Master in Computer Vision *Barcelona*

Module 3: Machine learning for computer vision

Project: Bag of Visual Words Image Classification

Lecturer: Ramon Baldrich, ramon.baldrich@uab.cat

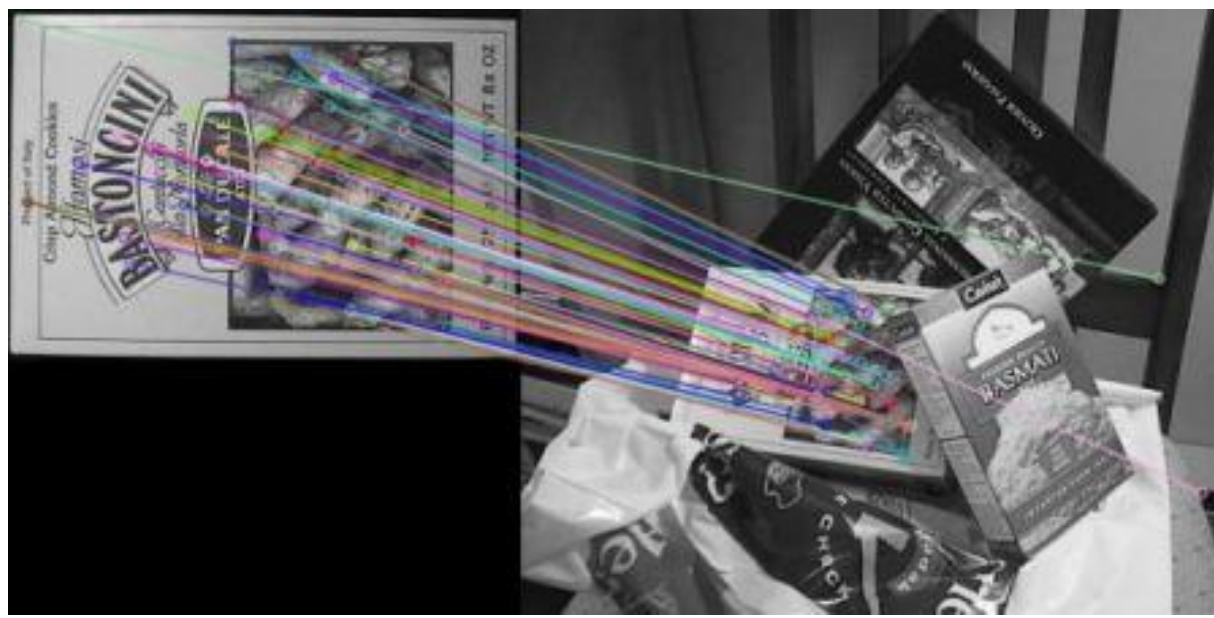
Preamble: Local descriptors

Local descriptors

- keypoint detection
- local description with strong invariance

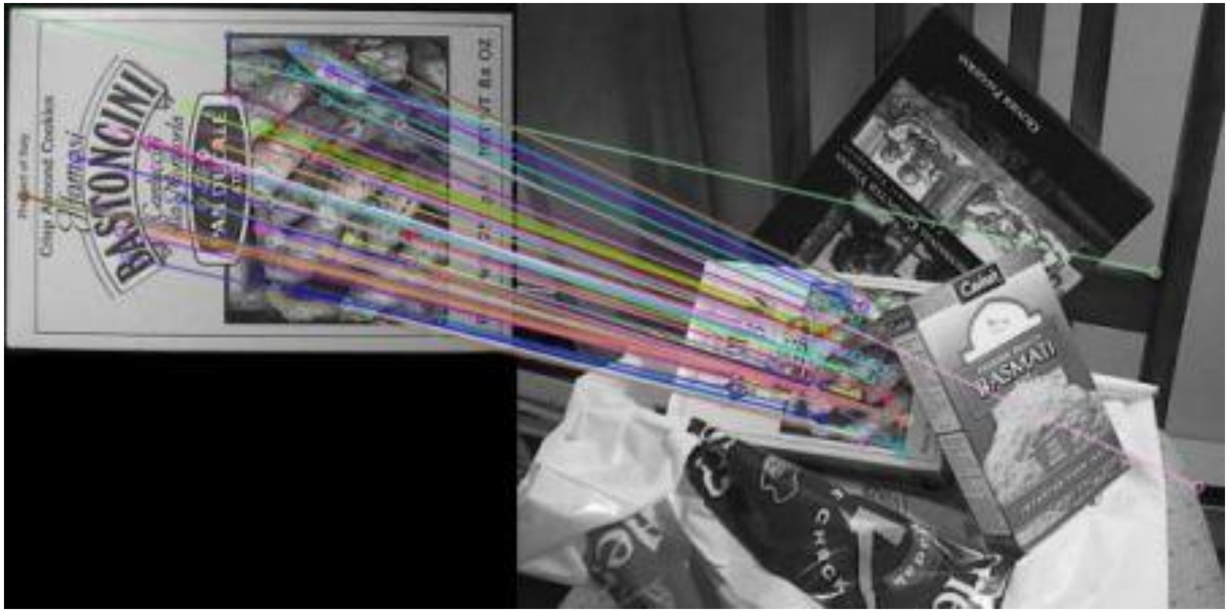
Object detection, localization, recognition, matching…

- Pair template objects within clutter environments



Preamble: Local descriptors

- SIFT (D. Lowe ICCV99, IJCV04)
- SURF,
- KAZE,
- BRIEF,
- BRISK,
- ORB...



Preamble: Local descriptors

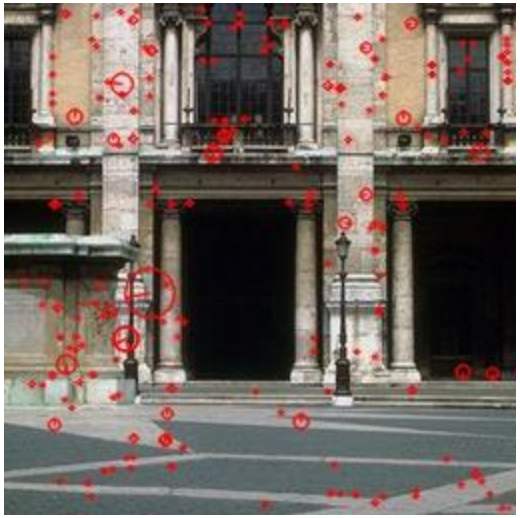
Can we use such local features for image categorization?

Preamble: Local descriptors

Use of local features (e.g. SIFT) for image categorization

Robust local features:

- Scale
- Viewpoint
- Partial occlusions
- Noise

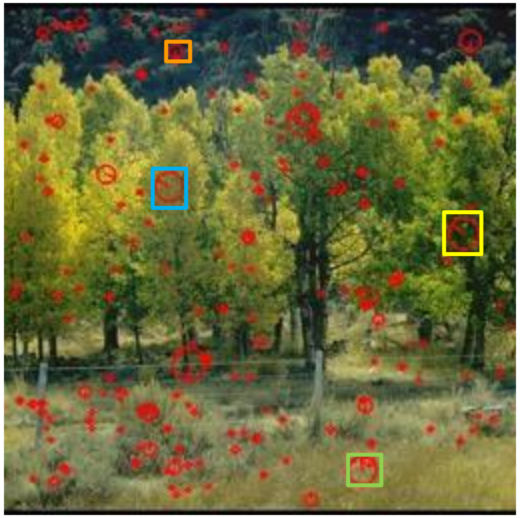


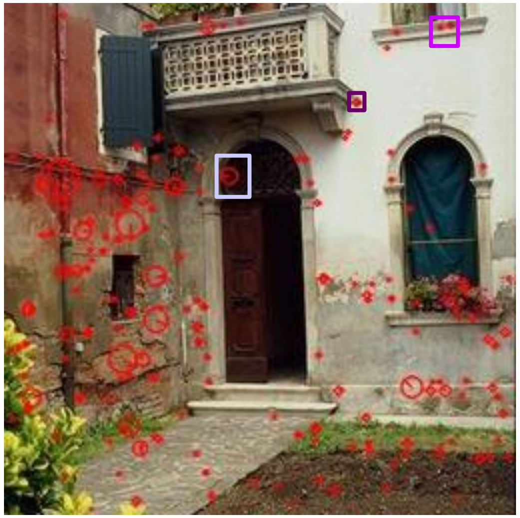
Preamble: Local descriptors

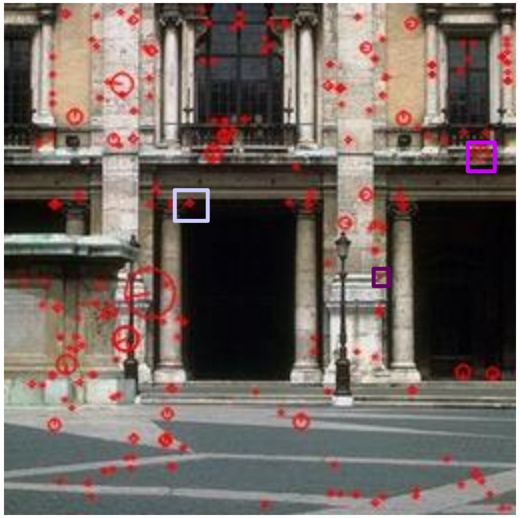
Use of local features for image categorization

Basic assumption:

- Images of the same class have similar local descriptors





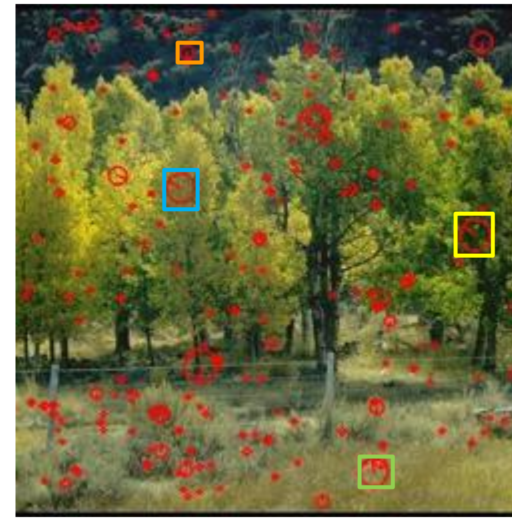


Preamble: Local descriptors

Use of local features for image categorization

Basic assumption:

- Images of the same class have similar local descriptors
- Images of different classes have different local descriptors

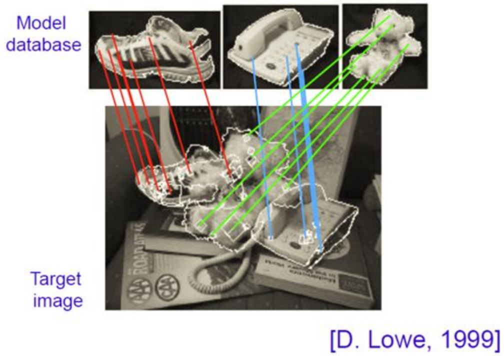


Motivation

Local features are **well suited** for **image categorization**

A **generic approach** could be:

1. Local feature extraction and description (ex. SIFT)
2. Matching local features based on similarity of local appearance
   - For every keypoint in one image find the closest keypoint (in the feature space) in the other image
   - Verify matches based on semi-local/global geometric relations



D. Lowe. *Object Recognition from Local Scale-Invariant Features*. ICCV 1999

Motivation

Local features are **well suited** for **image categorization**

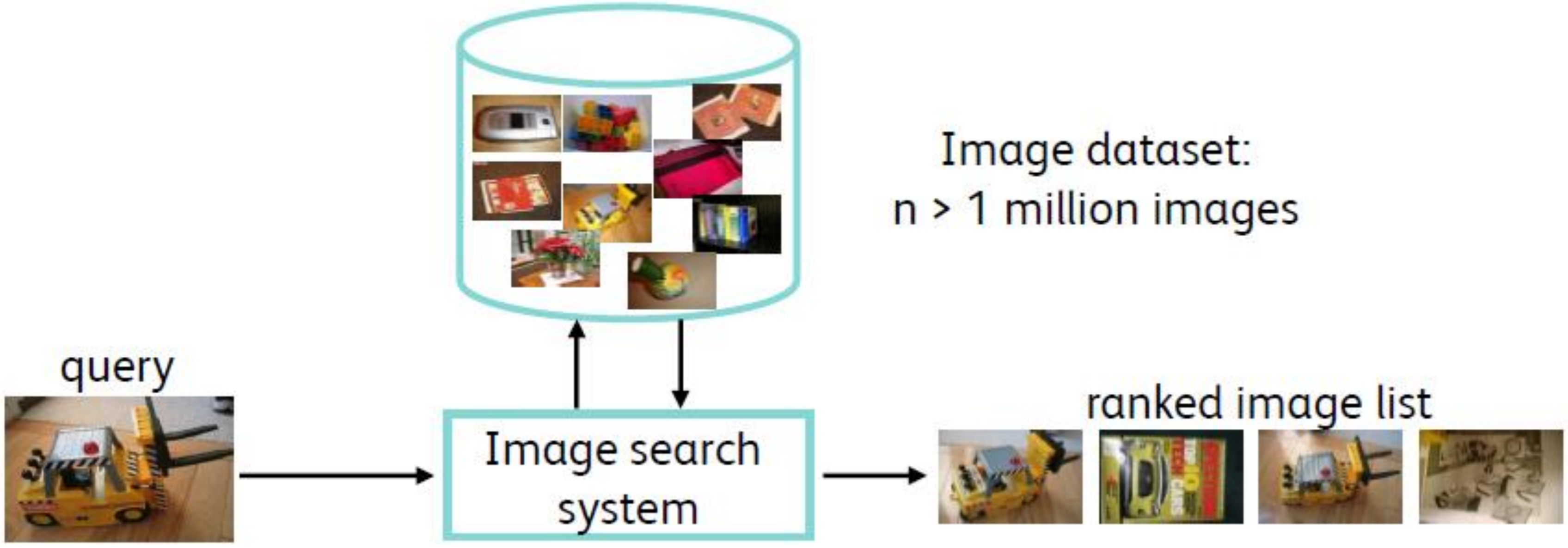
A **generic approach** could be:

1. Local feature extraction and description (ex. SIFT)
2. Matching local features based on similarity of local appearance
   - For every keypoint in one image find the closest keypoint (in the feature space) in the other image
   - Verify matches based on semi-local/global geometric relations

but…

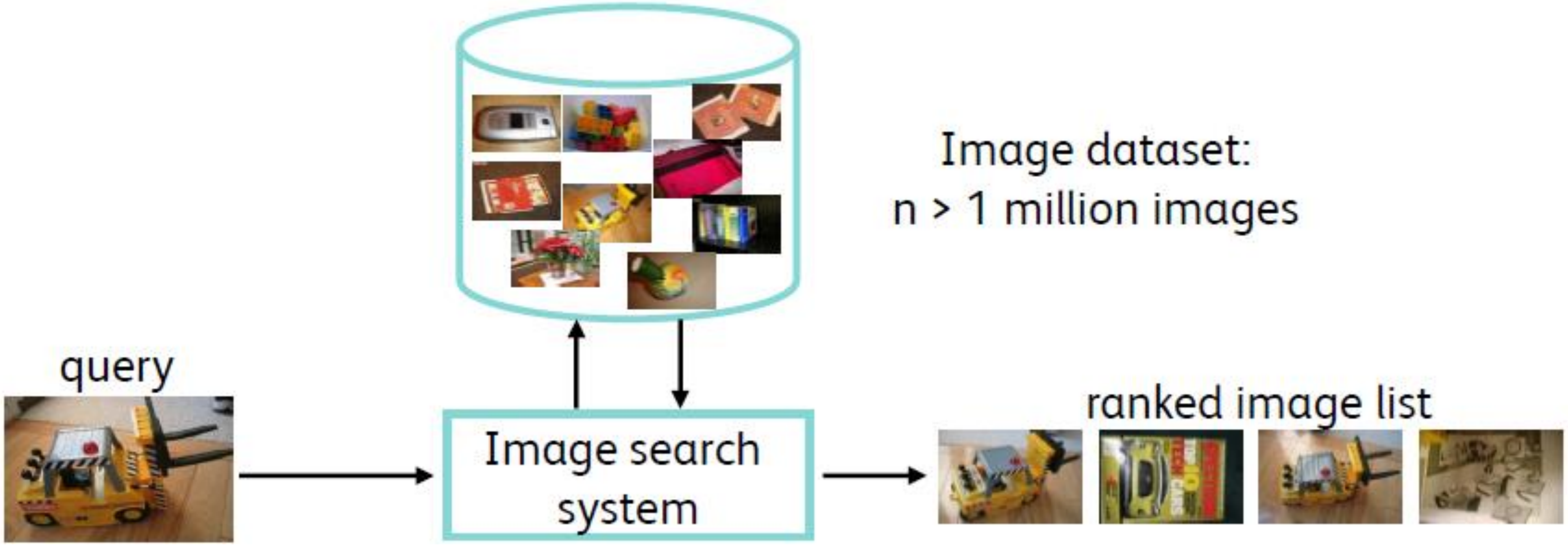
- Difficult to scale with the number of classes
- Computationally expensive
- Not well suited for applying machine learning

Let's do some numbers...



Slide credit: F. Perronin LSVR Tutorial CVPR13

Let's do some numbers...



- An image is described by m=1000 SIFT descriptors (d=128)
  - n*m= 1 billion descriptors to index
- Database representation: 128 GB RAM
- Search m^2 x n x d elementary operations!

Slide credit: F. Perronin LSVR Tutorial CVPR13

Motivation

Local features are **well suited** for **image categorization**

A **generic approach** could be:

1. Local feature extraction and description (ex. SIFT)
2. Matching local features based on similarity of local appearance
   - For every keypoint in one image find the closest keypoint (in the feature space) in the other image
   - Verify matches based on semi-local/global geometric relations

but…

- Difficult to scale with the number of classes
- Computationally expensive
- Not well suited for applying machine learning

… therefore, **better** if we can obtain a **global image representation** from the set of local features

Any thoughts??

Can we use such local features for image categorization?
So that:

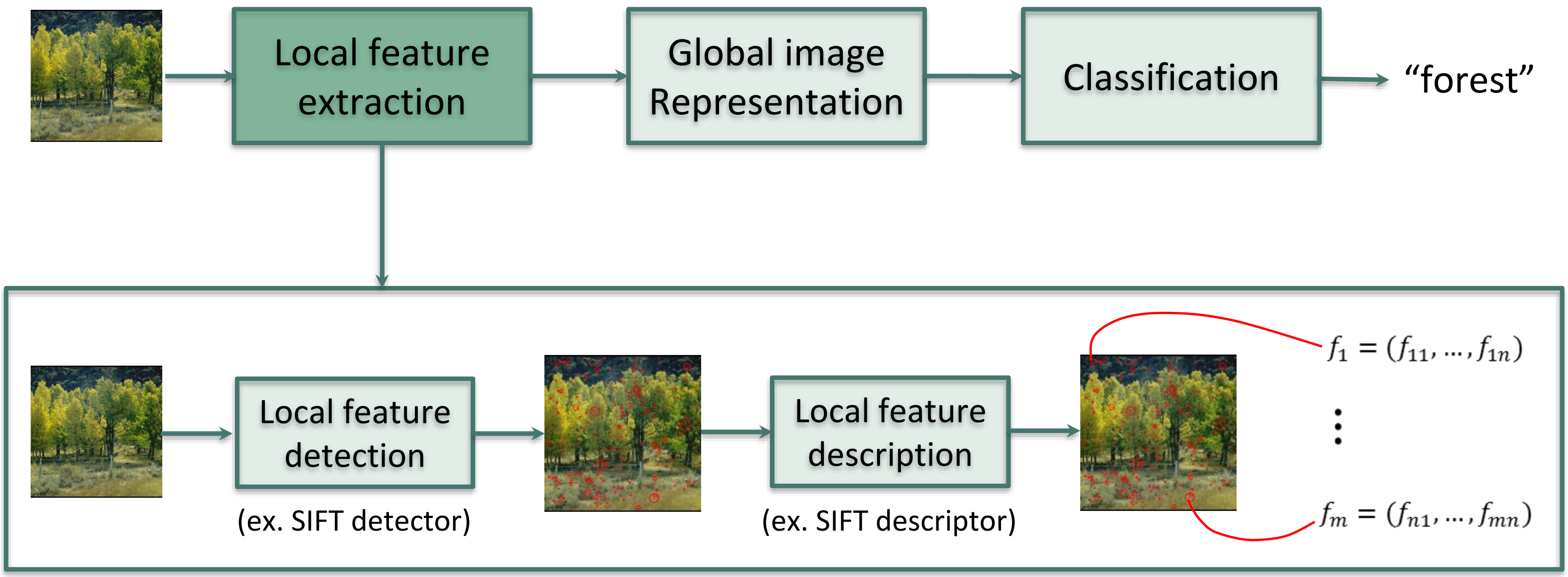
- keep discriminative power,
- we can scale,
- can apply statistical classifiers,
- ...

How can we go from a set of local descriptors to a single fixed-length global representation for each image?

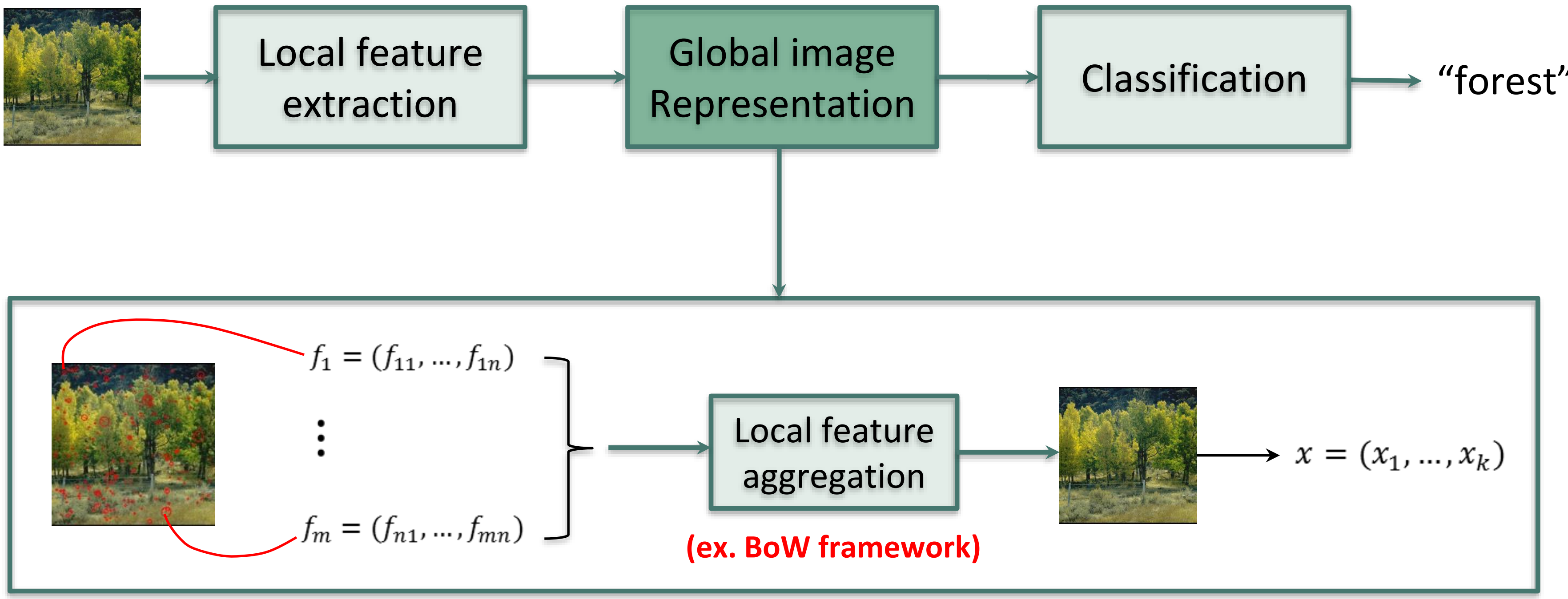
Pipeline for Image Categorization

Local feature extraction → Global image Representation → Classification → “forest”

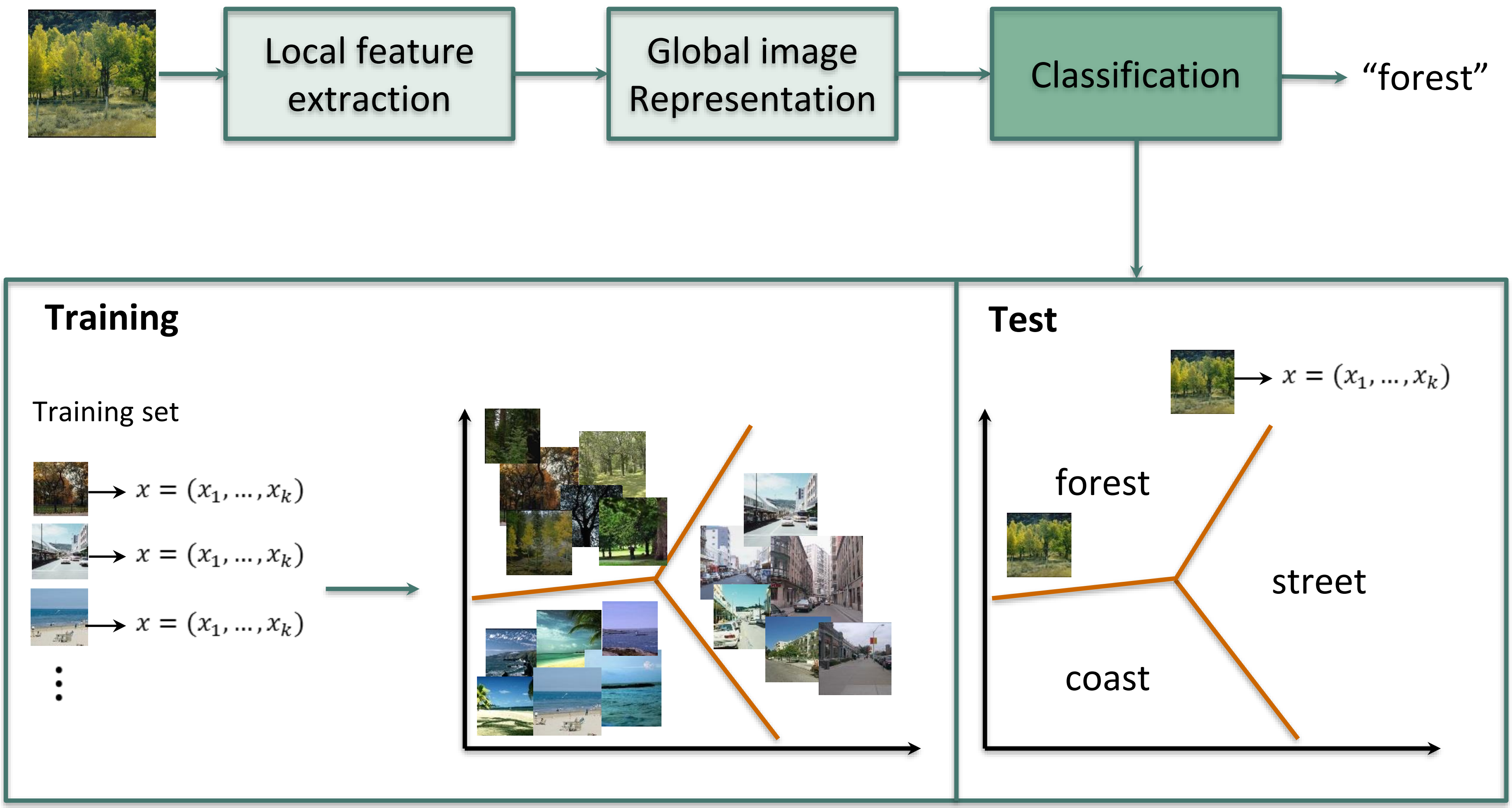
Pipeline for Image Categorization



Pipeline for Image Categorization

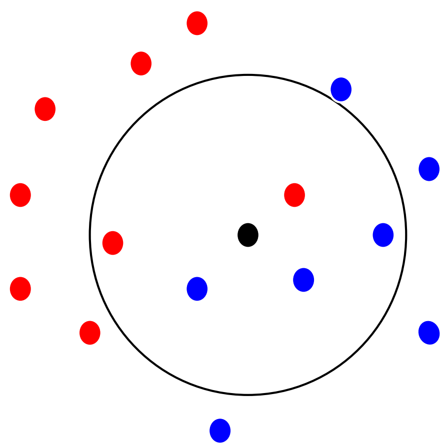


Pipeline for Image Categorization



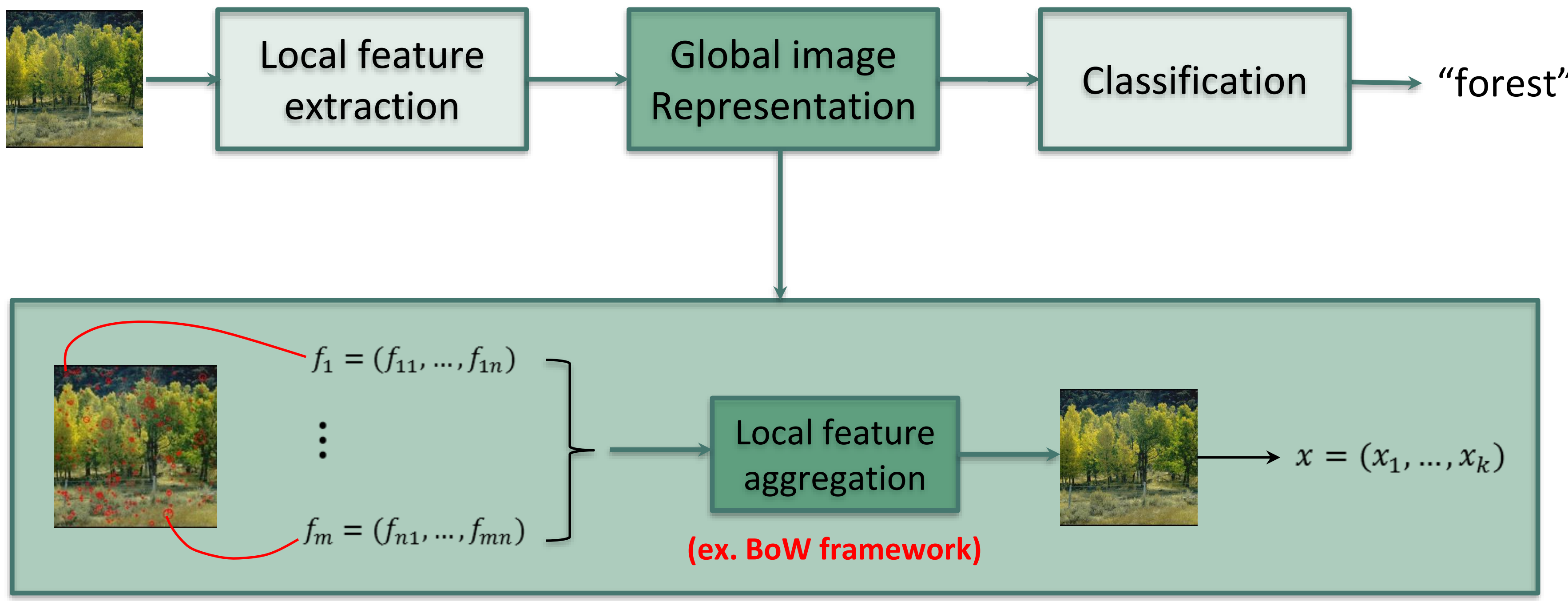
Pipeline for Image Categorization

Local feature extraction → Global image Representation → **Classification** → "forest"



k-NN classifier (for this first session), later on we will move to more powerful statistical classifiers (e.g. SVMs)

Image representation: The Bag of Words model



The Bag of Words model

Inspiration: document categorization

Tal y como los humanos usamos nuestros ojos y cerebros para comprender el mundo que nos rodea, la visión por computador trata de producir el mismo efecto para que las computadoras puedan percibir y comprender una imagen o secuencia de imágenes y actuar según convenga en una determinada situación. La adquisición de los datos se consigue por varios medios como secuencias de imágenes, vistas desde varias cámaras de video o datos multidimensionales desde un escáner médico.

El sentido de la vista o visión está asegurado por un órgano receptor, el ojo; una membrana, la retina, estos reciben las impresiones luminosas y las transmite al cerebro por las vías ópticas. El ojo es un órgano par situado en la cavidad orbitaria. Está protegido por los párpados y por la secreción de la glándula lagrimal. Es movilizado por un grupo de músculos extrínsecos comandados por los nervios motores del ojo.

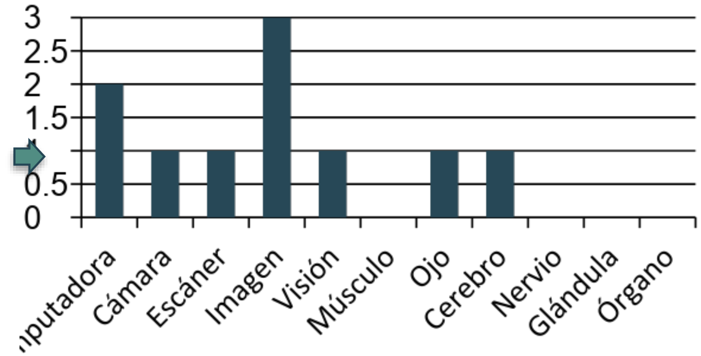
- Biology
- Computing

The Bag-of-Words model

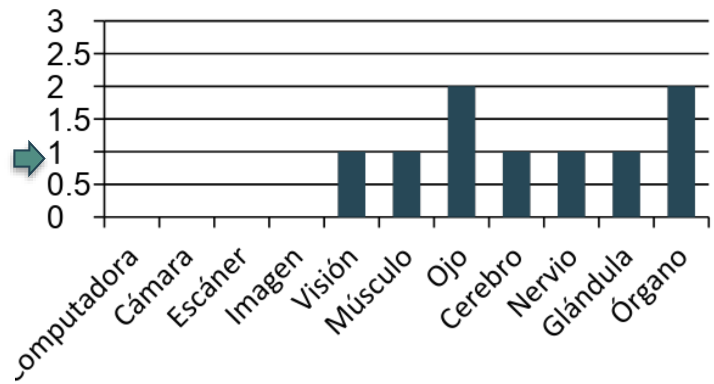
Inspiration: document categorization

Tal y como los humanos usamos nuestros ojos y cerebros para comprender el mundo que nos rodea, la visión por computador trata de producir el mismo efecto para que las computadoras puedan percibir y comprender una imagen o secuencia de imágenes y actuar según convenga en una determinada situación. La adquisición de los datos se consigue por varios medios como secuencias de imágenes, vistas desde varias cámaras de video o datos multidimensionales desde un escáner médico.

Histogram of representative words (Bag of Words)

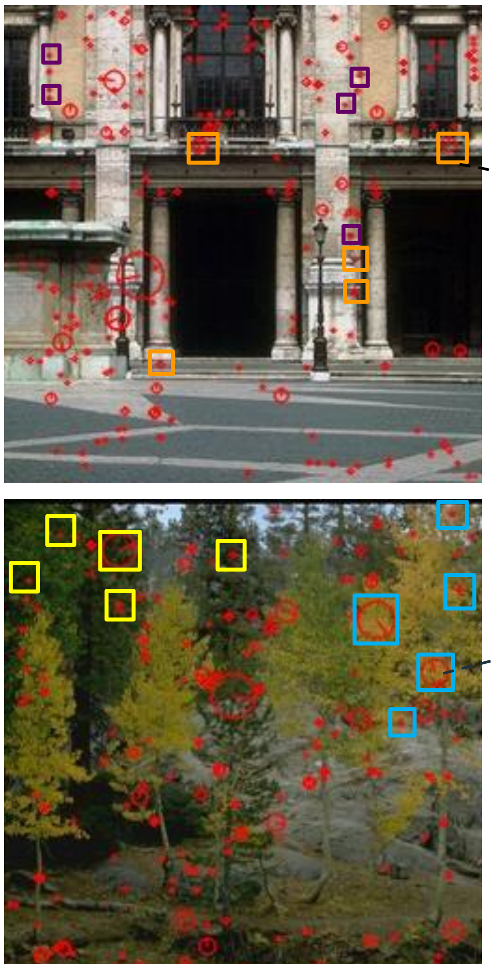


El sentido de la vista o visión está asegurado por un órgano receptor, el ojo; una membrana, la retina, estos reciben las impresiones luminosas y las transmite al cerebro por las vías ópticas. El ojo es un órgano par situado en la cavidad orbitaria. Está protegido por los párpados y por la secreción de la glándula lagrimal. Es movilizado por un grupo de músculos extrínsecos comandados por los nervios motores del ojo.



The Bag of Words model

Adapting the model to visual recognition: *Bag of Visual Words*

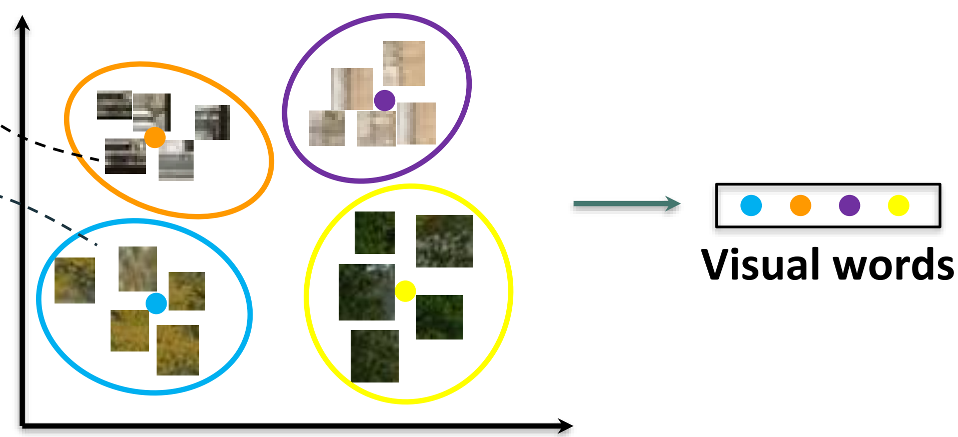


Training data

- We do not have a predefined set of relevant visual features
- We must Identify relevant common visual features: *visual words*

Vocabulary learning

Unsupervised learning: *clustering*



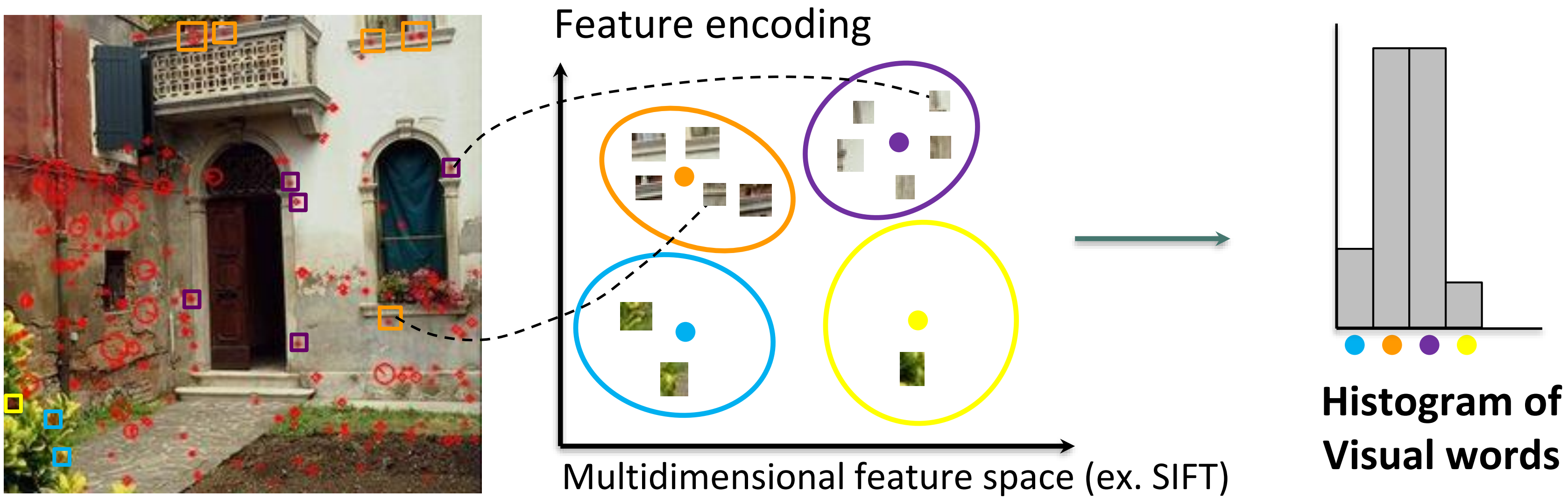
Multidimensional feature space (ex. SIFT)

The Bag of Words model

Adapting the model to visual recognition: *Bag of Visual Words*

- Every local feature in the image can be assigned to one visual word
- Image representation: histogram of *visual words*

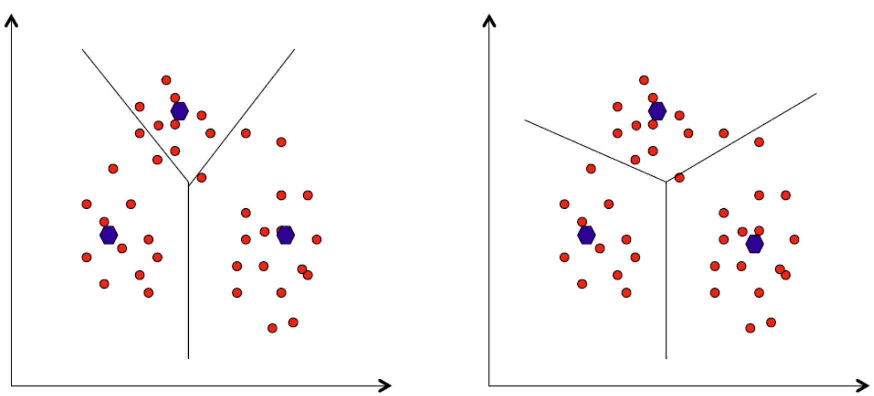
Image representation



Vocabulary learning

k-means algorithm (M 1 – Lecture 10)

- K-means algorithm: example (II)

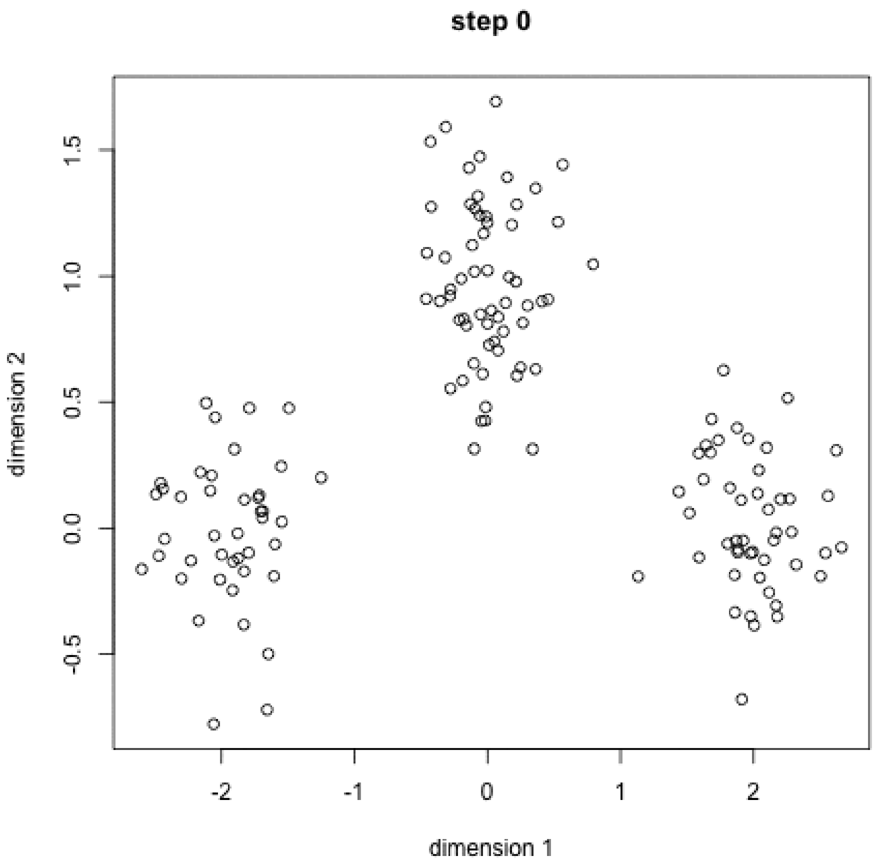


1. Initialize K classes. Compute the centers of each class
2. For each point:
   a. Compute the distances between the point and the class centers
   b. Assign the point to the closest class
3. Update the class centers
4. Repeat 2 & 3 until no change (in assignments or center values) is observed.

Max Lloyd algorithm

Vocabulary learning

k-means algorithm (M 1 – Lecture 10)



1. Initialize K classes. Compute the centers of each class
2. For each point:
   a. Compute the distances between the point and the class centers
   b. Assign the point to the closest class
3. Update the class centers
4. Repeat 2 & 3 until no change (in assignments or center values) is observed.

Max Lloyd algorithm

BoVW recap

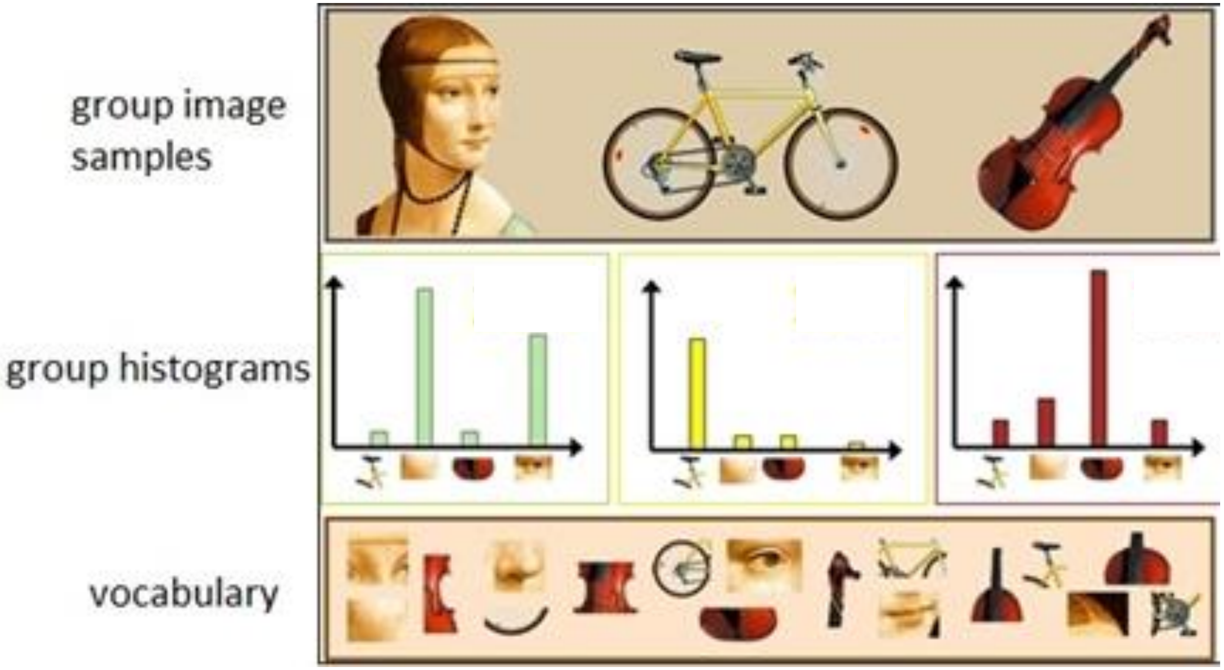
group image
samples

BoVW recap

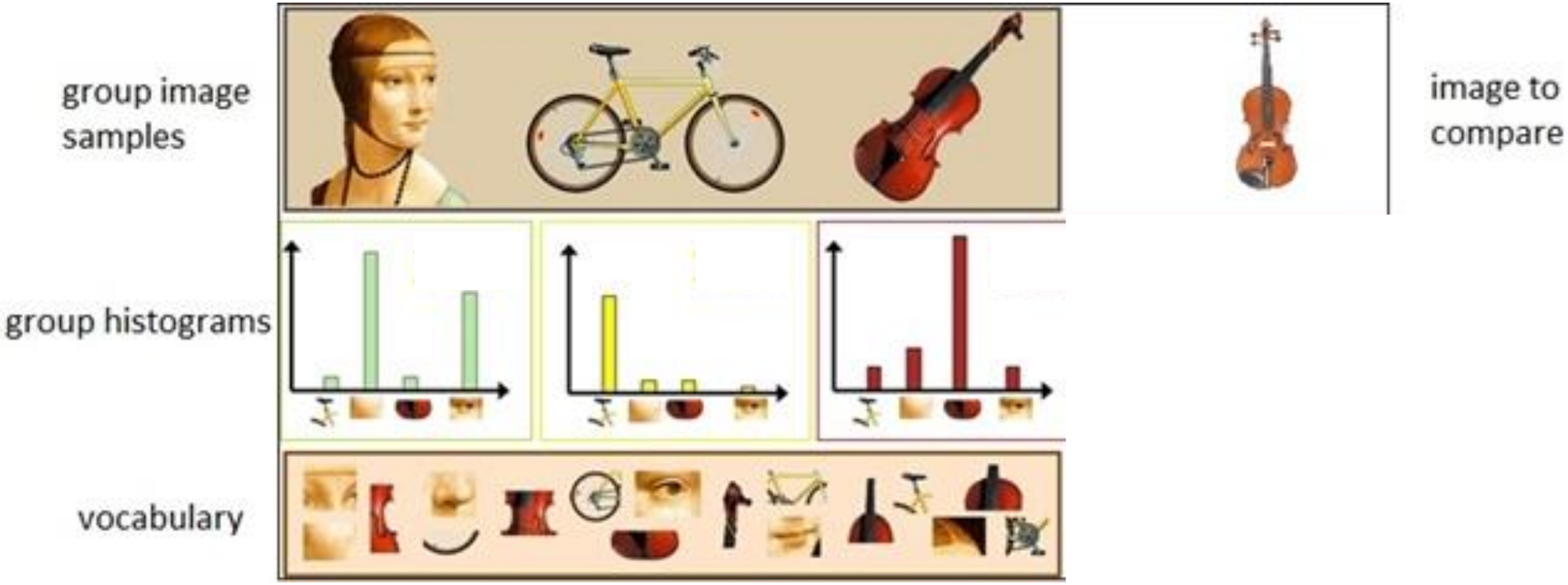
group image samples

vocabulary

BoVW recap

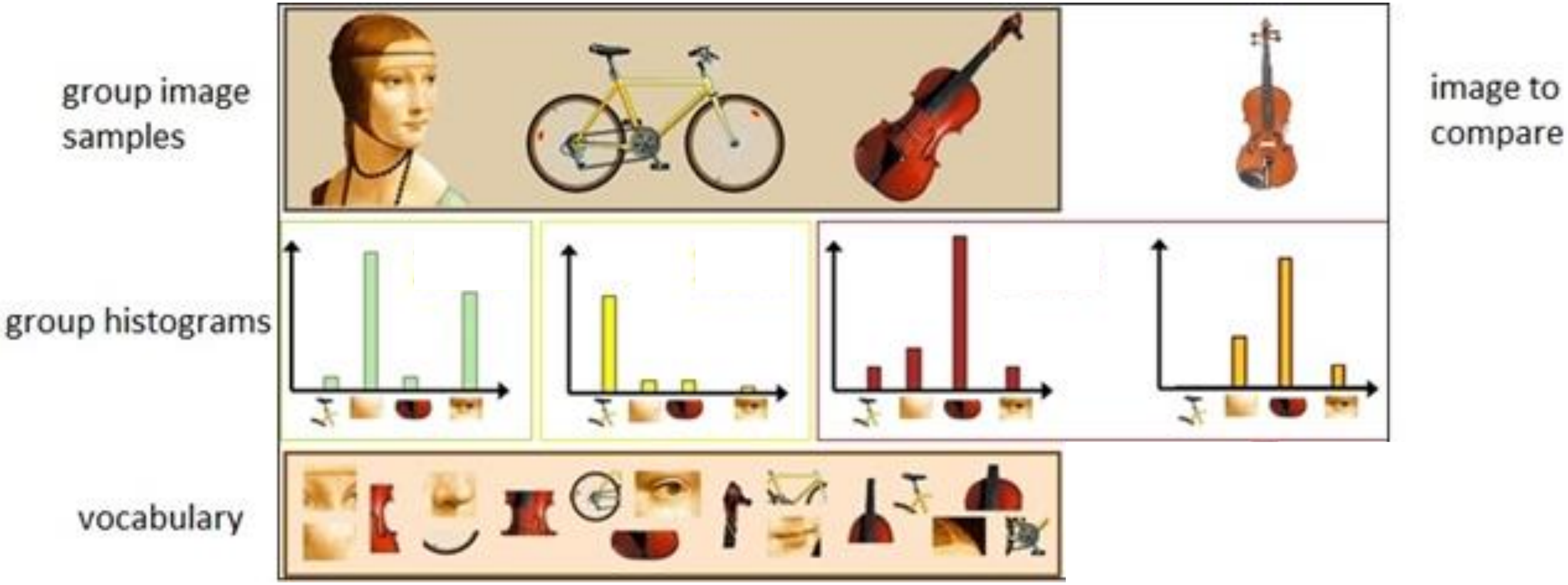


BoVW recap



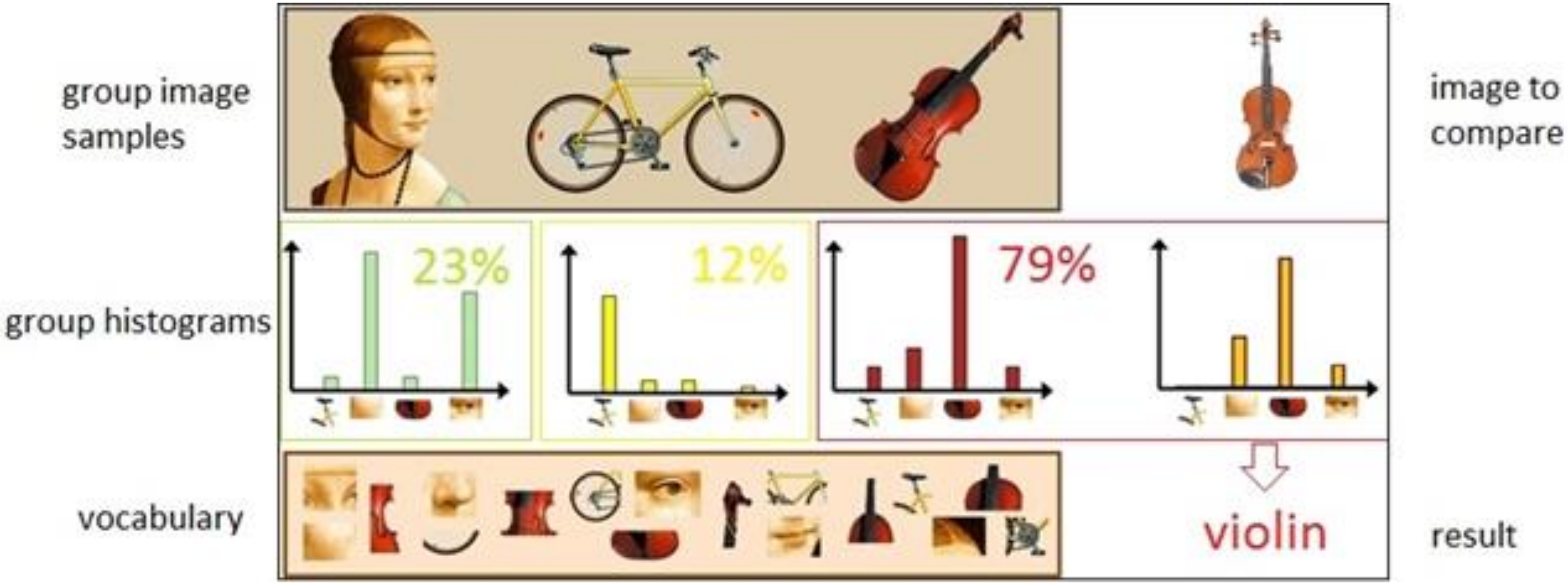
group image samples
image to compare
group histograms
vocabulary

BoVW recap



group image samples
image to compare
group histograms
vocabulary

BoVW recap


group image samples
image to compare
group histograms
23%
12%
79%
vocabulary
violin
result

WAKE UP!! Surprise test!!

- Would it be a good idea to use the BoVW framework as explained using the ORB local descriptors? Why?

WAKE UP!! Surprise test!!

- Would it be a good idea to use the BoVW framework as explained using the ORB local descriptors? Why?

- What are the effects of choosing a too low or too high value for *k*?

WAKE UP!! Surprise test!!

- Would it be a good idea to use the BoVW framework as explained using the ORB local descriptors? Why?

- What are the effects of choosing a too low or too high value for *k*?

- Is there any way of performing well (i.e. have a good description) in images where the keypoint detector step tend to perform poorly (e.g. low textures, or repetitive patterns)?

WAKE UP!! Surprise test!!

- Would it be a good idea to use the BoVW framework as explained using the ORB local descriptors? Why?

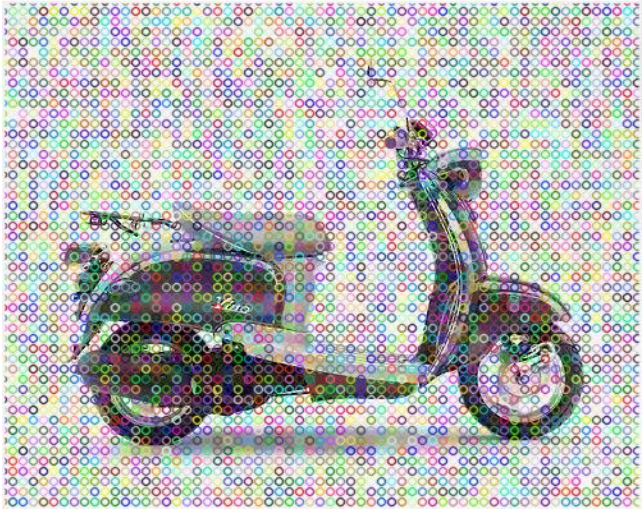
- What are the effects of choosing a too low or too high value for *k*?

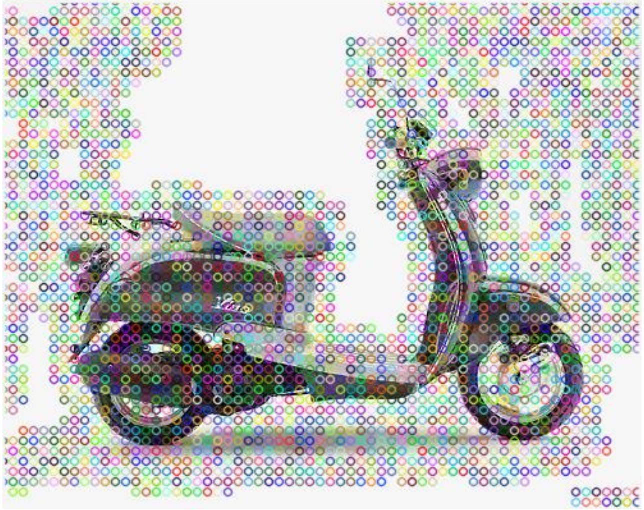
- Is there any way of performing well (i.e. have a good description) in images where the keypoint detector step tend to perform poorly (e.g. low textures, or repetitive patterns)?

- What kind of information is lost when using the BoVW framework compared to local keypoint matching? is there any way of include it in a coarse manner?

Beyond BoVW: Dense SIFT

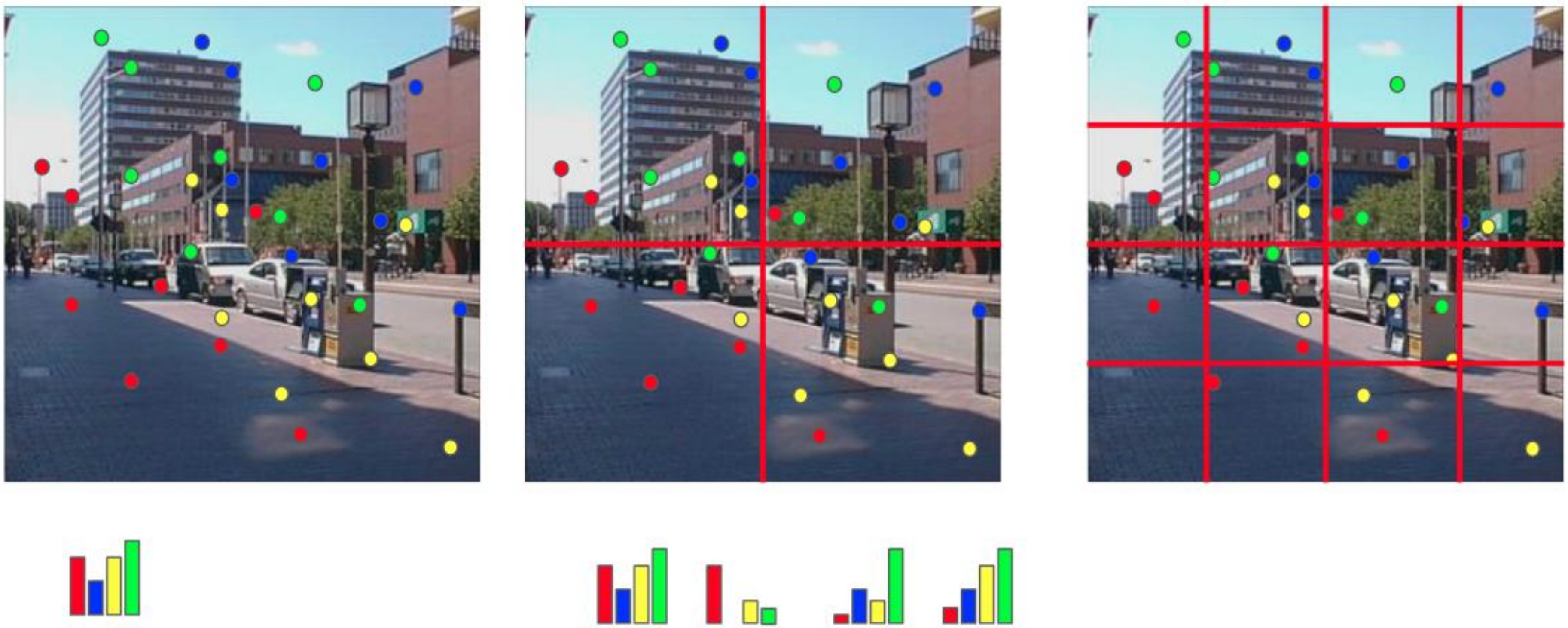
- Is there any way of performing well (i.e. have a good description) in images where the keypoint detector step tend to perform poorly (e.g. low textures, or repetitive patterns)?





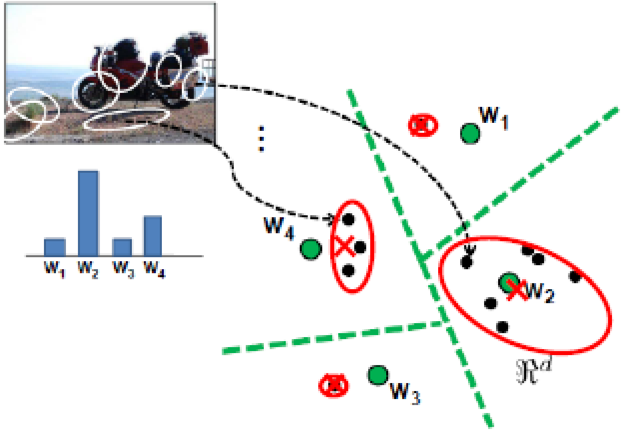
Beyond BoVW: Spatial Pyramids

- What kind of information is lost when using the BoVW framework compared to local keypoint matching? is there any way of include it in a coarse manner?



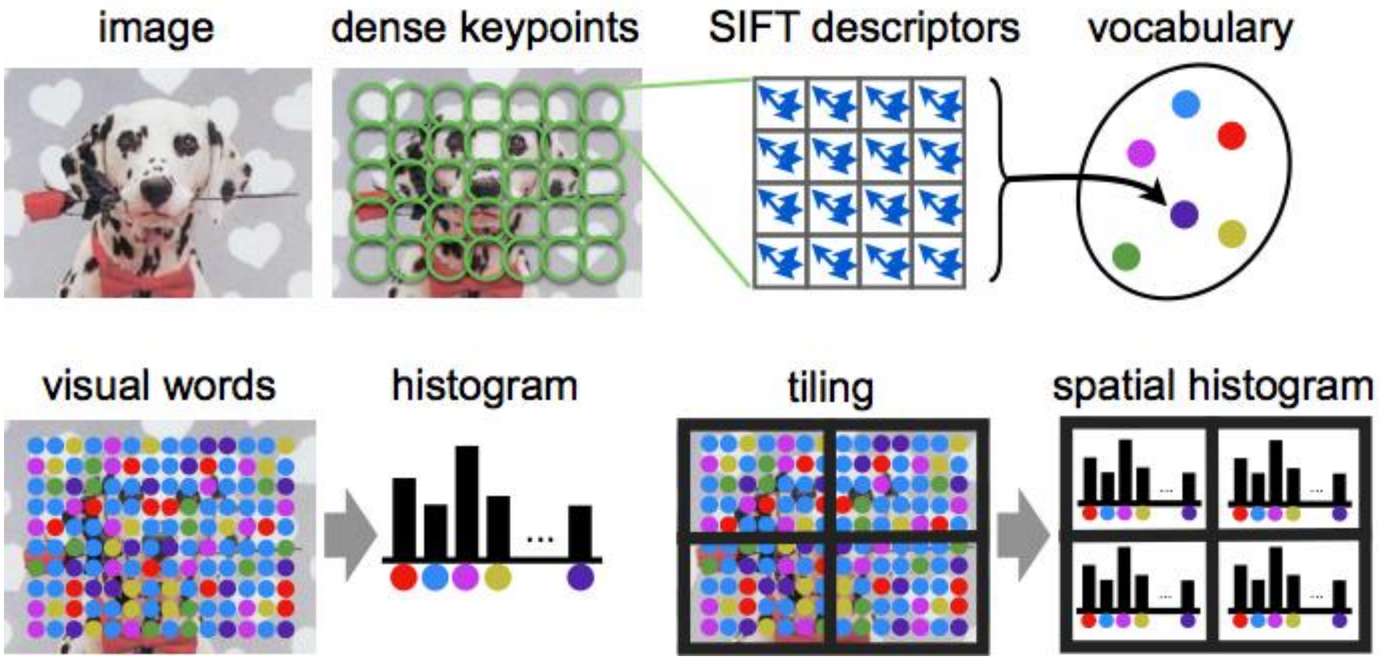
Beyond BoVW: Fisher Vectors

- BoVW is only counting the number of local descriptors assigned to each Voronoi cell
- Why not including higher order statistics?
  - Mean of local descriptors
  - Co-variance of local descriptors
- FV is typically 2 x D x k dimensional

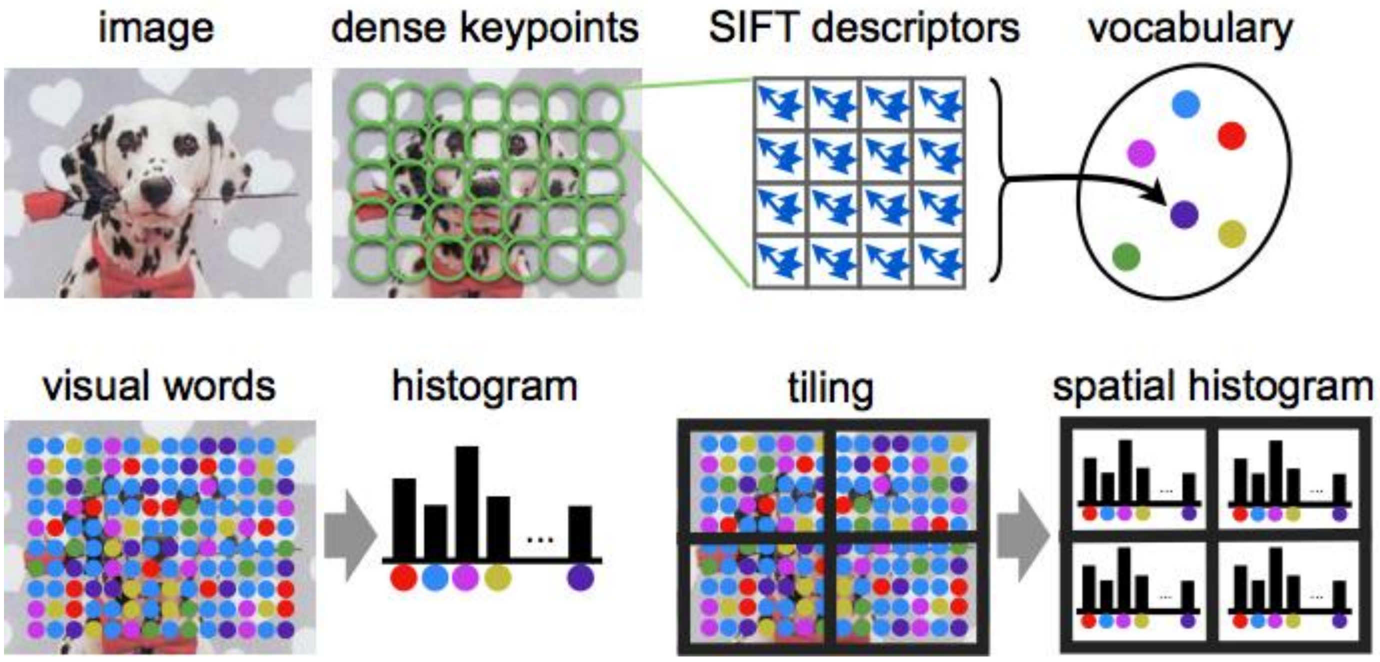


Slide creditF. Perroninn. Features for Large-Scale Visual Recognition

Everything together

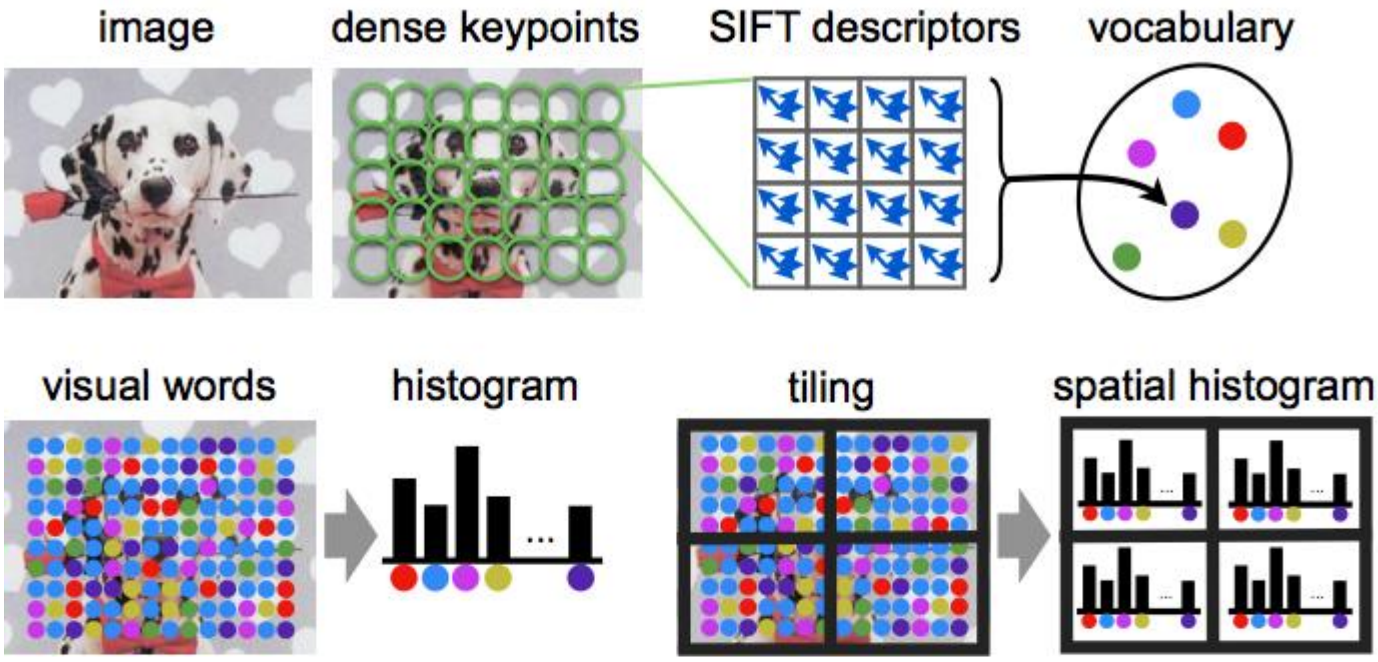


Everything together



- Image 512 x 512
- Dense SIFT extracted every 2 pixels, at 4 different scales
  - 256 x 256 x 4 = 262.144 descriptors per image…
- If we have 1M images in the dataset, how to compute the vocabulary?

Everything together



- k = 2048
- 1 level Spatial Pyramid
- BoVW dimension: 2048 x 5 = 10.240
- FV dimension: 2 x 128 x 2048 x 5 = 2.621.440
  2 x 128 x 32 x 5 = 40.960 (reducing k)
  2 x 64 x 32 x 5 = 20.480 (reducing SIFT dims)

Dimensionality Reduction

Goal: represent samples with fewer features

- Try to preserve as much **structure** in the data as possible

Feature selection

- Pick a subset of the original dimensions
- E.g. using information gain to decide which features to pick
- You are throwing out some of the features

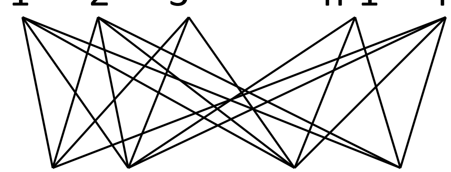
$x_1, x_2, x_3, \ldots, x_{n-1}, x_n$

$x_1, \mathbf{x_2}, x_3, \ldots, x_{n-1}, \mathbf{x_n}$

Feature extraction

- Construct a new set of *k* features (with *k* < *n*) combining existing ones
- The i^{th} feature given by: $z_i = f(x_1, x_2, \ldots, x_n)$
- The easiest way is by linearly combining the original features

$x_1, x_2, x_3, \ldots, x_{n-1}, x_n$

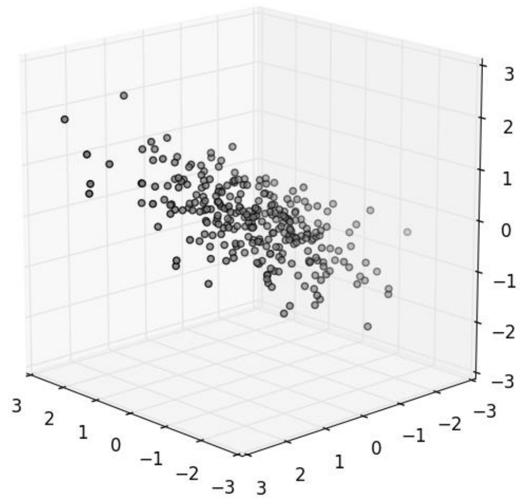


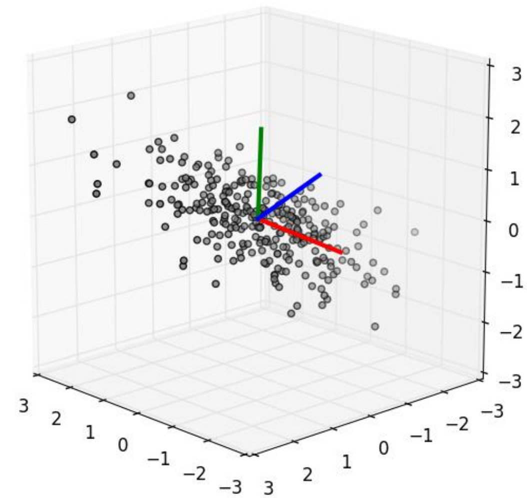
$z_1, z_2, \ldots, z_{k-1}, z_k$

Principal Component Analysis

PCA defines a set of principal components (a new set of dimensions, a new set of features)

- 1st dimension: direction of the greatest variability in the data
- 2nd dimension: perpendicular to the 1st, greatest variability of what's left to explain
- 3rd dimension: perpendicular to all the previous ones, greatest variability of what's left to explain
- … and so on until n (the original dimensionality)





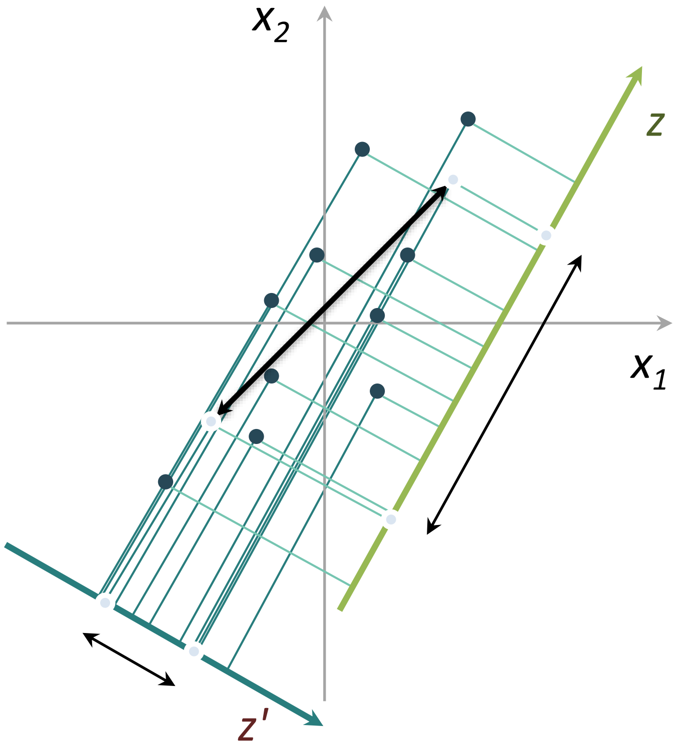
Why maximum variability

An example, reducing from $\mathbb{R}2$ to $\mathbb{R}1$

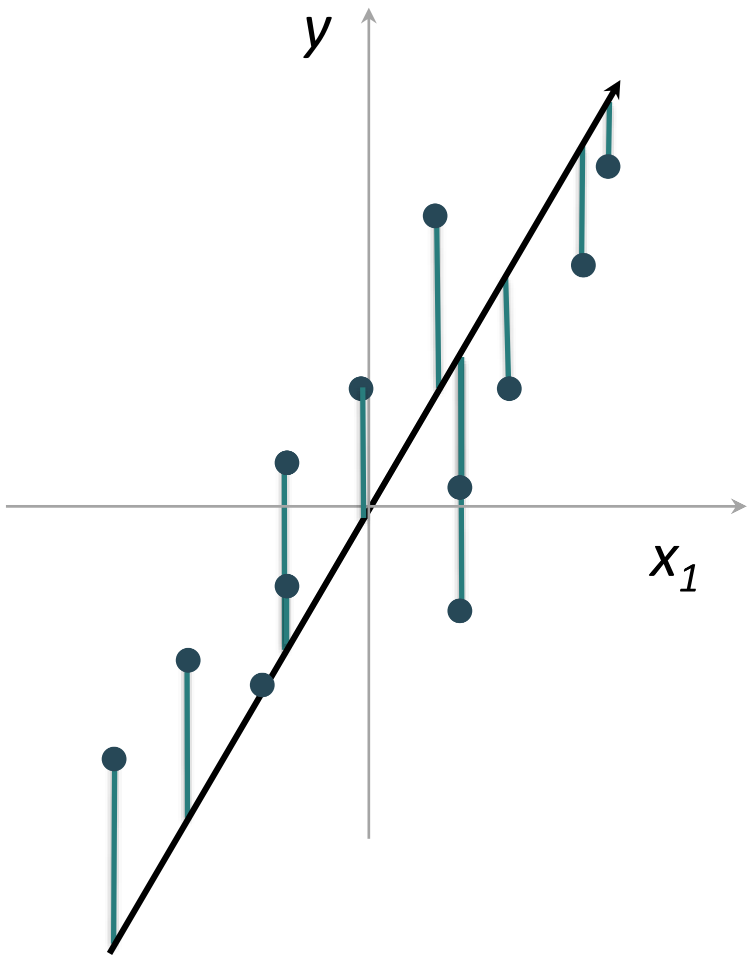
If we pick a direction *z* that maximises variability (green in the plot), it will maintain to a certain degree the relative distance between points in the original space:

If two points are far in the original (x_1, x_2)-space, they will most probably be far in the new (*z*)-space

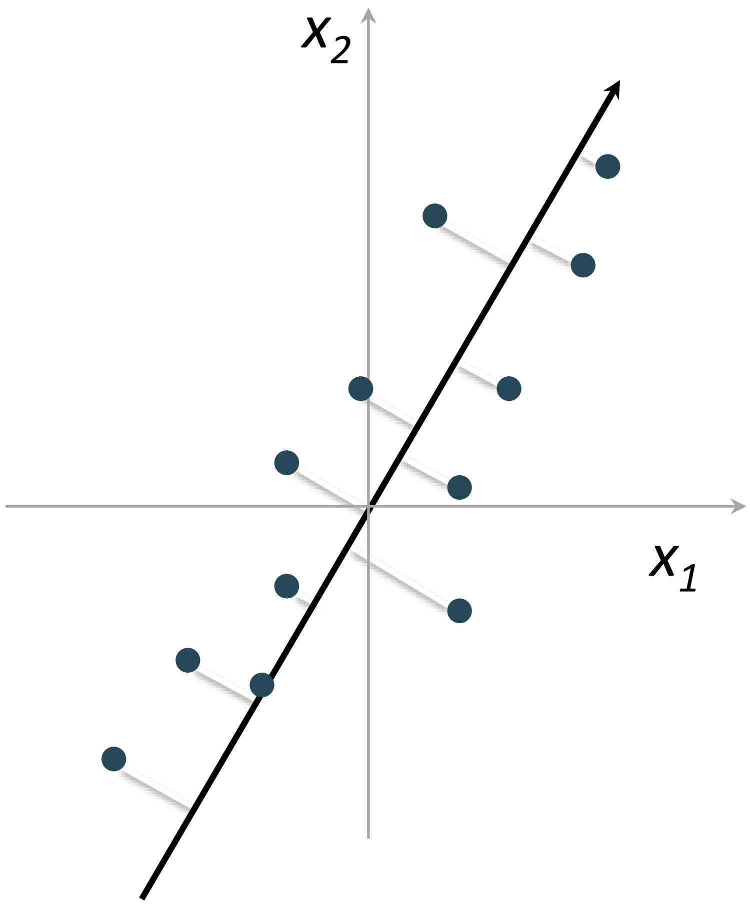
If we pick any other direction z' the relative distance between points in the original space is not preserved so well.



PCA is not linear regression



Linear regression



PCA

PCA algorithm summary

1. We start with correlated, high-dimensional data, $x \in \mathbb{R}^n$
2. Centre the points (optionally scale the features)
3. Computer the covariance matrix
4. Find the eigenvectors and the eigenvalues of the covariance matrix (e.g. using SVD)
5. Pick the $k \ll n$ eigenvectors with the highest eigenvalues
6. Project data points to the selected eigenvectors
7. Obtain uncorrelated low-dimensional data, $z \in \mathbb{R}^k$

PCA and classification

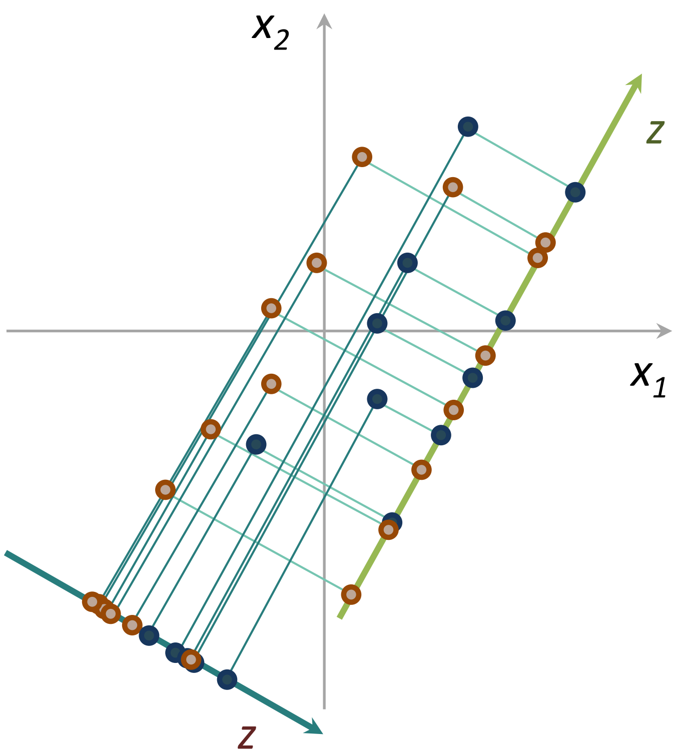
PCA can sometimes hurt instead of help, as it does not take into account the class labels

PCA is unsupervised

- Maximises overall variance along a small set of directions
- Does not know anything about class labels

Discriminative approach

- Look for a dimension that makes it easy to separate classes



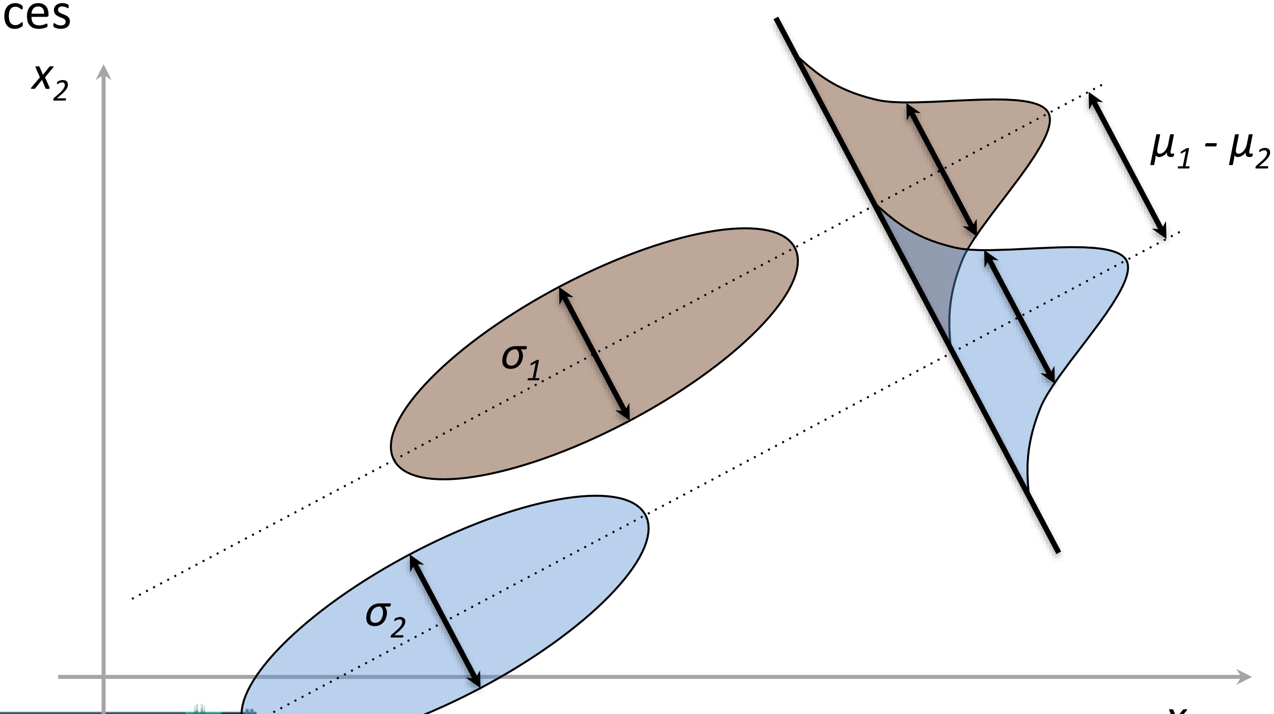
Linear Discriminant Analysis

LDA picks a new dimension that gives:

- Maximum separation between means of projected classes
- Minimum variance within each projected class

Solution: eigenvectors based on between-class and within-class covariance matrices

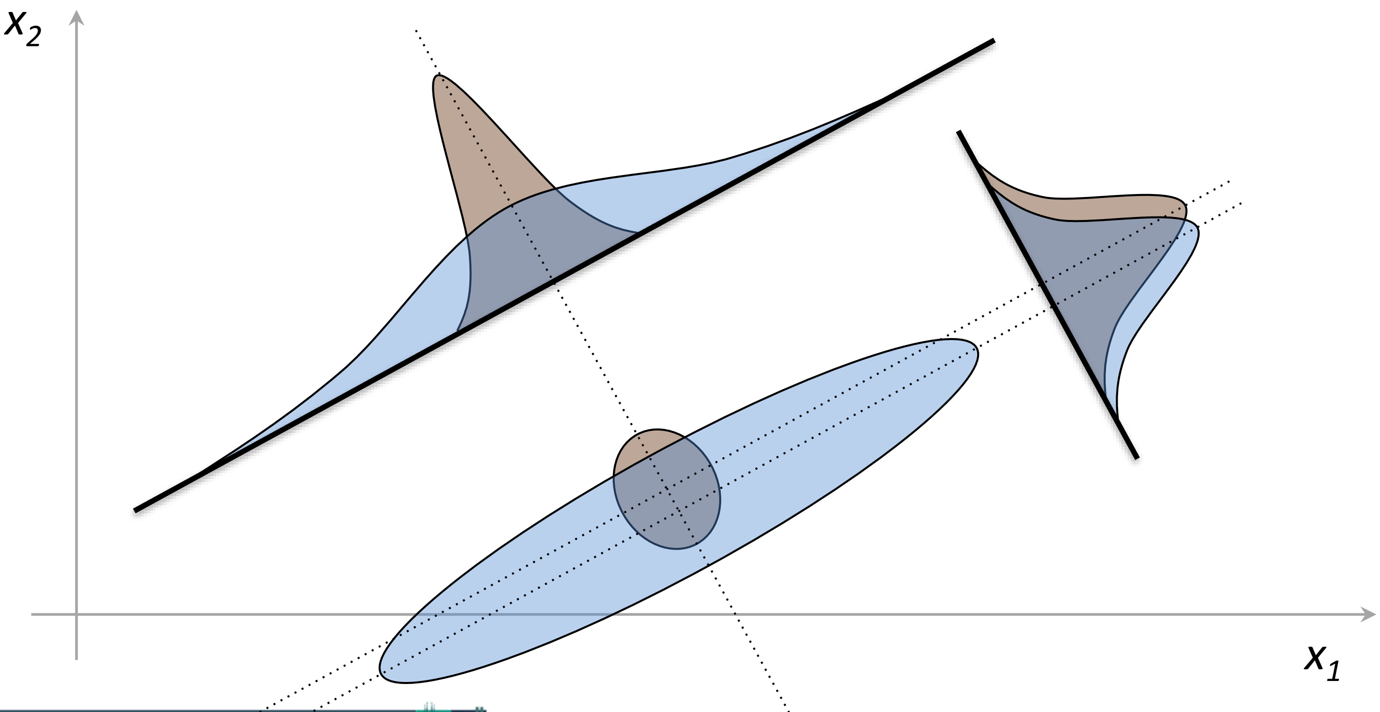
$$\max \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}$$



Linear Discriminant Analysis

LDA is not guaranteed to be better for classification

- Assumes that distributions are unimodal Gaussians
- Assumes that they are separable
- Fails when the discriminatory information is not in the mean but in the variance of the data



References

- D. Lowe. *Object Recognition from Local Scale-Invariant Features*. ICCV 1999
- D. Lowe. *Distinctive Image Features from Scale-Invariant Keypoints*. IJCV 2004
- E. Nowak, F. Jurie, B. Triggs. *Sampling Strategies for Bag-of-Features Image Classification*. ECCV'06
- J. Sivic and A. Zisserman. *Video Google: A Text Retrieval Approach to Object Matching in Videos*. ICCV 2003.
- F. Perronnin, C. Dance, G. Csurka, M. Bressan. *Adapted Vocabularies for Generic Visual Categorization*, ECCV 2006.
- J. Vogel, B.Schiele. *Semantic Modeling of Natural Scenes for Content-Based Image Retrieval*. International Journal of Computer Vision, 2006.
- L. Fei-Fei and P. Perona*. A Bayesian hierarchical model for learning natural scene categories.*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2005.
- G. Csurka, C. R. Dance, L. Fan, J. Willamowski, C. Bray. *Visual Categorization with Bags of Keypoints*. ECCV 2004

References

- J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. *Discovering object categories in image collections.* Technical Report A. I. Memo 2005-005, Massachusetts Institute of Technology, 2005.
- J. Savarese, A. Winn and T. Criminisi, *Object Categorization by Learned Universal Visual Dictionary.* CVPR 2006
- A. Vedaldi, A. Zisserman. *Efficient additive kernels via explicit feature maps.* PAMI 2011
- S. Lazebnik, C. Schmid, J. Ponce. *Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories*. CVPR 2006.
- K. Grauman, T. Darrell. *The Pyramid Match Kernel: Efficient Learning with Sets of Features*. Journal of Machine Learning Research, 2008
- N. Elfiky, F. Khan, J. Van de Weijer, J. González. *Discriminative Compact Pyramids for Object and Scene Categorization*. PR, 2011
- F. Perronin, J. Sánchez, T. Mensink. *Improving the Fisher Kernel for large-scale image classification*. ECCV 2010
- F. Perronin, C. Dance. *Fisher Kernels on Visual Vocabularies for Image Categorization.* CVPR 2007
- K. Chatfield, V. Lempitsky, A. Vedaldi, A. Zisserman. *The devil is in the details: an evaluation of recent feature encoding methods*. BMVC 2011

Toy Dataset

8 classes

Coast
244 train
116 test

Forest
227 train
101 test

Highway
184 train
76 test

Inside city
214 train
94 test

Mountain
260 train
114 test

Open country
292 train
118 test

Street
212 train
80 test

Tall building
248 train
108 test

To Do...

Improve the BoVW code with:

- Test different amounts of local features. What performs best?
- Use dense SIFT instead of detected keypoints. Conclusions?
- Test different amounts of codebook sizes *k.* What performs best?
- Test different values of *k* for the k-nn classifier. What performs best?
- Test other distances in k-nn classifier. Does that make a difference? Why?
- Play with reducing dimensionality. Conclusions?
- Cross-validate everything (topic covered on Wednesday)

Next session:

- SVM classifier.
- Linear, RBF and histogram intersection kernels.
- Spatial Pyramid.
- Fisher Vectors (OPTIONAL, check out yael library…)

Warning: provided code might not work out of the box depending on the used versions (OpenCV, numpy, sklearn…) do not panic, and ~~RTFM~~ read the documentation

Deliverable

- A **single Python notebook file per group** reporting all the work done,
  - with the different experiments,
  - code,
  - plots,
  - explanations, etc.
  - **EVERYTHING EXECUTED!**

- To deliver by Tuesday, january, 5th @ 10 A.M. by email (ramon.baldrich@uab.cat)

  Please, state clearly your group.