Projekt Computer Vision
Part 4: Introduction to Image Recognition

Vincent Christlein
11.12.2017
Pattern Recognition Lab (CS 5)

Main references:
http://web.stanford.edu/class/cs231a/
https://sites.google.com/site/lsvr13/home/part-i-features-for-large-scale-visual-recognition

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

What’s visual recognition?

A possible definition: recognizing and identifying the key semantic aspects of a scene from images

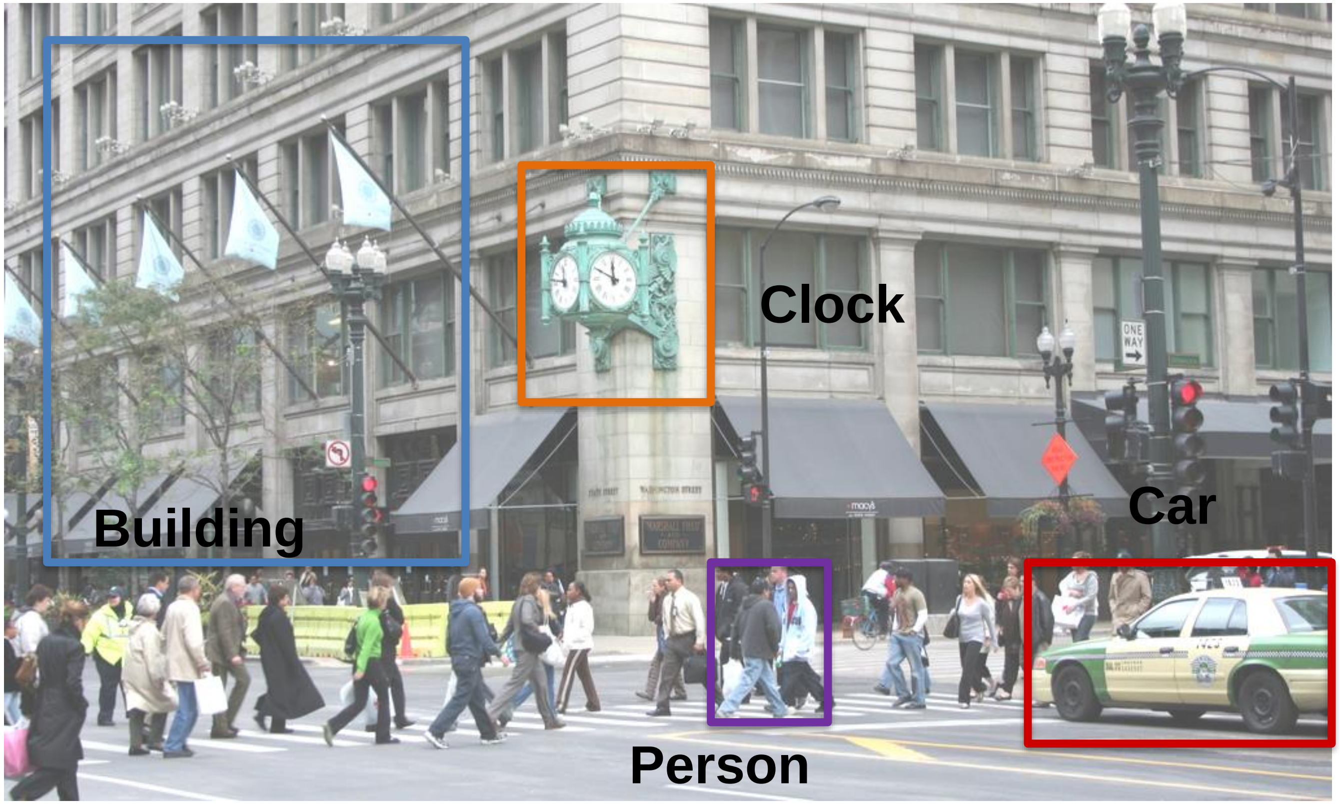
Classification

Does this image contain a building? [yes/no]

Detection

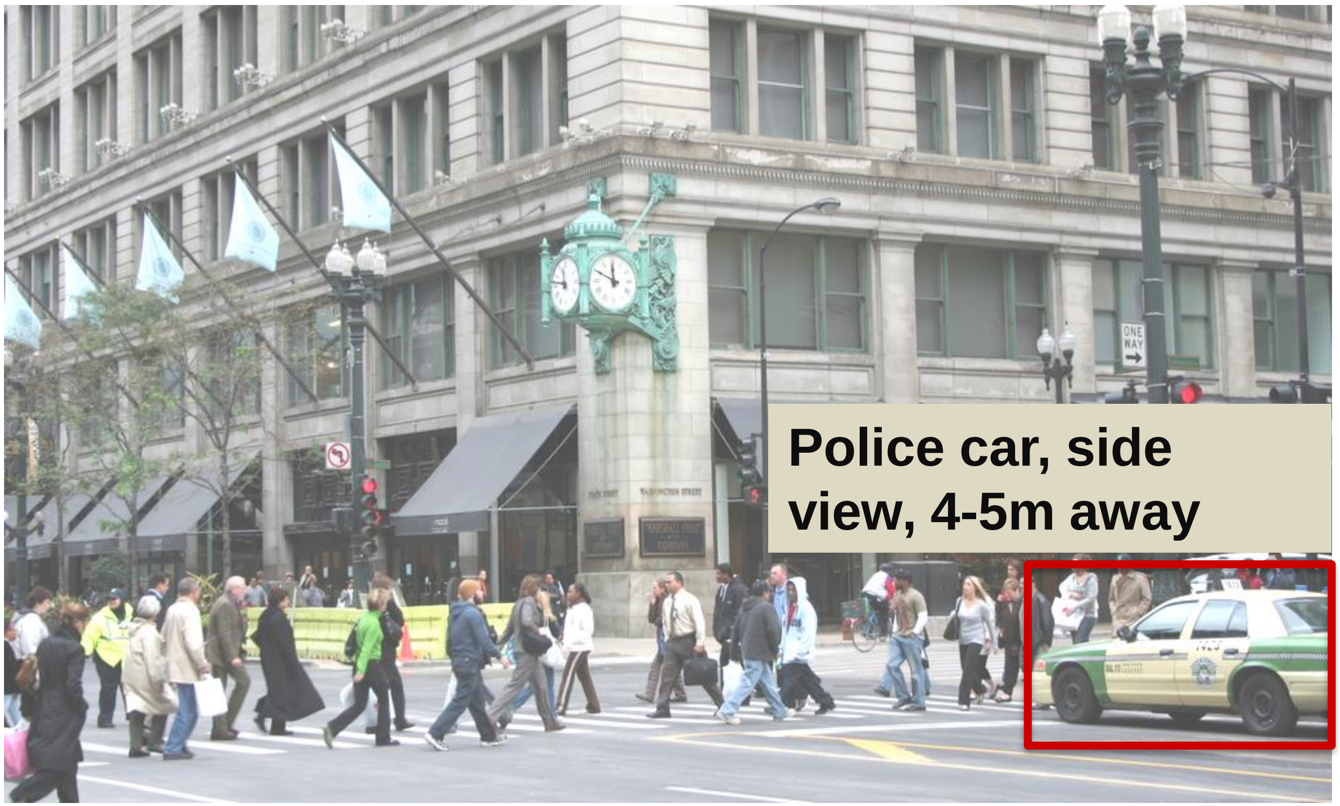
Does this image contain a car? [where?]

Detection



Which object does this image contain? [where?]

Detection



Estimating object semantic & geometric attributes

Categorization vs. single instance recognition

Which building is this?

Image search & image grouping

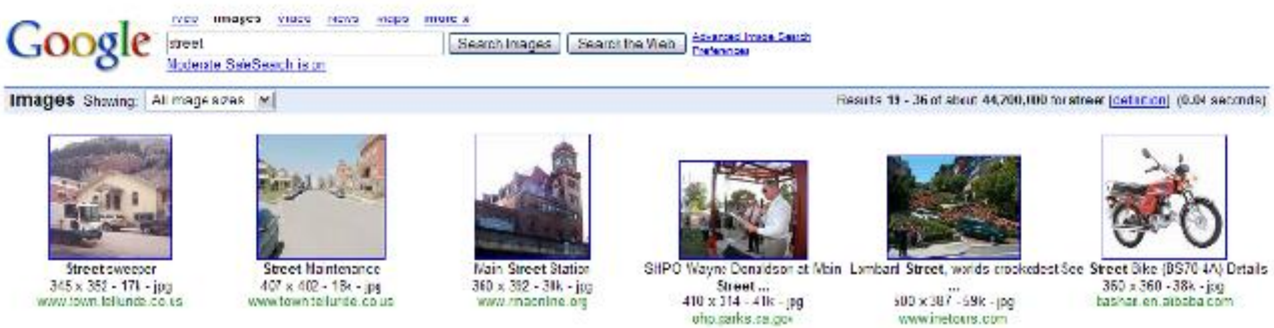


Image retrieval

Query Retrieval list

Visual recognition fields

Computational photography

Assistive technologies

Surveillance / Security

Assistive driving

Industrial machine vision

Other sciences

Visual Recognition

Design algorithms that are capable to

- Classify images or videos
- Detect and localize objects
- Estimate semantic and geometrical attributes
- Classify human activities and events

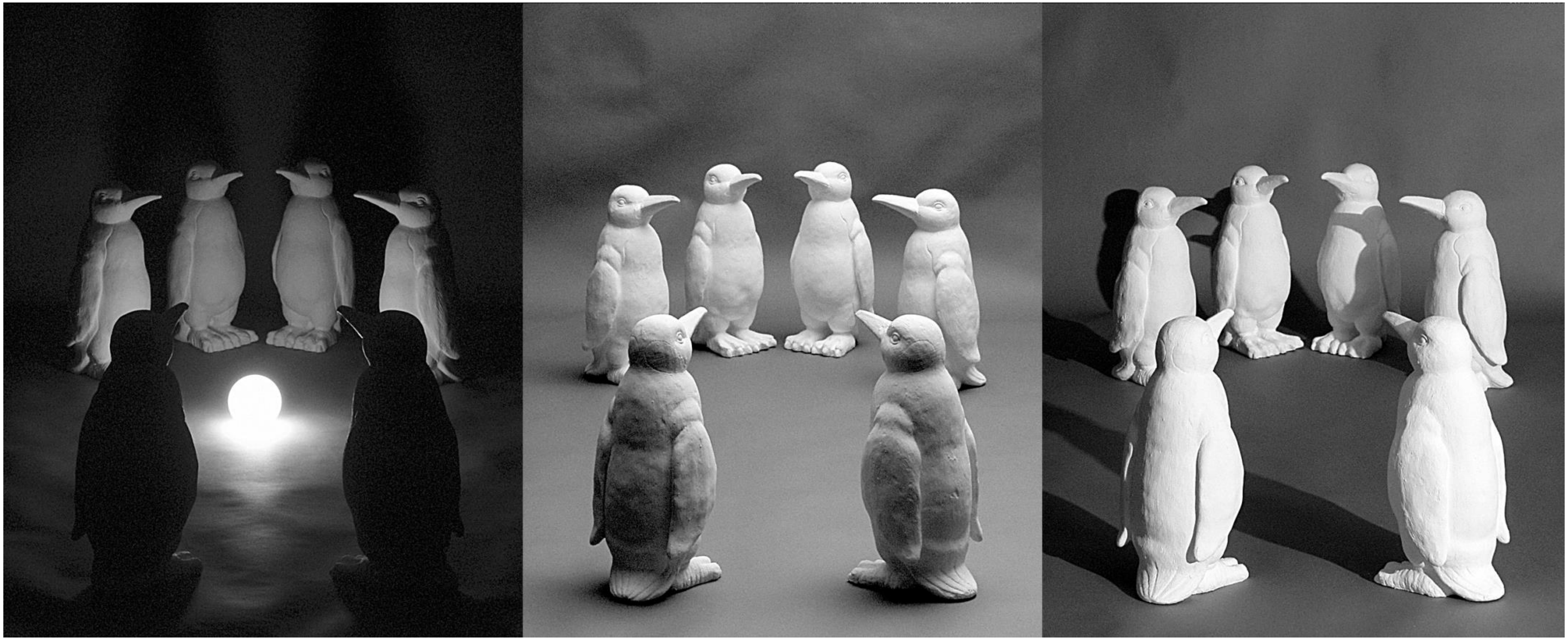
Why is this challenging?

10k-30k object categories if we only consider super-categories (e.g., a "car") → much larger if fine-grain categories are included (e.g. "SUV")

Challenges: viewpoint variation

Michelangelo 1475-1564

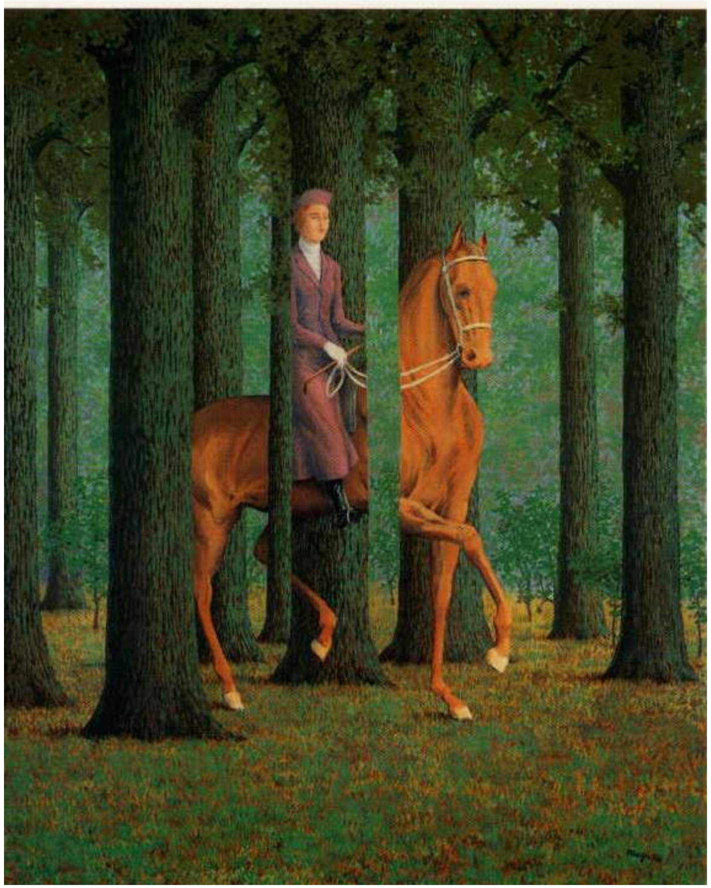
Challenges: illumination



Challenges: scale

Challenges: Deformation

Challenges: Occlusion and Background clutter



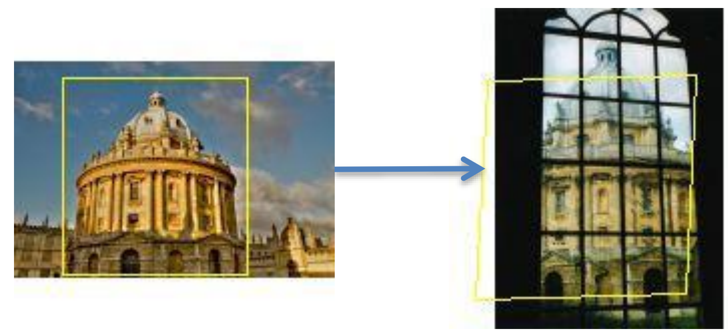
Magritte 1957

Kilmeny Niland. 1995

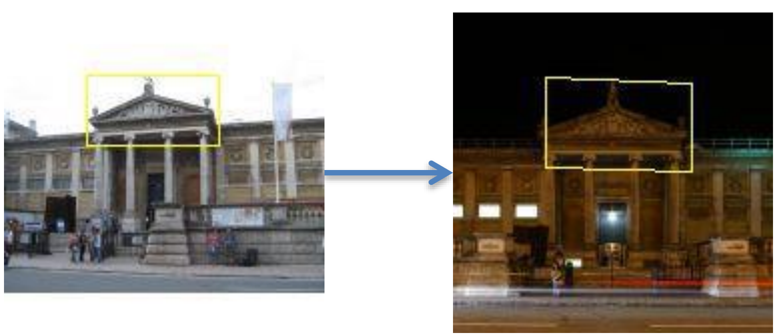
Challenges (realistic examples)

- Scale

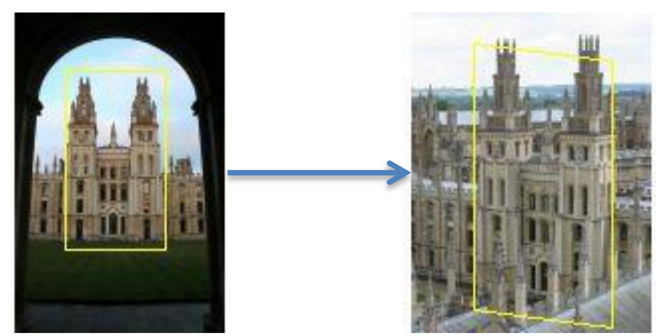
- Occlusion



- Lighting

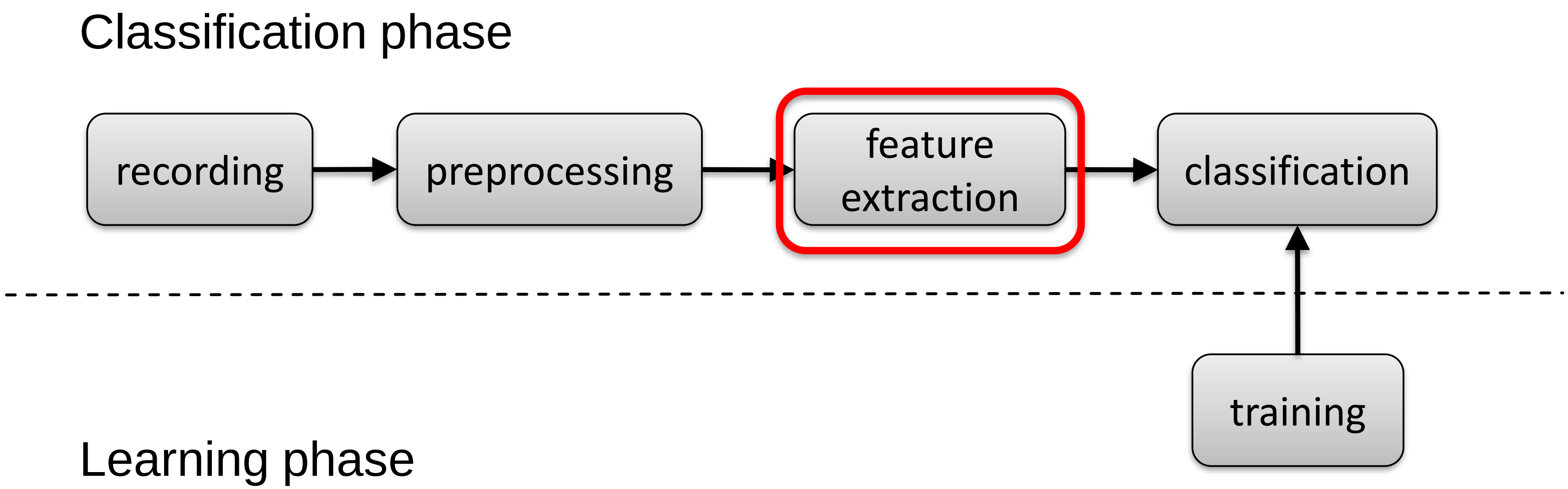


- Viewpoint



Challenges: intra-class variation

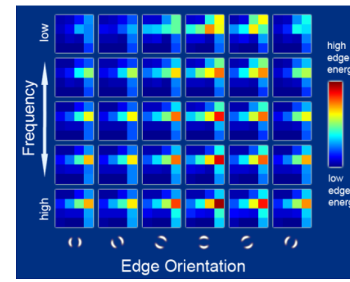
Machine Learning Pipeline



Global vs. Local Feature Descriptors

Global descriptors (of pixel statistics)

- Color Histogram: high invariance but limited discriminative power (Swain, Ballard, “Color indexing”, IJCV’91)
- GIST of a scene:
  - Oliva, Torralba, “Modeling the shape of the scene: a holistic representation of the spatial envelope”, IJCV’01.
  - Douze, Jegou, Sandhawalia, Amsaleg, Schmid, “Evaluation of GIST descriptors for web-scale image search”, CIVR’09
- CENTRIST: CENsus Transform hISTogram
  - Wu, Rehg, “CENTRIST: a visual descriptor for scene categorization”, TPAMI’11.

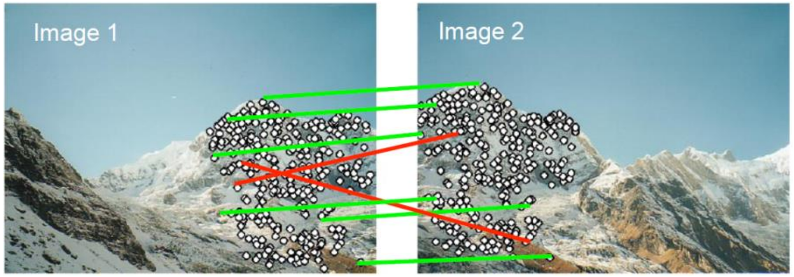


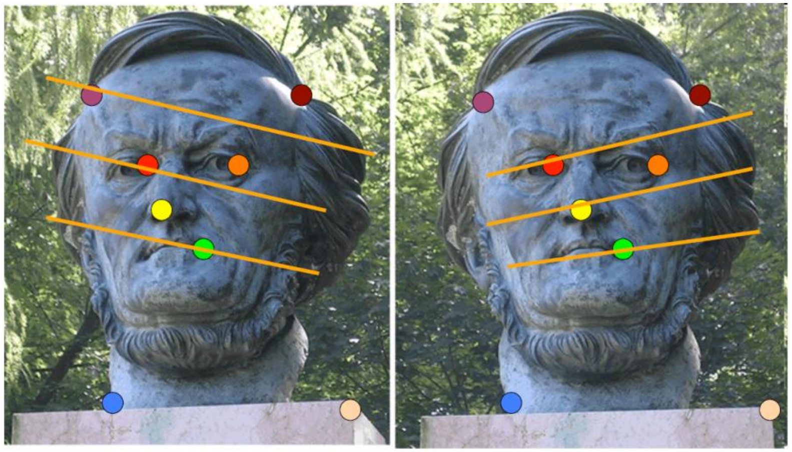
Highly efficient to compute + match → **perfect for large scale visual recognition (LSVR)**
But **robustness vs informativeness tradeoff is hard to set**

Local feature descriptors

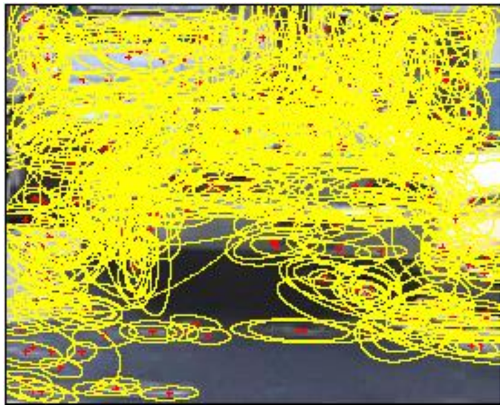
Motivation

- Image Stitching
- Calibration
- Stereo Vision
- Tracking
- …
- Image classification



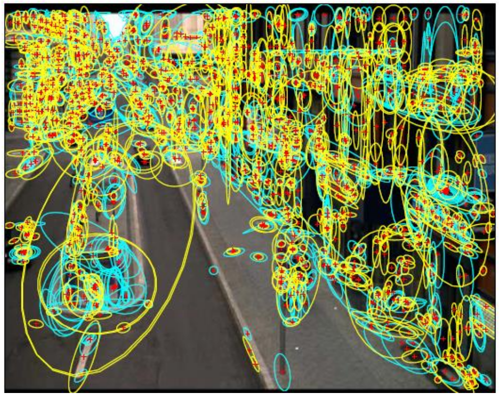


Sampling strategies



Interest operators

Dense, uniformly



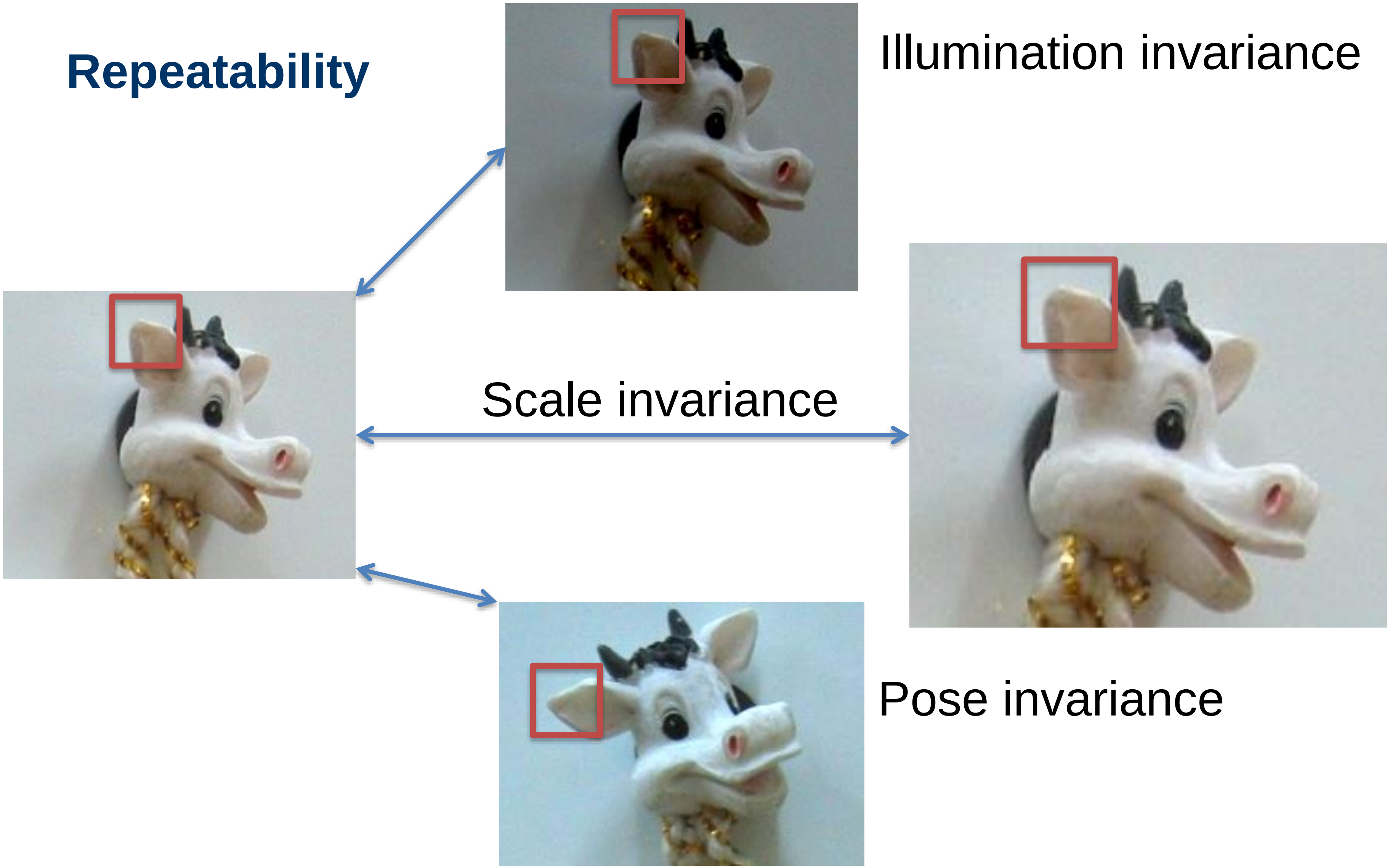
Multiple interest operators

Randomly

Properties of a "good" feature detector

- Repeatability
  - The same feature location can be found in several images despite geometric and photometric transformations
- Saliency
  - Each feature is found at an "interesting" region of the image
- Locality
  - A feature occupies a "relatively small" area of the image

Repeatability



Saliency & locality

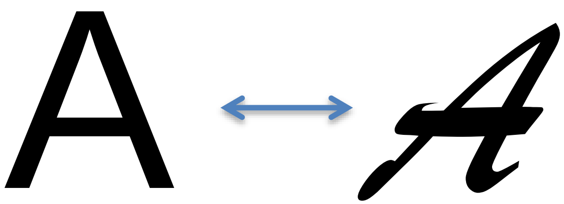
- Saliency

- Locality

Properties of a “good” feature descriptor

Highly dependent on the application, a descriptor must incorporate information that is:

- Invariant w.r.t.:
  - Illumination
  - Pose + Scale (affine transformations)
  - Intraclass variability



- Highly distinctive → allows a single feature to find its correct match with good probability in a large database of features

Feature detection & feature description

- (Edge detectors)
  - Sobel
  - Canny
- Corner detectors
  - Harris
  - FAST
  - AGAST
- Blob detectors
  - DoG (difference of Gaussian)

- SIFT (scale invariant feature transformation)
- SURF (speeded up robust features)
- BRIEF (binary robust independent elementary features)
- ORB (oriented FAST and rotated BRIEF)
- FREAK (fast retina keypoint)
- KAZE
- …

Note: often no separation between detection and description made

SIFT descriptor (David G. Lowe. **"Distinctive image features from scale-invariant keypoints."** *IJCV* 60 (2), 04)

Location and characteristic scale given by DoG detector

1. Compute gradient at each pixel
2. Gaussian center weighting
3. NxN spatial bins
4. Compute a histogram h_i of M orientations for each bin
5. Concatenate h_i for i=1 to N^2 to form a $1xMN^2$ vector H
   (Typically: M=8, N=4 → H: 1x128d)
6. Normalize to unit norm

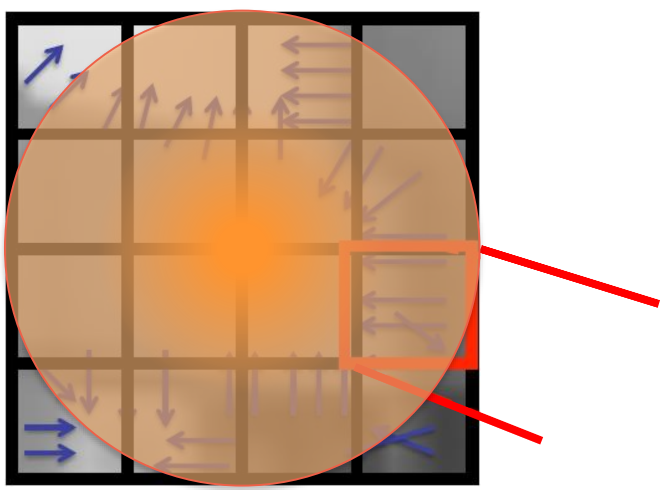
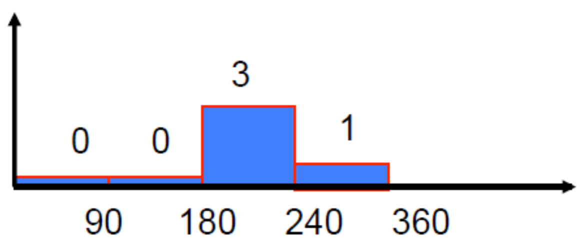


Image patch



SIFT properties

Robust w.r.t. small variation in:

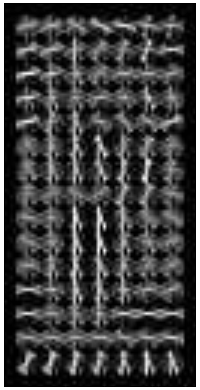
- Illumination (thanks to gradient & normalization)
- Pose (small affine variation thanks to orientation histogram)
- Scale (scale is fixed by DoG)
- Intra-class variability (small variations thanks to histograms)

HOG – histogram of oriented gradients

Navneet Dalal and Bill Triggs, “Histograms of Oriented Gradients for Human Detection”, CVPR 2005

Like SIFT, but…

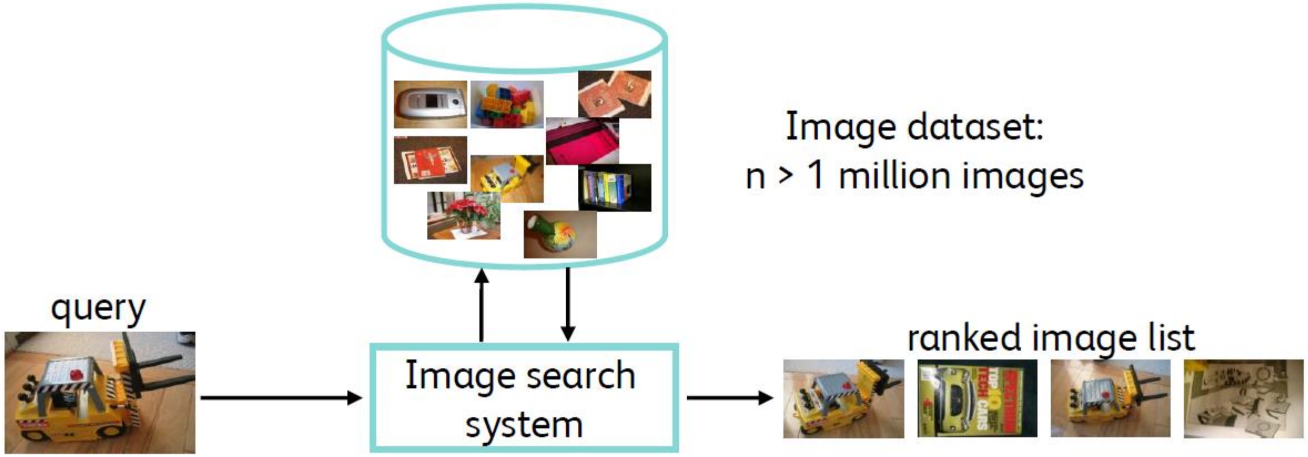
- Sampled on a dense, regular grid around the object
- Gradients are contrast normalized in overlapping blocks



Bag of (visual) Words

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Direct matching: a retrieval example



Assume an image described by m=1000 descriptors (dimension d=128)

→ n*m=1 billion descriptors to index

Database representation in RAM: ~128 GB with 1 byte per dimension

Search: **m^2** x n x d elementary operations

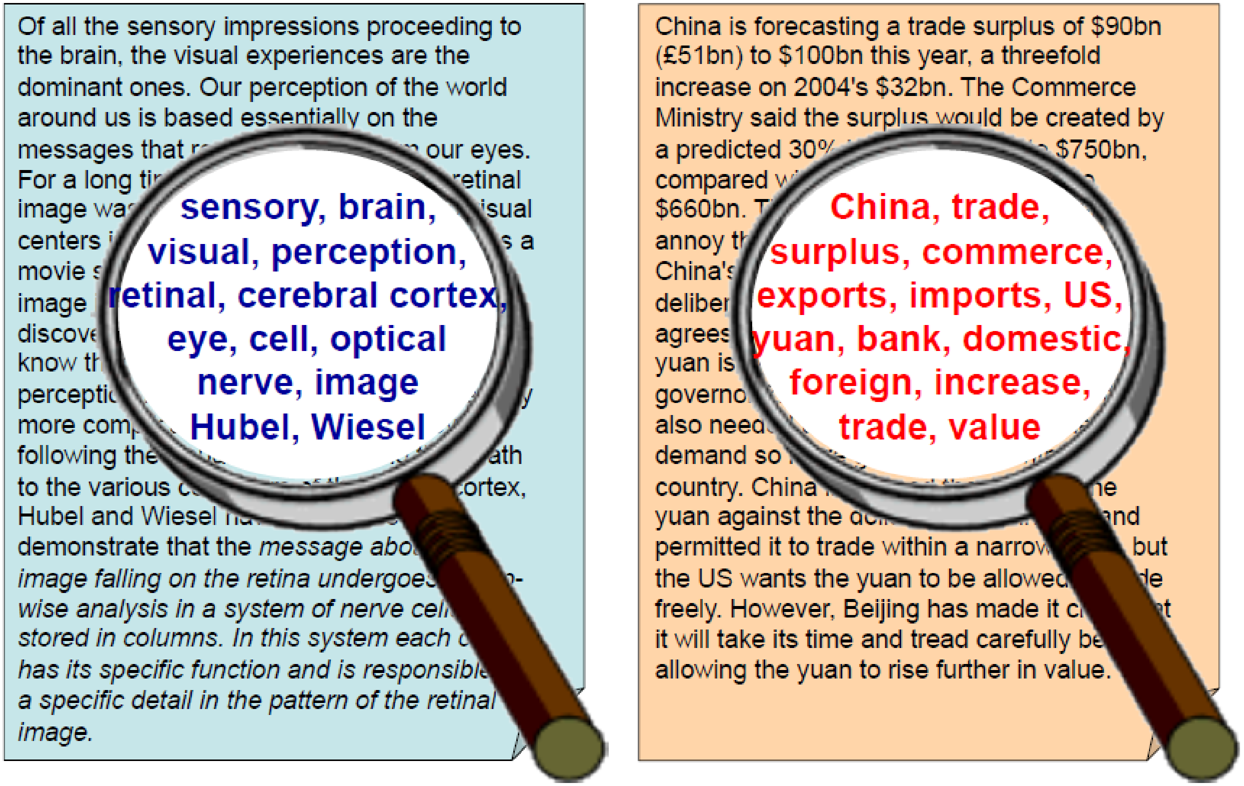
→ **10^{14} computationally intractable**

Bag of words: inspired by works on document analysis

- Early “bag of words” models: mostly texture recognition, e.g. Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003;

- Hierarchical Bayesian models for documents (pLSA, LDA, etc.) Hoffman 1999; Blei, Ng & Jordan, 2004; Teh, Jordan, Beal & Blei, 2004

BoW: Analogy to documents



Basis Idea: Represent a document as a *distribution* of words (spatial structure that connects the words is lost)

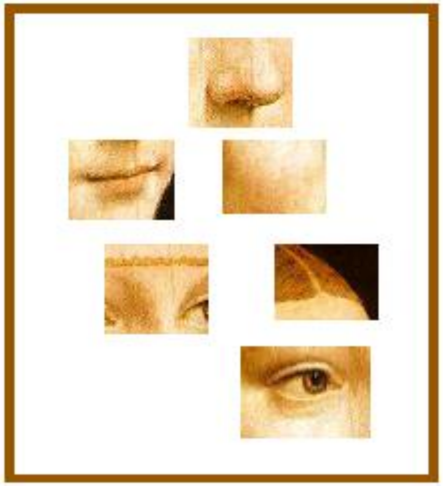
BoW: main principle

Object / Image → Bag of (visual) 'words'

BoW Example

- Independent features

face



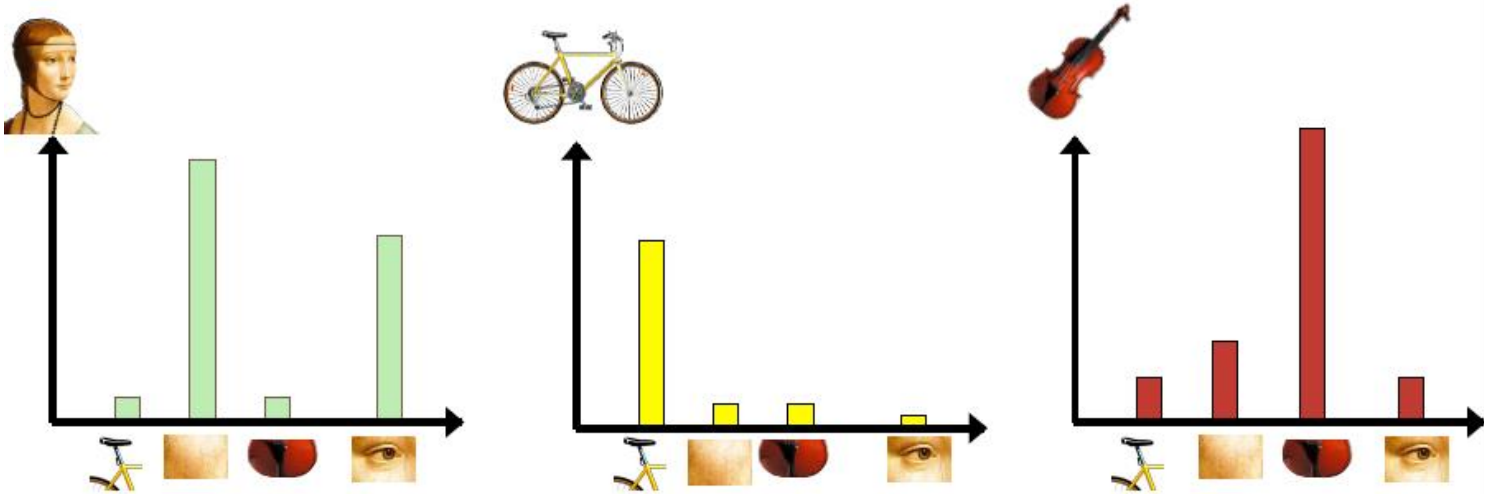
bike

violin

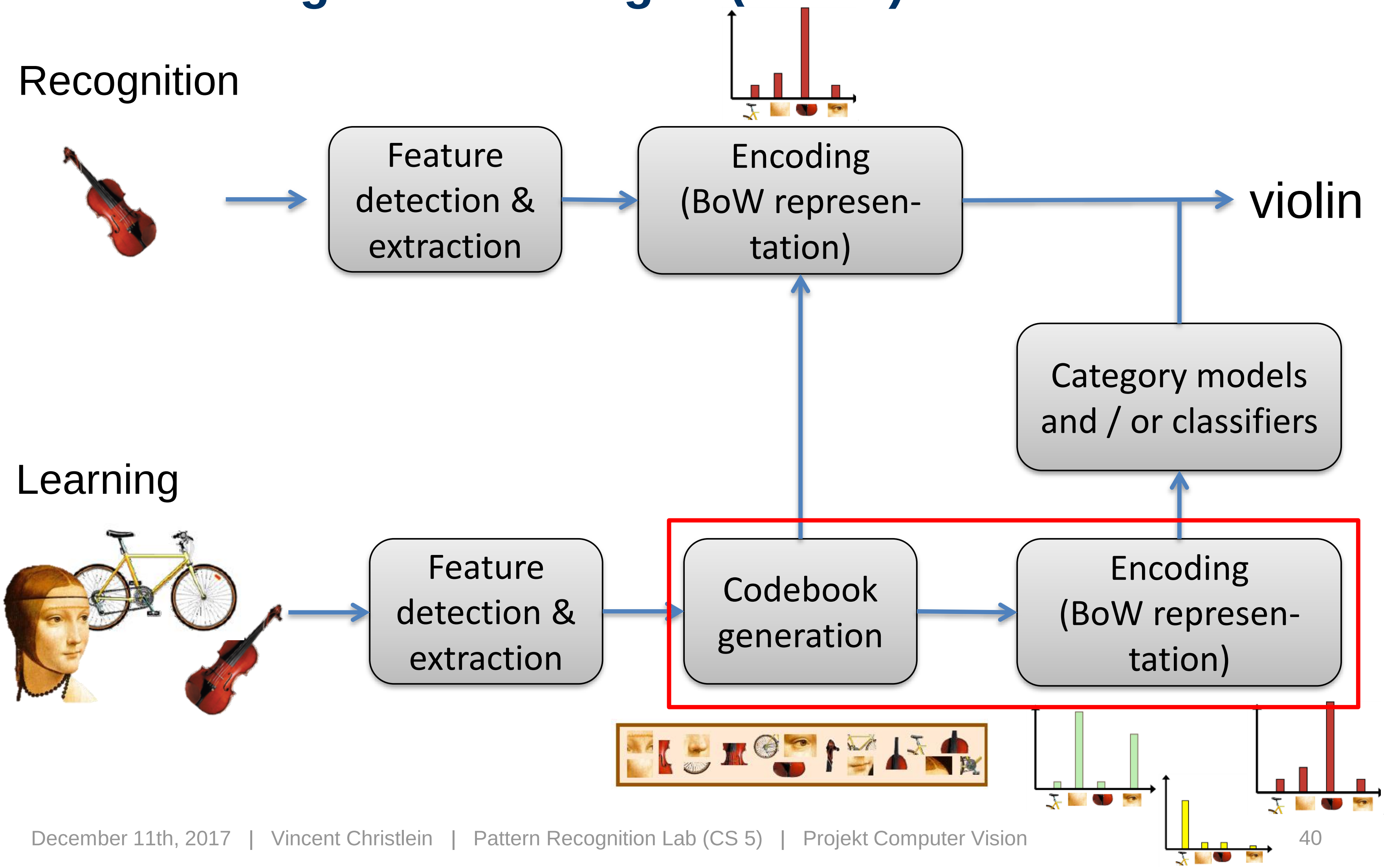
BoW Example

- Independent features
- Train codebook (codewords dictionary) / background model
- Encoding: e.g. histogram representation: represent each image as a frequency of codewords

dictionary



Visual Recognition via bag of (visual) words



Encoding

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

BoW basic encoding: vector quantization

- Local Feature descriptors $\mathbf{x}_i \in \mathbb{R}^D,\ i = \{1, ..., N\}$
- Run k-means to obtain dictionary $C = \{\mu_1, ...\mu_K\},\ \mu_i \in \mathbb{R}^D$
- Count number of assigned codewords:

$$s_k = \sum_{i=1}^{N} \gamma(\mathbf{x}_i)\,, \quad \gamma(x) := \begin{cases} 1 \text{ if } \mathrm{NN}(x) = \mu_k \\ 0 \text{ else} \end{cases} \quad \mathrm{NN}(x) = \arg\min_{\mu \in C} \|x - \mu\|^2$$

$$\mathbf{s} = (s_1, ..., s_K)^T \in \mathbb{R}^K$$

- Normalization: $l_1,\ l_2$
- Sidenote: Often this simplest encoding method is denoted as “bag of (visual) words”

Visual vocabulary size

For LSVR: need image signatures containing **fine-grained information**:

- retrieval: larger dataset → higher probability to find similar but irrelevant image
- classification: more classes → higher probability to find class which is similar to any given class

BoW (with VQ) answer to the problem: increase visual vocabulary size

- practical problem: assignment of descriptors to visual words becomes costly

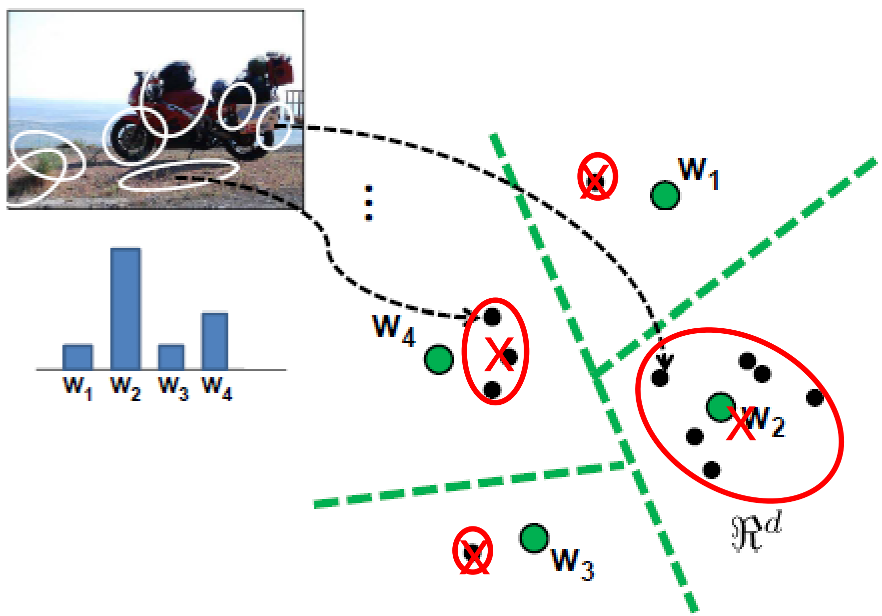
How to increase amount of information **without increasing the visual vocabulary size**?

→ higher-order representations

Higher order representations

VQ: **count** the number of local descriptors assigned to each Voronoi region. Why not including **other statistics**? For instance:

- mean of local descriptors x
- (co)variance of local descriptors



VLAD

- Local Feature descriptors $\mathbf{x}_i \in \mathbb{R}^D,\ i = \{1, ..., N\}$
- Run k-means to obtain dictionary $C = \{\mu_1, ...\mu_K\},\ \mu_i \in \mathbb{R}^D$
- Count number of residuals:

$$\mathbf{s}_k = \sum_{i=1}^{N} \gamma(\mathbf{x}_i)(\mathbf{x}_i - \mu_k)\,, \quad \gamma(x) := \begin{cases} 1 \text{ if } \mathrm{NN}(x) = \mu_k \\ 0 \text{ else} \end{cases}$$

$$\mathbf{s} = (\mathbf{s}_1^T, ..., \mathbf{s}_K^T)^T \in \mathbb{R}^{KD}$$

- Normalization:
  - Intra normalization = component-wise ($\mathbf{s}_k$) l_2 normalization, followed by global l_2
  - Signed square root (power normalization): $\hat{s}_i = \sqrt{|s_i|},\ i = 1, ..., KD$ followed by global l_2

Fisher vectors

- Based on Fisher Kernel
- Local Feature descriptors $\mathbf{x}_i \in \mathbb{R}^D,\ i = \{1, ..., N\}$

- Train GMM, w. parameters: $\Theta = (\mu_k, \mathbf{\Sigma}_k, \pi_k : k = 1, \ldots, K)$
  - Likelihood: $p(\mathbf{x} \mid \Theta) = \sum_{k=1}^{K} w_k g_k(\mathbf{x})\,,$
  - Gaussian density:

$$g_k(\mathbf{x}) = g(\mathbf{x}\,;\ \mu_k, \mathbf{\Sigma}_k) = \frac{1}{\sqrt{(2\pi)^D |\mathbf{\Sigma}_k|}} \mathrm{e}^{-\frac{1}{2}(\mathbf{x}-\mu_k)^\top \mathbf{\Sigma}_k^{-1}(\mathbf{x}-\mu_k)}\ .$$

- Compute association values (posteriors):

$$q_{ik} = \frac{\pi_k g_k(\mathbf{x}_i)}{\sum_{t=1}^{K} \pi_t g_t(\mathbf{x}_i)}$$

Fisher vectors (cont.)

- Compute first and second order statistics:

$$u_{jk} = \frac{1}{N\sqrt{\pi_k}} \sum_{i=1}^{N} q_{ik} \frac{x_{ji} - \mu_{jk}}{\sigma_{jk}},$$

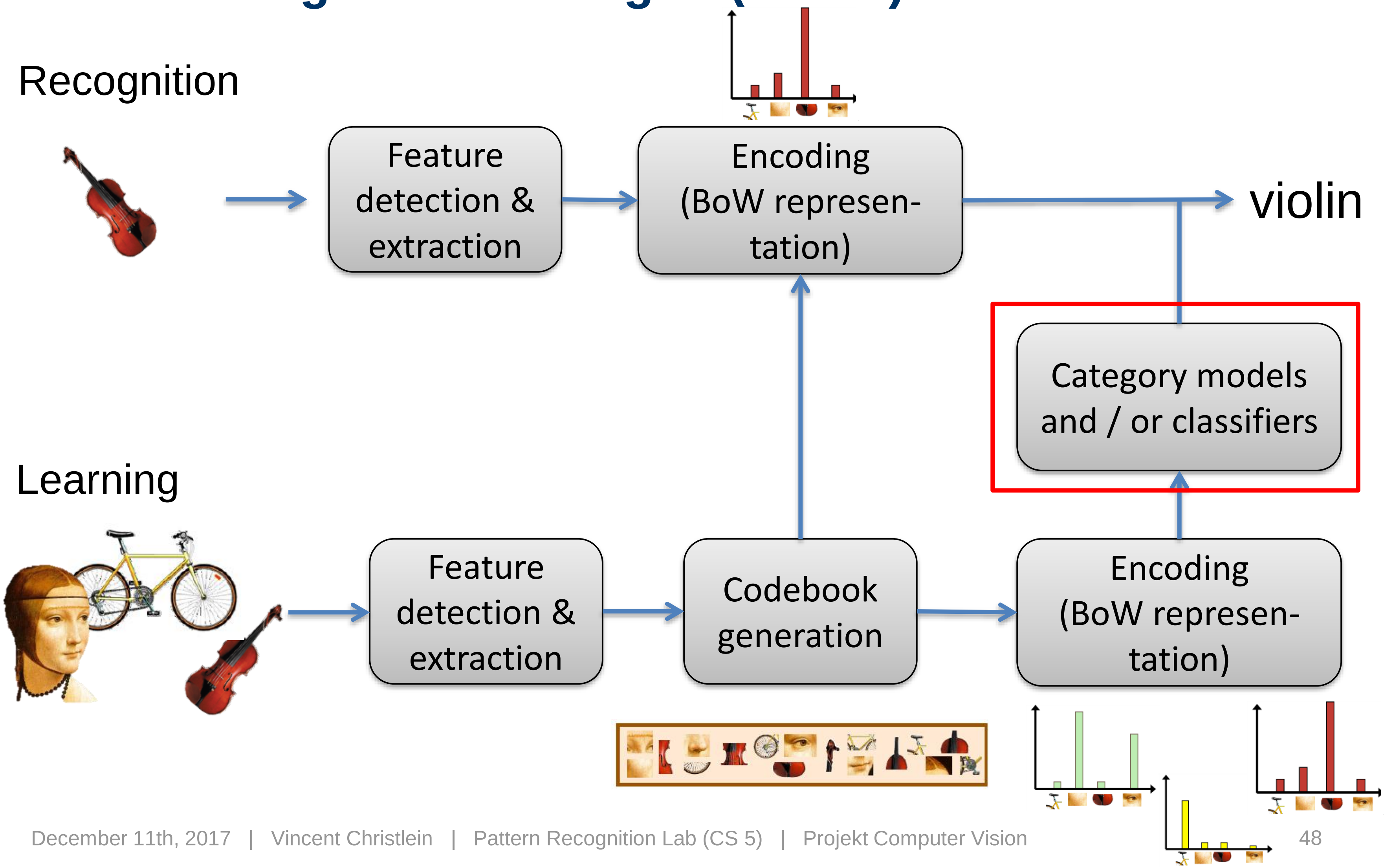
$$v_{jk} = \frac{1}{N\sqrt{2\pi_k}} \sum_{i=1}^{N} q_{ik} \left[\left(\frac{x_{ji} - \mu_{jk}}{\sigma_{jk}} \right)^2 - 1 \right].$$

- Full descriptor:

$$\mathbf{s} := \begin{bmatrix} \vdots \\ \mu_k \\ \vdots \\ \mathbf{v}_k \\ \vdots \end{bmatrix}$$

- Normalization:
  Signed square root (power normalization): $\hat{s}_i = \sqrt{|s_i|},\ i = 1, ..., 2KD$
  followed by global l_2

Visual Recognition via bag of (visual) words

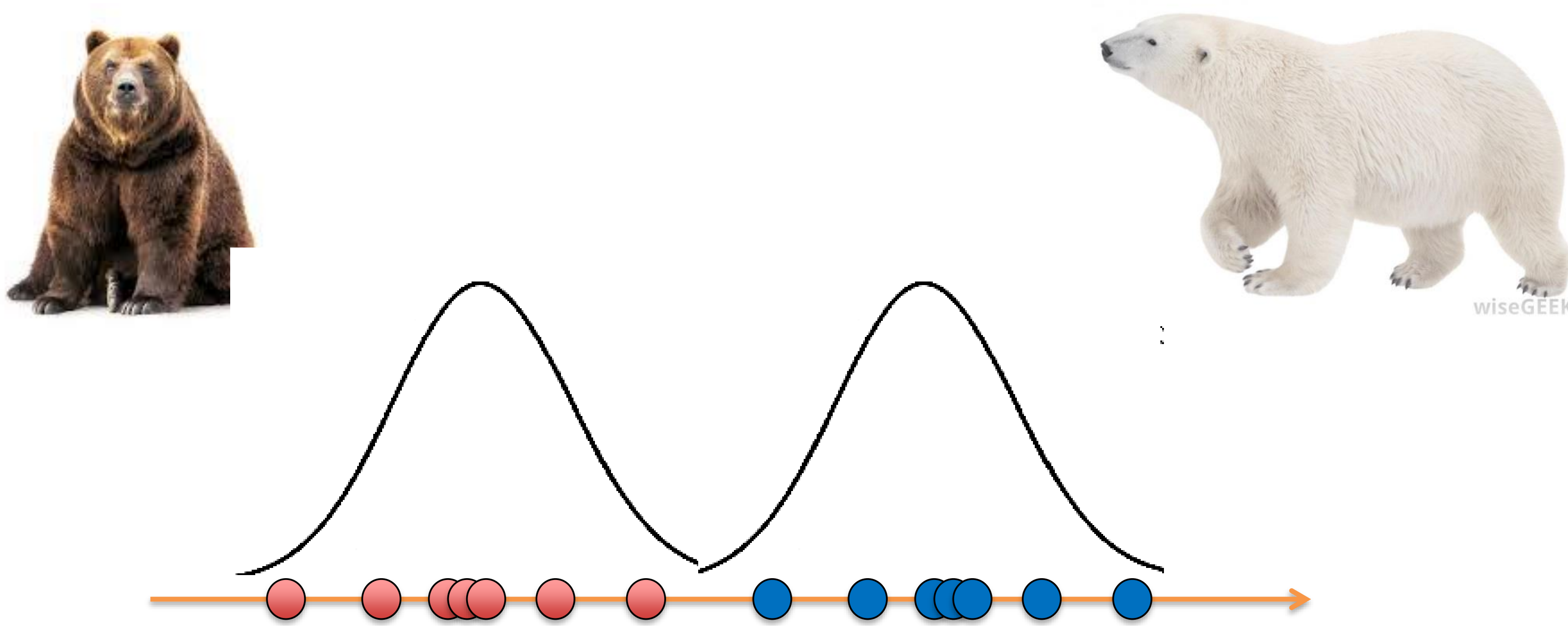


Classification

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

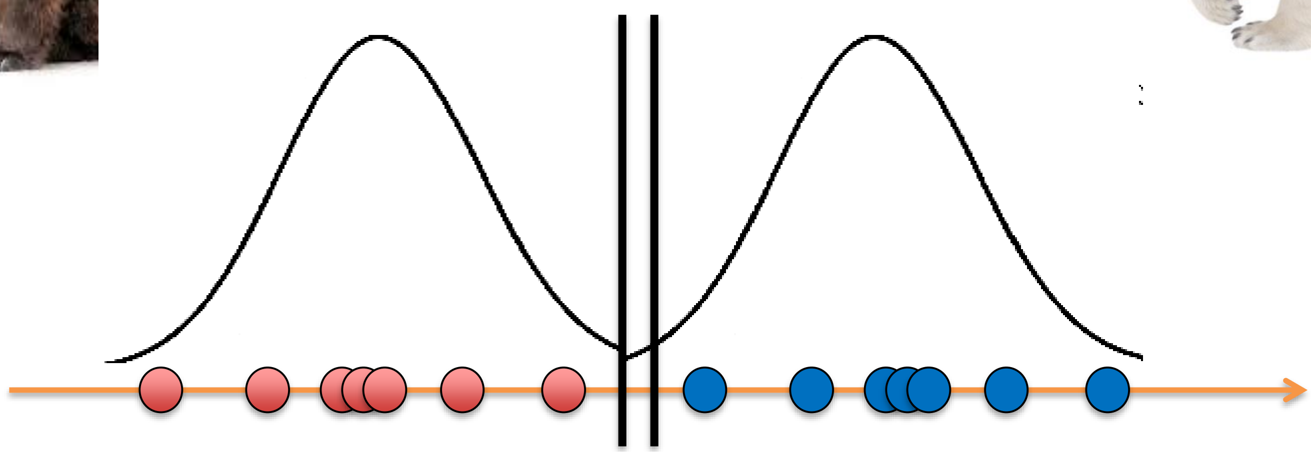
Generative vs. discriminative

Generative: Infer function that can generate (explain) your observations



Generative vs. discriminative

Discriminave: Infer a function that can separate (discriminate) your observations



Generative models

- Naïve Bayes classifier
  - Csurka Bray, Dance & Fan, 2004
- Hierarchical Bayesian topic models (e.g. pLSA and LDA)
  - Object categorization: Sivic et al. 2005, Sudderth et al. 2005
  - Natural scene categorization: Fei-Fei et al. 2005
- 2D Part based models
  - Constellation models: Weber et al 2000; Fergus et al 200
  - Star models: ISM (Leibe et al 05)
- 3D part based models:
  - multi-aspects: Sun, et al, 2009

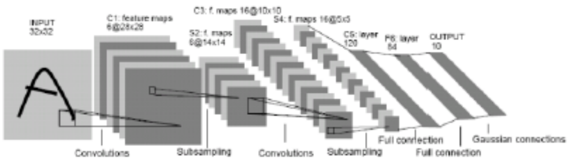
Discriminative models

Nearest neighbor

10^6 examples

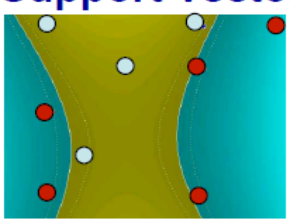
Shakhnarovich, Viola, Darrell 2003
Berg, Berg, Malik 2005...

Neural networks



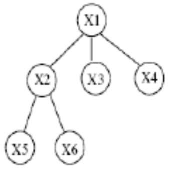
LeCun, Bottou, Bengio, Haffner 1998
Rowley, Baluja, Kanade 1998
…

Support Vector Machines



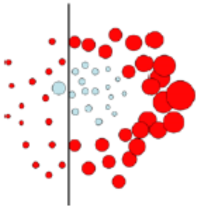
Guyon, Vapnik, Heisele,
Serre, Poggio…

Latent SVM
Structural SVM



Felzenszwalb 00
Ramanan 03…

Boosting



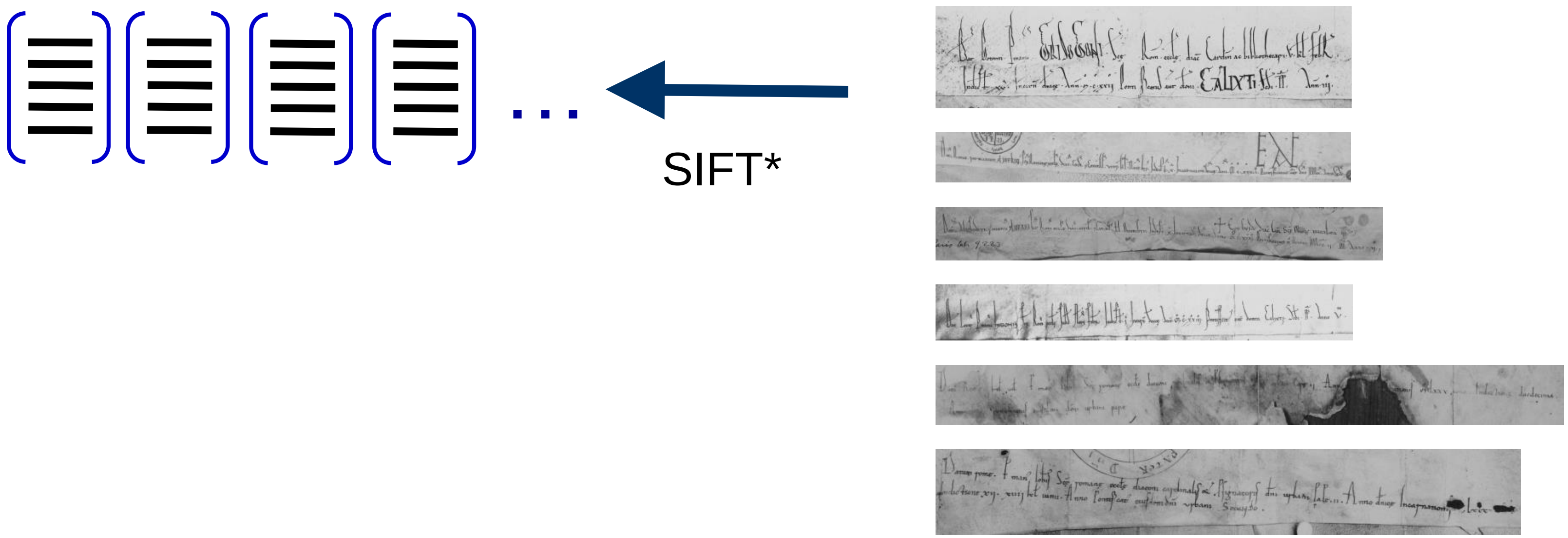
Viola, Jones 2001,
Torralba et al. 2004,
Opelt et al. 2006,…

Practical example: classification of historical dating lines

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Texture classification via BoW

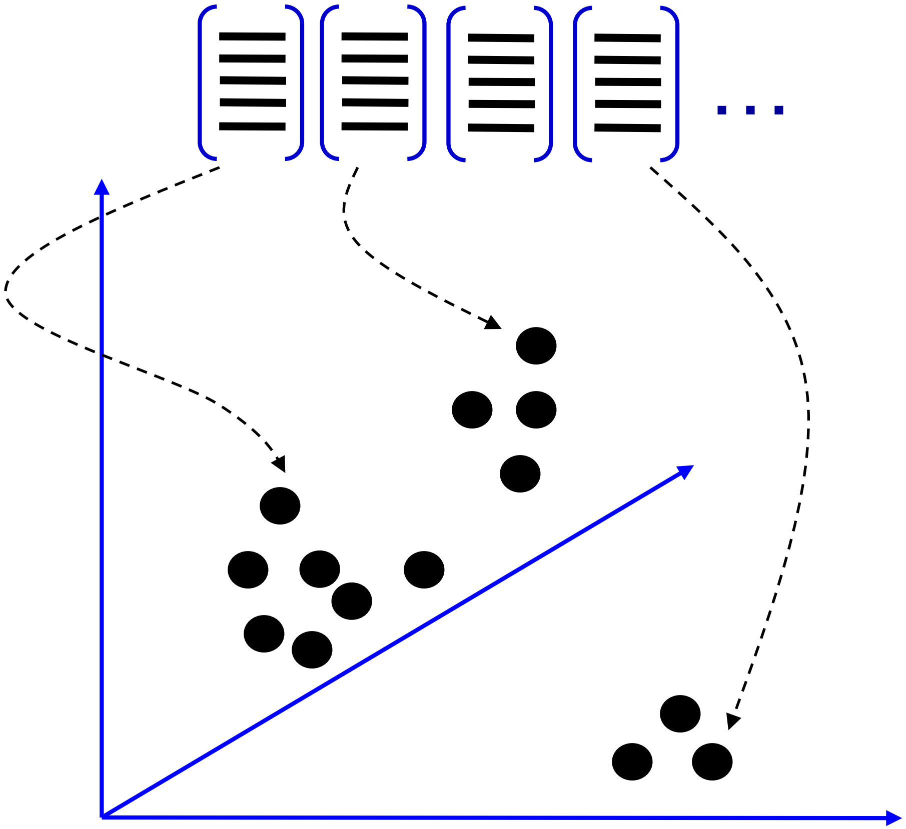
1. Feature detection and extraction



* Lowe, D. G. "Object recognition from local scale-invariant features". Proceedings of the International Conference on Computer Vision, pp. 1150–1157, 1999

Texture classification via BoW

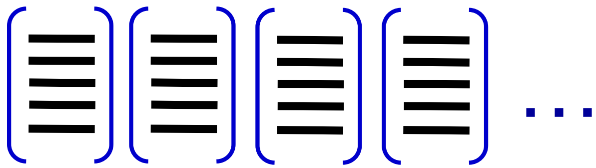
2. Code words dictionary formation

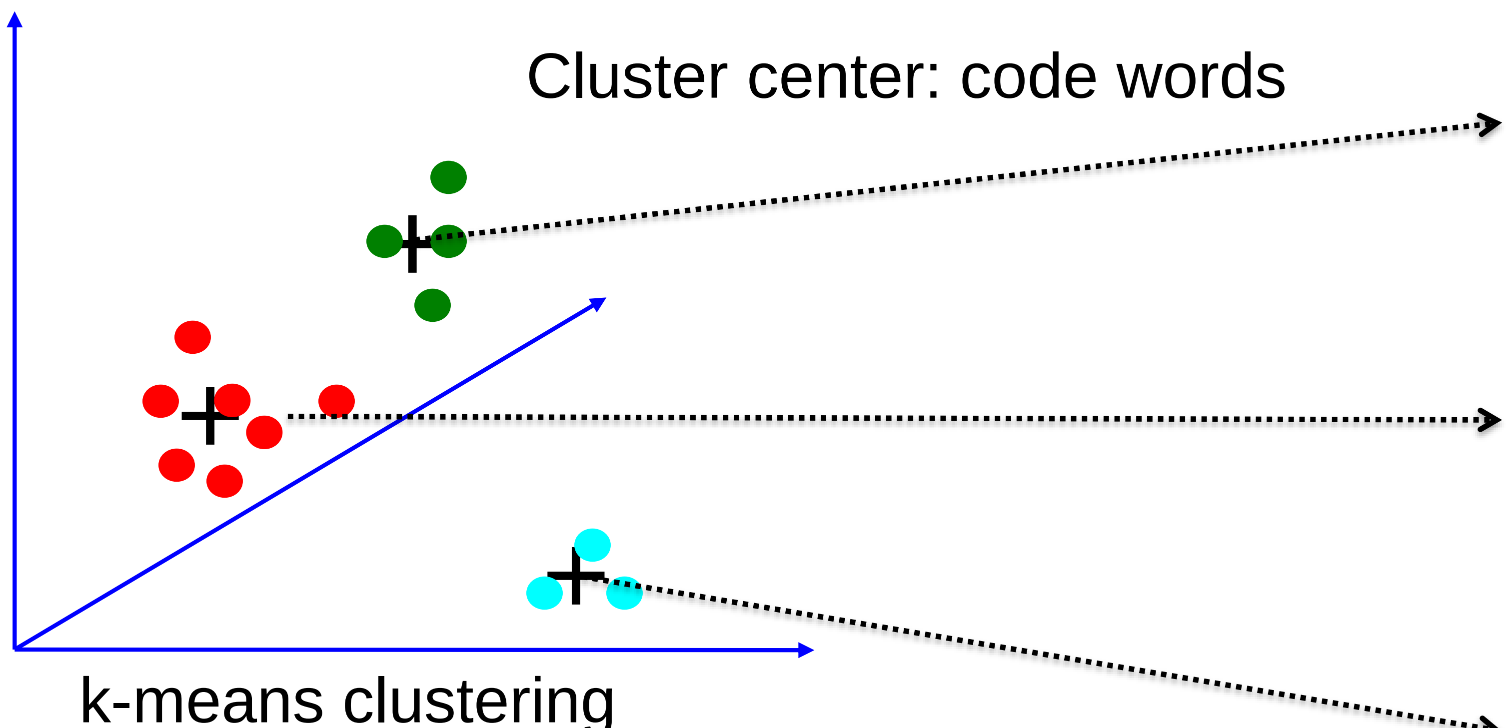


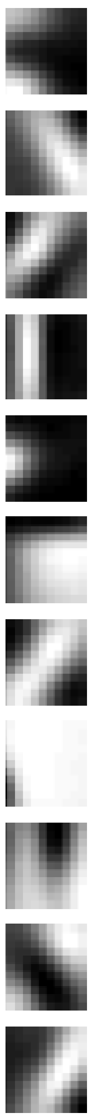
128-D SIFT space

Texture classification via BoW

2. Code words dictionary formation

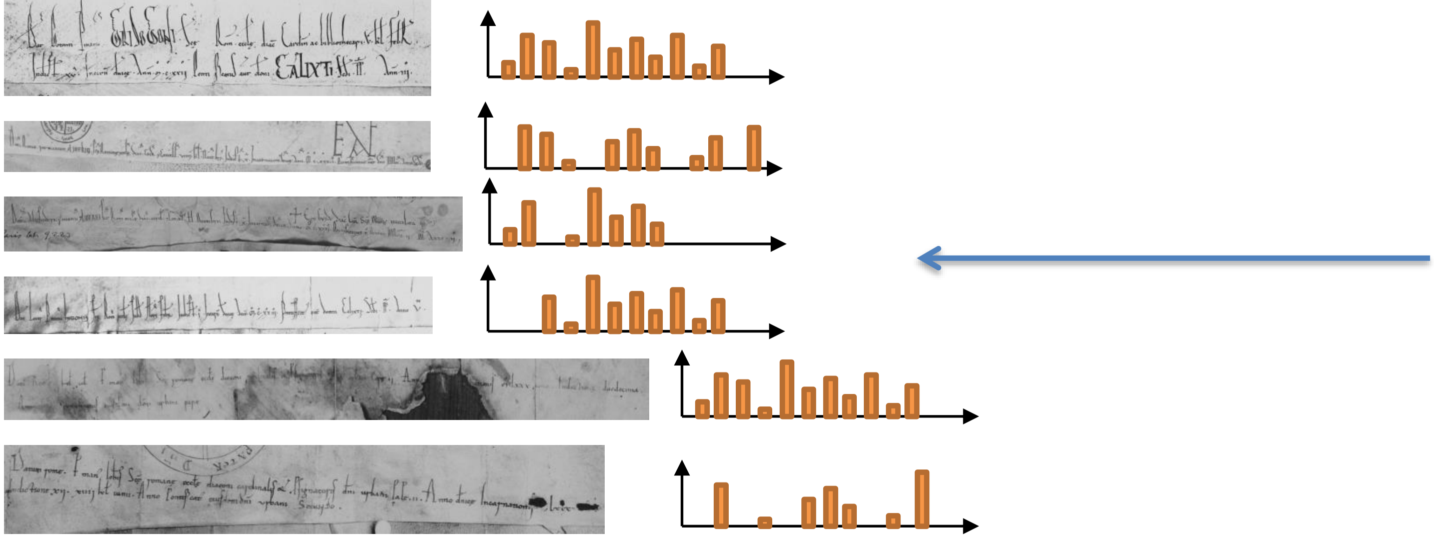


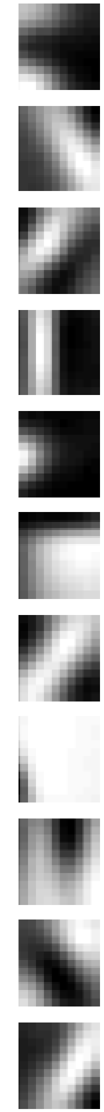




Texture classification via BoW

Represent each line as frequencies of code words





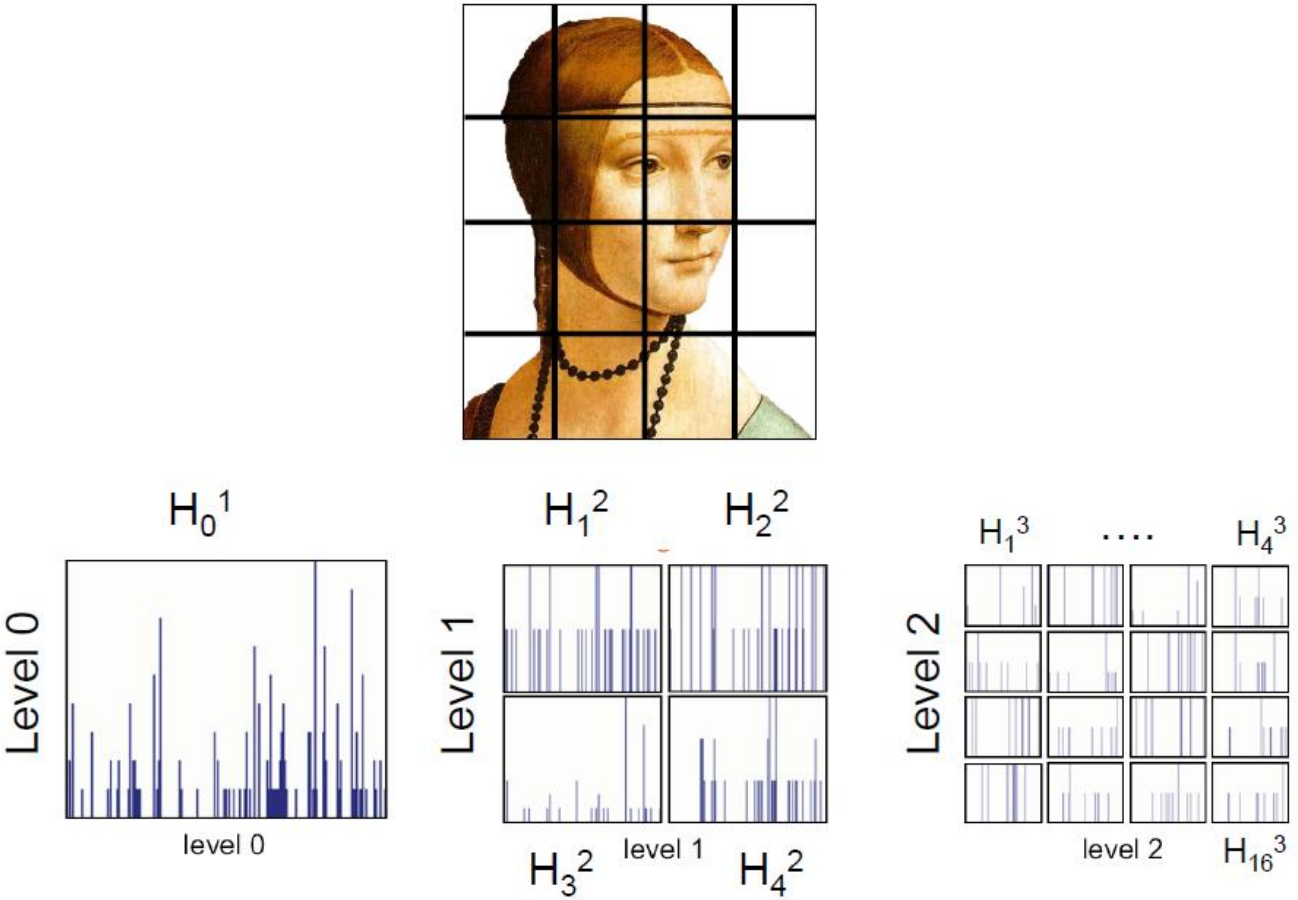
BoW Issues

- How to choose vocabulary size?
  - Too small: visual words not representative of the object appearance distribution
  - Too large: quantization artifacts, sparse histograms, overfitting

- Computational efficiency
  - Vocabulary trees (Nister & Stewenius, 2006)
  - Product quantization (Jégou 2011)

- Localization
  - Spatial pyramid matching (Lazebnik 2009)

Spatial Pyramid Matching



$$H = [H_0^1 \; H_1^2 \ldots H_4^2 \; H_1^3 \; \ldots H_{16}^3]$$

Conclusions

- Pros
  - Very simple and effective
  - Still used today for image retrieval problems

- Cons
  - Not ideal for solving detection problems
  - Outperformed by convolutional neural networks (CNNs) but require way less training data

Questions?

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT