CAP 5415 Computer Vision
Fall 2005

Dr. Alper Yilmaz

Univ. of Central Florida

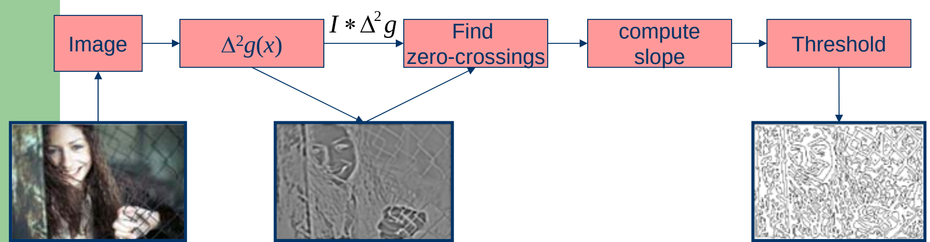
www.cs.ucf.edu/courses/cap5415/fall2005

Office: CSB 250

Recap (Edge Detection)

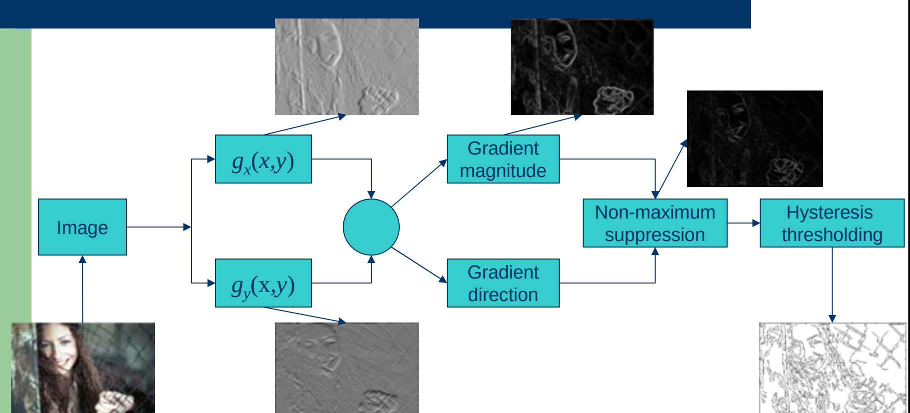
- Marr-Hildreth and Canny edge detectors
  - Gaussian smoothing
  - Compute 2nd order derivatives
    - In x and y directions
  - Find zero crossings
  - Threshold zero crossings
- Difference between Marr-Hildreth and Canny
  - Marr-Hildreth use 2nd order derivative
  - Marr-Hildreth thresholds slope of zero-crossings

Marr-Hildreth Edge Detector



Alper Yilmaz, Fall 2005 UCF

Canny Edge Detector



Alper Yilmaz, Fall 2005 UCF

Region Segmentation

Alper Yilmaz, Fall 2005 UCF

Applications of Segmentation

- Object recognition
- MPEG-4 video compression

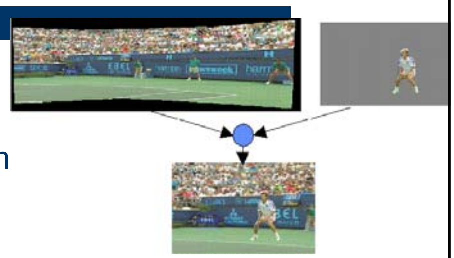
Alper Yilmaz, Fall 2005 UCF

Object Recognition Using Region Properties

- Training
  - For all training samples of each model object
    - Segment the image
    - Compute region properties (features)
- Recognition
  - Given an image of unknown object,
    - Segment the image
    - Compute its feature vector
    - Compare with the training set

Alper Yilmaz, Fall 2005 UCF

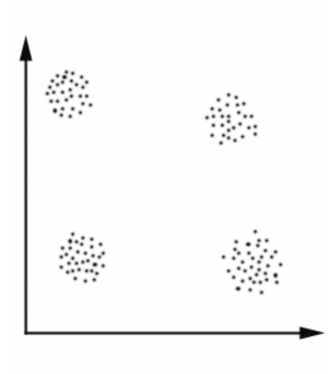
MPEG4 Compression Object Based Compression



- Advantages of OBC
  - High compression ratio
  - Allows insertion deletion of objects
- How does it work?
  - Find objects (*Object Segmentation*)
  - Code objects and their locations
  - Build mosaics of globally static objects
  - Render scene at receiver

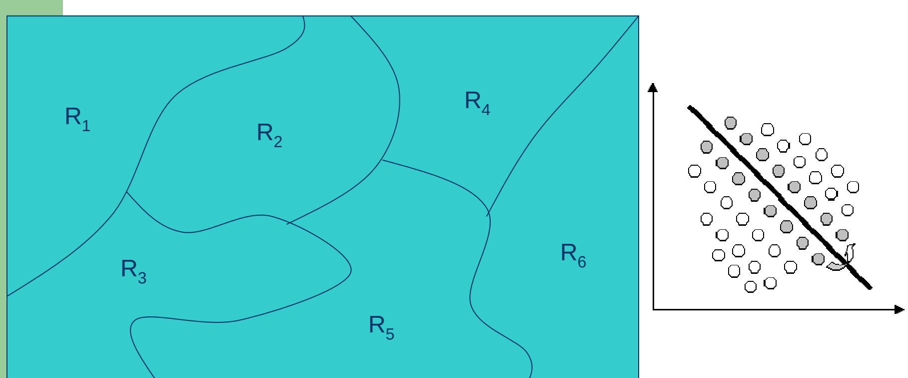
Alper Yilmaz, Fall 2005 UCF

Clustering



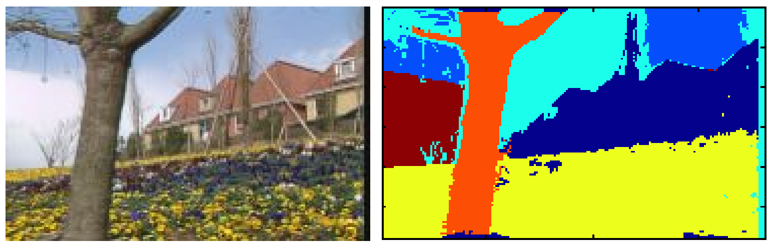
Alper Yilmaz, Fall 2005 UCF

Segmentation-Clustering



Alper Yilmaz, Fall 2005 UCF

Region Segmentation



Alper Yilmaz, Fall 2005 UCF

Layer Representation

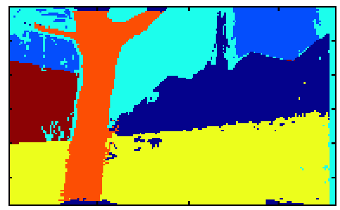
UCF

Segmentation

- Find set of regions $R_1, R_2, \ldots, R_n$ such that

$$\bigcup_{i=1}^{n} R_i = I \qquad \forall i \neq j, R_i \cap R_j = \varnothing$$

- All pixels in region i satisfy some similarity constraint



Alper Yilmaz, Fall 2005 UCF

Similarity Constraints

- All pixels in any sub-image musts have the **same** gray levels.
- All pixels in any sub-image **must not differ** more than some threshold
- All pixels in any sub-image **may not differ** more than some threshold from the mean of the gray of the region
- The **standard deviation** of gray levels in any sub-image must be small.

Alper Yilmaz, Fall 2005 UCF

Simple Segmentation

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

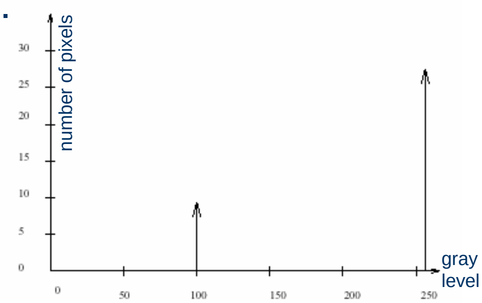
$$B(x,y) = \begin{cases} 1 & \text{if } T_1 < I(x,y) < T_2 \\ 0 & \text{Otherwise} \end{cases}$$

$$B(x,y) = \begin{cases} 1 & \text{if } I(x,y) \in Z \\ 0 & \text{Otherwise} \end{cases}$$

Image Histogram

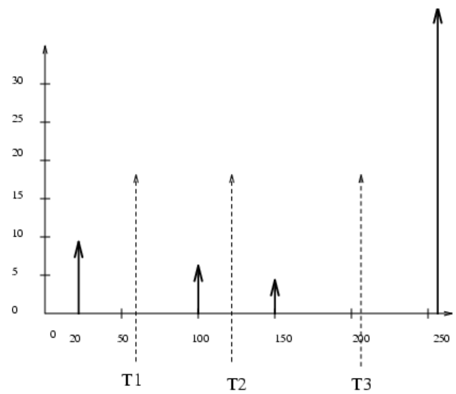
- Histogram graphs the number of pixels with a particular gray level as a function of the image of gray levels.

255	255	255	255	255	255
255	255	255	255	255	255
255	255	100	100	100	255
255	255	100	100	100	255
255	255	100	100	100	255
255	255	255	255	255	255



Segmentation Using Histogram Simple Case

255	255	255	255	255	255	255	20
255	255	255	100	100	255	20	20
255	255	255	100	100	255	20	20
255	255	255	100	100	255	20	20
255	255	255	255	255	255	20	20
255	255	255	255	255	255	255	255
150	150	255	255	255	255	255	255
150	150	255	255	255	255	255	255



Segmentation Using Histogram Simple Case

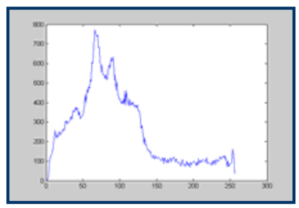
$$B_1(x,y) = \begin{cases} 1 & \text{if } 0 < f(x,y) < T_1 \\ 0 & \text{Otherwise} \end{cases}$$

$$B_2(x,y) = \begin{cases} 1 & \text{if } T_1 < f(x,y) < T_2 \\ 0 & \text{Otherwise} \end{cases}$$

$$B_3(x,y) = \begin{cases} 1 & \text{if } T_2 < f(x,y) < T_3 \\ 0 & \text{Otherwise} \end{cases}$$

Realistic Histograms

Not realistic

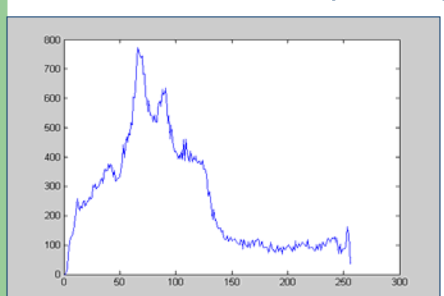


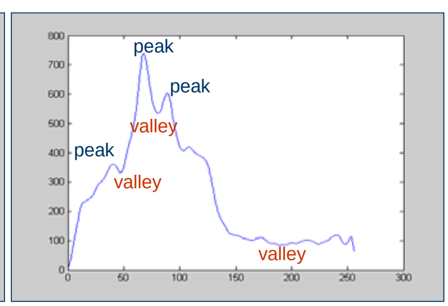
Real (noise)

Alper Yilmaz, Fall 2005 UCF

Realistic Histograms

- Smooth out noise
  - Convolve hist. by averaging or 1D Gaussian filter



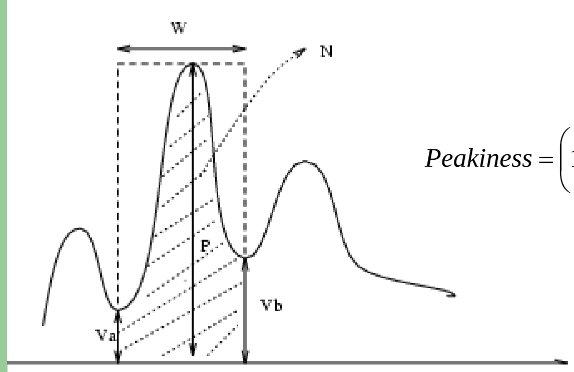


Alper Yilmaz, Fall 2005 UCF

Segmentation Using Histogram
Real image histograms

1. Compute the histogram of a given image.
2. Smooth the histogram by averaging peaks and valleys in the histogram.
3. **Detect good peaks by applying thresholds at the valleys.**
4. Segment the image into several binary images using thresholds at the valleys.
5. **Apply connected component algorithm to each binary image find connected regions.**

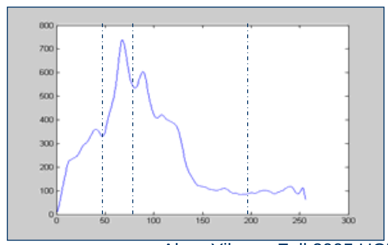
Good Peaks
Peakiness Test



$$Peakiness = \left(1 - \frac{(V_a + V_b)}{2P}\right) \cdot \left(1 - \frac{N}{(W.P)}\right)$$

Segmentation Using Histograms

- Select the valleys as thresholds
  - Apply threshold to histogram
  - Label the pixels within the range of a threshold with same label, i.e., a, b, c … or $1, 2, 3$ …



Alper Yilmaz, Fall 2005 UCF

Connected Components

- Disjoint segments with same labels need to be split

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 & 0 & 0 & a & 0 \\ b & b & 0 & a & a \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c & c & 0 \\ 0 & d & 0 & c & 0 \end{bmatrix}$$

may be added to segment c

Alper Yilmaz, Fall 2005 UCF

Recursive Connected Component Algorithm

1. Scan the binary image left to right, top to bottom.
2. If there is an unlabeled pixel with a value of '1' assign a new label to it.
3. Recursively check the neighbors of the pixel in step 2 and assign the same label if they are unlabeled with a value of '1'.
4. Stop when all the pixels of value '1' have been labeled.

Figure 3.7: Recursive Connected Component Algorithm.

Sequential Connected Component Algorithm

1. Scan the binary image left to right, top to bottom.
2. If an unlabeled pixel has a value of '1', assign a new label to it according to the following rules:

$$\begin{array}{cc} & 0 \\ 0 & 1 \end{array} \rightarrow \begin{array}{cc} & 0 \\ 0 & L \end{array} \qquad \begin{array}{cc} & 0 \\ L & 1 \end{array} \rightarrow \begin{array}{cc} & 0 \\ L & L \end{array}$$

$$\begin{array}{cc} & L \\ 0 & 1 \end{array} \rightarrow \begin{array}{cc} & L \\ 0 & L \end{array} \qquad \begin{array}{cc} & L \\ M & 1 \end{array} \rightarrow \begin{array}{cc} & L \\ M & L \end{array} \text{ (Set } L = M\text{).}$$

3. Determine equivalence classes of labels.
4. In the second pass, assign the same label to all elements in an equivalence class.

Figure 3.8: Sequential Connected Component Algorithm.

Sequential Connected Component Algorithm

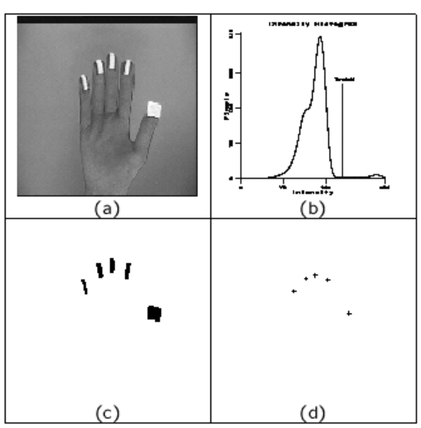
$$\begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 0 & a & 0 \\ b & b & 0 & a & a \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c & c & 0 \\ 0 & d & c & c & 0 \end{bmatrix}$$

$d=c$

Equivalence class

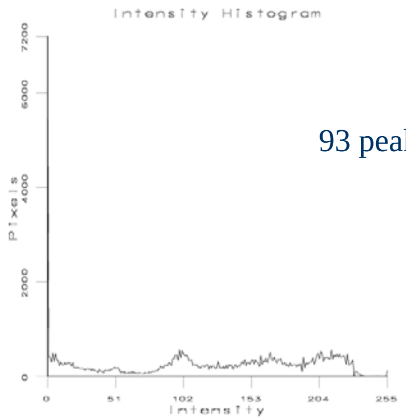
Alper Yilmaz, Fall 2005 UCF

Example
Detecting Finger Tips (marked white)



Alper Yilmaz, Fall 2005 UCF

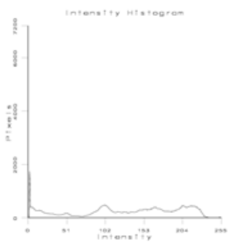
Example
Segmenting a bottle image



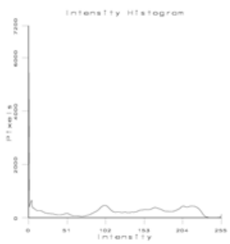
93 peaks

Alper Yilmaz, Fall 2005 UCF

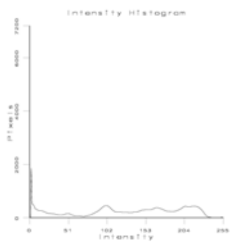
Example
Segmenting a bottle image



Smoothed histogram
(averaging using mask
Of size 5)
54 peaks (once)
After peakiness 18



Smoothed histogram
21 peaks (twice)
After peakiness 7

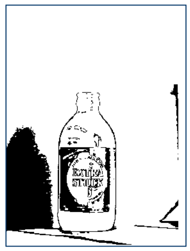


Smoothed histogram
11 peaks (three times)
After peakiness 4

Alper Yilmaz, Fall 2005 UCF

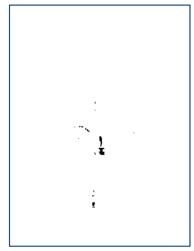
Example
Segmenting a bottle image

(0,40)



(40, 116)

(116,243)



(243,255)

Exercise

- Implement histogram based region segmentation (histogram computation, peakiness test, recursive connected component analysis).
- Report should include: Histograms, resulting region as binary images with various histograms generated with several smoothing operations and hard copy of the source code.
  - Smooth once find peaks.
  - Smooth twice find peaks.
  - Smooth 5 times find peaks.
- Due date: 28 September 2005

Suggested Reading

- Chapter 3, Mubarak Shah, “Fundamentals of Computer Vision”