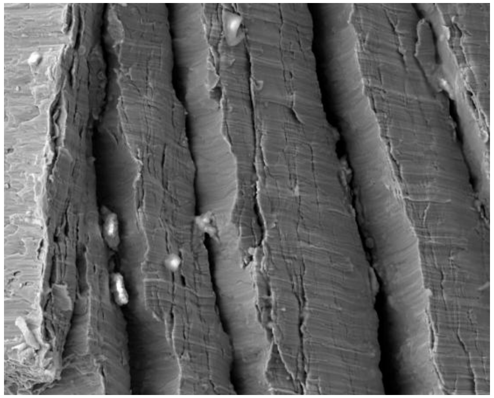
Bildverarbeitung und Mustererkennung

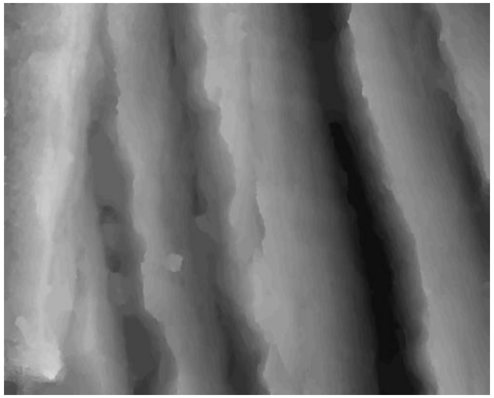
Image Processing and Pattern Recognition

710.080 2VO

710.081 1KU

$$\int_{\Omega} f(x, u(x), \nabla u(x))dx \Leftrightarrow \sup_{\phi \in K} \int_{\Omega \times R} \phi \cdot D\mathbf{1}_u$$





Bildverarbeitung und Mustererkennung

Image Processing and Pattern Recognition

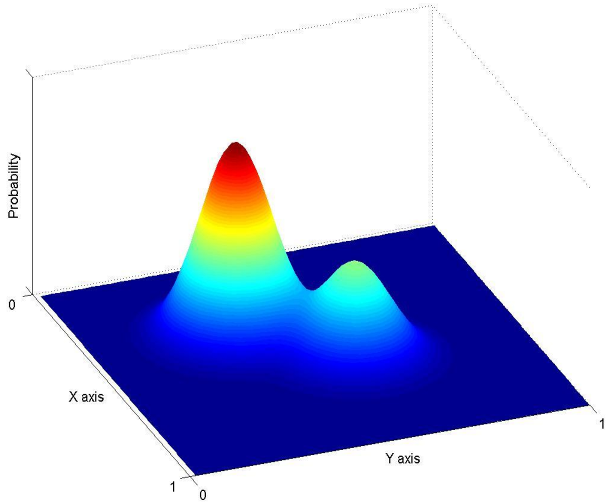
- Mean Shift

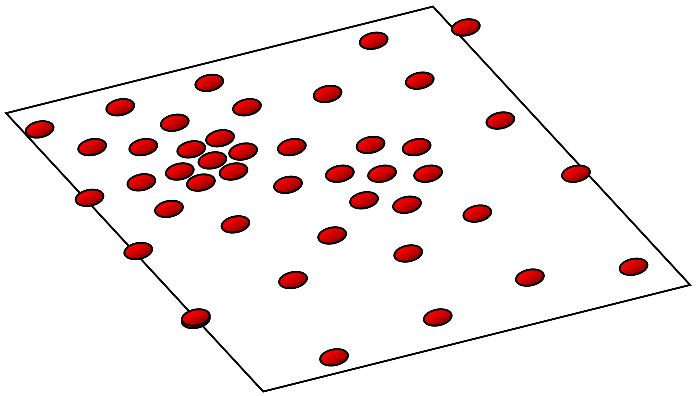
Mean Shift

- Analysis of the feature space
- Features: x,y-position, gray value, color, texture, motion
- Detects modes (local maxima) in the feature space
- Many Applications
  - Denoising
  - Segmentation
  - Tracking

Parametric Density Estimation

The data points are sampled from an underlying parametric PDF
E.g. Normal Distribution





Assumed Underlying PDF

Real Data Samples

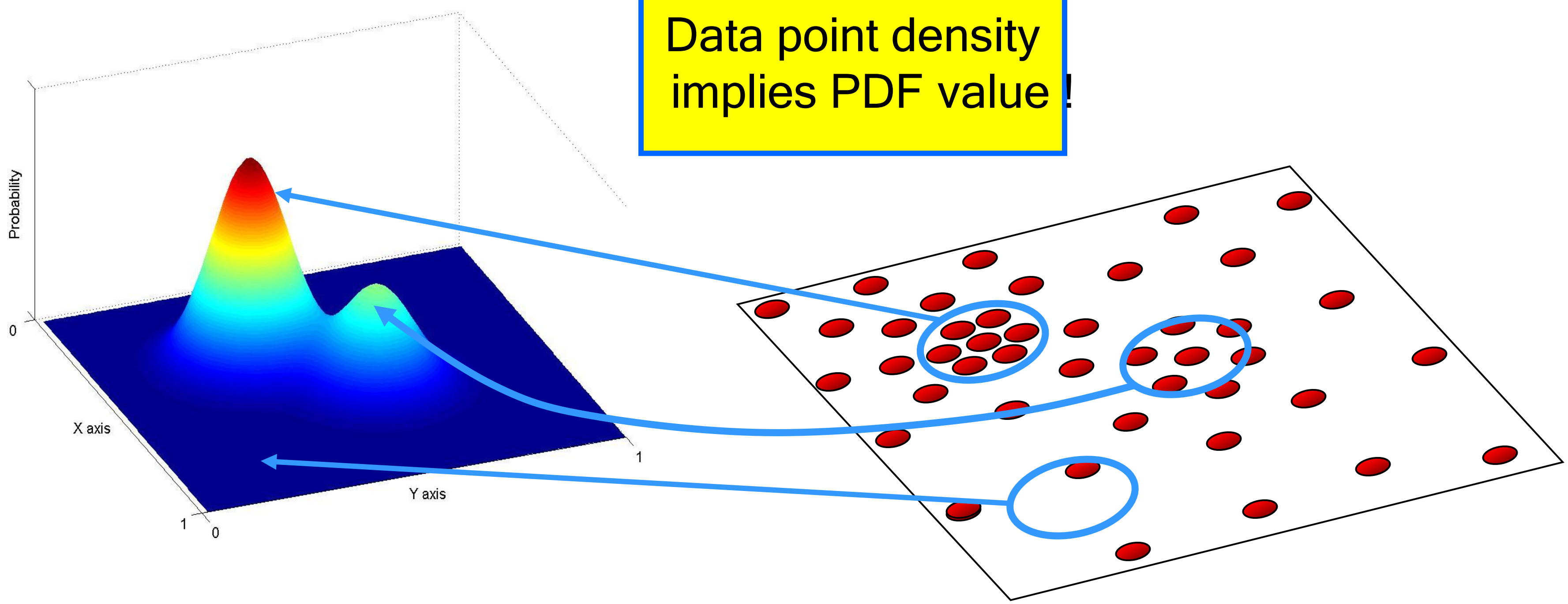
$$P(x_i) \sim \omega \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{(x_i-\mu_1)^2}{2\sigma_1^2}} + (1-\omega) \frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{(x_i-\mu_2)^2}{2\sigma_2^2}}$$

(Mixture of 2 Gaussians)

Non-Parametric Density Estimation

The data points are sampled from an unknown underlying PDF
Only Assumption: Resolution of Data Points

Data point density implies PDF value !

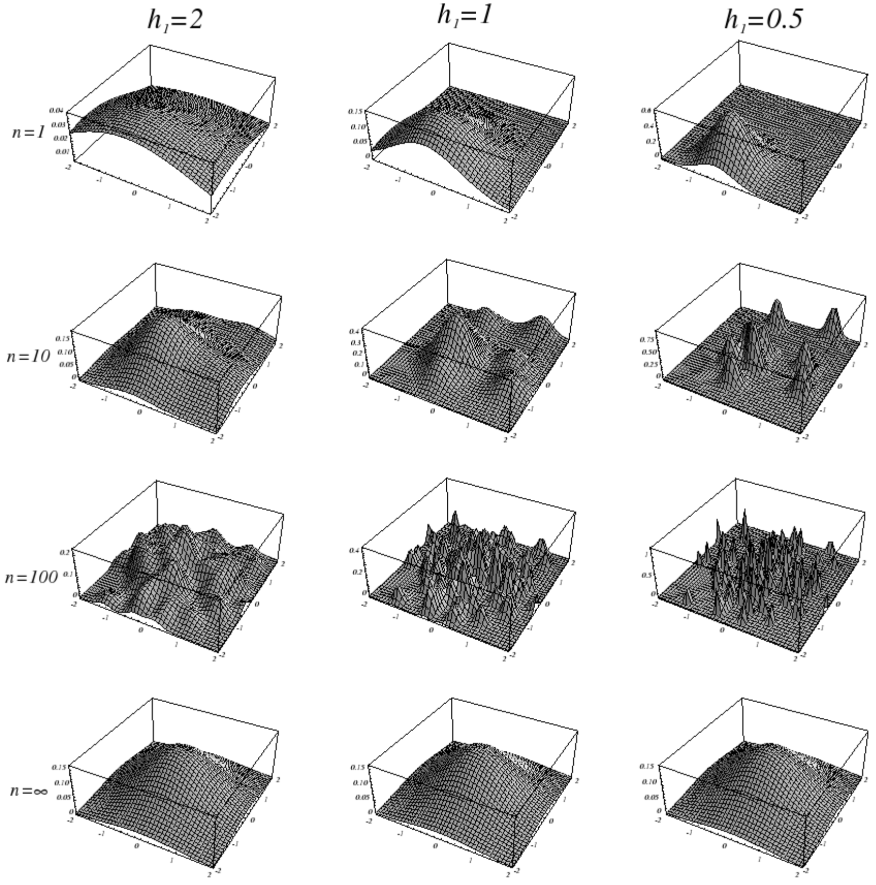


No Assumption about the underlying PDF

Real Data Samples

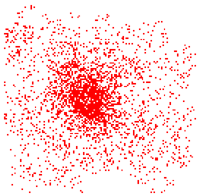
Non-Parametric Density Estimation

- Example with different number of samples and bandwidths



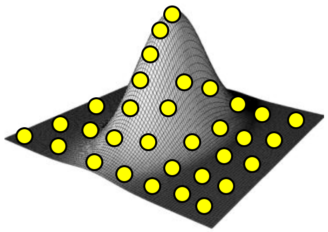
What is Mean Shift ?

- Finds modes in a set of data samples, manifesting an underlying probability density function (PDF)



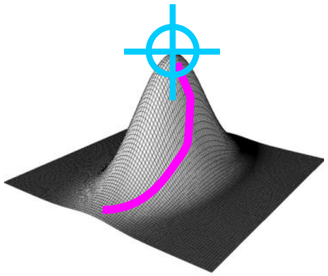
Data

Non-parametric
Density Estimation



Discrete PDF Representation

Non-parametric
Density GRADIENT Estimation
(Mean Shift)

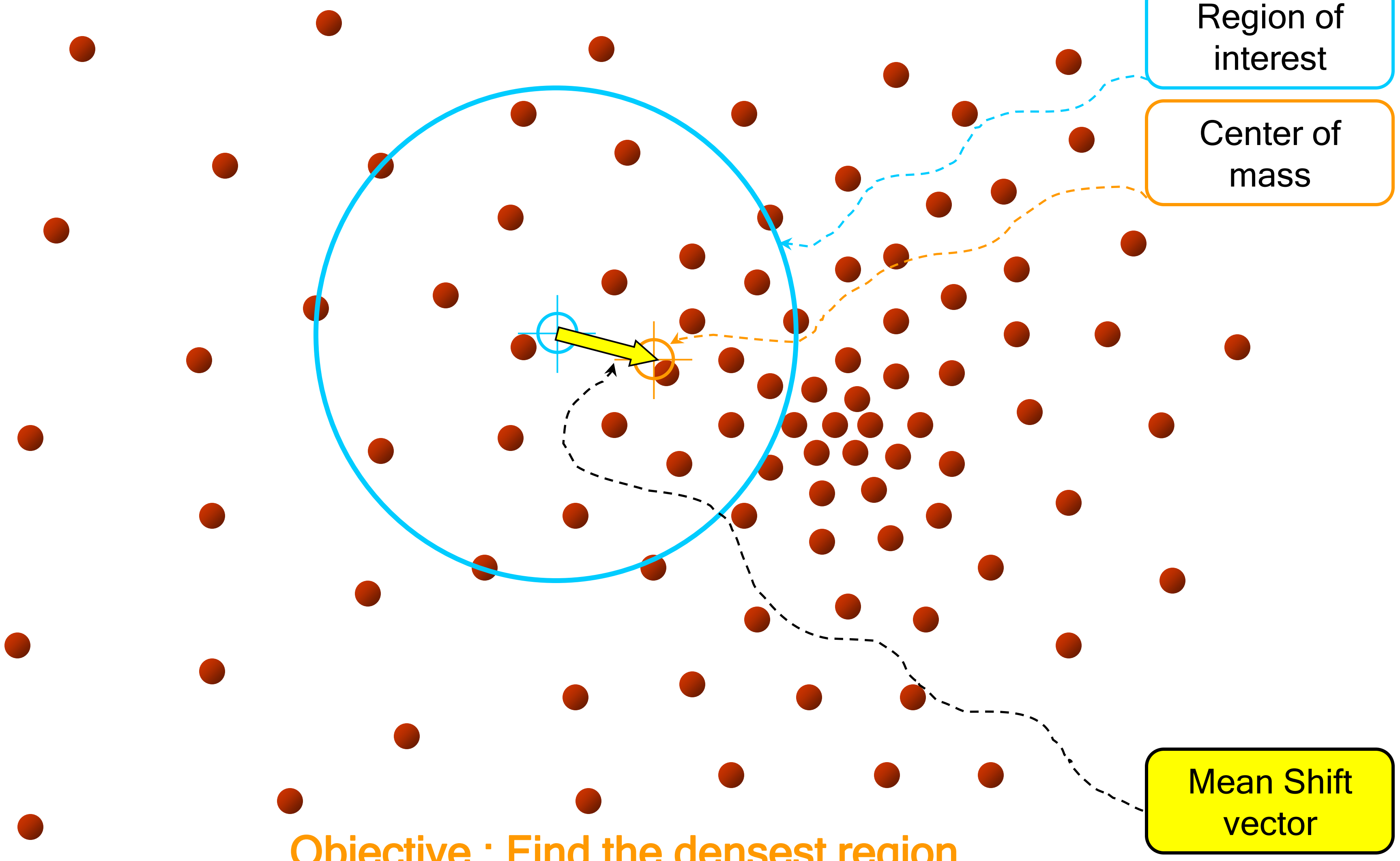


PDF Analysis

Mean Shift

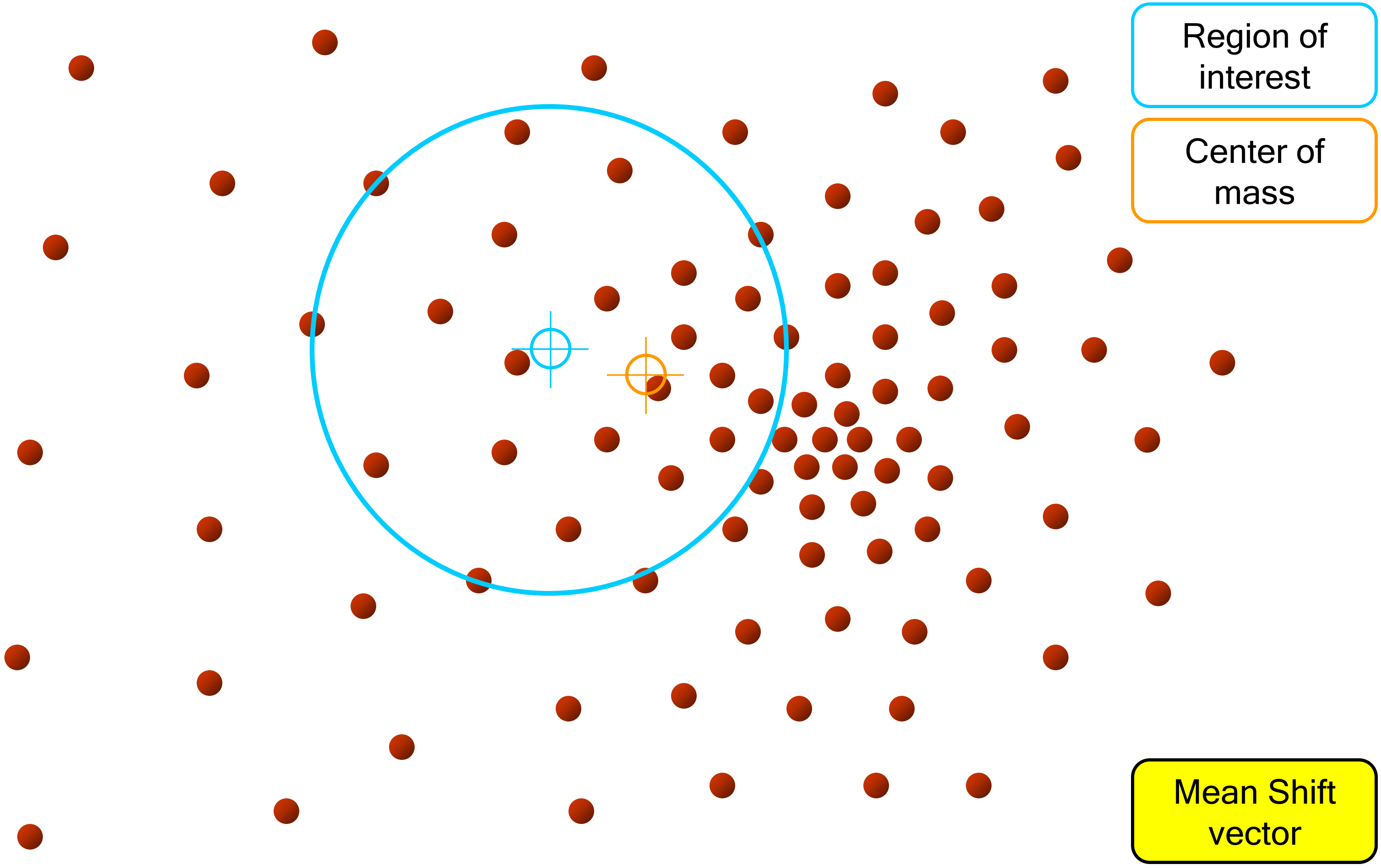
- Derivation of the mean shift procedure from kernel density estimation

Mean Shift Illustration



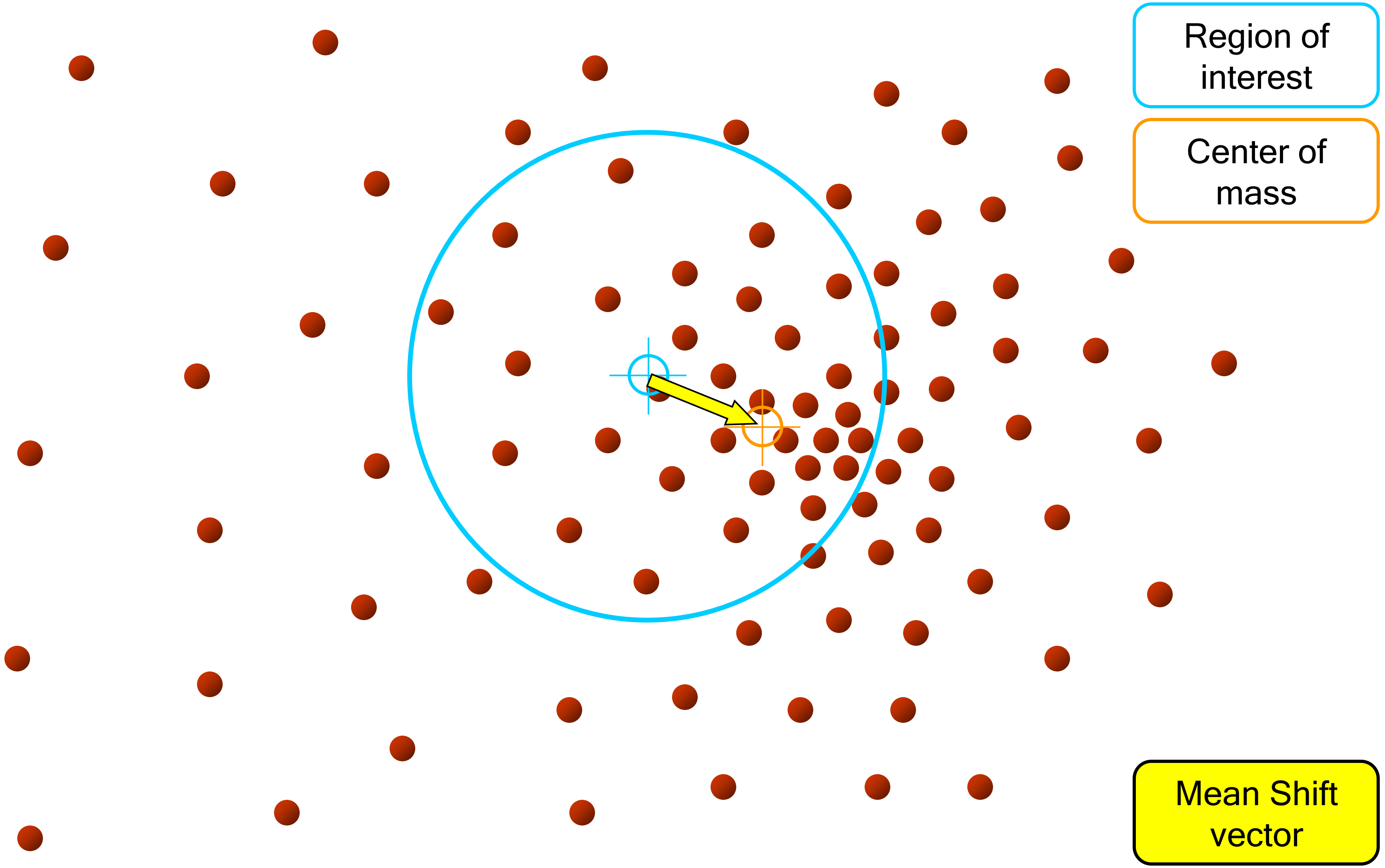
Objective : Find the densest region

Mean Shift Illustration



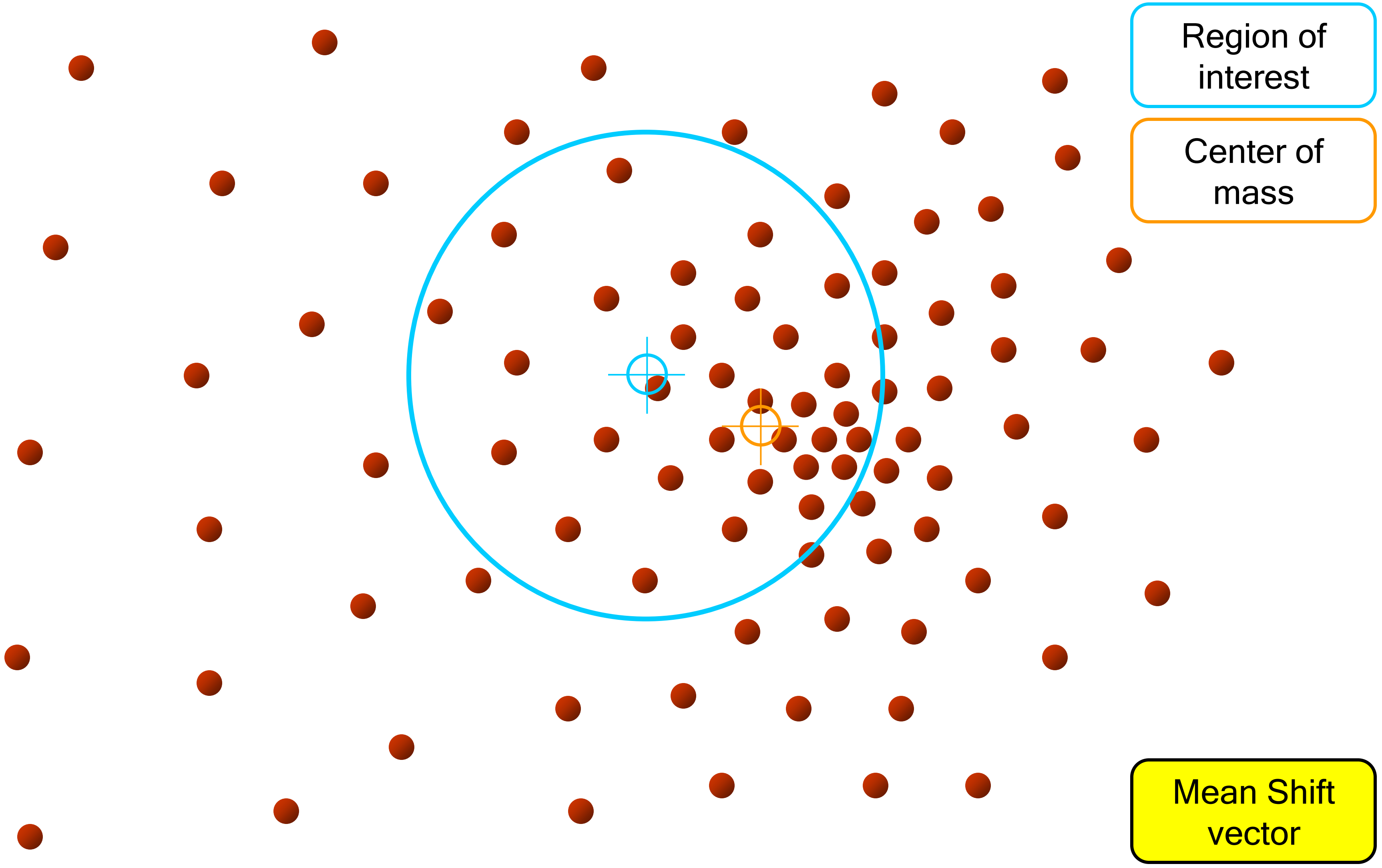
Objective : Find the densest region

Mean Shift Illustration



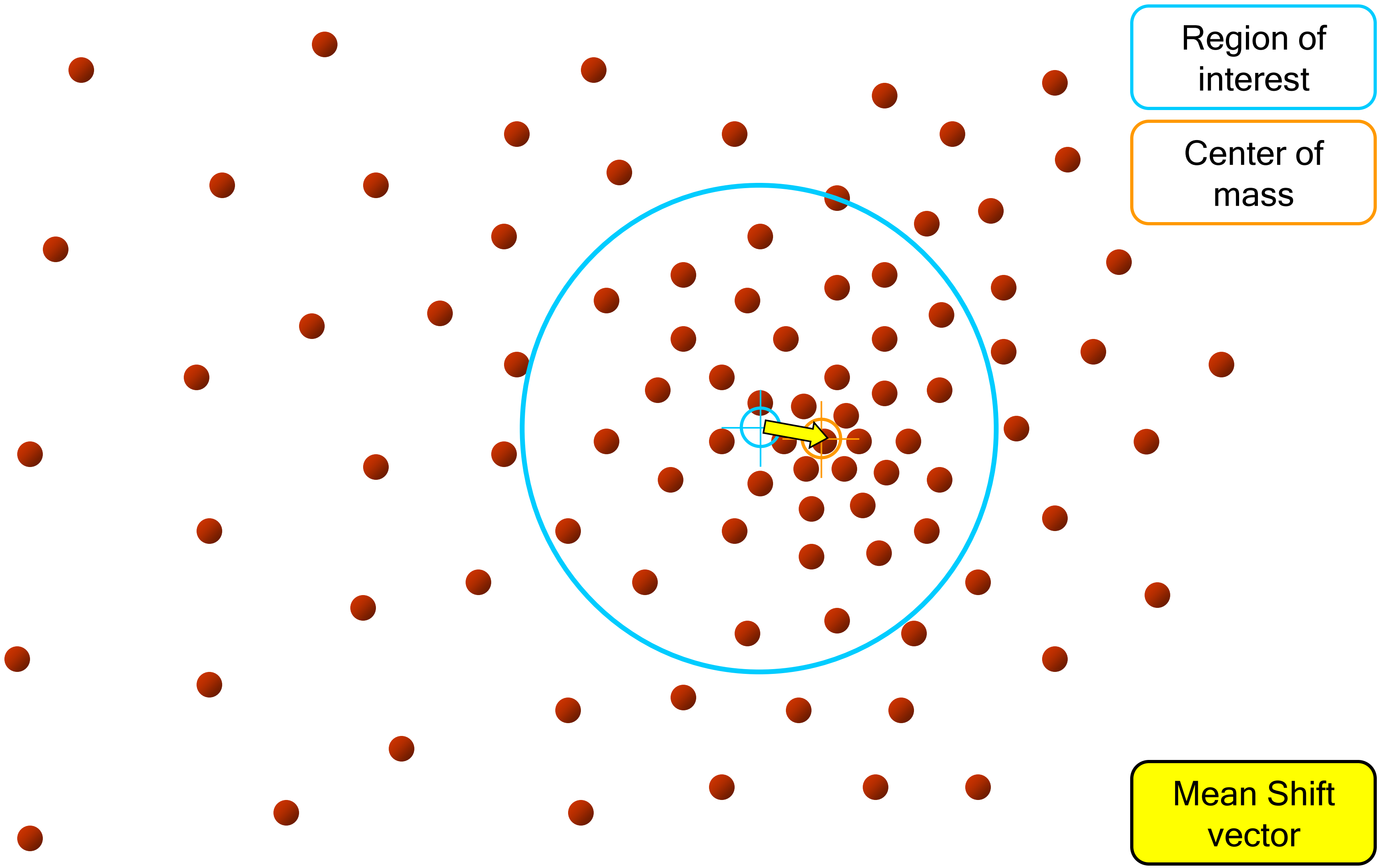
Objective : Find the densest region

Mean Shift Illustration



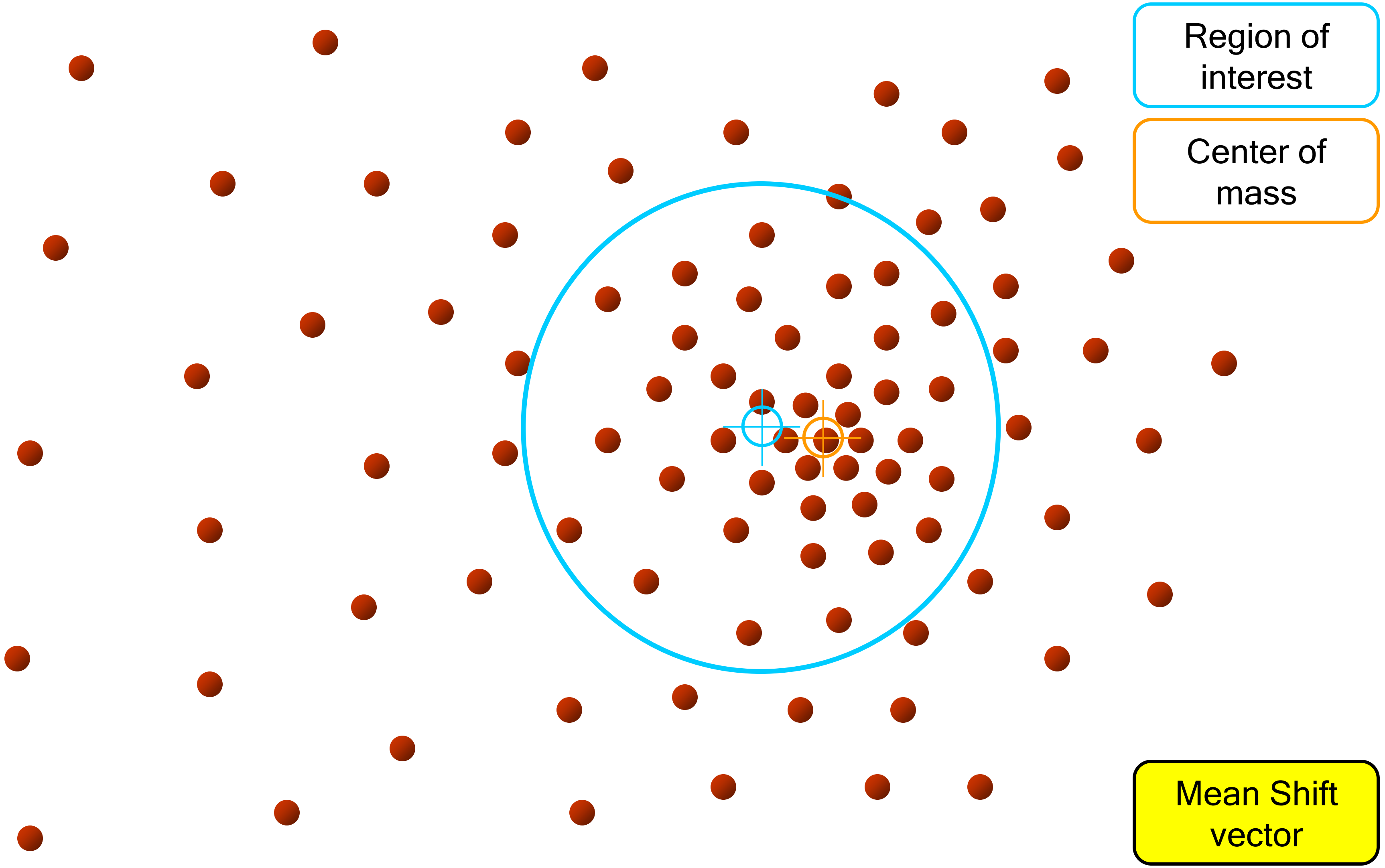
Objective : Find the densest region

Mean Shift Illustration



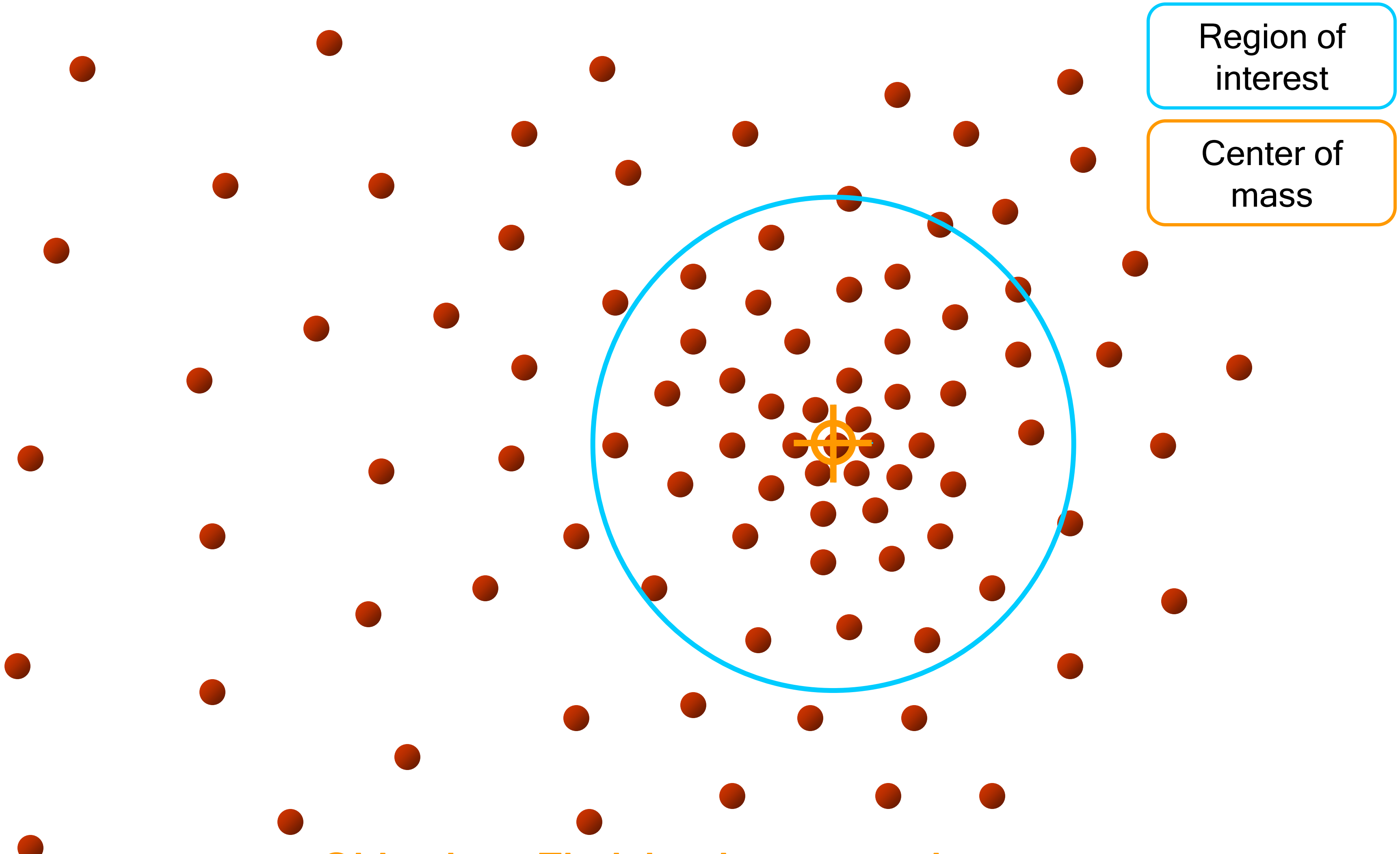
Objective : Find the densest region

Mean Shift Illustration



Objective : Find the densest region

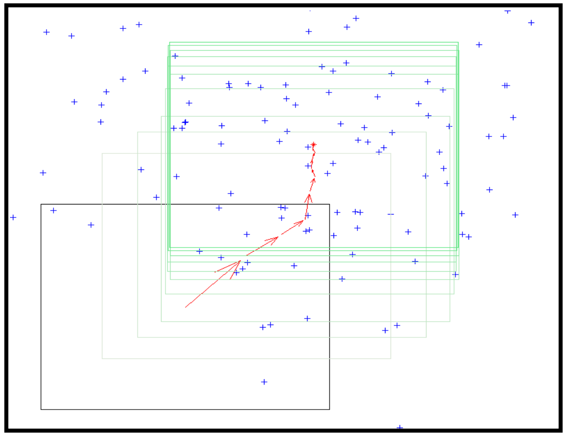
Mean Shift Illustration



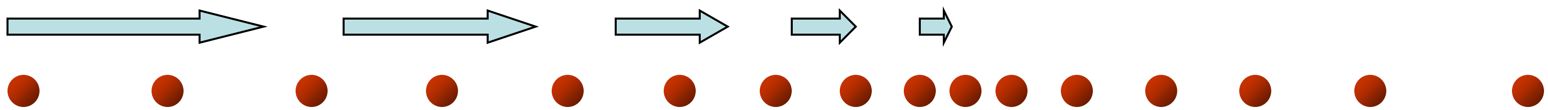
Objective : Find the densest region

Mean Shift Prozedur

1. Choose a search window size.
2. Choose the initial location of the search window.
3. Compute the mean location (centroid of the data) in the search window.
4. Center the search window at the mean location computed in Step 3.
5. Repeat Steps 3 and 4 until convergence.



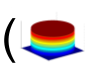
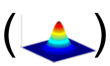
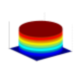
Mean Shift Properties



- Automatic convergence speed – the mean shift vector size depends on the gradient itself.
- Near maxima, the steps are small and refined

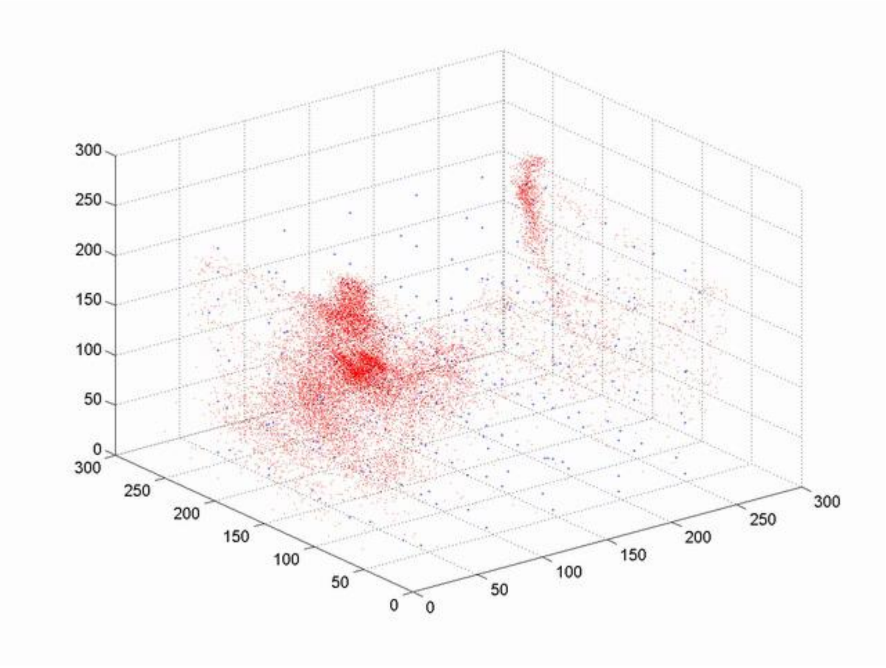
} **Adaptive** Gradient Ascent

- Convergence is guaranteed for infinitesimal steps only ➔ infinitely convergent,

- For Uniform Kernel (), convergence is achieved in a finite number of steps
- Normal Kernel () exhibits a smooth trajectory, but is slower than Uniform Kernel ().

Color Feature space

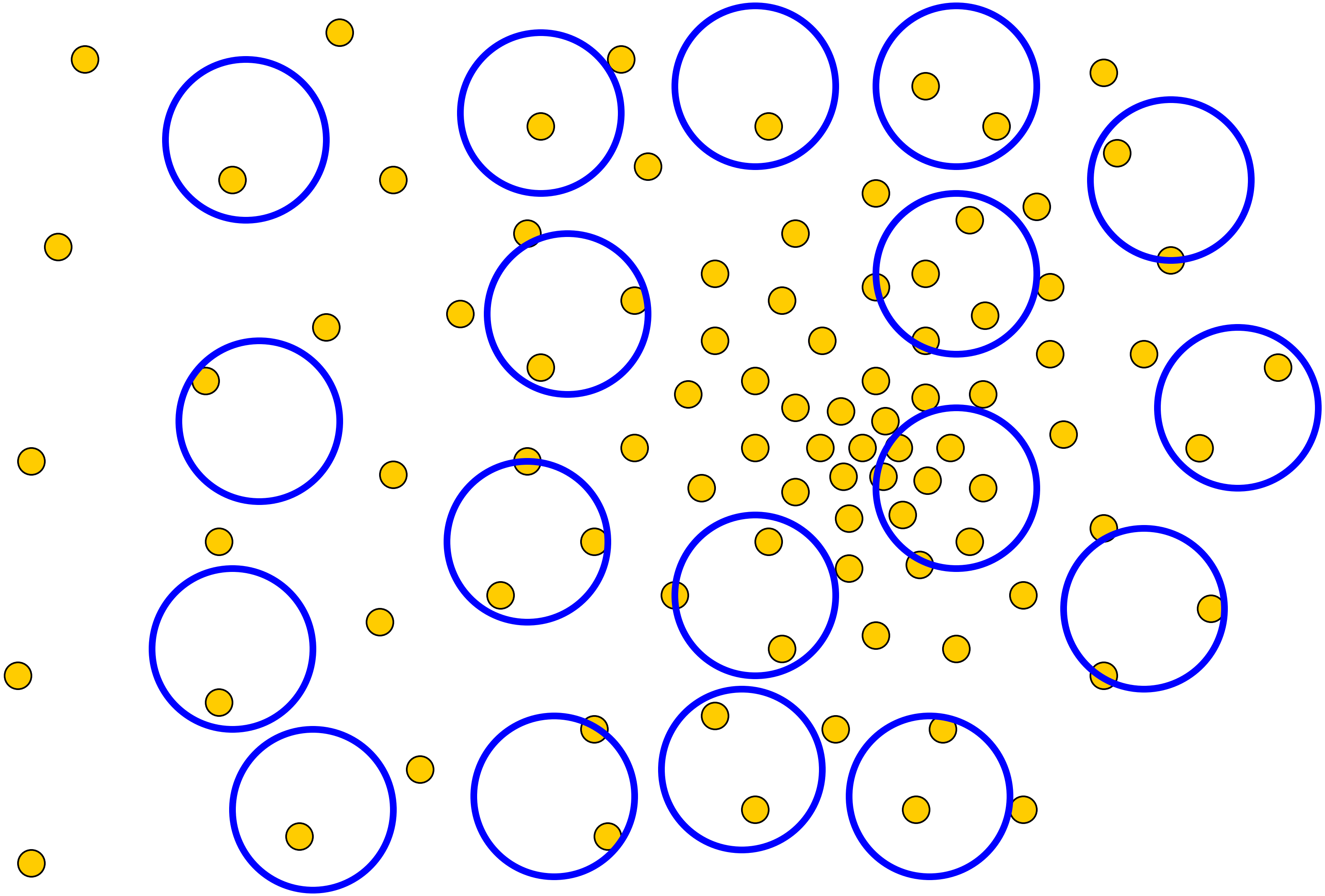
- Image is transformed into a 3-dimensional feature space
- Location of a pixel in feature space is determined by its R,G,B values
- Pixels with similar colors are located close to each other in feature space



Segmentation algorithm

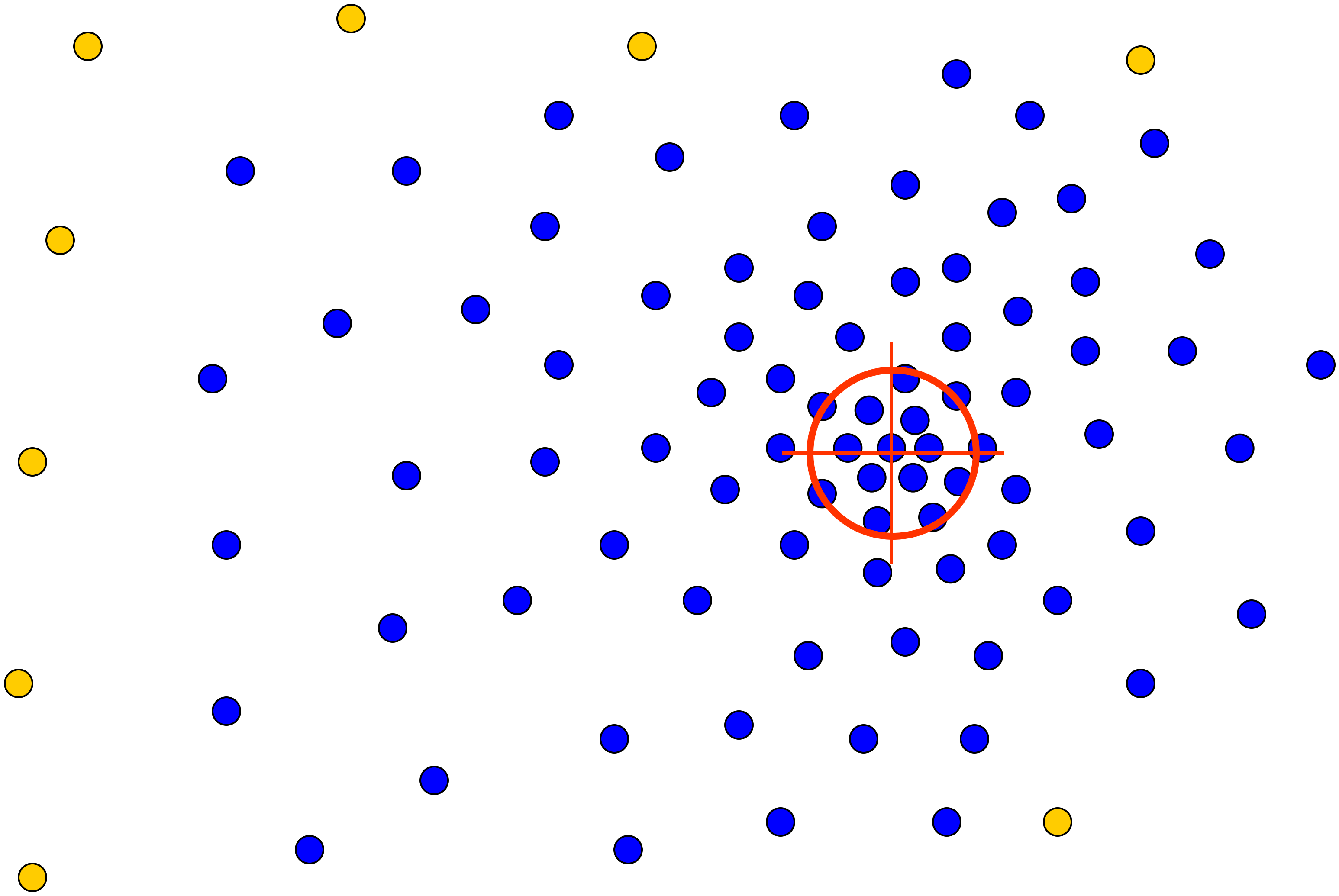
1. Convert the image into feature space (via color, gradients, texture measures etc).
2. Choose initial search window locations uniformly in the data (e.g. each pixel).
3. Repeat until convergence:
   Compute the mean shift window location for each initial position
4. Merge windows that end up on the same “peak” or mode.
5. The data within the traversed windows are clustered together.

Segmentation algorithm



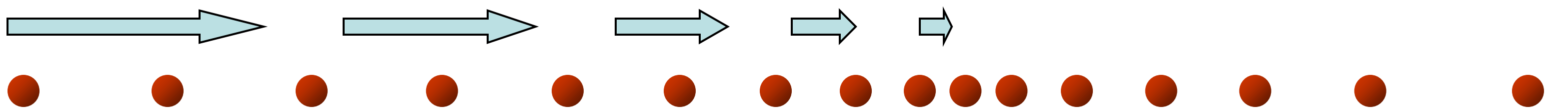
Spread seed points over feature space

Segmentation algorithm



Points that converge to the same mode form a segment

Mean Shift Strengths & Weaknesses



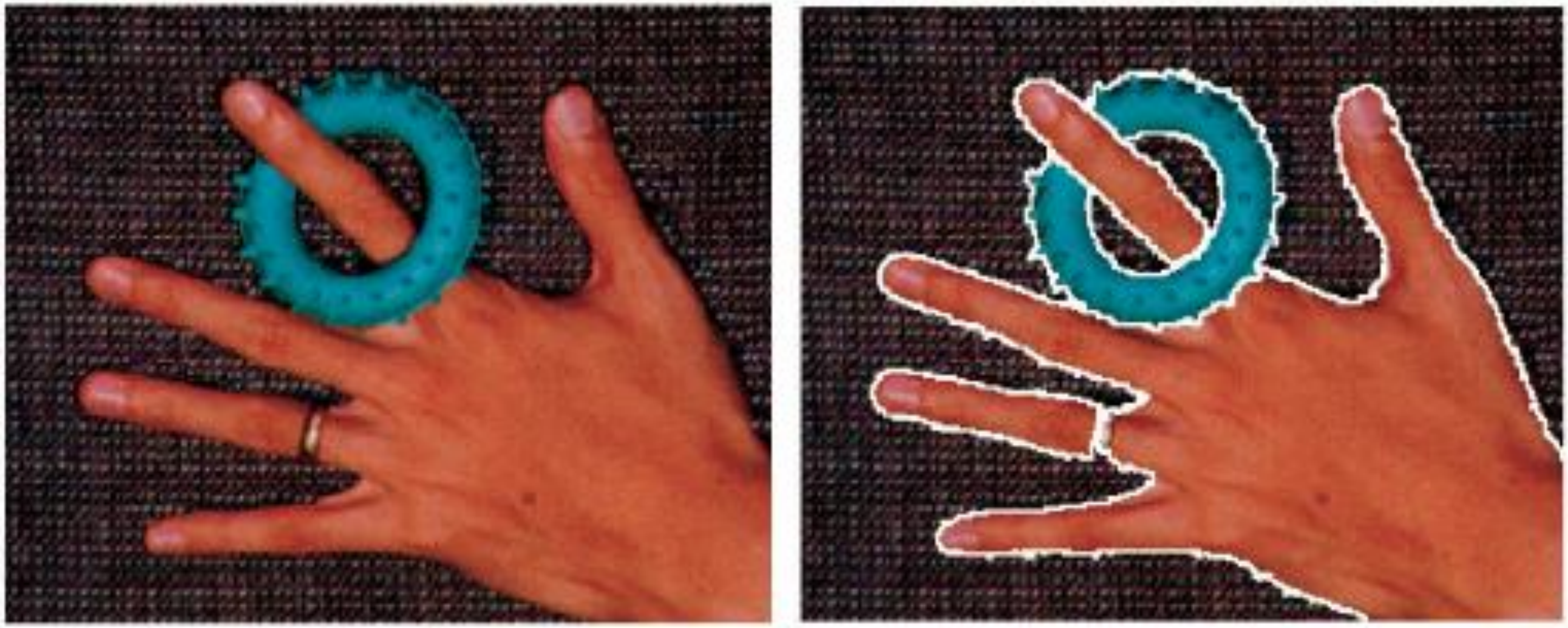
Strengths :

- Application independent tool
- Suitable for real data analysis
- Does not assume any prior shape (e.g. elliptical) on data clusters
- Can handle arbitrary feature spaces
- Only ONE parameter to choose (window size)

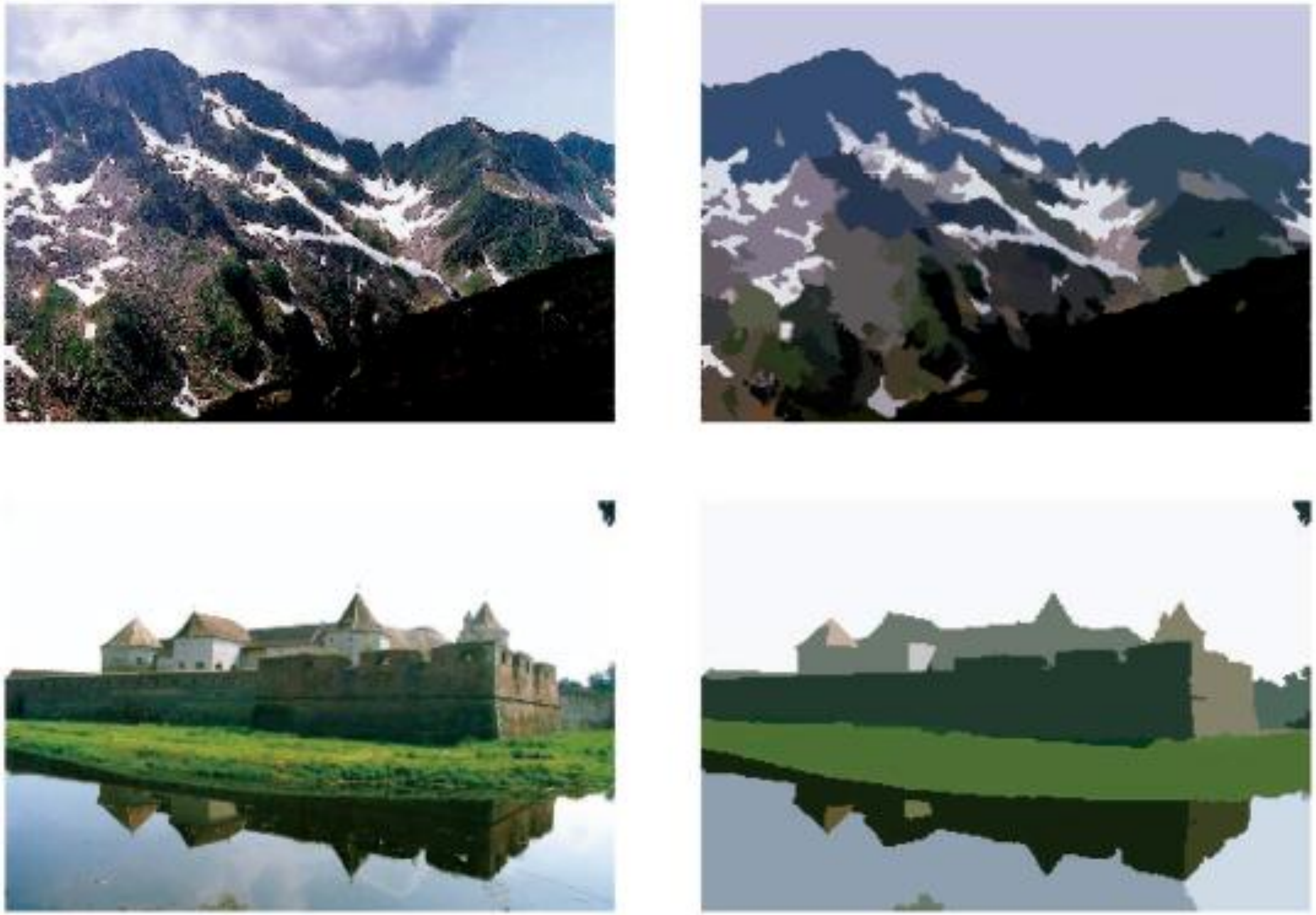
Weaknesses :

- The window size (bandwidth selection) is not trivial
- Inappropriate window size can cause modes to be merged, or generate additional "shallow" modes
  ➔ Use adaptive window size

Examples



Examples



Examples




Mean Shift Tracking

- The mean shift procedure can be used for object tracking.
- Tracked region is a simple ellipsoidal region
- A simple histogram based model is used to describe the tracked object inside the region
- The mean shift procedure is used to find the most probable image location for the region in the next frame

Color Model for the Target

- The target is represented by an ellipsoidal region in the image, normalized to a unit circle. Let $\{x_i\}_{i=1,\ldots n}$ be the normalized pixel locations in the region centered at 0.
- The probability of the feature (color) of the target is modeled by its histogram with kernel k:

$$q_u = C\sum_{i=1}^{n} k(\| x_i \|^2)\delta[b(x_i) - u], \quad u = 1,\ldots m \text{ bins}$$

- Contribution to histogram is weighted by kernel.

$$C = \frac{1}{\sum_{i=1}^{n} k(\|x_i\|^2)}$$

Target Candidate

- Let $\{x_i\}_{i=1,\ldots n_h}$ be the pixel locations of the target, centered at *y* in the current frame. The target candidate is modeled as:

$$p_u(y) = C_h \sum_{i=1}^{n_h} k\left(\left\| \frac{y - x_i}{h} \right\|^2\right) \delta[b(x_i) - u], \quad u = 1,\ldots m \text{ bins}$$

$$C_h = \frac{1}{\sum_{i=1}^{n_h} k\left(\left\| \frac{y - x_i}{h} \right\|^2\right)}$$

Similarity Function

- The similarity function is defined as the metric distance between the candidate and the target model:

$$d(y) = \sqrt{1 - \rho[p(y), q]}$$

- Choose ρ as the Bhattacharyya coefficients (it is a divergence type measure)

$$\rho[p(y), q] = \sum_{u=1}^{m} \sqrt{p_u(y) q_u}$$

- Minimizing the distance is equivalent to maximizing ρ .

Maximization with Mean Shift

- Assume the target candidate histogram does not change drastically, using Taylor expansion around the values $p_u(y_0)$ at location y_0 :

$$\rho[p(y),q] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{p_u(y_0)q_u} + \frac{1}{2}\sum_{u=1}^{m} p_u(y)\sqrt{\frac{q_u}{p_u(y_0)}}$$

$$\approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{p_u(y_0)q_u} + \frac{C_h}{2}\sum_{i=1}^{n_h} w_i k\left(\left\|\frac{y-x_i}{h}\right\|^2\right)$$

where $w_i = \sum_{i=1}^{m}\sqrt{\frac{q_u}{p_u(y_0)}}\delta[b(x_i)-u]$

- Only need to maximize the second term, which is the density estimate with kernel profile *k*(.) at y in the current frame, with the data being weighted by w_i.

Tracking Algorithm with Mean Shift

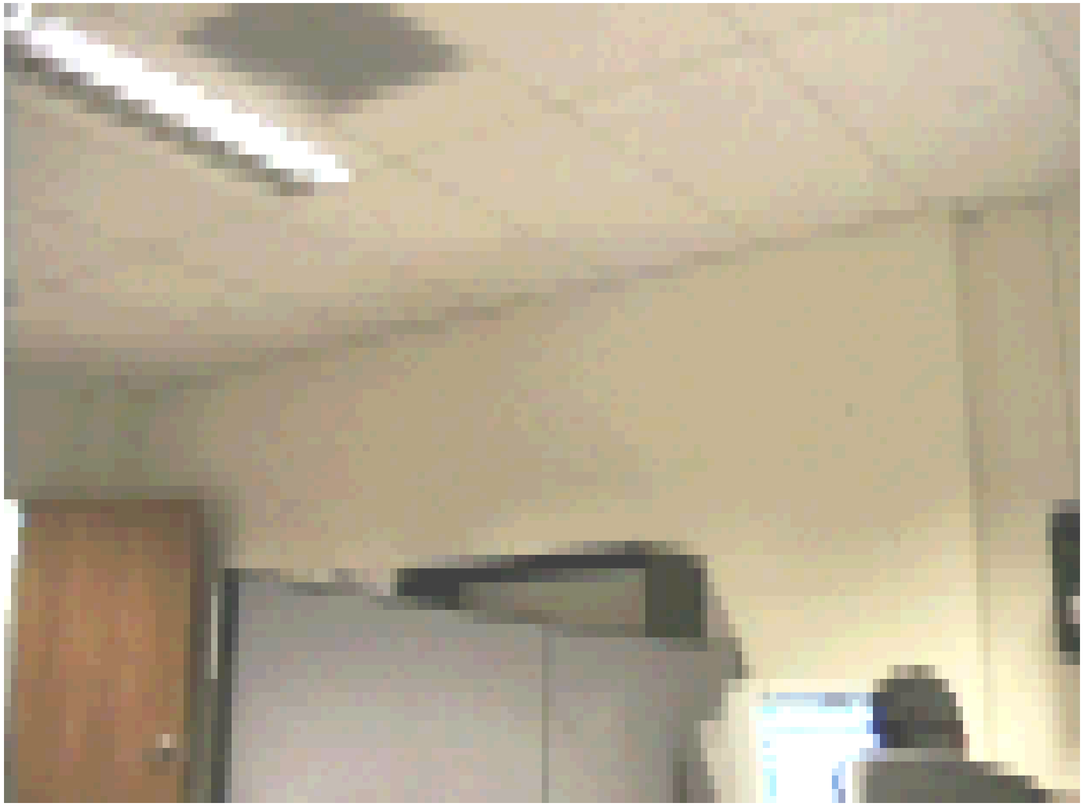
- Given the target model q_u and its location y_0 in the previous frame.

1. Initialize the location at the current frame with y_0 .
2. Compute the next location y_1 according to the mean shift update equation using the weights w_i.
3. Iterate 1 and 2 until converge.

$$y_1 = \frac{\sum_{i=1}^{n_h} x_i w_i g\left(\left\|\frac{y - x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n_h} w_i g\left(\left\|\frac{y - x_i}{h}\right\|^2\right)}, \qquad g = -k'$$

Example

Example



Example