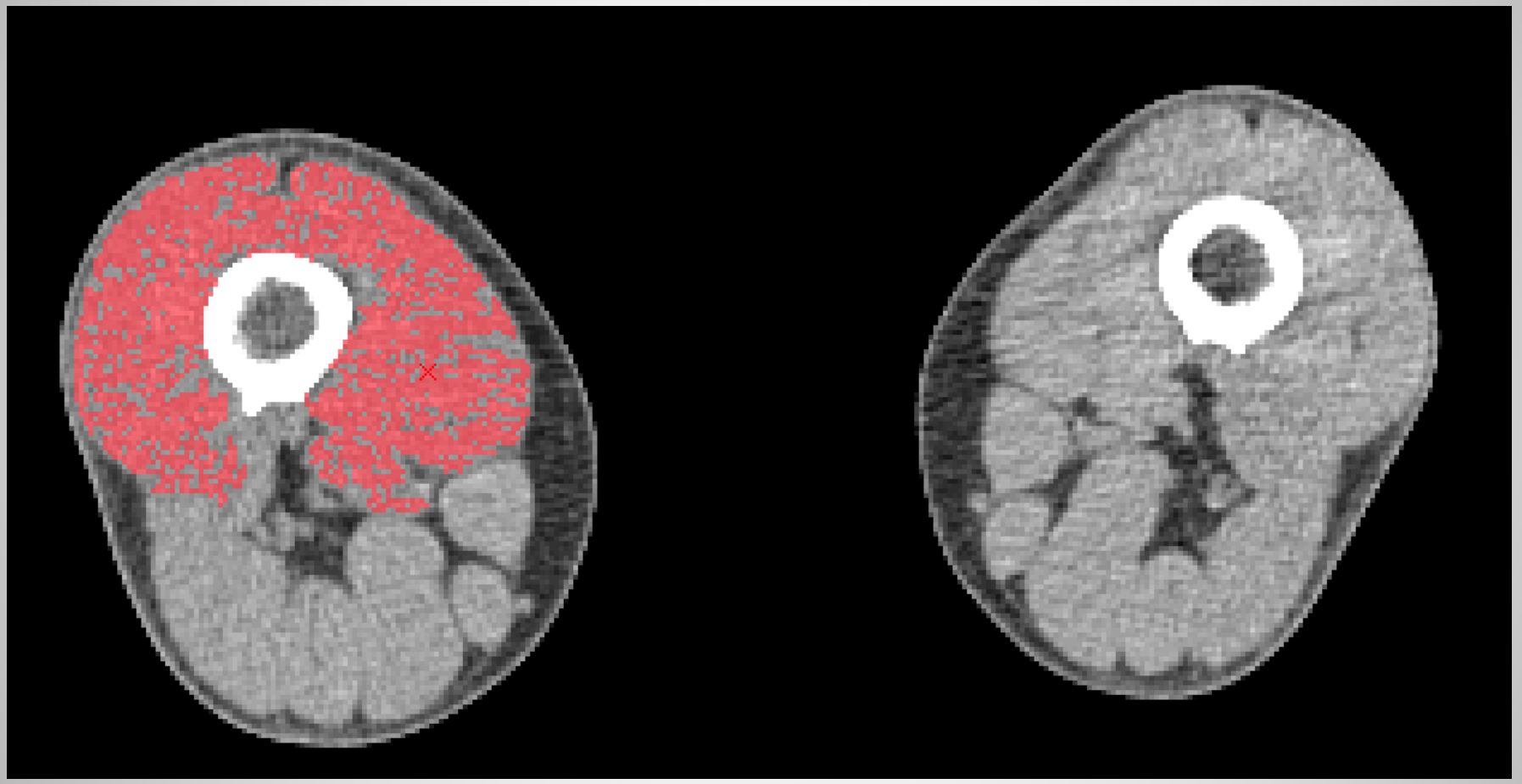
Ex: Muscle/Bone Segmentation in CT Scans



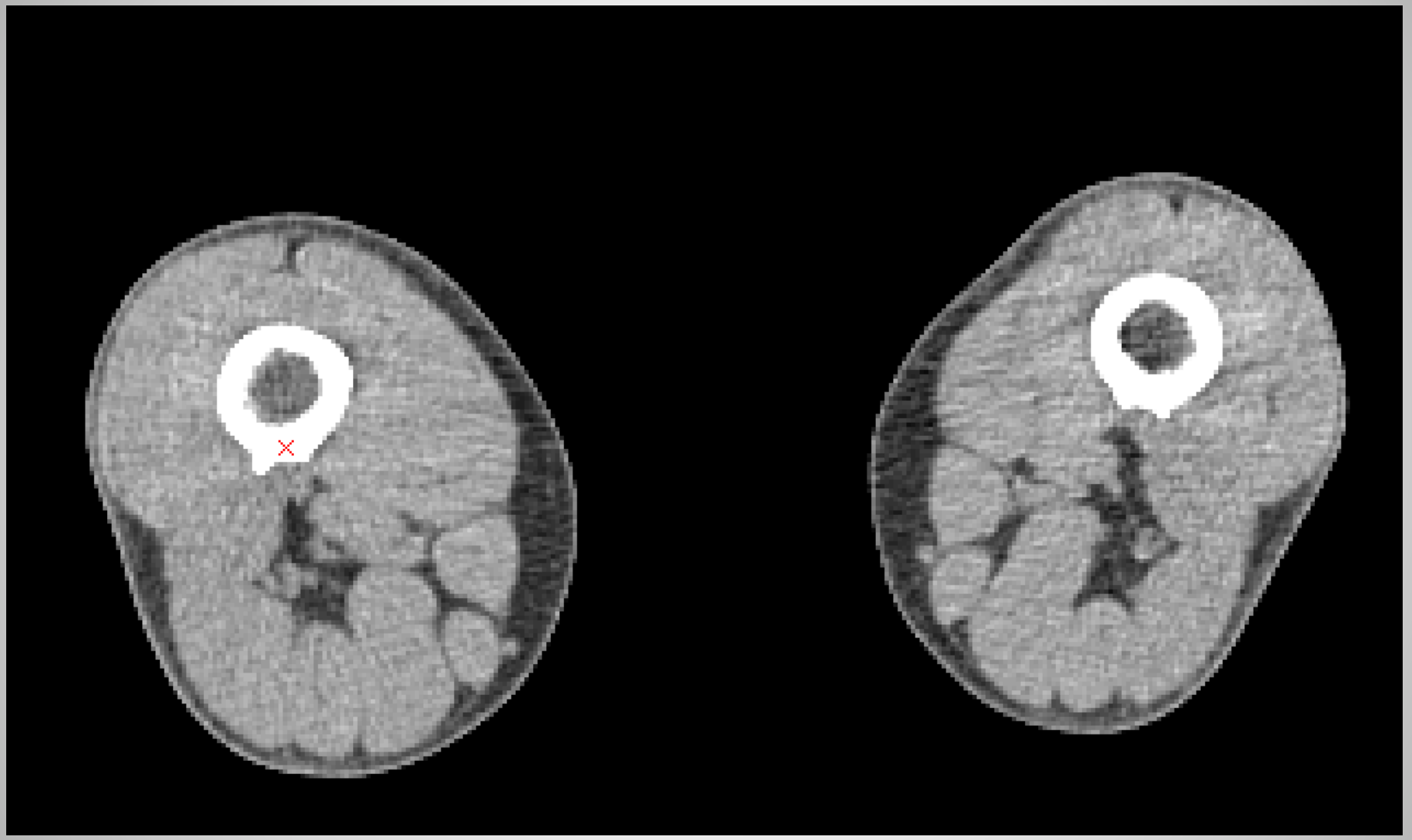
Ex: Muscle/Bone Segmentation in CT Scans



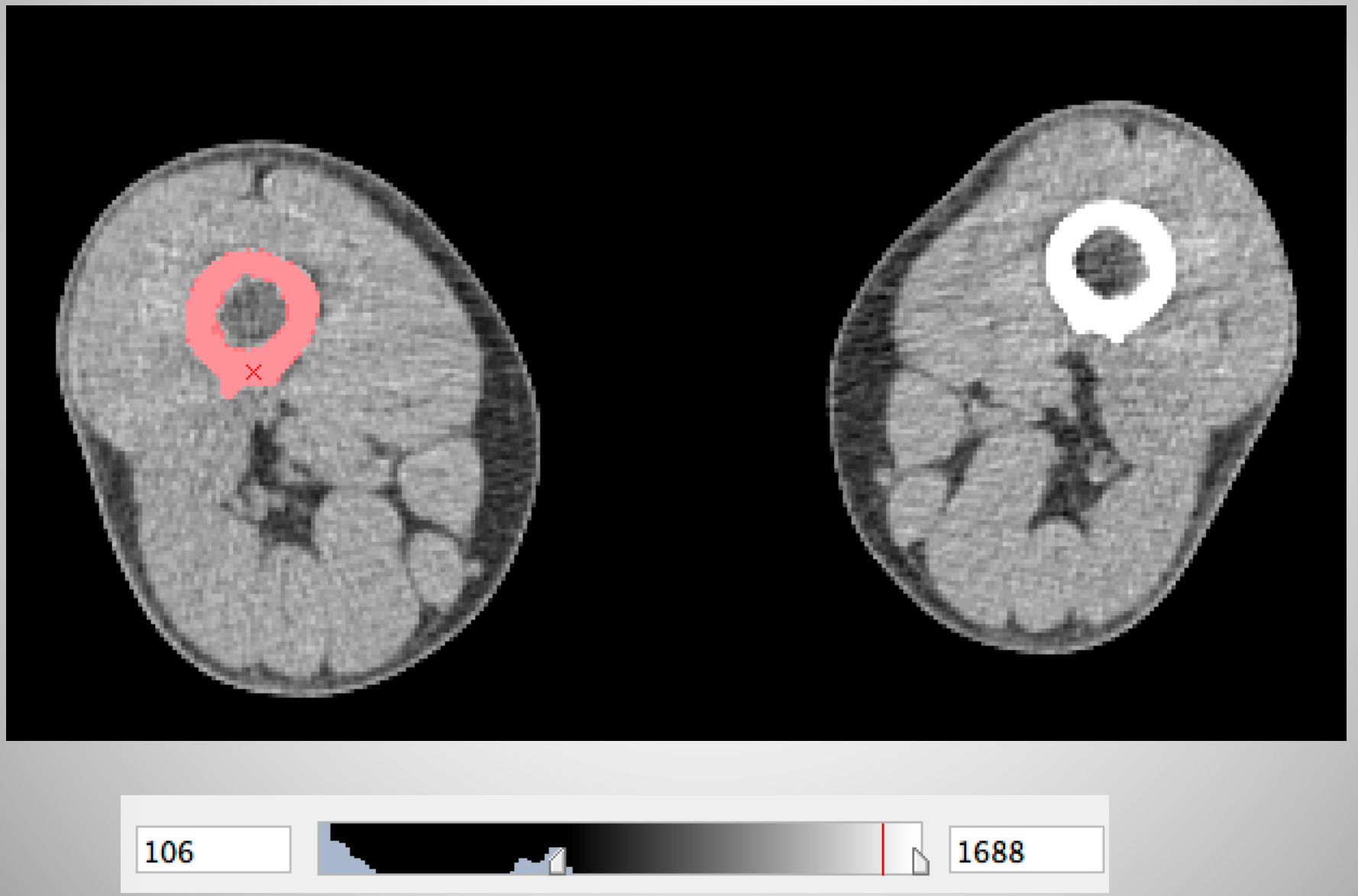
Ex: Muscle/Bone Segmentation in CT Scans



Ex: Muscle/Bone Segmentation in CT Scans



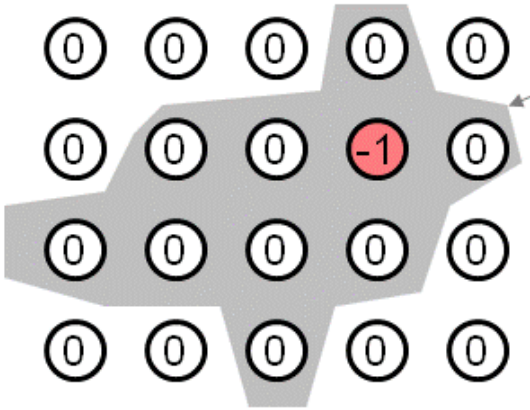
Ex: Muscle/Bone Segmentation in CT Scans



Region Growing Implementation

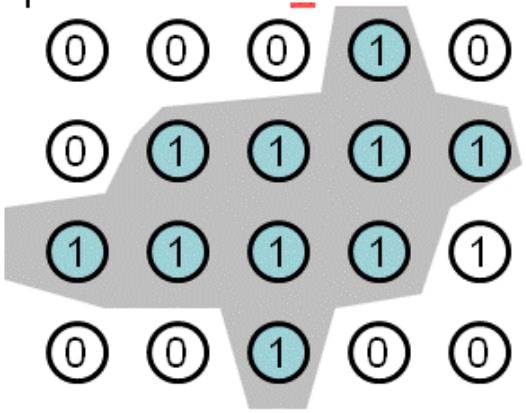
growRegion: red nodes are the "active_front" (queue or stack)

add seed into active_front



object with small *intensity* variation within the image

stop when active_front is empty

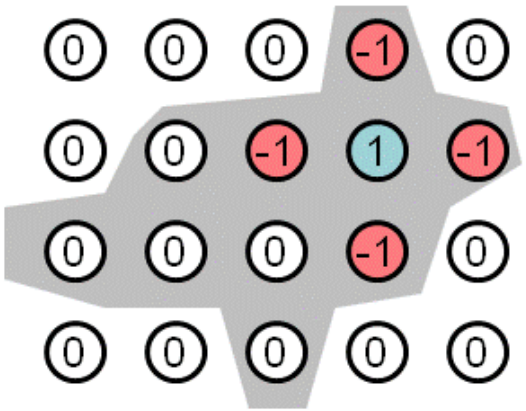


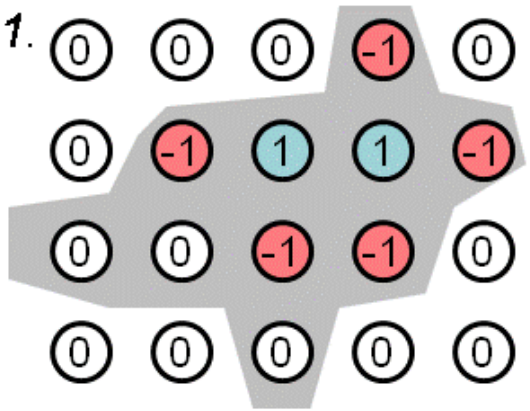
ALGORITHM:

Remove pixel *p* from active_front and mark it as ***region[p]*** = 1 .
Add all neighbors *q* such that:

$region[q] == 0,\ |I_p - I_q| < T$

and set ***region[q]*** = −1.





Limitations of Region Growing

Note that a complete segmentation of an image must satisfy a number of criteria:

1) All pixels must be assigned to regions

2) Each pixel must belong to a single region only

3) Each region must be a connected set of pixels

4) Each region must be uniform

5) Any merged pair of adjacent regions must be non-uniform

Comparison of Thresholding and Region Growing

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

threshold $T \geq 10$

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

threshold $T \geq 11$

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

threshold $T \geq 12$

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

region growing with variance of 2 in respect to value 11 with reference to threshold $T \geq 11$

Region splitting and Merging Segmentation

- Region splitting:
  - Unlike region growing, which starts from a set of seed points, region splitting starts with the whole image as a single region and subdivides it into subsidiary regions recursively while a condition of homogeneity is not satisfied.

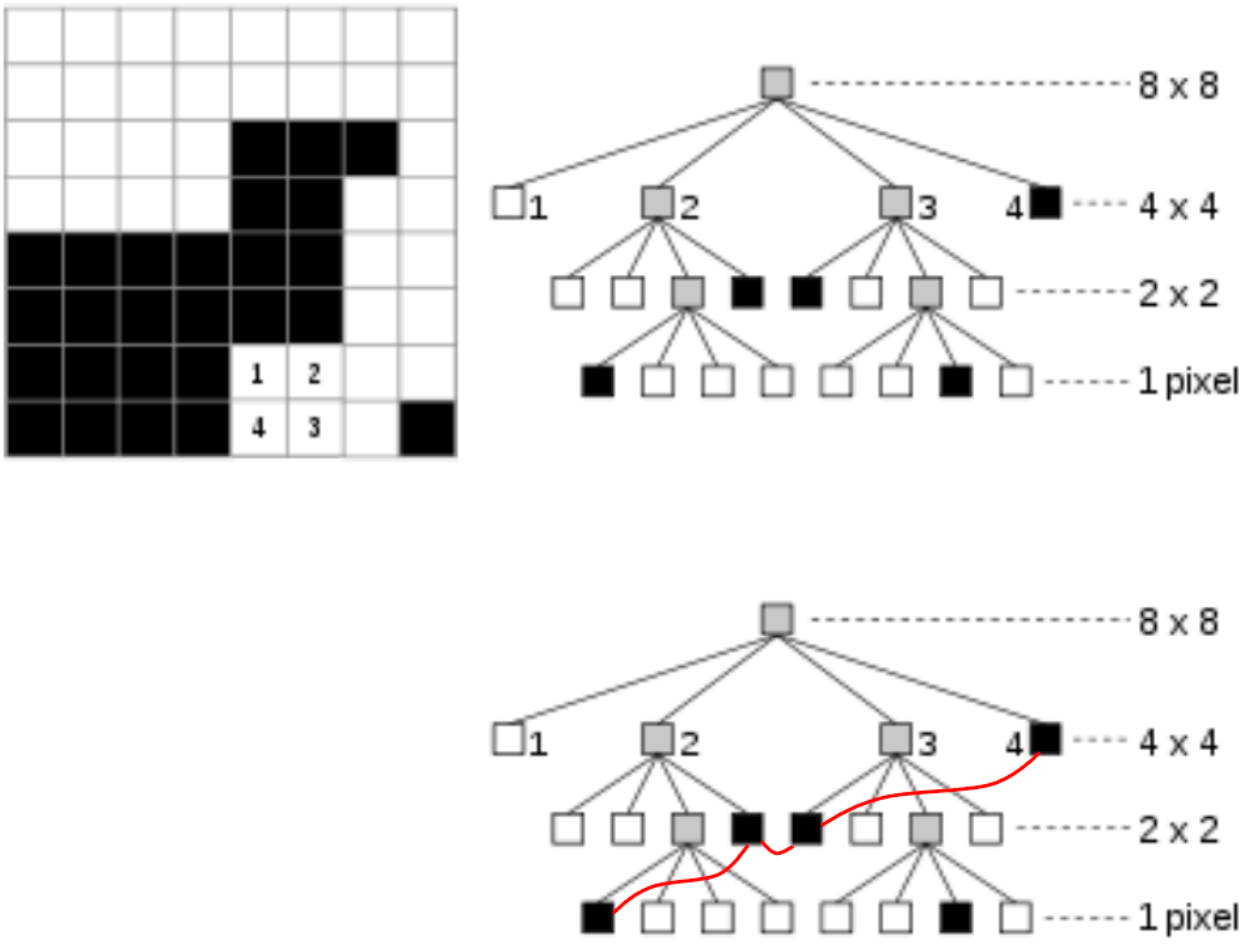
Region splitting and Merging Segmentation

- Region splitting:
  - Unlike region growing, which starts from a set of seed points, region splitting starts with the whole image as a single region and subdivides it into subsidiary regions recursively while a condition of homogeneity is not satisfied.
- Region merging:
  - Region merging is the opposite of splitting, and works as a way of avoiding over-segmentation

Region splitting and Merging Segmentation

- Region splitting:
  - Unlike region growing, which starts from a set of seed points, region splitting starts with the whole image as a single region and subdivides it into subsidiary regions recursively while a condition of homogeneity is not satisfied.
- Region merging:
  - Region merging is the opposite of splitting, and works as a way of avoiding over-segmentation
  - Start with small regions (2x2 or 4x4 regions) and merge the regions that have similar characteristics (such as gray level, variance).

Region splitting and Merging Segmentation



RAG with adjacency relations (in red) for big black region.

- RAG: region adjacency graph
- **Quadtree** for splitting (top-down) procedure

Region splitting and Merging Segmentation

Algorithm:

- If a region R is inhomogeneous (P (R)=FALSE), then R is split into four sub-regions.
- If two adjacent regions R_i,R_j are homogeneous (P(R_iUR_j)=TRUE), they are then merged.
- The algorithm stops when no further splitting or merging is possible.

Region splitting and Merging Segmentation

0	1	0	0	7	7	7	7
1	0	2	2	7	7	7	7
0	2	2	2	7	7	7	7
4	4	2	2	7	7	7	7
0	0	1	1	3	3	7	7
1	1	2	2	3	7	7	7
2	4	3	0	5	7	7	7
2	3	3	5	5	0	7	7

original image

0	1	0	0	7	7	7	7
1	0	2	2	7	7	7	7
0	2	2	2	7	7	7	7
4	4	2	2	7	7	7	7
0	0	1	1	3	3	7	7
1	1	2	2	3	7	7	7
2	4	3	0	5	7	7	7
2	3	3	5	5	0	7	7

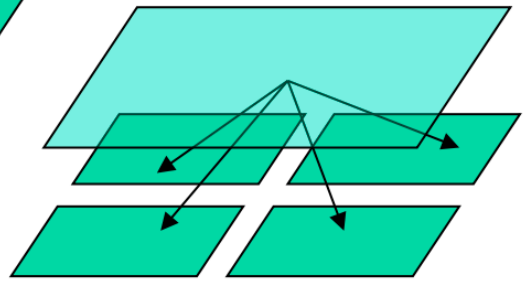
split 1

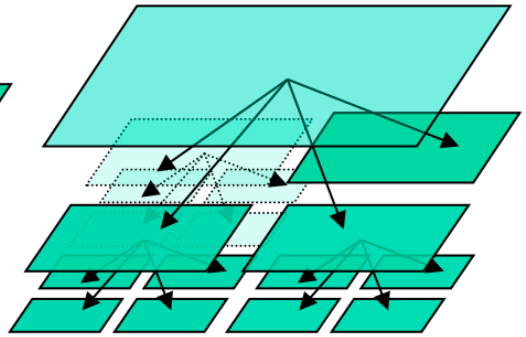
0	1	0	0	7	7	7	7
1	0	2	2	7	7	7	7
0	2	2	2	7	7	7	7
4	4	2	2	7	7	7	7
0	0	1	1	3	3	7	7
1	1	2	2	3	7	7	7
2	4	3	0	5	7	7	7
2	3	3	5	5	0	7	7

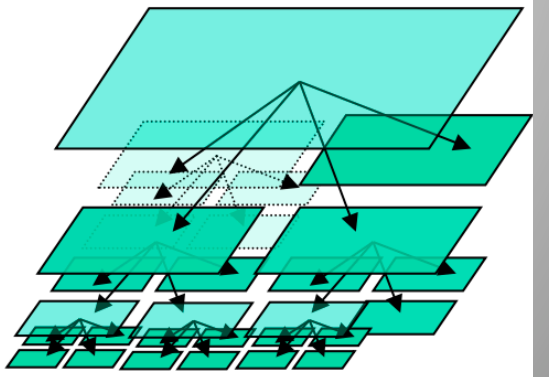
split 2

0	1	0	0	7	7	7	7
1	0	2	2	7	7	7	7
0	2	2	2	7	7	7	7
4	4	2	2	7	7	7	7
0	0	1	1	3	3	7	7
1	1	2	2	3	7	7	7
2	4	3	0	5	7	7	7
2	3	3	5	5	0	7	7

split 3







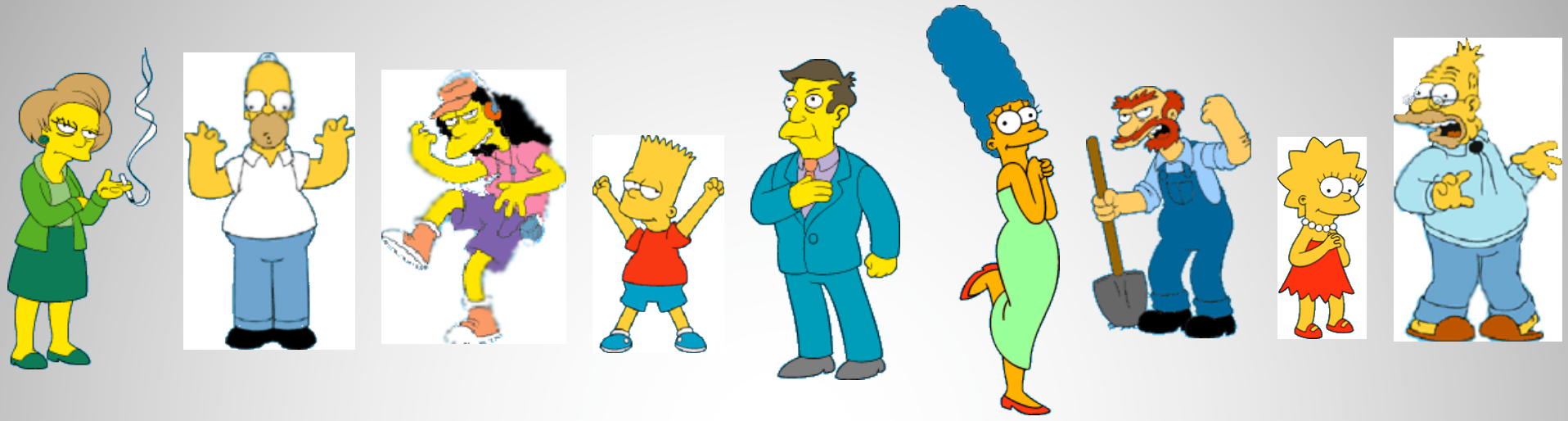
Clustering Based Segmentation Methods

What is Clustering?

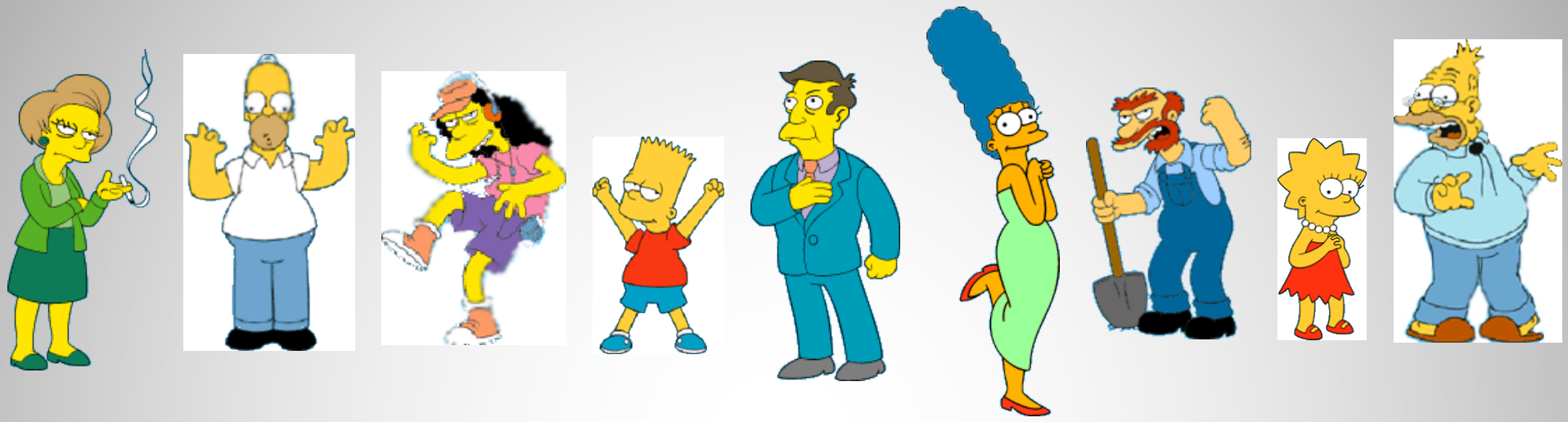
- Organizing data into classes such that:
  - High intra-class similarity
  - Low inter-class similarity

- Finding the class labels and the number of classes directly from the data (as opposed to *classification tasks*)

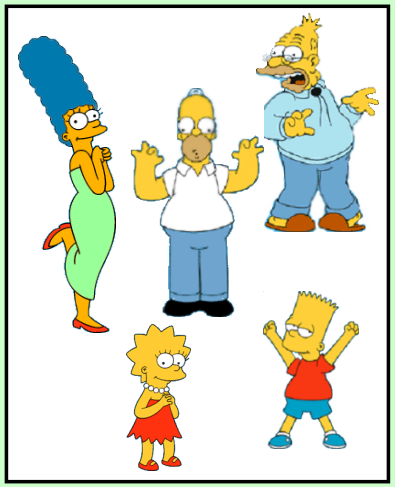
What is a natural grouping?



What is a natural grouping?

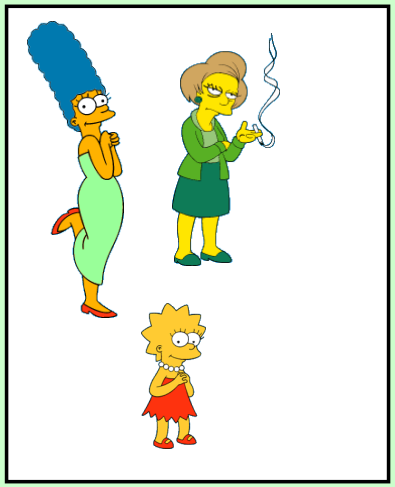


Clustering is subjective



Simpson's Family

School Employees



Females

Males

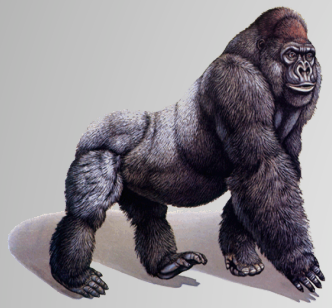
What is similarity ?

What is similarity ?

Cluster by features

- **Color**
- **Intensity**
- **Location**
- **Texture**
- **….**

Distance metrics



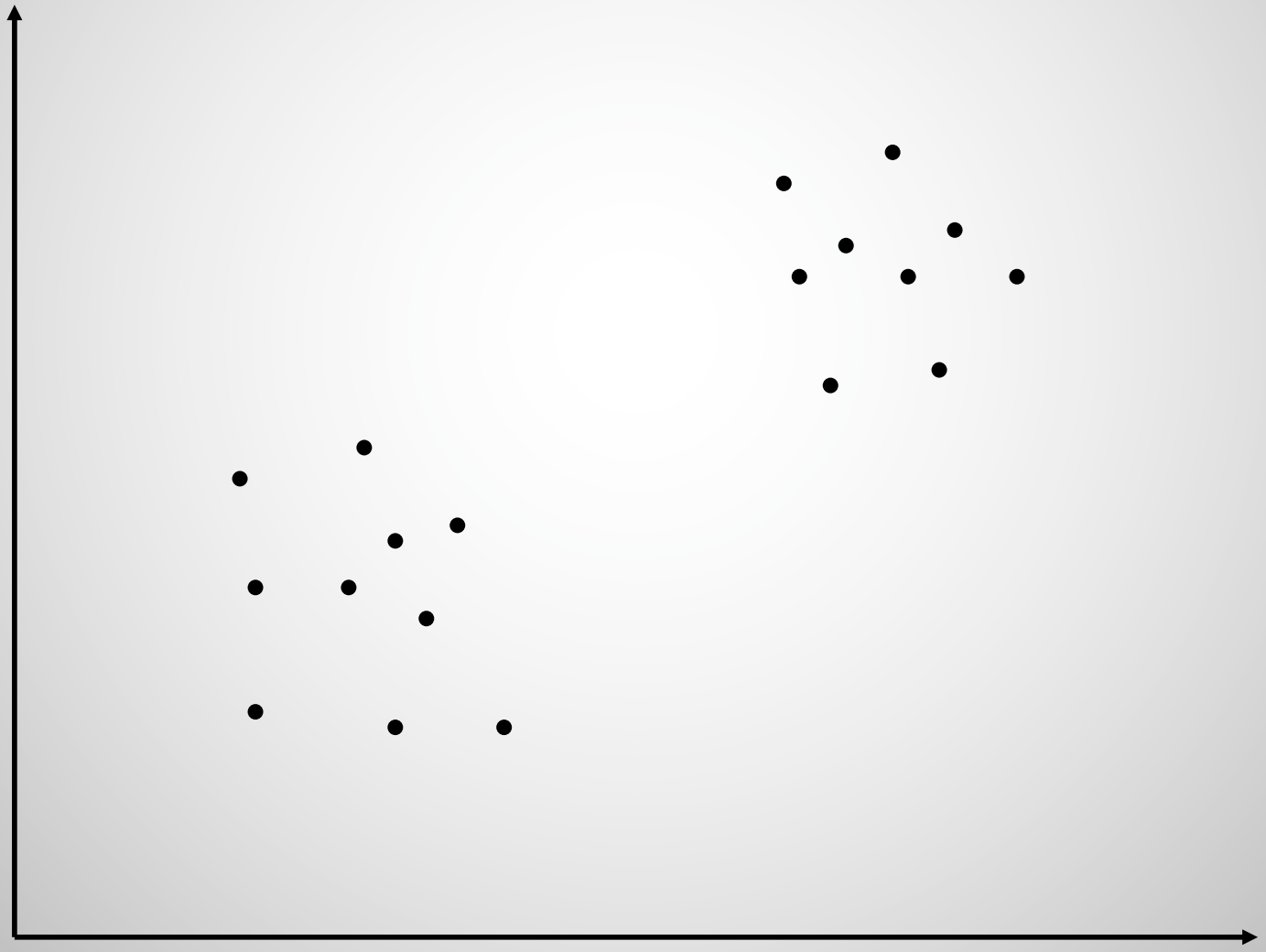
Peter Piotr

0.23

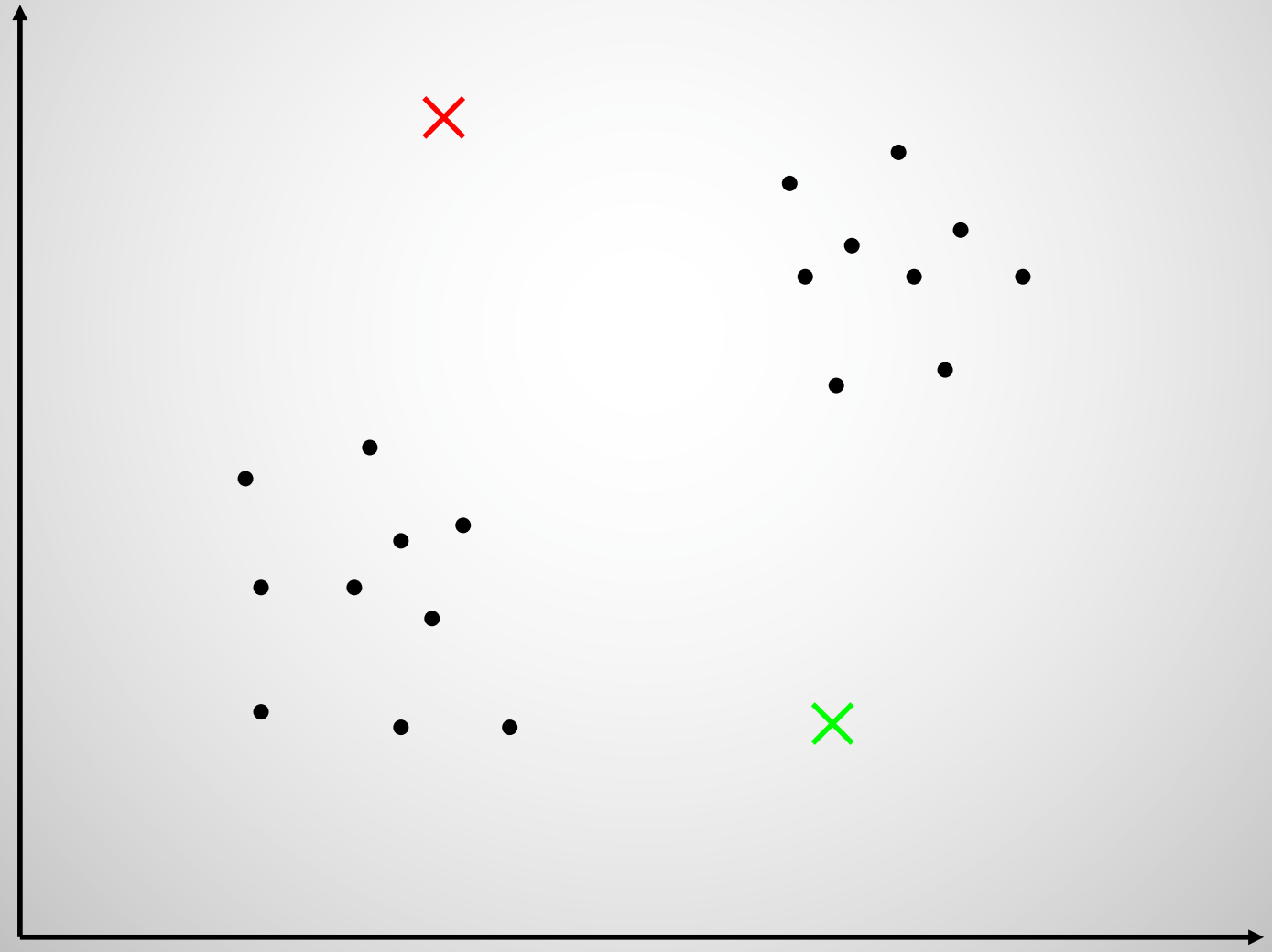
3

342.7

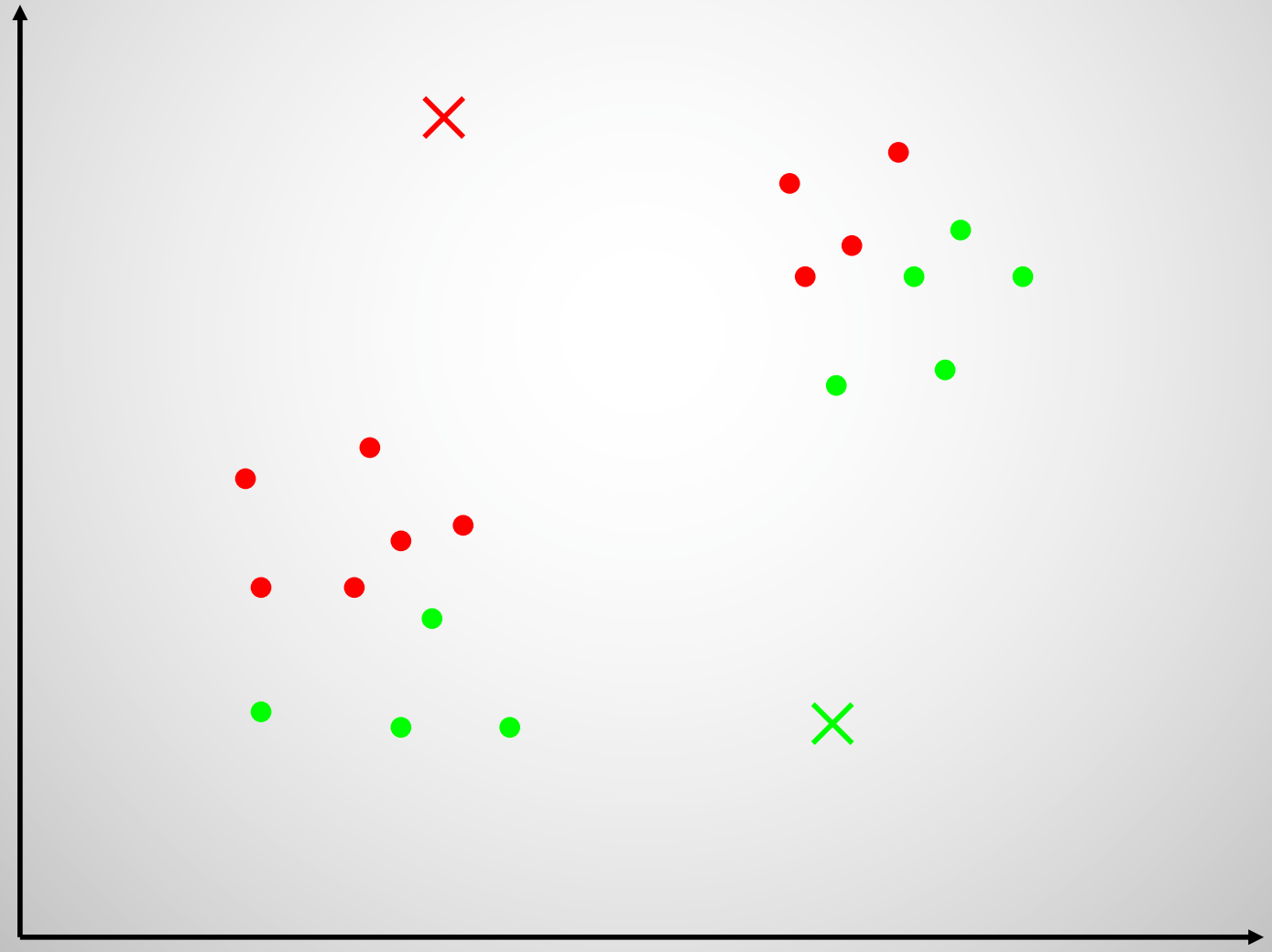
K-means Clustering



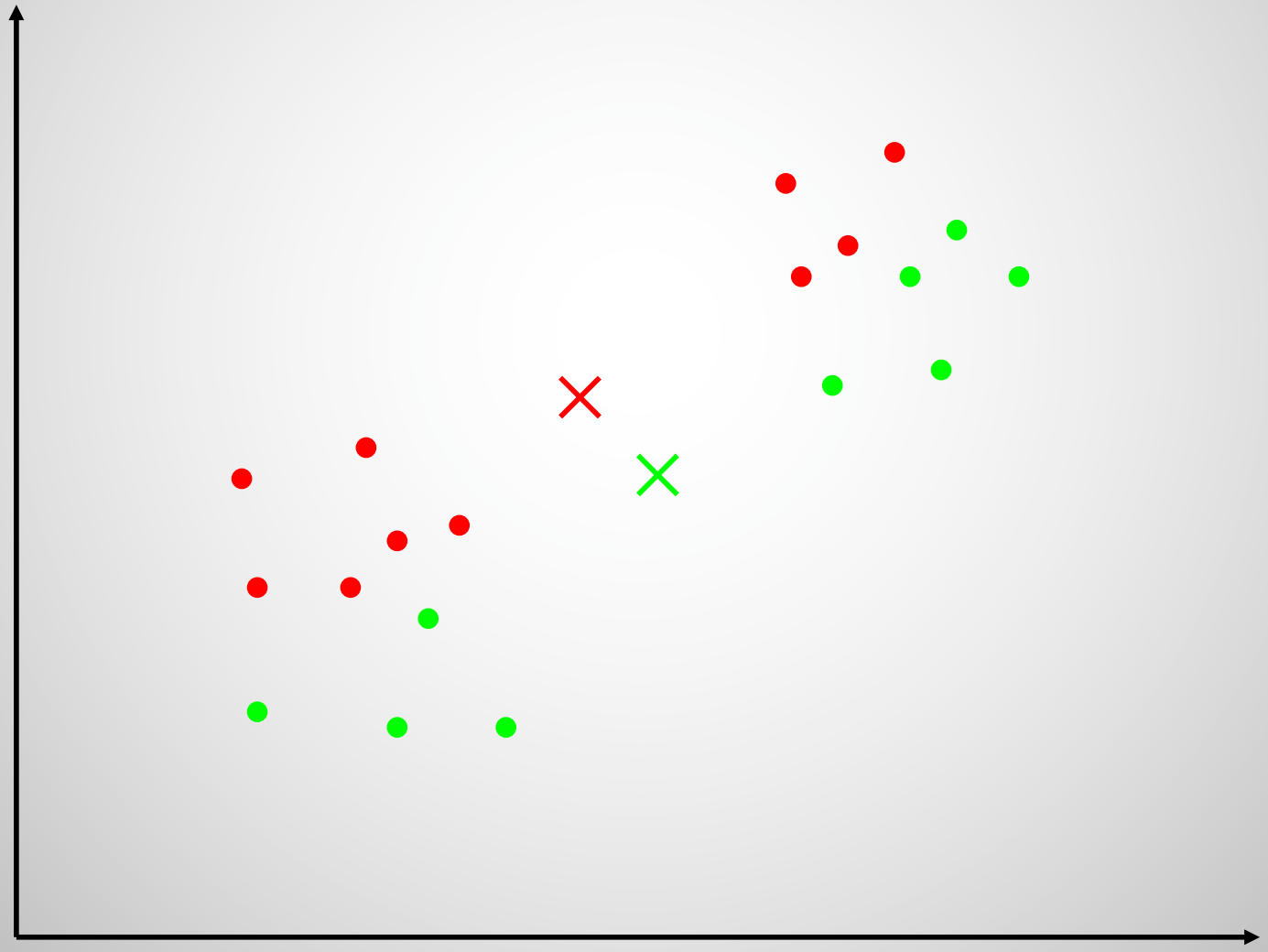
K-means Clustering



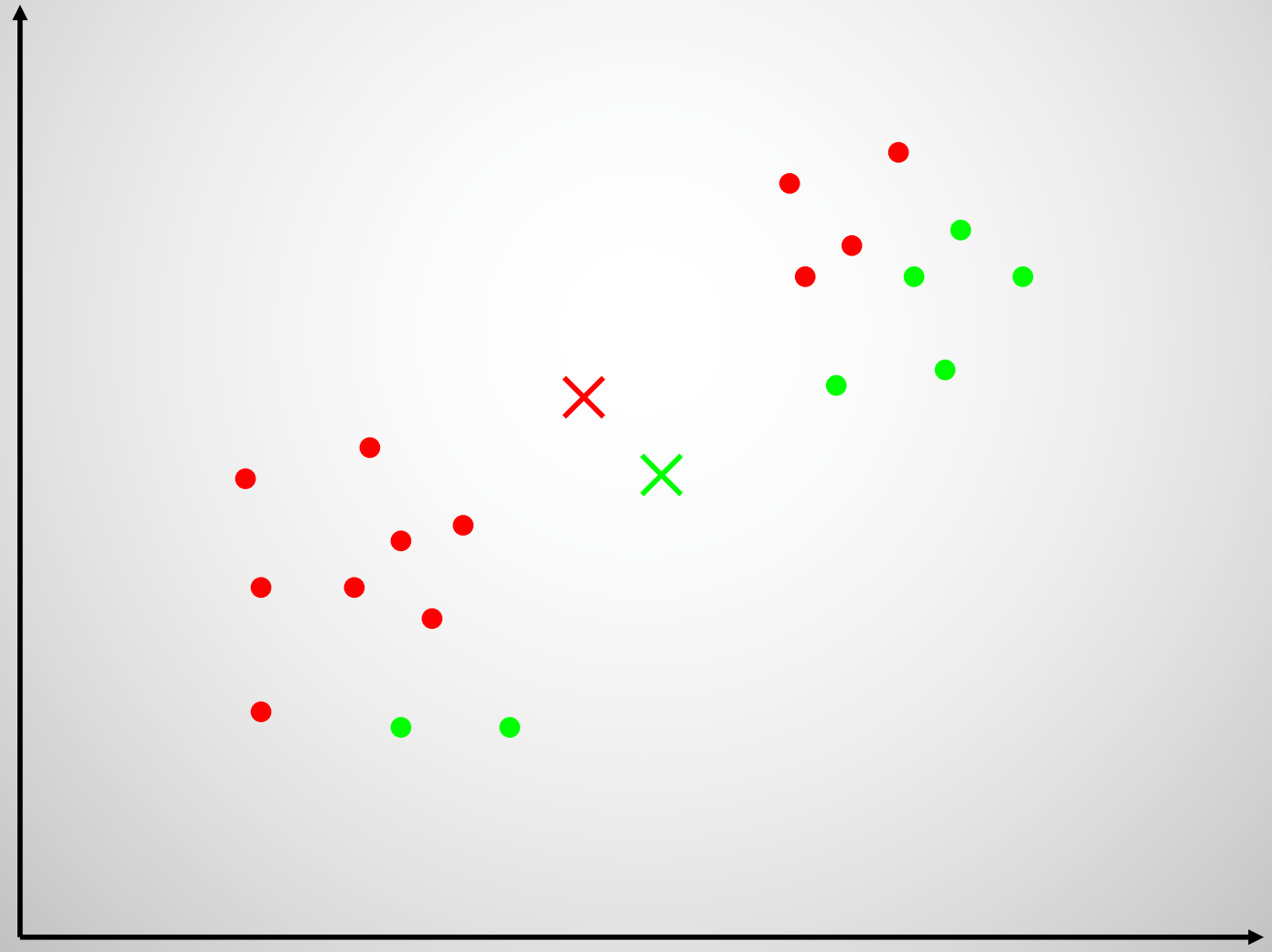
K-means Clustering



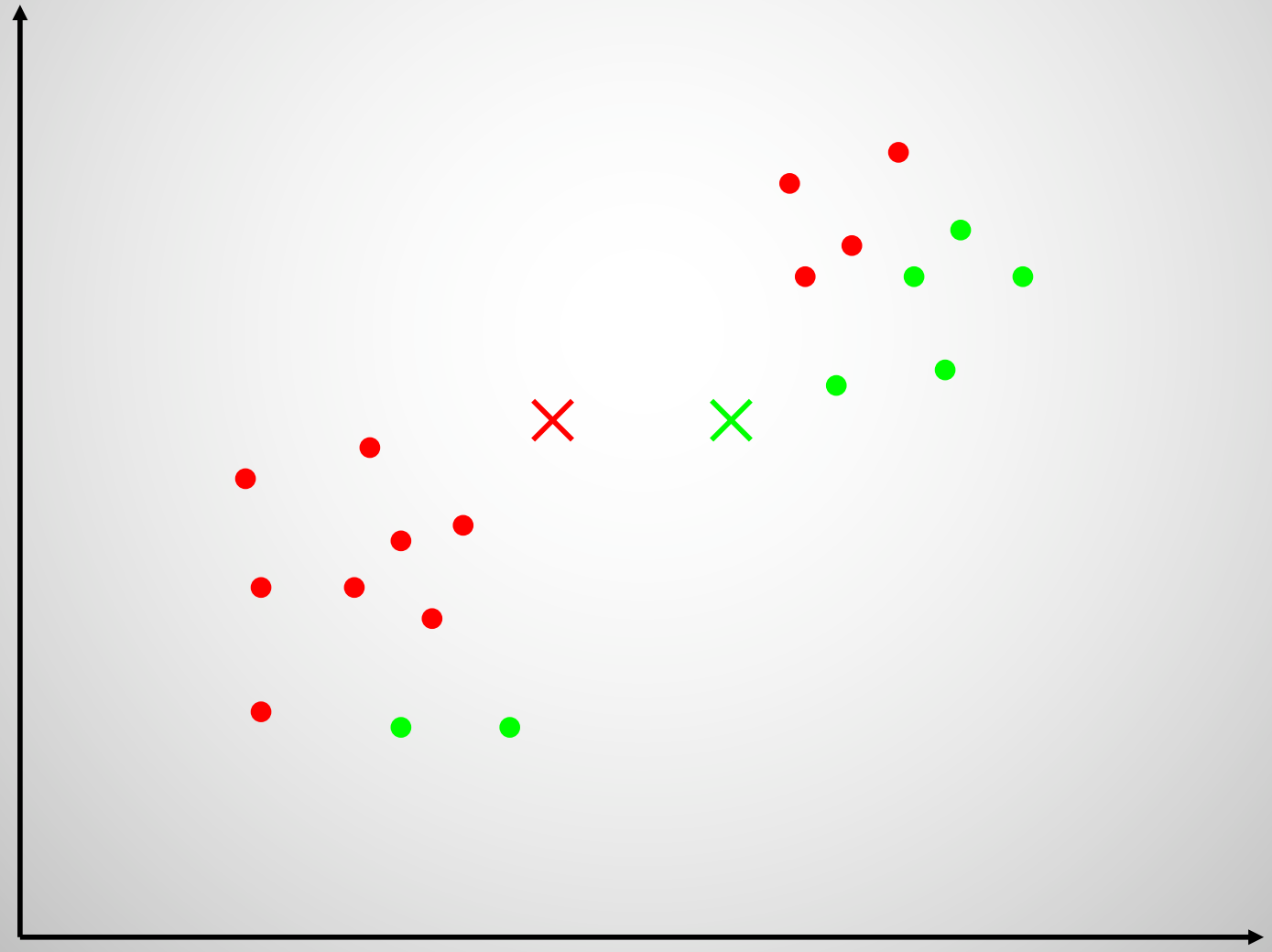
K-means Clustering



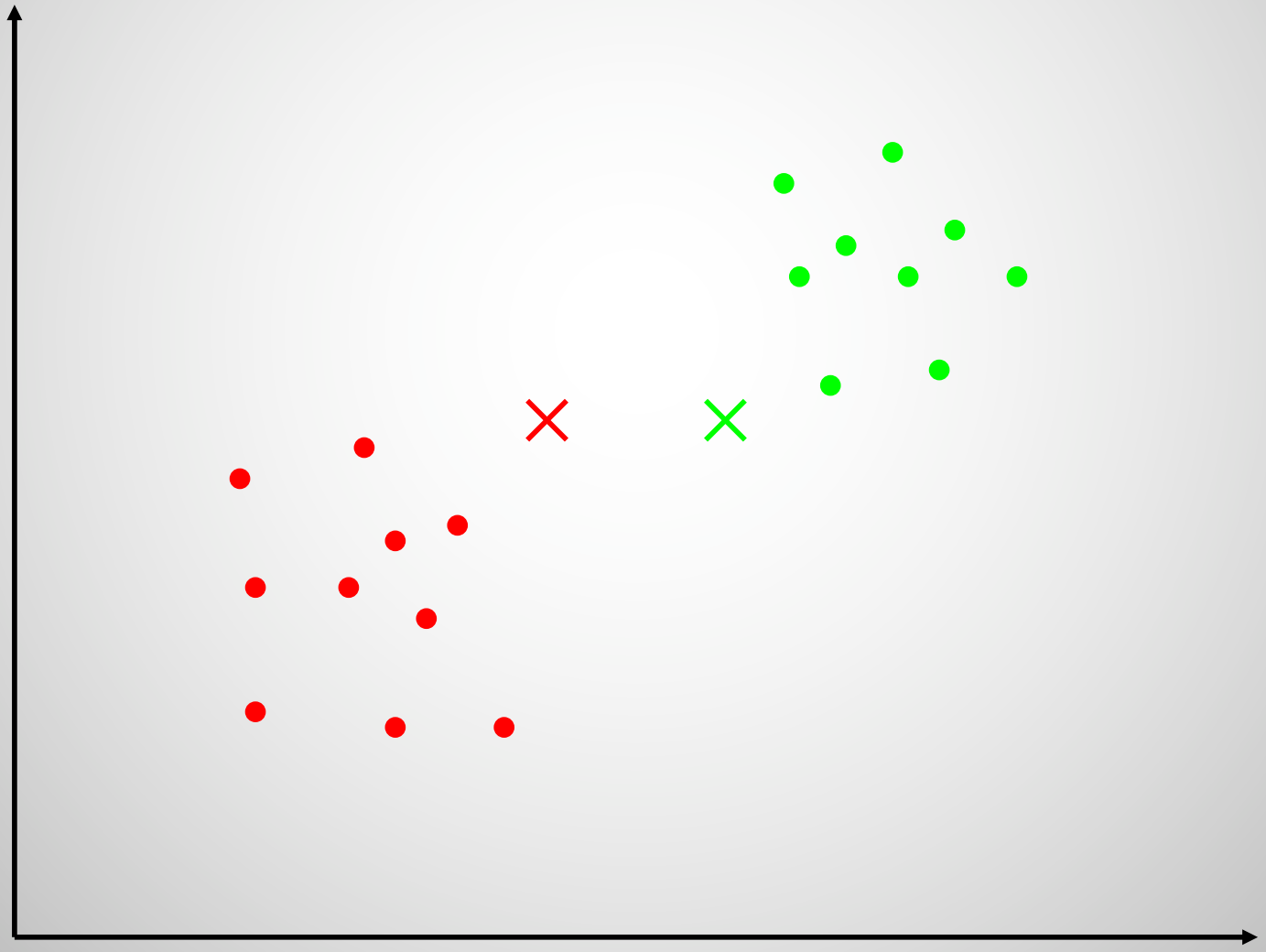
K-means Clustering



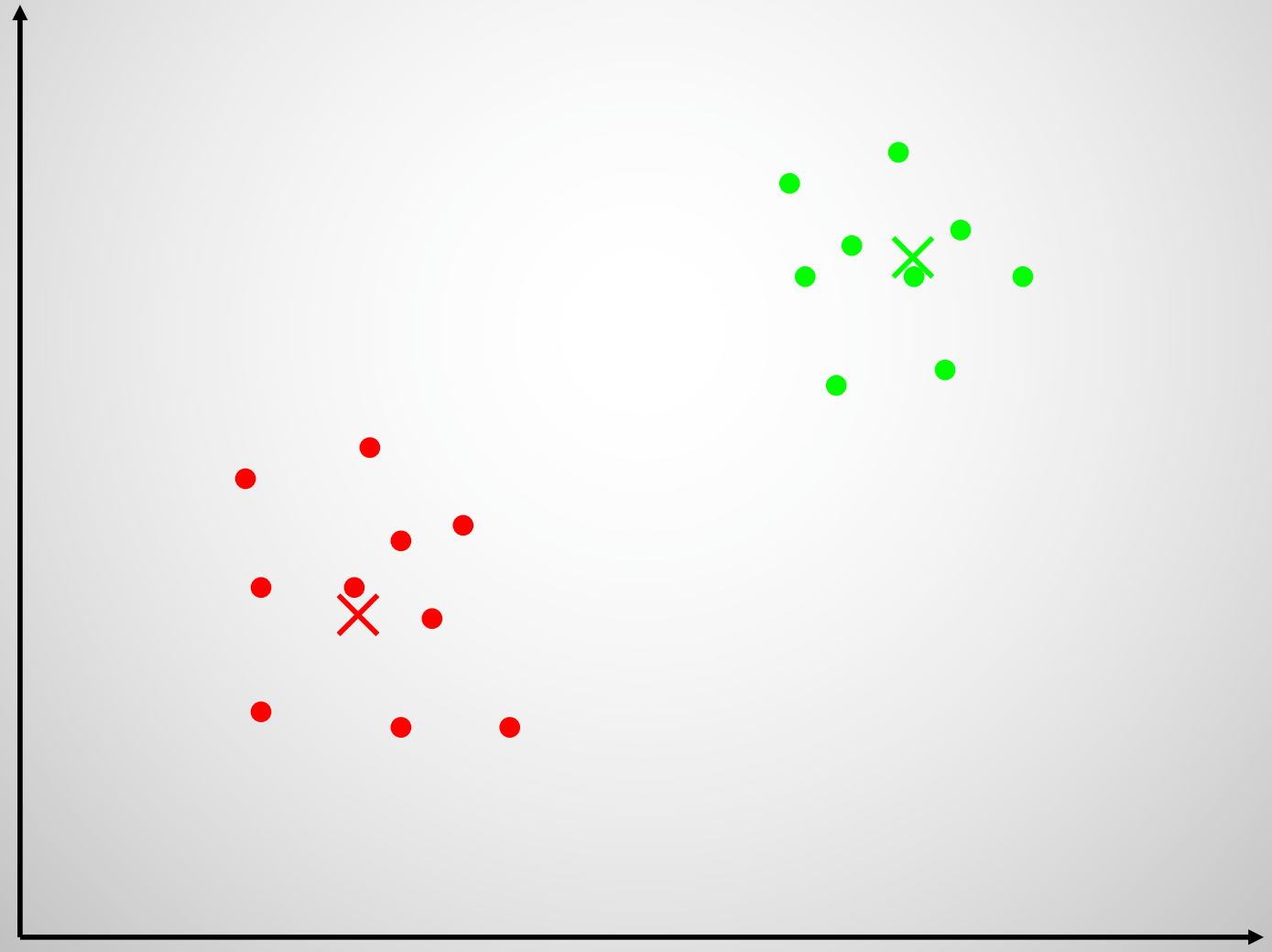
K-means Clustering



K-means Clustering

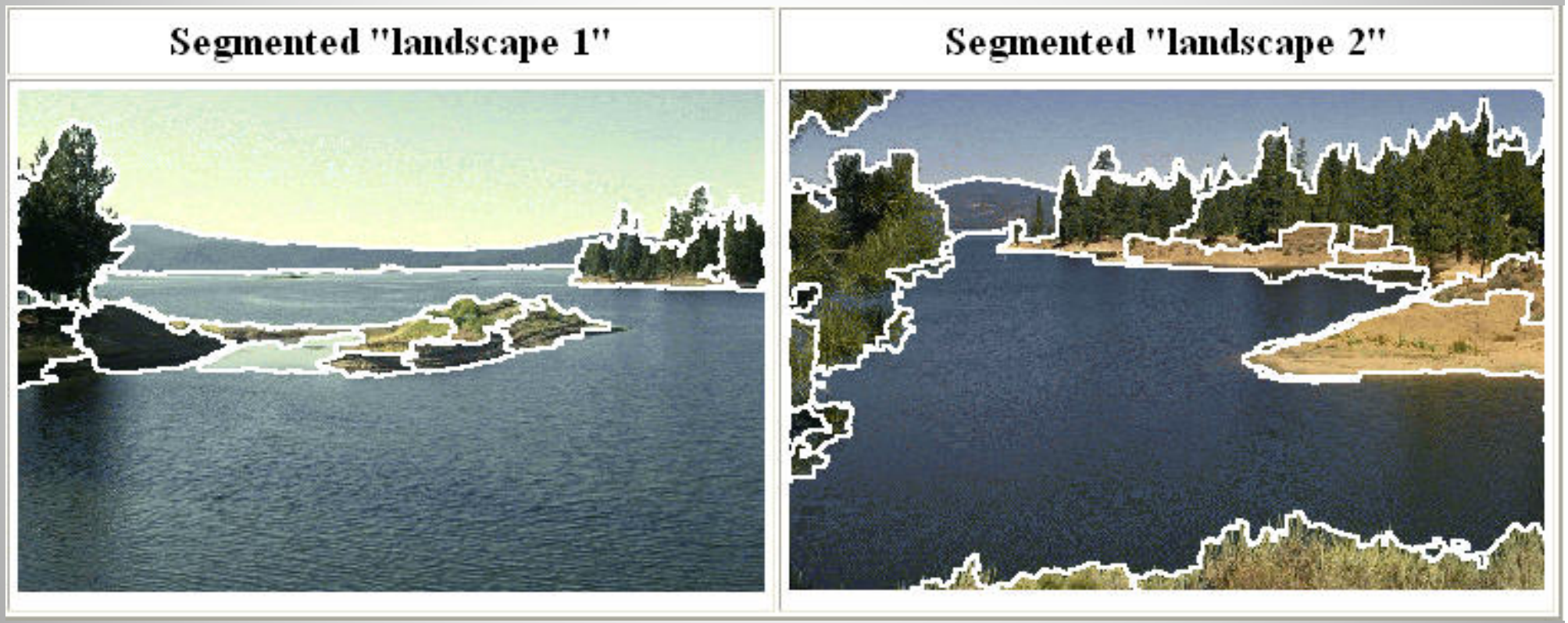


K-means Clustering



Mean-Shift Segmentation

- **Mean-shift** is a variant of an iterative steepest-ascent method to seek stationary points (i.e., peaks) in a density function, which is applicable in many areas of multi-dimensional data analysis.



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

Mean-Shift Segmentation

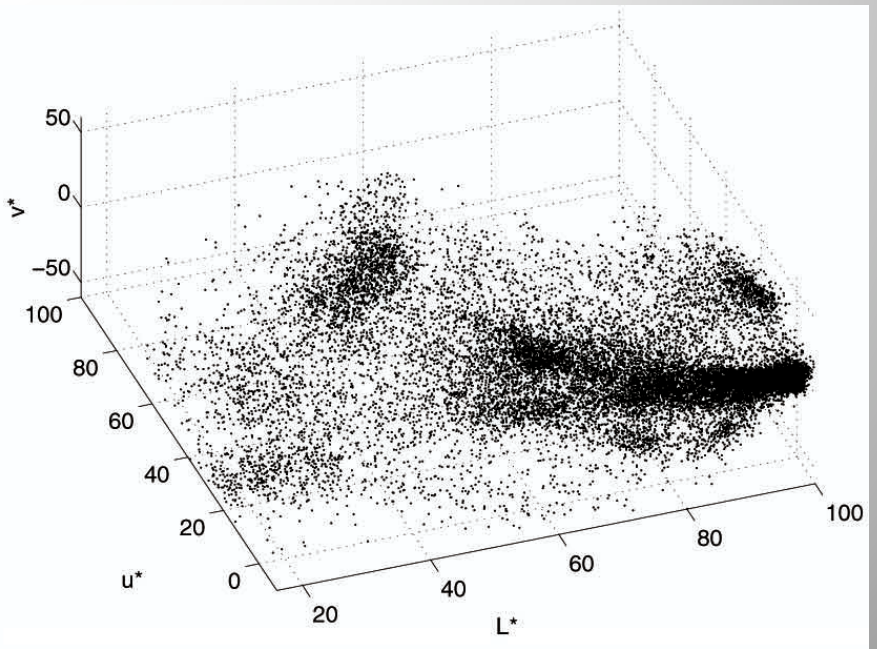
- **Mean-shift** is a variant of an iterative steepest-ascent method to seek stationary points (i.e., peaks) in a density function, which is applicable in many areas of multi-dimensional data analysis.
- openCV -> meanShift method available.
- Y.Cheng, PAMI 1995 and Comaniciu and Meer, PAMI 2002 papers.

Mean-Shift Segmentation

- **Mean-shift** is a variant of an iterative steepest-ascent method to seek stationary points (i.e., peaks) in a density function, which is applicable in many areas of multi-dimensional data analysis.
- openCV -> meanShift method available.
- Y.Cheng, PAMI 1995 and Comaniciu and Meer, PAMI 2002 papers.
- Attempts to find all possible cluster centers in feature space (unlike k-means, where there is a requirement to know the number of different clusters).

Mean-Shift Segmentation

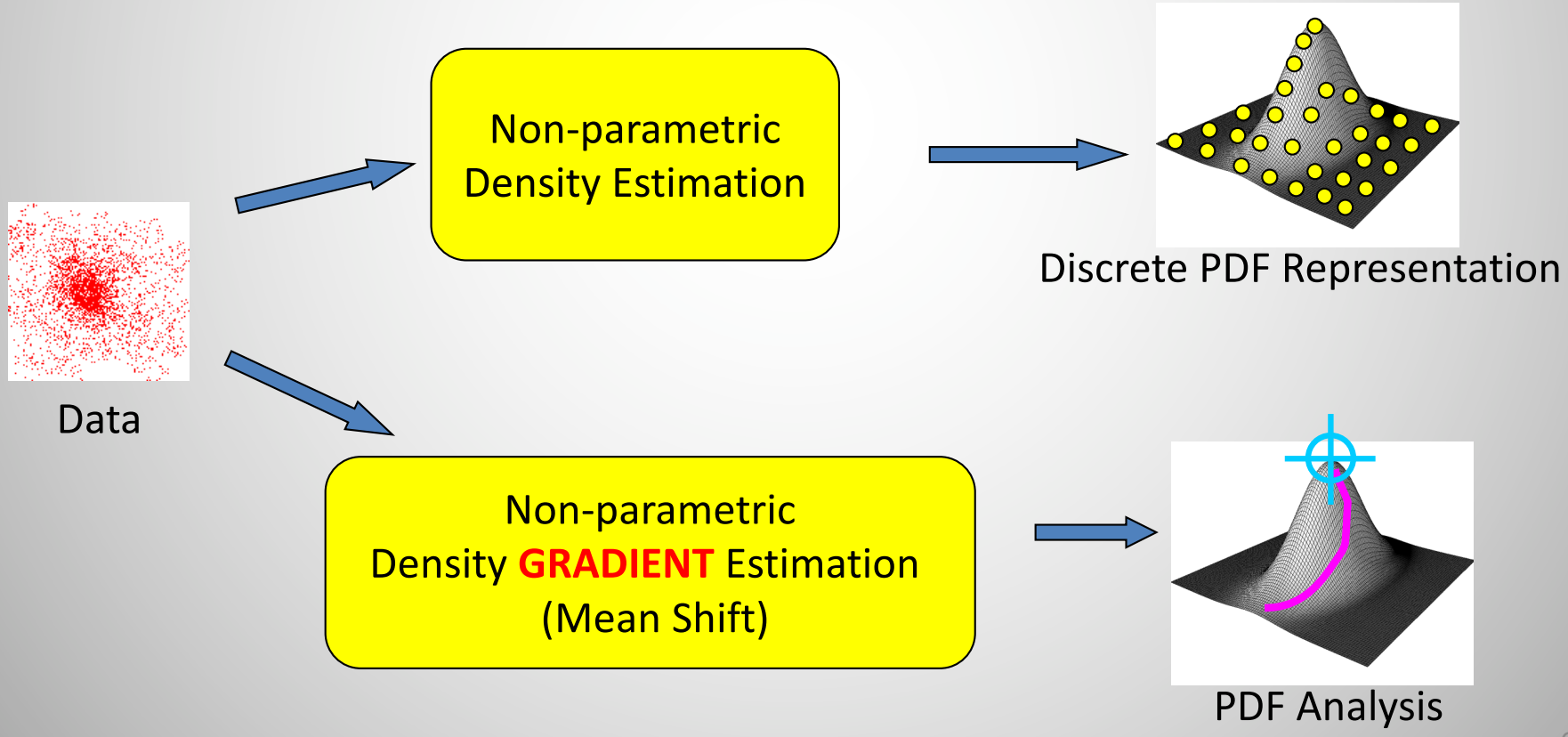
- **Mean-shift** is a variant of an iterative steepest-ascent method to seek stationary points (i.e., peaks) in a density function, which is applicable in many areas of multi-dimensional data analysis.
- openCV -> meanShift method available.
- Y.Cheng, PAMI 1995 and Comaniciu and Meer, PAMI 2002 papers.
- The mean shift algorithm seeks *modes* or local maxima of density in the feature space



Mean-Shift

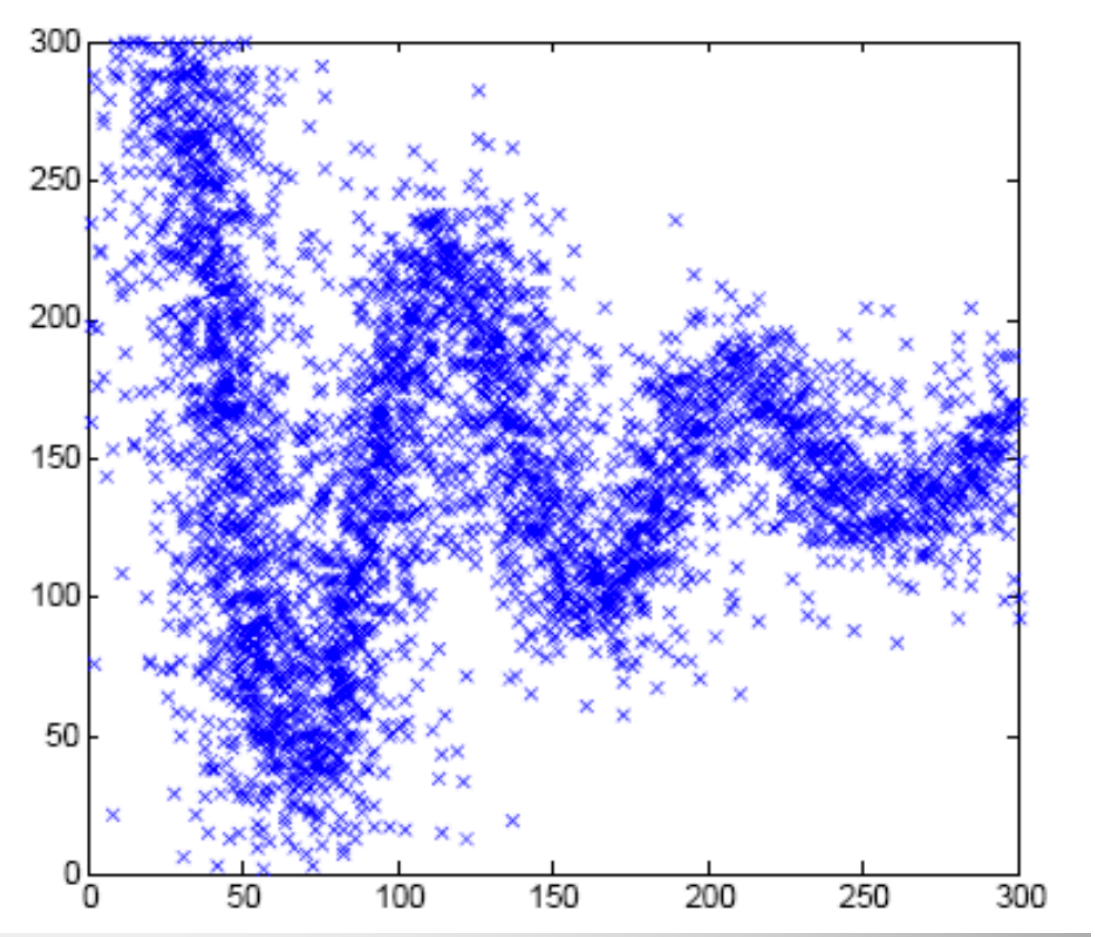
A tool for:

Finding modes in a set of data samples, manifesting an underlying probability density function (PDF) in R^N



Density Estimation

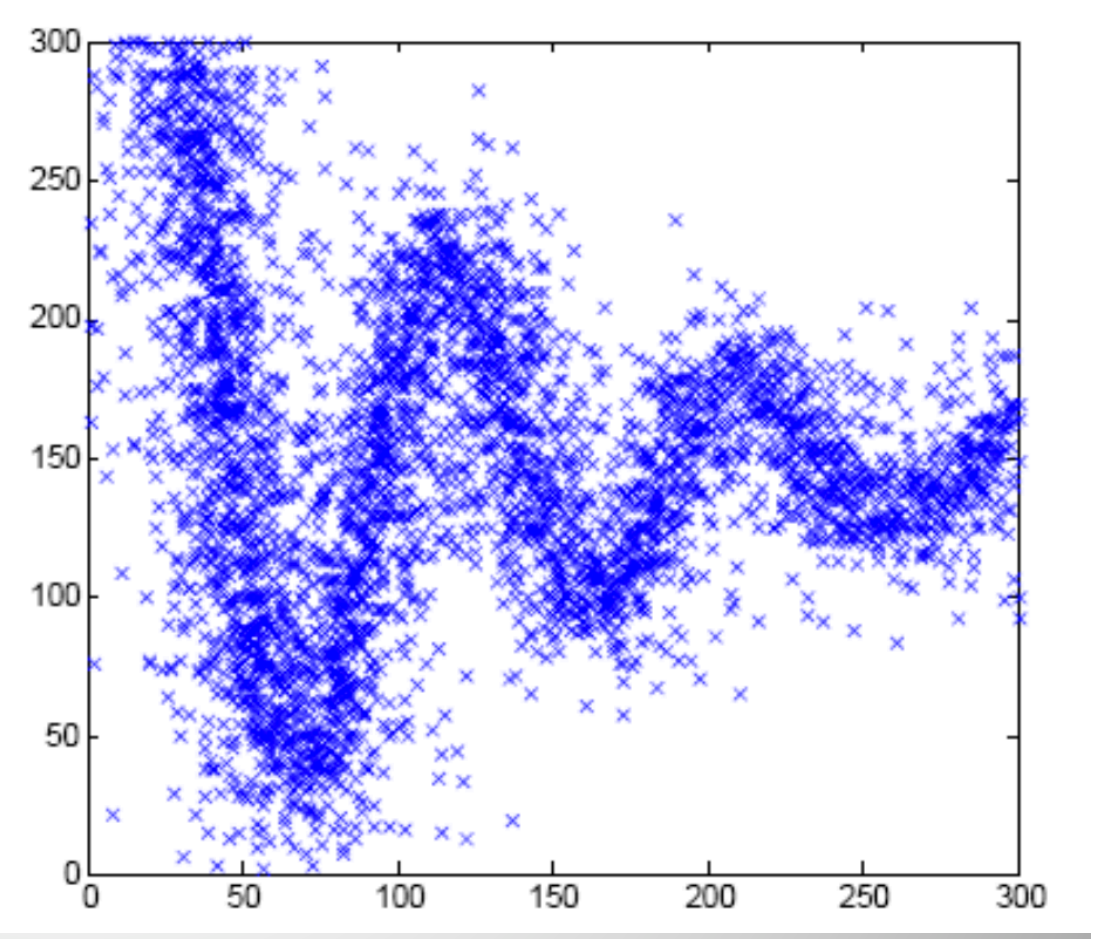
- What is the distribution that generated these points?



Density Estimation

- What is the distribution that generated these points?
- **Parametric model:**

Can express distr. with a few parameters (mean And variance)



Density Estimation

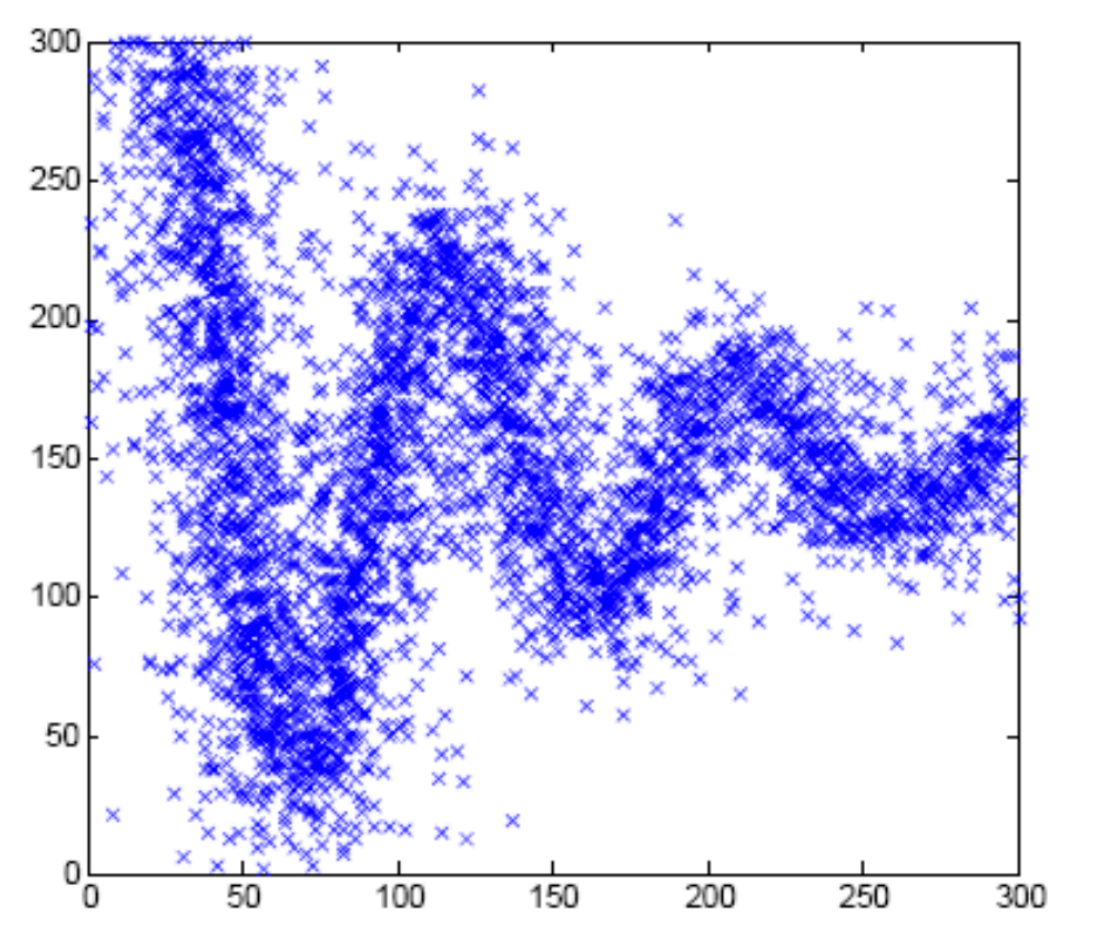
- What is the distribution that generated these points?
- **Parametric model:**

Can express distr. with

a few parameters (mean

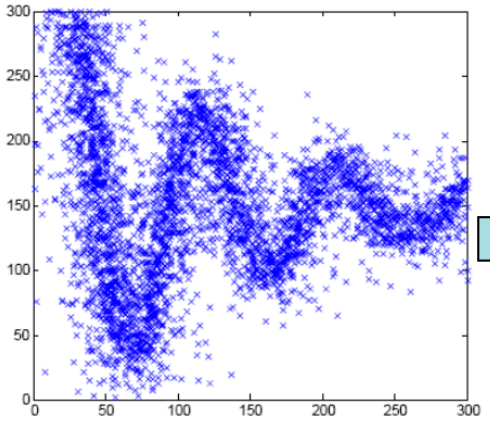
And variance)

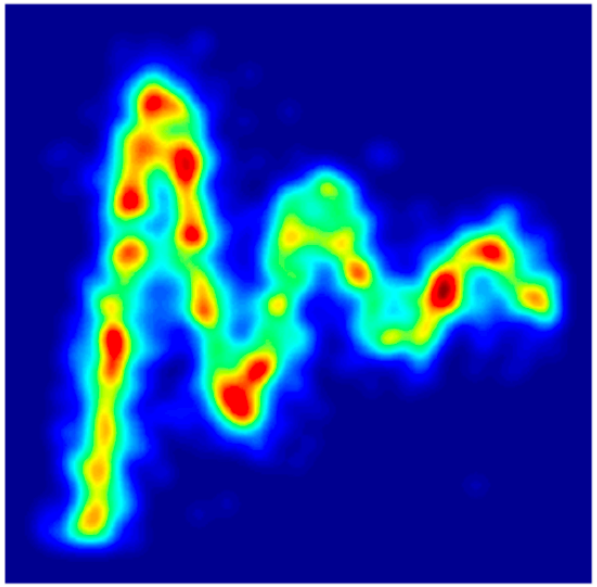
- Limited in flexibility!



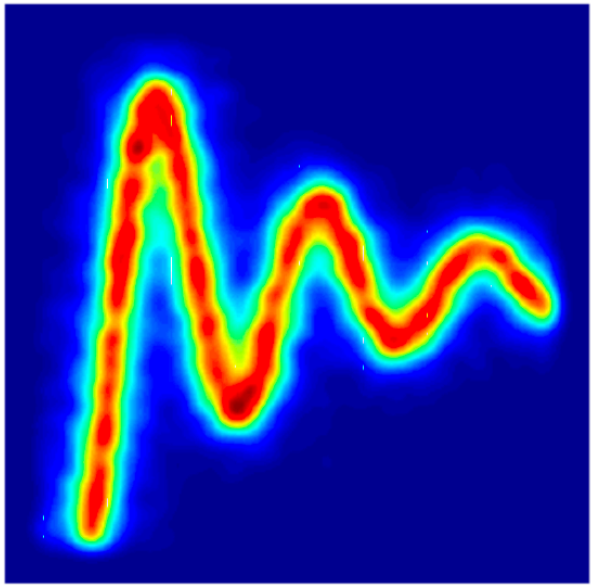
Non-parametric density estimation

- Focus on kernel density estimates, using the data to define the distribution
- Build distribution by putting a little mass of probability around each data-point





(a) 2000 Samples

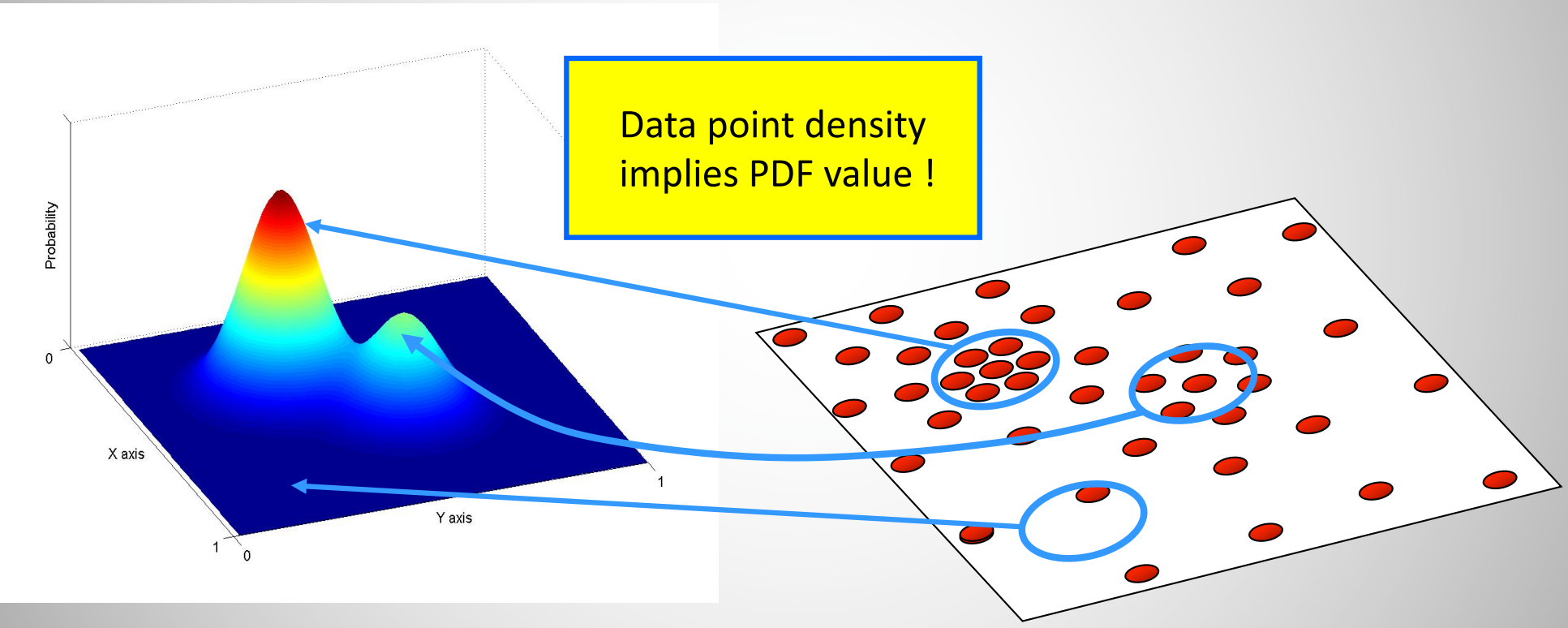


(b) 20000 Samples

M. Tappen, PhD. Thesis

Nonparametric Density Estimation

Determining parameters of unknown number of probability functions of an unknown type will be difficult!



Assumed Underlying PDF

Real Data Samples

Basics Assumptions

- The probability density functions of the mixture models have only one max and that this maximum represents the mean of the function
- Combining the probability functions in the mixture model preserves the maxima, local maxima are the modes of the density functions
- The local minima of the mixture model segment the feature space so that each segment contain only one of the local maxima

Recap: Nonparametric Density Estimation

- Consider image pixels as data points (x).

Recap: Nonparametric Density Estimation

- Consider image pixels as data points (x).
- Probability that x belongs to a sub-domain D

$$P = \int_D p(x)dx$$

Recap: Nonparametric Density Estimation

- Consider image pixels as data points (x).
- Probability that x belongs to a sub-domain D

$$P = \int_D p(x)dx$$

- If D is small, pdf is fairly constant inside, so

Recap: Nonparametric Density Estimation

- Consider image pixels as data points (x).
- Probability that x belongs to a sub-domain D

$$P = \int_D p(x)dx$$

- If D is small, pdf is fairly constant inside, so

$$P \approx p(x) \int_D dx$$

Recap: Nonparametric Density Estimation

- Consider image pixels as data points (x).
- Probability that x belongs to a sub-domain D

$$P = \int_D p(x)dx$$

- If D is small, pdf is fairly constant inside, so

$$P \approx p(x) \int_D dx$$

$$P \approx p(x) \int_D dx = p(x)V$$

Recap: Nonparametric Density Estimation

- Consider image pixels as data points (x).
- Probability that x belongs to a sub-domain D

$$P = \int_D p(x)dx$$

- If D is small, pdf is fairly constant inside, so

$$P \approx p(x) \int_D dx$$

$$P \approx p(x) \int_D dx = p(x)V$$

Recap: Nonparametric Density Estimation

- Consider image pixels as data points (x).
- Probability that x belongs to a sub-domain D

$$P = \int_D p(x)dx$$

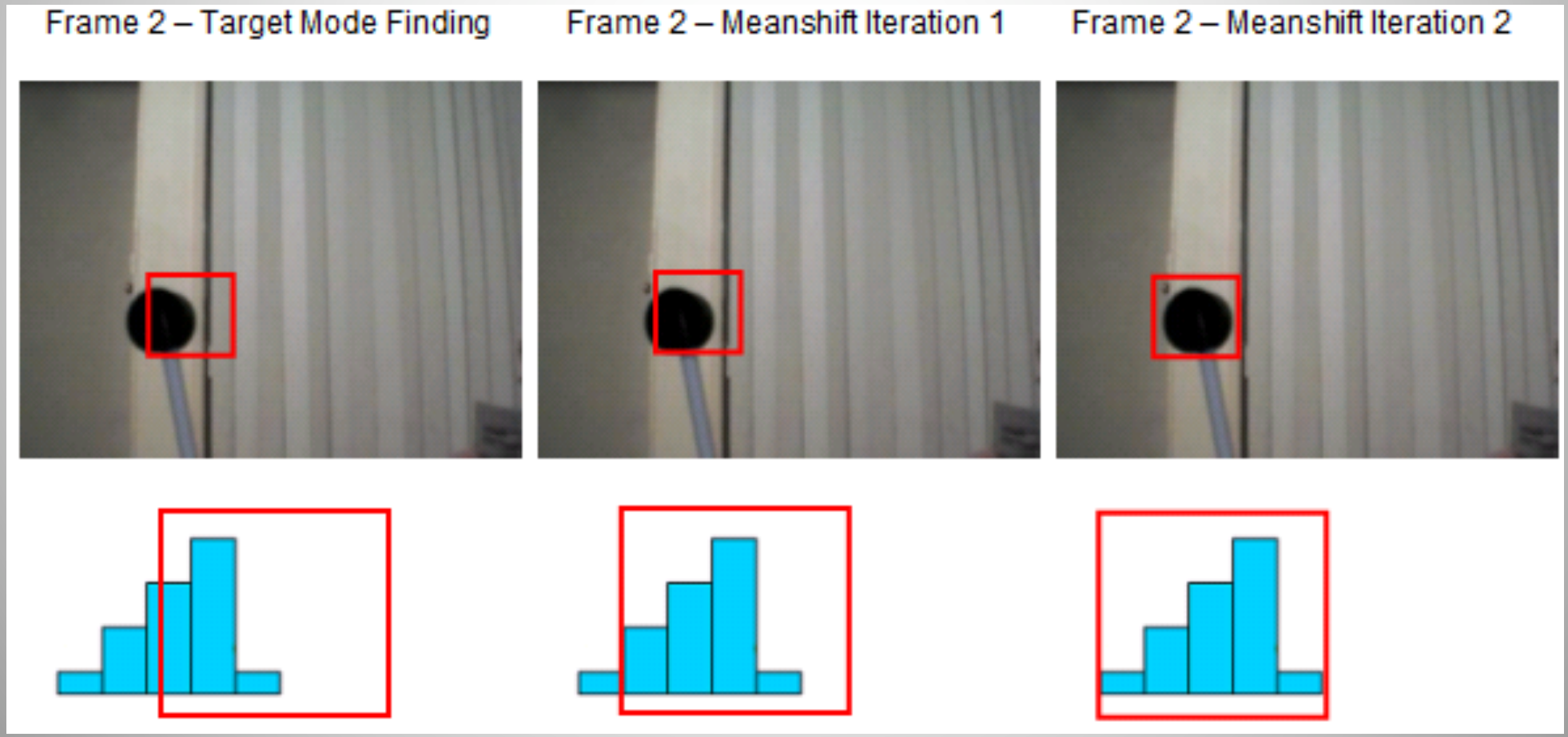
- If D is small, pdf is fairly constant inside, so

$$P \approx p(x) \int_D dx$$

$$P \approx p(x) \int_D dx = p(x)V \longrightarrow p(x) = \frac{P}{V}$$

Mean-Shift Basics

- Mean-Shift is a procedure for locating maxima of a density function given discrete data samples from that function.



Mean-Shift Basics

- Mean-Shift is a procedure for locating maxima of a density function given discrete data samples from that function.
- This is an iterative method, and we start with an initial estimation of *x*.

Mean-Shift Basics

- Mean-Shift is a procedure for locating maxima of a density function given discrete data samples from that function.
- This is an iterative method, and we start with an initial estimation of *x*.
- Let a kernel function $K(x_i-x)$ be given, determining the weight of nearby points for re-estimation of the mean.

Mean-Shift Basics

- Mean-Shift is a procedure for locating maxima of a density function given discrete data samples from that function.
- This is an iterative method, and we start with an initial estimation of *x*.
- Let a kernel function $K(x_i-x)$ be given, determining the weight of nearby points for re-estimation of the mean.
- The weighted mean of the density in the window determined by K

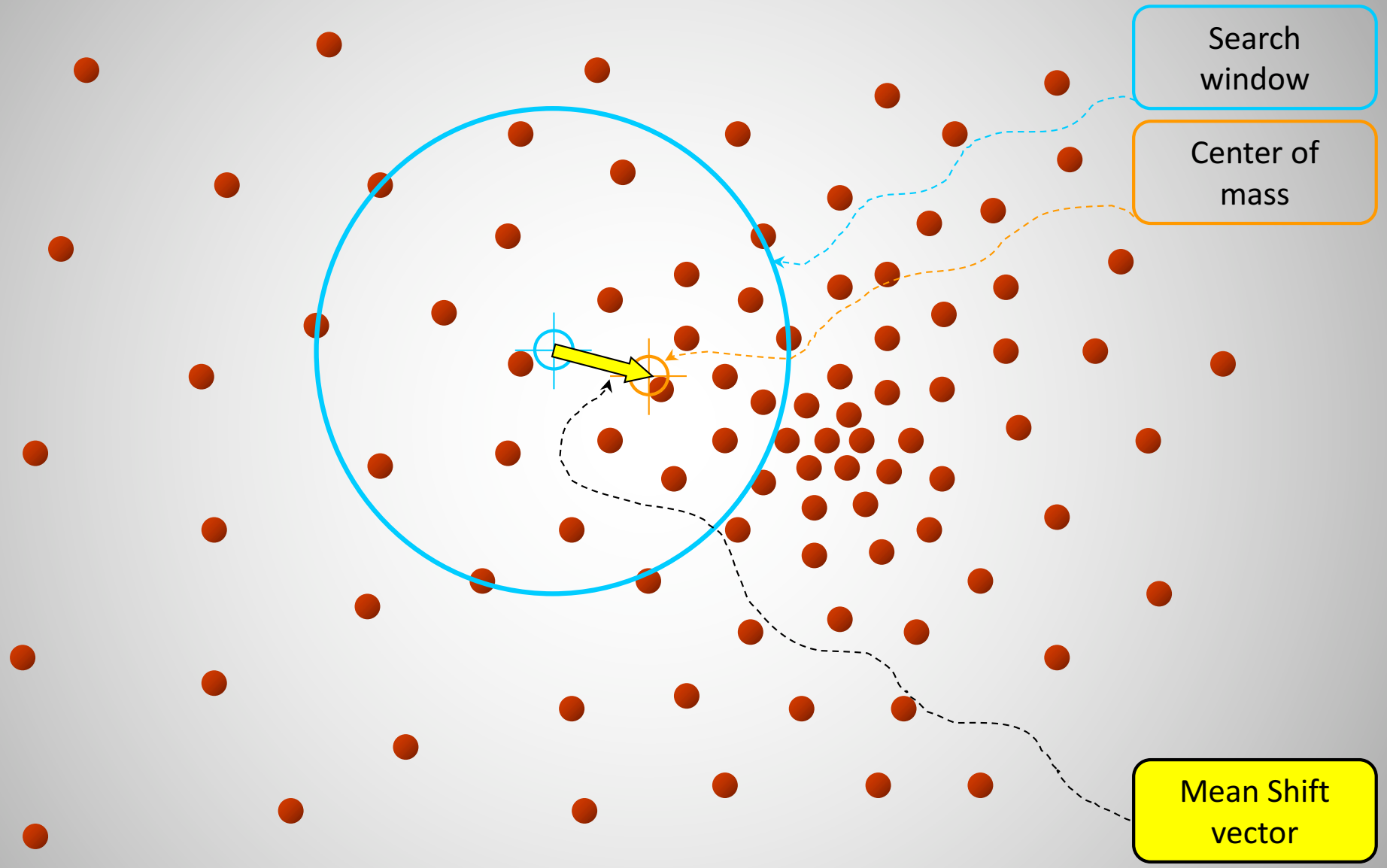
Mean-Shift Basics

- Mean-Shift is a procedure for locating maxima of a density function given discrete data samples from that function.
- This is an iterative method, and we start with an initial estimation of *x*.
- Let a kernel function $K(x_i-x)$ be given, determining the weight of nearby points for re-estimation of the mean.
- The weighted mean of the density in the window determined by K

$$m(x) = \frac{\sum_{x_i \in N(x)} K(x_i - x) x_i}{\sum_{x_i \in N(x)} K(x_i - x)}$$

N(x): neighborhood of x
m(x) – x: mean shift

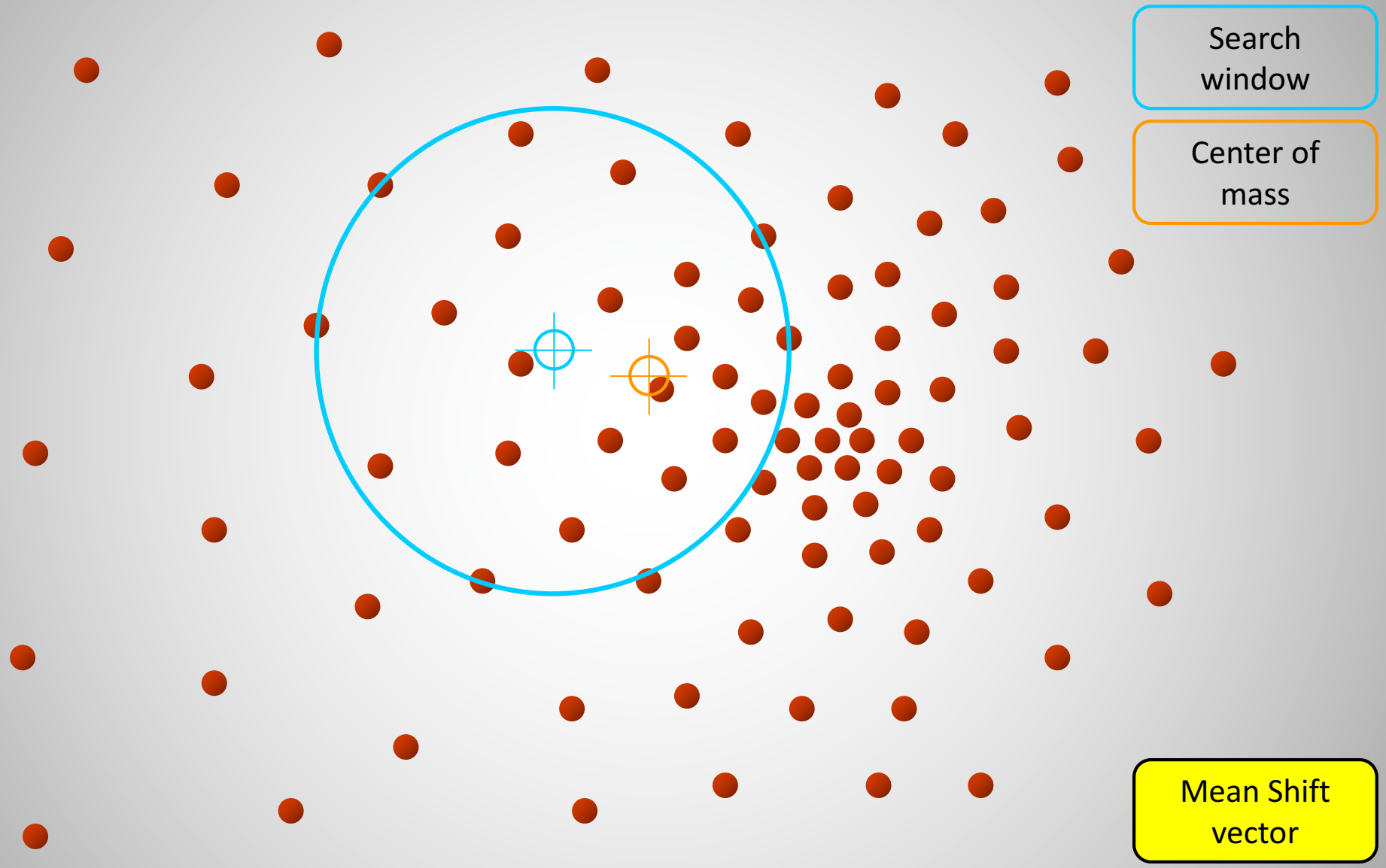
Mean-Shift



FIND Ddensest REGION!

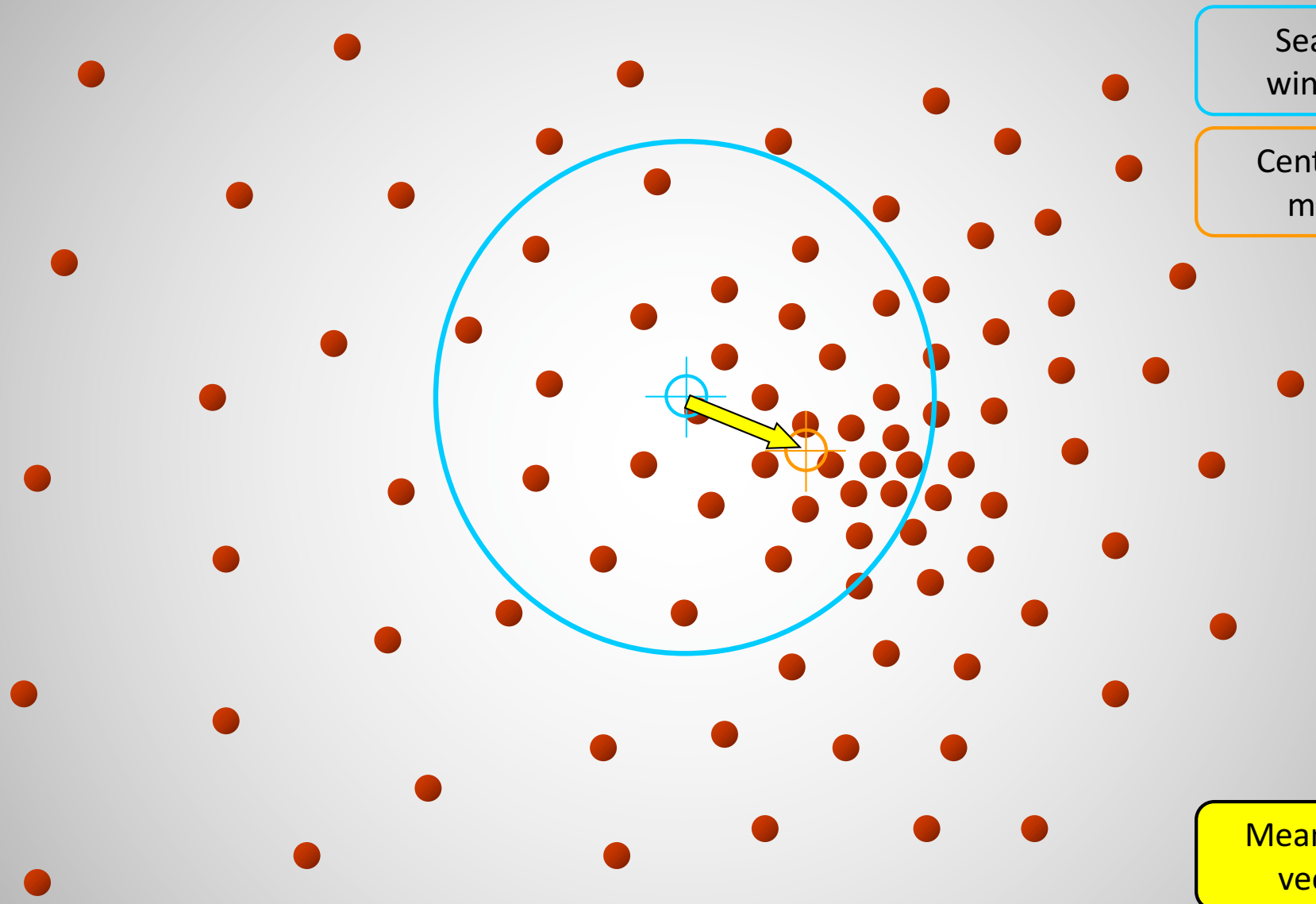
Slide by Y. Ukrainitz & B. Sarel

Mean-Shift



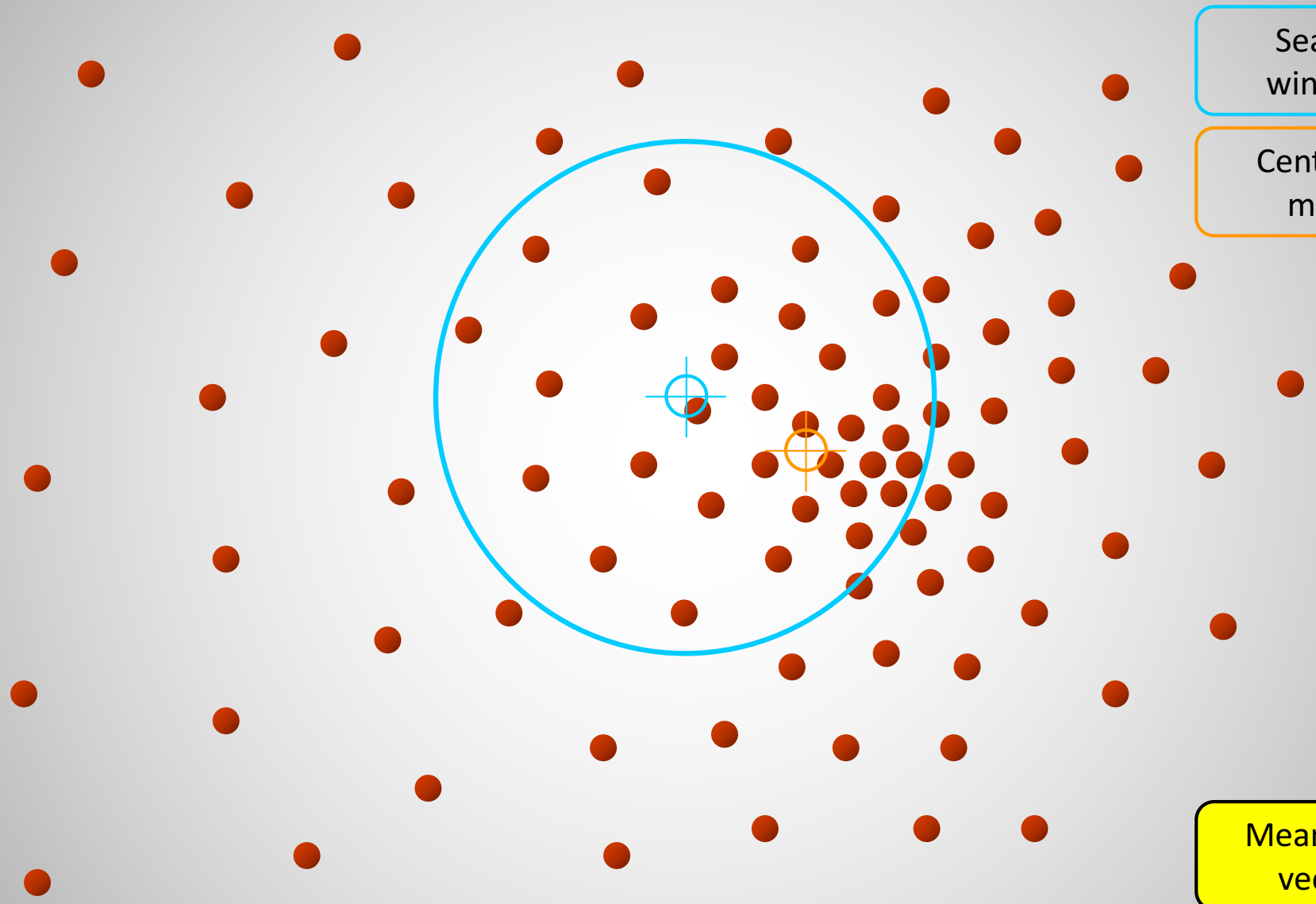
Slide by Y. Ukrainitz & B. Sarel

Mean-Shift



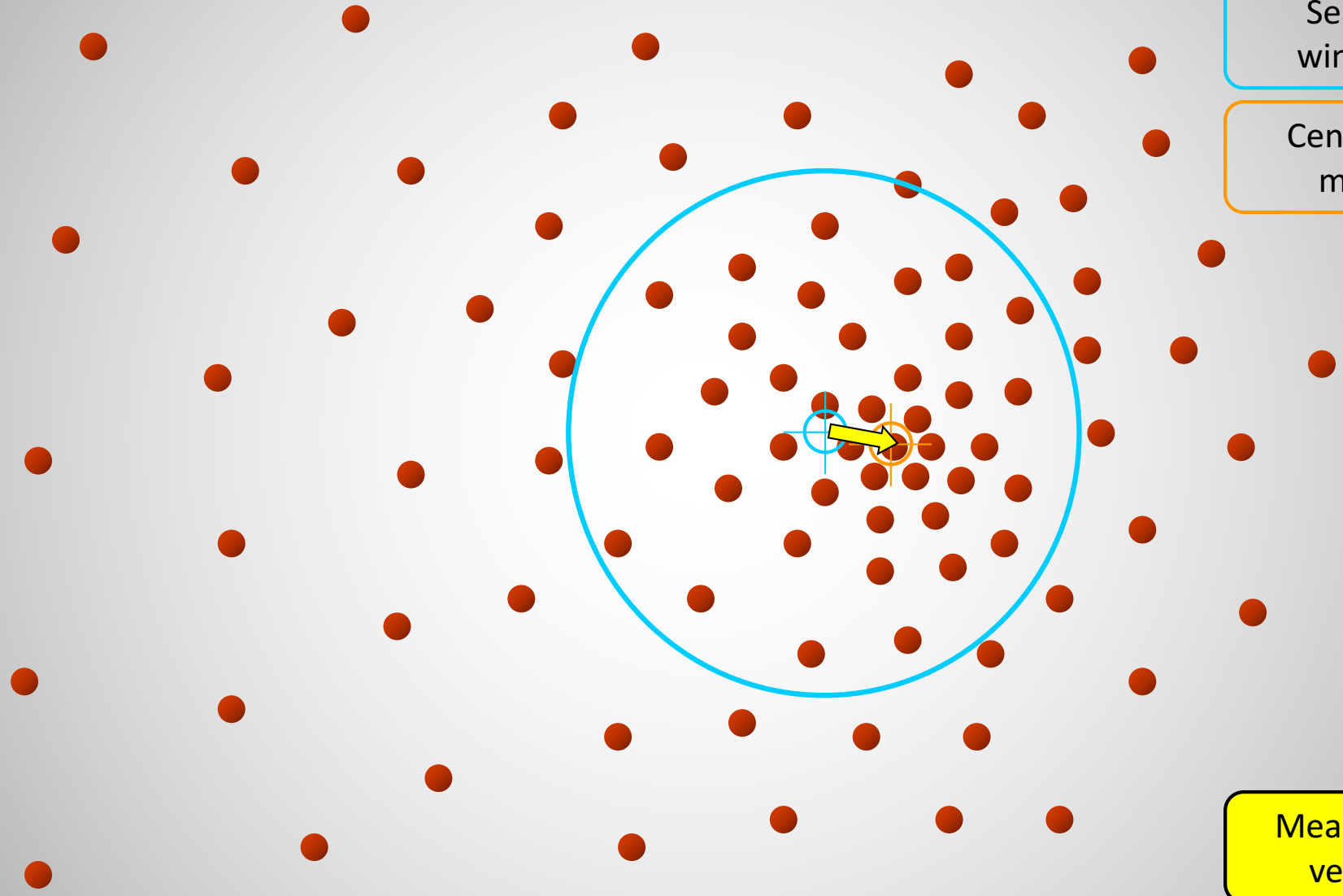
Slide by Y. Ukrainitz & B. Sarel

Mean-Shift



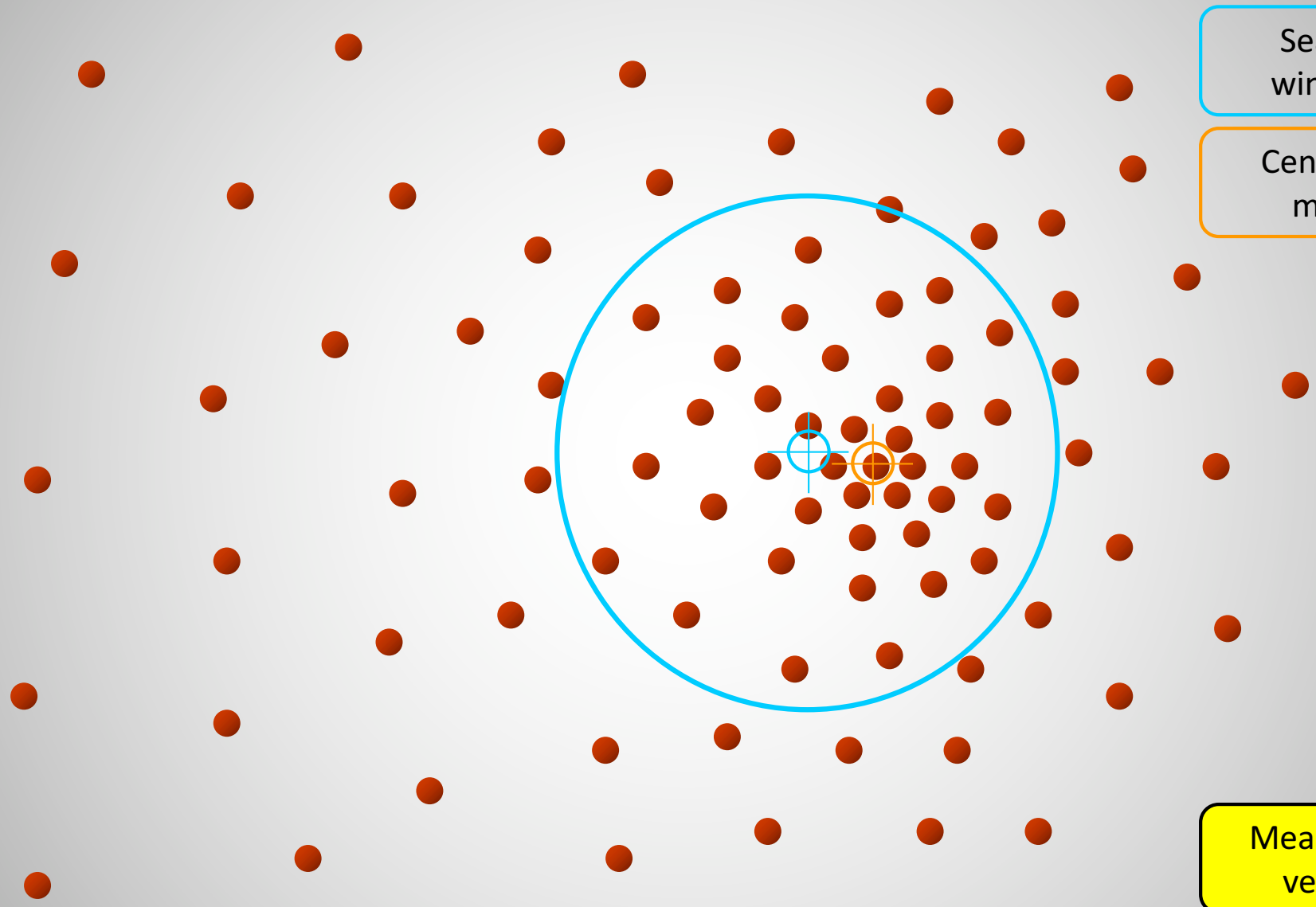
Slide by Y. Ukrainitz & B. Sarel

Mean-Shift



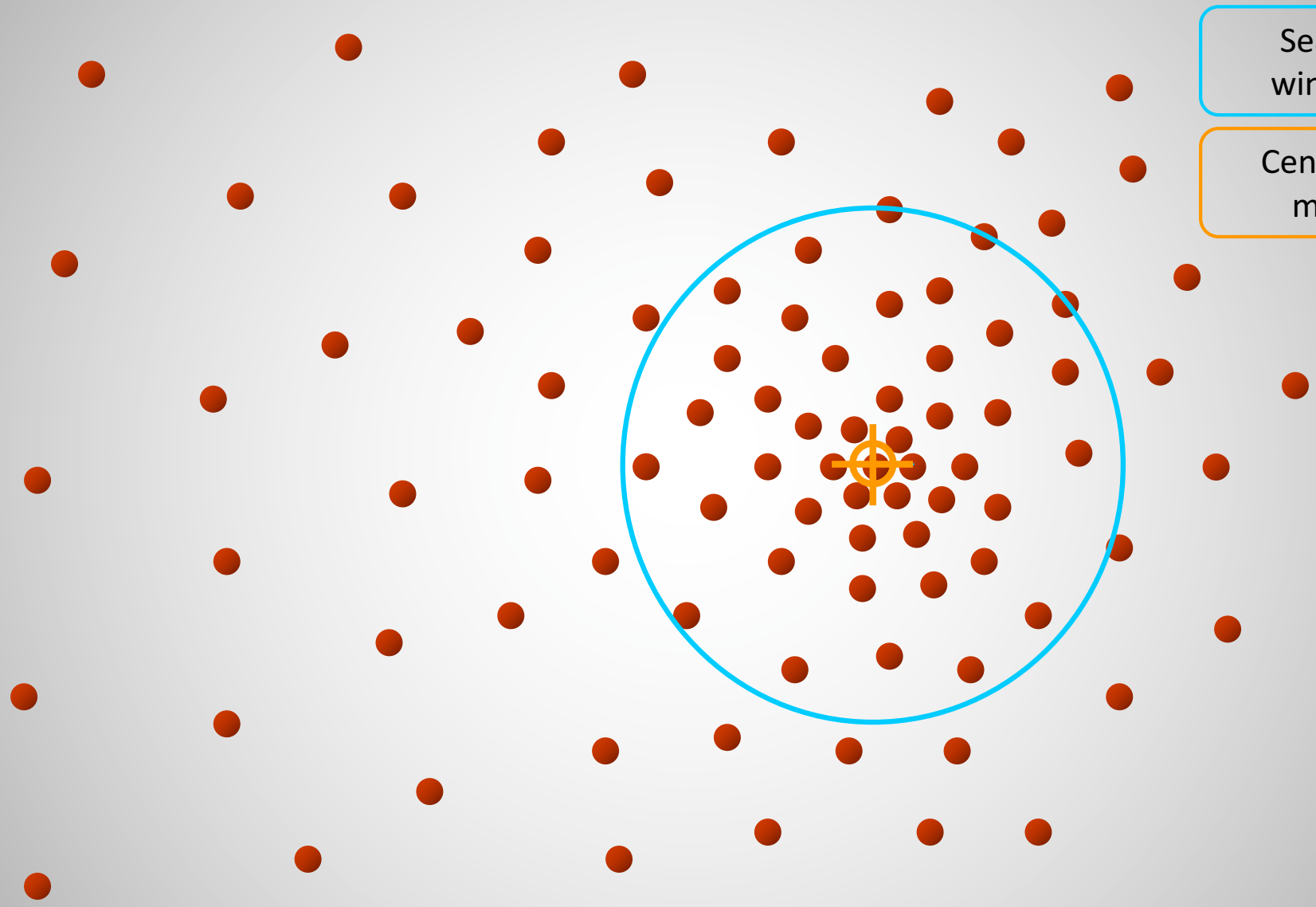
Slide by Y. Ukrainitz & B. Sarel

Mean-Shift



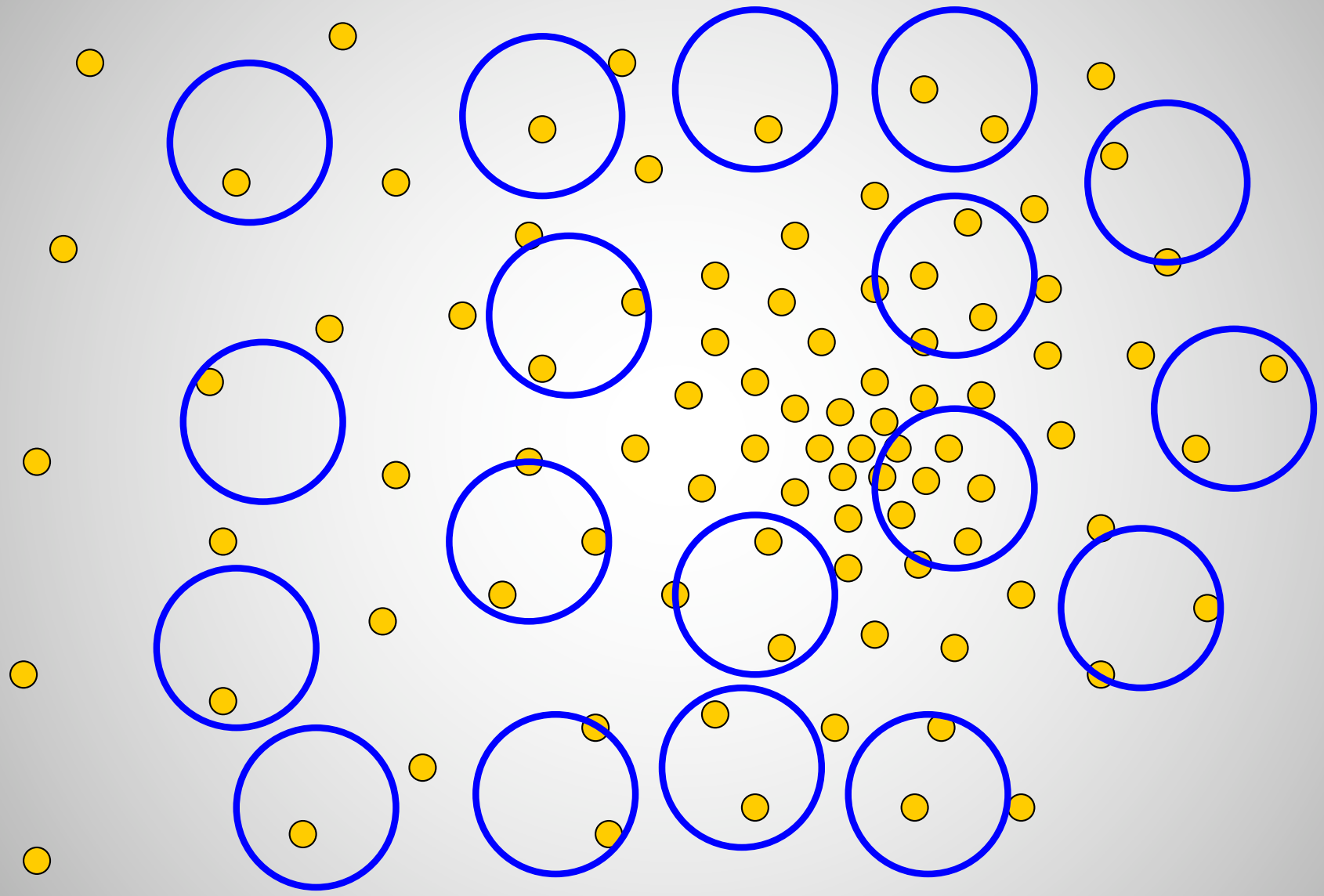
Slide by Y. Ukrainitz & B. Sarel

Mean-Shift



Slide by Y. Ukrainitz & B. Sarel

Practical Knowledge

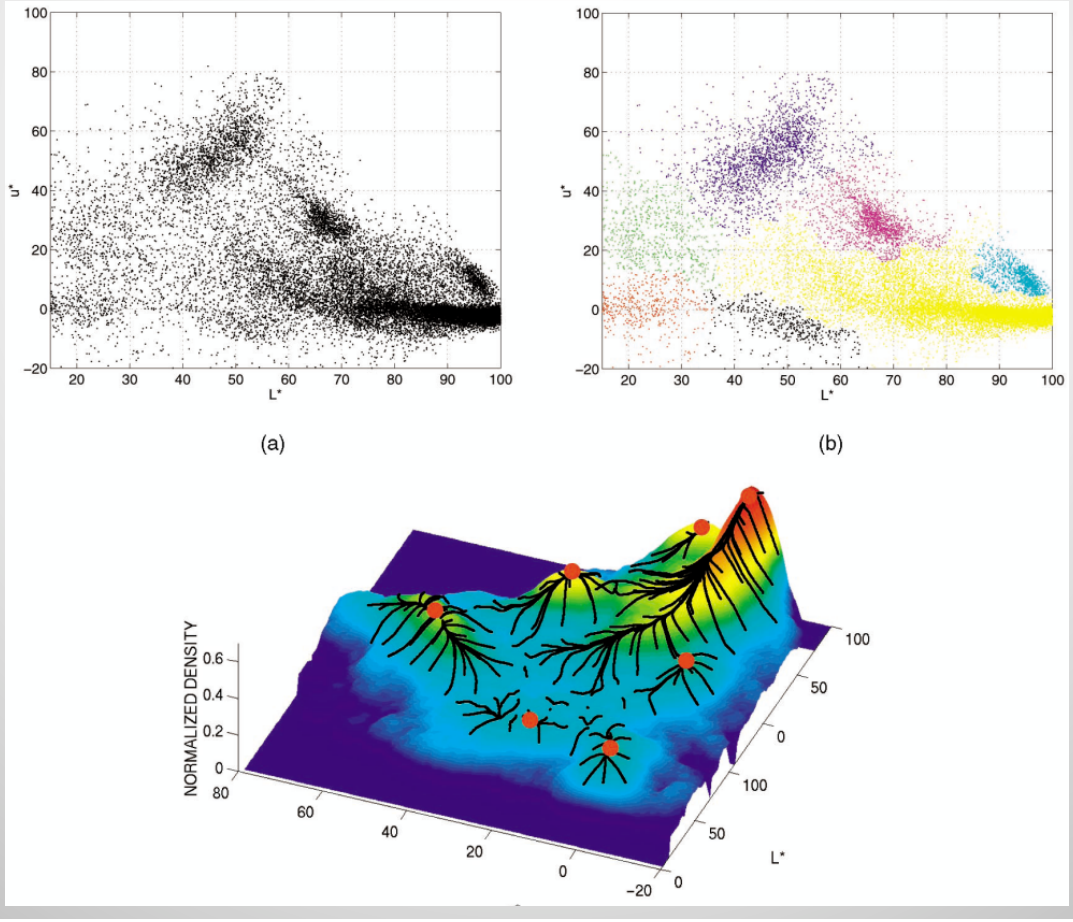


Tessellate the space with windows

Run the procedure in parallel

Mean-Shift Clustering

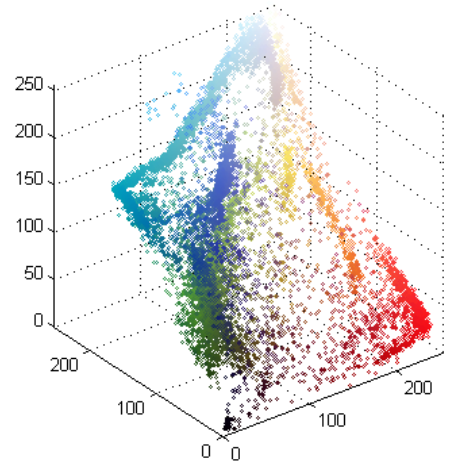
- Find features (color, gradients, texture, etc)
- Initialize windows at individual feature points
- Perform mean shift for each window until convergence
- Merge windows that end up near the same “peak” or mode



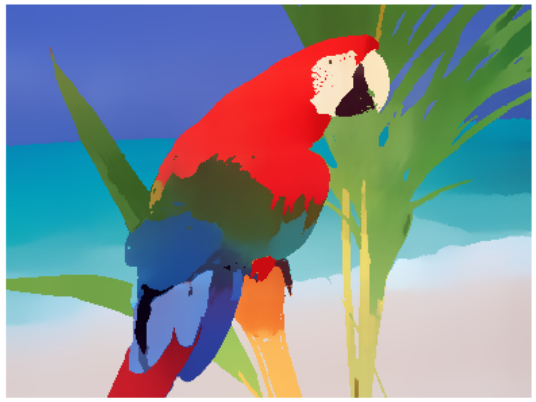
Mean-Shift Clustering

input image

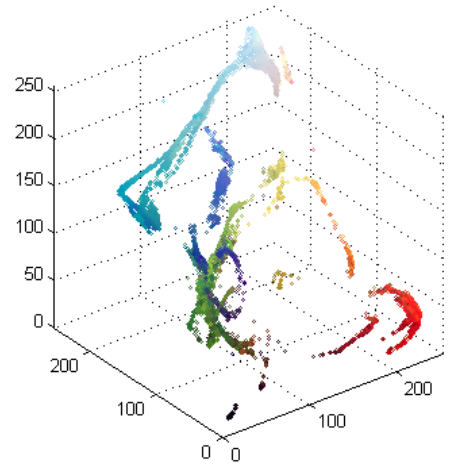
Pixel Distribution Before Meanshift



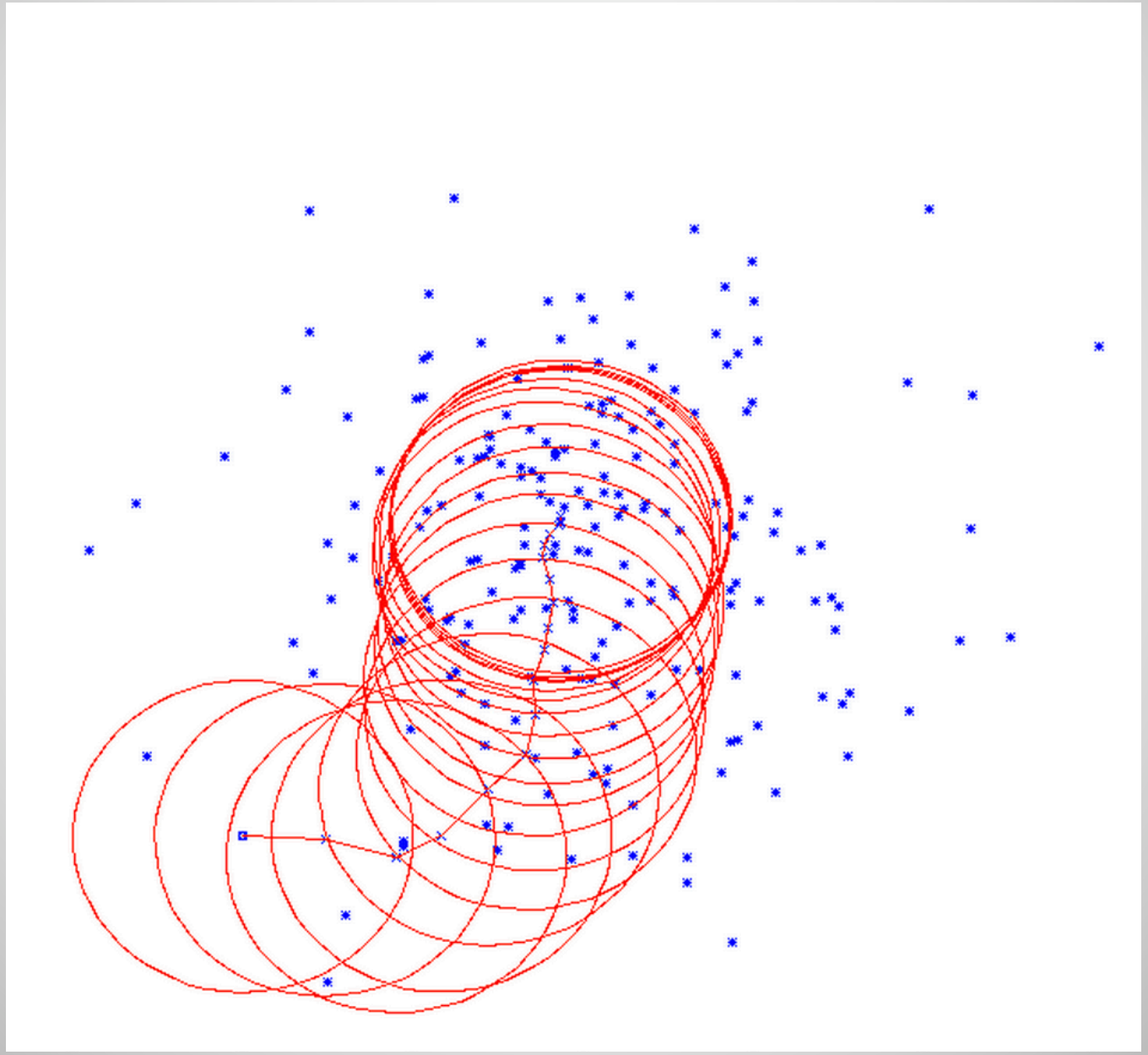
output image



Pixel Distribution After Meanshift

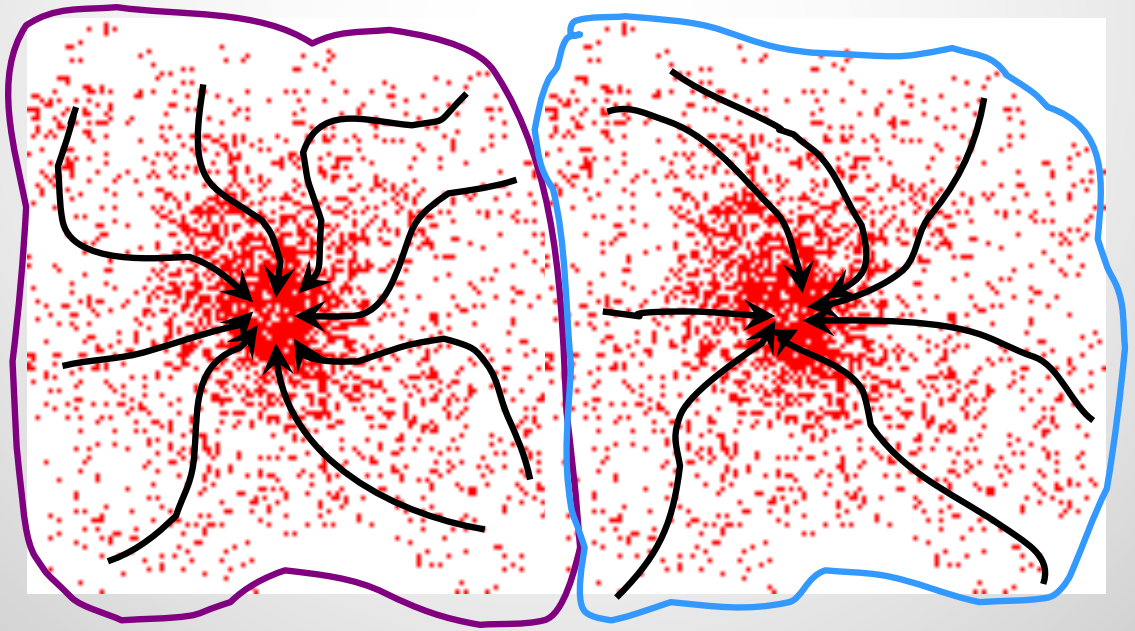


Trajectory: x, m(x), (m(m(x)), ...



Mean-Shift Clustering

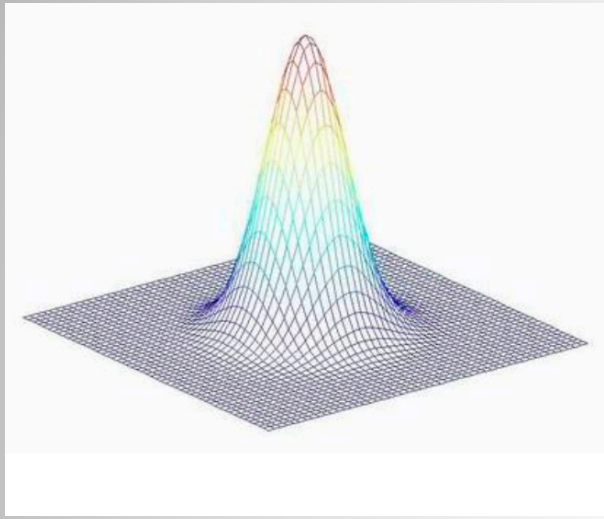
- Gradient ascent requires computing a gradient of the density function.
- Density function is only approximated by a distribution of samples, the gradient can only be approximated as well !!!
- This is done by a kernel window estimator.



Slide by Y. Ukrainitz & B. Sarel

Practical Knowledge-2

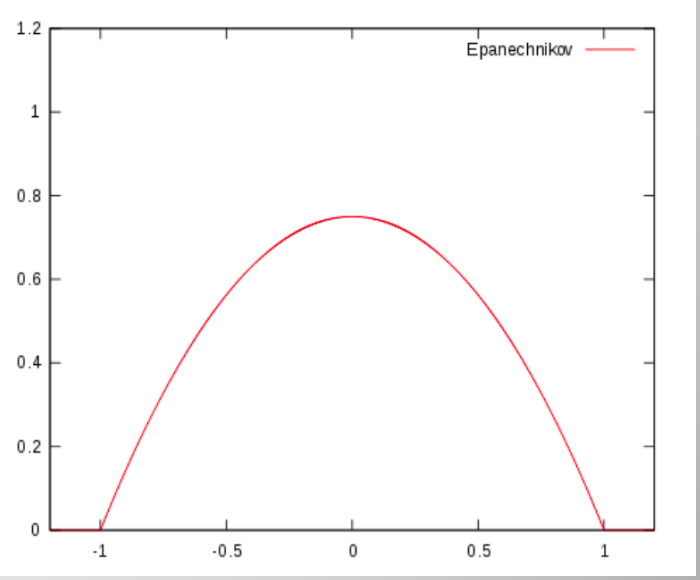
- The most popular kernel is Gaussian! (i.e., K)



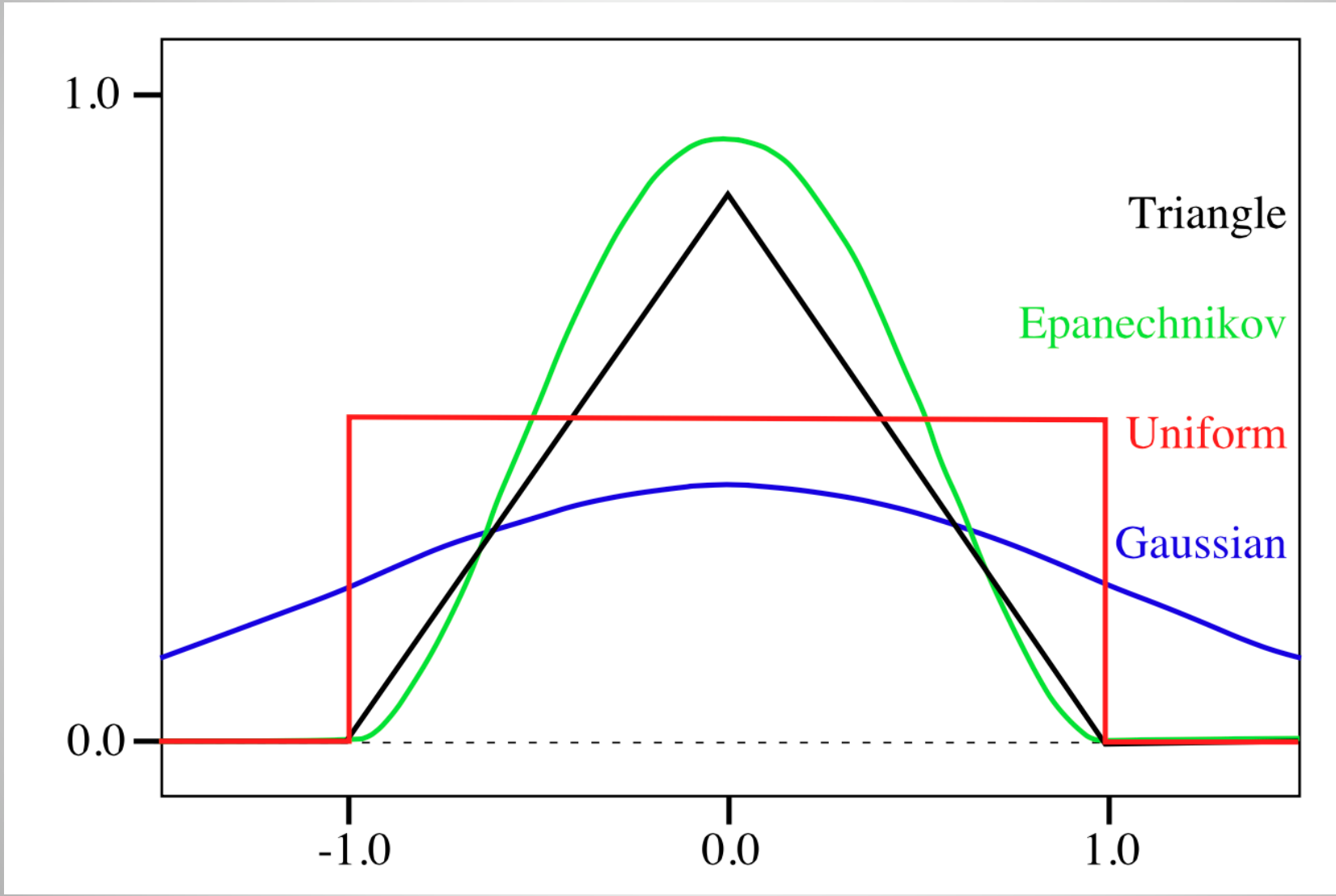
$$K(x) = exp(-|x|^2)$$

- Epanechikov kernel

$$K(x) = \begin{cases} 3(1-x^2)/4 & \text{if } |x| \leq 1 \\ 0 & \text{else} \end{cases}$$



Practical Knowledge-3



Parameters of the Mean-Shift Segmentation

- h_s: spatial resolution parameter
  - Affects the smoothing, connectivity of segments
- h_r: range resolution parameter
  - Affects the number of segments
- M: size of smallest segment
  - Should be choosen based on size of noisy patches

Parameters of the Mean-Shift Segmentation

Original

(hs,hr)=(8,4)

(hs,hr)=(8,7)

Ex: Mean-Shift Segmentation

Ex: Mean-Shift Segmentation

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

Slide Credits and References

- R.Klette, Concise Computer Vision, 2014.
- Y. Ukrainitz & B. Sarel, Weizmann
- Thu Huong Nhuyen, TU Dresden
- Jens N. Kaftan and Andr`e A. Bell and Til Aach . Mean Shift Segmentation Evaluation of Optimization Techniques
- Cheng, Y. (1995). Mean Shift, Mode Seeking, and Clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(8):790–799.
- KEINOSUKE FUKUNAGA, AND LARRY D. HOSTETLER. The Estimation of the Gradient of a Density Function, with Applications in Pattern – Recognition. IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. IT-21, NO. 1, JANUARY 1975
- Comaniciu, D. and Meer, P. (1997). Robust Analysis of Feature Spaces: Color Image Segmentation.
- In IEEE Conference on Computer Vision and Pattern Recognition. CVPR 1997, pages 750–755.
- Comaniciu, D. and Meer, P. (1999). Mean Shift Analysis an Applications. In International Conference on Computer Vision. ICCV 1999, volume 2, pages 1197–1203.
- Comaniciu, D. and Meer, P. (2002). Mean Shift: A Robust Approach Toward Feature Space Analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(5):603–619.
- K. Fukunaga, L.D. Hostetler, The estimation of the gradient of a density function, with applications in pattern recognition, IEEE Trans. Inf. Theory 21 (1975) 3240.
- Yiping Hong , Jianqiang Yi and Dongbin Zhao . Improved mean shift segmentation approach for natural images