Kernel density estimation techniques with applications to image filtering and segmentation

Prof. PhD. Vasile Gui

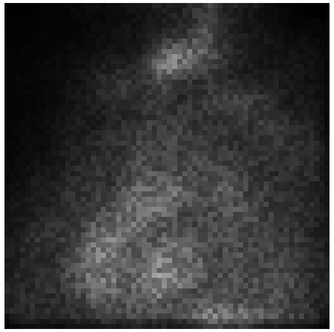
Polytechnic University of Timisoara

Content

- Introduction
- Brief review of linear operators
- Nonlinear image smoothing techniques
  - Bilateral filter
- Nonparametric density estimation
  - Brief introduction
  - Mean shift filter
  - Mean shift segmentation

Introduction

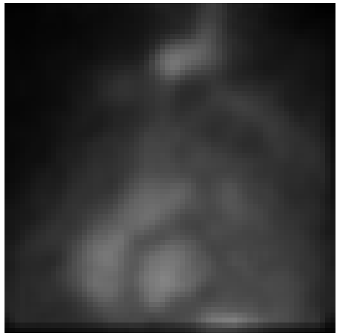
Why do we need image smoothing?



What is "image" and what is "noise"?

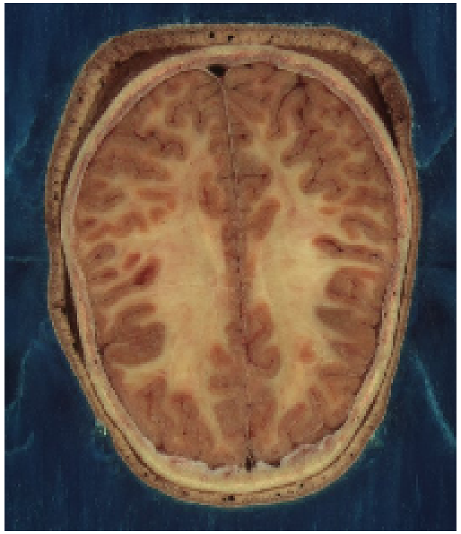
- Frequency spectrum
- Statistical properties

Prior knowledge



Introduction

Why do we need segmentation?



Brief Review of Linear Operators [Pratt 1991]

- Generalized 2D linear operator

$$g(m,n) = \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} O(j,k;m,n) f(j,k)$$

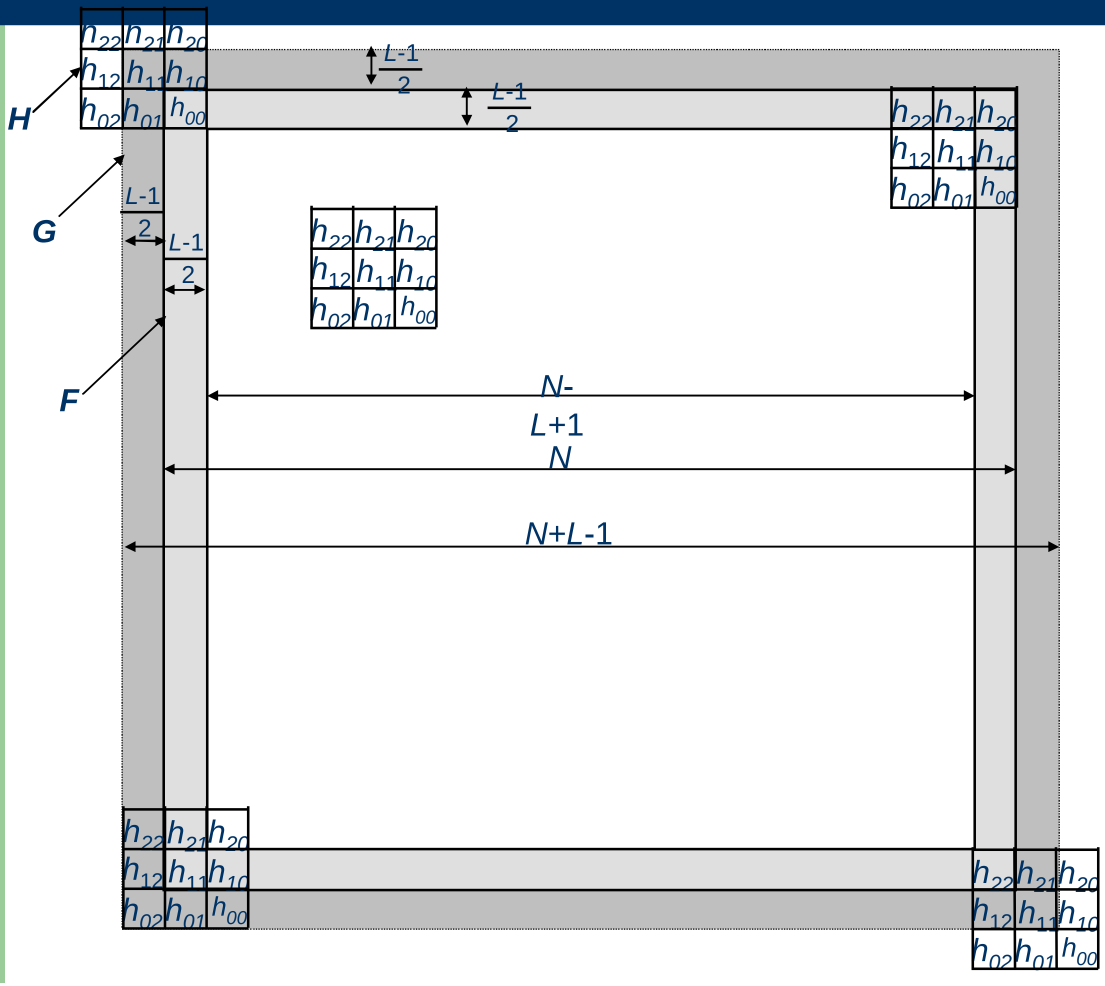
- Space invariant operator:

$$O(j,k;m,n) = O(m-j,n-k) = H(m-j,n-k)$$

$$g(m,n) = \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} h(m-j,n-k) f(j,k)$$

Convolution sum: weighted average of pixels within a window

Brief Review of Linear Operators



- Geometrical interpretation of 2D convolution

Linear Image smoothing techniques

Box filters. Arithmetic mean L´L operator

$$\mathbf{h} = \frac{1}{L^2}\begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & 1 \\ 1 & 1 & \cdots & 1 \end{bmatrix} = \frac{1}{L}\begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} * \frac{1}{L}\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

Linear Image smoothing techniques

Box filters. Arithmetic mean L´L operator

Optimality properties

- Signal and additive white noise

$$\boldsymbol{g} = \frac{1}{N}\sum_{k=1}^{N}(\boldsymbol{f}_k + \boldsymbol{n}_k) = \frac{1}{N}\sum_{k=1}^{N}\boldsymbol{f}_k + \frac{1}{N}\sum_{k=1}^{N}\boldsymbol{n}_k .$$

$$\boldsymbol{z} = \frac{1}{N}\sum_{k=1}^{N}\boldsymbol{n}_k .$$

- Noise variance is reduced *N* times

$$\sigma_z^2 = \mathrm{E}\{z^{\mathrm{T}} z\} = \frac{1}{N^2}\mathrm{E}\{\sum_{k=1}^{N}\sum_{l=1}^{N} n_l n_k\} = \frac{1}{N^2}\sum_{k=1}^{N}\sum_{l=1}^{N}\mathrm{E}\{n_l n_k\} =$$

$$\frac{1}{N^2}\sum_{k=1}^{N}\sum_{l=1}^{N}\delta(l,k)\sigma^2 = \frac{1}{N}\sigma^2$$

Linear Image smoothing techniques

Box filters. Arithmetic mean L´L operator

- Unknown constant signal plus noise
- Minimize MSE of the estimation g:

$$\varepsilon^2(g) = \sum_{k=1}^{N} (f_k - g)^2$$

$$\frac{\partial \varepsilon^2(g)}{\partial g} = 0$$

$$\hat{g} = \frac{1}{N} \sum_{k=1}^{N} f_k$$

Linear Image smoothing techniques

Box filters. Arithmetic mean L´L operator

- i.i.d. Gaussian signal with unknown mean.

$$p(f) = ct.\exp\{-\frac{(f-\mu)^2}{2\sigma^2\ \mu}\}$$

- maximize the probability to obtain the observed samples,

$$\prod_{k=1}^{N} p(f_k) = ct.\exp\{-\frac{1}{2\sigma^2}\sum_{k=1}^{N}(f_k-\mu)^2\}$$

- Optimal solution: • = arithmetic mean of observed samples

Linear Image smoothing techniques

Box filters. Arithmetic mean L´L operator

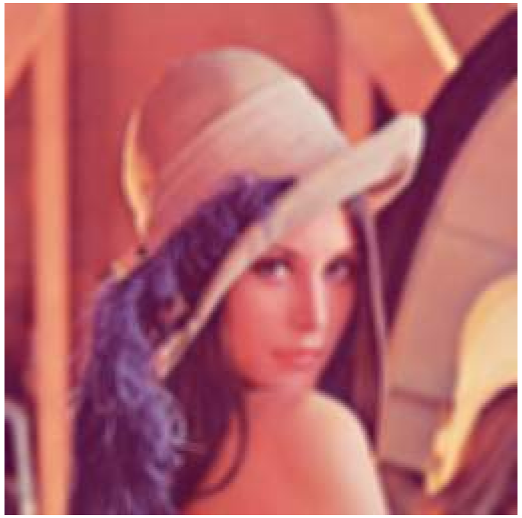
- Image smoothed with 3×3, 5×5, 9×9

and 11 ×11 box filters

Linear Image smoothing techniques

Box filters. Arithmetic mean L´L operator

Original Lena image



Lena image filtered with

5x5 box filter

Linear Image smoothing techniques

Binomial filters [Jahne 1995]

- Computes a weighted average of pixels in the window
- Less blurring, less noise cleaning for the same size
- The family of binomial filters can be defined recursively
- The coefficients can be found from $(1+x)^n$

Linear Image smoothing techniques

Binomial filters. 1D versions

$$\mathbf{b}^1 = \frac{1}{2}\begin{bmatrix}1 & 1\end{bmatrix}$$

$$\mathbf{b}^2 = \frac{1}{4}\begin{bmatrix}1 & 2 & 1\end{bmatrix} = \mathbf{b}^1 * \mathbf{b}^1$$

$$\mathbf{b}^4 = \frac{1}{16}\begin{bmatrix}1 & 4 & 6 & 4 & 1\end{bmatrix} = \mathbf{b}^1 * \mathbf{b}^1 * \mathbf{b}^1 * \mathbf{b}^1$$

$$\mathbf{b}^6 = \frac{1}{64}\begin{bmatrix}1 & 6 & 15 & 20 & 15 & 6 & 1\end{bmatrix} = \mathbf{b}^1 * \mathbf{b}^1 * \mathbf{b}^1 * \mathbf{b}^1 * \mathbf{b}^1 * \mathbf{b}^1$$

As size increases, the shape of the filter is closer to a Gaussian one

Linear Image smoothing techniques

Binomial filters. 2D versions

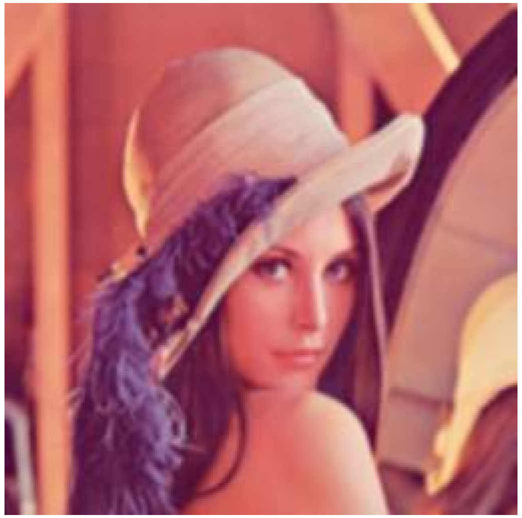
$$\mathbf{b}^2 = \frac{1}{4}\begin{bmatrix}1 & 2 & 1\end{bmatrix} * \frac{1}{4}\begin{bmatrix}1\\2\\1\end{bmatrix} = \frac{1}{16}\begin{bmatrix}1 & 2 & 1\\2 & 4 & 2\\1 & 2 & 1\end{bmatrix}$$

$$\mathbf{b}^4 = \frac{1}{16}\begin{bmatrix}1 & 4 & 6 & 4 & 1\end{bmatrix} * \frac{1}{16}\begin{bmatrix}1\\4\\6\\4\\1\end{bmatrix} = \frac{1}{256}\begin{bmatrix}1 & 4 & 6 & 4 & 1\\4 & 16 & 24 & 16 & 4\\6 & 24 & 36 & 24 & 6\\4 & 16 & 24 & 16 & 4\\1 & 4 & 6 & 4 & 1\end{bmatrix}$$

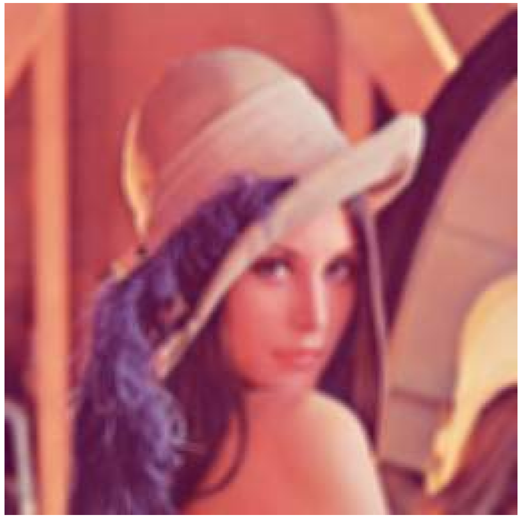
Linear Image smoothing techniques

Binomial filters. Example

Original Lena image



Lena image filtered with binomial 5x5 kernel

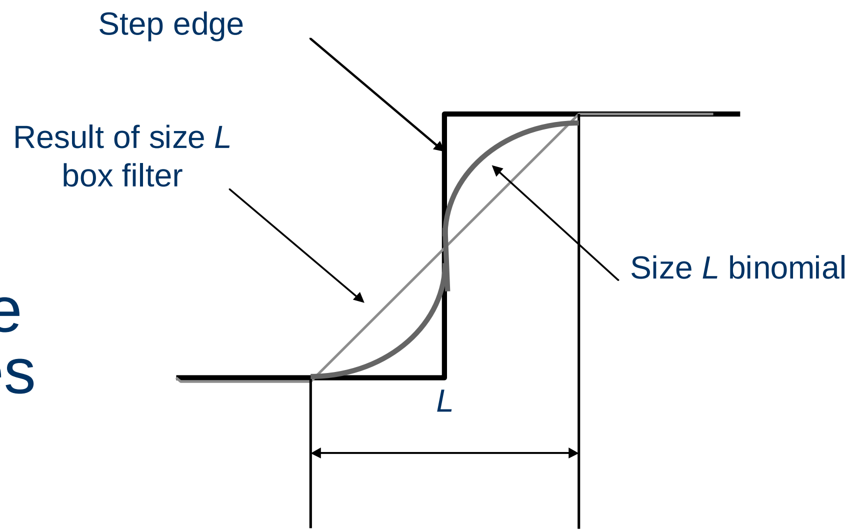


Lena image filtered with box filter 5x5

Linear Image smoothing techniques

Binomial and box filters. Edge blurring comparison

- Linear filters have to compromise smoothing with edge blurring
- Optimization underlying assumptions are violated at edges
- Be careful what you wish J …
- It might come true L



Nonlinear image smoothing

Conditional mean

- Pixels in a neighbourhood are averaged only if they differ from the central pixel by less than a given threshold:

$$g(m,n) = \sum_{k=-L}^{L} \sum_{l=-L}^{L} h(k,l) f(m-k, n-l),$$

$$h(k,l) = \begin{cases} 1, & if \quad | f(m-k,n-l) - f(m,n) | \; < \; th \\ 0, & otherwise \end{cases}$$

L is a **space** scale parameter and th is a **range** scale parameter

Nonlinear image smoothing
Conditional mean

- Example with L=3, th=32

Nonlinear image smoothing

Bilateral filter [Tomasi 1998]

- **Space** and **range** are treated in a similar way
- Space and range similarity is required for the averaged pixels
- Tomasi and Manduchi [1998] introduced soft weights to penalize the space and range dissimilarity.

$$h(k,l) = s(k,l)r(f(m-k,n-l)-f(m,n))$$

$$g(m,n) = \frac{1}{K}\sum_{k}\sum_{l} h(k,l)\, f(m-k,n-l),$$

$$K = \sum_{k}\sum_{l} h(k,l)$$

s() and r() are space and range similarity functions (Gaussian functions of the Euclidian distance between their arguments).

Nonlinear image smoothing

Bilateral filter

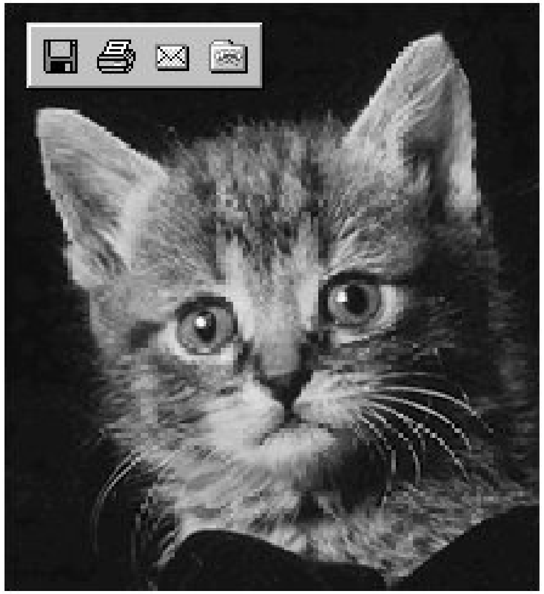
- The filter can be seen as weighted averaging in the joint space-range space (3D for monochromatic images and 5D – x,y,R,G,B - for colour images)
- The vector components are supposed to be properly normalized (divide by variance for example)
- The weights are given by:

$$h(\mathbf{x}) = \exp\{-\frac{\|\mathbf{x}_c - \mathbf{x}\|^2}{s}\}$$

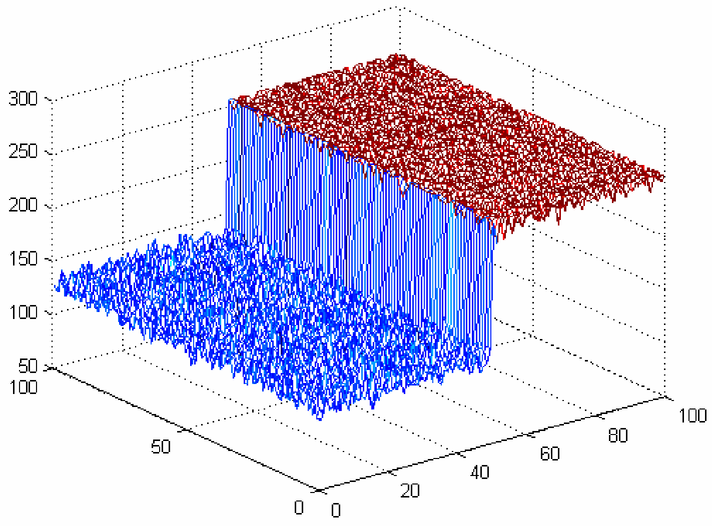
$$h(\mathbf{x}) = \mathrm{K}(\mathrm{d}(\mathbf{x}_c - \mathbf{x}); s)$$

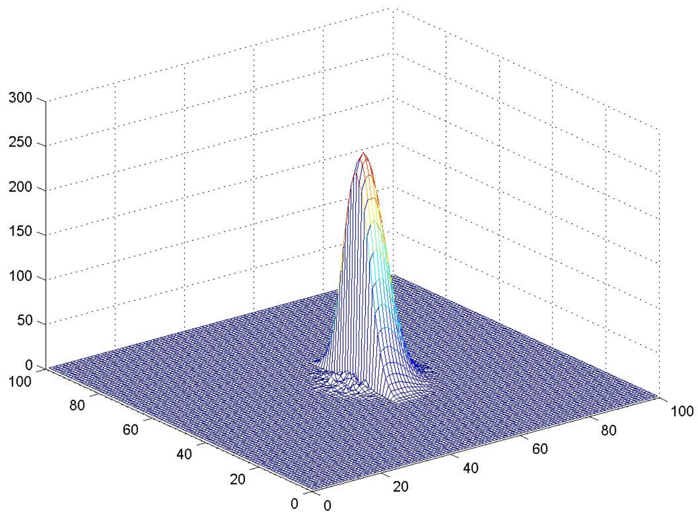
Nonlinear image smoothing
Bilateral filter

- Example of Bilateral filtering
- Low contrast texture has been removed
- Yet edges are well preserved



Bilateral filtering
Step edge image





Left: noisy step image. Right: kernel weights near the edge

Nonlinear image smoothing

Mean shift filtering [Comaniciu 1999, 2002]

- Mean shift filtering replaces each pixel's value with the most probable local value, given the observed pixel.
- The multivariate pdf can be found by a nonparametric probability density estimation method.
- The closest maxima of the pdf to the current pixel is found through the mean shift algorithm without having to estimate the whole pdf.
- The mean shift filter is related to the bilateral filter

$P \approx p(\mathbf{x})\int_R d\mathbf{x} = p(\mathbf{x})V$

Mean shift filtering

Brief introduction to nonparametric density estimation

$$S = \{\mathbf{x}_i\}_{i=1\ldots n} \qquad \mathbf{x}_i \in R^d$$

Image pixels =data points

The probability that $\mathbf{x}$ belongs to a sub-domain D of the space is

$$P = \int_D p(\mathbf{x})d\mathbf{x}$$

Probability density function (pdf)

If D is small, the pfd is fairly constant inside, so:

$$P \approx p(\mathbf{x})\int_D d\mathbf{x} = p(\mathbf{x})V$$

This leads to a pdf estimate at $\mathbf{x}$, inside the small domain D as

$$\hat{p}(\mathbf{x}) = \frac{P}{V} = \frac{\int_D p(\mathbf{y})d\mathbf{y}}{\int_D d\mathbf{y}}$$

Mean shift filtering
Brief introduction to nonparametric density estimation

The best size of the domain *D* is an important issue.
•If the volume V tends to 0, the estimate is infinite at all data sample points and 0 elsewhere (Dirac pulses).
•If the volume V tends to infinity, the estimate is the same (constant) everywhere.
•None of these extremes is desirable
•Statistics literature solutions [Duda 2000]:
  •Parzen window – V=1/sqrt (n)
  •knn estimator – inflate D until obtaining k samples

$$P \approx p(\mathbf{x})\int_R d\mathbf{x} = p(\mathbf{x})V$$

Mean shift filtering
Brief introduction to nonparametric density estimation

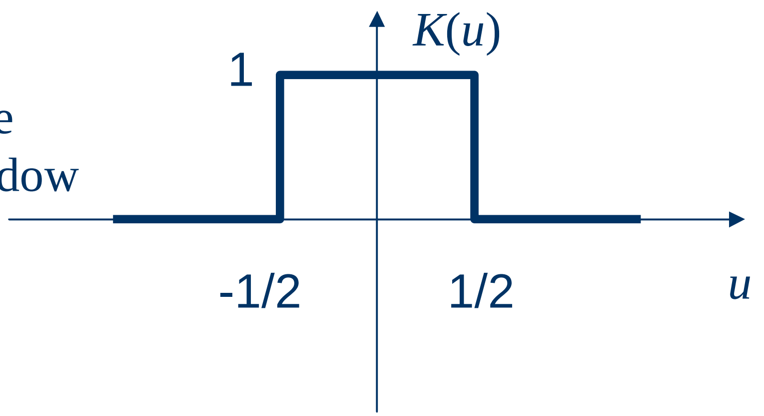
The Parzen estimator is using a hipercube shaped window of radius h, having the volume

$$V = h^d$$

Define the Parzen window function:

$$K(\mathbf{u}) = \begin{cases} 1, & if \quad |u_i| \leq 1/2, i = 1,2,\ldots,d \\ 0 & otherwise \end{cases}$$

1D example: the rectangular window



Mean shift filtering
Brief introduction to nonparametric density estimation

The number of samples in a hypercube with edge length h centered on $\mathbf{x}$ is

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

The multivariate kernel density estimate obtained at the point $\mathbf{x}$ with the kernel $K(\mathbf{x})$ and window radius h is:

$$\hat{p}(\mathbf{x}) = \frac{k}{V} = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{\mathbf{x} - \mathbf{x_i}}{h}\right)$$

Note, the estimate is continuous:
$\mathbf{x}$ is not supposed to belong to the data set, $\boldsymbol{S}$

Mean shift filtering

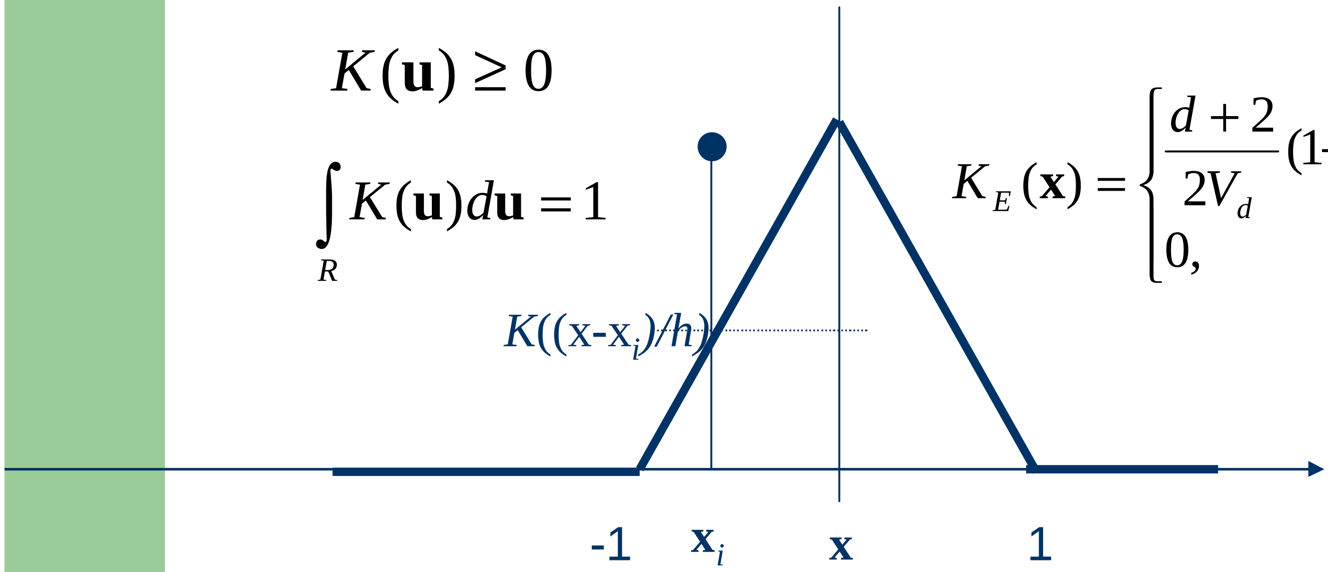
Brief introduction to nonparametric density estimation

The last equation suggests a generalization: $K()$ can have different shapes.
Mild conditions: $K()$ has to be nonnegative and integrate to 1.

$$K(\mathbf{u}) \geq 0$$

$$\int_R K(\mathbf{u})d\mathbf{u} = 1$$

$$K_E(\mathbf{x}) = \begin{cases} \frac{d+2}{2V_d}(1 - \| x \|^2), & \| \mathbf{x} \| \leq 1 \\ 0, & \| \mathbf{x} \| > 1 \end{cases}$$



Note: $p(\mathbf{x})$ is a superposition (weighted average) of the contributions of all saples at $\mathbf{x}$. Example: Epanechnikov kernel (minimize MISE).
Also Gaussian window (trunctated) used frequently.

Mean shift filtering [Comaniciu 1999, 2002]

For the Epanechnikov kernel, the estimated normalized density gradient is proportional to the *mean shift*:

$$\frac{h^2}{d+2}\frac{\hat{\nabla} f(\mathbf{x})}{\hat{f}(\mathbf{x})} = m_h(\mathbf{x}) = \frac{1}{n_{\mathbf{x}}}\sum_{\mathbf{x}_i \in S_h(\mathbf{x})} \mathbf{x_i} - \mathbf{x}$$

S is a sphere of radius h, centered on $\mathbf{x}$ and n_{x} is the number of samples inside the sphere. More generally, for any kernel,

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

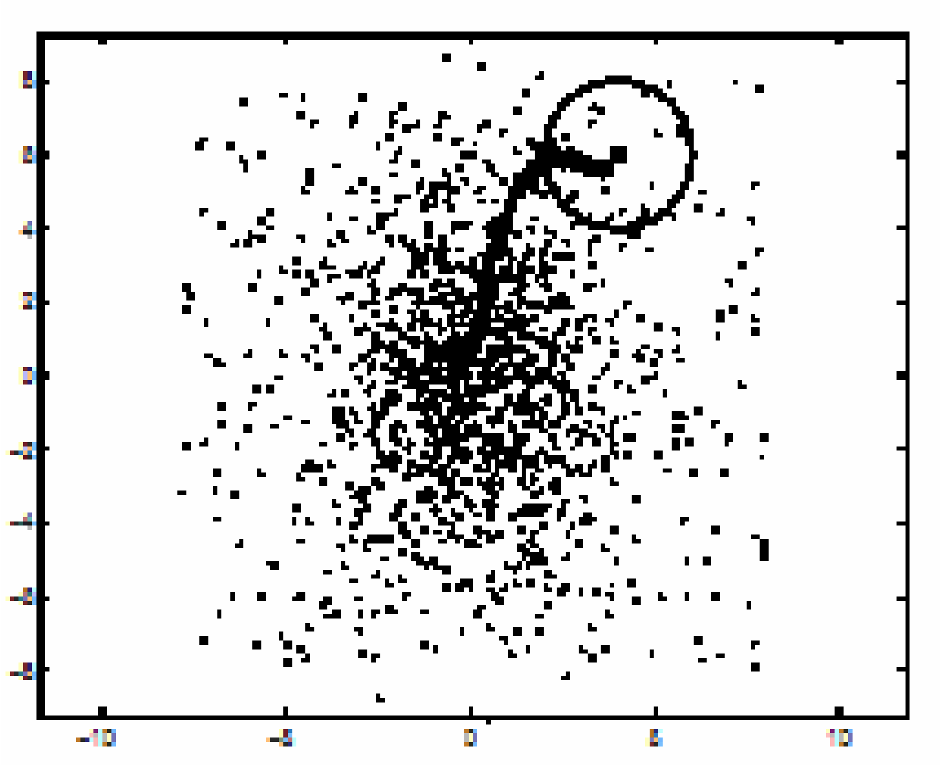
$$K(\mathbf{x}) = c_{k,d}\,k(\|\mathbf{x}\|^2),$$

$$g(x) = -k'(x).$$

$k(x)$ is called the profile of kernel $K(\mathbf{x})$.

Mean shift filtering [Comaniciu 1999, 2002]

- The mean shift procedure is a gradient ascent method to find local modes (maxima) of the probability density and is guaranteed to converge.
- Step1: compute of the mean shift vector $\mathbf{m}_h(\mathbf{x})$.
- Step2: translate the window by $\mathbf{m}_h(\mathbf{x})$.
- Iterations start for each pixel (5D point) and tipically converge in 2-3 steps.



Mean shift filtering [Comaniciu 1999, 2002]

The *mean shift algorithm*:

Compute the closest local mode of the pdf to any location, $\mathbf{x}$

Set initially $\mathbf{y}_1 = \mathbf{x}$ and $j=1$, then:

Step1: compute

$$\mathbf{y}_{j+1} = \frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\left\|\frac{\mathbf{y}_j - \mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{\mathbf{y}_j - \mathbf{x}_i}{h}\right\|^2\right)}.$$

granted

Step2: Make $j=j+1$and repeat step 1 until convergence: $\|\mathbf{m}(\mathbf{y})\| = \|\mathbf{y}_{j+1} - \mathbf{y}_j\| <$ epsilon. Then $\mathbf{y}_{j+1} = \mathbf{y}_c$ is the location of the pdf local maxima closest to $\mathbf{x}$.

Mean shift filtering: change each $\mathbf{x}_i$ to the corresponding $\mathbf{y}_c$ obtained by through mean shift algorithm.

Mean shift filtering [Comaniciu 1999, 2002]

- Example1.

Mean shift filtering examples

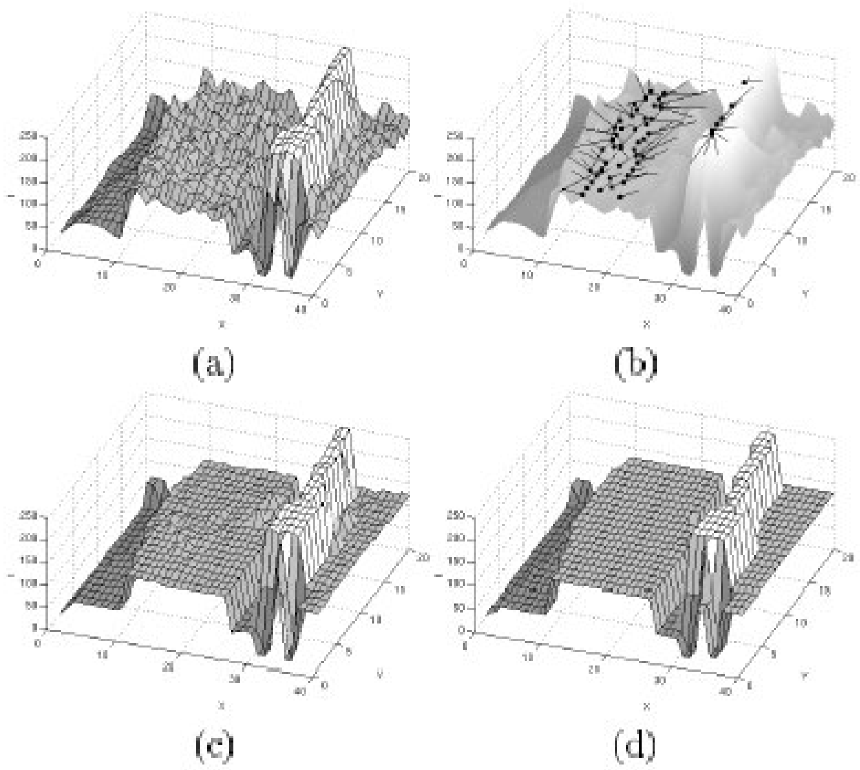
Detail of a 24x40 window from the cameraman image

a) Original data

b) Mean shift paths for some points

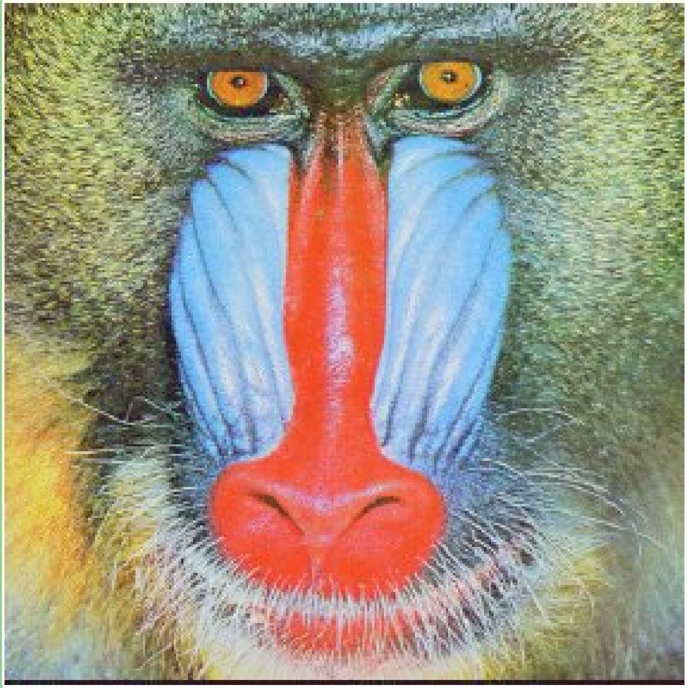
c) Filtered data

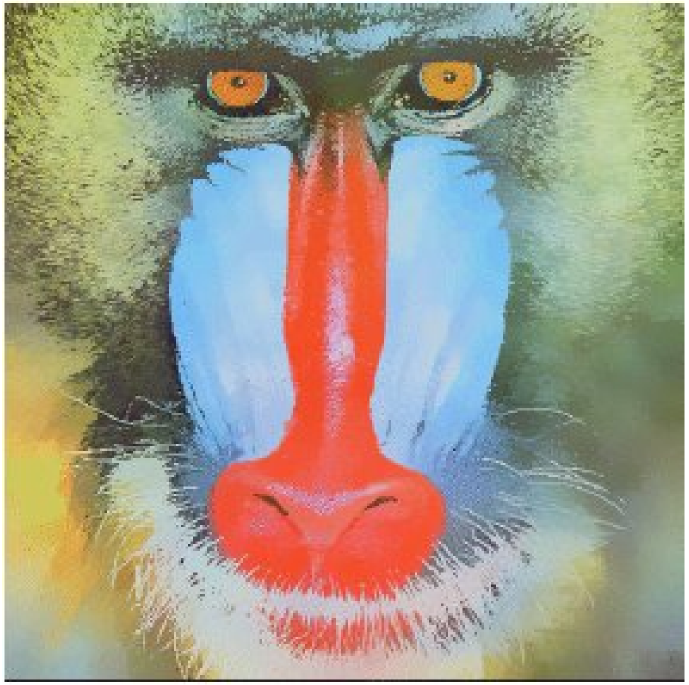
d) Segmented data



Mean shift filtering examples

- Example 2





Comparison to bilateral filtering

- Both methods based on simultaneous processing of both the spatial and range domains
- While the bilateral filtering uses a static window, the mean shift window is dynamic, moving in the direction of the maximum increase of the density gradient.

Mean shift segmentation

Step1. Mean shift filter the image

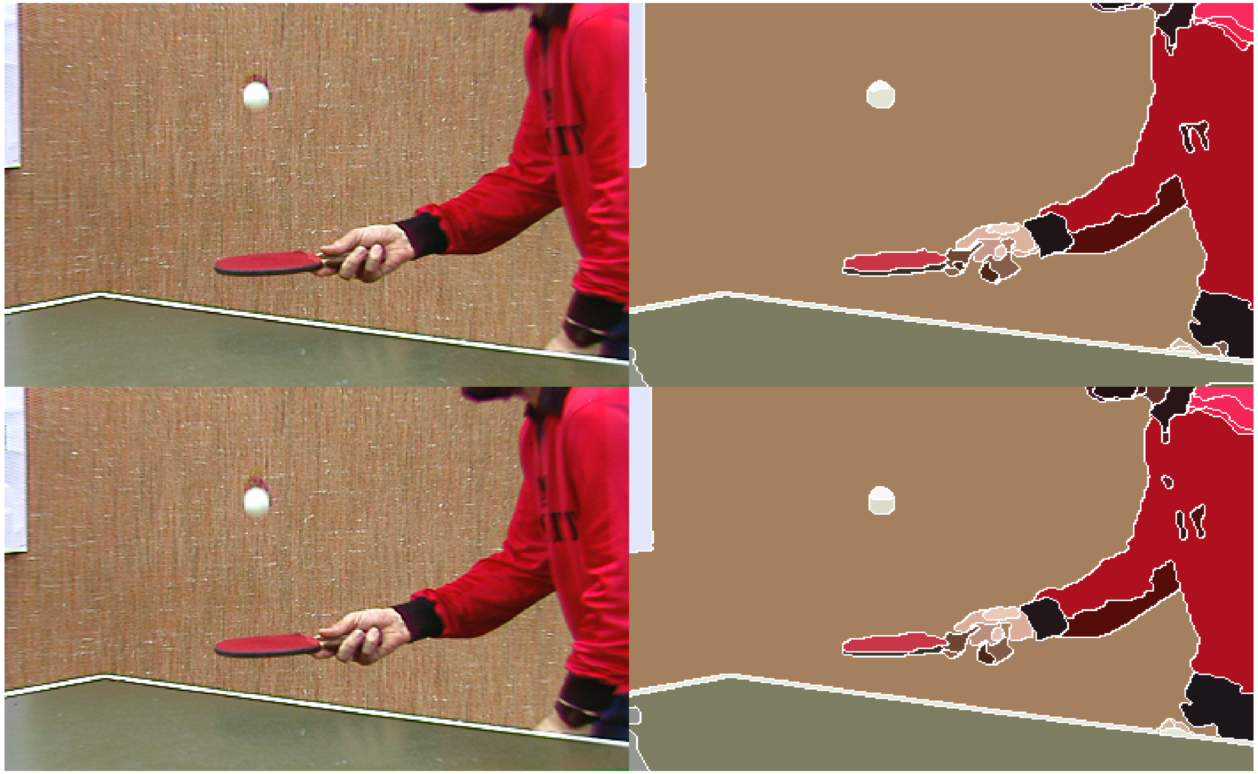
Step2. Link all pixels converging to the same mode.

Optional: merge regions with close modes

Optional: clear small regions

Segmentation example

Video image segmentation



Background estimation and subtraction for video surveillance

Thank you for your attention!

J

REFERENCES

- D. Comaniciu, P. Meer: Mean shift analysis and applications. 7th International Conference on Computer Vision, Kerkyra, Greece, Sept. 1999, 1197-1203.
- D. Comaniciu, P. Meer: Mean shift: a robust approach toward feature space analysis. IEEE Trans. on PAMI Vol. 24, No. 5, May 2002, 1-18.
- R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification. 2nd edition, Wiley 2000.
- M. Elad: On the origin of bilateral filter and ways to improve it. IEEE trans. on Image Processing Vol. 11, No. 10, October 2002, 1141-1151.
- B. Jahne: Digital image processing. Springer Verlag, Berlin 1995.
- W.K. Pratt: Digital image processing. John Wiley and sons, New York 1991
- C. Tomasi, R. Manduchi: Bilateral filtering for gray and color images. Proc. Sixth Int'l. Conf. Computer Vision , Bombay, 839-846.