ITMO

# Categorization

## Computer Vision

# Images Categorization

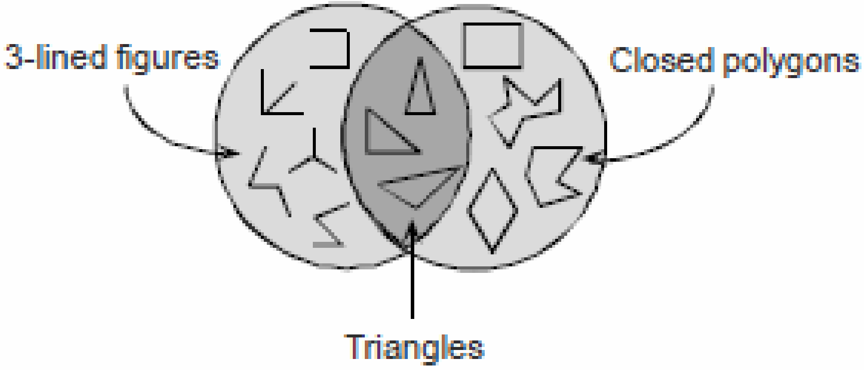
# Images Categorization

- Many objects behave similarly.
- The concept of *categories* contains information about what we can do with an object.
- There are fewer categories (words in a language) than there are objects in the world.
- Object functions depend on the observer.

# Classic Point of View

1. A category is defined by a set of properties common to all elements from the category.
2. Belonging to the category is binary.
3. All items in a category are the same.

- **Example:** Triangles are three-sided closed polylines.



# Natural Categories

- Determined by best examples (prototypes).
- Set the degree of category matching.
- Fuzzy rules.

# Prototypes and Examples

## Model with prototypes

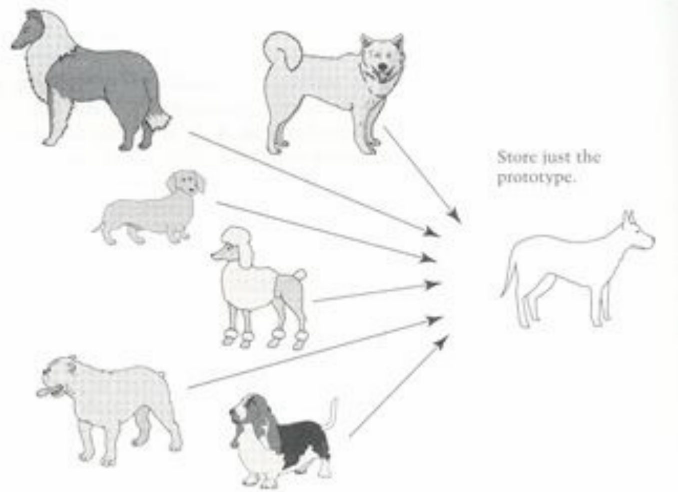



*Figure 7.3.* Schematic of the prototype model. Although many exemplars are seen, only the prototype is stored. The prototype is updated continually to incorporate more experience with new exemplars.

The decision on whether the category matches is made by comparing the object with the prototype

## Model with examples

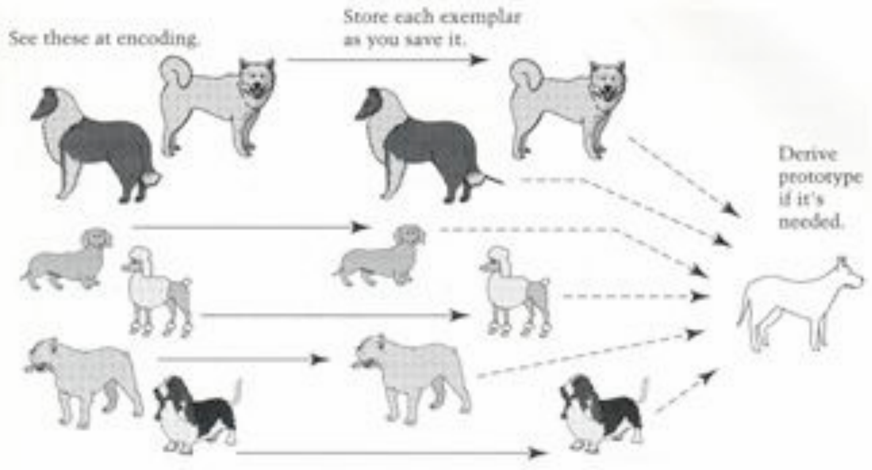


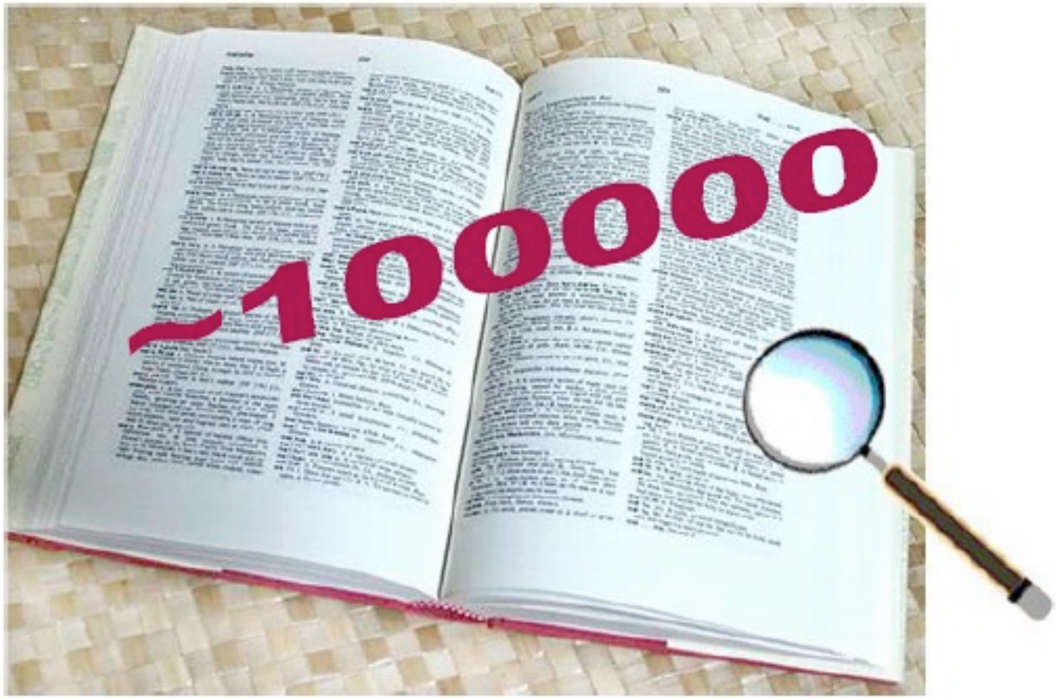

*Figure 7.4.* Schematic of the exemplar model. As each exemplar is seen, it is encoded into memory. A prototype is abstracted only when it is needed, for example, when a new exemplar must be categorized.

The category match is decided by comparing the object with a set of examples from the category

# Canonical Perspective

- The best view of an object by which it is most easily identified.

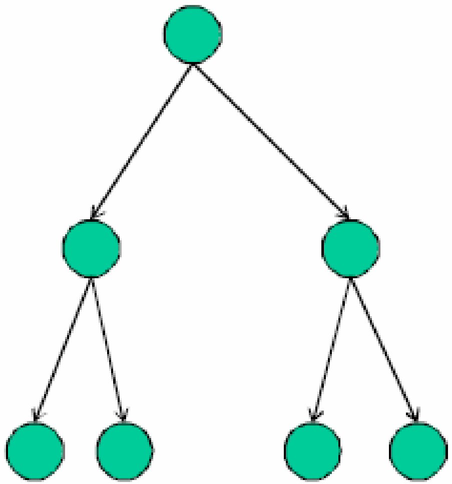
# Number of Categories




171 000+ words total, 47 000 obsolete words, half of them are nouns

# Category Hierarchy

- You can combine categories into groups by building a hierarchy of categories:



# Image Categorization Task

- We should determine if an object (scene) of a given category is present in an image.

Is it a city?

Yes

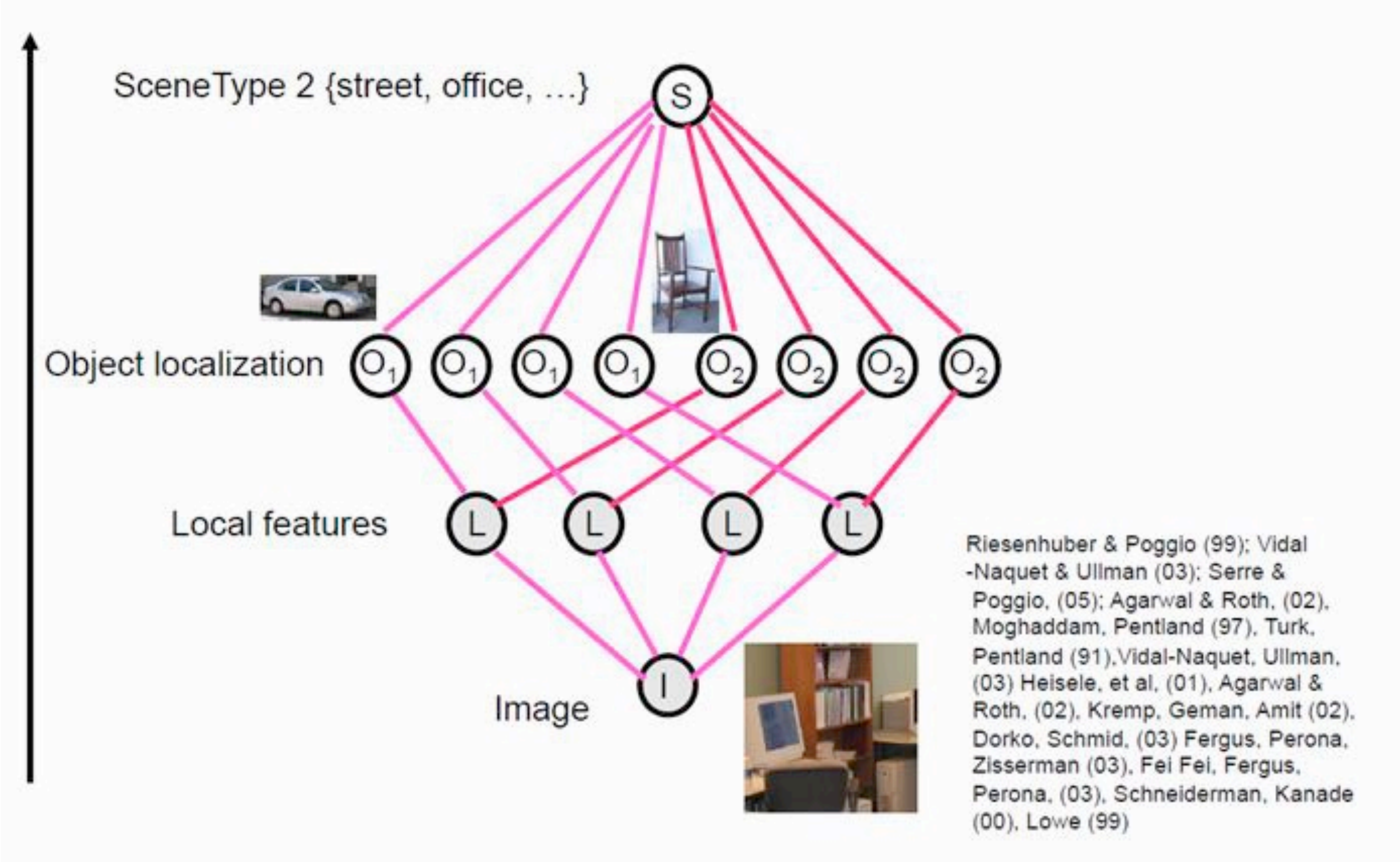
No

Is it a bus?

No

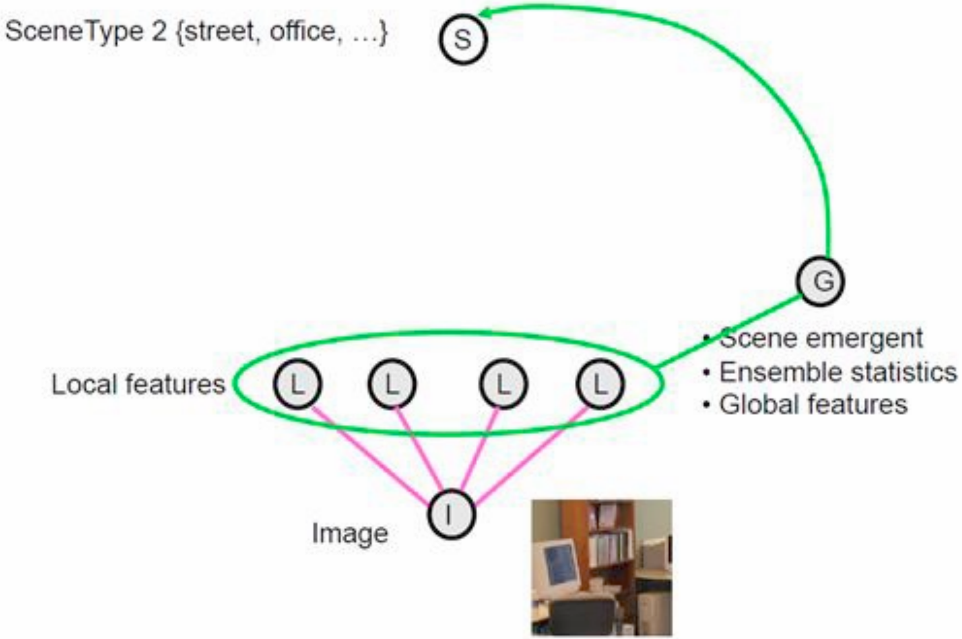
Yes

# Image Model

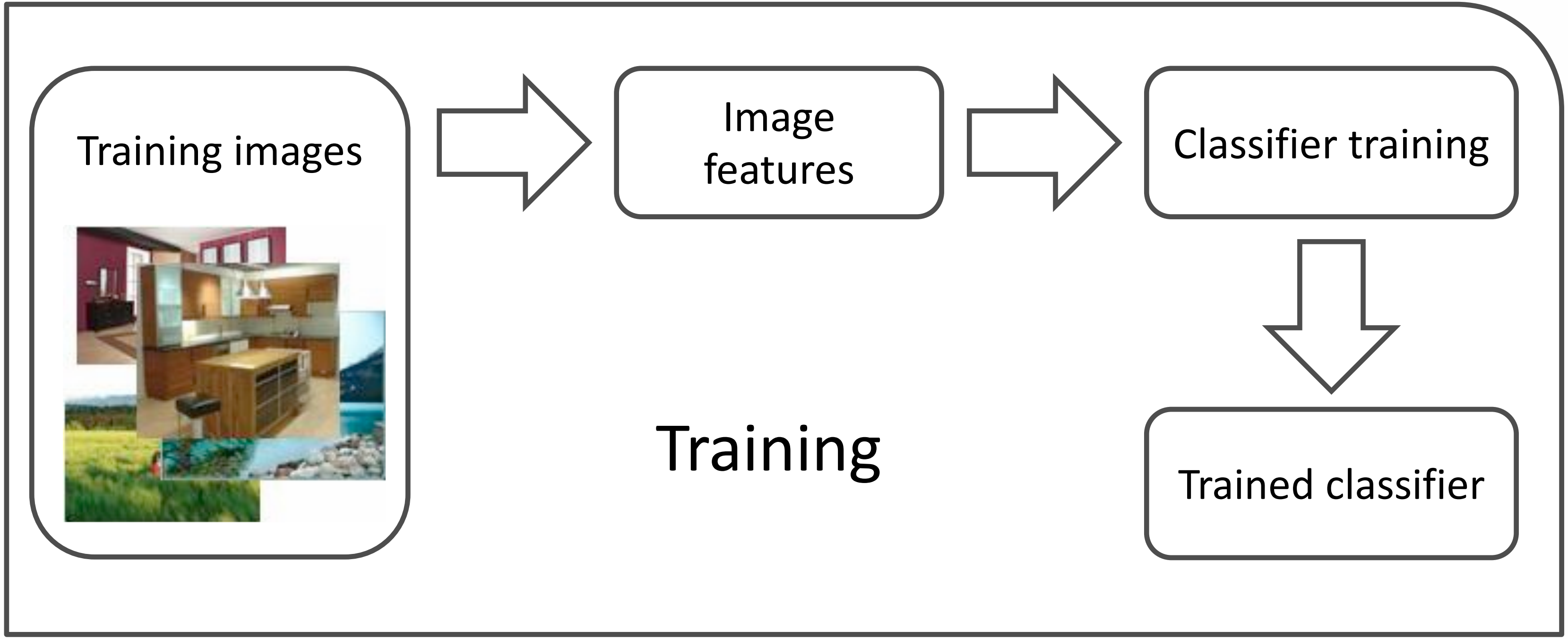

SceneType 2 {street, office, ...}
S
Object localization
O1
O2
Local features
L
Image
I
Riesenhuber & Poggio (99); Vidal -Naquet & Ullman (03); Serre & Poggio, (05); Agarwal & Roth, (02), Moghaddam, Pentland (97), Turk, Pentland (91),Vidal-Naquet, Ullman, (03) Heisele, et al, (01), Agarwal & Roth, (02), Kremp, Geman, Amit (02), Dorko, Schmid, (03) Fergus, Perona, Zisserman (03), Fei Fei, Fergus, Perona, (03), Schneiderman, Kanade (00), Lowe (99)

# General Approach to Image Categorization

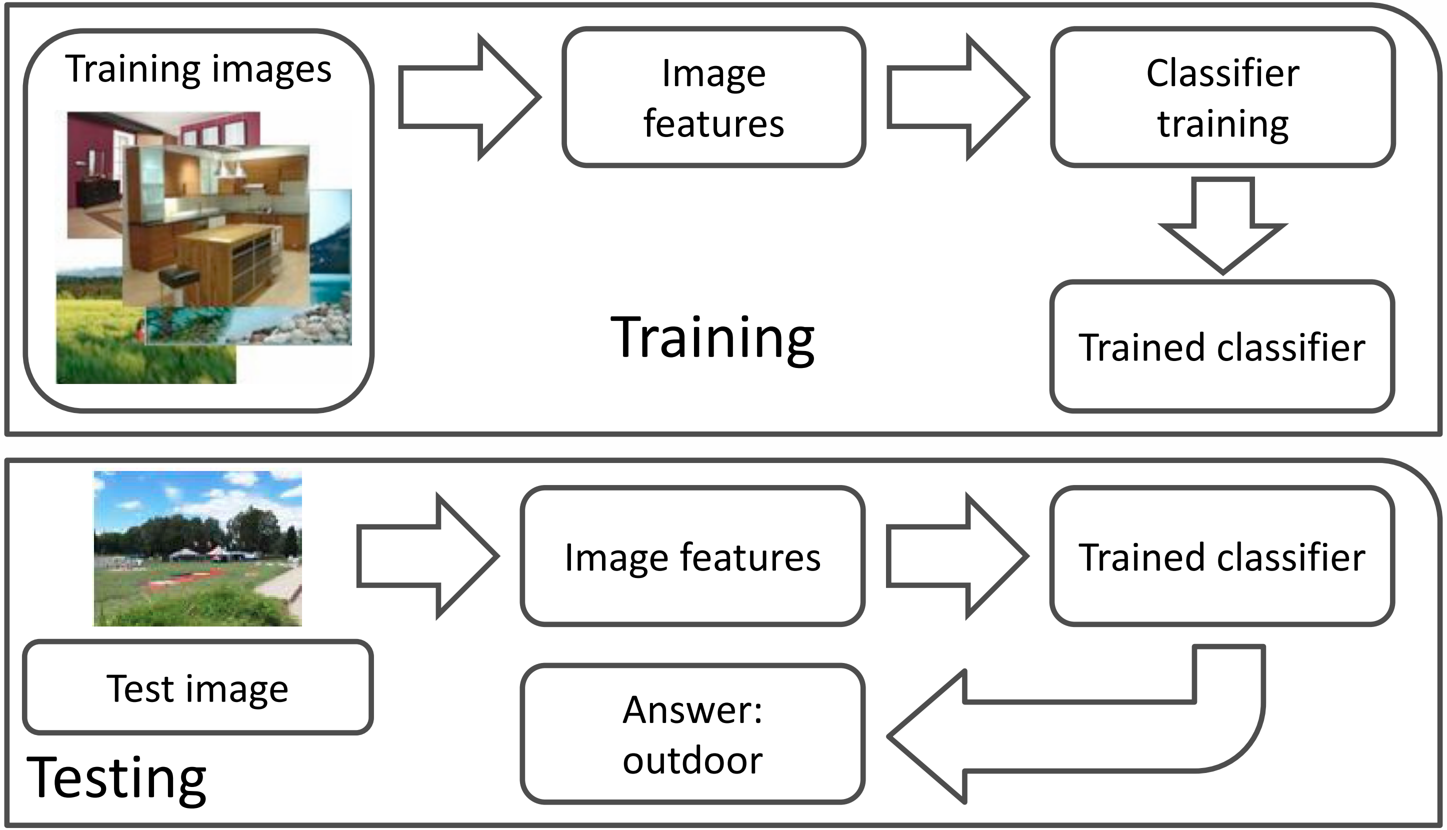
- **Approach:** analysis of only local features of the image, without selection and analysis of individual objects.



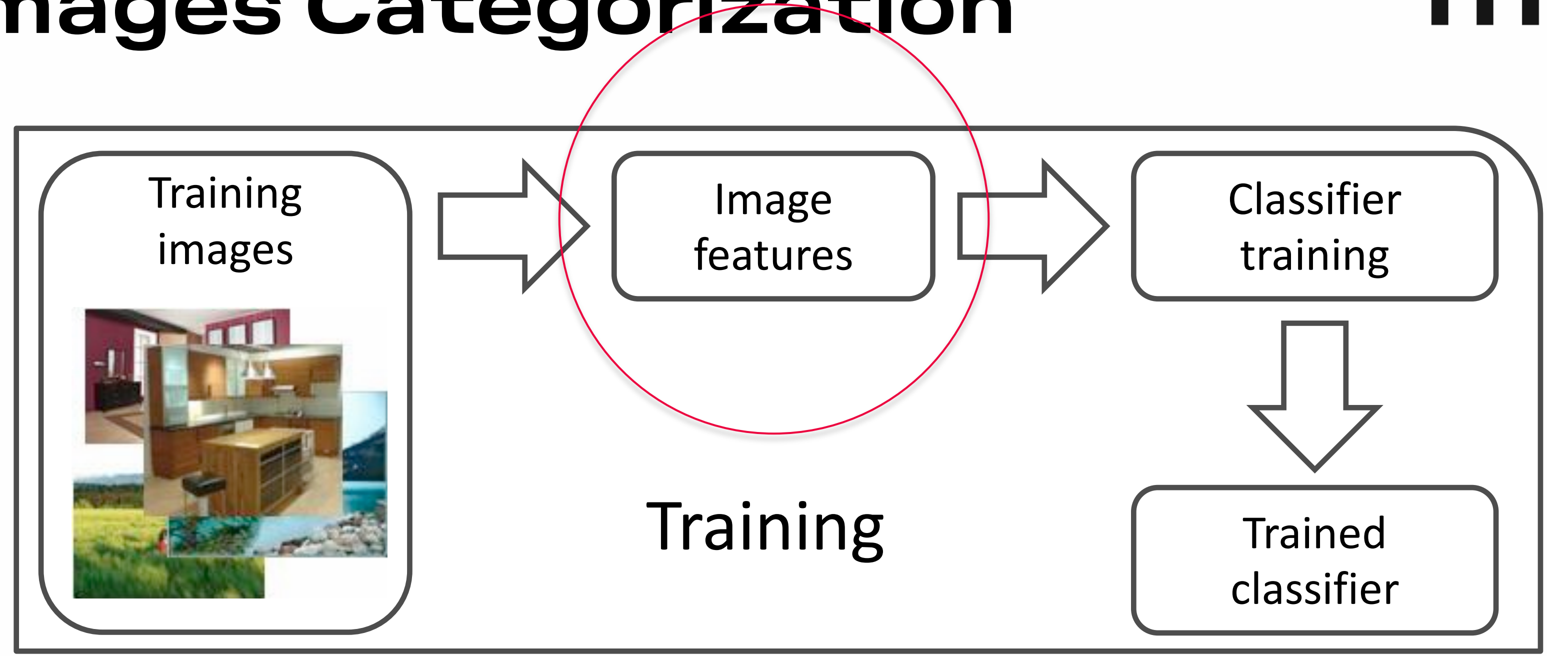
# Images Categorization



# Images Categorization

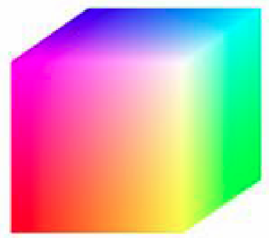


# Images Categorization

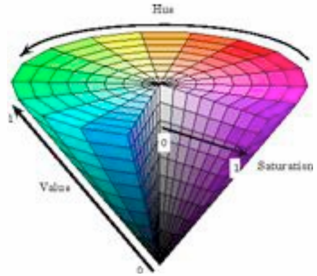



- We will consider the features of the image.
- Various classification methods can be used, SVM is often used.

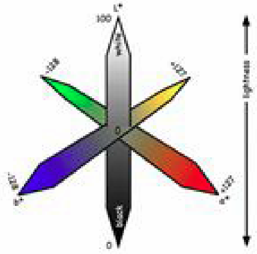
# Images Categorization



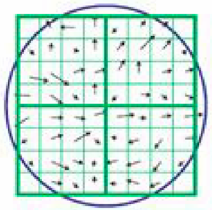
Space
RGB colors



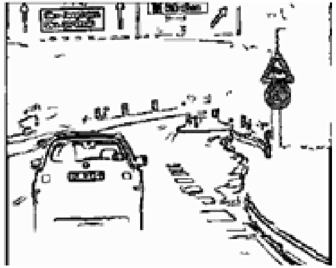

Space
HSV colors



Space
L*a*b colors




Gradients in every pixel



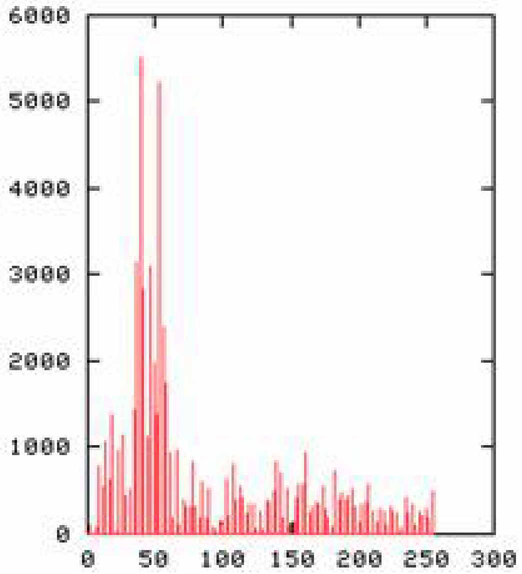
Edge presence and
orientation in each pixel

# Using All Features Directly

1. Bring all images to the same size.
2. We stretch the image into a vector, the pixel signs will become vector elements.
3. For recognition, the feature vector must be of a certain length.

# Images Categorization

- Various attributes can be used (color, edges, gradients, texture, etc.).
- Using histograms is a standard way of non-parametric feature distribution.




# Features Quantization

- If the resolution is too high, then many elements of the feature vector are obtained.
- Quantization is used to reflect features at a lower resolution:
  - For example, all gradient directions are rounded by 45 degrees (8 directions).

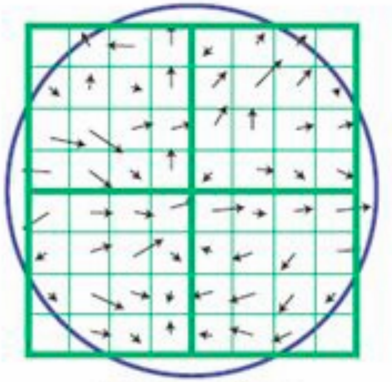
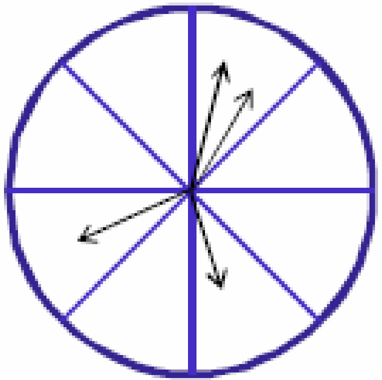
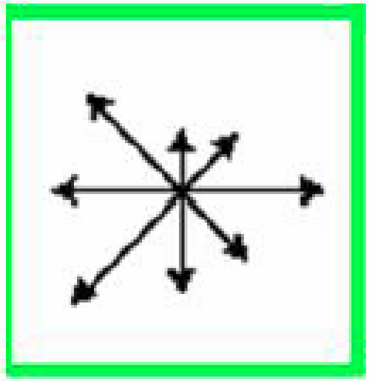


Image gradients



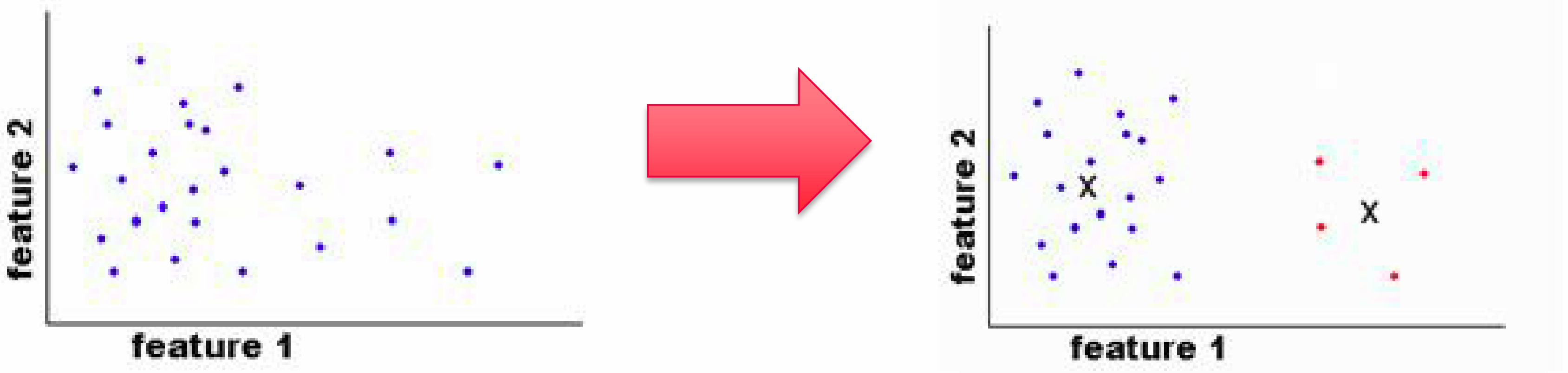


# Features Quantization

- We want to distinguish photographs of tigers from polar bears, for example, by color.
- It is not optimal to use a color histogram on the interval [0, 255] because not all colors occur.

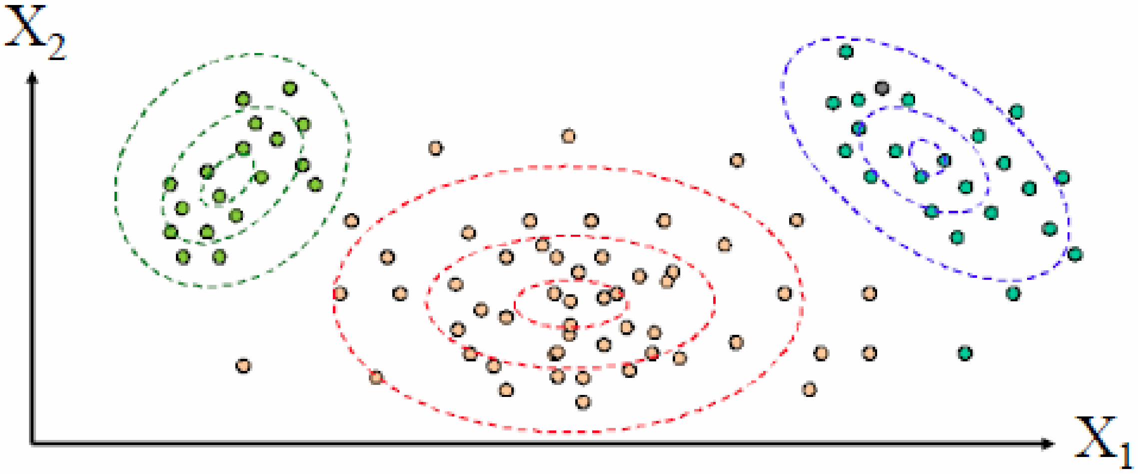
# Adaptive Quantization

- We want to split elements (for example, special pixels) into frequently occurring groups.
- We assign a group number to each element (clustering).



# Clustering

- Given a training sample $X_m = \{x_1, \dots, x_m\}, x_i \in R^m$.
- Sample objects are independent and taken from some unknown distribution $x_i \in P(x)$.
- Needed for all new $x$ evaluate values $P(x)$.



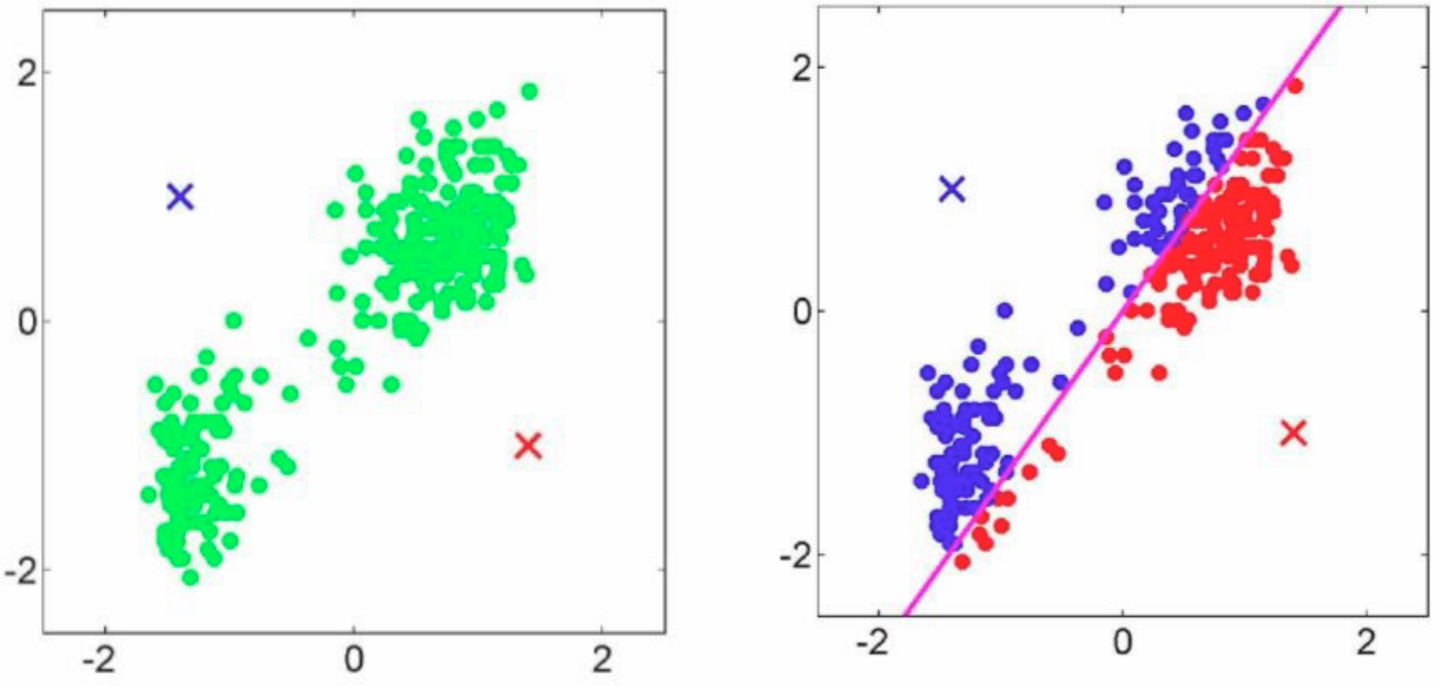
# $k$-Means Clustering

- Minimize the sum of squared Euclidean distances between the points $x_i$ and the nearest cluster centers $m_k$:

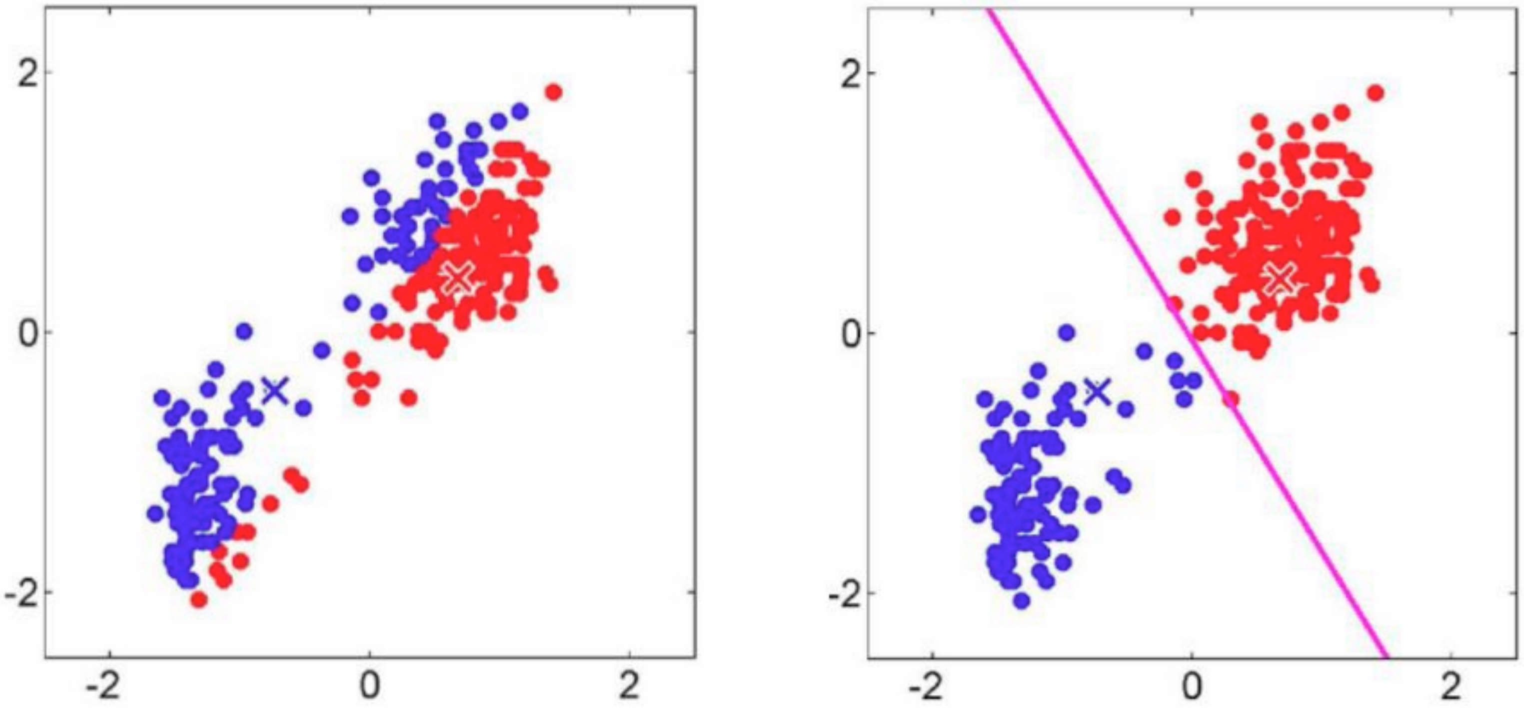
$$D(X, M) = \sum_k \sum_i (x_i - m_k)^2.$$

- **Algorithm:**
  1. Randomly initialize $k$ cluster centers.
  2. Repeat until convergence:
     a) Assign each point to the nearest center.
     b) Recalculate the center of each cluster as the average of all assigned points.

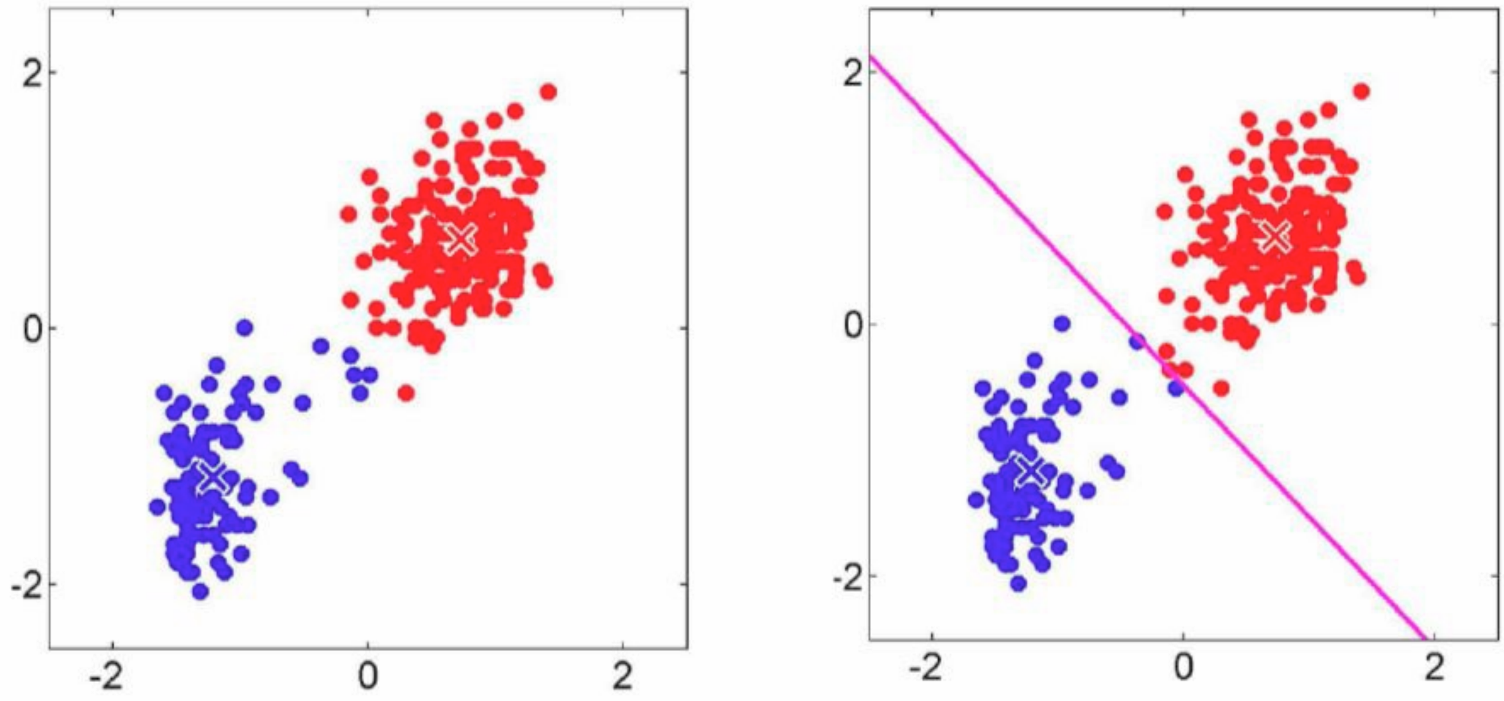
# $k$-Means Clustering



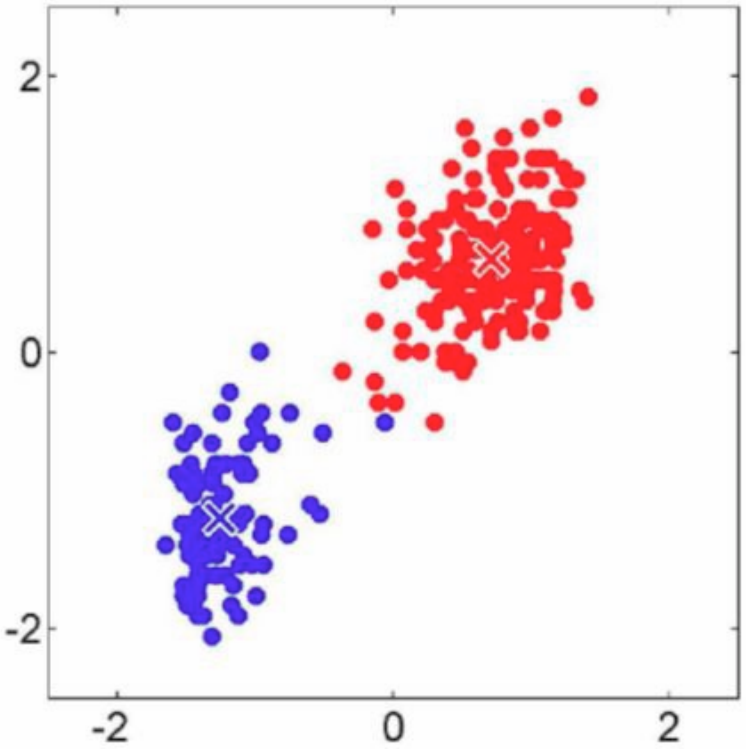
# $k$-Means Clustering



# $k$-Means Clustering

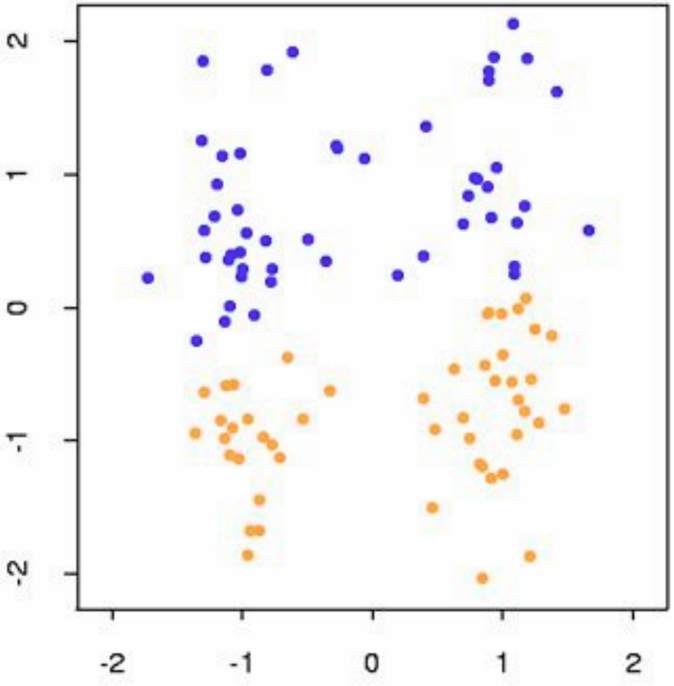


# $k$-Means Clustering



# $k$-Means Clustering

- **Algorithm features:**
  - **Single parameter** – requires knowledge only about the number of clusters.
  - **Randomized** – depends on the initial approximation.
  - **Does not consider the structure** of the clusters themselves.



# Histogram Comparison

- Intersection of histograms (normalized):

$$D(h_i, h_j) = 1 - \sum_{m=1}^{K} \min\left(h_i(m), h_j(m)\right).$$

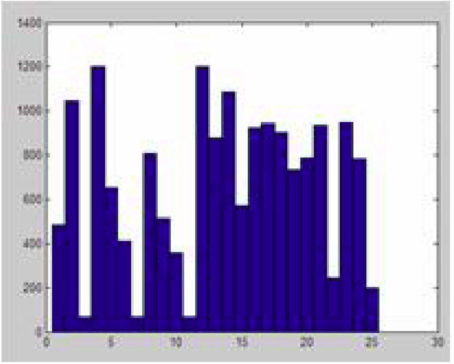
- Distance $L_1$:

$$D(h_1, h_2) = \sum_{i=1}^{N} |h_1(i), h_2(i)|.$$

- Distance $\chi^2$:

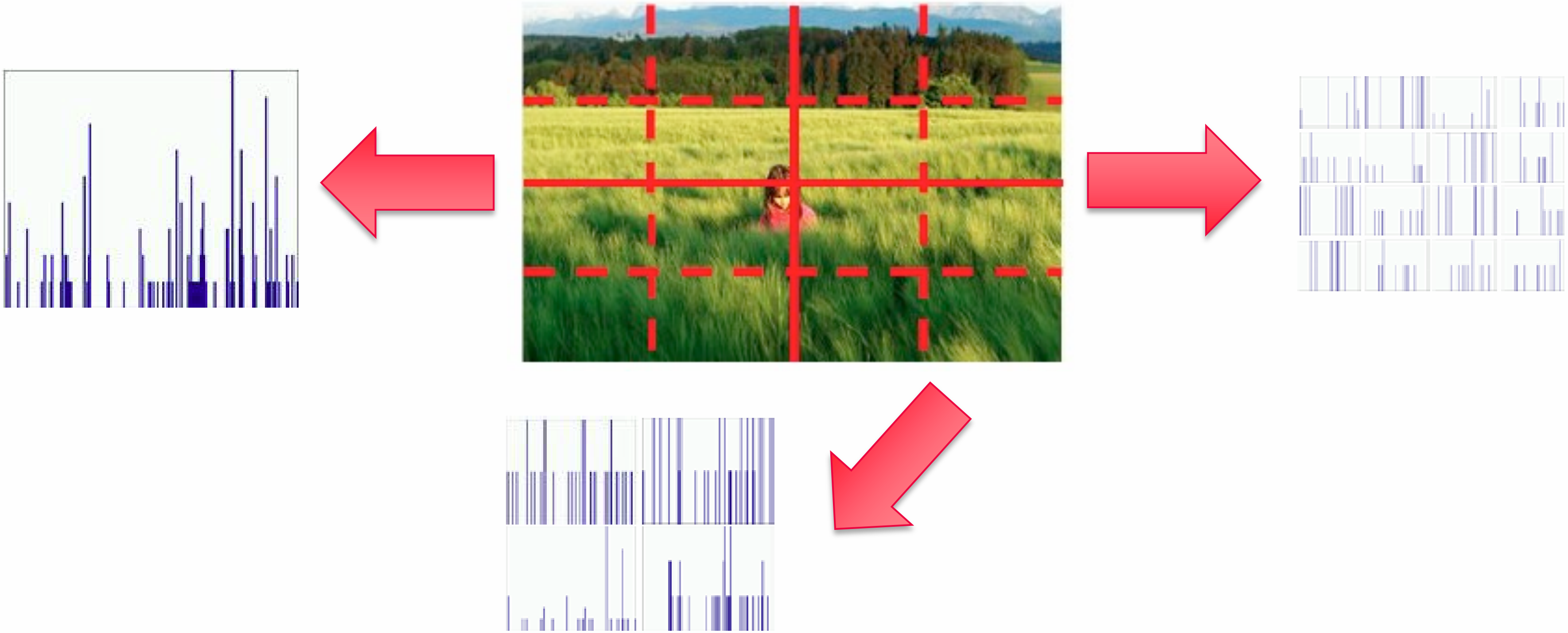
$$D(h_1, h_2) = \sum_{i=1}^{N} \frac{\left(h_1(i) - h_2(i)\right)^2}{h_1(i) + h_2(i)}.$$

# Spatial Distribution




All three of these images have similar color histograms

# Spatial Pyramid of Images

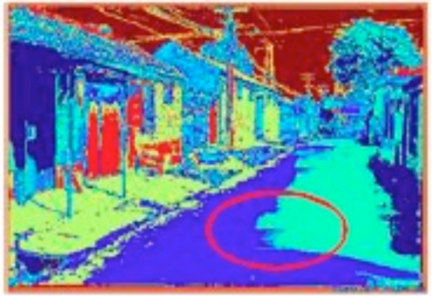


Calculate the histogram in each block and combine all histograms into one feature vector

# General Quantization Scheme

Calculation signs by image → Quantize features → Defining of the area to calculate → Calculate statistics for the selected area

RGB



Quantization 10 levels

71% 29%

Calculated histogram

# THANK YOU FOR YOUR TIME!

ITsMOre than a UNIVERSITY

s.shavetov@itmo.ru