Object Detection and Image Segmentation

Image Segmentation

- Group similar components (such as, pixels in an image, image frames in a video) to obtain a compact representation.
- Applications: Finding tumors, veins, etc. in medical images, finding targets in satellite/aerial images, finding people in surveillance images, summarizing video, etc.
- Methods: Thresholding, K-means clustering, etc.

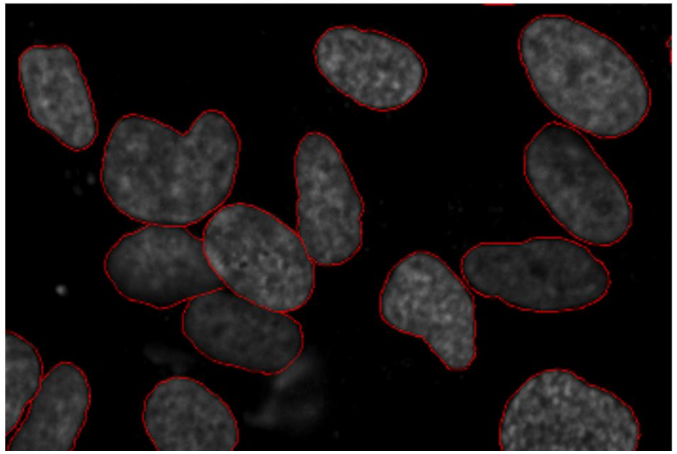


Image Segmentation

Segmentation algorithms for monochrome images generally are based on one of two basic properties of gray-scale values:

- **Discontinuity**
  - The approach is to partition an image based on abrupt changes in gray-scale levels.
  - The principal areas of interest within this category are detection of isolated points, lines, and edges in an image.
- **Similarity**
  - The principal approaches in this category are based on thresholding, region growing, and region splitting/merging.

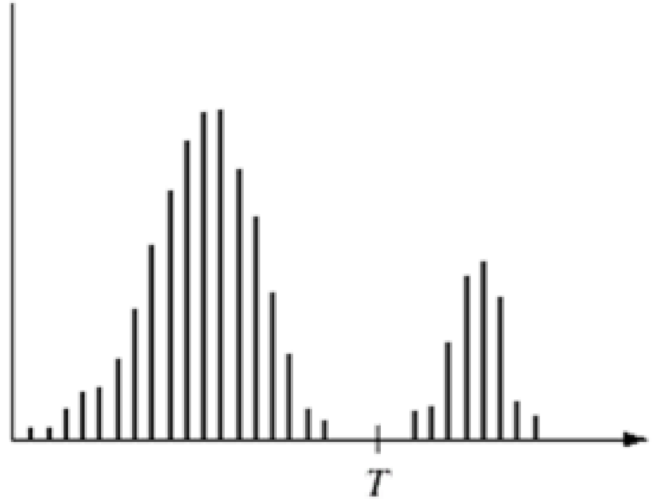
Image Segmentation

Segmentation algorithms for monochrome images generally are based on one of two basic properties of gray-scale values:

- **Discontinuity**
  - The approach is to partition an image based on abrupt changes in gray-scale levels.
  - The principal areas of interest within this category are detection of isolated points, lines, and edges in an image.
- **Similarity**
  - The principal approaches in this category are based on thresholding, region growing, and region splitting/merging.

Image Segmentation - Thresholding

Suppose that an image, $f(x, y)$, is composed of light objects on a dark background, and the following figure is the histogram of the image.



- A histogram of a gray-tone image is an array H[*] of bins, one for each gray tone.
- H[i] gives the count of how many pixels of an image have gray tone i.
- P[i] (the normalized histogram) gives the percentage of pixels that have gray tone i.

Then, the objects can be extracted by comparing pixel values with a threshold T.

⇒ **Global Thresholding**

$$g(x,y)=\begin{cases}1\,if\;\; f(x,y)>T\\ 0\,if\;\; f(x,y)\le T\end{cases}$$

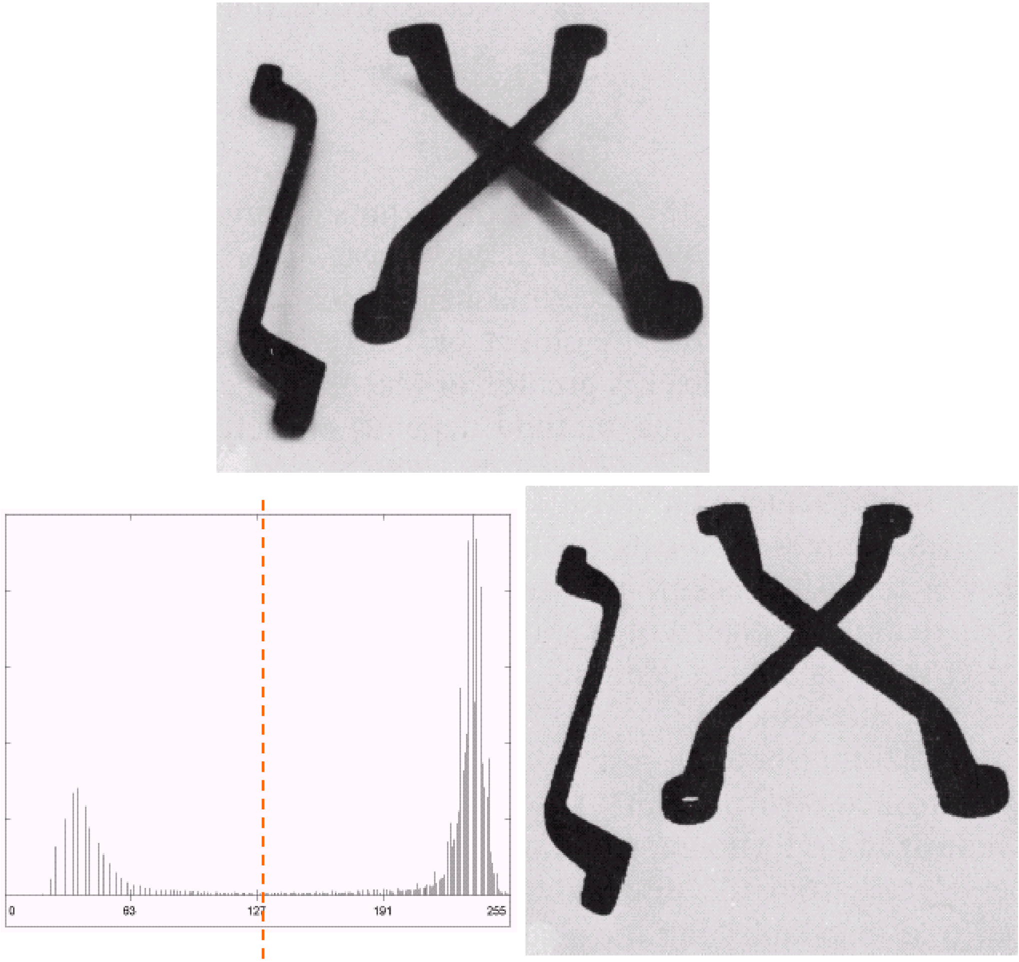
Image Segmentation - Thresholding

How to calculate T?

1. Select an initial estimate for T (typically the average grey level in the image)
2. Segment the image using T to produce two groups of pixels: G_1 consisting of pixels with grey levels $> T$ and G_2 consisting pixels with grey levels $\leq\ T$
3. Compute the average grey levels of pixels in G_1 to give μ_1 and G_2 to give μ_2.
4. Compute a new threshold value:
$$T = \frac{\mu_1 + \mu_2}{2}$$
5. Repeat steps 2 – 4 until the difference in T in successive iterations is less than a predefined limit T_∞

Works very well for finding thresholds when the histogram is suitable.

Image Segmentation - Thresholding



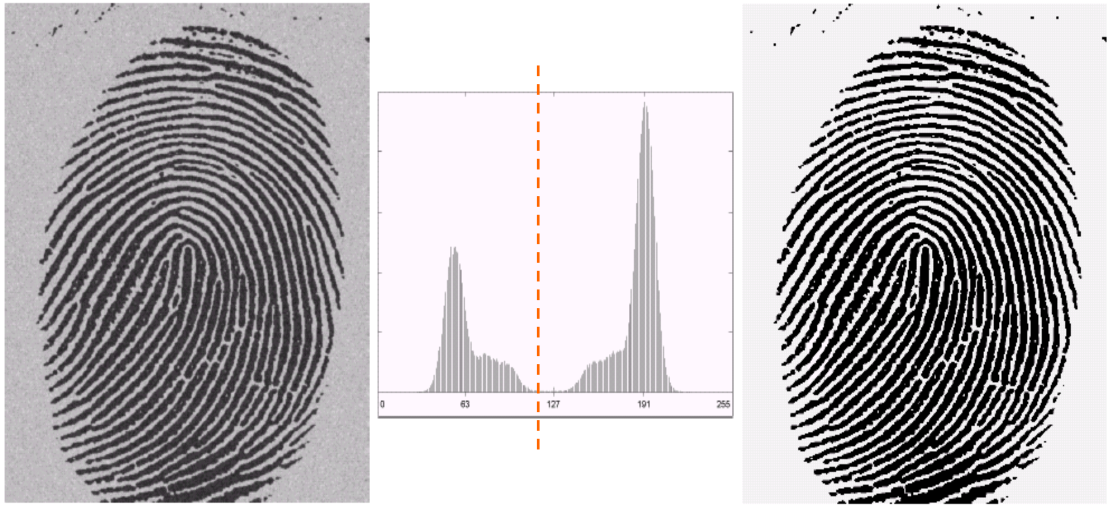
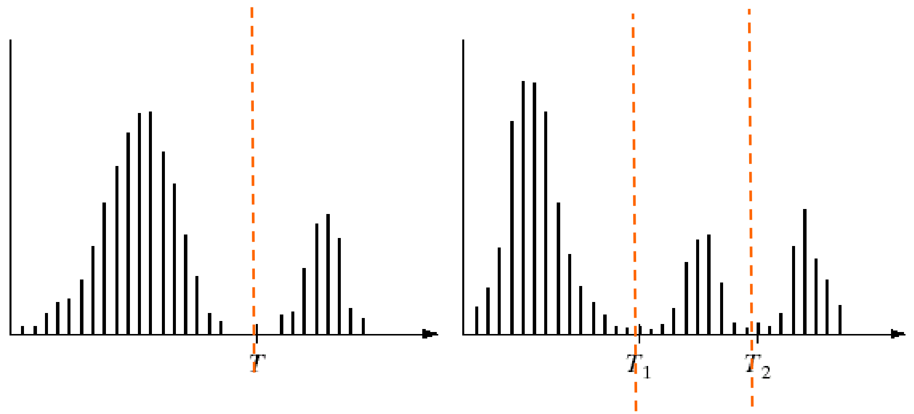


Image Segmentation - Thresholding

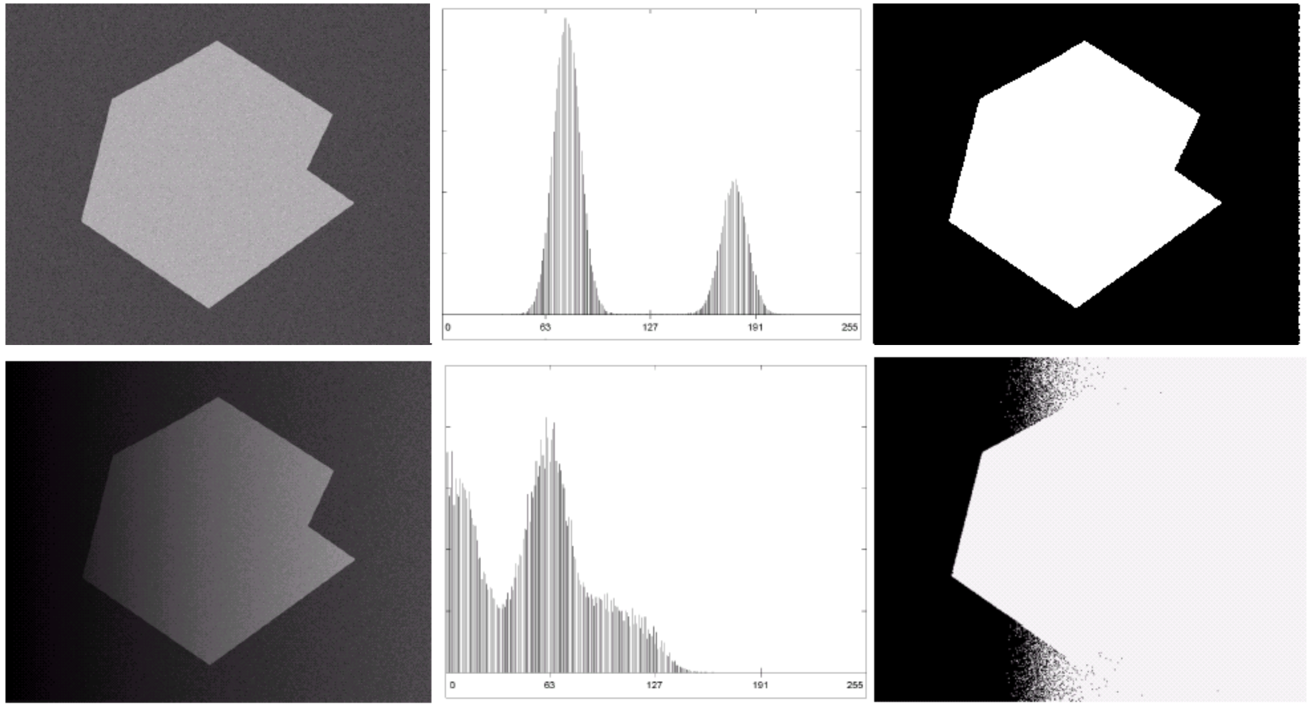
It is also possible to extract objects that have a specific intensity range using multiple thresholds.

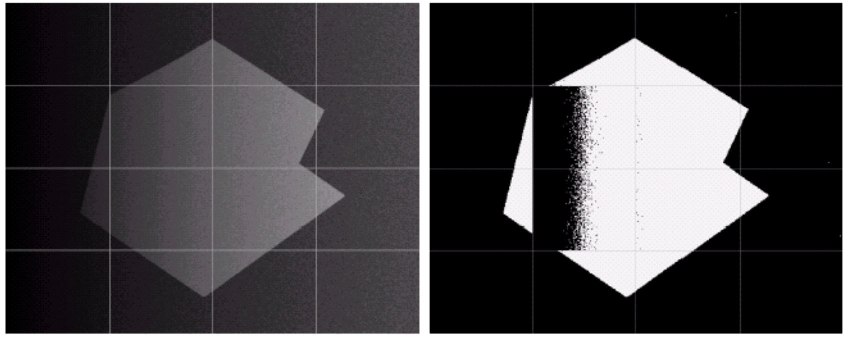


Extension to color images is straightforward: There are three color channels, in each one specify the intensity range of the object... Even if objects are not separated in a single channel, they might be with all the channels... Application example: Detecting/Tracking faces based on skin color...

Image Segmentation - Thresholding

- Non-uniform illumination may change the histogram in a way that it becomes impossible to segment the image using a single global threshold.
- Choosing local threshold values may help ⇒ **Adaptive threshold**.





Region Oriented Segmentation

Region Growing

- Region growing is a procedure that groups pixels or subregions into larger regions.
- The simplest of these approaches is *pixel aggregation*, which starts with a set of “seed” points and from these grows regions by appending to each seed points those neighboring pixels that have similar properties (such as gray level, texture, color, shape).
- Region growing based techniques are better than the edge-based techniques in noisy images where edges are difficult to detect.

Region Oriented Segmentation

Suppose that we have the image given below.

(a) Use the region growing idea to segment the object. The seed for the object is the center of the image. Region is grown in horizontal and vertical directions, and when the difference between two pixel values is less than or equal to 5.

(b) What will be the segmentation if region is grown in horizontal, vertical, and diagonal directions?

Table 2: Show the result of Part (b) on this figure.

10	10	10	10	10	10	10
10	10	10	69	70	10	10
59	10	60	64	59	56	60
10	59	10	<u>60</u>	70	10	62
10	60	59	65	67	10	65
10	10	10	10	10	10	10
10	10	10	10	10	10	10

Region Oriented Segmentation

Region Splitting

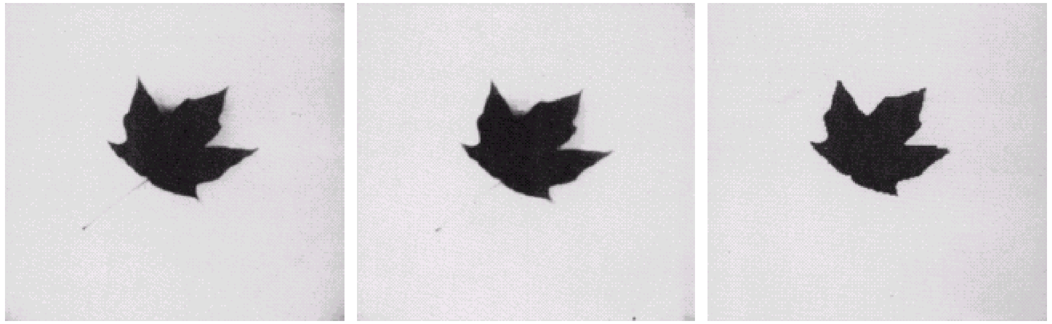
- Region growing starts from a set of seed points.
- An alternative is to start with the whole image as a single region and subdivide the regions that do not satisfy a condition of homogeneity.

Region Merging

- Region merging is the opposite of region splitting.
- Start with small regions (e.g. 2x2 or 4x4 regions) and merge the regions that have similar characteristics (such as gray level, variance).
- Typically, splitting and merging approaches are used iteratively.

a b c

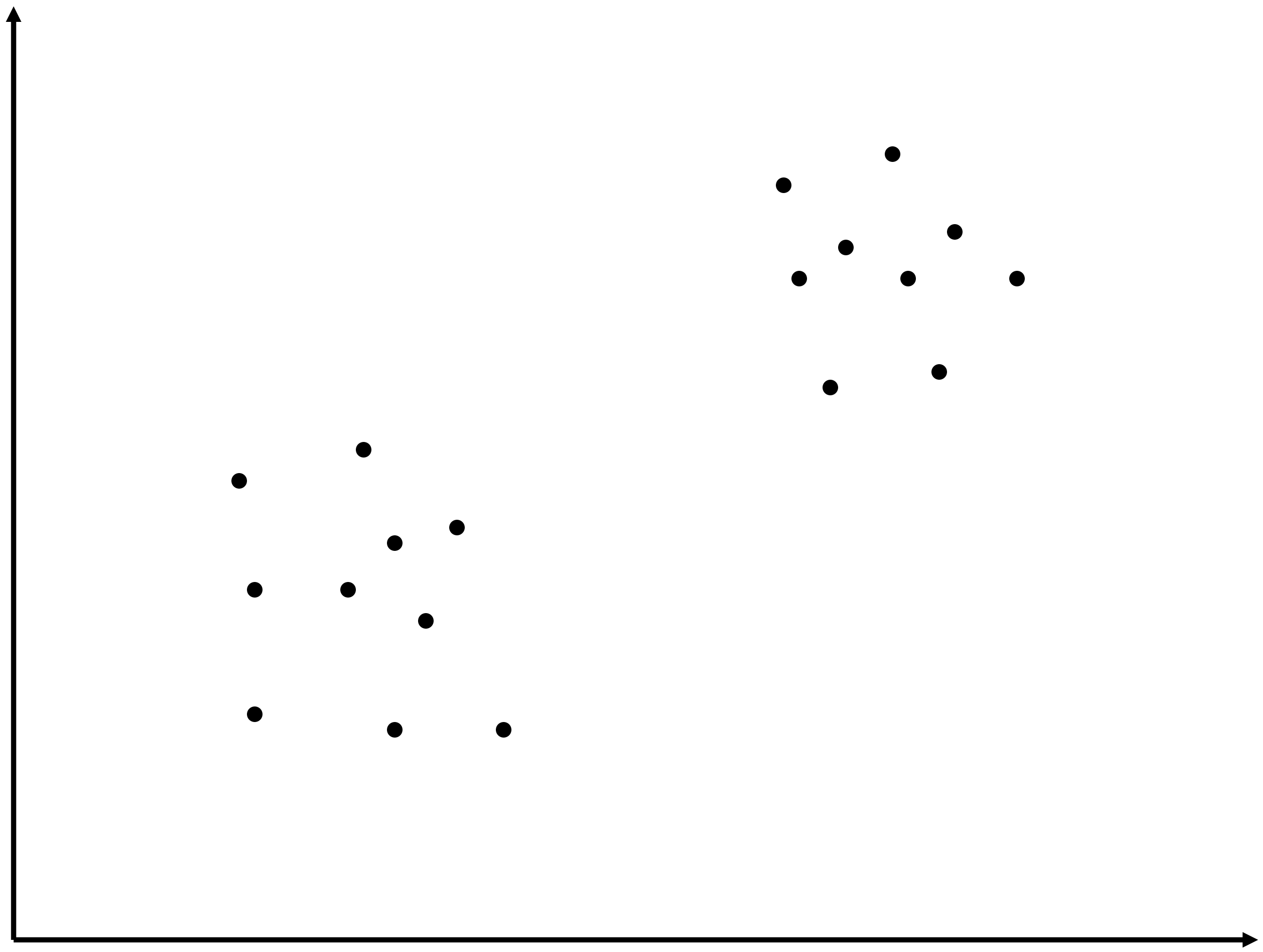
FIGURE 10.43 (a) Original image. (b) Result of split and merge procedure. (c) Result of thresholding (a).



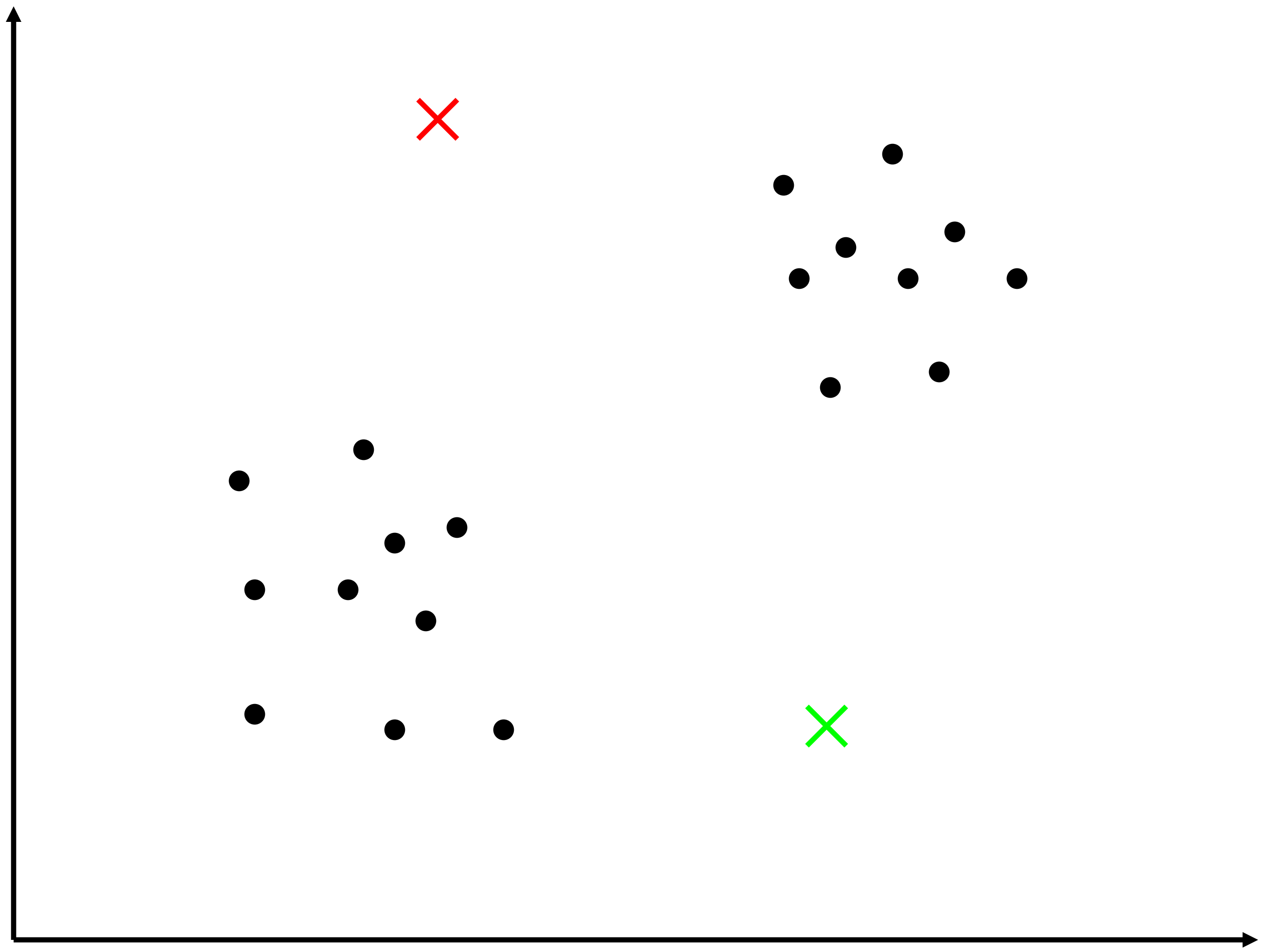
K-Mean Clustering

1. Partition the data points into K clusters randomly. Find the centroids of each cluster.
2. For each data point:
    - Calculate the distance from the data point to each cluster.
    - Assign the data point to the closest cluster.
3. Recompute the centroid of each cluster.
4. Repeat steps 2 and 3 until there is no further change in the assignment of data points (or in the centroids).

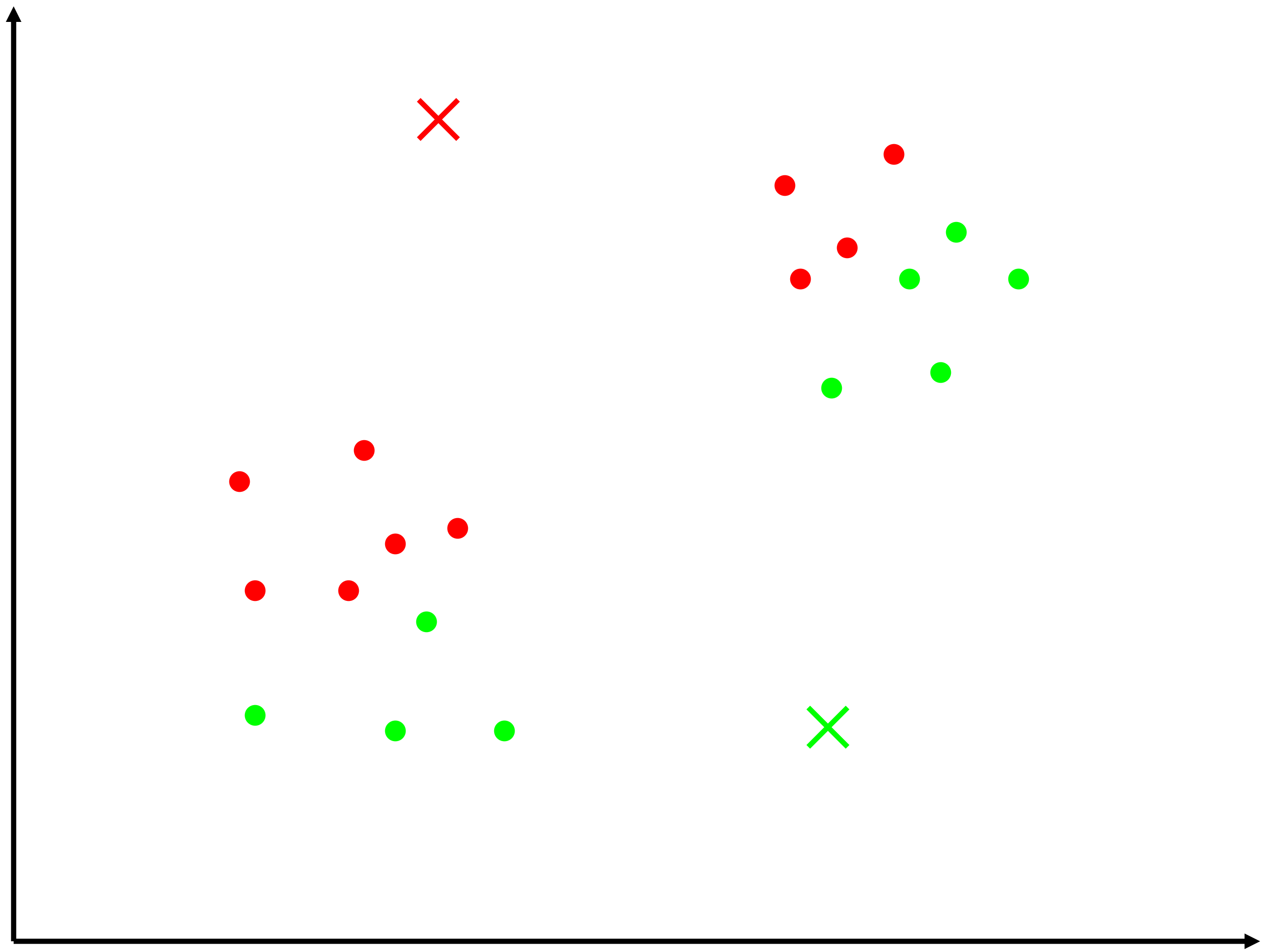
K-Means Clustering



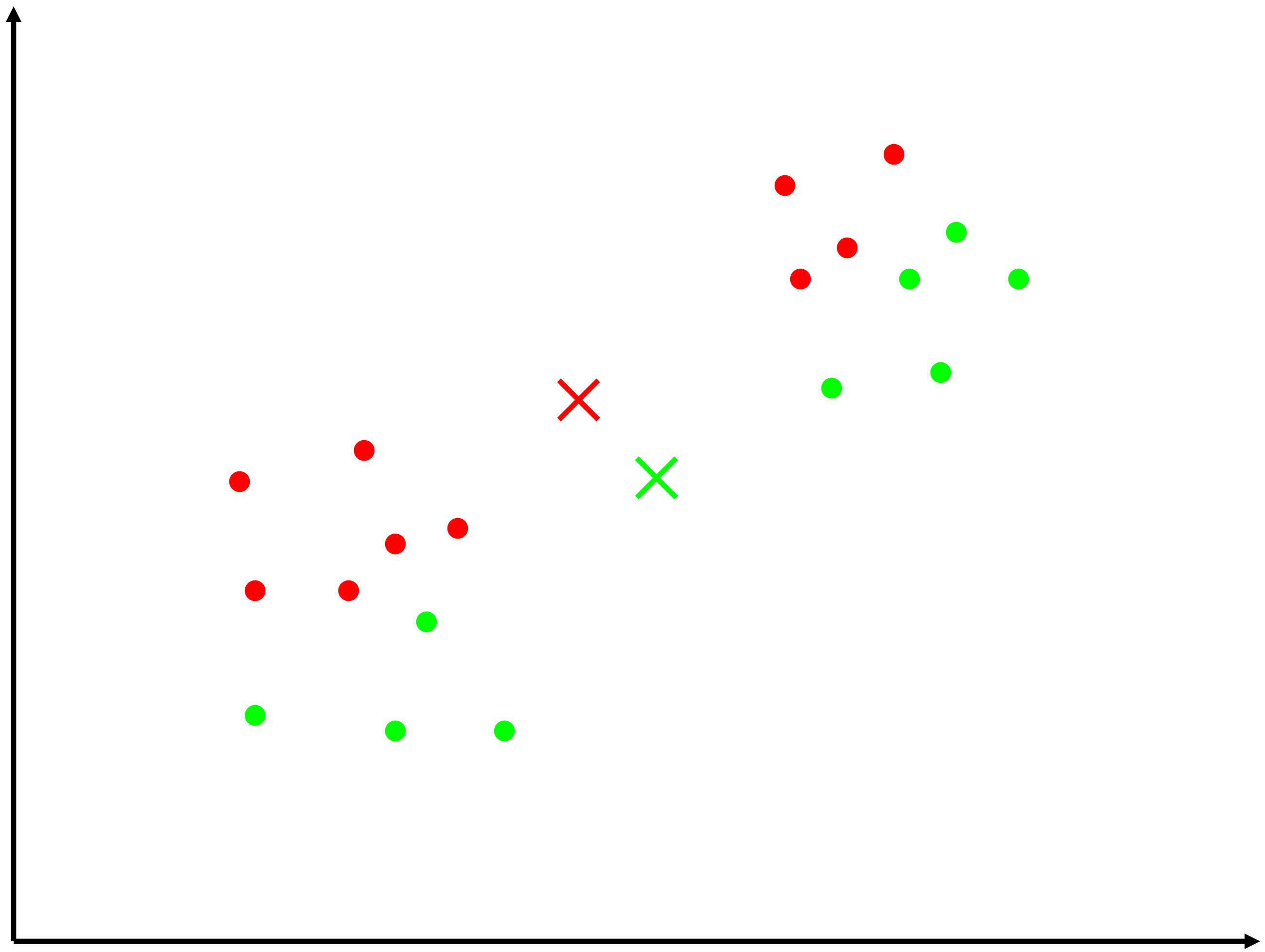
K-Means Clustering



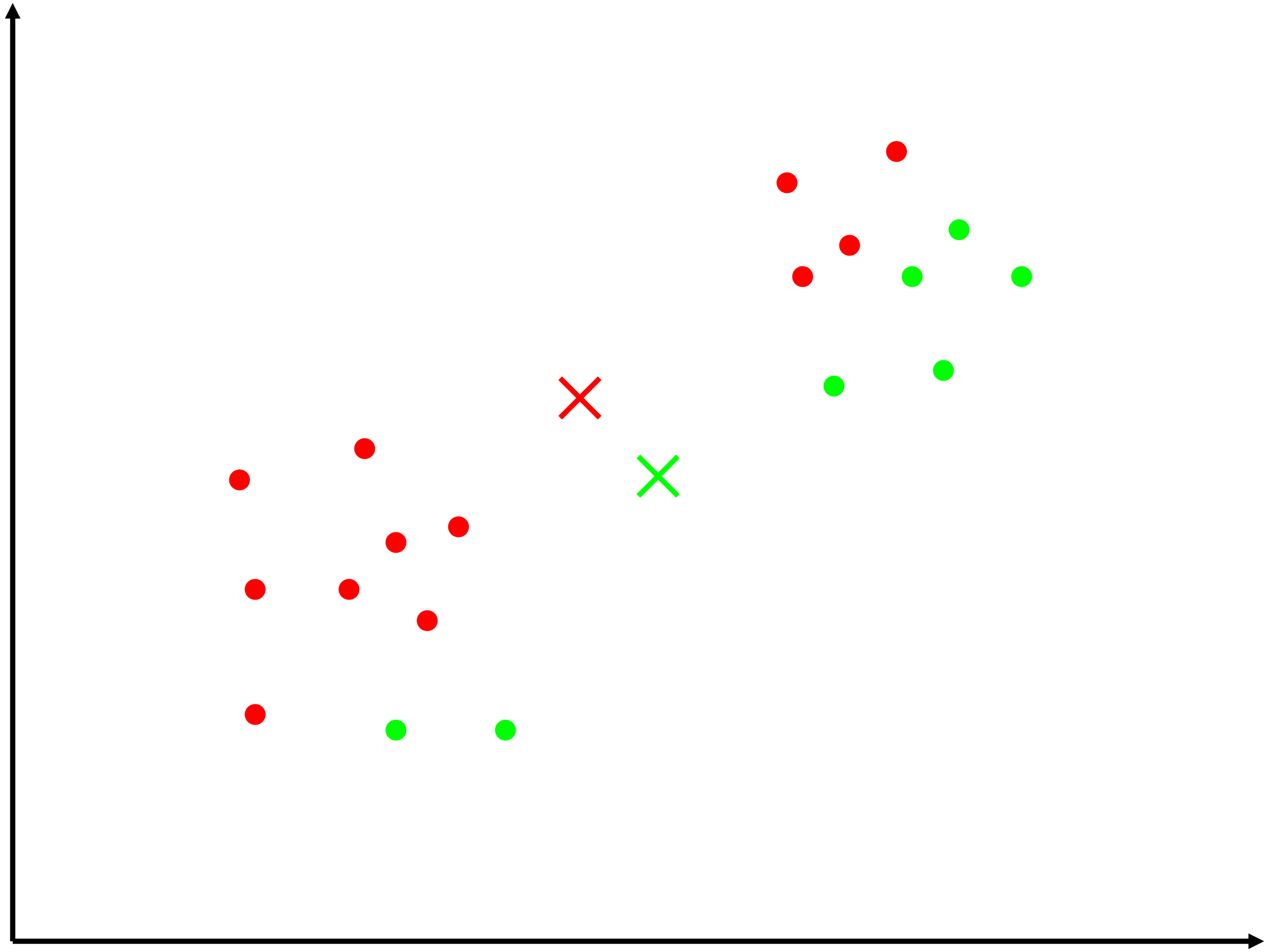
K-Means Clustering



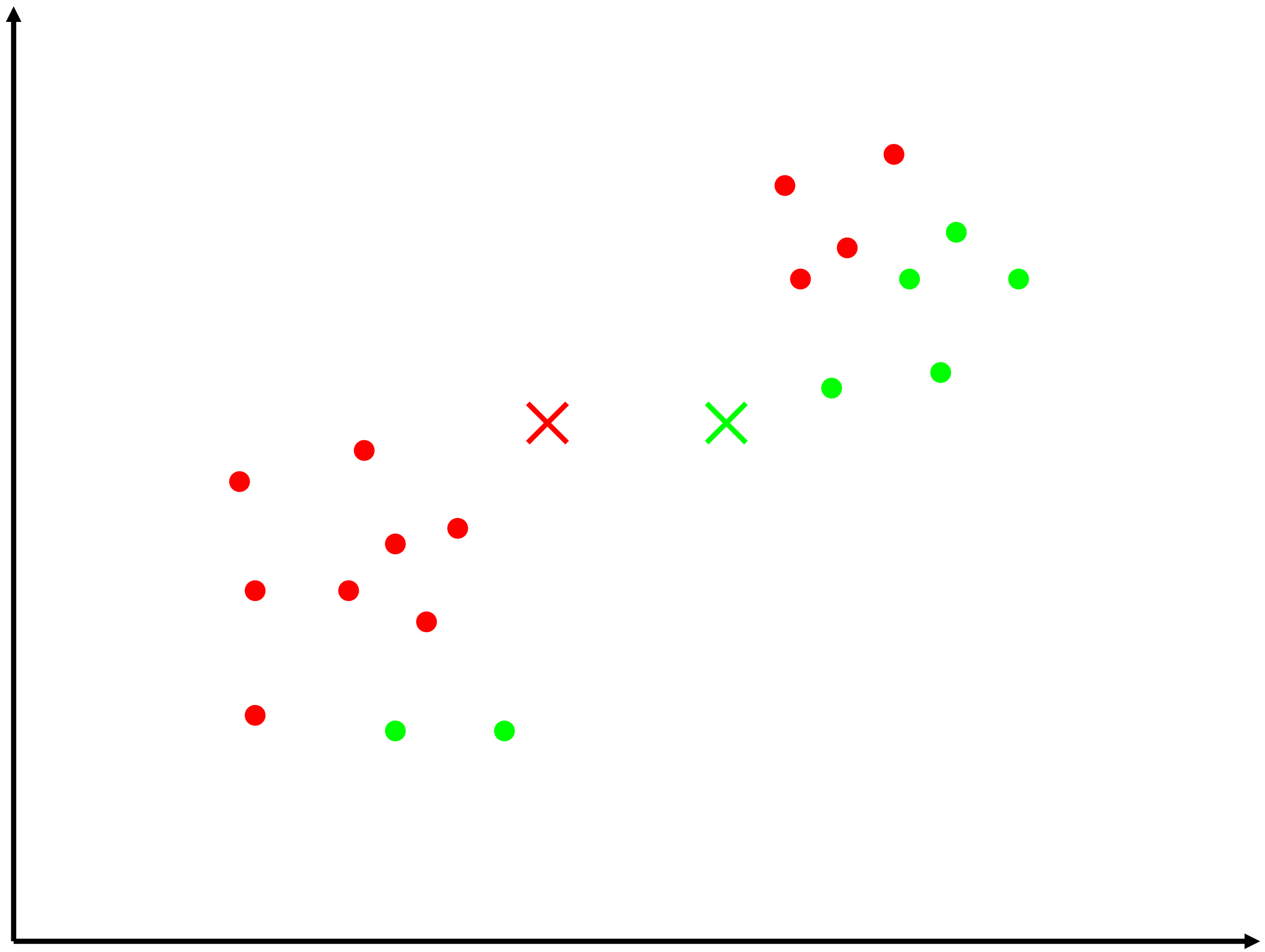
K-Means Clustering



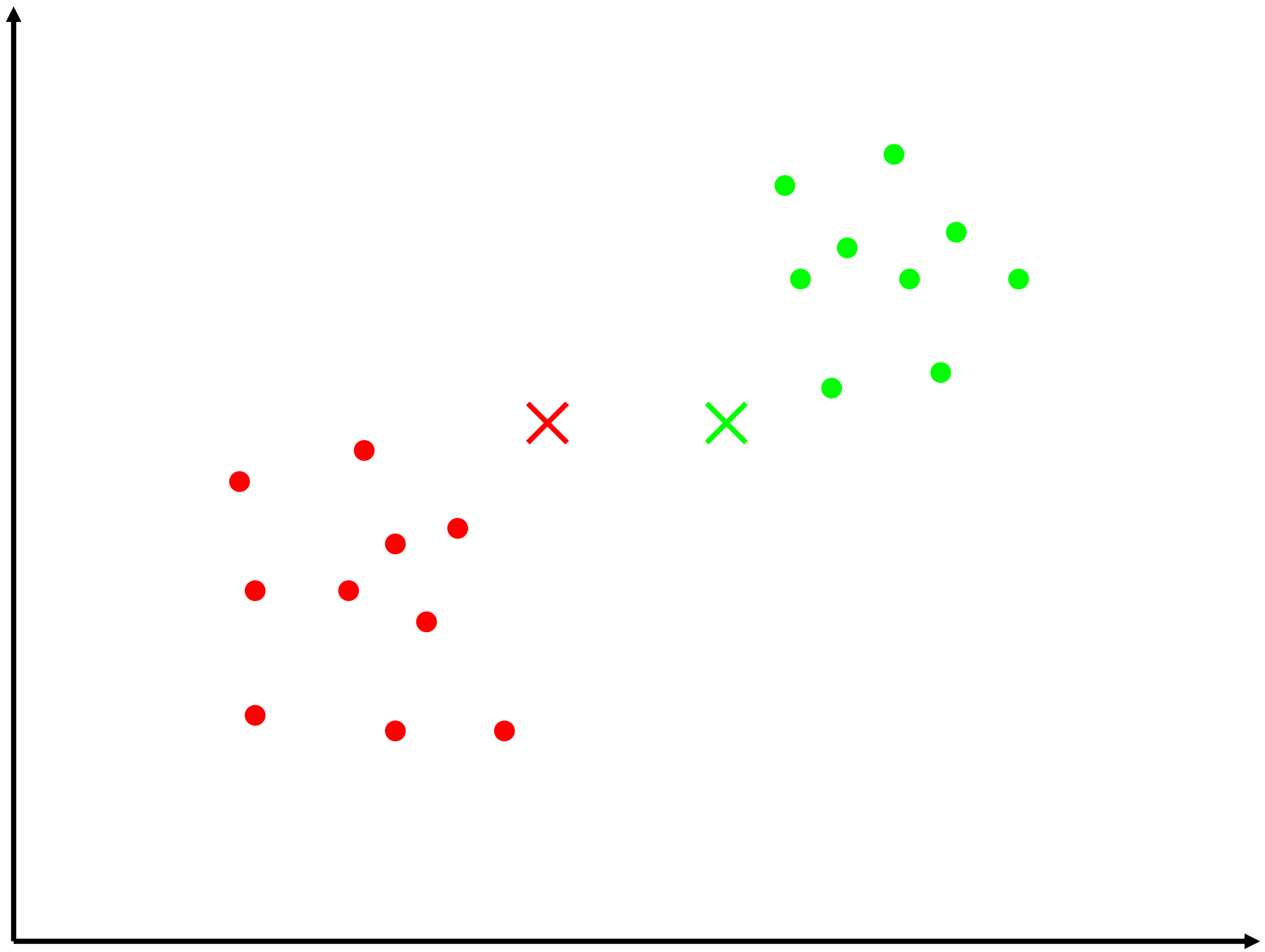
K-Means Clustering



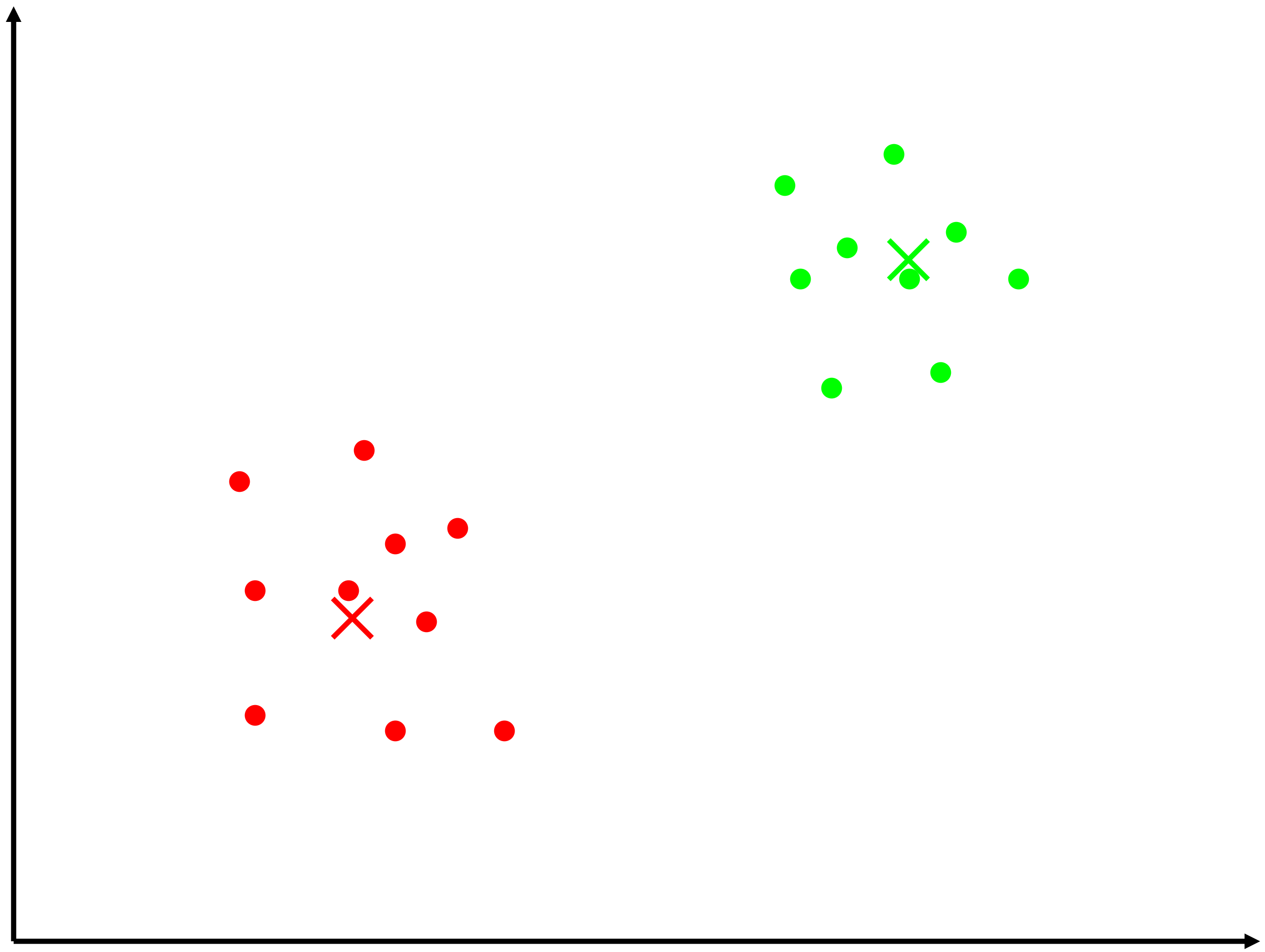
K-Means Clustering



K-Means Clustering

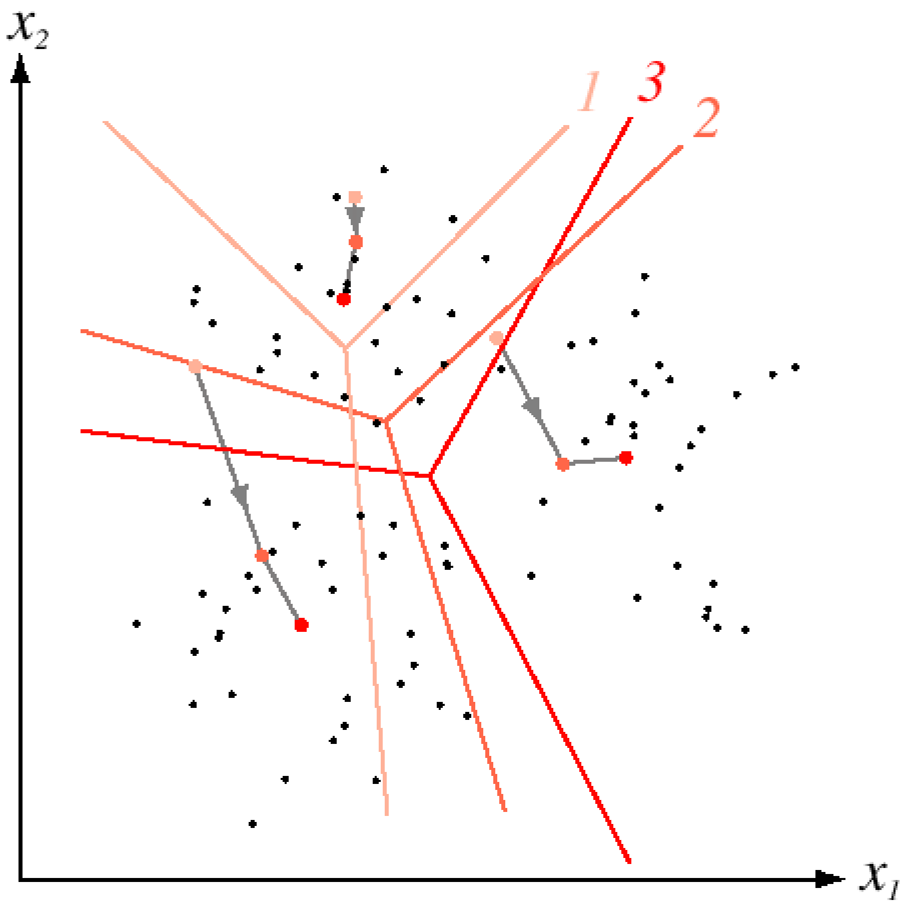


K-Means Clustering



K-Means Clustering

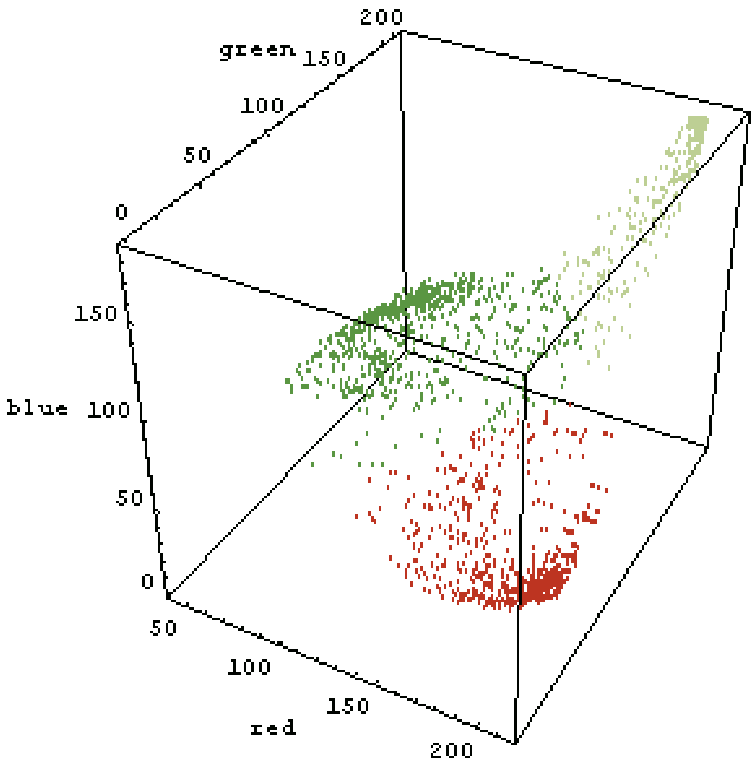
- Example



Duda et al.

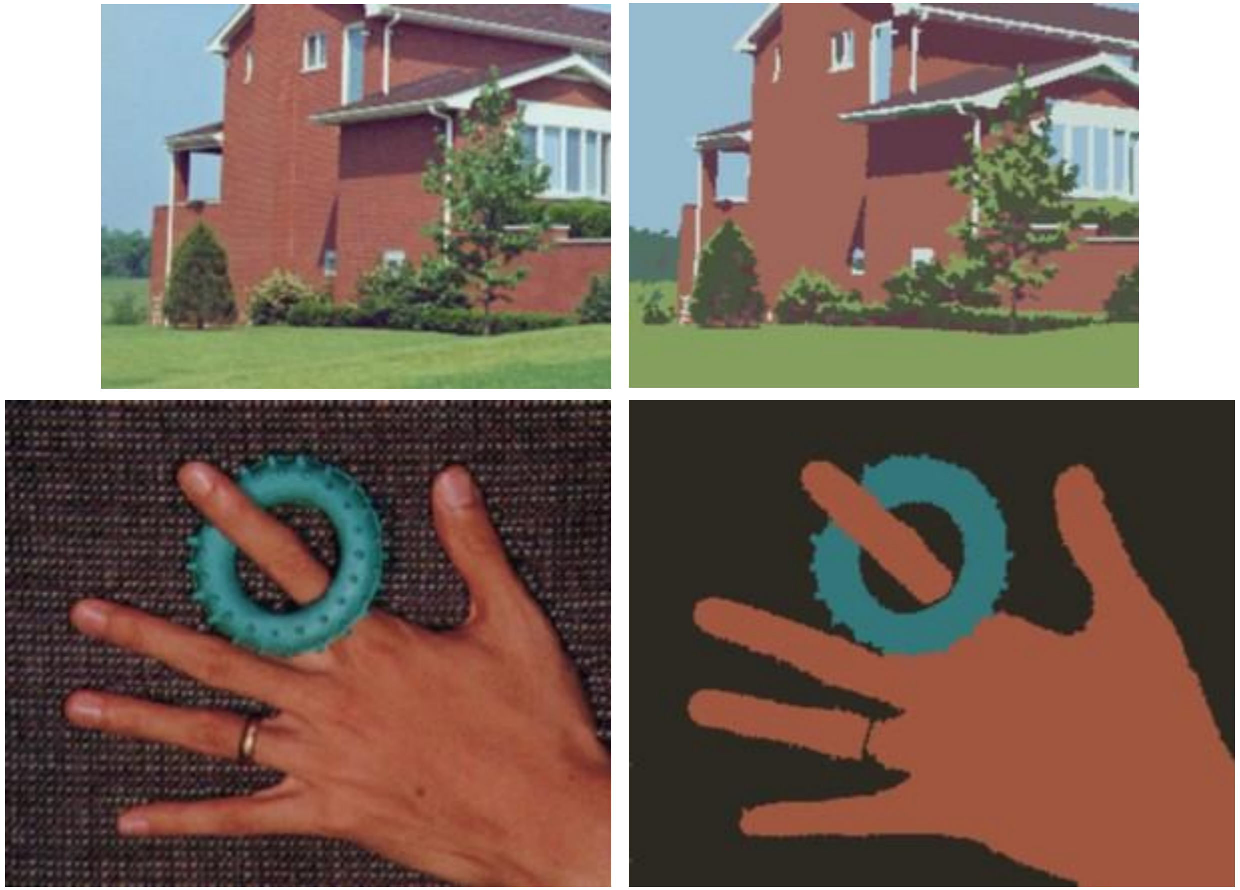
K-Means Clustering

RGB vector



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

Clustering



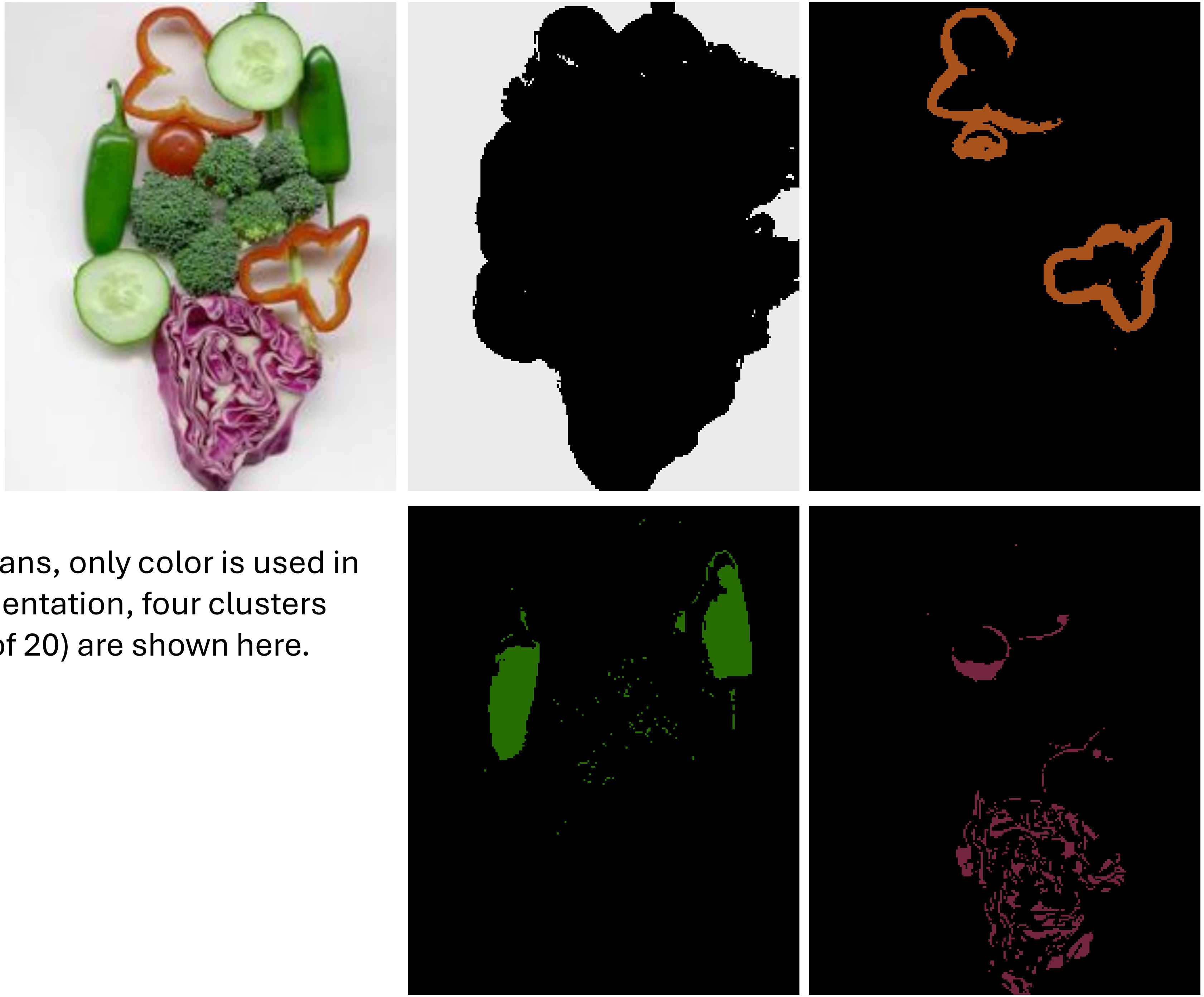
D. Comaniciu and P. Meer, *Robust Analysis of Feature Spaces: Color Image Segmentation*, 1997.

K-Means Clustering

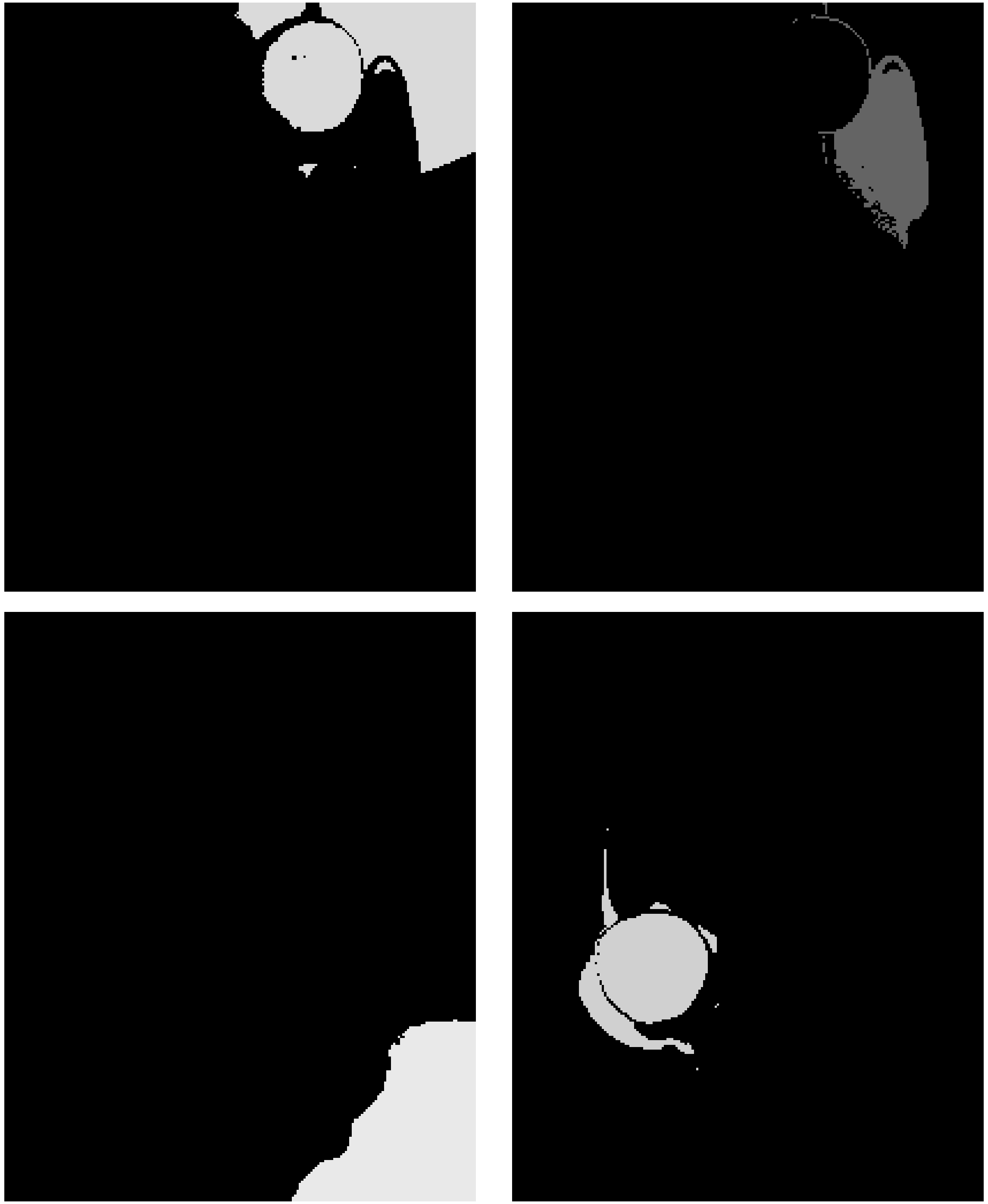
Original

K=5

K=11



K-means, only color is used in segmentation, four clusters (out of 20) are shown here.



K-means, color and position is used in segmentation, four clusters (out of 20) are shown here.

Each vector is (R,G,B,x,y).

K-Means Clustering: Axis Scaling

Features of different types may have different scales.

For example, pixel coordinates on a 100x100 image vs. RGB color values in the range [0,1].

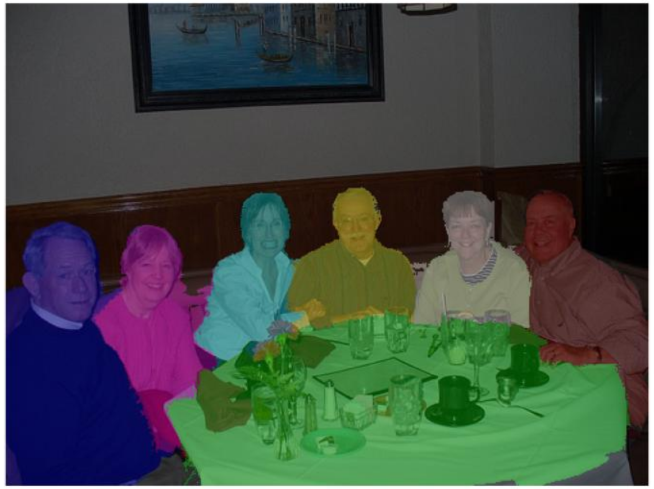
Problem: Features with larger scales dominate clustering.

Solution: Scale the features.

Types of Image Segmentation

- **Semantic segmentation**
  - Label pixels into object classes
  - Traditional methods: conditional random fields
  - Deep learning methods: deconvolution, atrous convolution
- **Instance segmentation**
  - Separate object instances in the same class
  - Detection + segmentation inside each box
- **Panoptic segmentation**: combine both semantic and instance segmentation.

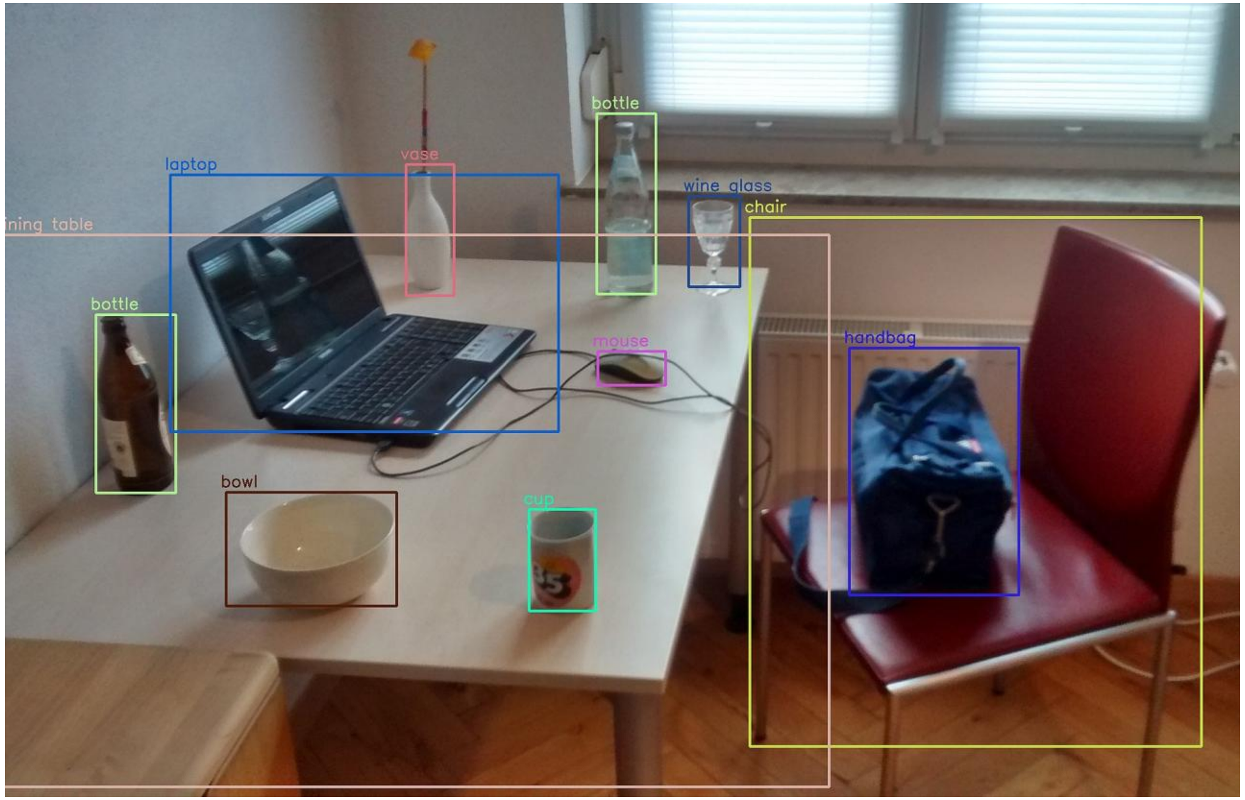
Semantic Segmentation



Instance Segmentation

Object Detection

- Localize objects in images and classify them

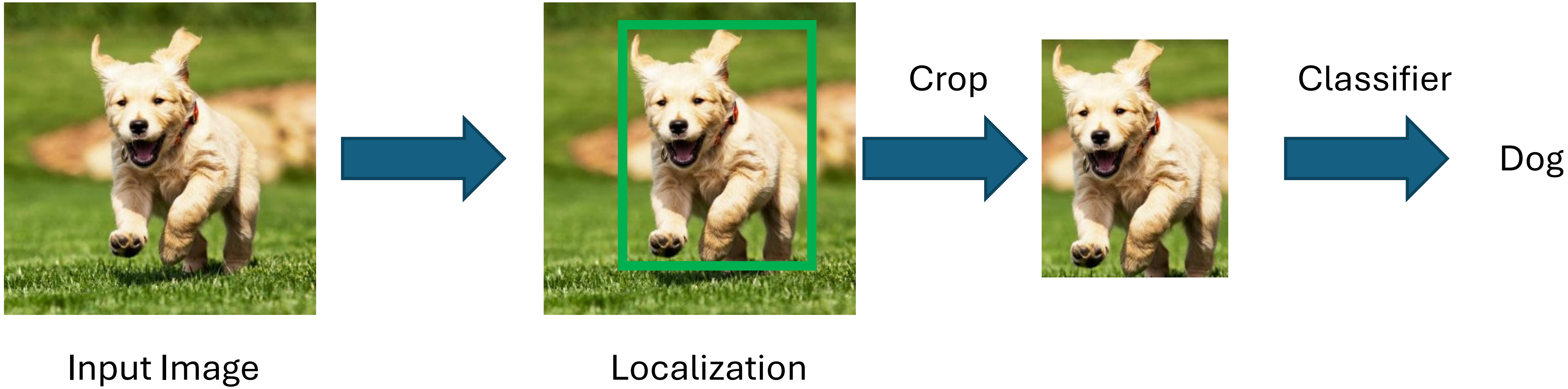


Why using bounding boxes?

- Easy to store
  - (x, y, w, h): box center with width, height
  - (x1, y1, x2, y2): top left corner and bottom right corner
- Easy for image processing
  - Crop a region

Object Detection

- Localization + Detection

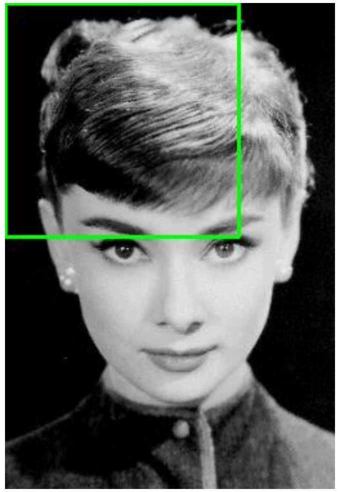


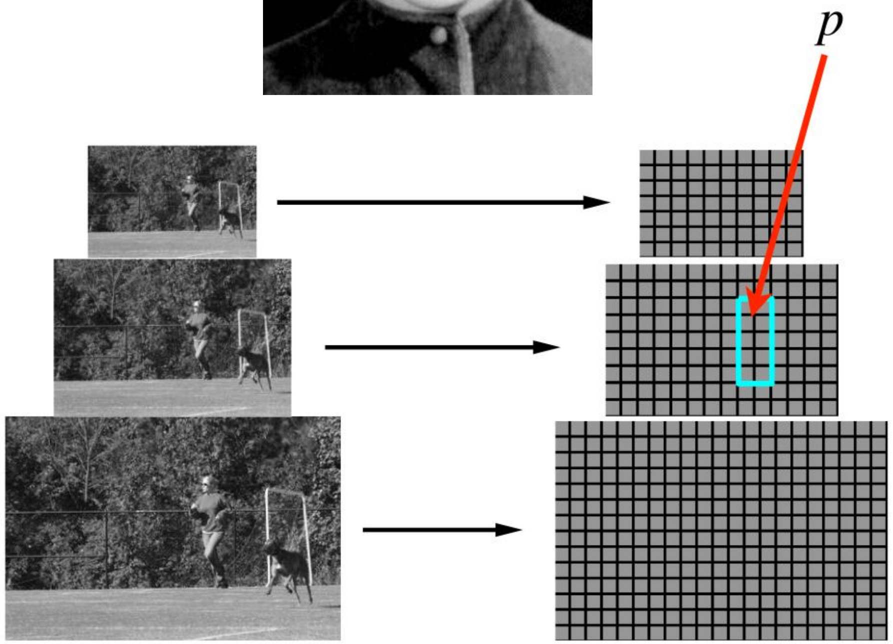
Localization – Sliding window

- Select a window with a fixed size

- Scan the input image with the window (bounding box)

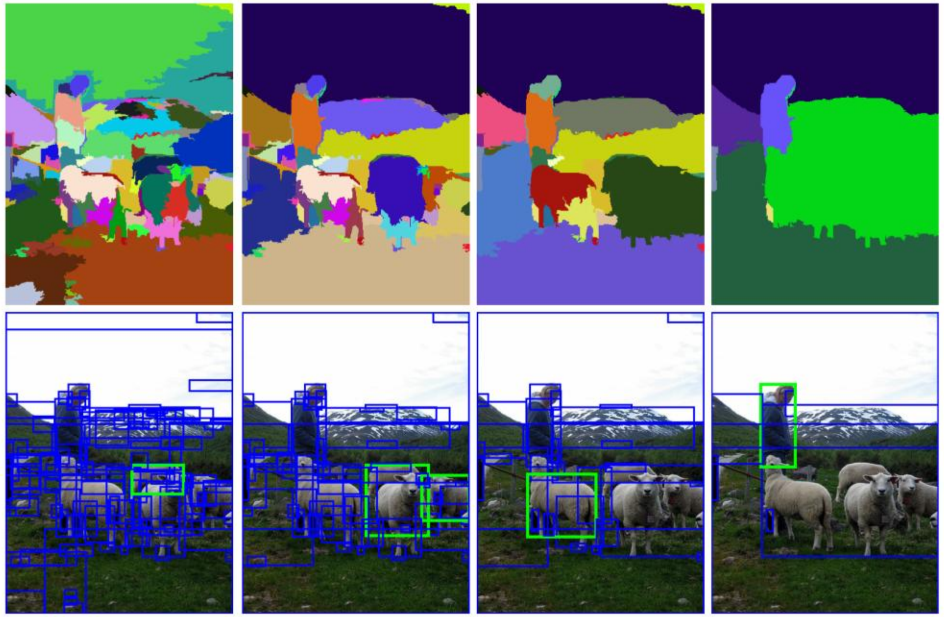
- How to deal with different object scales and aspect ratios?
    - Use boxes with different aspect ratios
    - Image pyramid



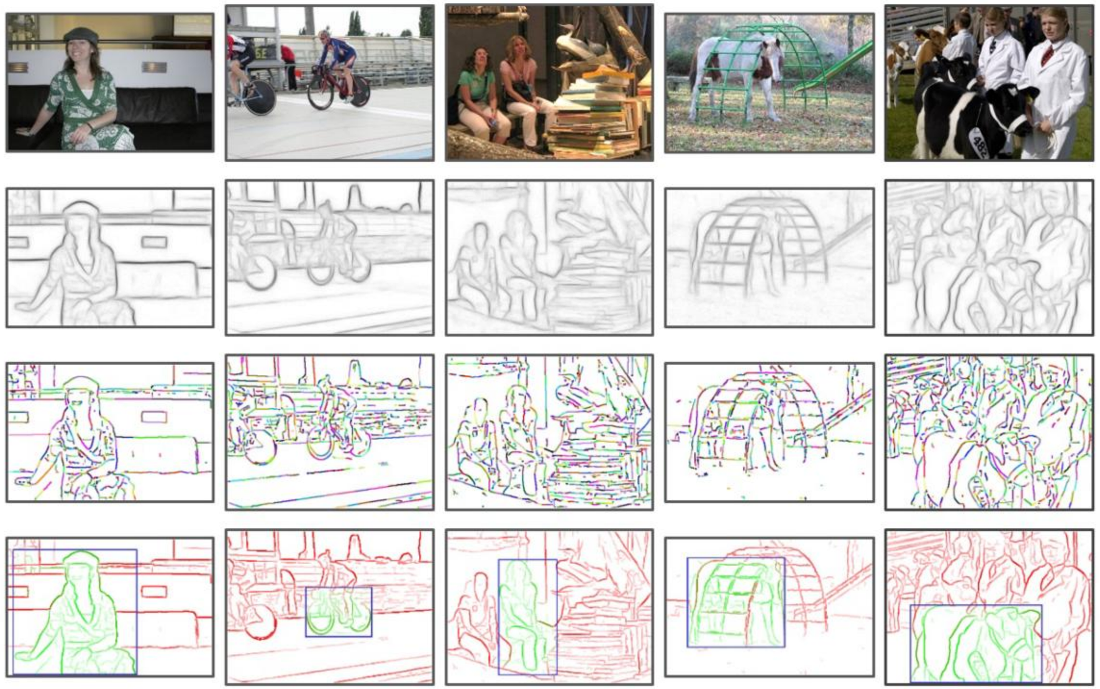


Localization – Region Proposal

- Leverage methods that can generate regions with high likelihood of containing objects
  - E.g., bottom-up segmentation methods, using edges



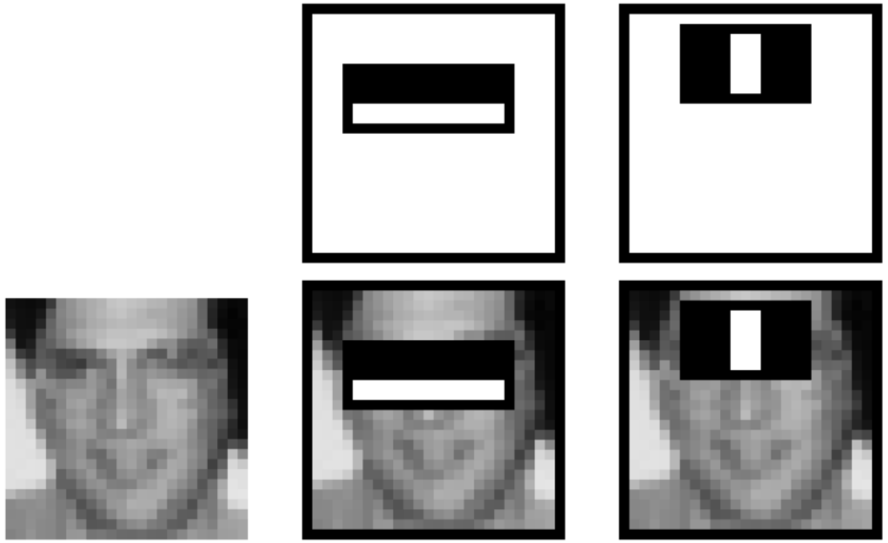
Selective Search, Sande et al., ICCV’11



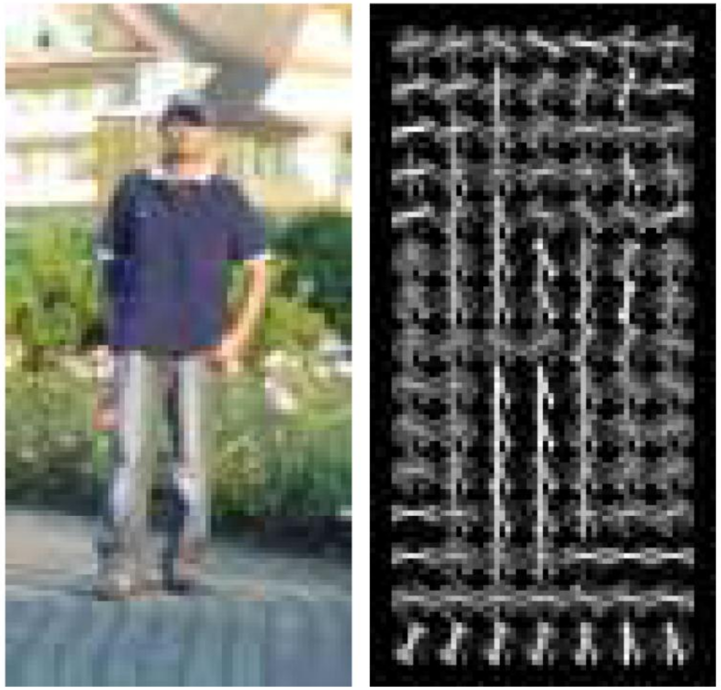
Edge Boxes. Zitnick & Dollar, ECCV’14

Classification – Features

- Traditional methods: Hand-crafted features
- Deep learning methods: learned features in the network



Viola and Jones: rectangle features

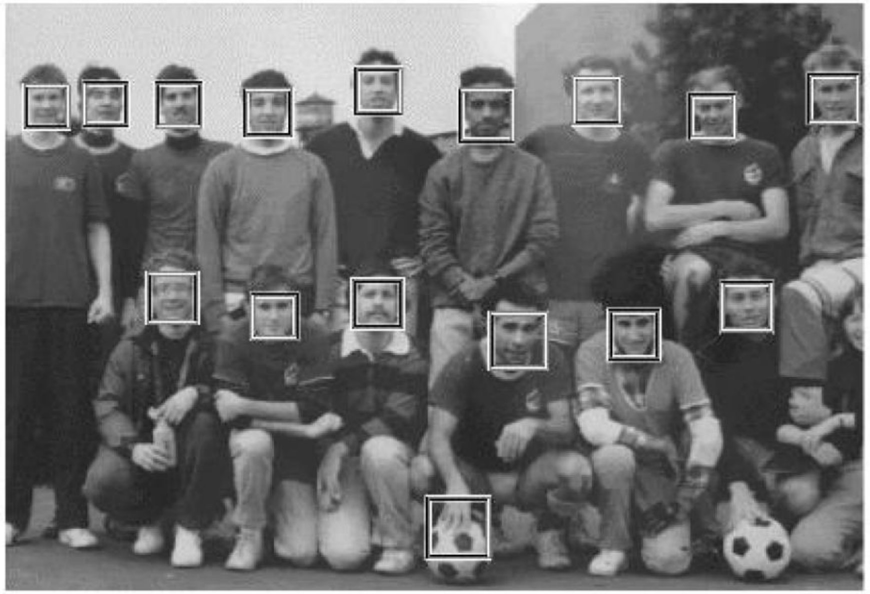


Dadal & Triggs: Histograms of Oriented Gradients

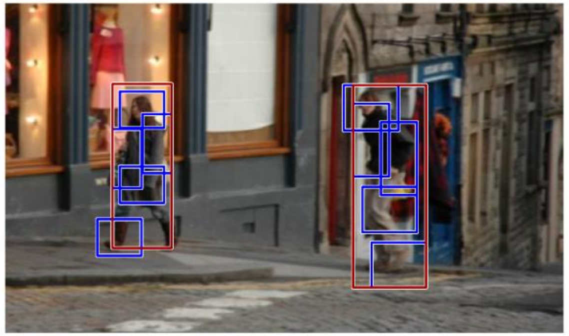
Classification – Classifier

- Traditional methods
  - AdaBoost
  - Support vector machines (SVMs)

- Deep learning methods
  - Neural networks



Viola and Jones: AdaBoost
Robust Real-time Object Detection. IJCV, 2001.



Felzenszwalb et al: SVM
Object detection with discriminatively trained part-based models . TPAMI, 2009.