Digital Image Processing

Mean-Shift Segmentation

Suyash P. Awate

Segmentation

- Partitioning an image into regions
  - Labeling
- Criteria
  - **Small variation** of intensities / patterns **within** segment
  - **Large variation** of intensities / patterns **between** segments

Mean-Shift Segmentation

- References
  - https://en.wikipedia.org/wiki/Mean-shift
  - Mean Shift: A Robust Approach Toward Feature Space Analysis.
    D Comaniciu and P Meer.
    IEEE Transactions on Pattern Analysis and Machine Intelligence 2002. 24(2):603-619

Mean-Shift Segmentation

- **Goal**: Estimate probability density function (PDF) f(x) given observations x_i (d-dimensional) drawn from f(x)
- **Nonparametric / kernel density estimation**
- **Strategy**: Superpose kernel functions placed at x_i

$$f(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right)$$

$$\int_{R^d} K(\mathbf{x}) \mathrm{d}\mathbf{x} = 1$$

$$\int_{R^d} \mathbf{x} K(\mathbf{x}) \mathrm{d}\mathbf{x} = 0$$

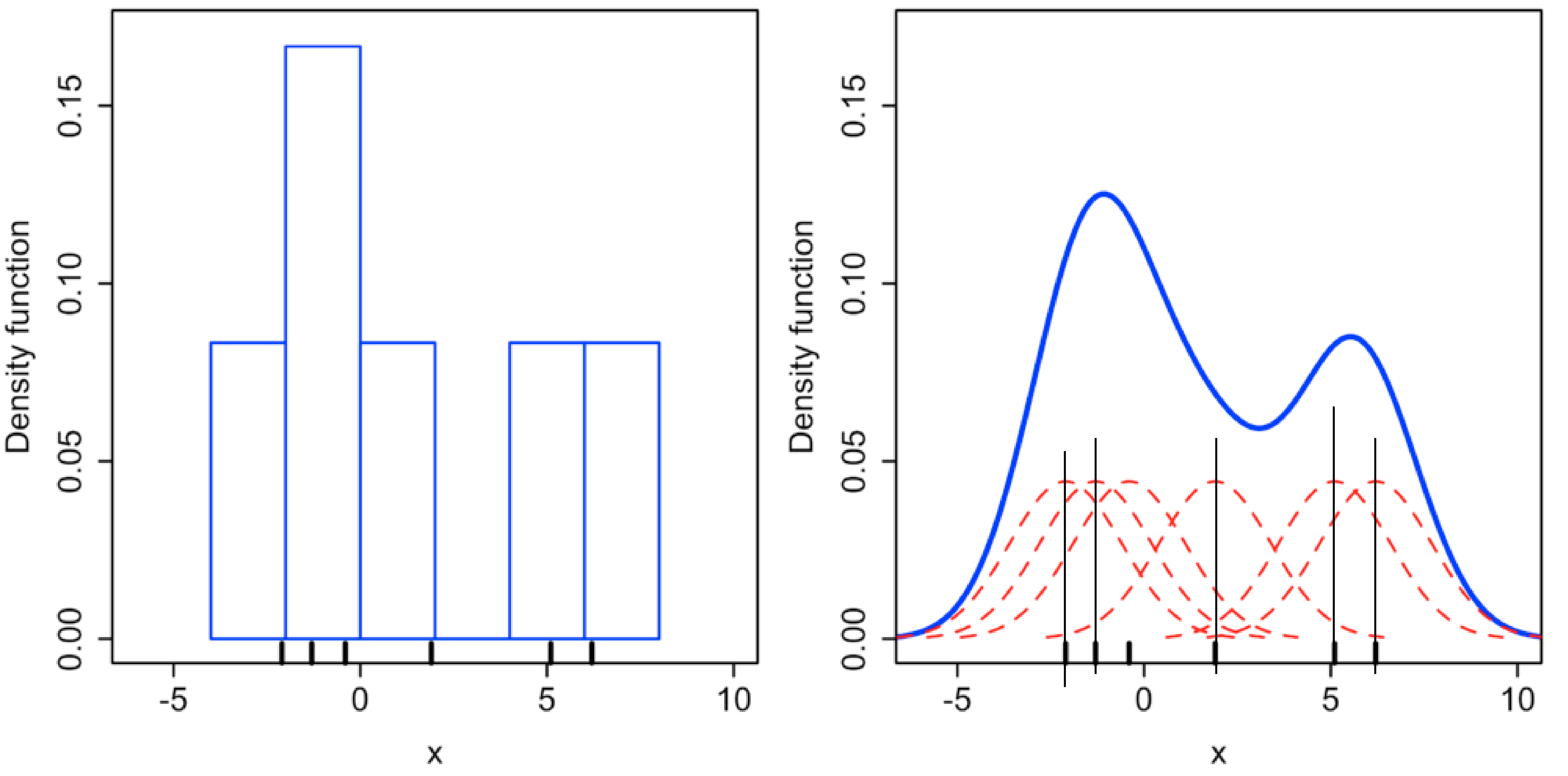
$$\lim_{\|\mathbf{x}\| \to \infty} \|\mathbf{x}\|^d K(\mathbf{x}) = 0$$

  - K(.) = **kernel**
    - Non-negative, integrates to 1, mean 0, finite valued, decays to 0 sufficiently fast
    - Typically, radially symmetric $K(\mathbf{x}) = c_{k,d} k(\|\mathbf{x}\|^2)$
      - k(.) = non-increasing, c = normalization constant
  - h = **bandwidth** parameter

Mean-Shift Segmentation

- Nonparametric / kernel density estimation
  - Histogram (left) versus Kernel density estimate (right)

histogram in position



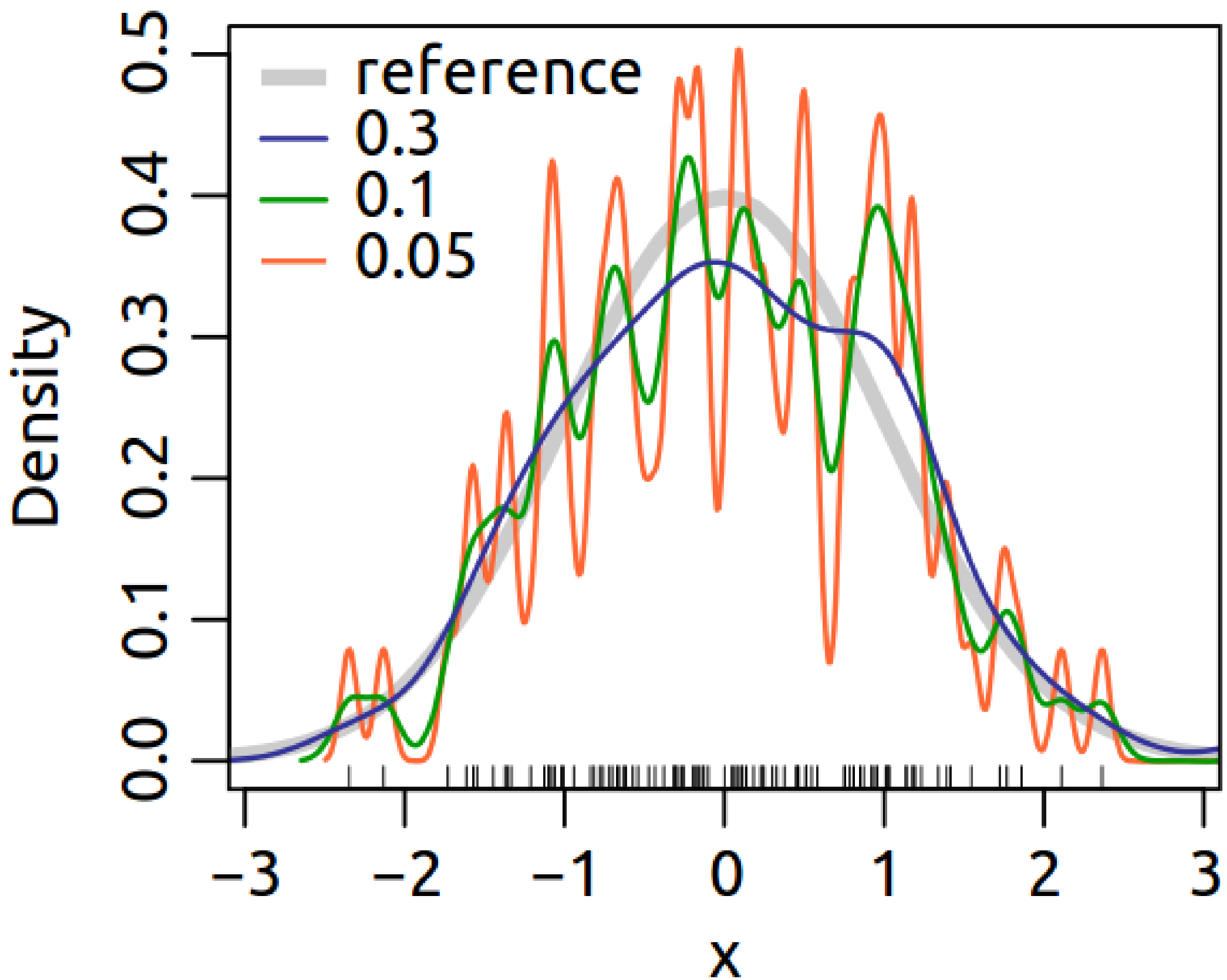
Mean-Shift Segmentation

- Nonparametric / kernel density estimation
  - **Bandwidth** (h) **selection**
    - Too small: orange
    - Small: green
    - Desirable: blue
    - True: gray
    - What if
      h is too large ?

graph will coinside to x axis

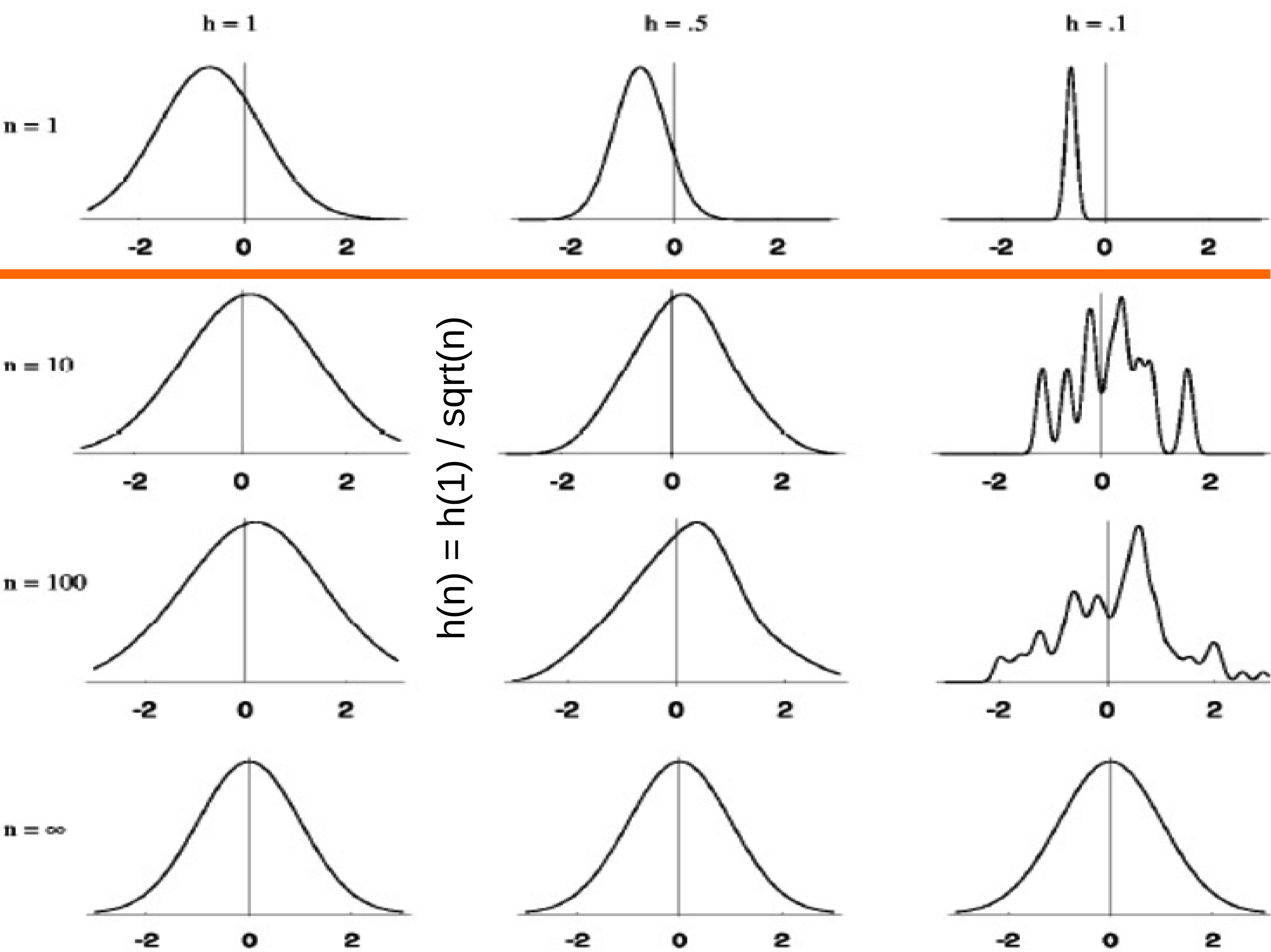
    - What if
      h → 0 ?

there will be spikes in the graph

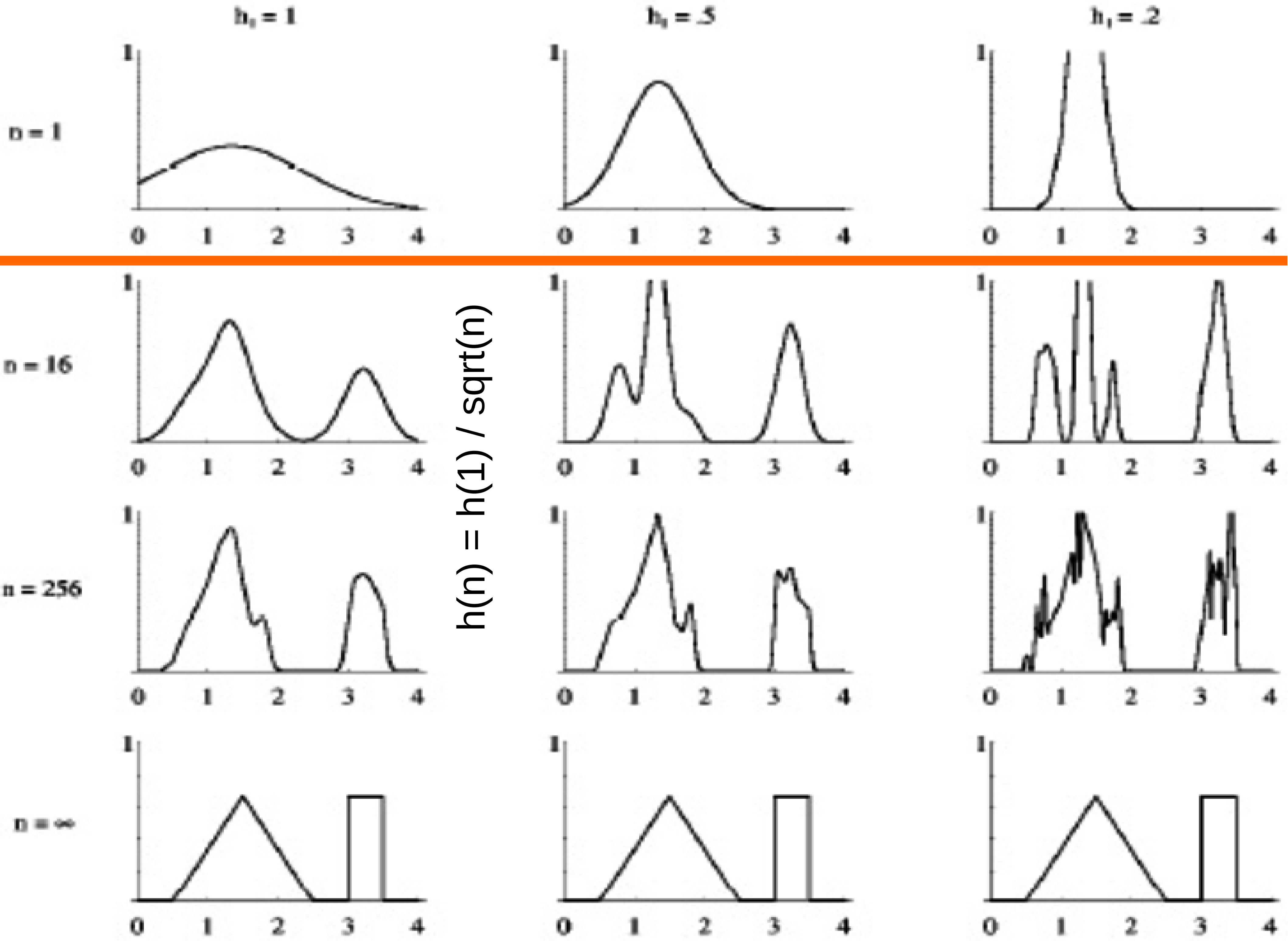


Mean-Shift Segmentation

- Nonparametric / kernel density estimation
- Density estimate convergence as sample size $n \rightarrow \infty$
  - Bandwidth h must reduce to 0 with increasing n
    - Sufficiently fast
    - Sufficiently slow
    - $\mathrm{Lim}_{n \rightarrow \infty} \quad [\, h(n) \,]^d = 0$ h should be function of no of observation
      $\mathrm{Lim}_{n \rightarrow \infty} \quad n\, [\, h(n) \,]^d = \infty$
    - Example for 1D case (d=1),
      h(n) = h(1) / sqrt(n)
  - Guaranteed convergence
    - $P_n(x)$ is estimate of density p(x) at 'x', using sample size 'n'
    - $\lim_{n \rightarrow \infty} E \quad [\, P_n(x) \,] = p(x)$
    - $\lim_{n \rightarrow \infty} \mathrm{Var}\, [\, P_n(x) \,] = 0$ (convergence in mean square)

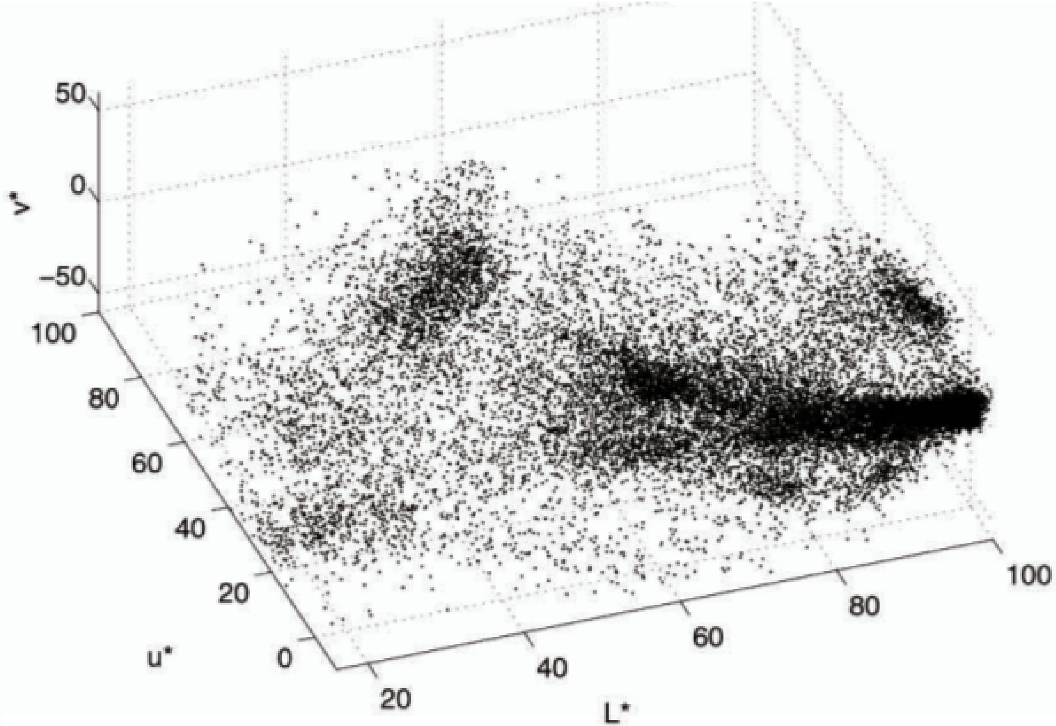


h = 1
h = .5
h = .1
n = 1
n = 10
n = 100
n = ∞
h(n) = h(1) / sqrt(n)
-2
0
2


h1 = 1
h1 = .5
h1 = .2
n = 1
n = 16
n = 256
n = ∞
h(n) = h(1) / sqrt(n)

Mean-Shift Segmentation

- Application to images
  - Left: color image, 3 color components
  - Right: Scatter plot of color 3-tuples
  - Assumption: Each object → cluster of color values
  - *Color can be replaced with any other feature*



Mean-Shift Segmentation

- Segmentation algorithm:
  - (1) Choose bandwidth ‘h’
    - User-defined / “free” parameter
  - (2) Use pixel colors $\{x_i\}$ and ‘h’ to get kernel density estimate f(x)
  - (3) For each observation x_i, do the following:
    - Calculate gradient of log(f(x)) at observation x_i
    - Update / move x_i along that gradient direction
      - (Kernel density estimate gets sharper next time)
  - (4) Repeat last 2 steps

Mean-Shift Segmentation

- Evaluate gradient of log-PDF
  - PDF (with **Gaussian** kernel)

$$f(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right)$$

$$K(\mathbf{x}) = c_{k,d} k(\|\mathbf{x}\|^2)$$

  - Gradient of f(x)

$$\nabla f(\mathbf{x}) \quad = \quad \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{x}) g\left(\left\|\frac{\mathbf{x} - \mathbf{x}_i}{h}\right\|^2\right)$$

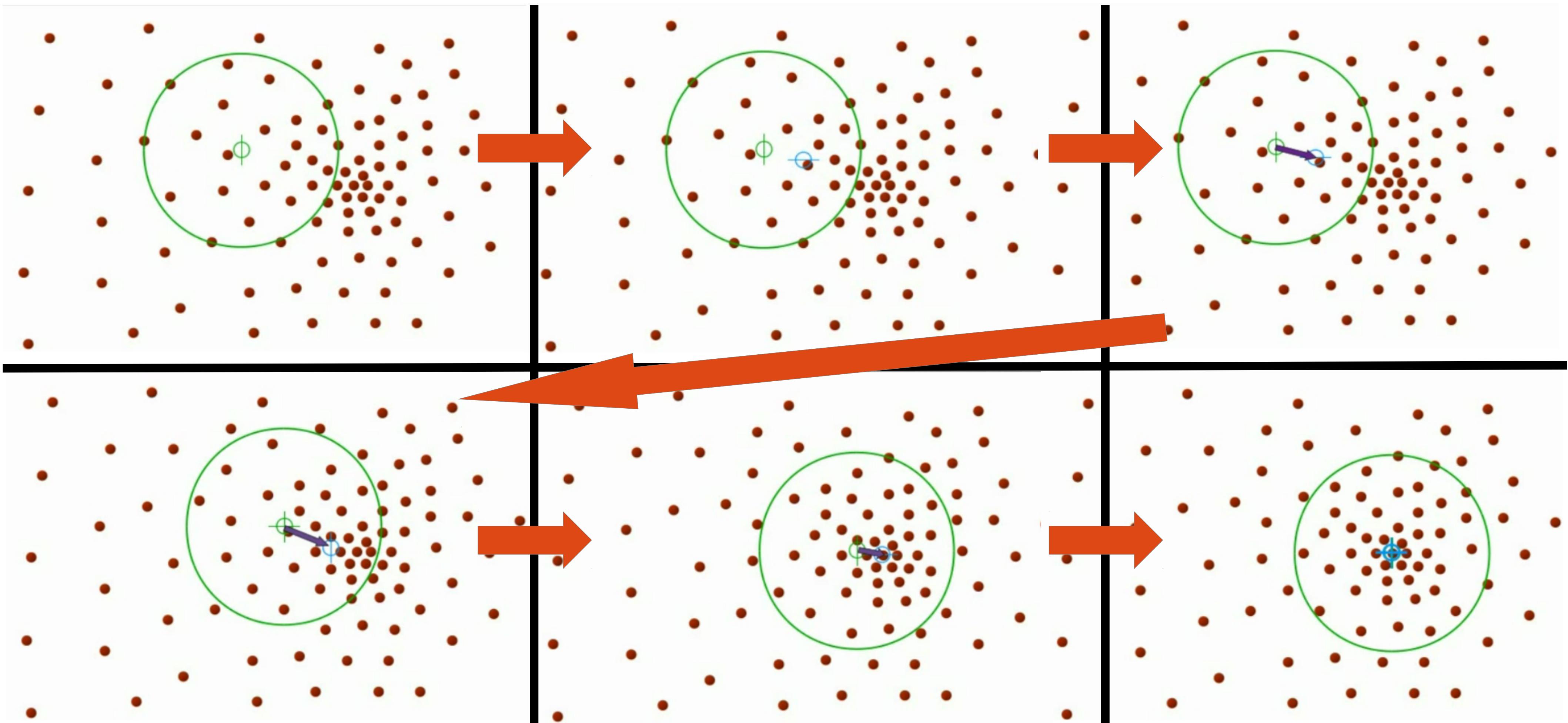
  - Gradient of **log** f(x) is proportional to:

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

  - This displacement is called the “**mean shift**”

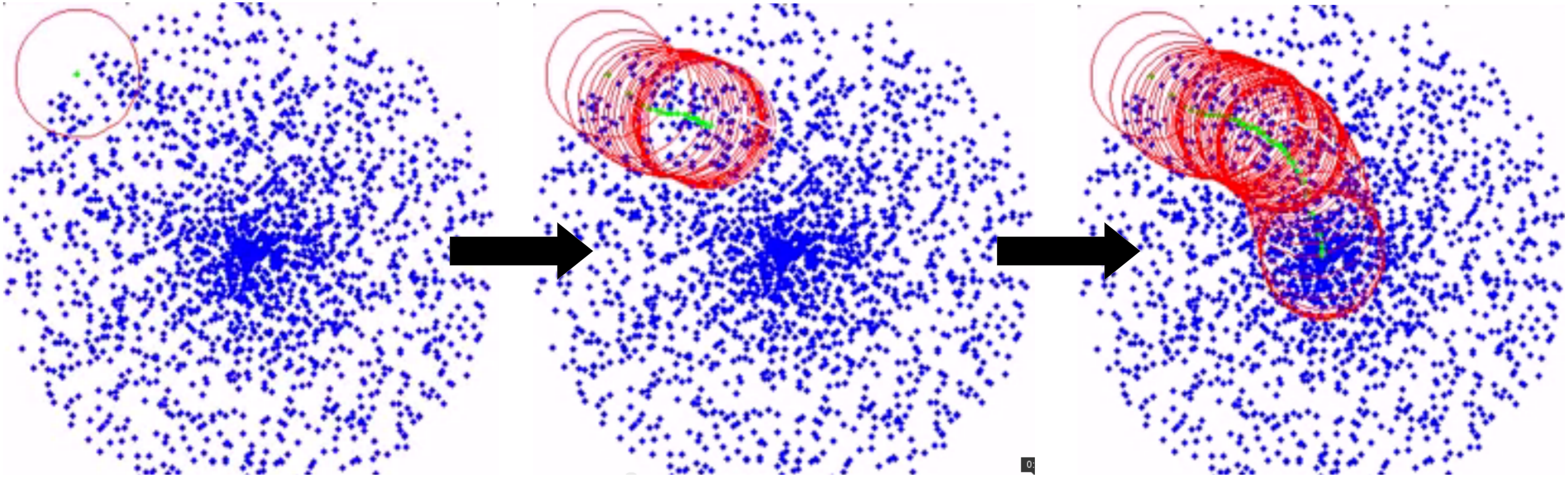
Mean-Shift Segmentation

- Mesh-shift updates
  - Only 1 point moves, others remain at the same position
  - https://www.youtube.com/watch?v=kmaQAsotT9s



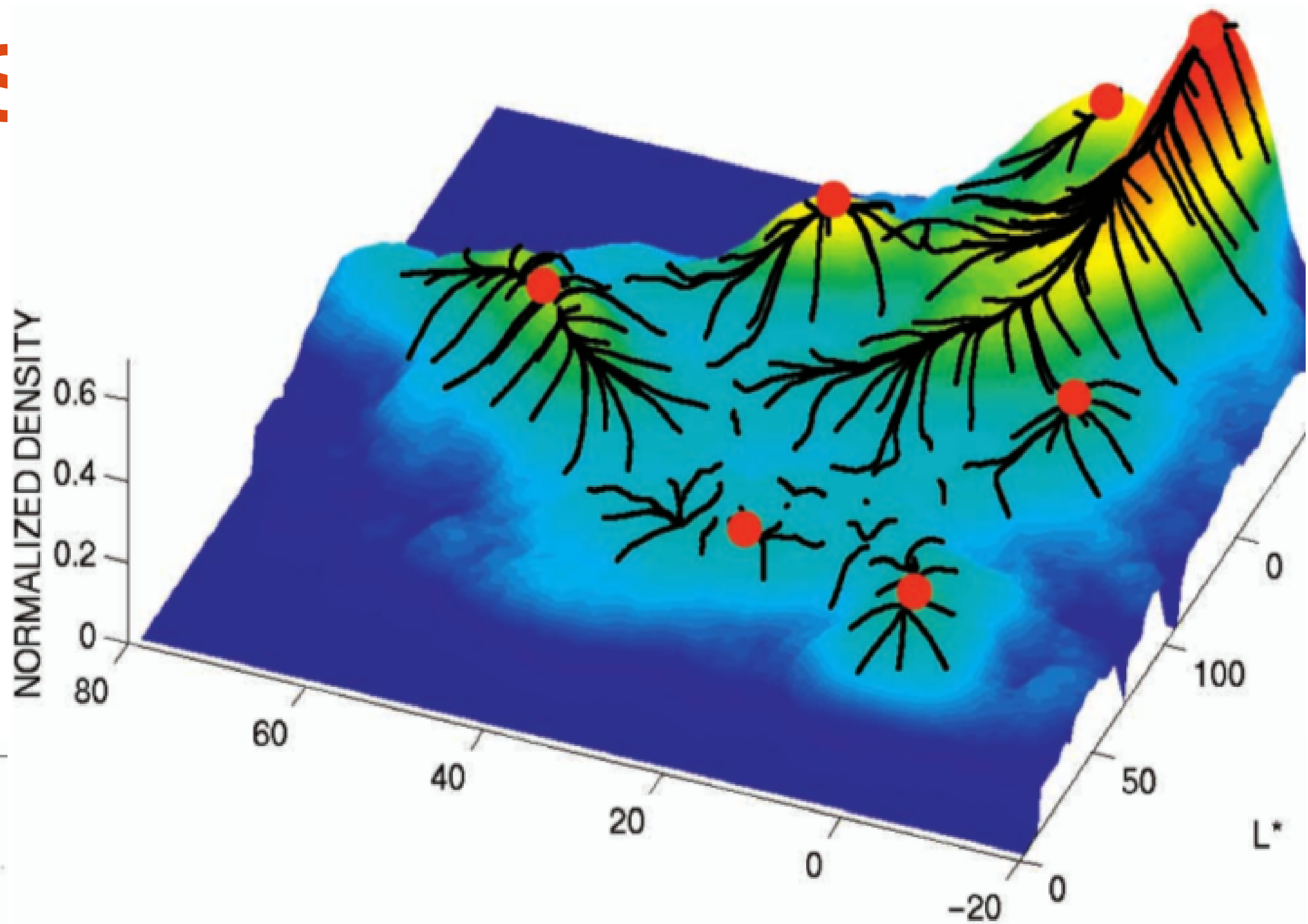
Mean-Shift Segmentation

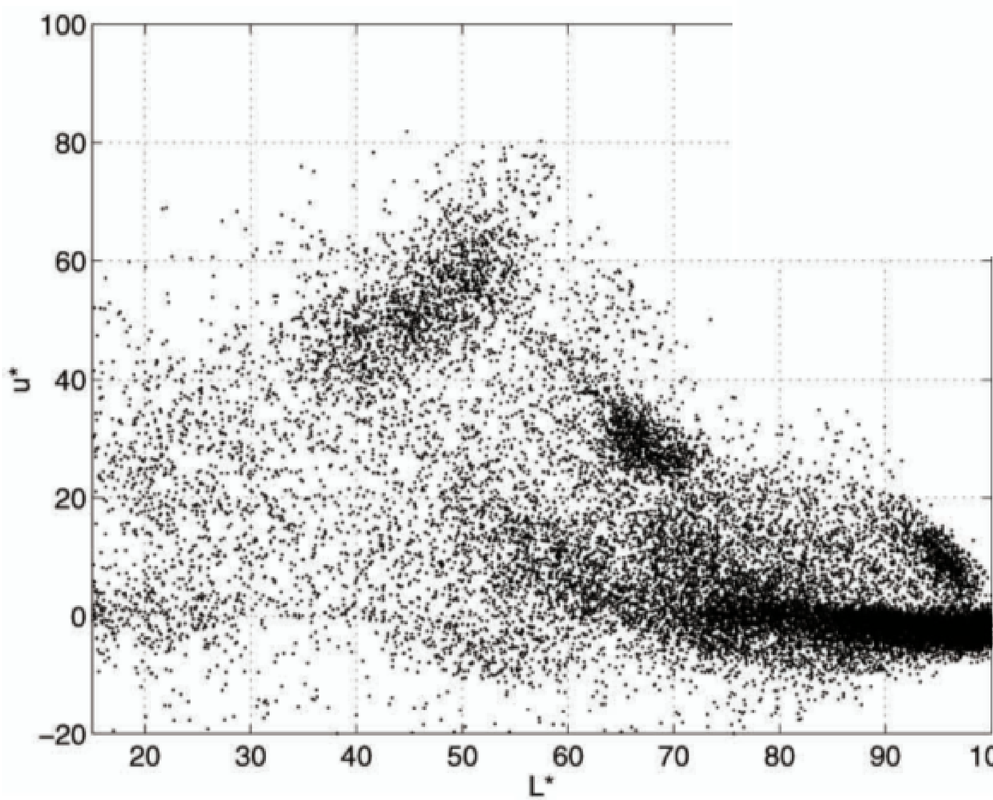
- Mean-shift updates
  - Demo 2: https://www.youtube.com/watch?v=hJg7ik4x95U
  - Red circle = support of kernel placed at green point

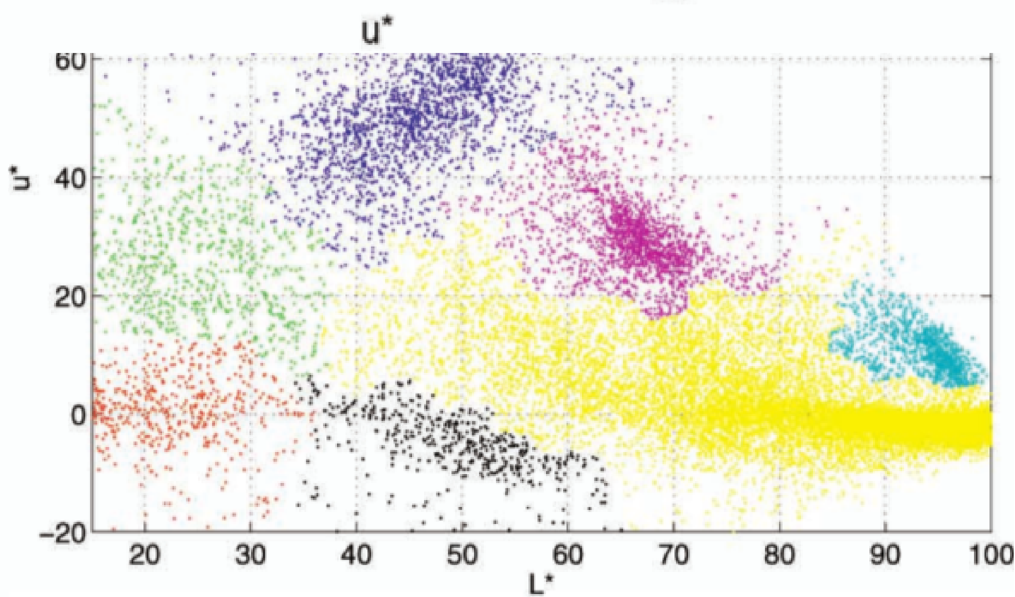


Mean-S

- Trajectories on log PDF
  - Steady state
  - **Finite**-support kernel





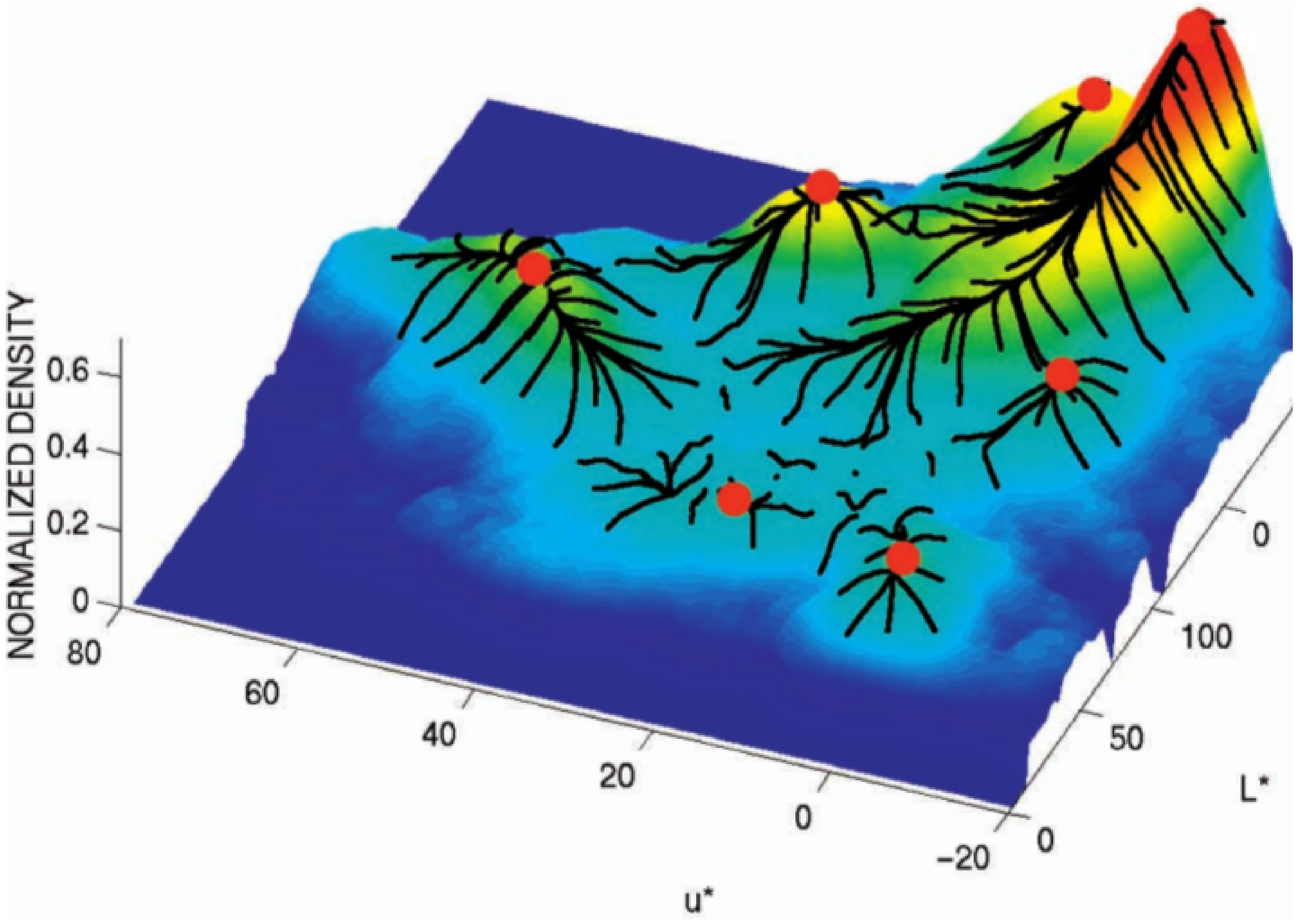


Mean-Shift Segmentation

- Cluster results to get segmentation
  - Assign the same label to
    a group of points coalesced (very close) in feature space
  - K-means clustering
- Why not K-means ?
  - (1) How to select K ?
    - Mean shift implicitly does so by selecting bandwidth ‘h’
    - Selecting ‘h’ may be more intuitive than selecting K
  - (2) K-means doesn’t use PDF-gradient information
    - K-means can put points around a PDF’s local mode into 2 clusters, unlike mean shift
- Why does mean-shift use gradient of **log**-PDF ?

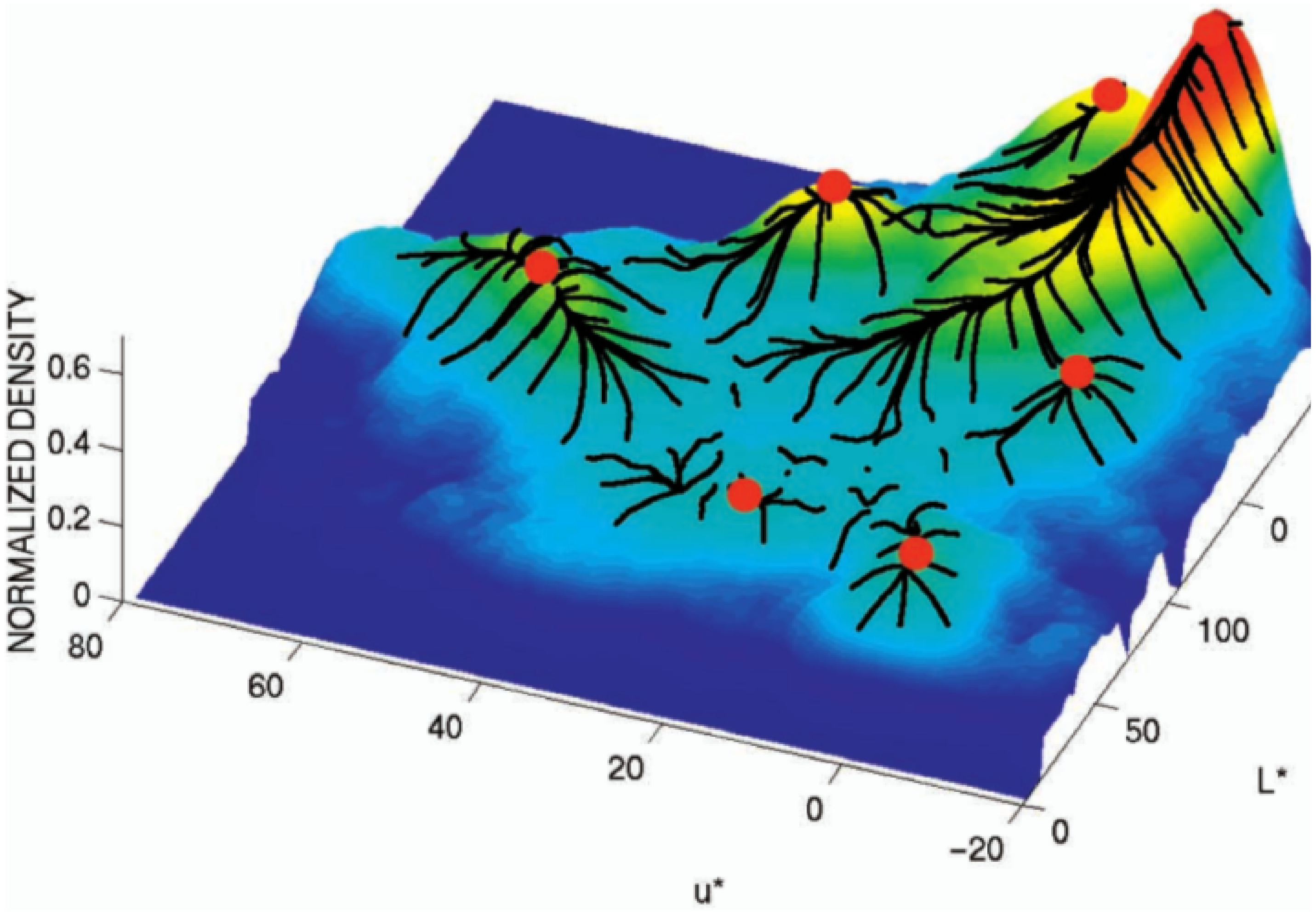
Mean-Shift Segmentation

- Trajectories on log PDF
  - What is steady state with kernel of infinite support, e.g., Gaussian ?



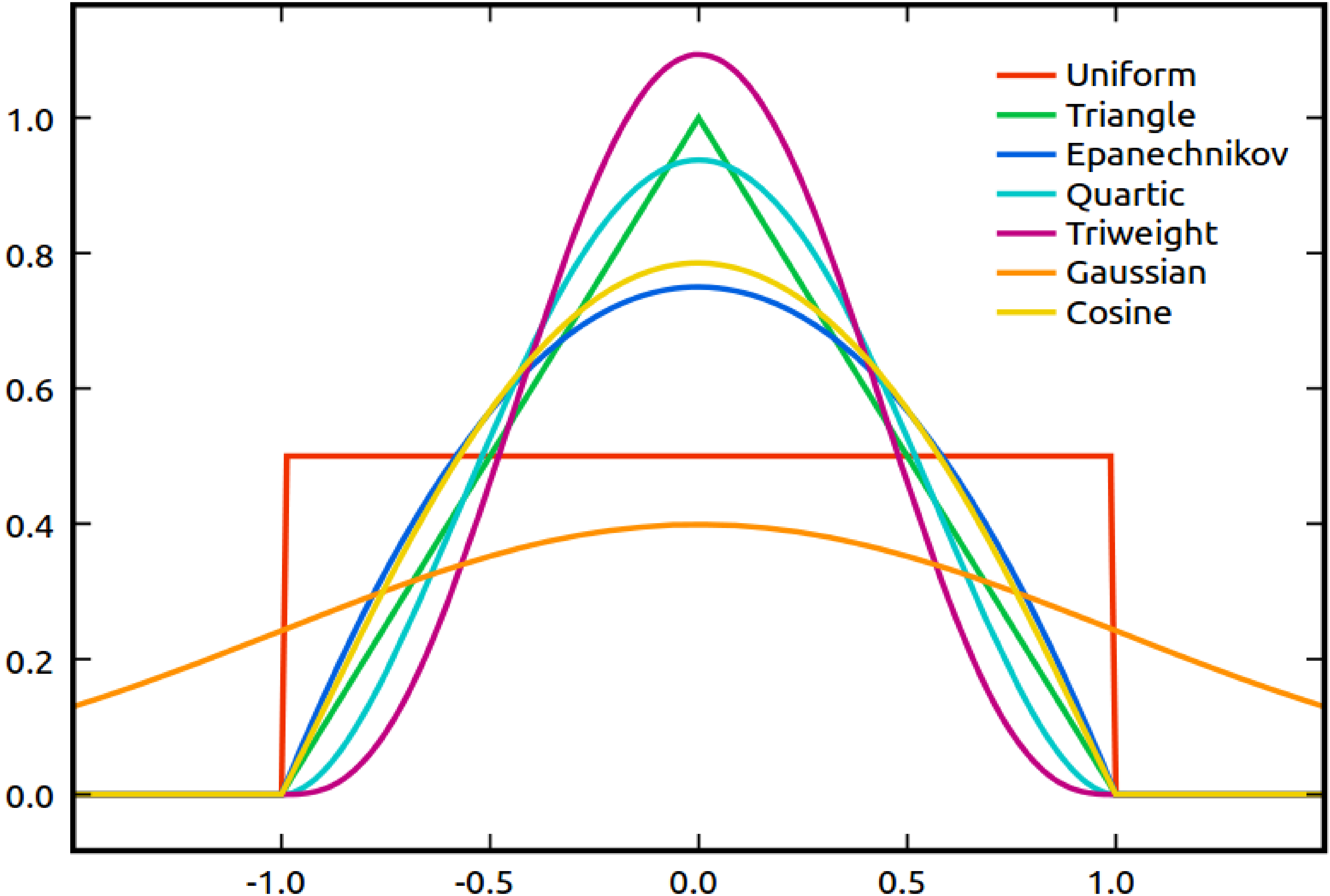
Mean-Shift Segmentation

- Trajectories on log PDF
  - How to achieve this (in picture) as the steady state ?
  - Kernel needs to be of finite support
    - See next slide for examples



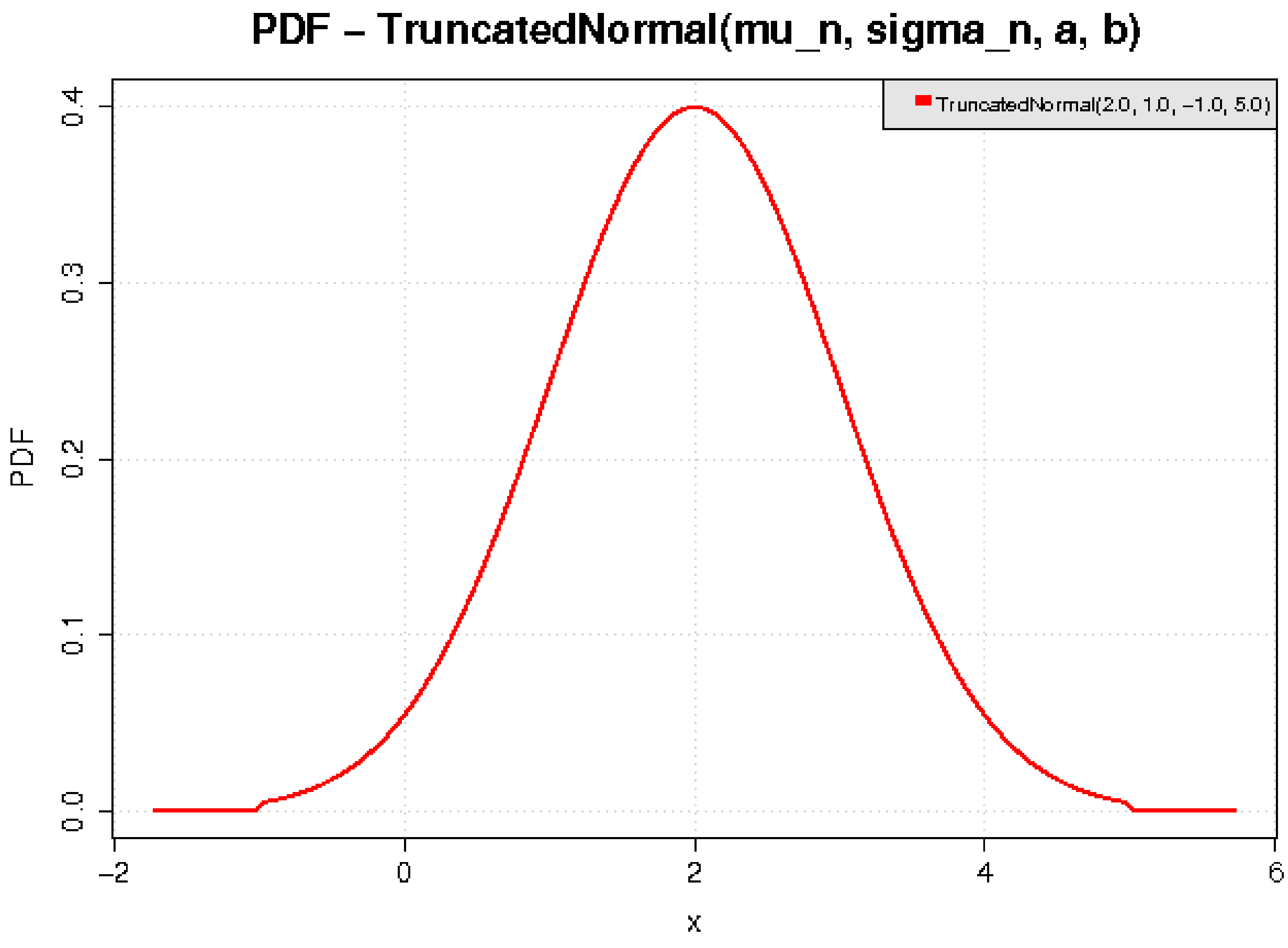
Mean-Shift Segmentation

- Finite-support kernels (compared with the Gaussian)



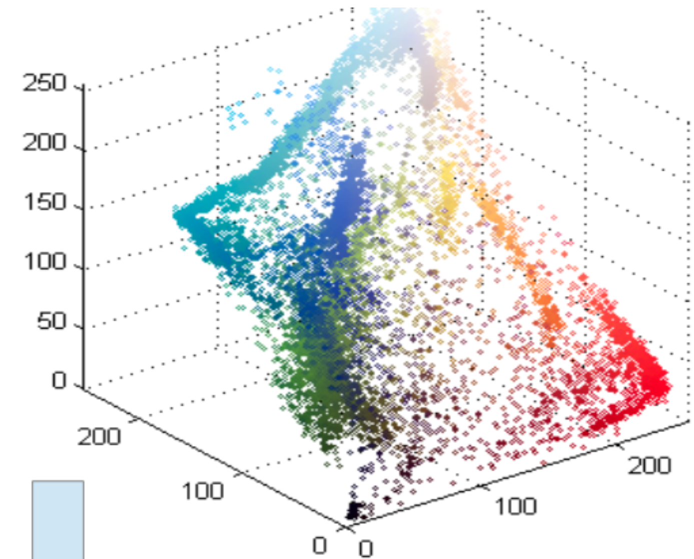
Mean-Shift Segmentation

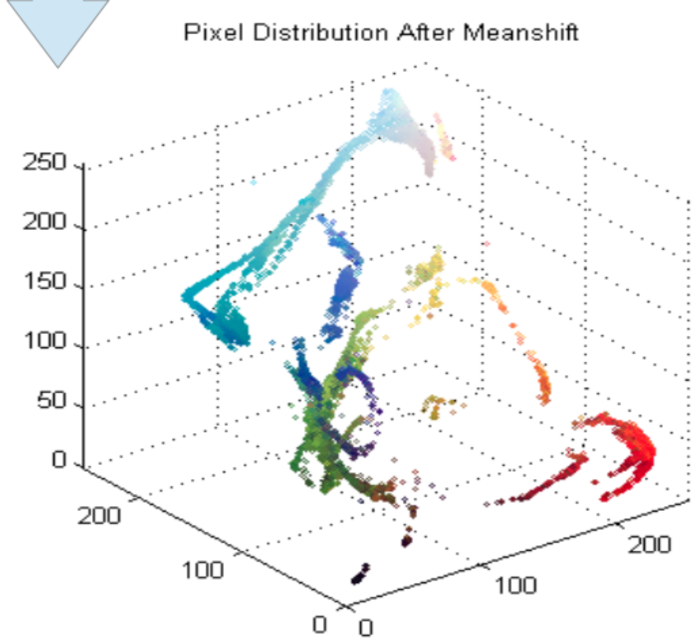
- Finite-support kernels
  - Truncated Gaussian



Mean-Shift Segmentation

- Algorithm in action





Mean-Shift Segmentation

- Algorithm convergence
  - Guaranteed when using Epanechnikov kernel
    - Proof relies on 2 properties:

    1) Given finite sample size, the kernel density estimate is bounded (finite)

    2) Each update increases the kernel density estimate

Mean-Shift Segmentation

- Combining color and spatial features
  - How to differentiate between different objects of similar color, but spatially apart ?
  - Need kernel on the joint space of pixel color + pixel coordinate
    - Product of kernels on each space
    - 2 bandwidth parameters

$$K_{h_s,h_r}(\mathbf{x}) = \frac{C}{h_s^2 h_r^p} k\left(\left\|\frac{\mathbf{x}^s}{h_s}\right\|^2\right) k\left(\left\|\frac{\mathbf{x}^r}{h_r}\right\|^2\right)$$

Mean-Shift Segmentation




Mean-Shift Segmentation

Mean-Shift Segmentation

Mean-Shift

- Effect of bandwidth
  - Increase hr → more color mixing
  - Increase hs → more smooth

$(h_s, h_r) = (8, 8)$

$(h_s, h_r)=(8, 16)$

$(h_s, h_r)=(16, 8)$