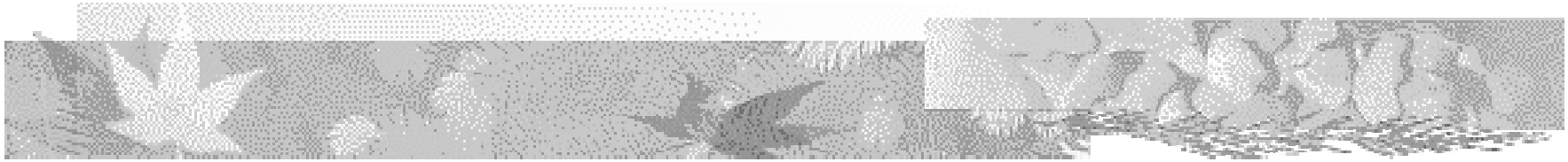
CAP5415 Computer Vision
Spring 2003

Khurram Hassan-Shafique



Region Segmentation

Region Segmentation

- Find sets of pixels, $R_1, R_2, \ldots, R_n$ such that
  - $\bigcup_i R_i = I$
  - $\forall i \neq j, R_i \cap R_j = \varnothing$
  - All pixels in region *i* satisfy some constraint of similarity.

K-Means

- Choose a fixed number of clusters
- Choose cluster centers and point-cluster allocations to minimize error
- can't do this by search, because there are too many possible allocations.

- Algorithm
  - fix cluster centers; allocate points to closest cluster
  - fix allocation; compute best cluster centers
- x could be any set of features for which we can compute a distance (careful about scaling)

$$\sum_{i \in \text{clusters}} \left\{ \sum_{j \in \text{elements of i'th cluster}} \left\| x_j - \mu_i \right\|^2 \right\}$$

K-Means

```
Choose k data points to act as cluster centers

Until the cluster centers are unchanged

    Allocate each data point to cluster whose center is nearest

    Now ensure that every cluster has at least
    one data point; possible techniques for doing this include .
    supplying empty clusters with a point chosen at random from
    points far from their cluster center.

    Replace the cluster centers with the mean of the elements
    in their clusters.

end
```

Algorithm 16.5: *Clustering by K-Means*

Image Segmentation by K-Means

- Select a value of K
- Select a feature vector for every pixel (color, texture, position, or combination of these etc.)
- Define a similarity measure between feature vectors (Usually Euclidean Distance).
- Apply K-Means Algorithm.
- Apply Connected Components Algorithm.
- Merge any components of size less than some threshold to an adjacent component that is most similar to it.

Image

Clusters on intensity

Clusters on color

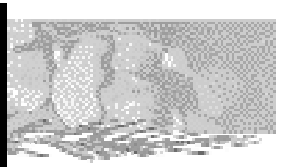
K-means clustering using intensity alone and color alone

Image

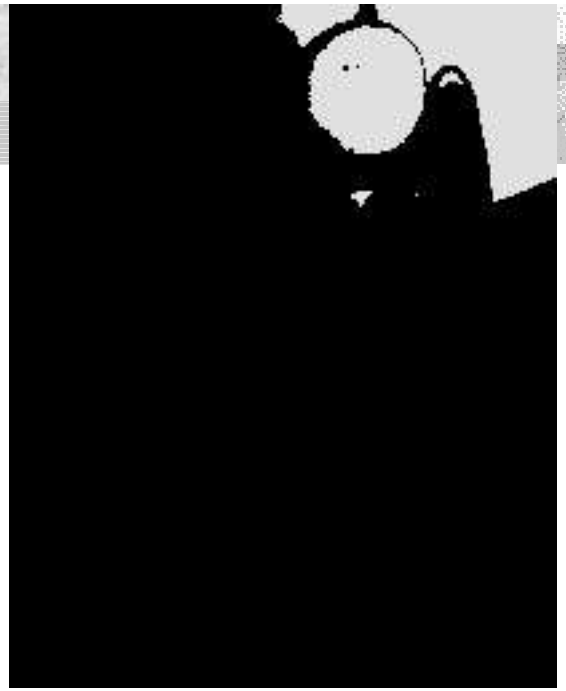
Clusters on color

K-means using color alone, 11 segments

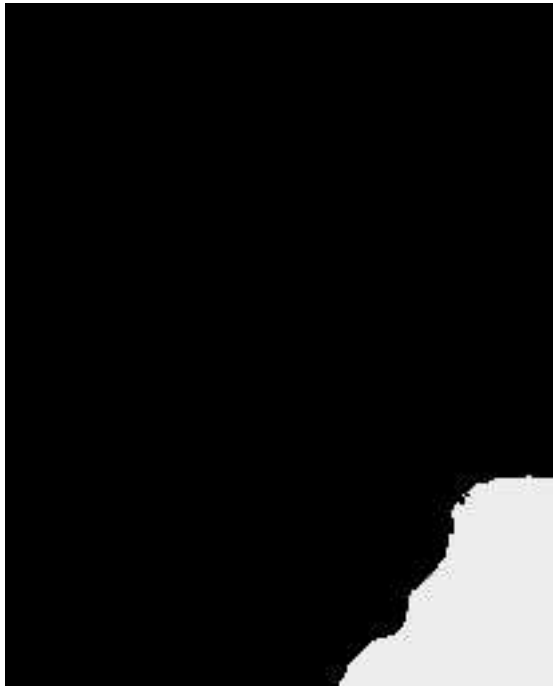
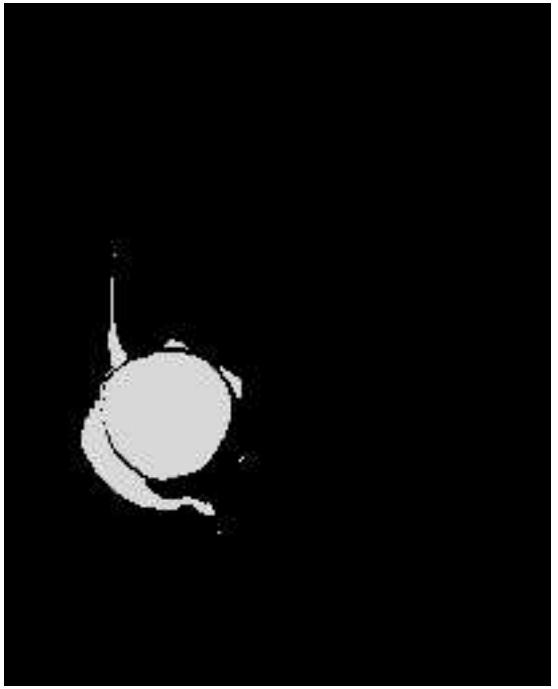


K-means using
color alone,
11 segments.

K-means using colour and position, 20 segments

How to find K

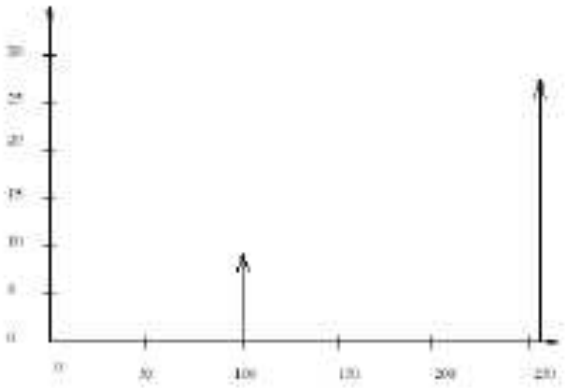
- Use prior knowledge about image.
- Apply the algorithm for different values of K and test for goodness of clusters.
- Analyze Image Histograms.

How to find K

Histogram

Histogram graphs the number of pixels in an image 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

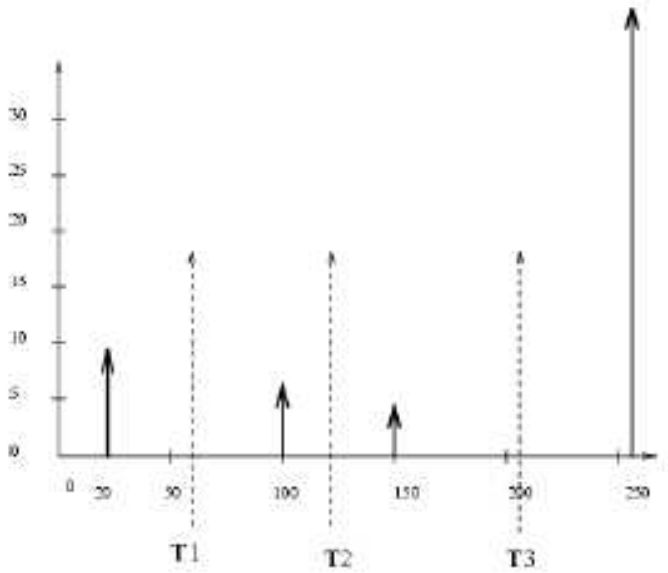


```
For (I=0, I<m, I++)
  For (J=0, J<m, J++)
    histogram[f(I,J)]++;
```

How to find K

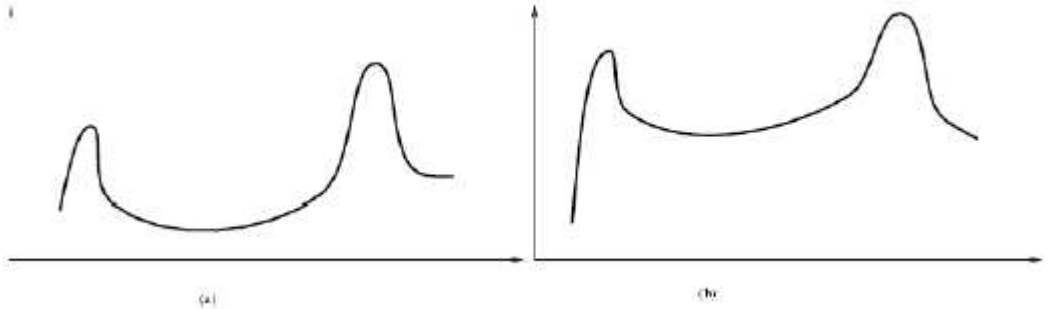
Example

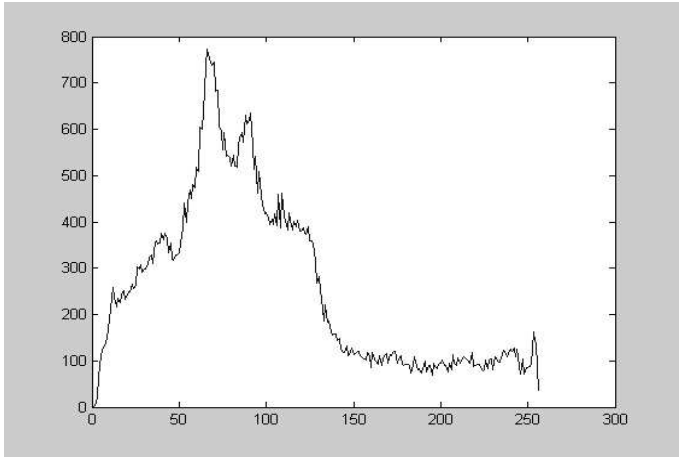
255	255	255	255	255	255	255	20
255	255	255	100	100	255	20	20
255	255	255	100	100	255	20	20
255	255	255	100	100	255	20	20
255	255	255	255	255	255	20	20
255	255	255	255	255	255	255	255
150	150	255	255	255	255	255	255
150	150	255	255	255	255	255	255



How to Find K

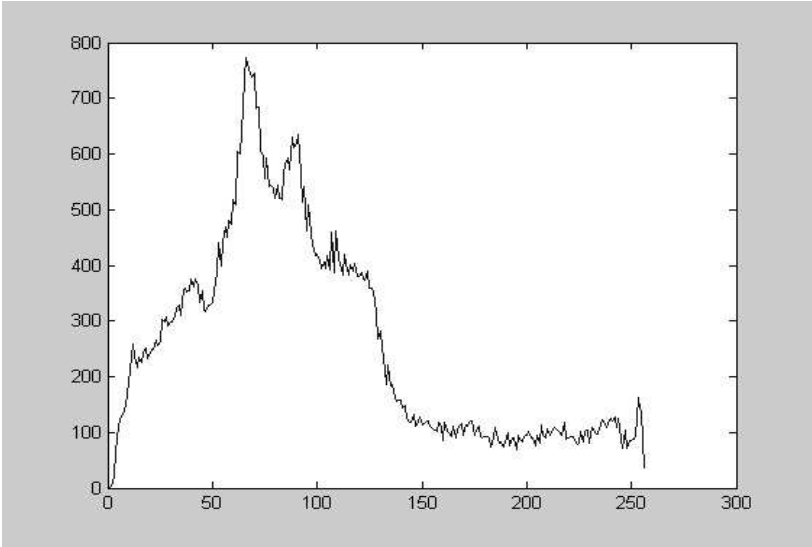
Realistic Histograms

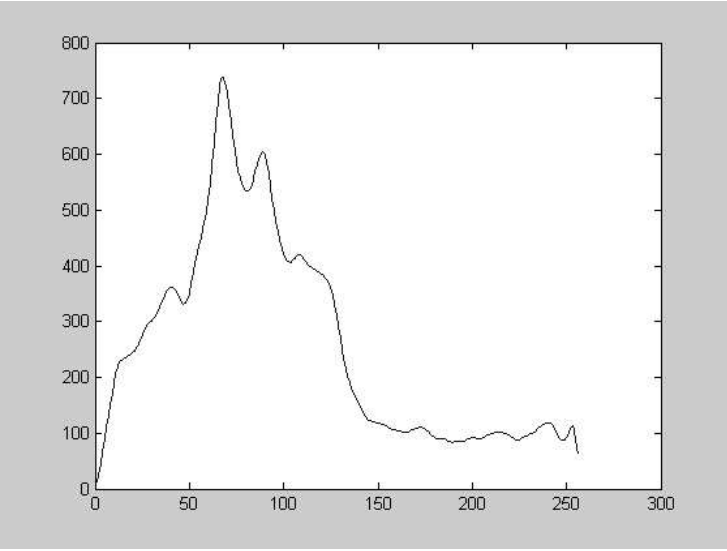




How to Find K

Smooth Histogram. (Convolve by averaging or Gaussian Filter)

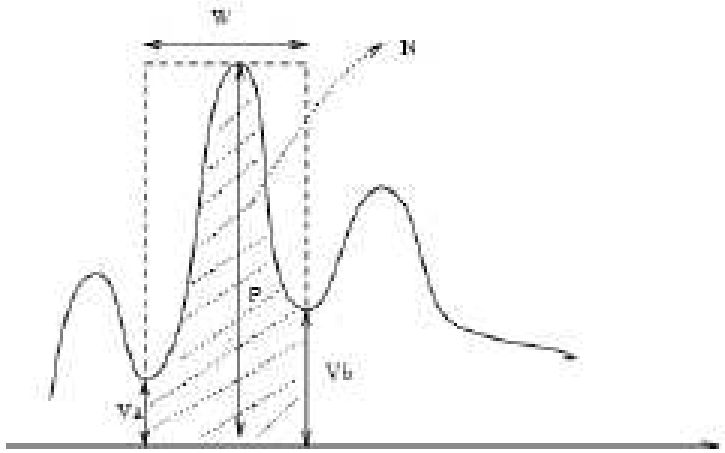




How to Find K

Find Peaks and Valleys and perform peakiness test.

Peakiness Test



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

Agglomerative clustering

- Assume that each cluster is single pixel (i.e. every pixel is a cluster itself).
- Merge Clusters i.e. attach closest to cluster it is closest to (if possible)
- Repeat step 2 until no more clusters can be merged.

Divisive clustering

- Assume that whole image is a single cluster.
- Split Clusters along best boundary (if exists)
- Repeat step 2 until no more clusters can be split.

Inter-Cluster distance

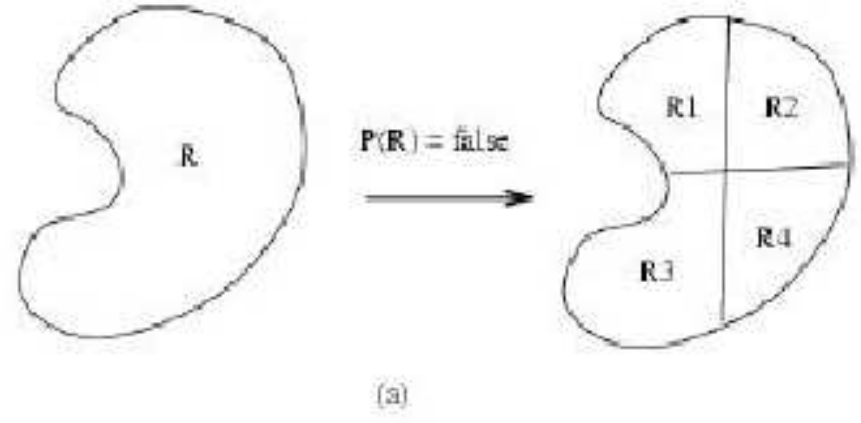
- single-link clustering: Minimum distance between an element of the first cluster and one of the second..
- complete-link clustering: Maximum distance between an element of the first cluster and one of the second.
- group-average clustering: Average of distances between elements in the clusters.

Segmentation by Split and Merge

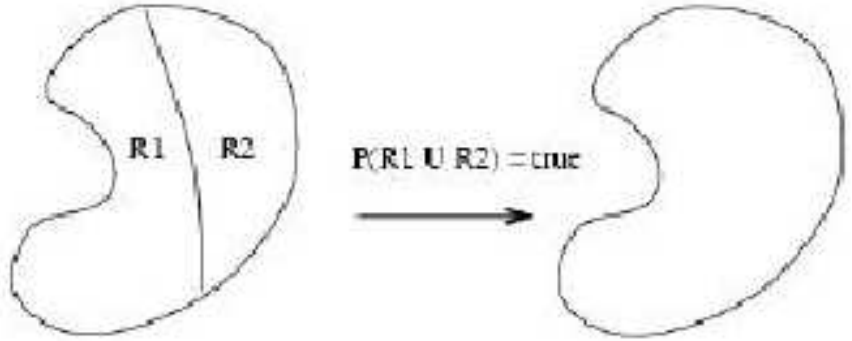
- Start with an initial segmentation (for example by K-Means).
- Define a criteria P for goodness of region such that
  - P(*R*)=True, if *R* satisfies the criteria
  - P(*R*)=False, otherwise

- For each region *R*, split *R* in four regions (quadrants), if P(*R*) = False
- Merge any two adjacent regions *R* and *Q* if $\mathrm{P}(Q \cup R) = \mathrm{True}$
- Repeat until no more clusters can be split or merged.

Segmentation by Split and Merge



(a)



Suggested Reading

- Chapter 14, David A. Forsyth and Jean Ponce, “Computer Vision: A Modern Approach”.
- Chapter 3, Mubarak Shah, “Fundamentals of Computer Vision”