Bag of Visual Words Model

Albert Chung
Department of Computer Science
University of Exeter

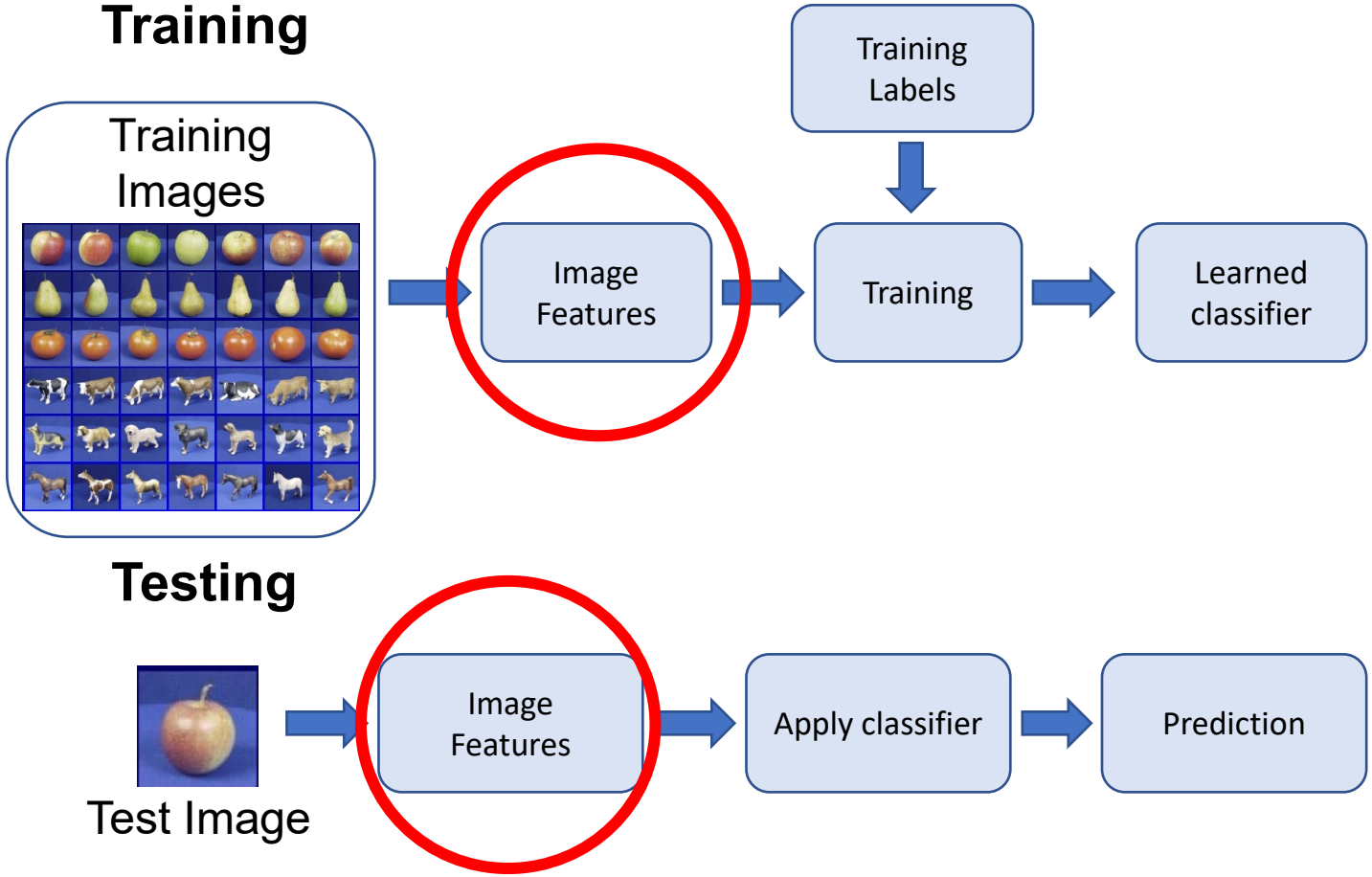
Spring Term 2023

References

- R. Szeliski, “Computer Vision: Algorithms and Applications” (2nd Edition):
  - Image Classification: Chapter 6.2
  - Large-scale Matching and Retrieval: Chapter 7.1.4
- C. M. Bishop, “Pattern Recognition and Machine Learning”:
  - Maximum Margin Classifiers: Chapter 7.1

Image classification

Training and testing steps



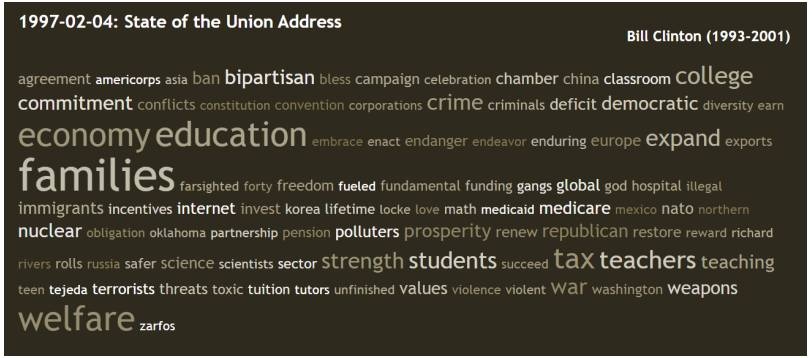
Motivation: Bags of words (BoW)

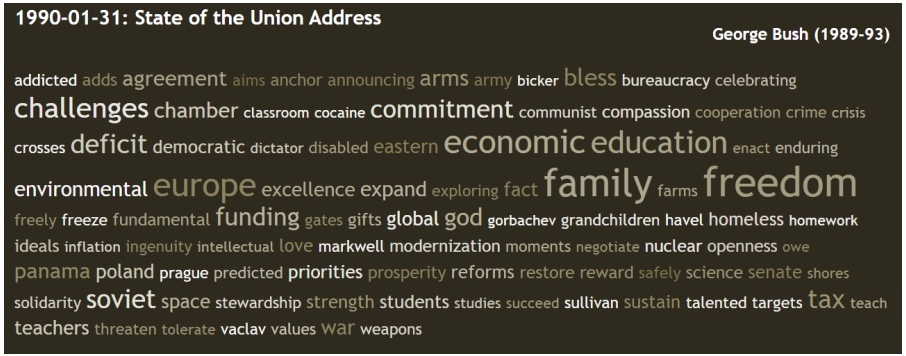
- Order-less document representation: frequencies of words from a dictionary

G. Salton and M. J. McGill, Introduction to Modern Information Retrieval. McGraw-Hill Book Co., New York, 1983.

Motivation: Bags of words (BoW), text mining, and similar document search

- Order-less document representation: frequencies of words from a dictionary

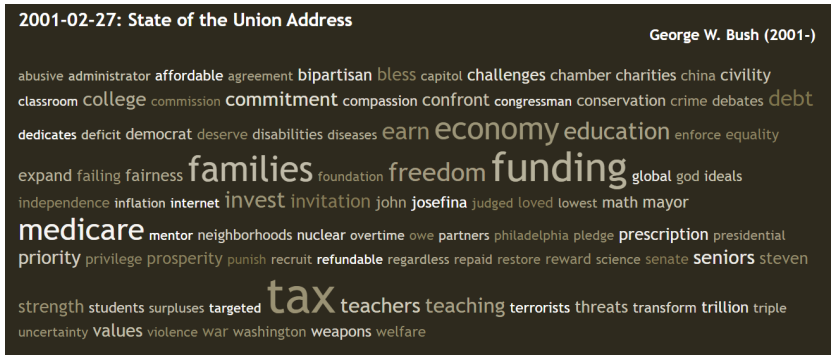


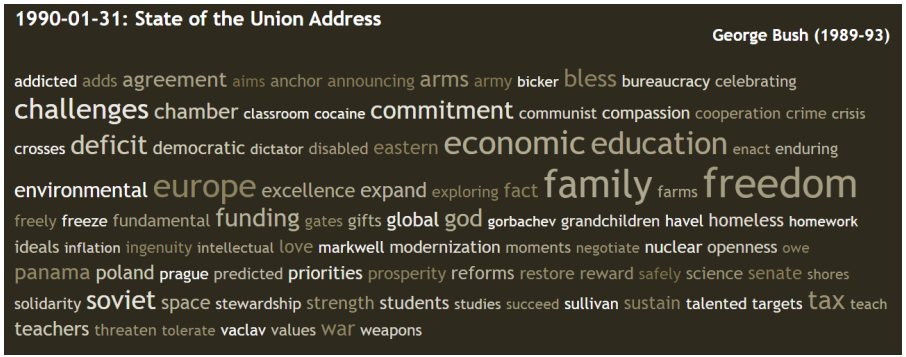


Ref Source: US Presidential Speeches Tag Cloud, https://chir.ag/projects/preztags/

Motivation: Bags of words (BoW), text mining, and similar document search

- Order-less document representation: frequencies of words from a dictionary





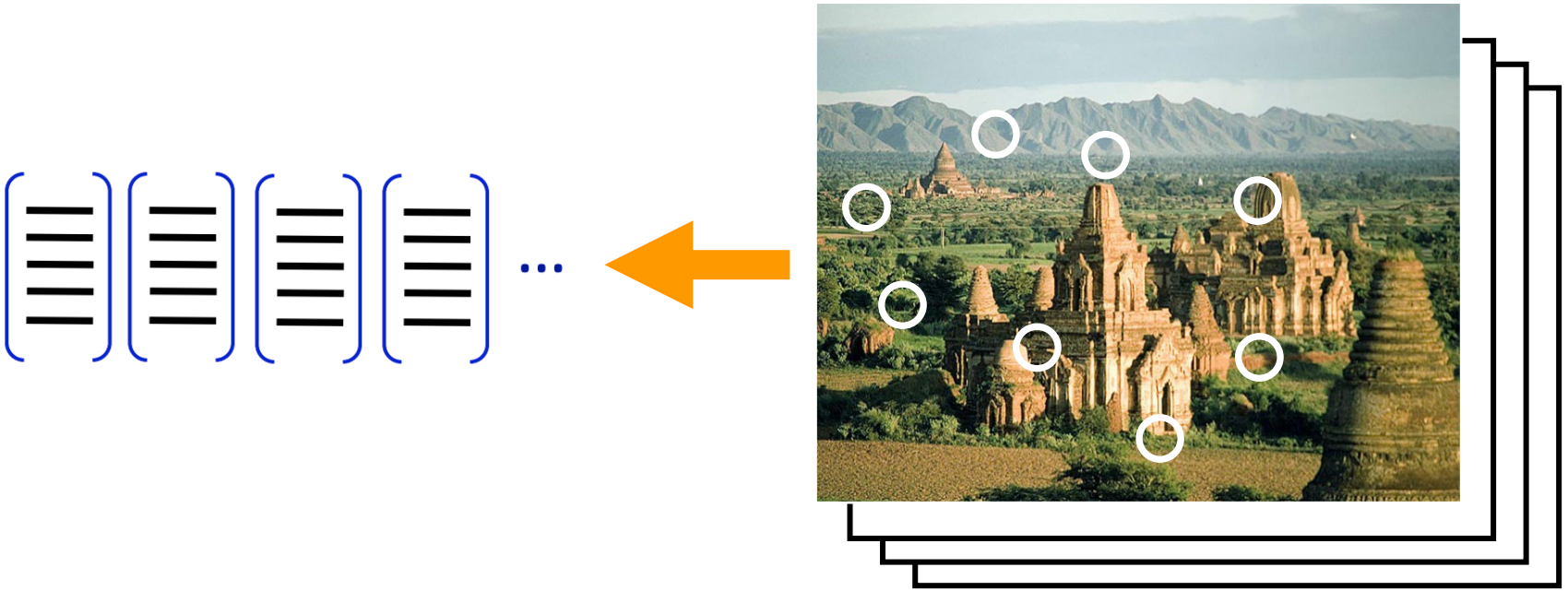
similar document

Ref Source: US Presidential Speeches Tag Cloud, https://chir.ag/projects/preztags/

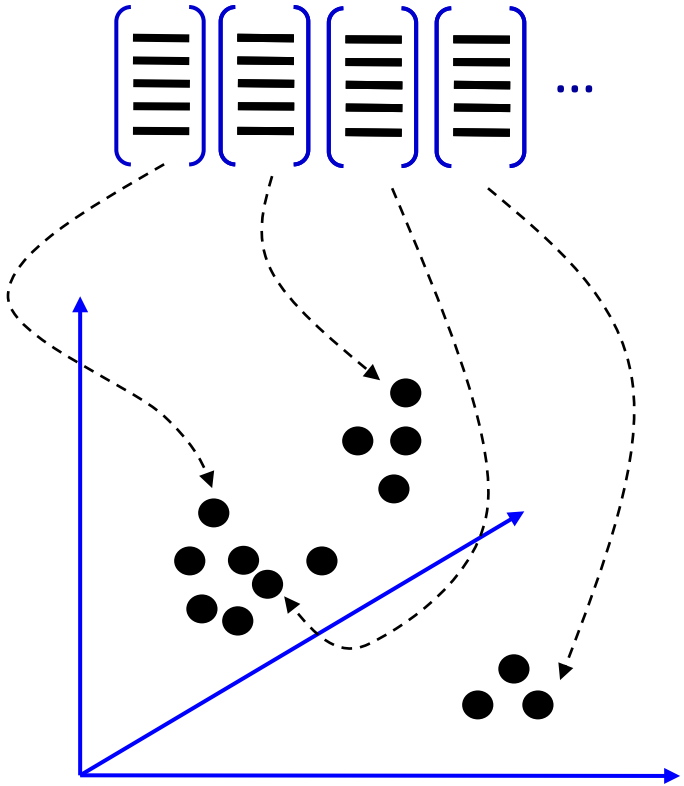
Bag of features / Bag of visual words

Feature Extraction

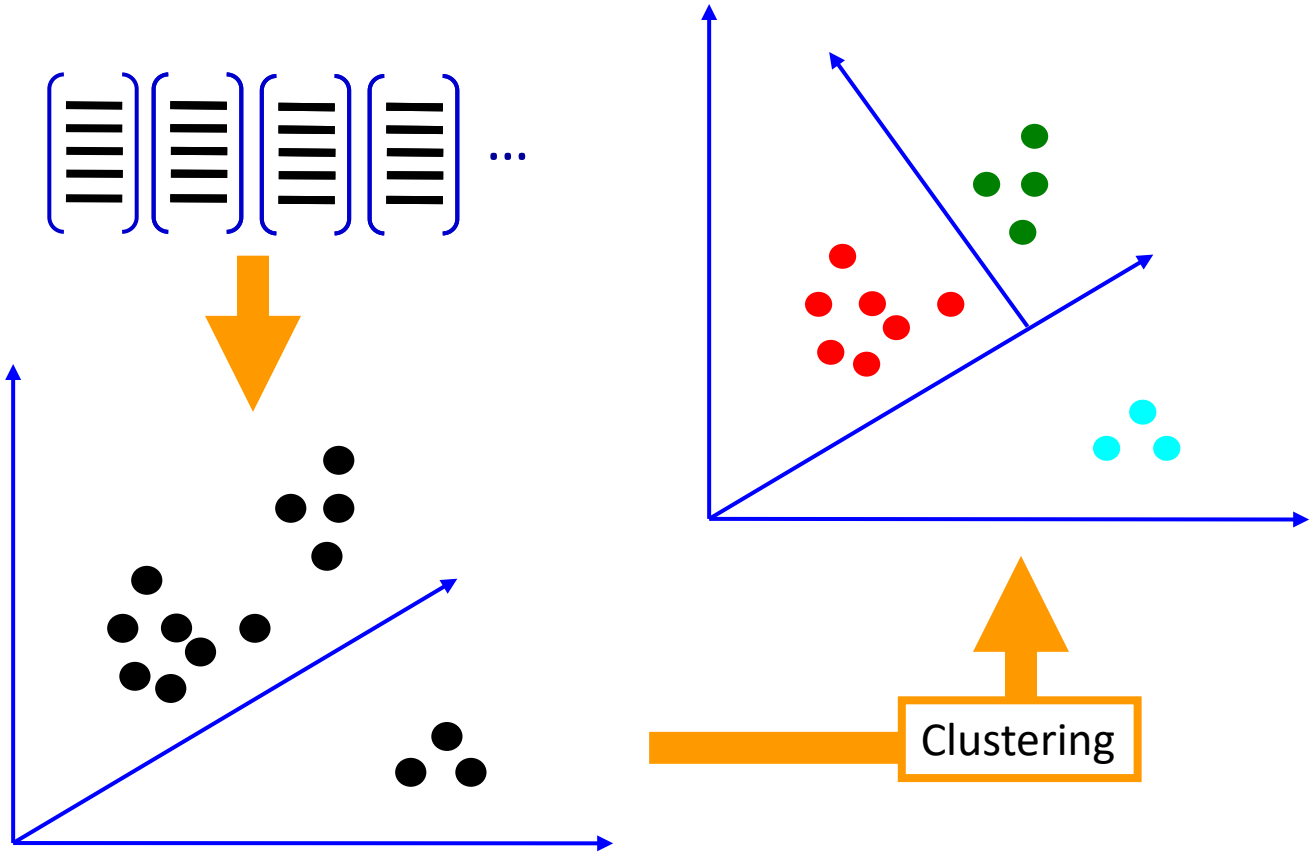
- Detecting features (e.g., corners) and computing descriptors from images
- Gathering descriptors from images in a collection



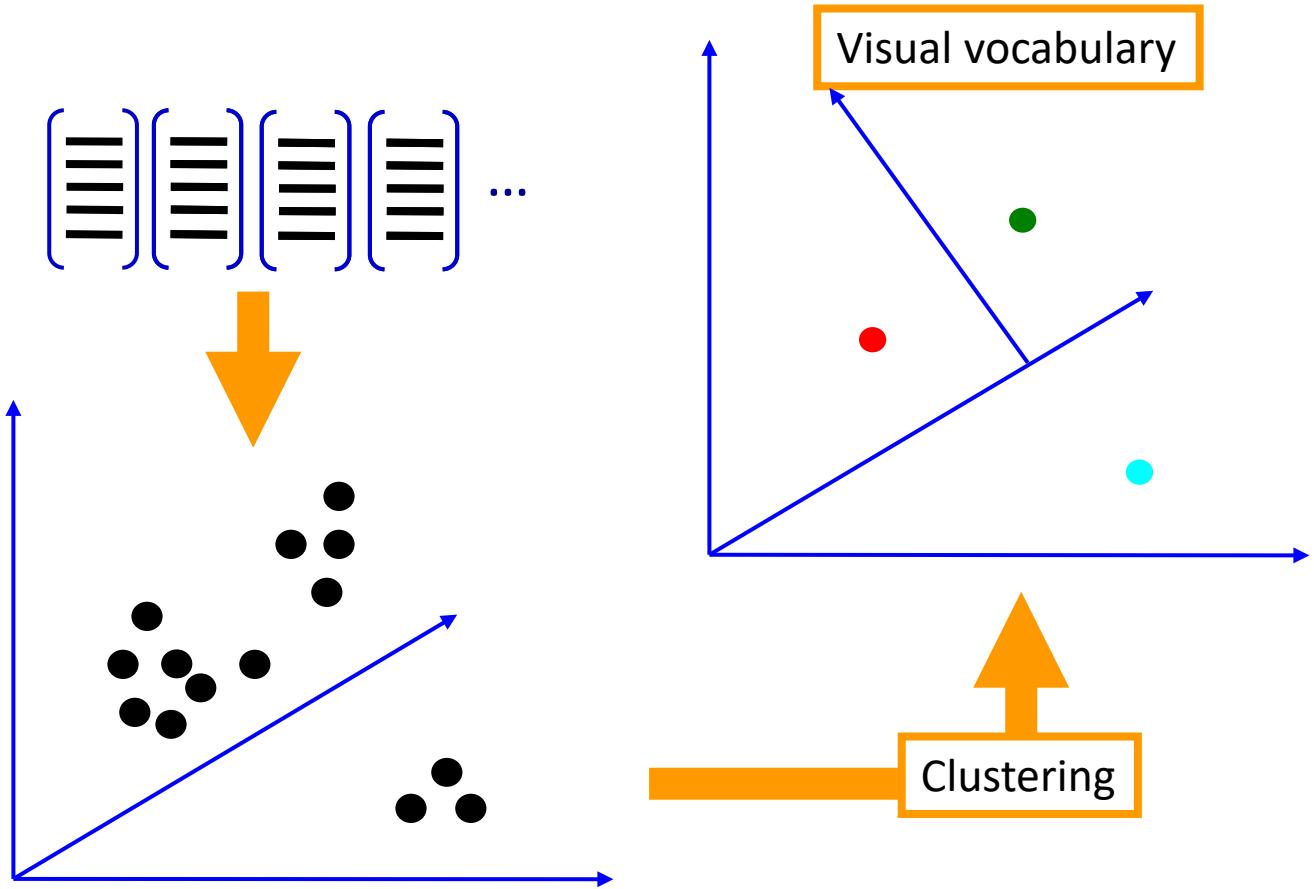
Extracted descriptors from the training set



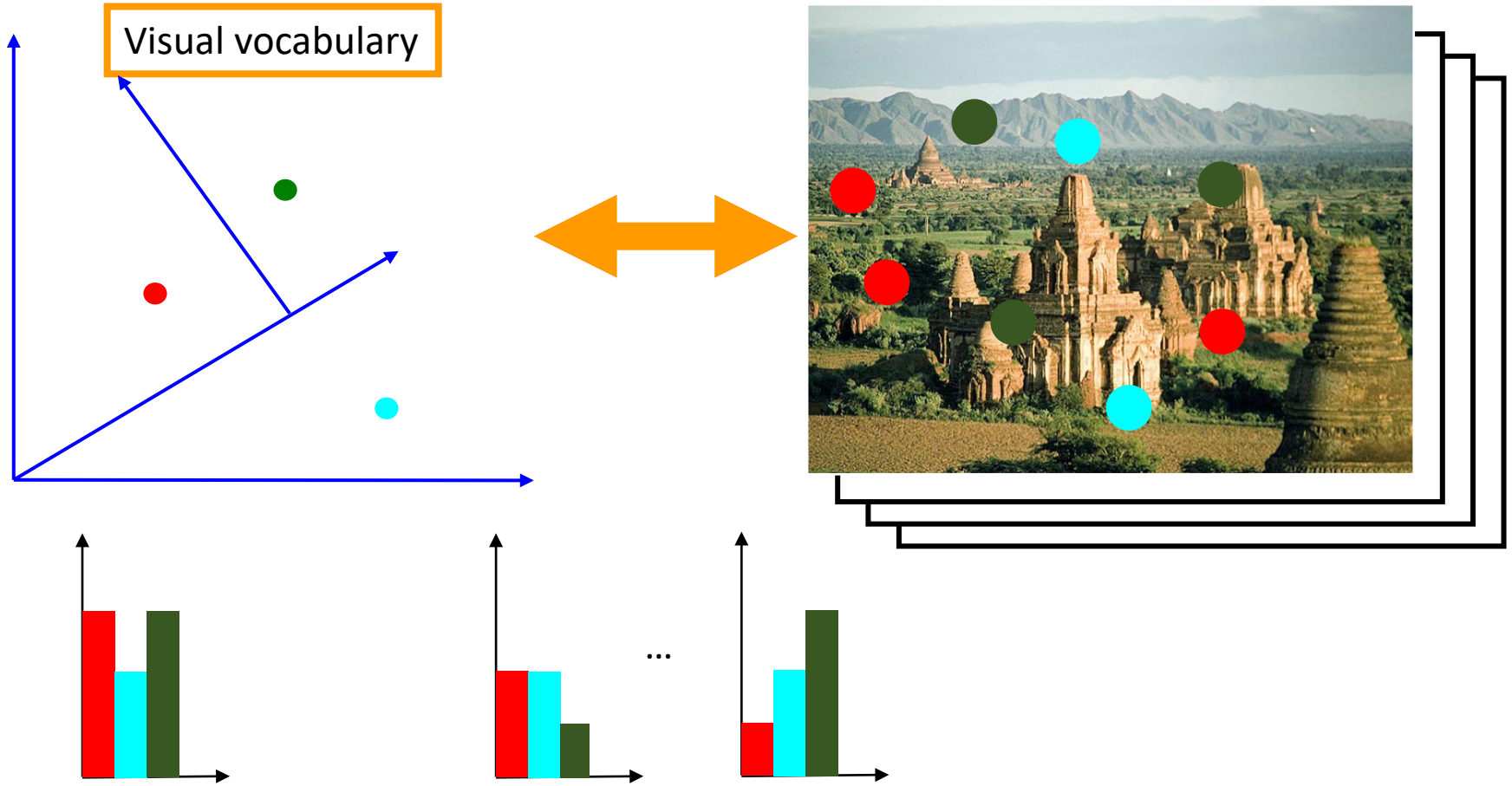
Learning the visual vocabulary



Learning the visual vocabulary or codebook



Feature Quantization

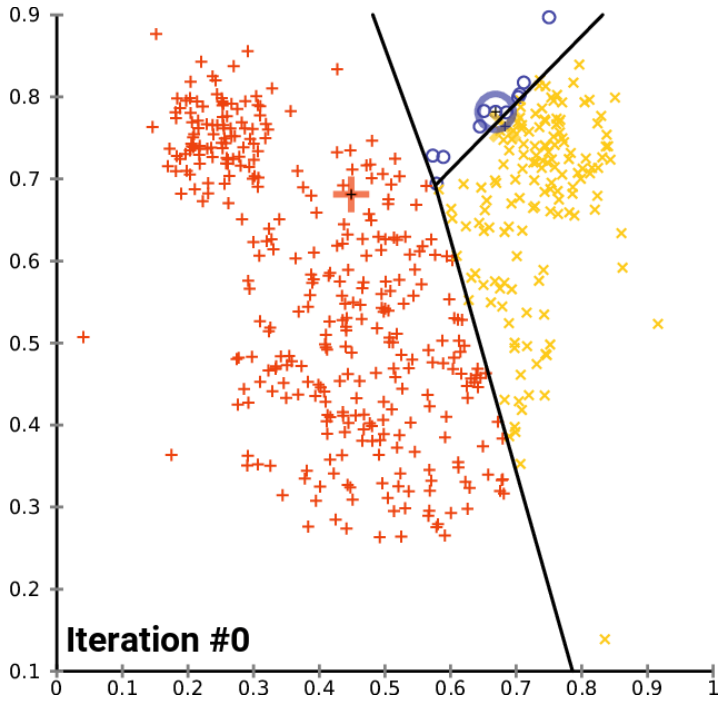


K-means clustering

- Want to minimize sum of squared Euclidean distances between features X_i and their nearest cluster centers m_k

$$D(X, M) = \sum_{k} \sum_{i \in C_k} (X_i - m_k)^2$$

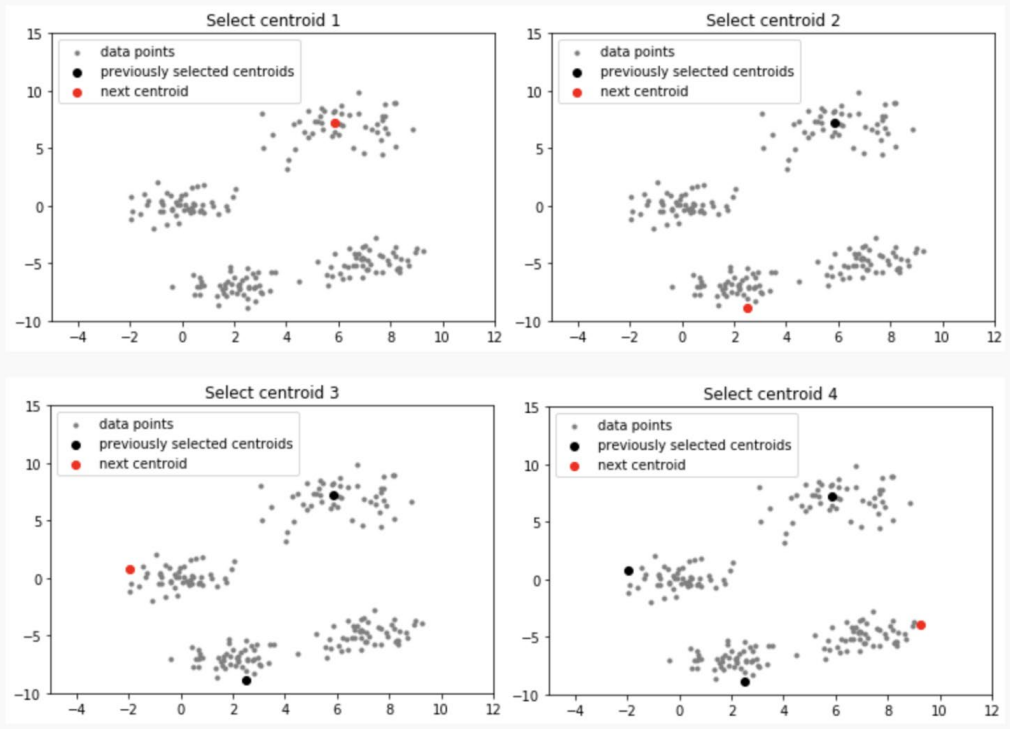
- Algorithm:
  - Randomly initialize K cluster centers
  - Iterate until convergence:
    - Assign each feature to the nearest center
    - Recompute each cluster center as the mean of all features assigned to it



Ref Source: https://en.wikipedia.org/wiki/K-means_clustering

K-means++ clustering

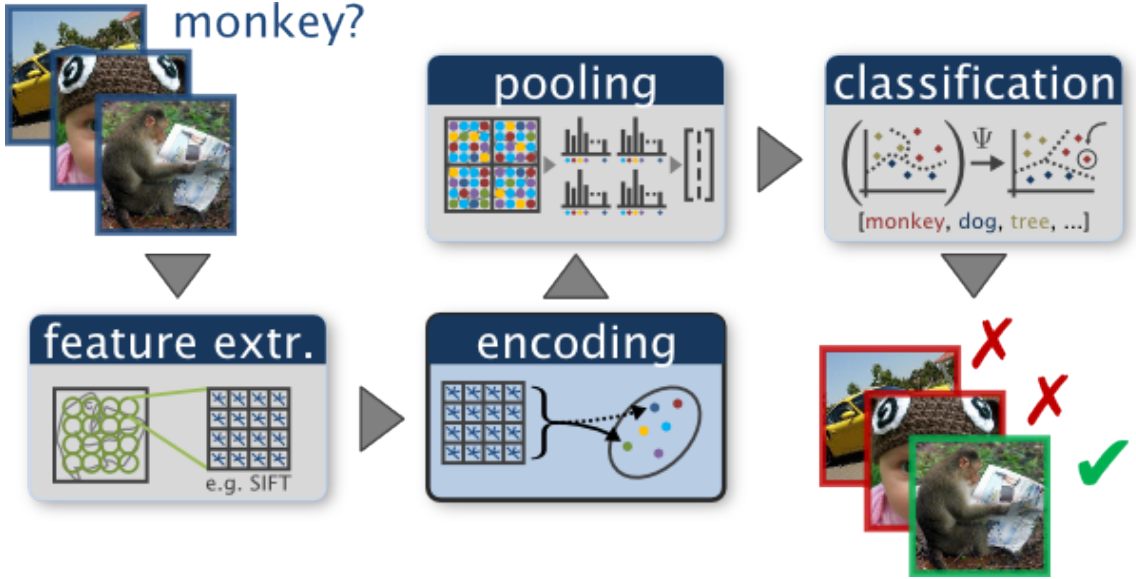
- Algorithm:
  - [1] Randomly select the first cluster center
  - [2] Select the next cluster center which is spread out
  - Do [1] and [2] for k cluster centers



K-means++: https://www.geeksforgeeks.org/ml-k-means-algorithm

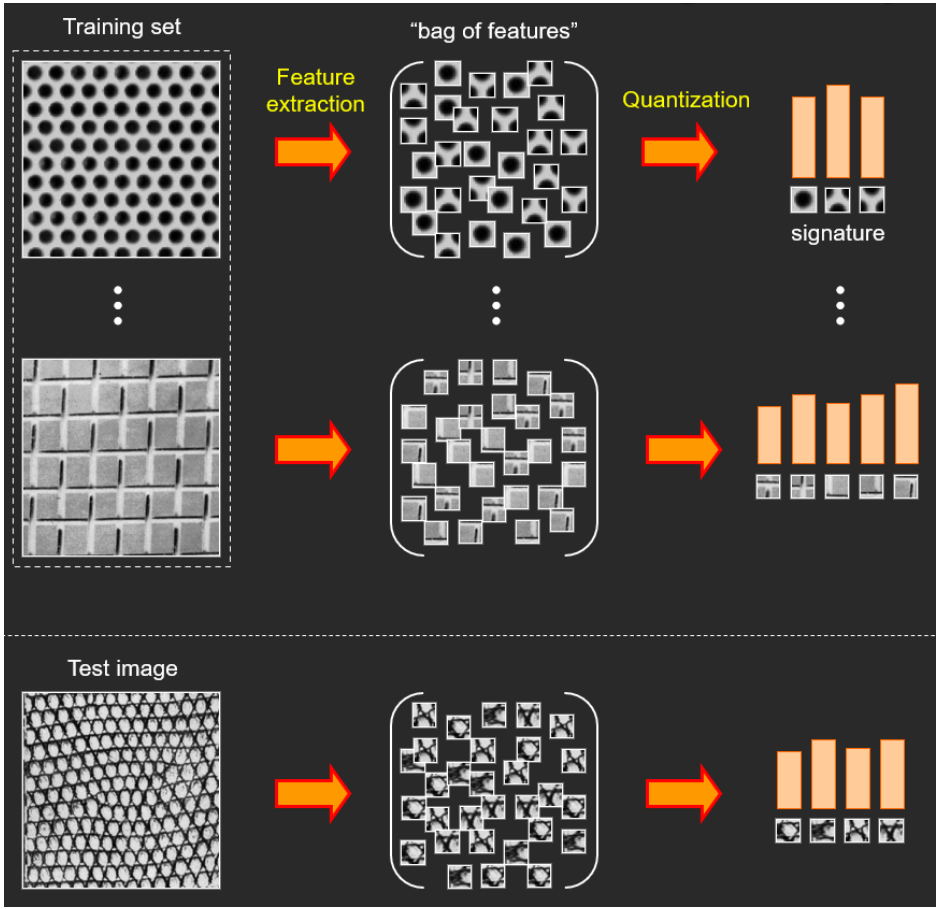
Bag of features: Outline

1. Extract local features
2. Learn visual vocabulary or cluster local features
3. Quantize or pool local features using visual vocabulary or codebook
4. Represent images by frequencies of “visual words”



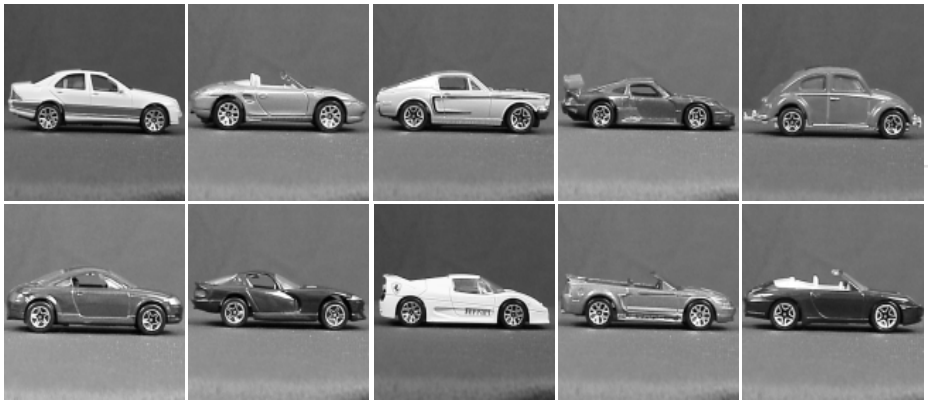
Bag of visual words

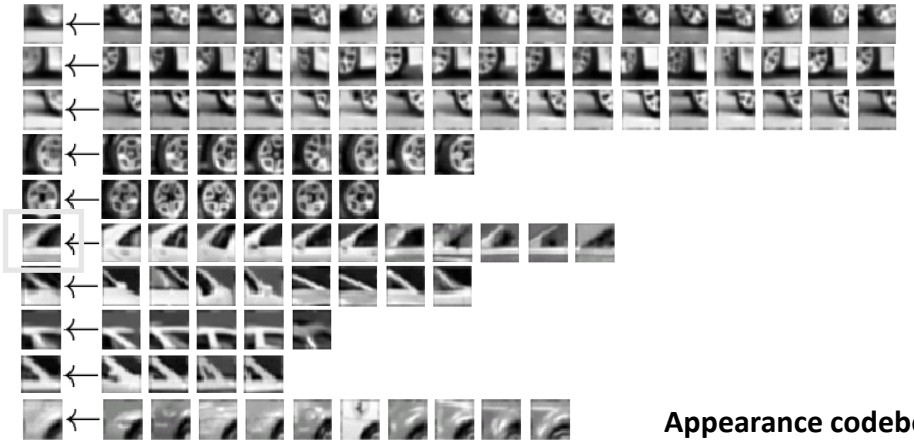
- The content can be inferred from the frequencies of words that happen in a document.
- Pixels are converted to visual words (a codebook/dictionary).
- The content can be inferred from the frequencies of visual words that appear in an image.



Ref Source: Local, Semi-Local and Global Models for Texture, Object and Scene Recognition, S. Lazebnik. Ph.D. Dissertation, May 2006

Recall: visual vocabularies





Appearance codebook

Necessity of spatial information

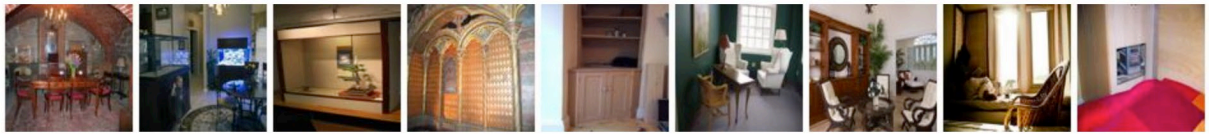
a_abbey(46368)

a_airfield(10910)

a_airplane_cabin(5152)

a_airport_terminal(16174)

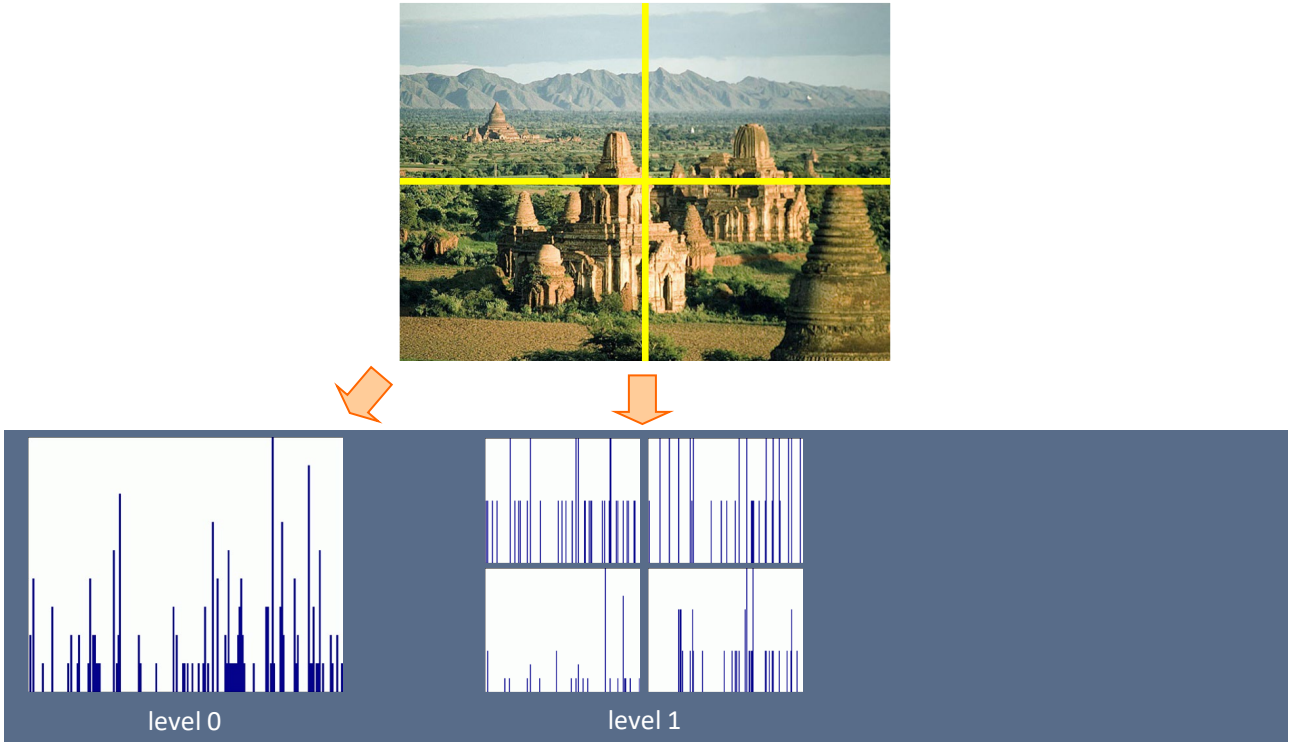
a_alcove(4966)



Spatial pyramids

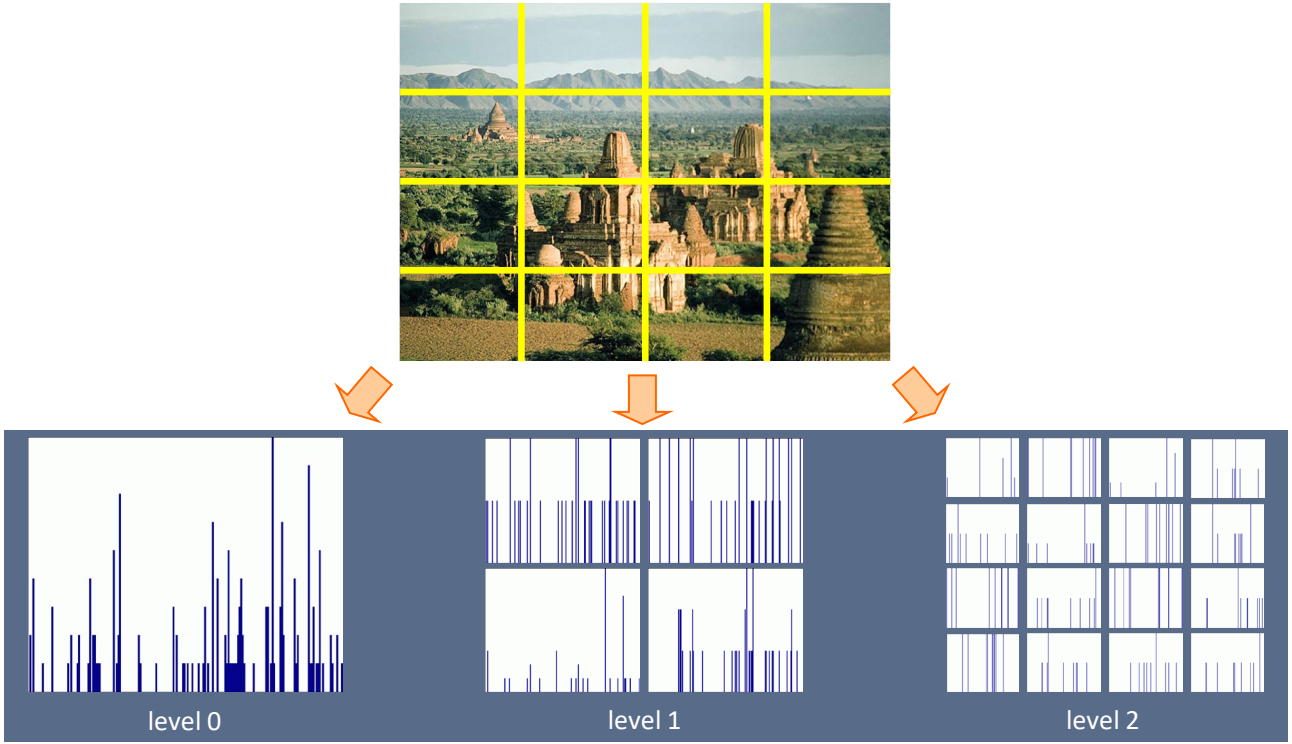
Lazebnik et al., Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories, CVPR, 2006.

Spatial pyramids



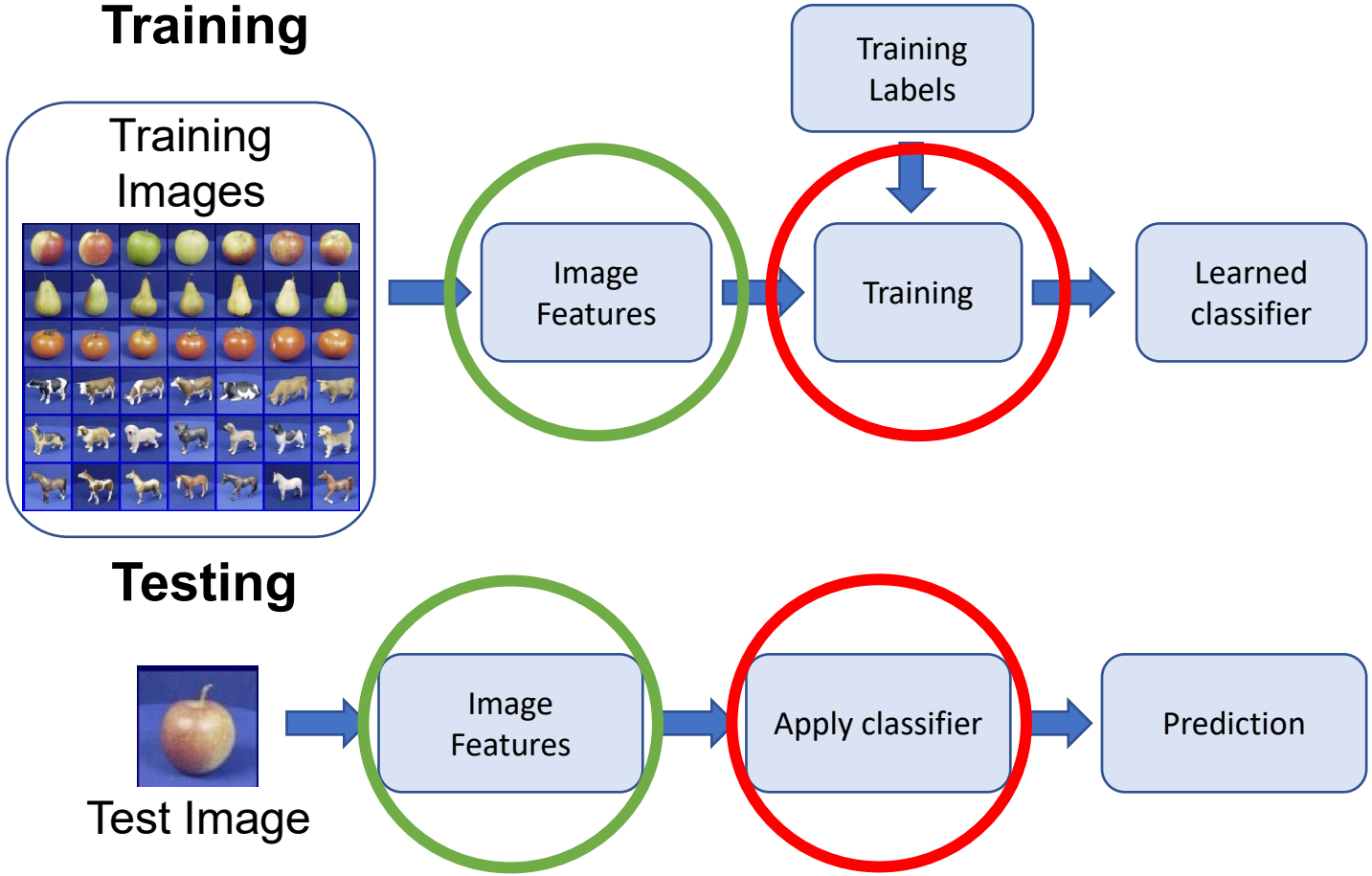
Lazebnik et al., Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories, CVPR, 2006.

Spatial pyramids



Lazebnik et al., Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories, CVPR, 2006.

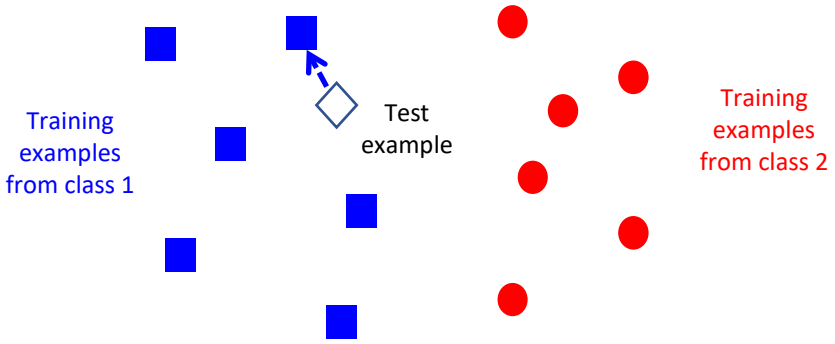
Training and testing steps



Classifier

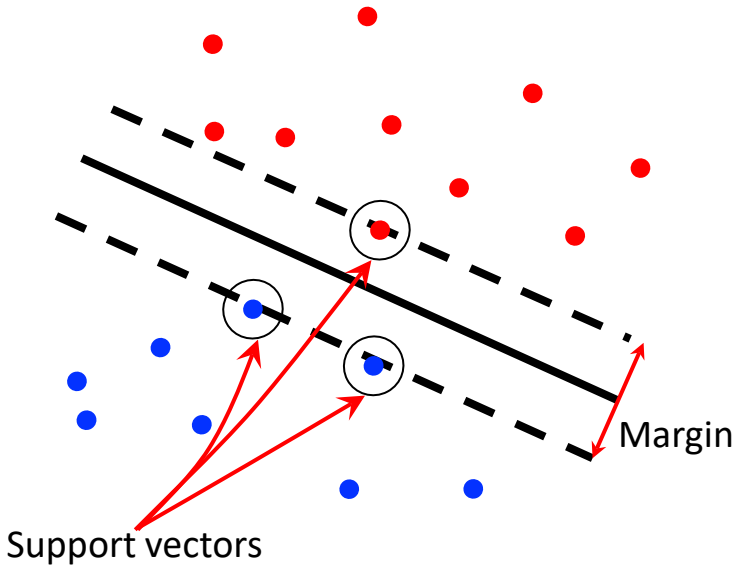
Nearest Neighbor Classifier

- f(**x**) = label of the training example nearest to **x**
- All we need is a distance or similarity function for our inputs



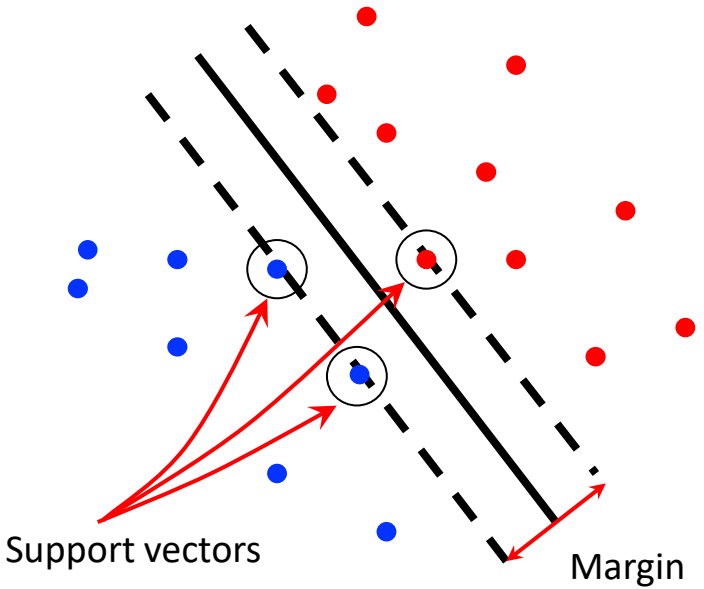
Large or Maximum Margin Classifier

- Find hyperplane that maximizes the margin between the positive and negative examples



Classifier: support vector machines

- Find hyperplane that maximizes the margin between the positive and negative examples



$\mathbf{x}_i$ positive $(y_i = 1)$: $\mathbf{x}_i \cdot \mathbf{w} + b \geq 1$

$\mathbf{x}_i$ negative $(y_i = -1)$: $\mathbf{x}_i \cdot \mathbf{w} + b \leq -1$

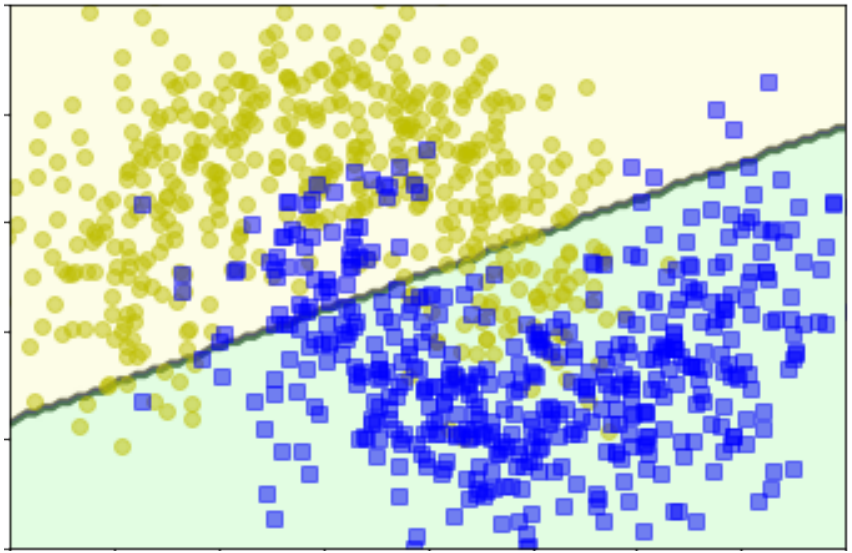
For support vectors, $\mathbf{x}_i \cdot \mathbf{w} + b = \pm 1$

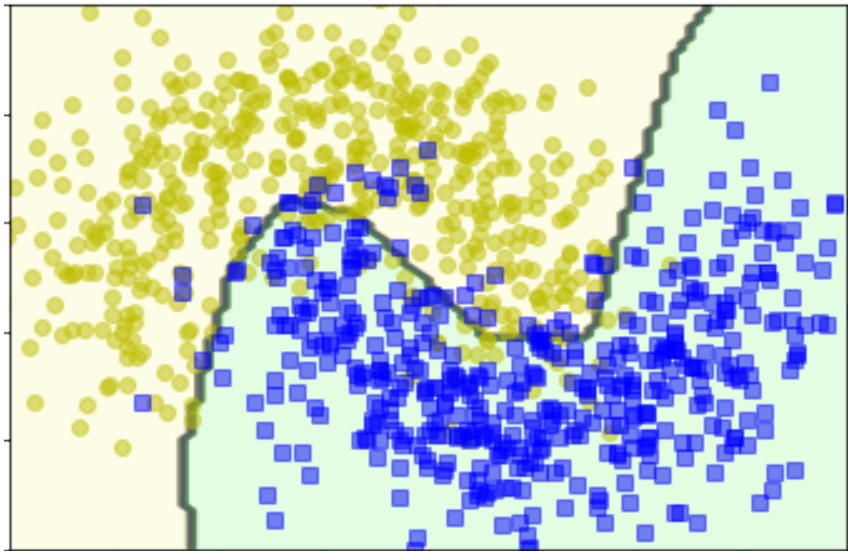
Distance between point and hyperplane: $\dfrac{|\mathbf{x}_i \cdot \mathbf{w} + b|}{||\mathbf{w}||}$

Therefore, the margin is $2 / ||\mathbf{w}||$

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998

Real data are non-linear



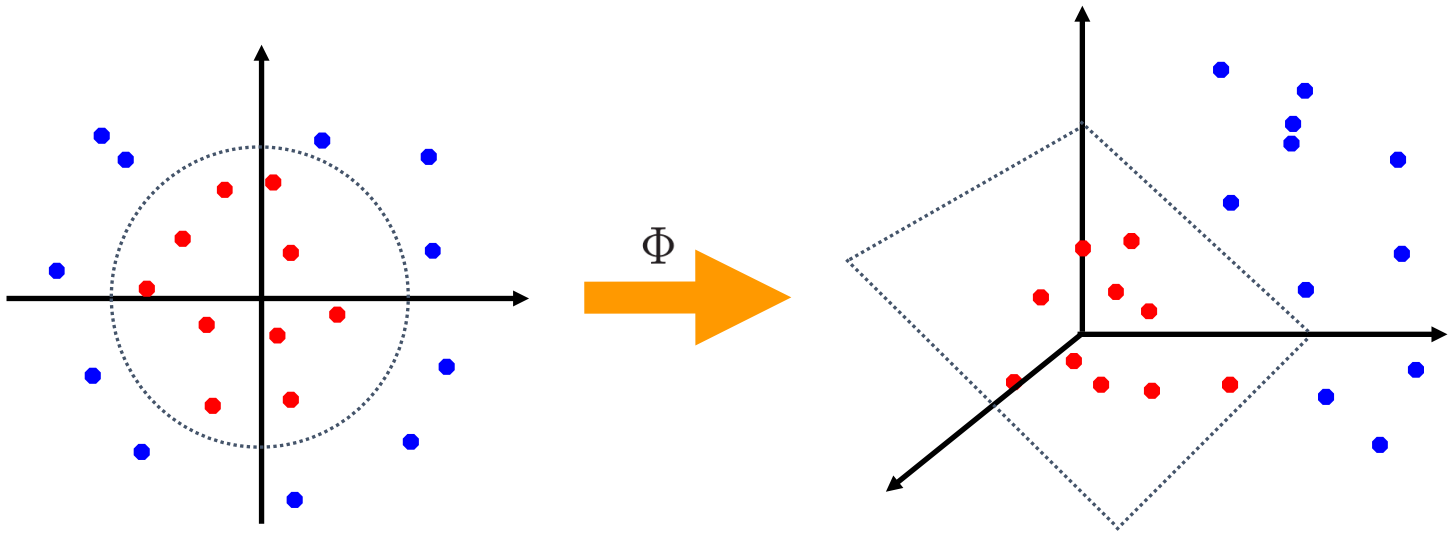


Nonlinear SVMs

Map the original input space to some higher-dimensional feature space where the training set is separable:

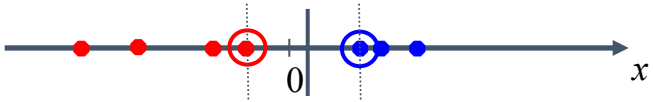
$$\Phi : \mathbb{R}^2 \to \mathbb{R}^3$$

$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}\, x_1 x_2, x_2^2)$$



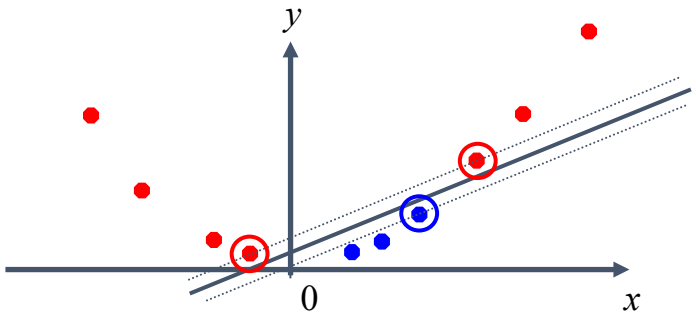
Nonlinear SVMs

- Linearly separable dataset in 1D:



- Non-separable dataset in 1D:

- We can map the data to a higher-dimensional space:



$y=ax^2$

Kernel trick

- **General idea:** the original input space can always be mapped to some higher-dimensional feature space where the training set is separable

- **The kernel trick:** instead of explicitly computing the lifting transformation $\boldsymbol{\varphi}(\mathbf{x})$, define a kernel function K such that

$$K(\mathbf{x}, \mathbf{y}) = \boldsymbol{\varphi}(\mathbf{x}) \cdot \boldsymbol{\varphi}(\mathbf{y})$$

- K is called the Gram matrix, which is positive semidefinite

Schölkopf and Smola, Kernel Methods and Support Vector Machines, 2003.
Implementation details: https://scikit-learn.org/stable/modules/svm.html

Summary

- Bag of visual words (BoVW) model for image classification
- Different steps of BoVW
- Spatial pyramids histogram for BoVW
- NN classifier and SVM