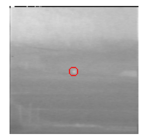
Lecture-9

Target Tracking Using Mean Shift

Alper YILMAZ
Computer Vision Lab.
University of Central Florida

Problem

- Given a target (object) in the first frame track it through all other frames.



Key Concepts

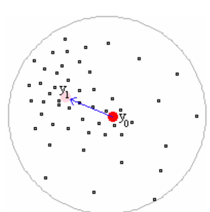
- Mean
- Mean shift
- Probability Density Functions (PDFs)
- Kernel Density
- Gradient of Kernel Density
- Bhatacharya coefficient

Mean Shift Vector

Given:
Data points and approximate location of the mean of this data.

Task:
Estimate the exact location of the mean of the data by determining the shift vector from initial mean.

Mean Shift Vector Example



$$M_h(\mathbf{y}) = \left[\frac{1}{n_x} \sum_{i=1}^{n_x} (\mathbf{x}_i - \mathbf{y}_0) \right]$$

n_x : number of points
$\mathbf{y}_0$: initial mean location
$\mathbf{x}_i$: data points

Mean shift vector always points towards the direction of the maximum increase in the density.

Modified Mean Shift (weighted)

$$M_h(\mathbf{y}_0) = \left[\frac{\sum_{i=1}^{n_x} w_i(\mathbf{y}_0)\mathbf{x}_i}{\sum_{i=1}^{n_x} w_i(\mathbf{y}_0)} \right] - \mathbf{y}_0$$

n_x : number of points in the kernel
$\mathbf{y}_0$: initial mean location
$\mathbf{x}_i$: data points
h : kernel radius

Weights are determined using kernels (masks):
Uniform, Gaussian or Epanechnikov

UCF Computer Vision Lab. 7

Properties of Mean Shift

- Mean shift vector has the direction of the gradient of the density estimate.

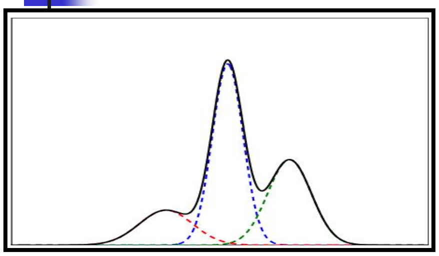
- It is computed iteratively for obtaining the maximum density in the local neighborhood.

UCF Computer Vision Lab. 8

Probability Density Functions (PDFs)

- Parametric models
  - Uniform
  - Gaussian
  - Exponential
  - ...
- Some PDFs can not be modeled by parametric models

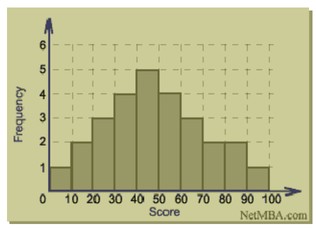
Probability Density Functions (PDFs)



Mixture of Gaussians

Histogram

- The histogram records the count of data points falling in different ranges, called bins.
- It captures the frequency distribution of the data.

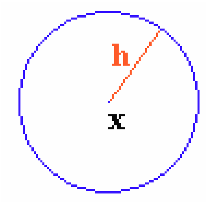


Kernel Density Estimation

- Kernel Density estimate can be used to represent any non-parametric pdf. (general)
- All data points are saved. (large storage)
- The probability of any given value is calculated by using all the data points.

Kernel Density Estimate

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



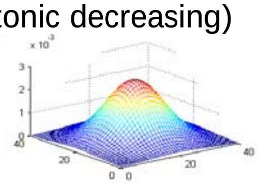
n : number of points in the kernel
h : window radius
x : mean vector
d : number of dimensions
K : Kernel density function

Possible Kernels

- Uniform kernel
- Normal kernel (convex, monotonic decreasing)

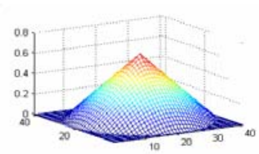
$$K_N = (2\pi)^{-d/2} e^{-\|\mathbf{x}\|^2/2}$$

d : number of dimensions



- Epanechnikov kernel (convex, monotonic decreasing)

$$K_E(\mathbf{x}) = \begin{cases} \frac{1}{2} c_d^{-1} (d+2)(1-\|\mathbf{x}\|^2) & if\ \|\mathbf{x}\| < 1 \\ 0 & otherwise \end{cases}$$

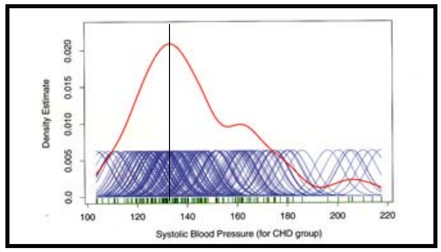


c_d : volume of unit d-dim sphere
d : number of dimensions

Kernel Density Estimation

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

x



Kernel Density

UCF Computer Vision Lab. 15

Estimate of Density Gradient

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

gradient of density estimate: $$\hat{\nabla} f(\mathbf{x}) \equiv \nabla \hat{f}(\mathbf{x}) = \frac{1}{nh^d}\sum_{i=1}^{n} \nabla K\left(\frac{\mathbf{x}-\mathbf{x}_i}{h}\right)$$

UCF Computer Vision Lab. 16

Mean Shift Vector in Terms of Epanechnikov Kernel

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

Using Epanechnikov kernel: $K_E(\mathbf{x}) = \frac{1}{2} c_d^{-1} (d+2)(1-\|\mathbf{x}\|^2)$

Homework

$$\hat{\nabla} f(\mathbf{x}) = \frac{d+2}{nh^{d+2} c_d} n \boxed{\left(\frac{1}{n} \sum_{x_i \in S_h(x)}^{n} [\mathbf{x}_i - \mathbf{x}]\right)} = \frac{d+2}{h^{d+2} c_d} \boxed{M_h(x)}$$

mean shift vector

n : number of points in unit d-dimensional sphere

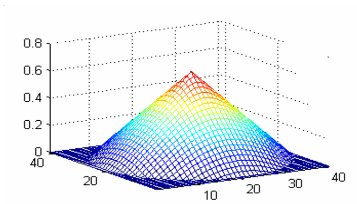
UCF Computer Vision Lab. 17

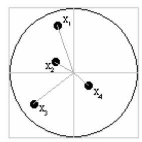
Target Model for Tracking

- Features used for tracking include:
  - Gray level
- Feature probability distribution are calculated by using weighted histograms.
- The weights are derived from Epanechnikov kernel.

UCF Computer Vision Lab. 18

Target Model for Tracking



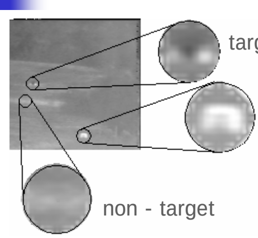


x_1, x_2, x_3, x_4 has the same value of the feature, such as gray level, *u*.

$$p(u) = C \sum_{\mathbf{x}_i \in S} K\left(\left\| \frac{\mathbf{x}_i - \mathbf{y}}{h} \right\|^2 \right) \delta[\ S(\mathbf{x}_i)\ - u]$$

UCF Computer Vision Lab. 19

Target Gray Level Feature



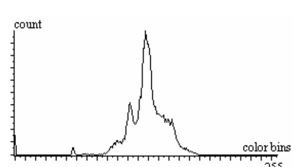
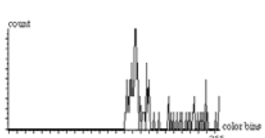
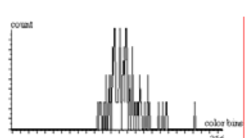


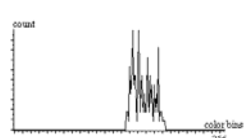
image histogram



target 1 distribution



target 2 distribution



non target distribution

UCF Computer Vision Lab. 20

Similarity of Target and Candidate Distributions

Target : q_u.
Candidate : $\hat{p}_u$.

$$d(\mathbf{y}) = \sqrt{1-\rho(\mathbf{y})}$$

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

$\rho(\mathbf{y})$:Bhattacharya coefficient.

Distance Minimization

Minimizing the distance corresponds to *maximizing* Bhattacharya coefficient.

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

Taylor expansion around $\hat{p}(\mathbf{y}_0)$

Homework

$$\rho[\hat{\mathbf{p}}(\mathbf{y}), \mathbf{q}] \cong \rho[\hat{\mathbf{p}}(\mathbf{y}_0), \mathbf{q}] + \frac{1}{2}\sum_{i=1}^{m} \hat{p}_u(\mathbf{y}) \sqrt{\frac{q_u}{\hat{p}_u(\mathbf{y}_0)}}$$

Maximizing Bhattacharya coefficient can be obtained by ***maximizing the blue term***.

Likelihood Maximization

$$\rho[\hat{\mathbf{p}}(\mathbf{y}),\mathbf{q}] \cong \rho[\hat{\mathbf{p}}(\mathbf{y}_0),\mathbf{q}] + \frac{1}{2}\sum_{i=1}^{m}\hat{p}_u(\mathbf{y})\sqrt{\frac{q_u}{\hat{p}_u(\mathbf{y}_0)}}$$

$$\downarrow$$

$$\frac{C_h}{2}\sum_{i=1}^{n_x}\left[\sum_{u=1}^{m}\delta[S(\mathbf{x}_i)-u]\sqrt{\frac{q_u}{\hat{p}_u(\mathbf{y}_0)}}\right]k\left(\left\|\frac{\mathbf{y}-\mathbf{x}_i}{h}\right\|\right)$$

h : radius of sphere
C_h : normalization constant
$S(x_i)$: gray level at x
y : kernel center
m : number of bins

likelihood maximization depends on maximizing w_i.

Likelihood Maximization Using Mean Shift Vector

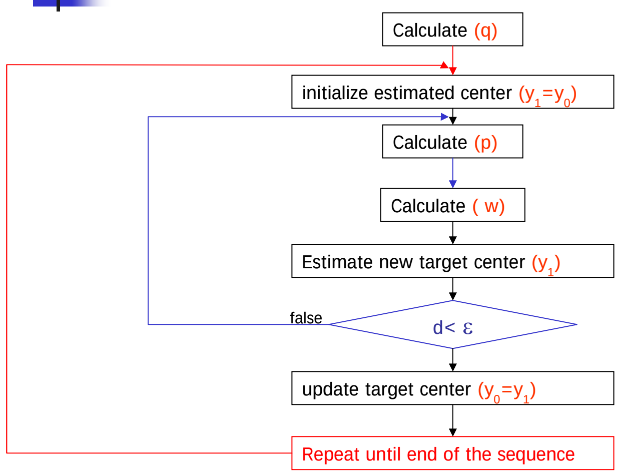
Maximization of the likelihood of target and candidate depends on the weights:

$$w_i(\mathbf{y}_o) = \sum_{u=1}^{m}\delta[S(\mathbf{x}_i)-u]\sqrt{\frac{q_u}{\hat{p}_u(\mathbf{y}_o)}} \quad where \quad 0 \le w_i \le 1$$

$$M_h(\mathbf{y}_0) = \frac{\sum_{i=1}^{n_x} w_i(\mathbf{y}_0)\mathbf{x}_i}{\sum_{i=1}^{n_x} w_i(\mathbf{y}_0)} - \mathbf{y}_0$$

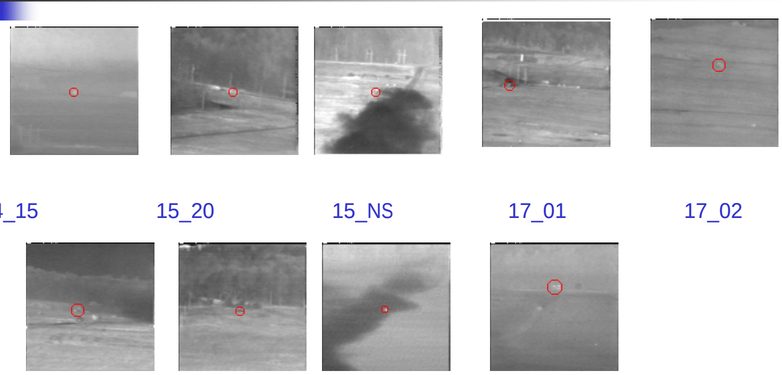
Thus, new target center is $\hat{\mathbf{y}} = \mathbf{y}_0 + M_h(\mathbf{y}_0)$

Algorithm



UCF Computer Vision Lab. 25

Tracking Results



14_15 15_20 15_NS 17_01 17_02

17_20 18_03 18_05 19_07

UCF Computer Vision Lab. 26

Papers

- D. Comaniciu, V. Ramesh, P. Meer, "Kernel-Based Object Tracking", IEEE Trans. Pattern Analysis Machine Intell., Vol. 25, No. 5, 2003.
  - http://www.cs.ucf.edu/courses/cap6412/2003/Kernel-based%20object%20tracking.pdf

- Target-Tracking in Airborne Forward Looking Infrared Imagery
  Image and Vision Computing Journal, Vol. 21, No. 7, 2003, pp. 623-635.
  - http://www.cs.ucf.edu/~vision/papers/yilmaz_ivc_2002.pdf

UCF Computer Vision Lab. 27