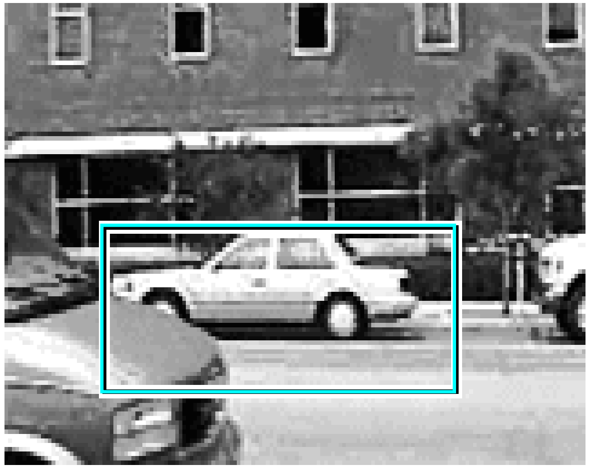
Probabilistic models of visual object categories

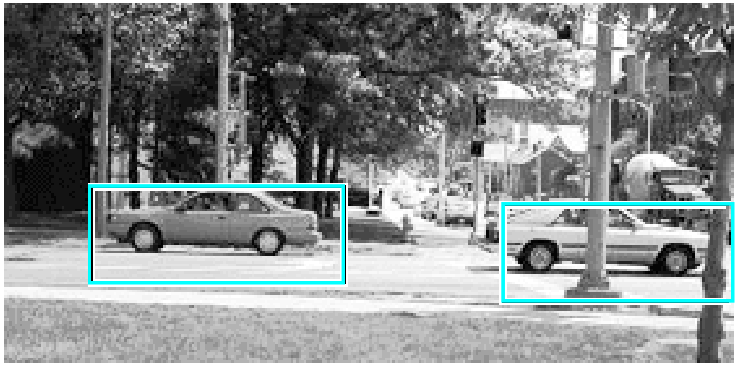
Andrew Zisserman
Visual Geometry Group
University of Oxford
http://www.robots.ox.ac.uk/~vgg

Includes slides from: Mark Everingham, Rob Fergus, Pawan Kumar, Bastian Leibe, Pietro Perona, Josef Sivic and Bernt Schiele

Object Recognition

- identify specific object, or
- identify class (car, face, airplane etc)
- determine location
  - multiple instances in a single image
- determine segmentation





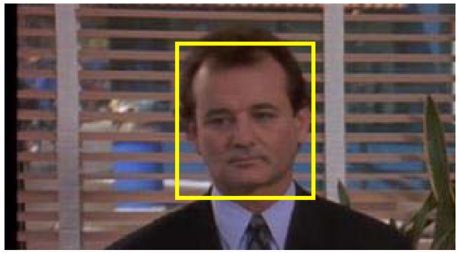
Motivation: Visually defined search

Given an object specified by its image, retrieve all images containing the object in a large image database, or all shots containing the object in a feature film

Visually defined query

"Find this clock"

"Find this person"



"Find this place"

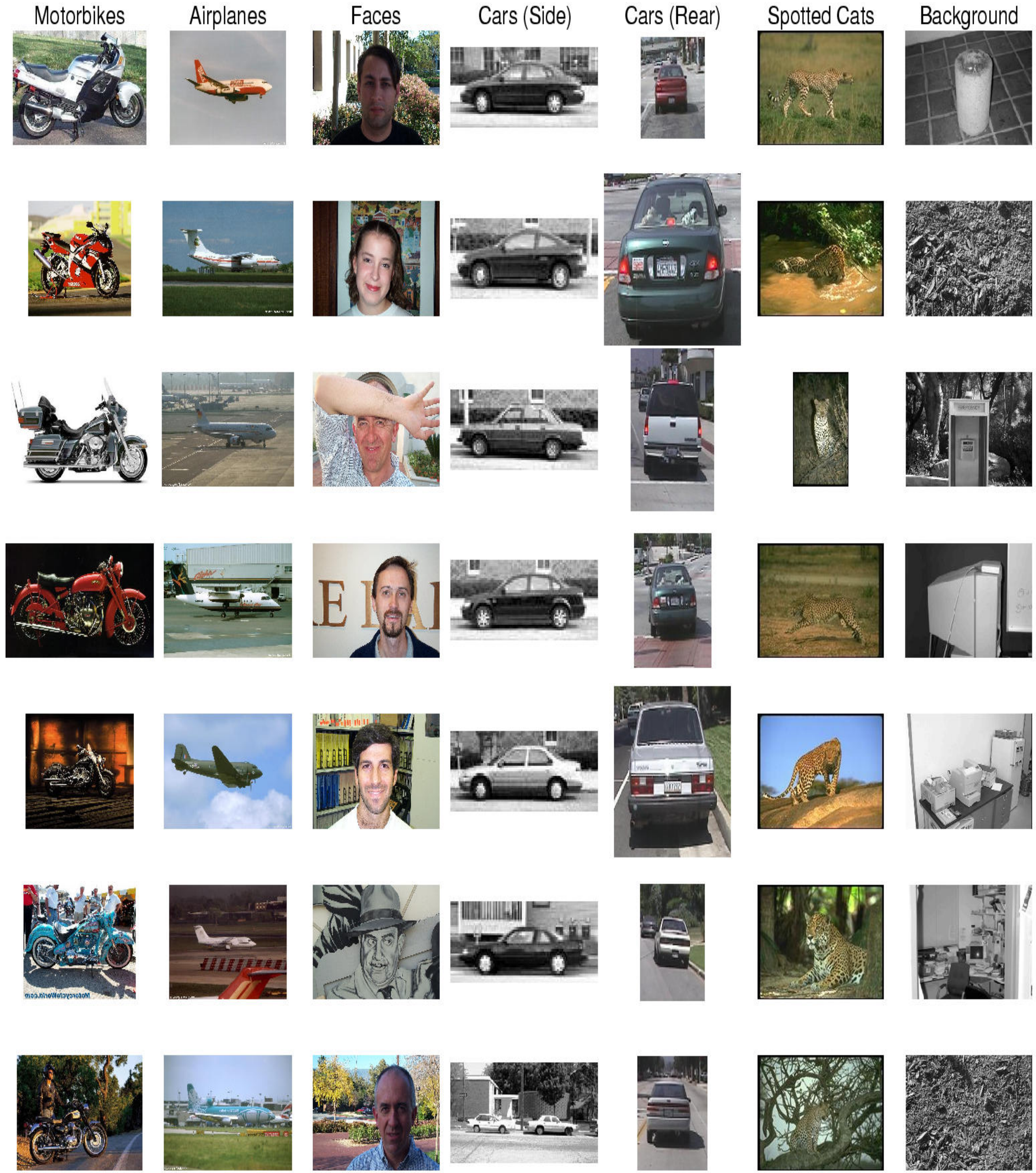
"Groundhog Day" [Rammis, 1993]

e.g. find people and places in your personal photo collection

Why is the recognition problem hard ?

- Scale and shape of the imaged object varies with viewpoint
- Occlusion (self- or by a foreground object)
- Lighting changes
- Background “clutter”

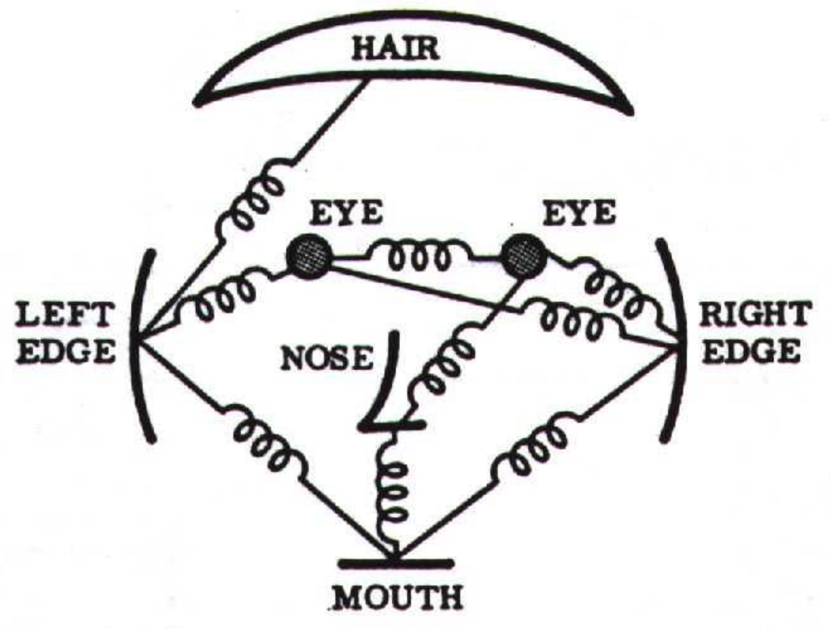
Some object classes (Caltech datasets)



Difficulties:

- Size/shape variation
- Partial occlusion
- Lighting
- Background clutter
- Intra-class variation

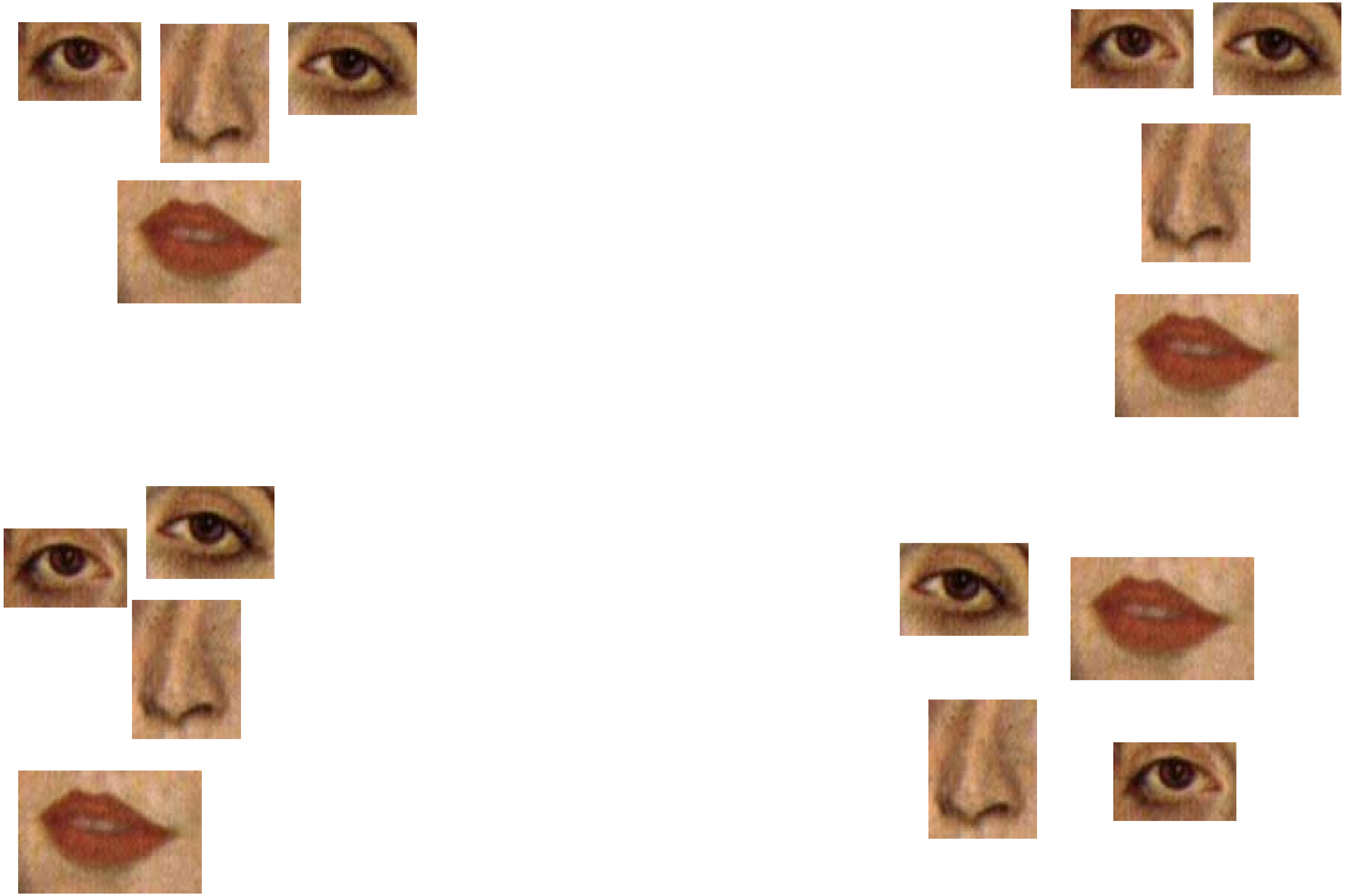
Class of model: Pictorial Structure



- Intuitive model of an object
- Model has two components
  1. parts (2D image fragments)
  2. structure (configuration of parts)
- Dates back to Fischler & Elschlager 1973

Is this complexity of representation necessary ?

Deformations

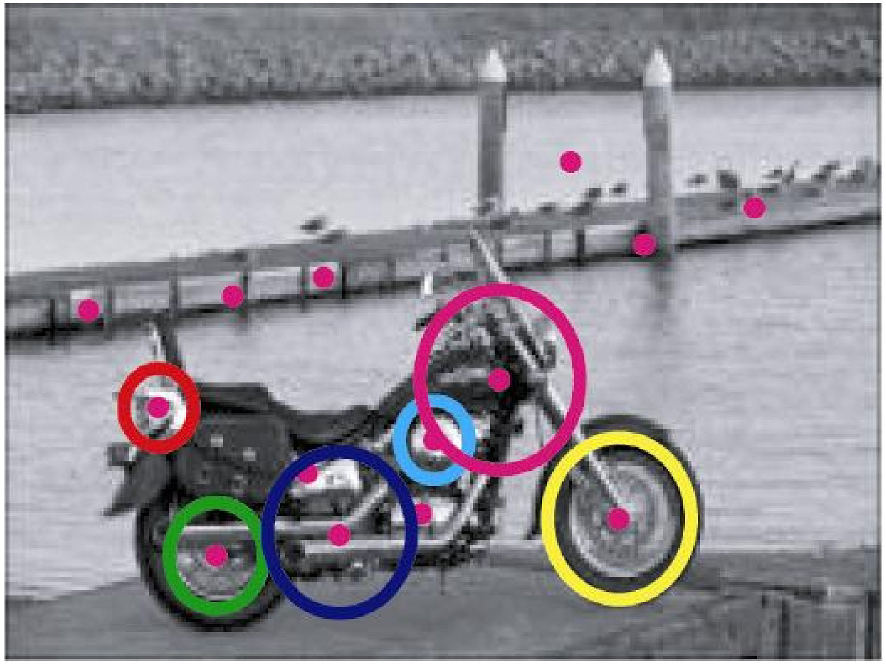


Object Representation

Main issues:

- Parts/fragments
  - appearance, shape
  - exemplars or explicit model
- Structure/configuration
  - model (e.g. implicit or explicit)
  - tight / loose / none
- Model learning
  - degree of supervision
  - from training data
- Model fitting (recognition)
  - complexity

Configuration of 'iconic' parts



Outline

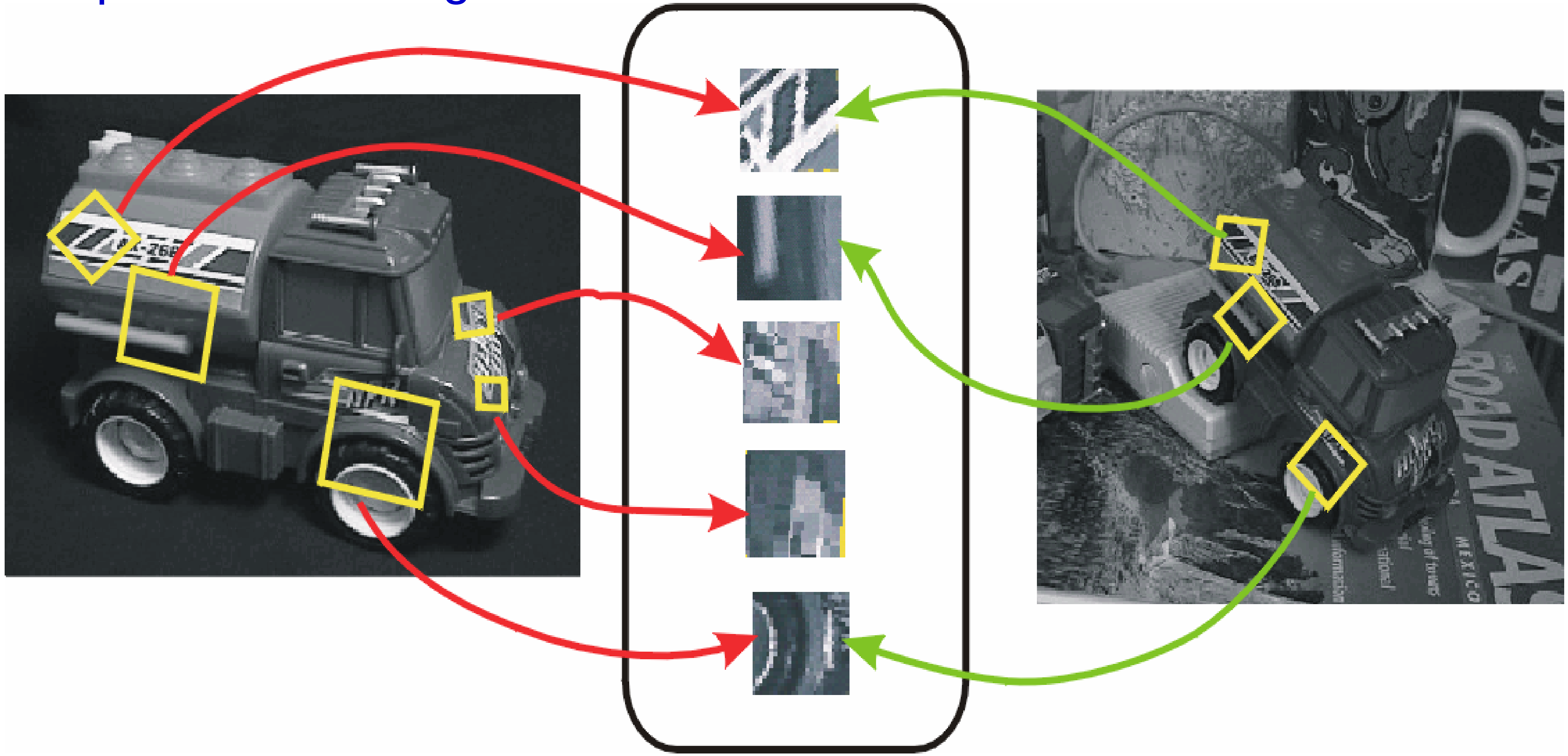
1. Bag of visual words model I: recognizing particular objects
    - Vector quantization to get visual vocabulary (parts)
    - Video Google retrieval algorithm
2. Bag of visual words model II: recognizing object categories
    - Learn classifier for image according to the object it contains
    - Naïve Bayes and SVM classifiers
3. Models of parts and structure
    - Implicit and explicit geometric configurations
4. Class based segmentation
    - Pixel level localization
5. Summary and open challenges

1. Bag of visual words model I: recognizing particular objects

Review: Retrieval using local invariant descriptors

Image content is transformed into local fragments that are invariant to translation, rotation, scale, and other imaging parameters

Example of visual fragments



• Fragments generalize over viewpoint and lighting

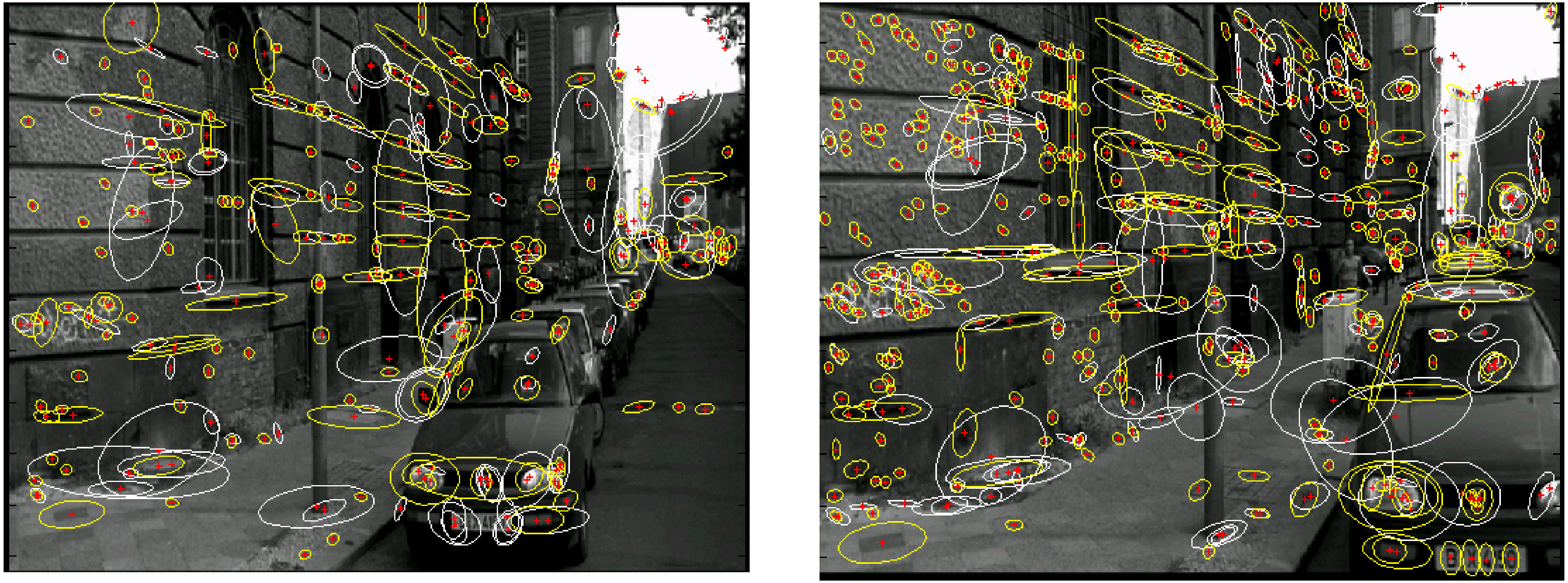
Lowe ICCV 1999

Viewpoint covariant segmentation

- Characteristic scales (size of region)
  - Lindeberg and Garding ECCV 1994
  - Lowe ICCV 1999
  - Mikolajczyk and Schmid ICCV 2001
- Affine covariance (shape of region)
  - Baumberg CVPR 2000
  - Matas et al BMVC 2002 — Maximally stable regions
  - Mikolajczyk and Schmid ECCV 2002
  - Schaffalitzky and Zisserman ECCV 2002

  Shape adapted regions
  - Tuytelaars and Van Gool BMVC 2000

Example of affine covariant regions



1000+ regions per image

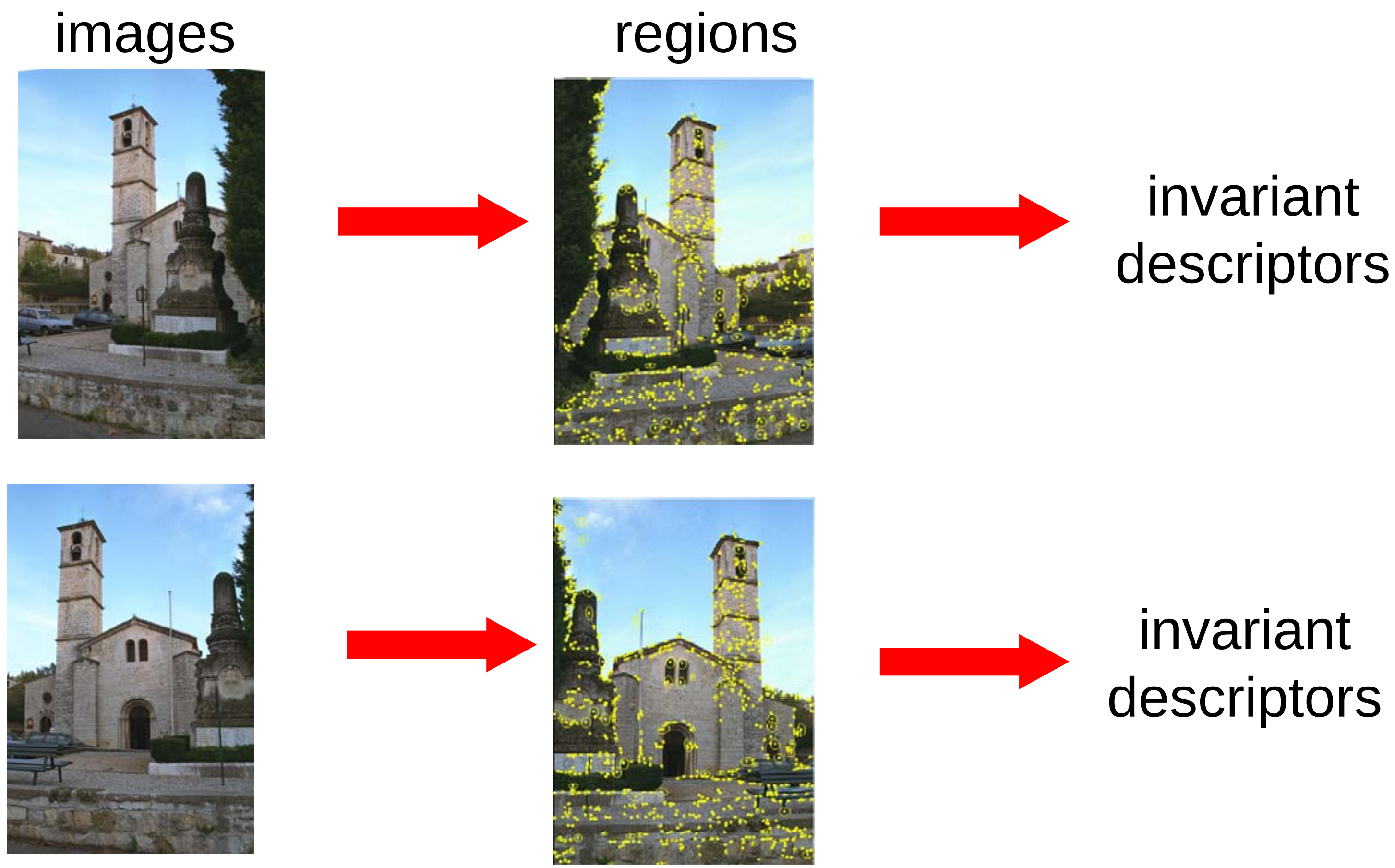
Harris-affine

Maximally stable regions

- a region's size and shape are not fixed, but
- automatically adapts to the image intensity to cover the same physical surface
- i.e. pre-image is the same surface region

Represent each region by SIFT descriptor (128-vector) [Lowe 1999]

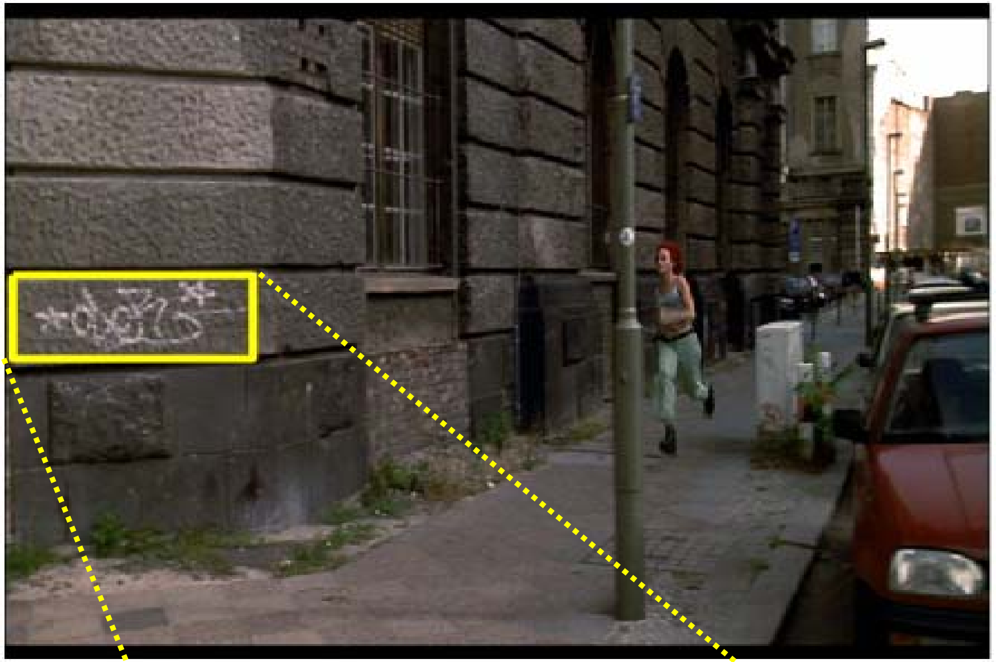
Outline of an object retrieval strategy



1. Compute regions in each image independently
2. “Label” each region by a vector of descriptors based on its local intensity neighbourhood
3. Find corresponding regions by matching to closest descriptor vector
4. Score each frame in the database by the number of matches

Finding corresponding regions transformed to finding nearest neighbour vectors

Example of model matching

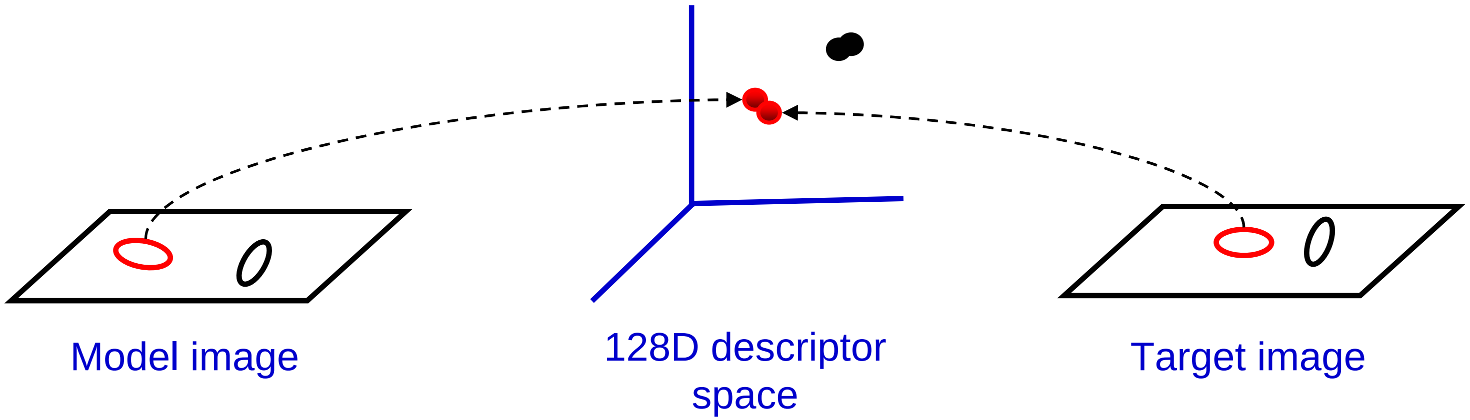


target image

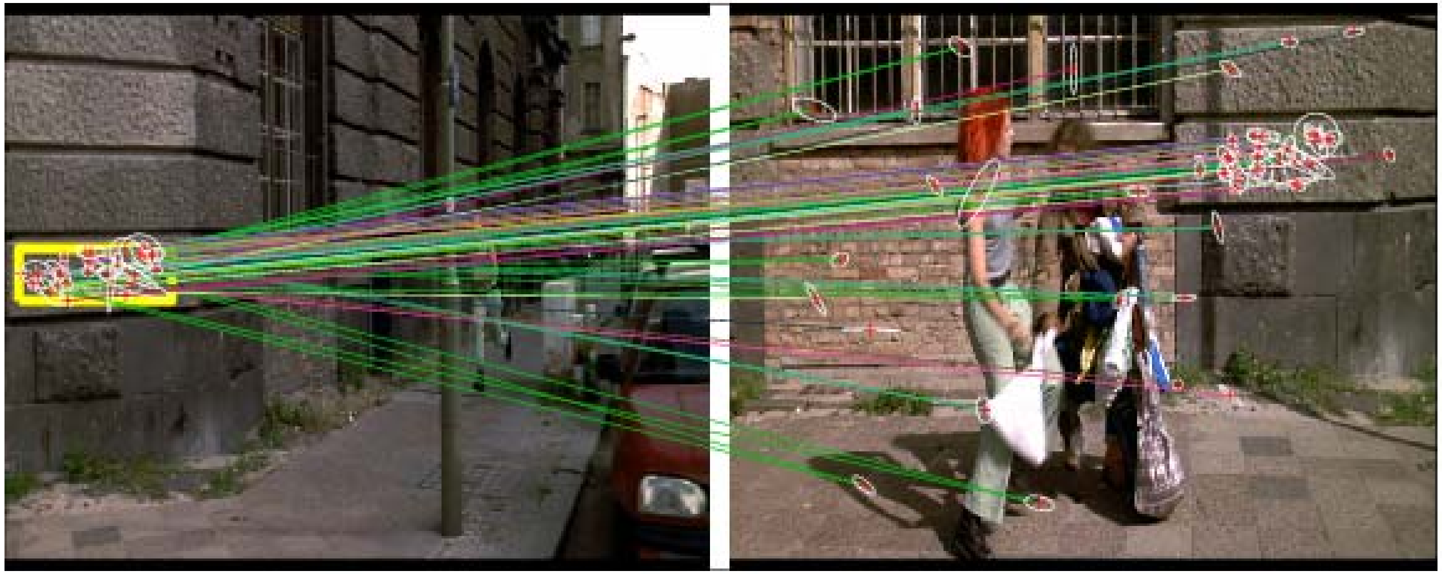
“model”

Object recognition

Establish correspondences between object model image and target image by nearest neighbour matching on SIFT vectors



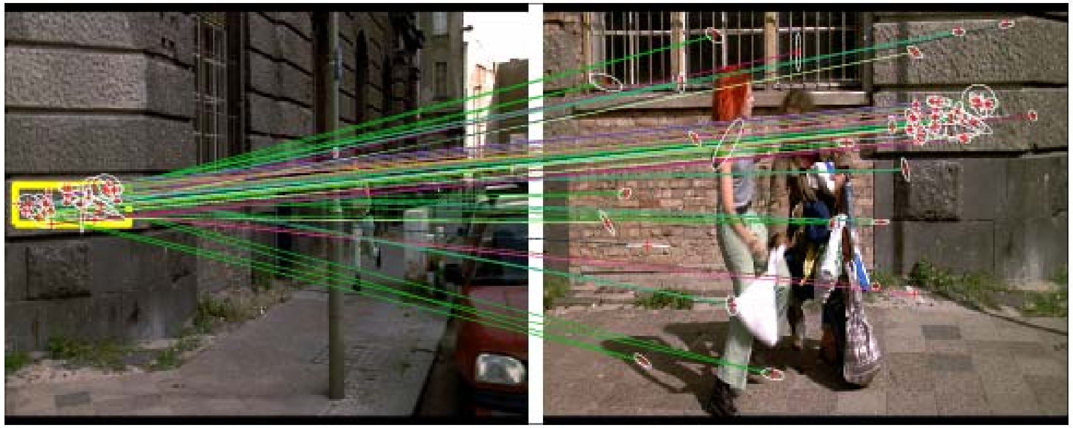
Problem with matching on descriptors alone



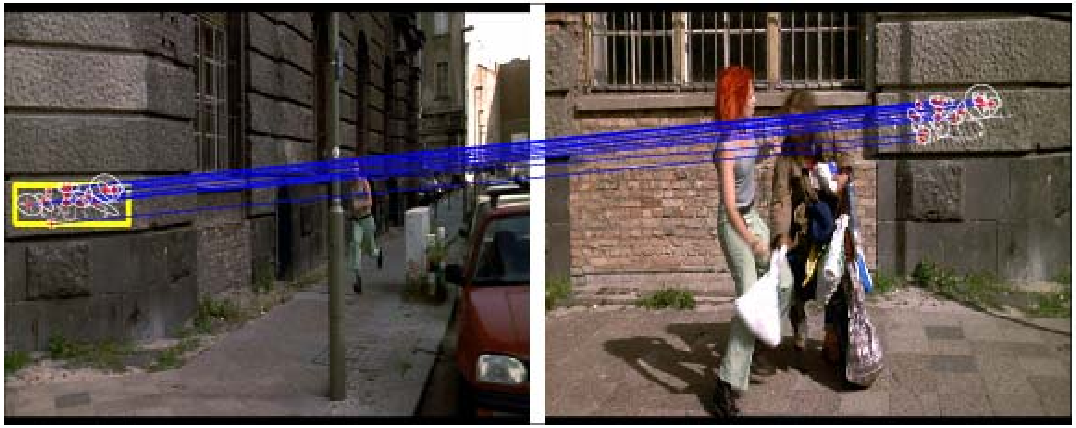
- too much individual invariance
- each region can affine deform independently (by different amounts)
- use semi-local and global spatial relations to verify matches, e.g.:
  - common affine transformation (Lowe 99) (strong requirement)
  - spatial neighbours match spatial neighbours (weak requirement)

Example I

Matches on descriptors



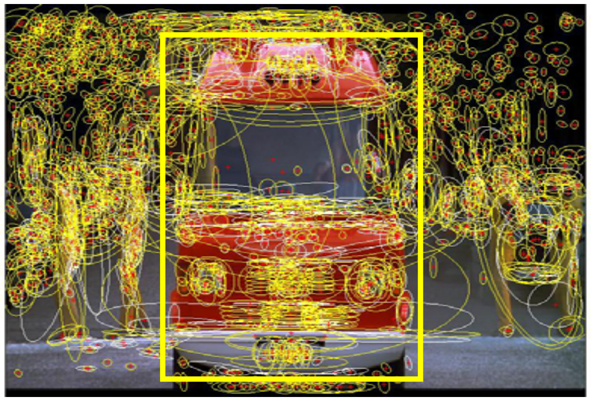
And with spatial consistency



Example II

In each frame independently

- determine elliptical regions (segmentation covariant with camera viewpoint)
- compute SIFT descriptor for each region [Lowe ‘99]

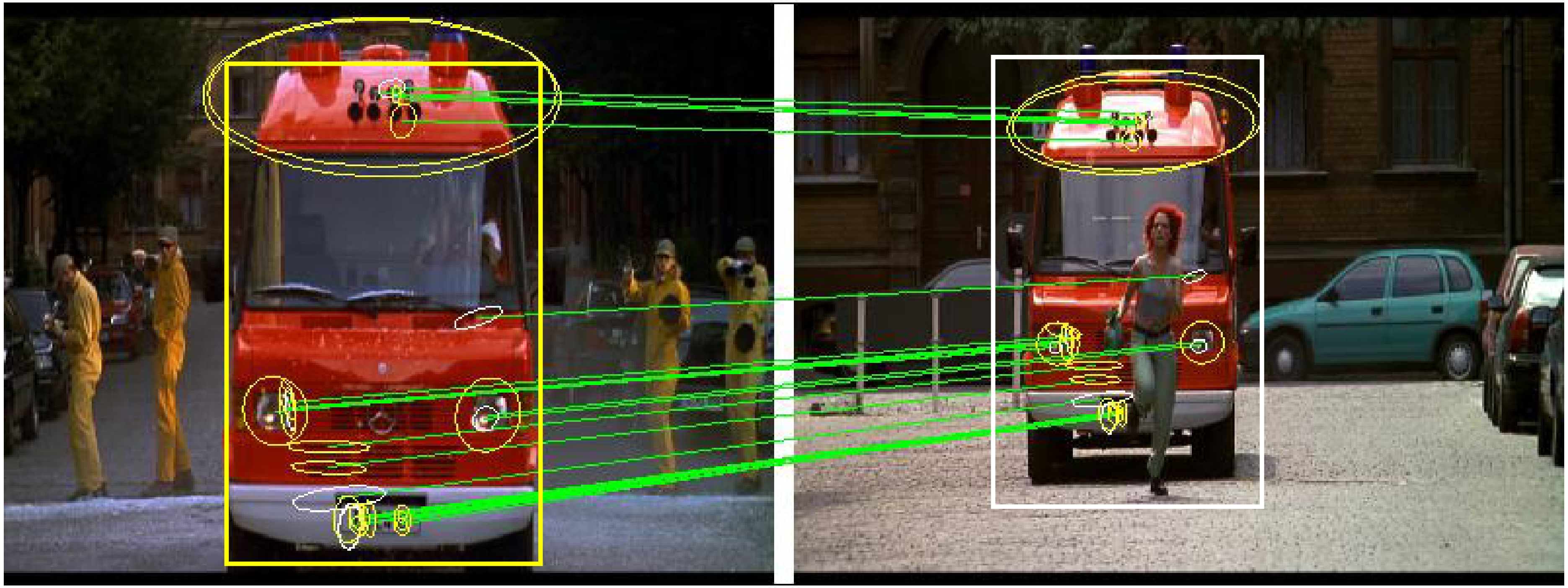




1000+ descriptors per frame

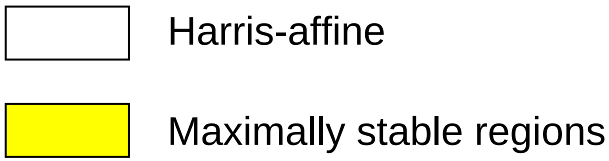
Harris-affine

Maximally stable regions

Match regions between frames using SIFT descriptors and spatial consistency



- Multiple fragments overcomes problem of partial occlusion
- Transfer bounding box to localize object

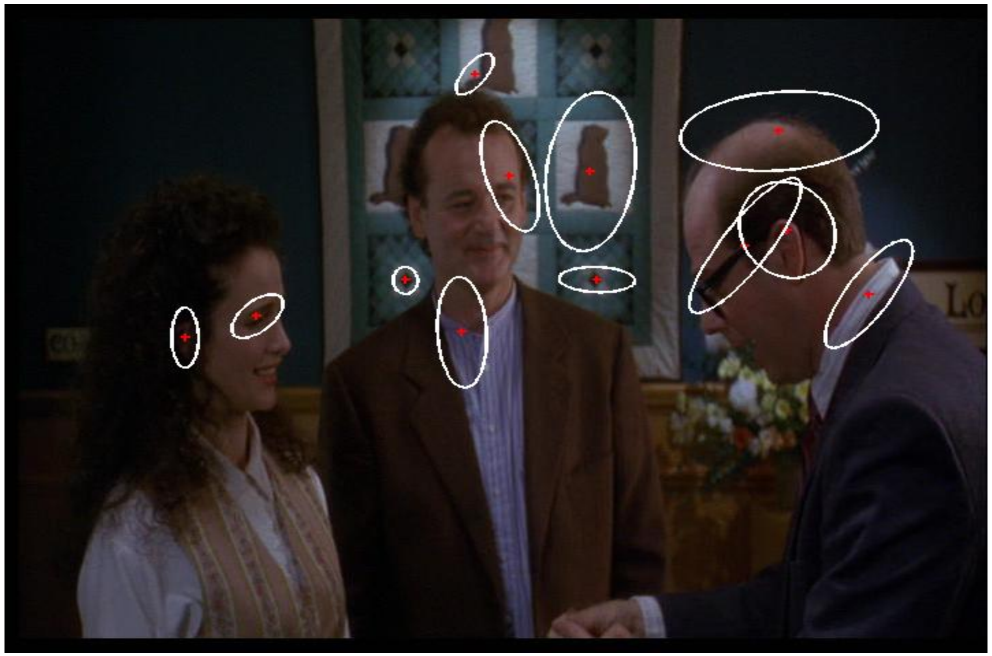


One-shot learning

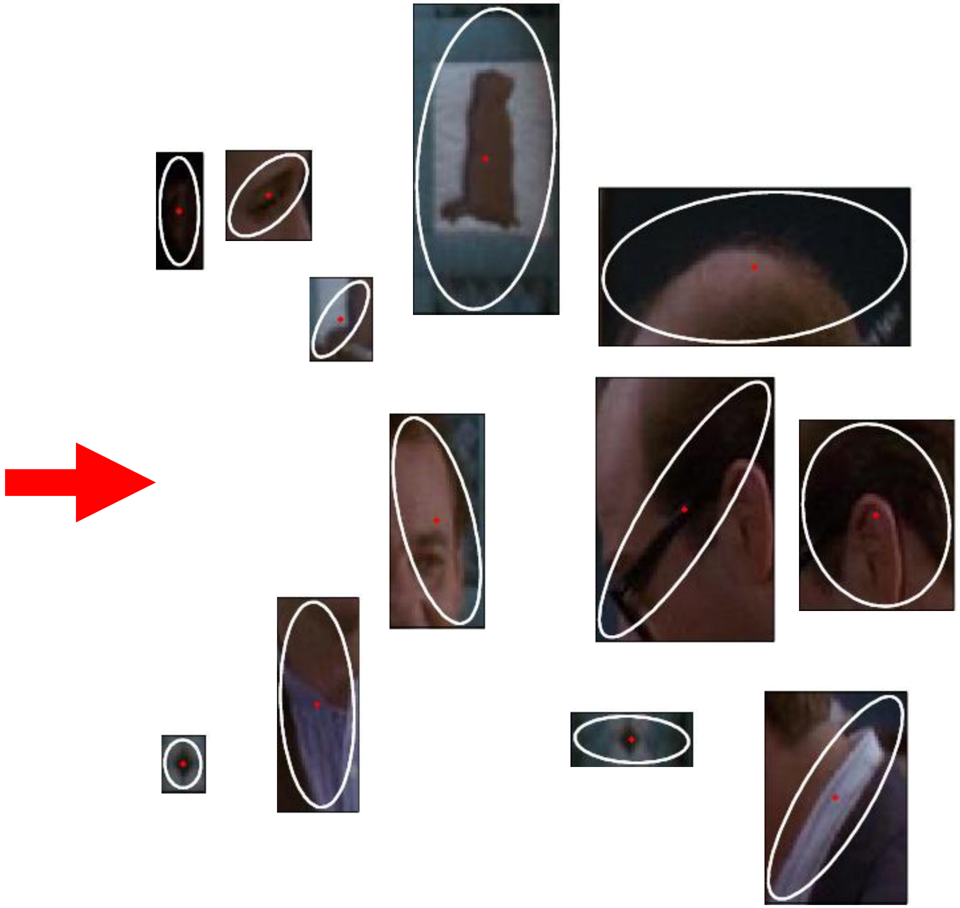
New representation: Bag of (visual) words

Visual words are 'iconic' image patches or fragments

- represent the frequency of word occurrence
- but not their position



Image

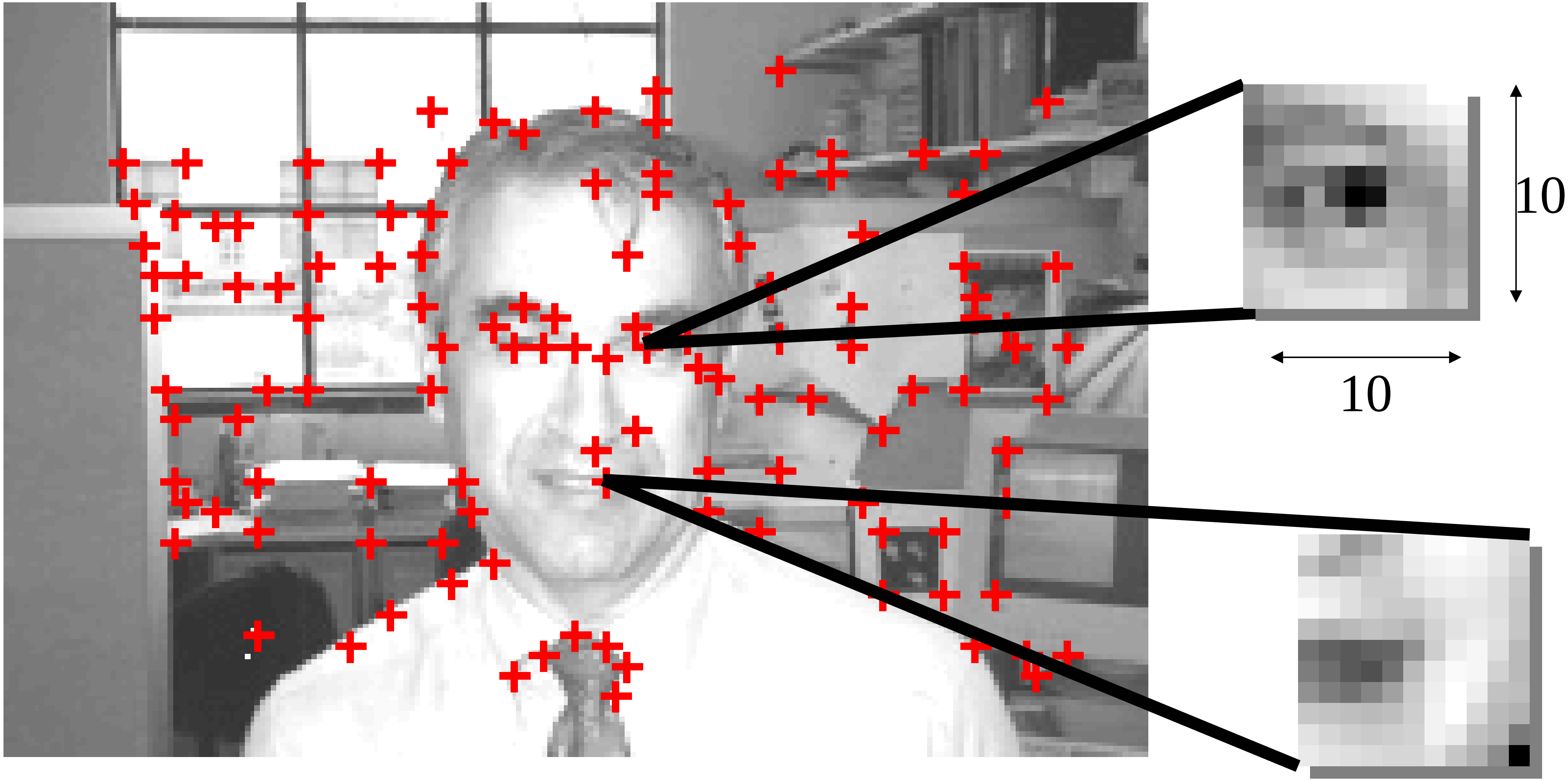


Collection of visual words

Image representation using visual words

histogram represents the co-occurrence of visual words

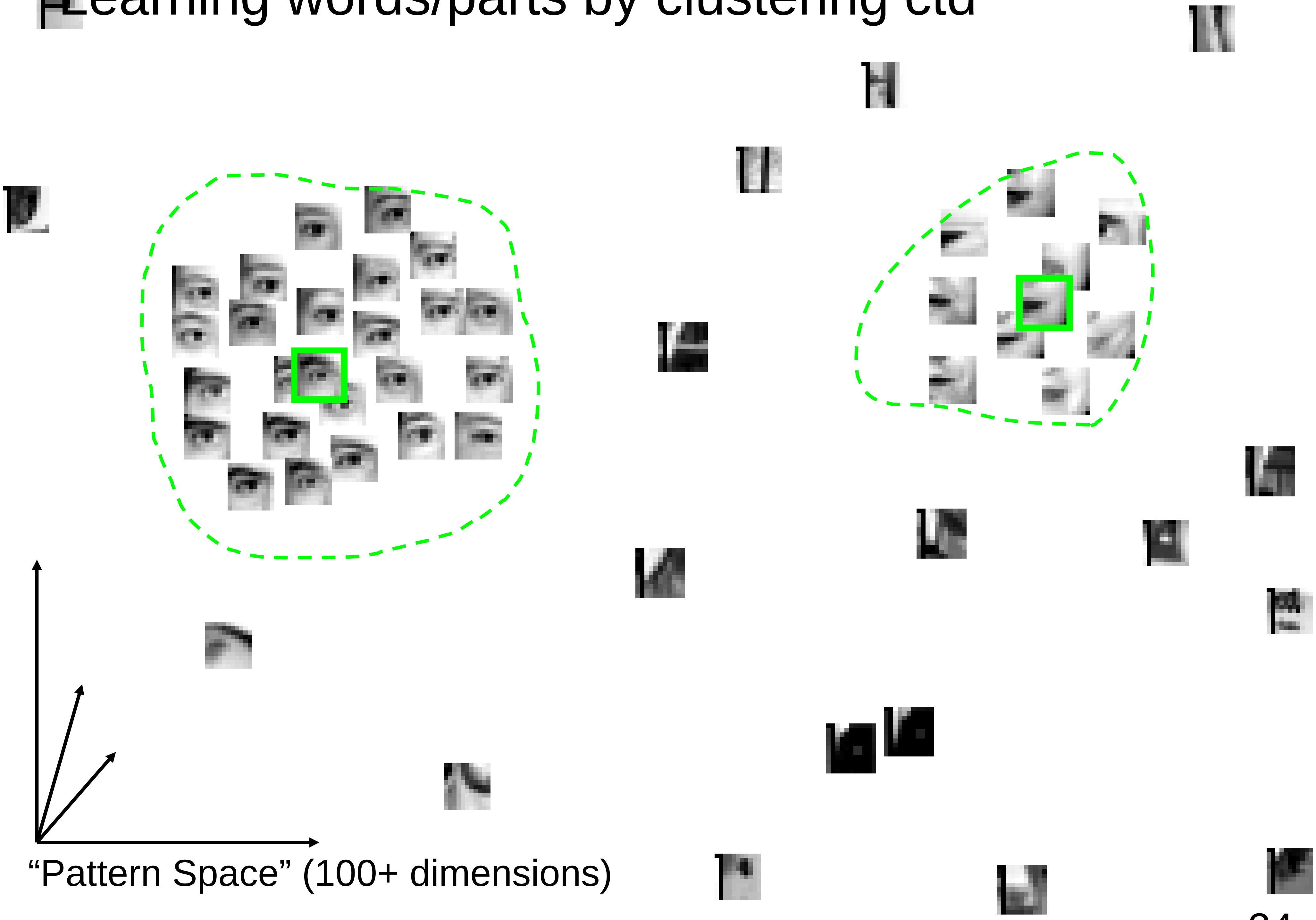
Example: Learn words/parts by clustering



- Interest point features: textured neighborhoods are selected
- produces 100-1000 regions per image

Weber, Welling & Perona 2000

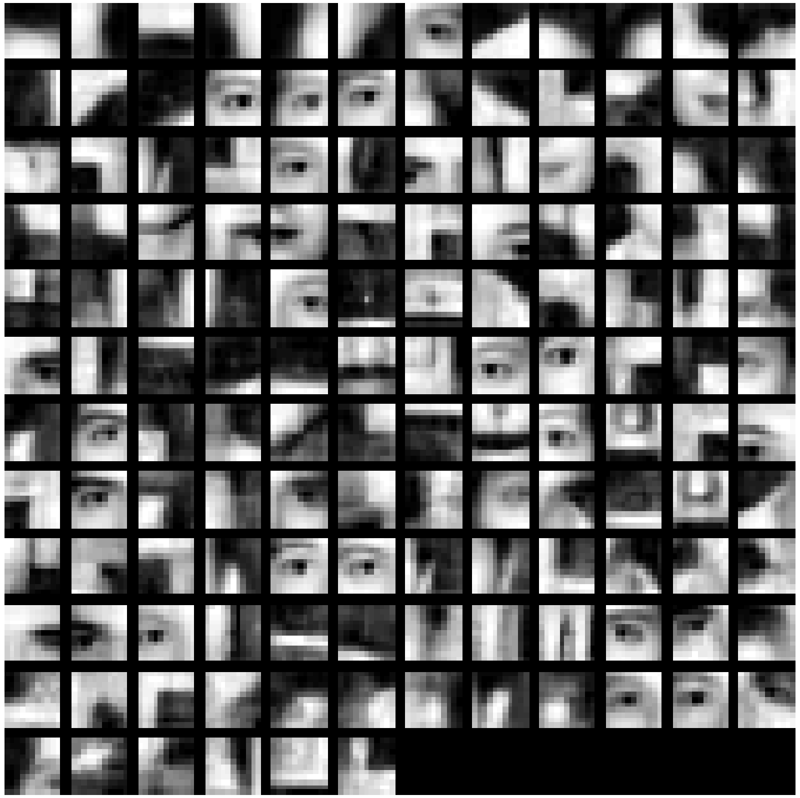
Learning words/parts by clustering ctd



"Pattern Space" (100+ dimensions)

Example of visual words learnt by clustering faces

100-1000 images

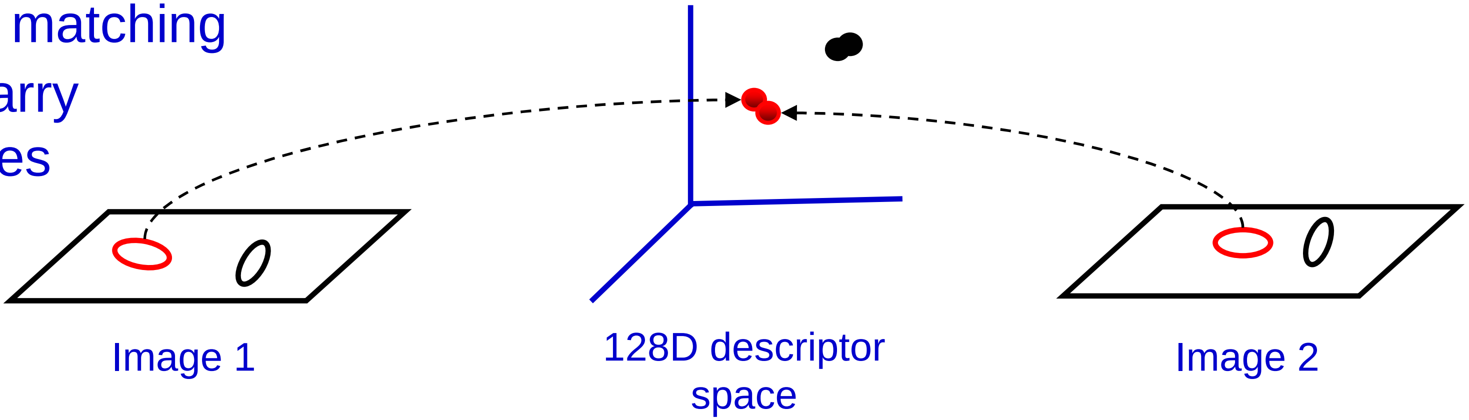


~100 parts

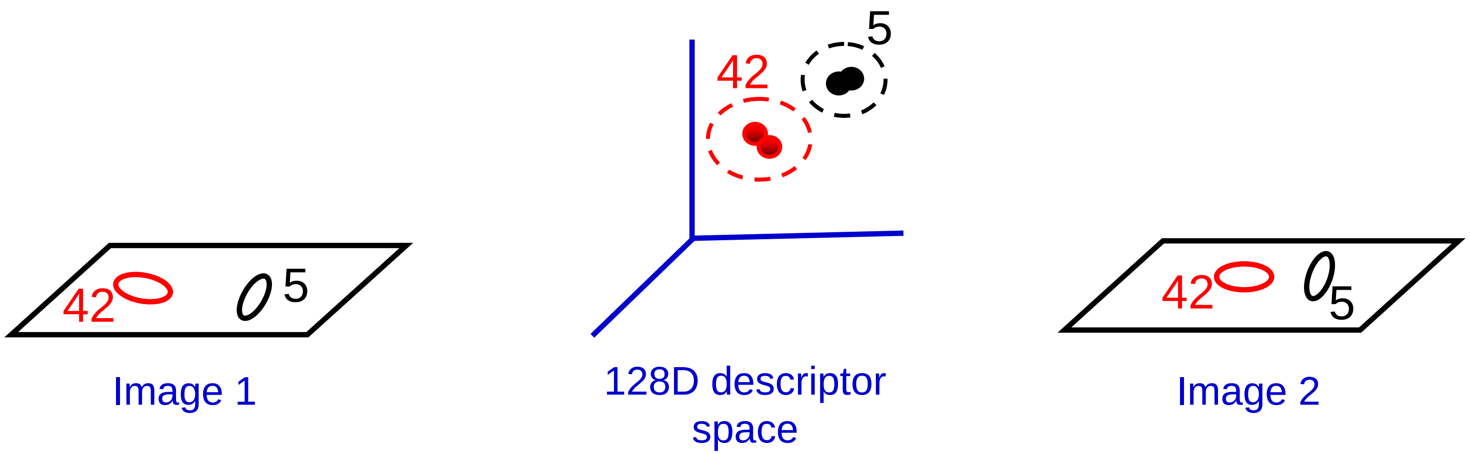
- Apply this representation to image retrieval from a database
- Advantage is that region matches are now pre-computed

Nearest neighbour matching

- expensive to carry out over all frames

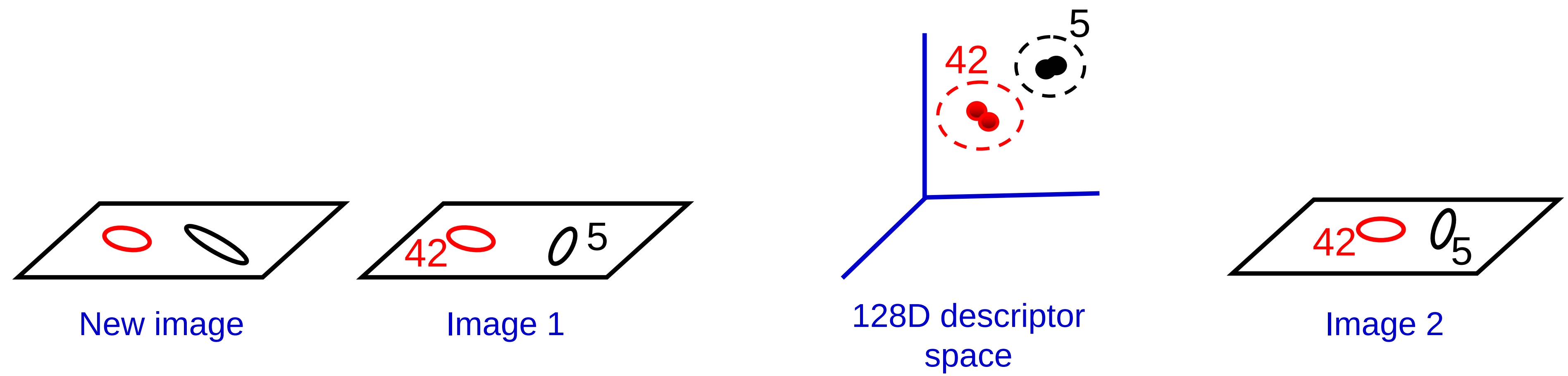


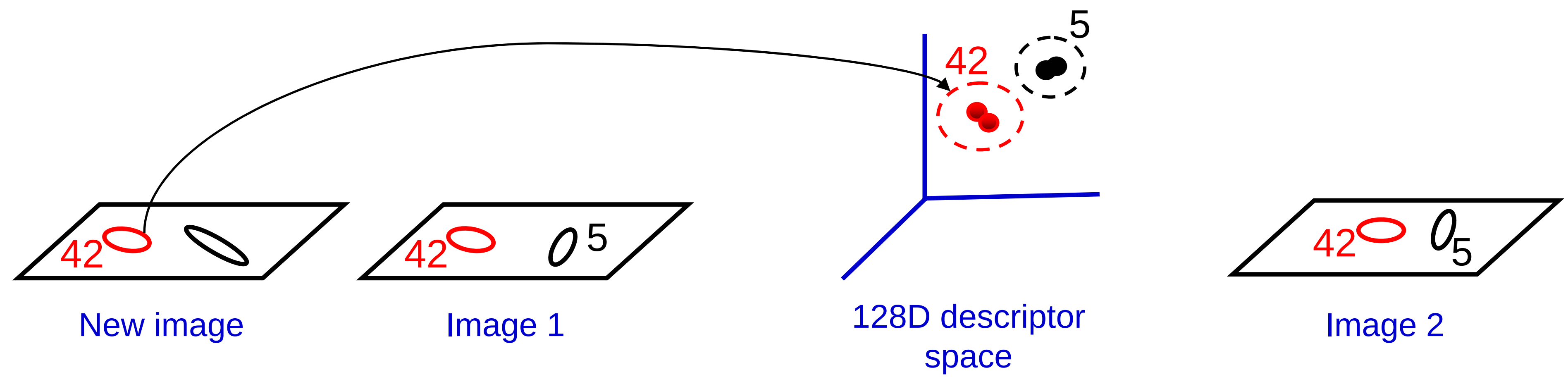
Vector quantize descriptors



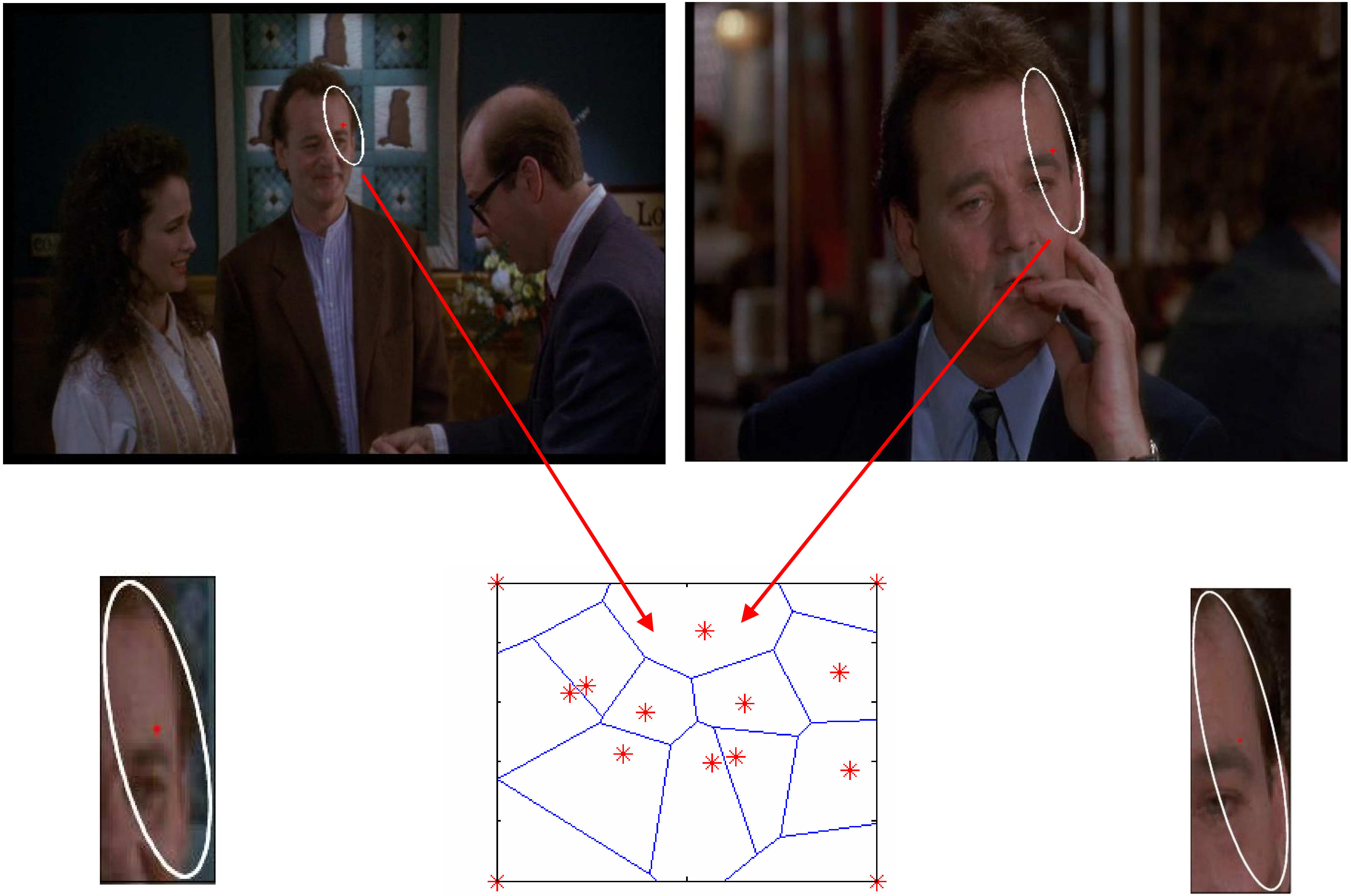
Making the search efficient

Vector quantize descriptors





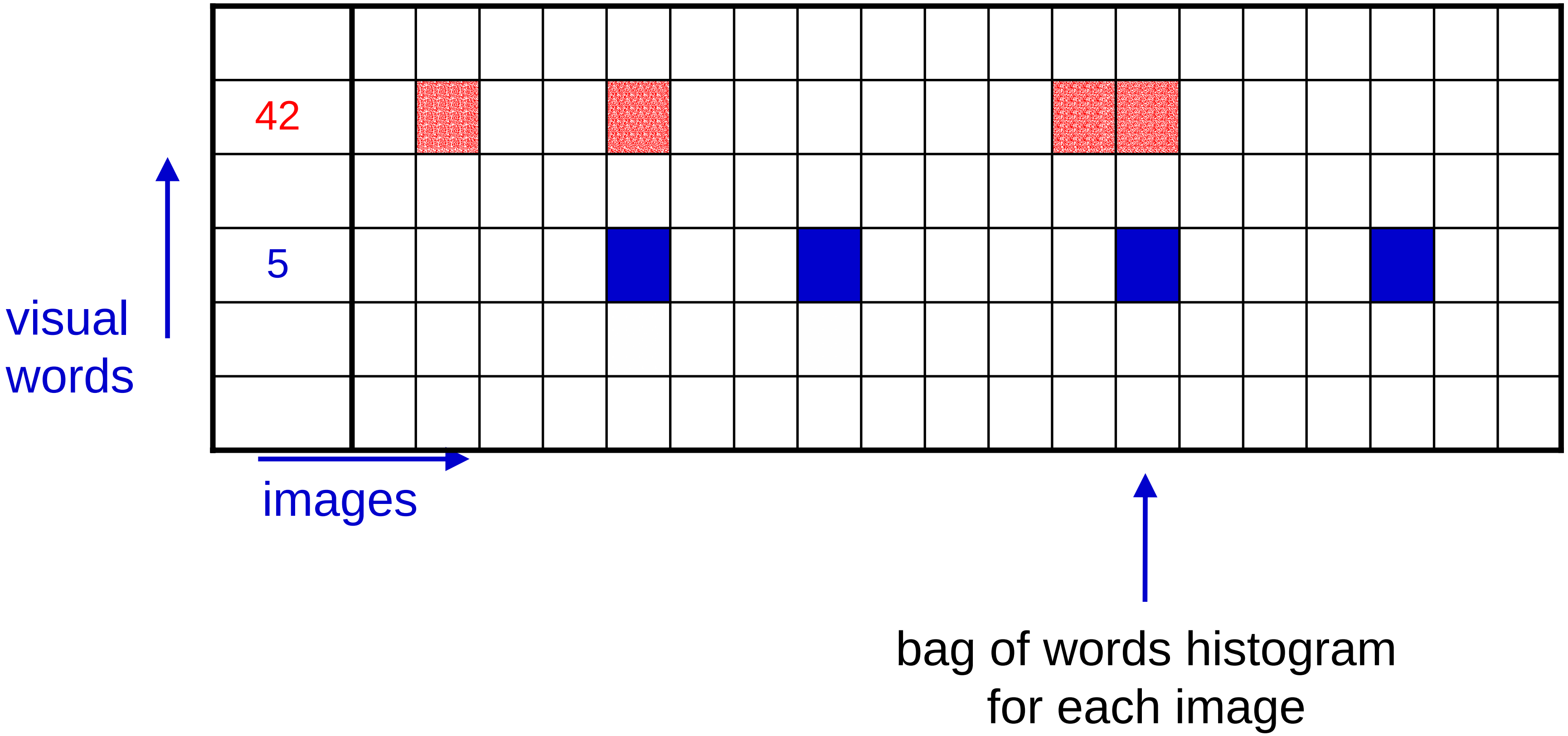
Vector quantize the descriptor space (SIFT)



The same visual word

Making the search efficient

Vector quantize descriptors – discrete set of visual elements “visual words”



i.e. matches have been pre-computed

Application: Efficient "Google like" object retrieval in a large database or a feature length movie

Example :

"Groundhog Day"

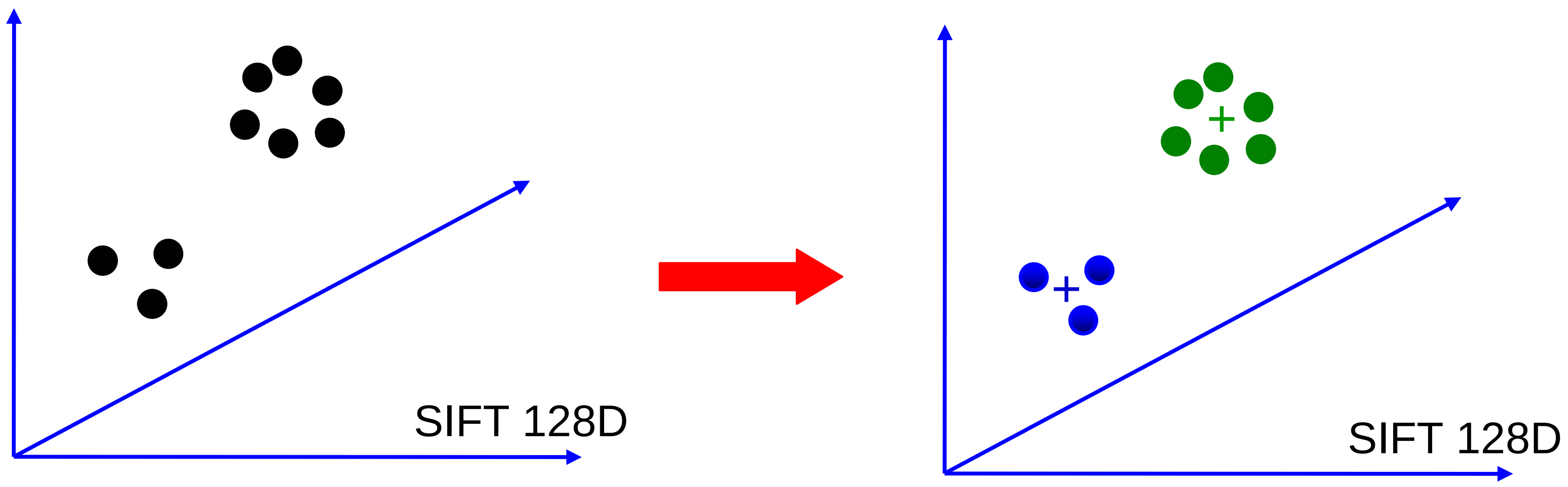
Search feature length movies in 0.1 seconds

- 100K -140K frames, 1000 shots, 5000 keyframes

1. Build a visual vocabulary for the movie

Vector quantize descriptors

• k-means clustering



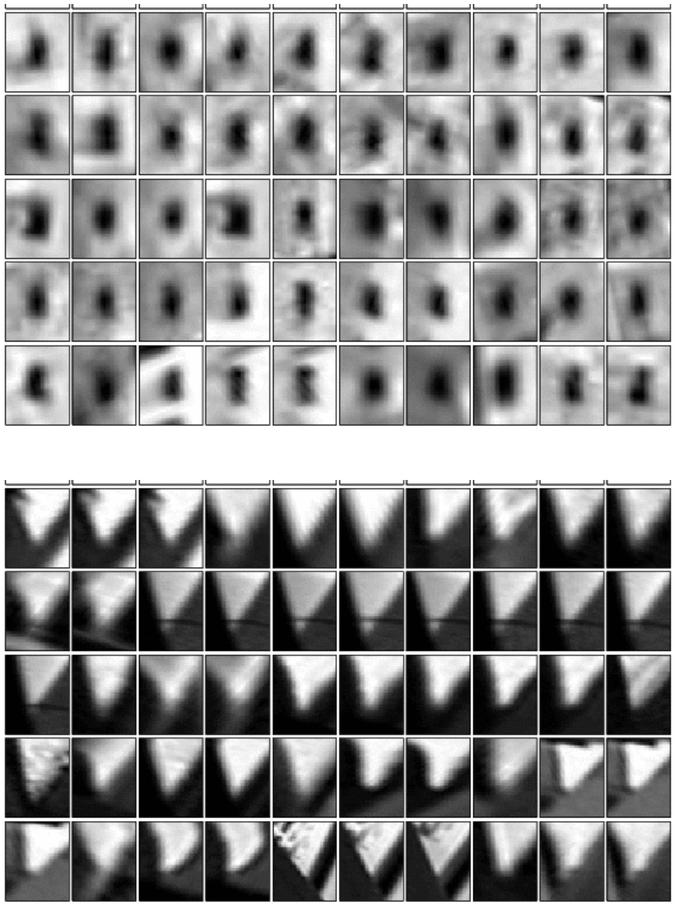
Implementation

• compute SIFT features on frames from 48 shots of the film

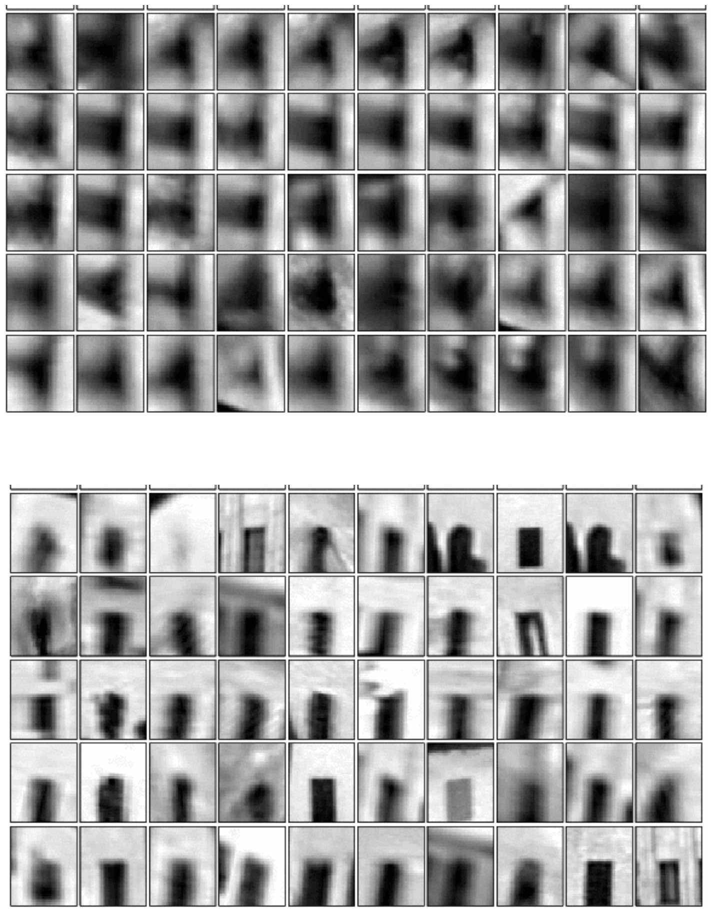
• 6K clusters for Shape Adapted regions

• 10K clusters for Maximally Stable regions

Samples of visual words (clusters on SIFT descriptors):



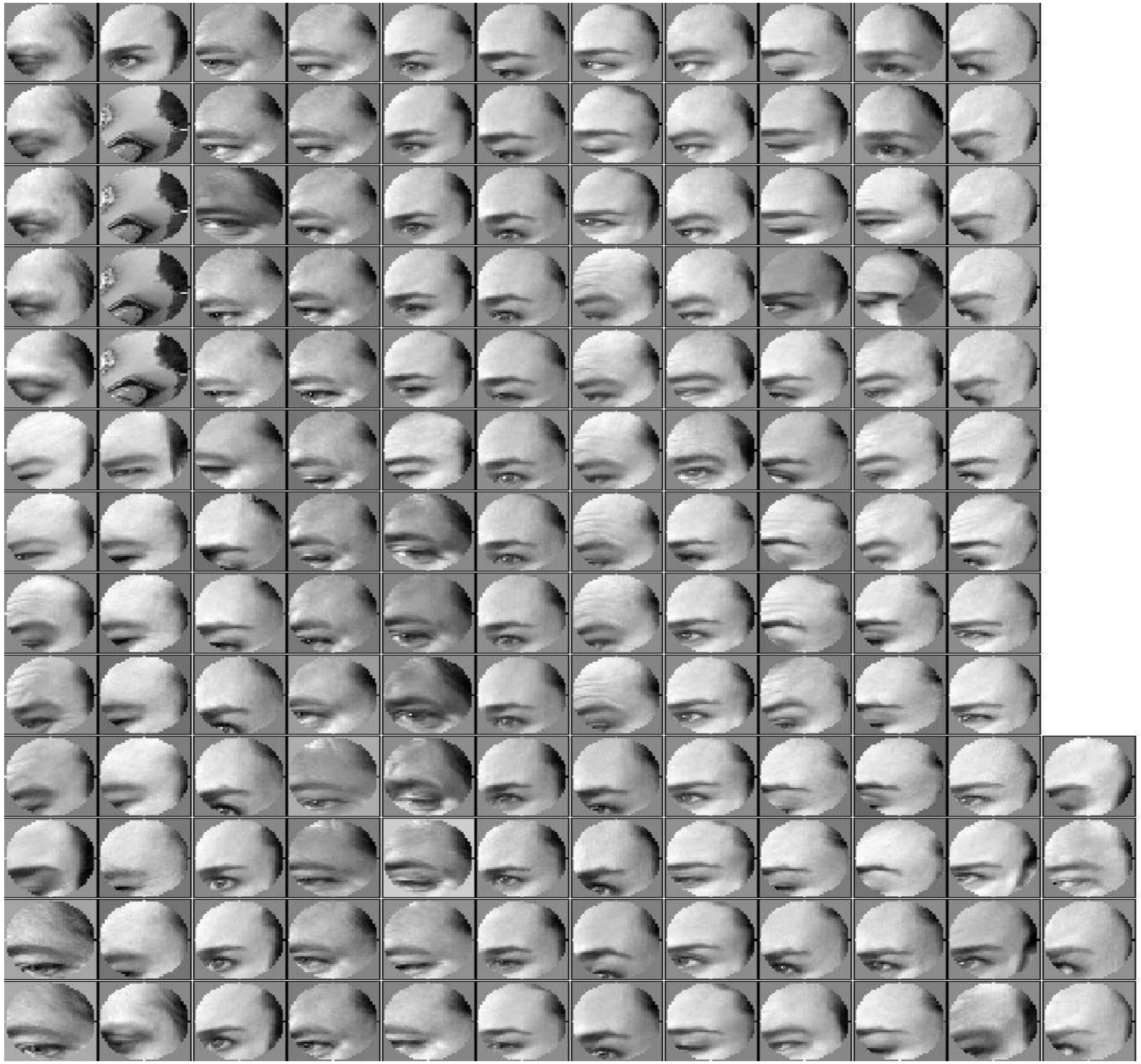
Shape adapted regions

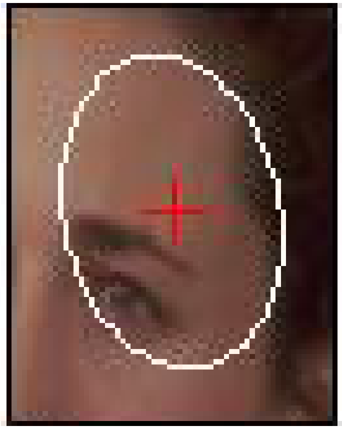


Maximally stable regions

generic examples – cf textons

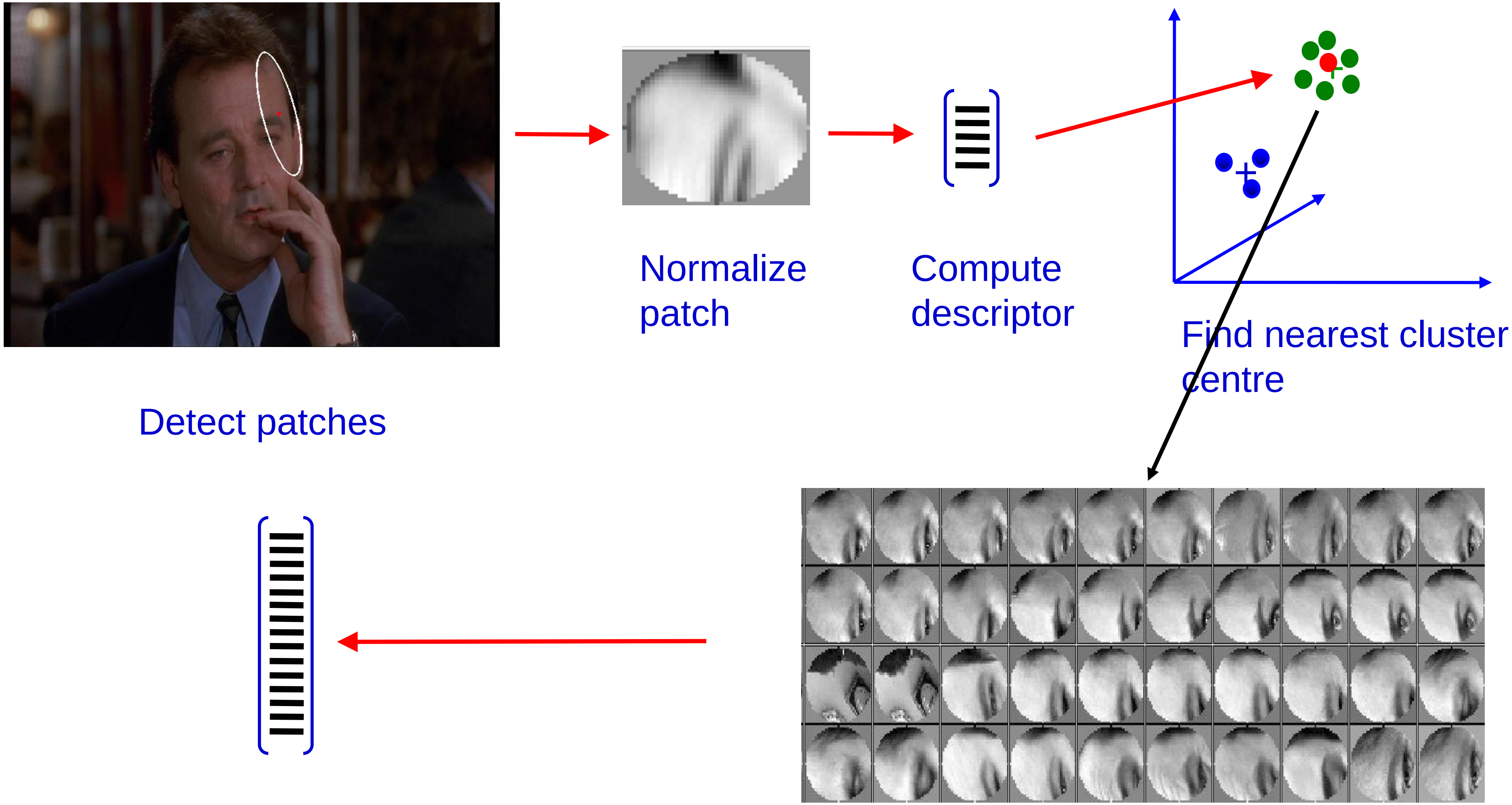
Samples of visual words (clusters on SIFT descriptors):





More specific example

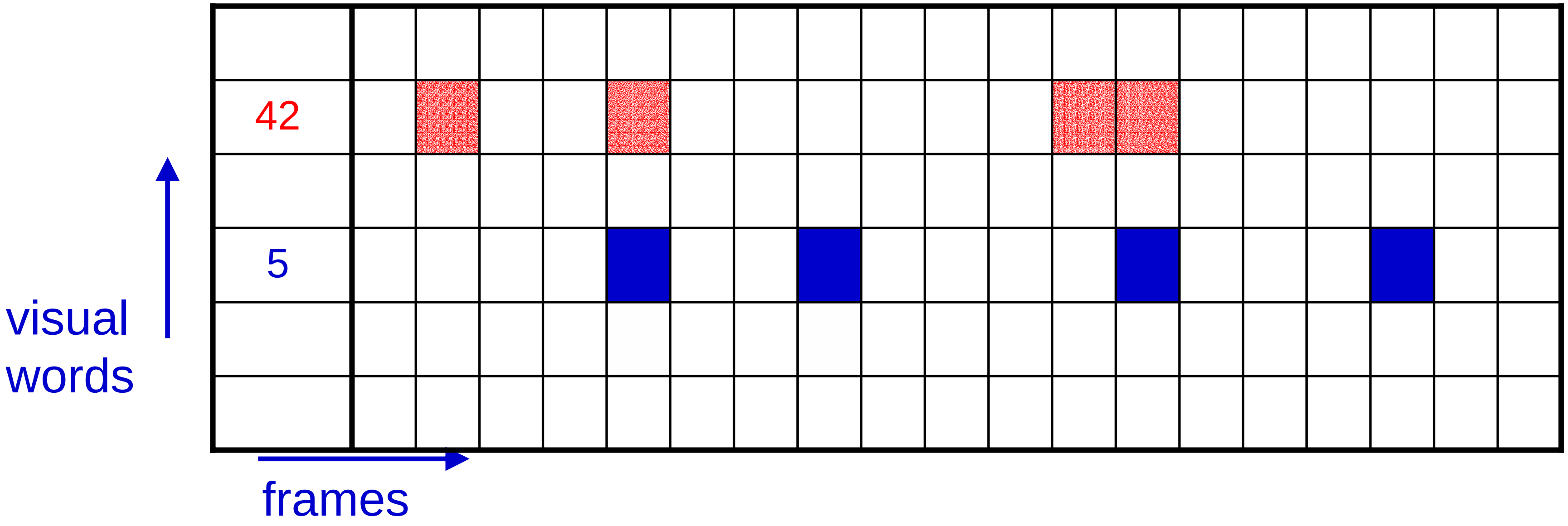
2. Assign words and compute histograms for each key frame in the video



Represent frame by histogram of visual word occurrences

Making the search efficient (Google like retrieval)

Vector quantize descriptors – discrete set of visual elements “visual words”



cf words vs documents (e.g. web pages) in text retrieval

Employ text-retrieval techniques e.g.

- Inverted file indexing
- Ranking (here on spatial consistency)
- Stop-list

Video Google Demo

Example : Groundhog Day

retrieved shots

Video Google, Sivic & Zisserman, ICCV 2003

Searching from other sources

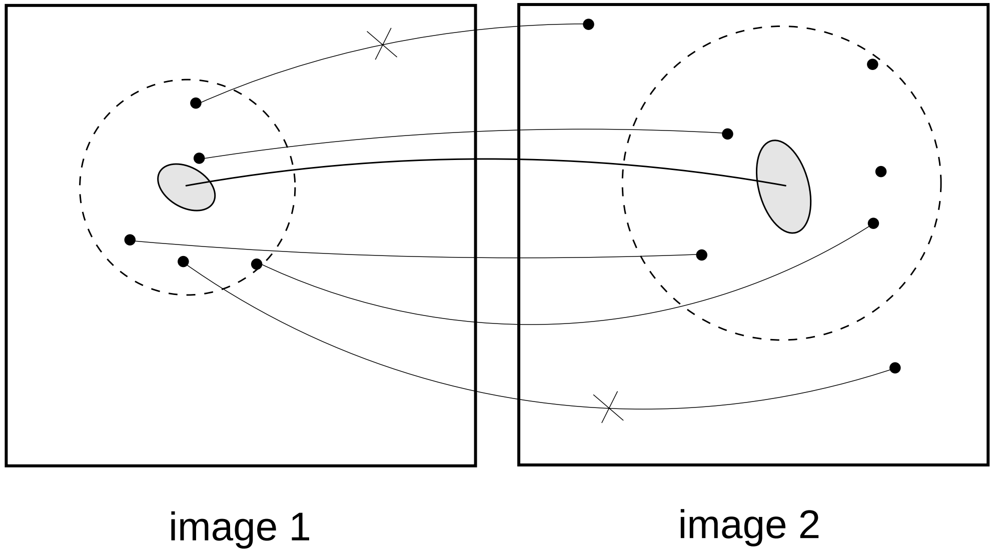
Sony logo

Retrieve shots from Groundhog Day

Retrieved shots in Groundhog Day for search on Sony logo

Object representation

- histogram represents the co-occurrence of visual words
- overlap encodes some structural information



- very weak measure of spatial consistency
- local orderless matching

2. Bag of visual words model II: recognizing object categories

Objectives

- Recognition of visual object classes
- Weakly-supervised learning



Weakly-supervised learning

- Learn model from a set of training images containing object instances

- Know if image contains object or not
- But no segmentation of object or manual selection of features

Visual words

Vector quantize SIFT descriptors to a vocabulary of iconic "visual words".

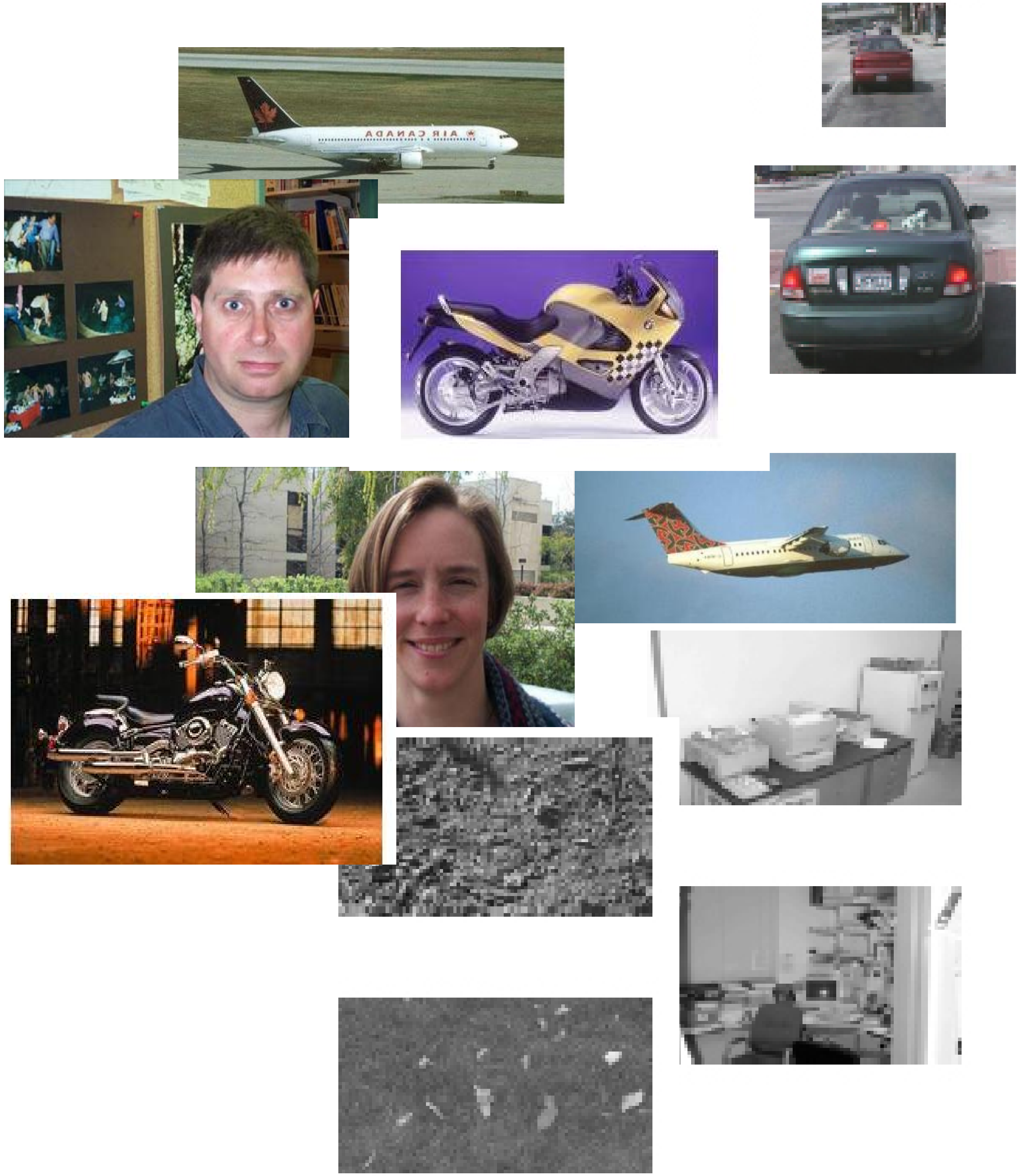
Design of descriptors makes these words invariant to:

- illumination
- affine transformations (viewpoint)

Size (granularity) of vocabulary is an important parameter

- fine grained – represent model instances
- coarse grained – represent object categories

Image collection: four object classes + background



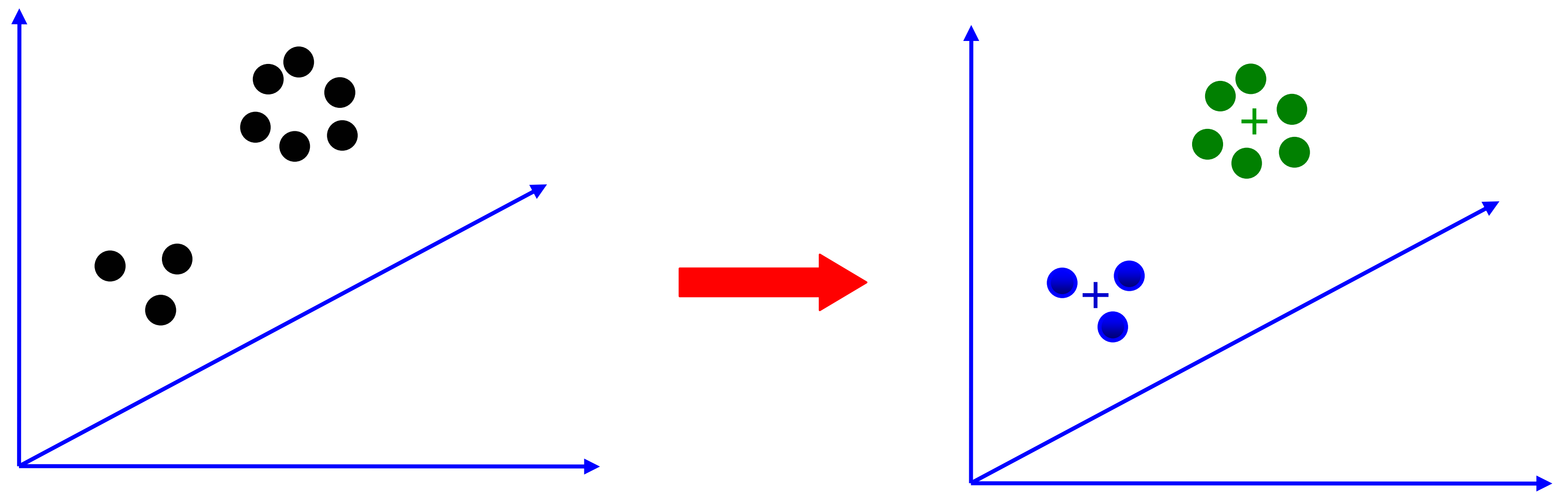
Faces	435
Motorbikes	800
Airplanes	800
Cars (rear)	1155
Background	900
Total:	**4090**

The “Caltech 5”

Building a visual vocabulary

Vector quantize SIFT descriptors

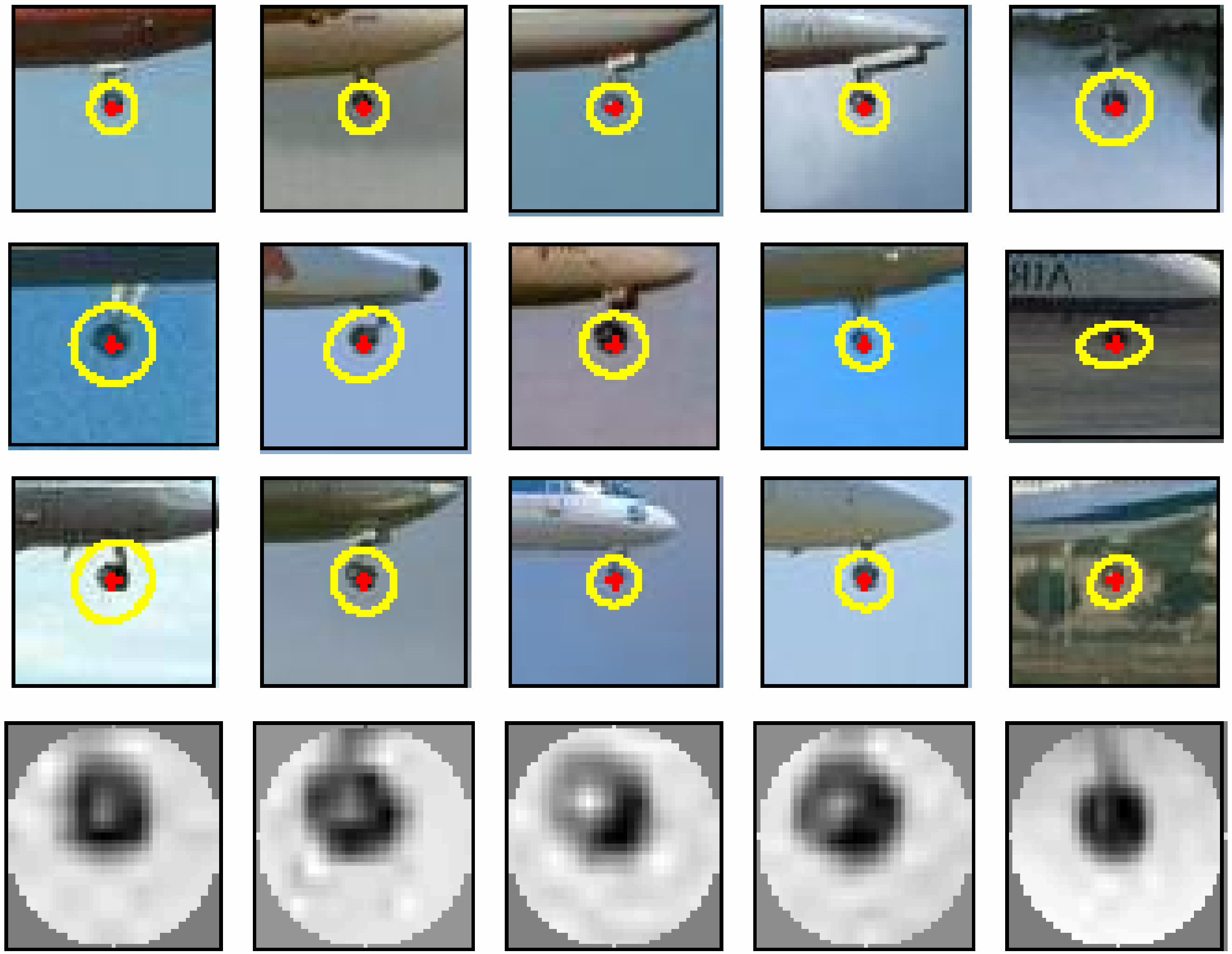
- k-means clustering



Implementation – a vocabulary of about 2K visual words

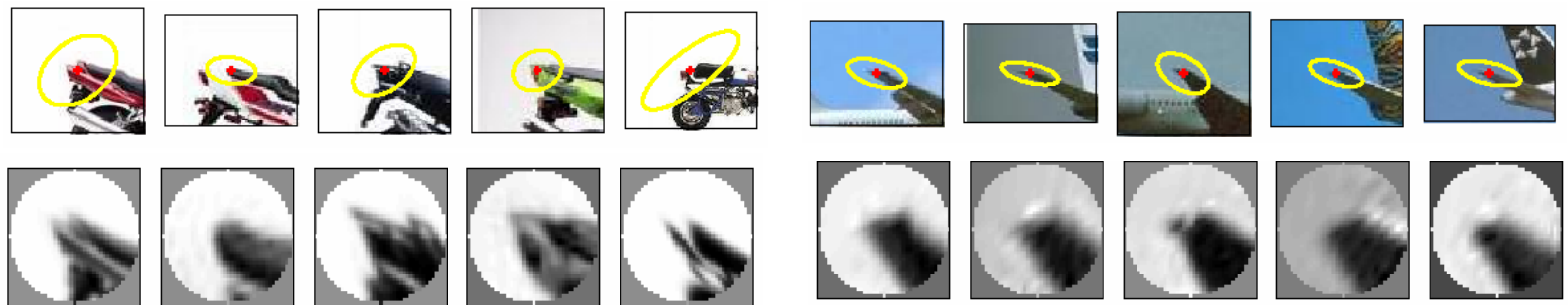
- select random subset of about 1/3rd images of each category
- a total of 300K descriptors

Examples of visual words

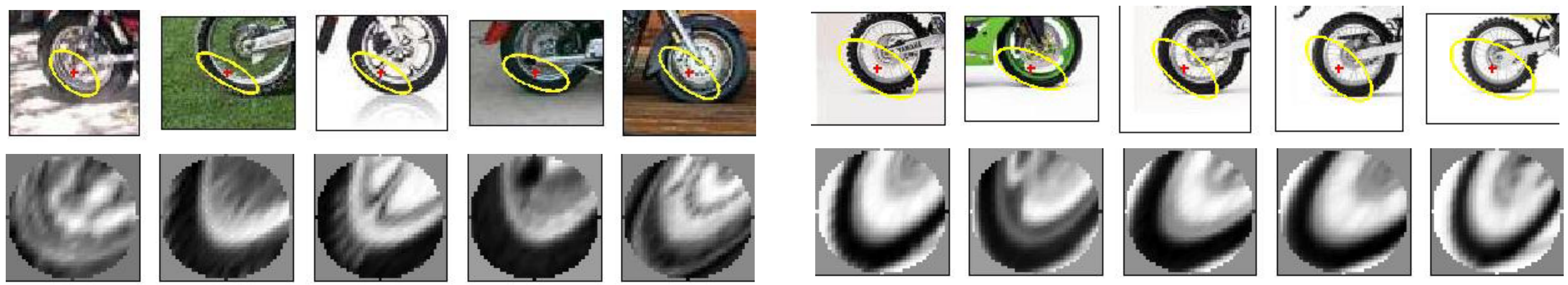


More visual words

Visual synonyms and polysemy

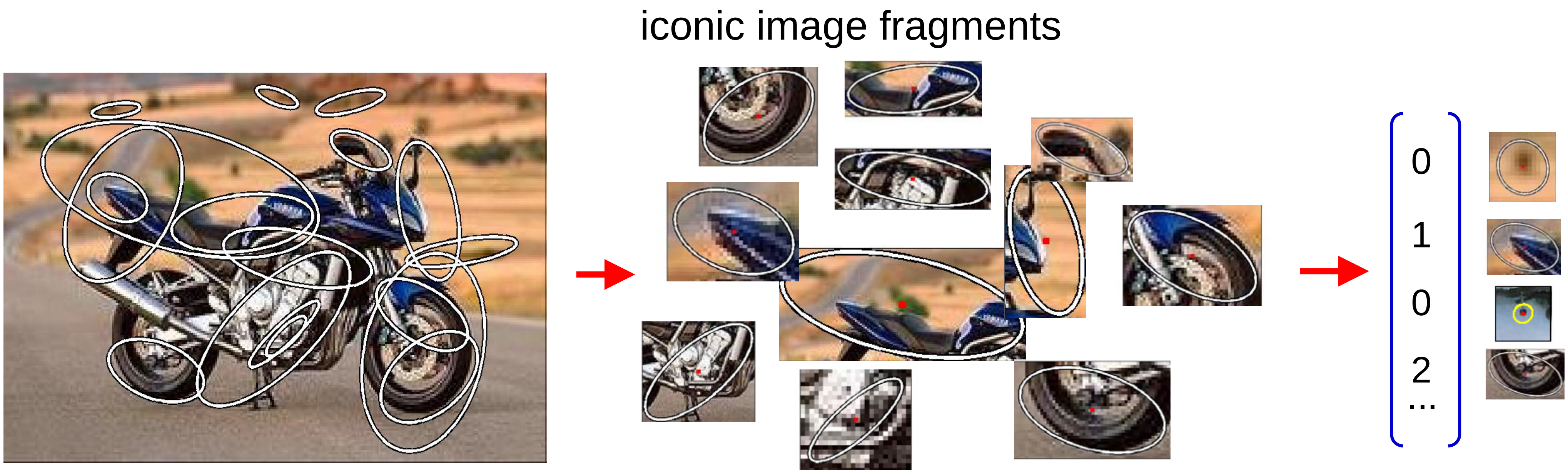


Visual Polysemy: Single visual word occurring on different (but locally similar) parts on different object categories.



Visual Synonyms: Two different visual words representing a similar part of an object (wheel of a motorbike).

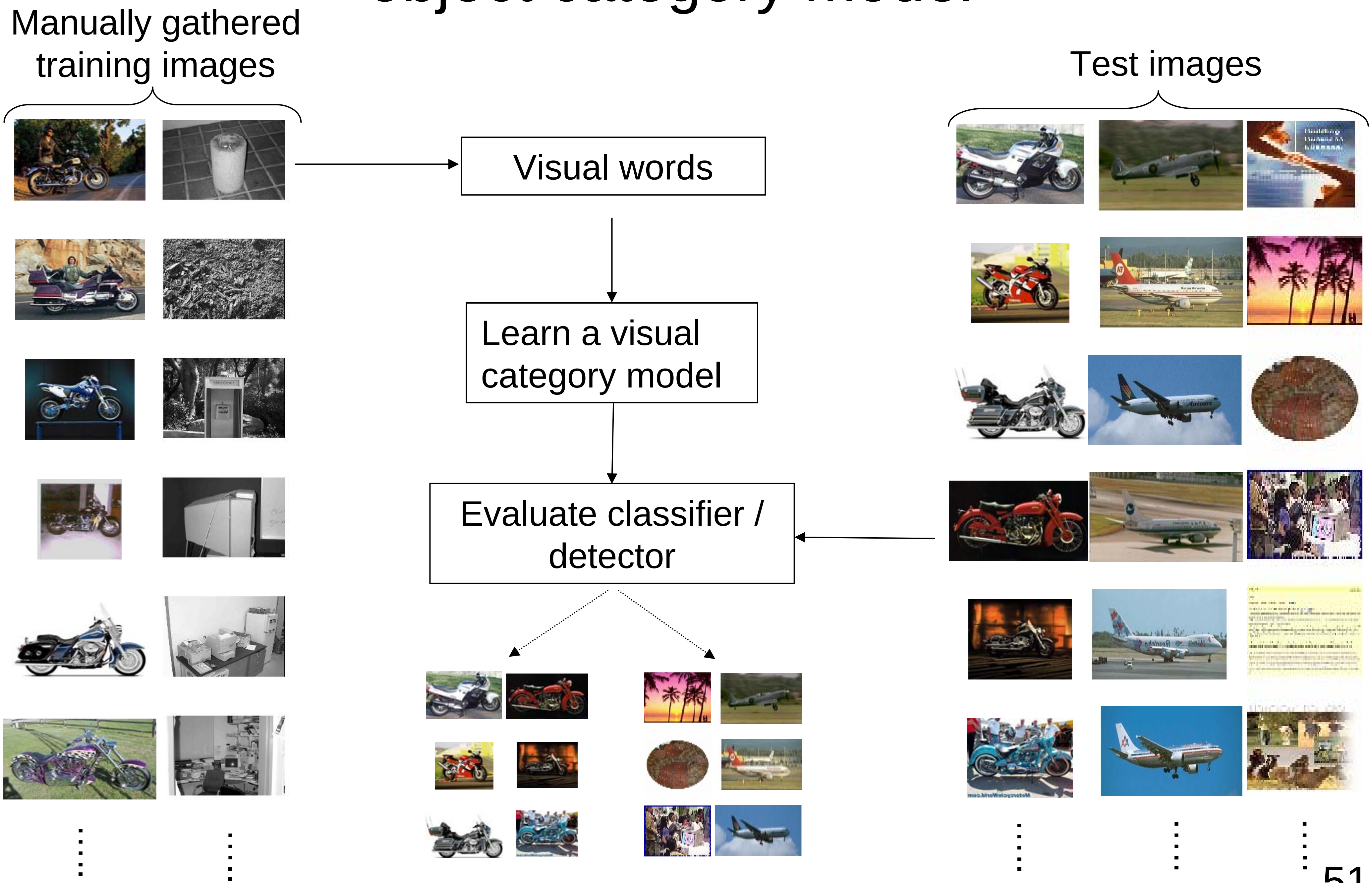
Represent an image as a histogram of visual words



Bag of words model

- Detect affine covariant regions
- Represent each region by a SIFT descriptor
- Build visual vocabulary by k-means clustering (K~1,000)
- Assign each region to the nearest cluster centre

Current Paradigm for learning an object category model



Levels of supervision for training object category model

- Object label + segmentation

[Viola & Jones]

[Agarwal & Roth, Leibe & Schiele, Torralba et al., Winn et al.]

- Object label only

weak supervision

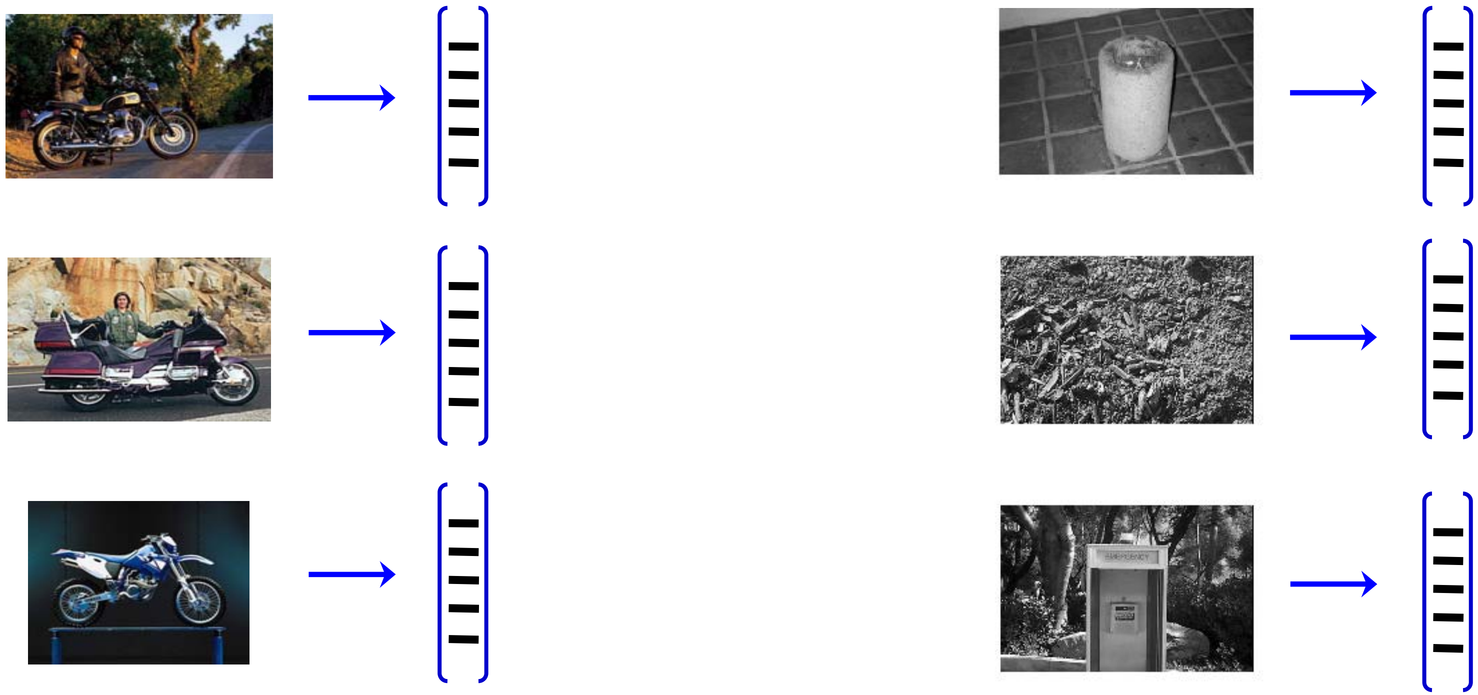
[Csurka et al., Dorko & Schmid, Fergus et al., Opelt et al., Winn and Jojic]

[Barnard et al.]

- None? Images only

Training data: vectors are histograms, one from each training image

positive negative



Train classifier, e.g.

- Naïve Bayes
- SVM

Example: weak supervision

Training
- 50% images
- No identifcation of
  object within image

Motorbikes

Airplanes

Frontal Faces

Testing
- 50% images
- Simple object
  present/absent test

Cars (Rear)

Background

Learning
- SVM classifier
- Gaussian kernel using χ^2 as distance between histograms

Result
- Between 98.3 – 100% correct, depending on class

Zhang et al 2005

The Naïve Bayes Model

$$p(C|w_1, w_2 \ldots, w_n) \;\alpha\; p(C)p(w_1, w_2 \ldots, w_n|C)$$

Prior prob. of
the object classes

Image likelihood
given the class

$$\alpha \; p(C) \prod_{i=1}^{n} p(w_i|C)$$

independence
assumption

Image classification decision

$$C^* = \arg\max_C p(C) \prod_{i=1}^{n} p(w_i|C)$$

The Naïve Bayes Model – implementation

$$p(w_i|C)$$

compute as sum over positive (negative) histogram bins in training data

$$\prod_{i=1}^{n} p(w_i|C) = \prod_{i=1}^{V} p(w_i|C)^{n_i}$$

bin count for word i in testing data

n: number of regions detected

V: size of vocabulary

Image classification decision – ratio of posteriors

$$\ln \frac{p(\text{object}|w_1,\ldots,w_n)}{p(\text{background}|w_1,\ldots,w_n)} \begin{cases} > 0 & \text{object} \\ < 0 & \text{background} \end{cases}$$

Comparison on the CalTech5 database

Categories	[Zhang et al ’05]	[Csurka et al ’04]	[Fergus et al ’03]
Airplanes	98.8	97.1	90.2
Cars(rear)	98.3	98.6	90.3
Faces	100	99.3	96.4
Motorbikes	98.5	98.0	92.5

SVM classifier

naïve Bayes classifier

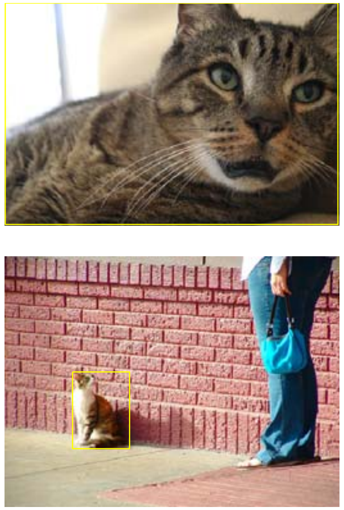
PASCAL Visual Object Classes Challenge 2006

Bicycle

Bus

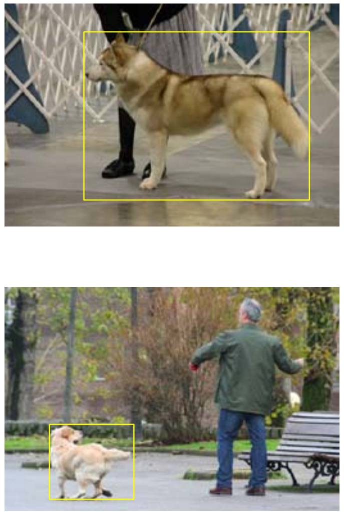
Car

Cat

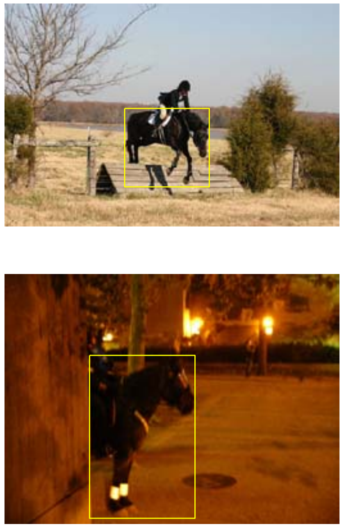


Cow

Dog

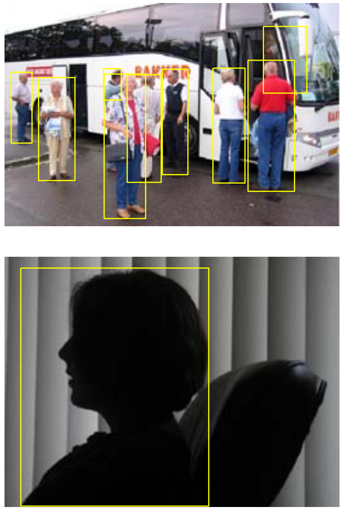


Horse

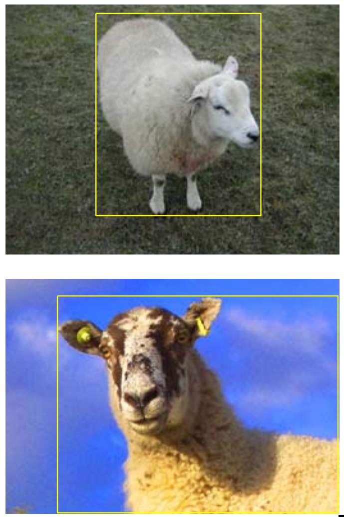


Motorbike

Person



Sheep

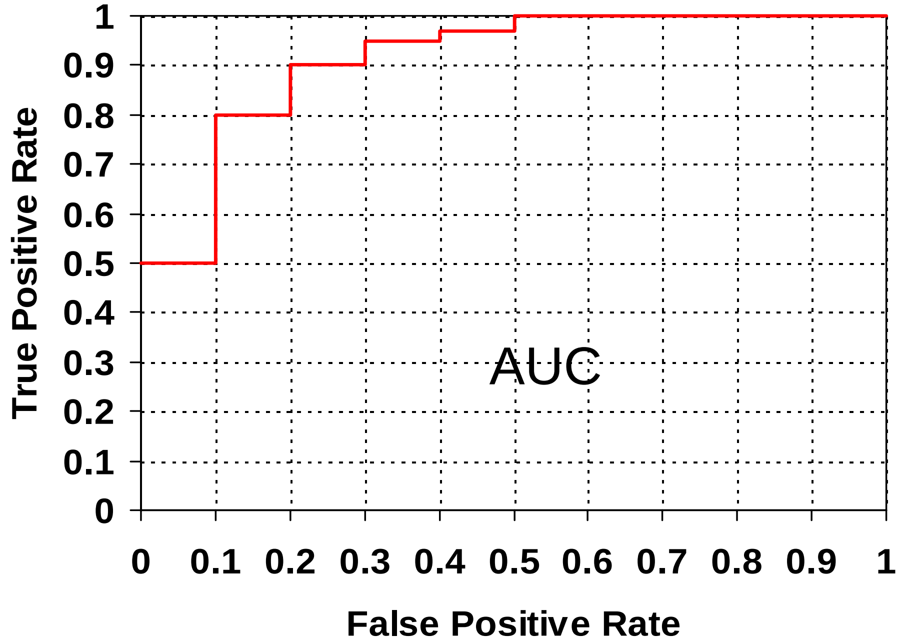


Dataset Statistics

	train		val		trainval		test	
	img	**obj**	**img**	**obj**	**img**	**obj**	**img**	**obj**
Bicycle	127	161	143	162	270	323	268	326
Bus	93	118	81	117	174	235	180	233
Car	271	427	282	427	553	854	544	854
Cat	192	214	194	215	386	429	388	429
Cow	102	156	104	157	206	313	197	315
Dog	189	211	176	211	365	422	370	423
Horse	129	164	118	162	247	326	254	324
Motorbike	118	138	117	137	235	275	234	274
Person	319	577	347	579	666	1156	675	1153
Sheep	119	211	132	210	251	421	238	422
Total	1277	2377	1341	2377	2618	4754	2686	4753

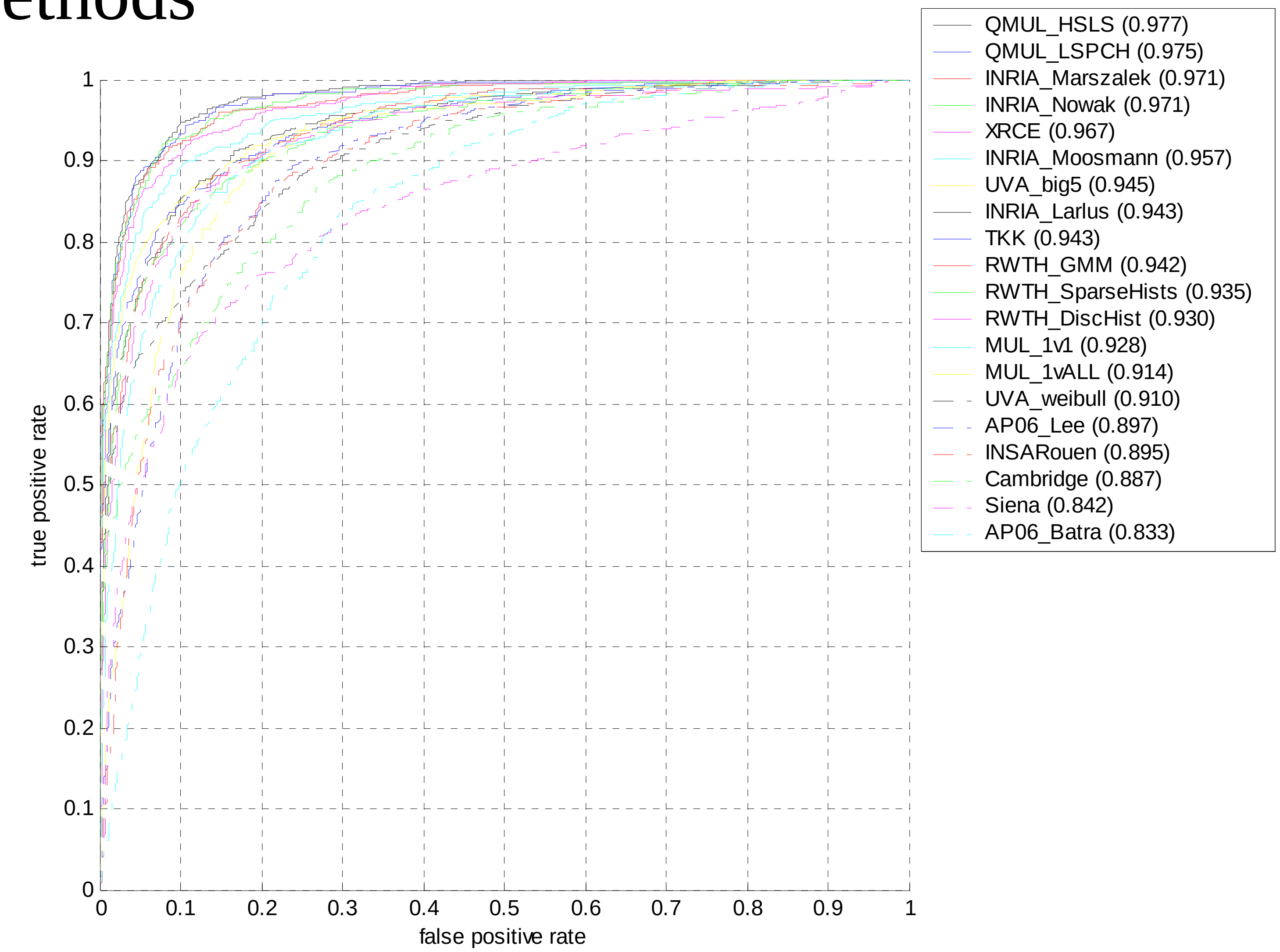
Evaluation

- Receiver Operating Characteristic (ROC)
  - Area Under Curve (AUC)



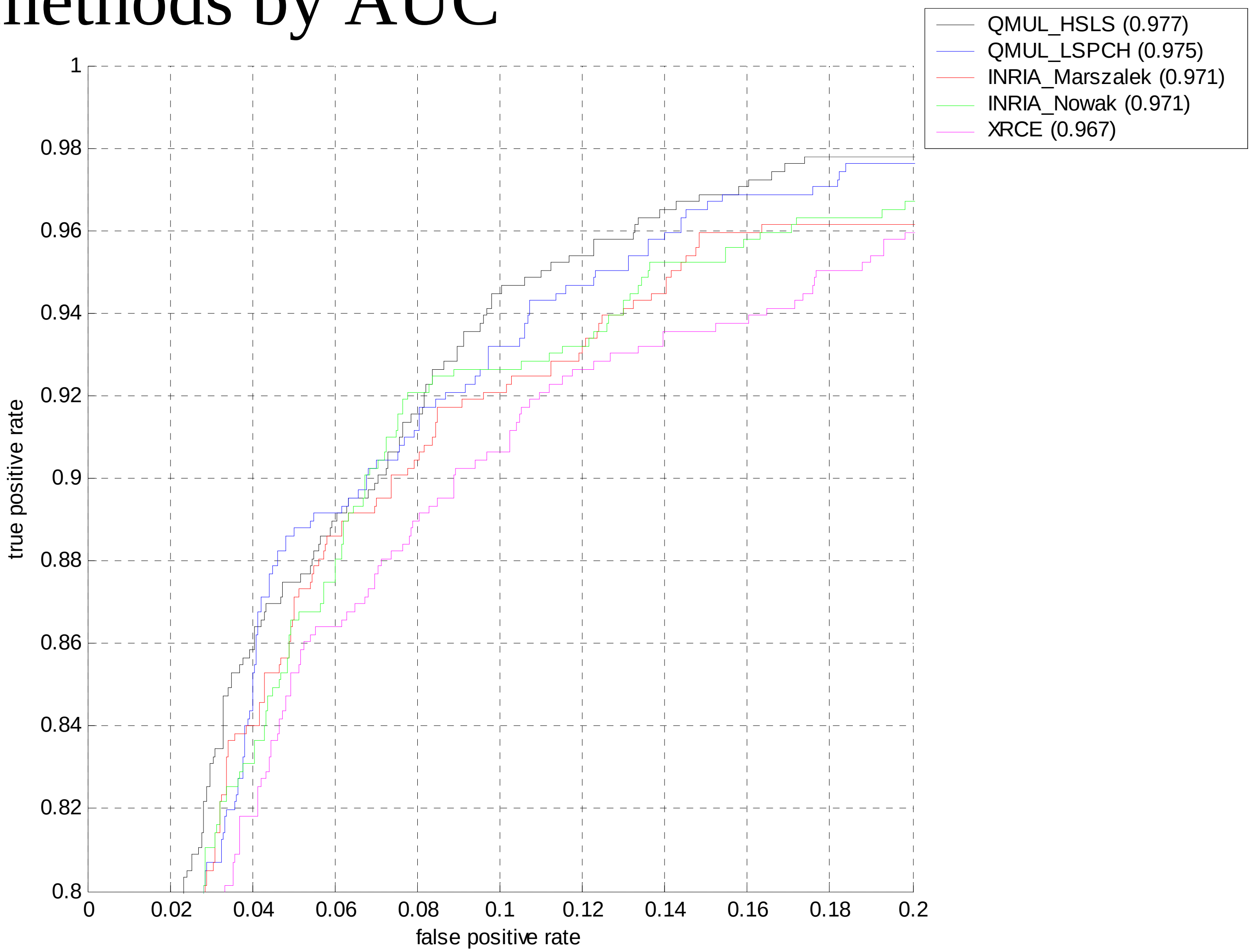
Competition 1: Car

- All methods



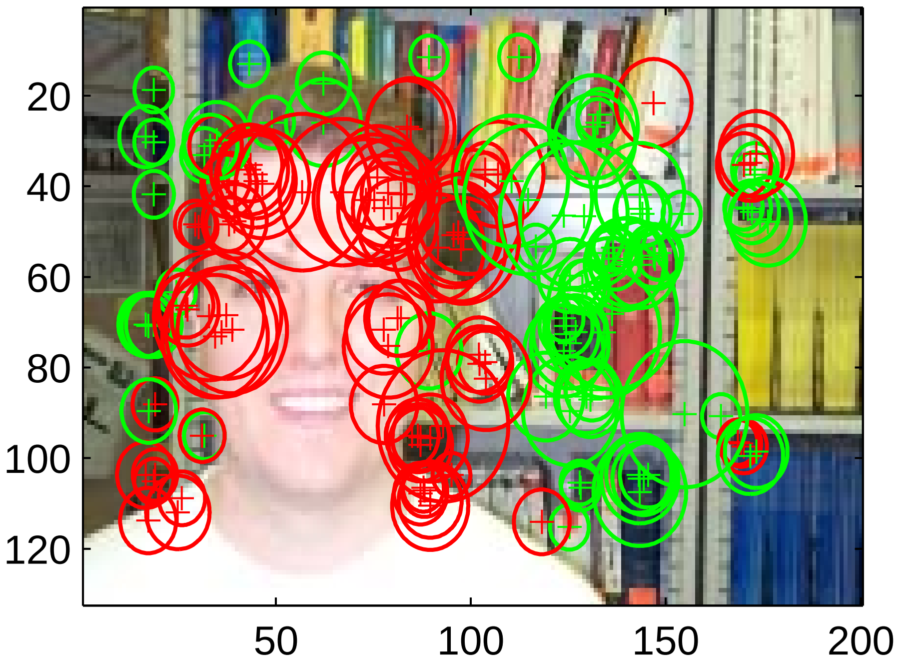
Competition 1: Car

- Top 5 methods by AUC

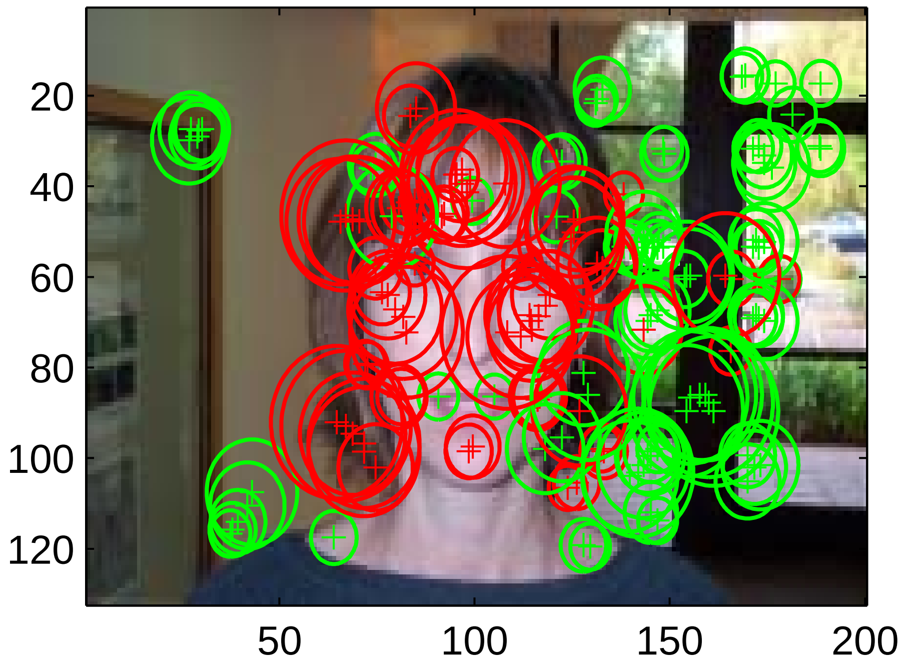


Localization according to visual word probability

Naïve Bayes



sparse segmentation



foreground word more probable

background word more probable

Summary

- Universal vocabulary over all classes

- Bag of visual words model:
  - Learns and uses co-occurrence of visual words
  - Very successful in classifying images according to the objects they contain
  - Still requires further testing for large changes in scale and viewpoint
  - No explicit use of configuration of visual word positions
  - Poor at localizing objects within an image

Outline

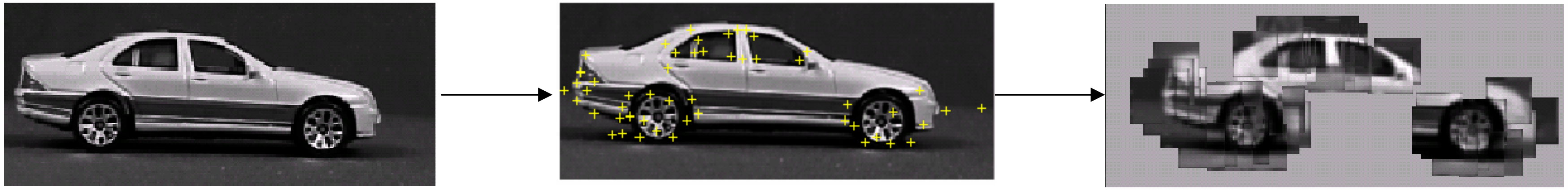
1. Bag of visual words model I: recognizing particular objects
    - Vector quantization to get visual vocabulary (parts)
    - Video Google retrieval algorithm
2. Bag of visual words model II: recognizing object categories
    - Learn classifier for image according to the object it contains
    - Naïve Bayes and SVM classifiers
3. Models of parts and structure
    - Implicit and explicit geometric configurations
4. Class based segmentation
    - Pixel level localization
5. Summary and open challenges

3. Models of parts and structure

- implicit configuration models
  - Leibe & Schiele, Agarwal & Roth
- explicit configuration models
  - Fergus et al, Crandall et al

Leibe & Schiele 2003/2004

- Extraction of local object patches
  - Interest Points (Harris detector)



- Example: training set of 160 car images
  - 16 views of 10 cars
  - results in 8'269 training patches

Visual Object Categorization

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Perceptual Computing Computer Vision

Visual Vocabulary (Codebook Entries)

- **Visual Clustering procedure**
  - agglomerative clustering: most similar clusters are merged (t > 0.7)

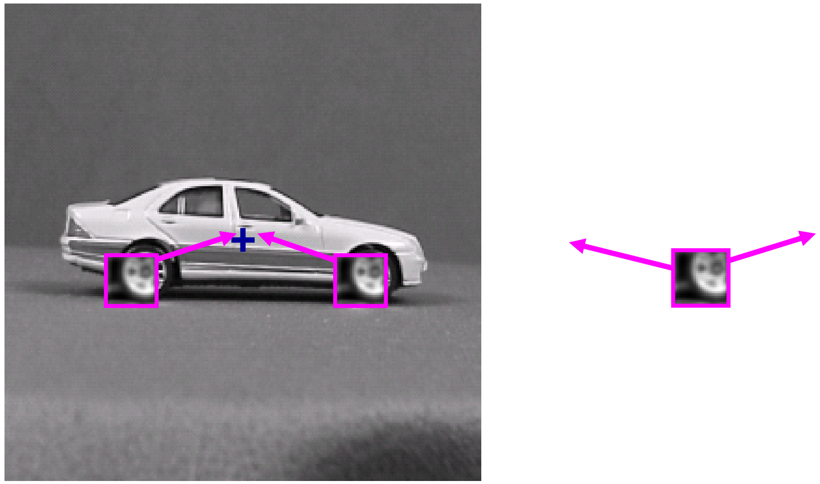
$$similarity(C_1, C_2) = \frac{\sum_{p \in C_1, q \in C_2} NGC(p,q)}{|C_1| \times |C_2|} > t$$

$$NGC(p,q) = \frac{\sum_i (p_i - \overline{p_i})(q_i - \overline{q_i})}{\sqrt{\sum_i (p_i - \overline{p_i})^2 \sum_i (q_i - \overline{q_i})^2}}$$

- **Examples (from 2519 codebook entries)**
  - visual similarity preserved
  - wheel parts, window corners, fenders, ...

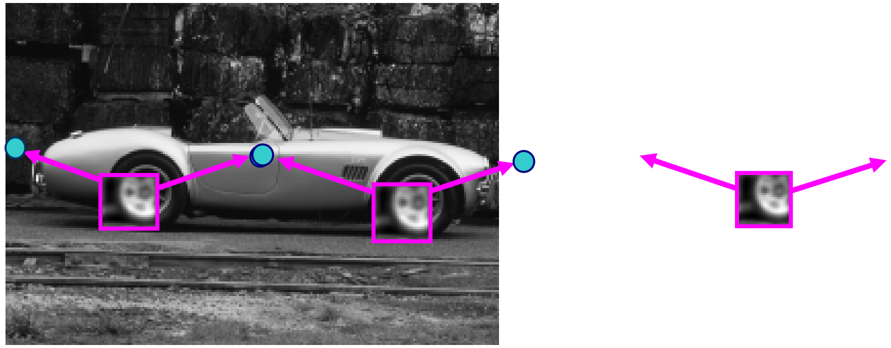
Structure: Generalized Hough Transform

- **Learning:** For every cluster, store possible “occurrences”



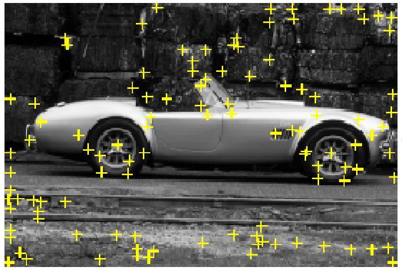
  - Object Identity
  - Pose
  - Relative position

- **Recognition:** For new image, let the matched patches vote for possible object positions

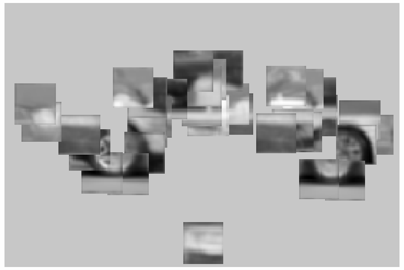


Object Categorization Procedure

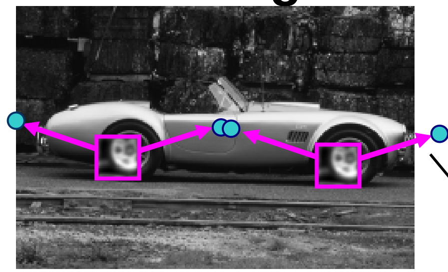
Interest Points

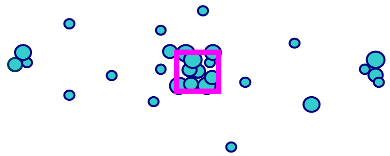


Matched Codebook Entries

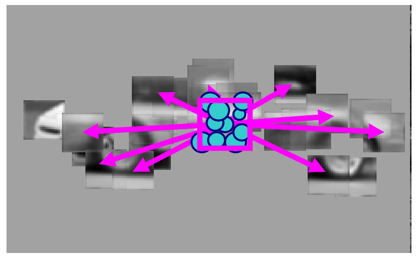


Probabilistic Voting

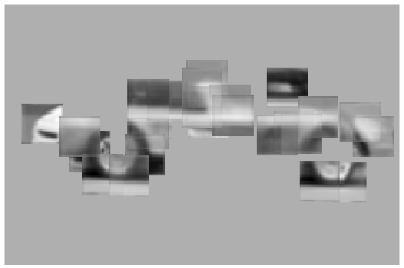




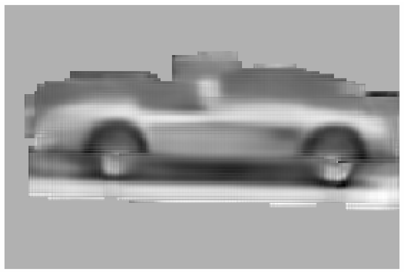
Voting Space (continuous)



Backprojection of Maximum



Backprojected Hypothesis

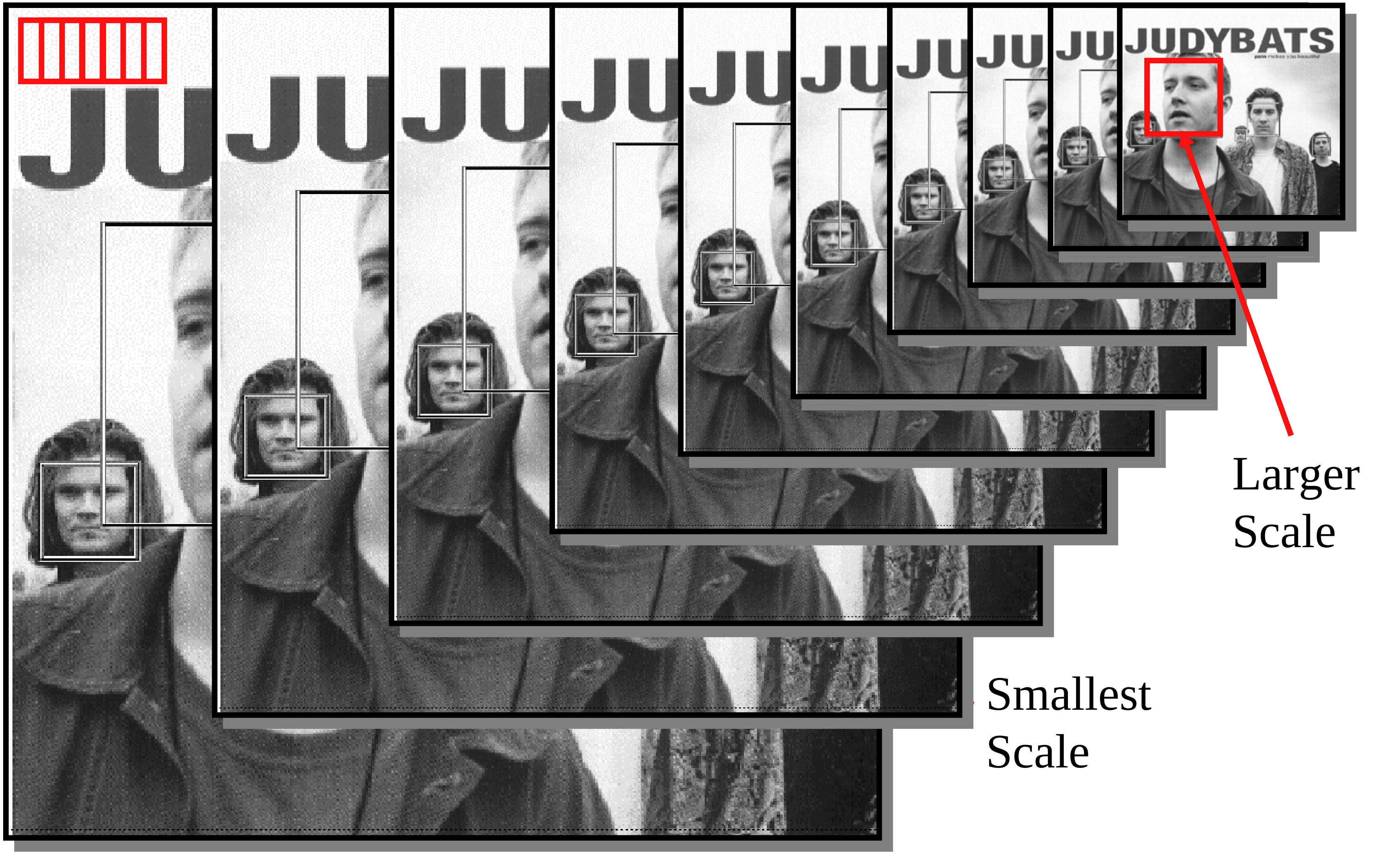


Refined Hypothesis (uniform sampling)

Detection Results

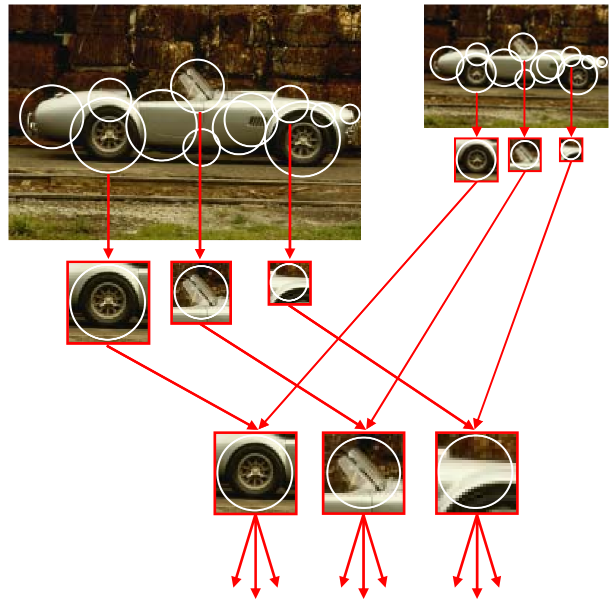
- Qualitative Performance
  - Recognizes different kinds of cars
  - Robust to clutter, occlusion, low contrast, noise

(1) search over scale



(2) Feature detector determines position *and* scale

Leibe & Schiele extension: Scale Invariance

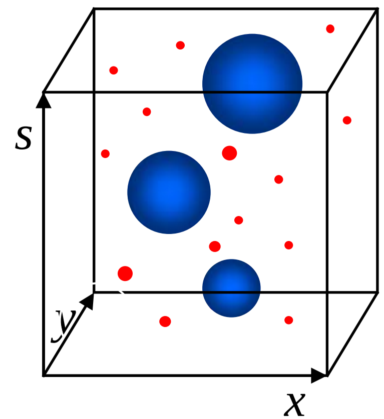


- Scale-invariant feature selection
  - Scale-invariant interest points
  - Rescale extracted patches
  - Match to constant-size codebook

- Generate scale votes
  - Scale as 3rd dimension in voting space

$$
\begin{aligned}
x_{vote} &= x_{img} - x_{occ}(s_{img}/s_{occ}) \\
y_{vote} &= x_{img} - y_{occ}(s_{img}/s_{occ}) \\
s_{vote} &= (s_{img}/s_{occ})
\end{aligned}
$$

  - Search for maxima in 3D voting space



Search window

Qualitative Detection Results

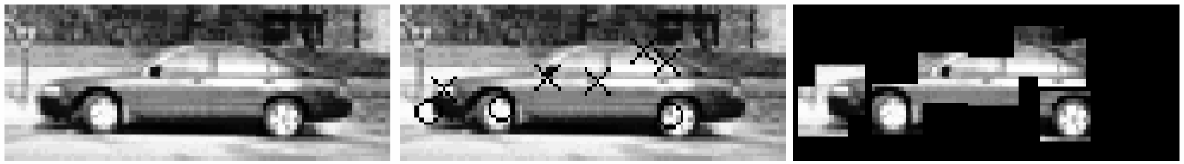
scale = 0.75

scale = 3.71

Altogether, objects detected with factor 5.0 scale differences

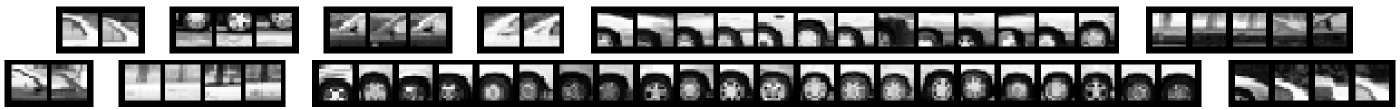
Agarwal & Roth 2002

- Interest points detected



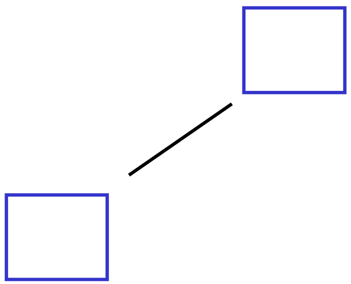
- Extracted fragments from training images

- Clustered Fragments (Dictionary) – 270 parts



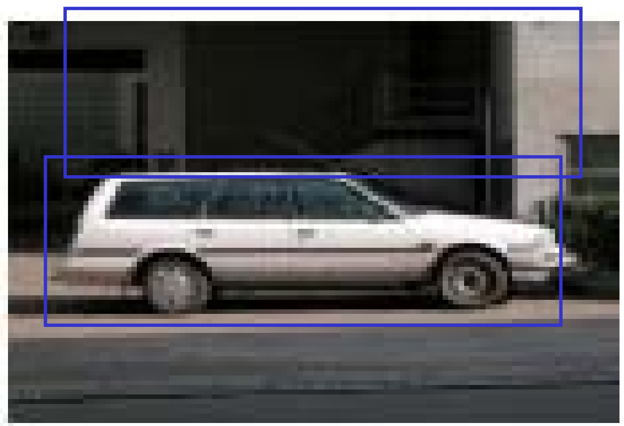
Learning: Structure

- Representation: binary feature vector
- Feature vector components
  - Part present/absent (270)
  - Pair wise relation between parts (20 of these for each pair)



Coarse representation of:

- angles (4 bins)
- distance (5 bins)

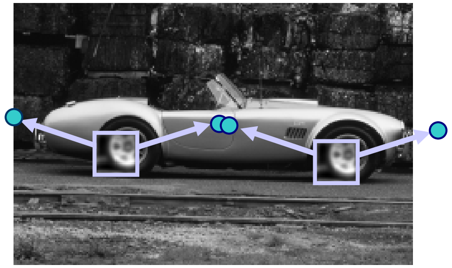


Use sliding window to measure feature vectors from positive and negative examples

Recognition

- Detect parts
- Apply sliding window
- Linear classifier on feature vector for window
- Use SNoW (Sparse network of Winnows)
    - suited to very large, very sparse vectors

Comparison with Leibe & Schiele



Agarawal & Roth:

- looser geometric relations
- more tolerant of structure deformation

Explicit structure

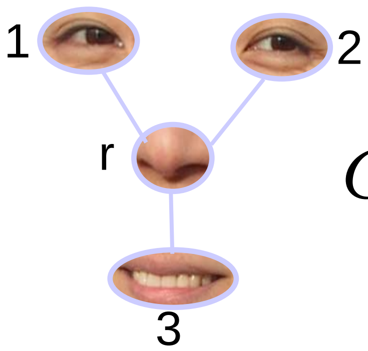
$$p(C|w_i, x_i) \;\alpha\; p(C)p(w_i|C)p(x_i|C)$$

appearance configuration

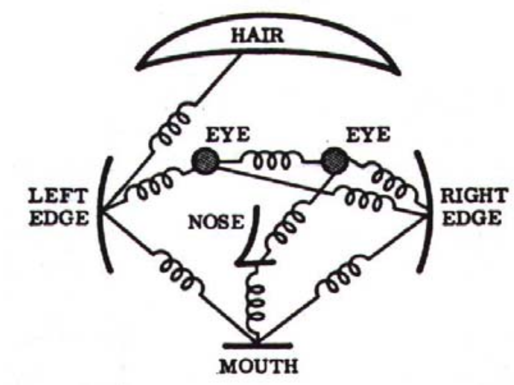
independence assumption

$$p(x_i|C)$$

- depends on relative position of parts
- usually Gaussian

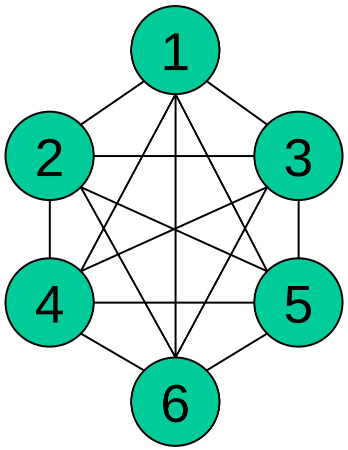


$$G(x_1 - x_r, x_2 - x_r, x_3 - x_r)$$



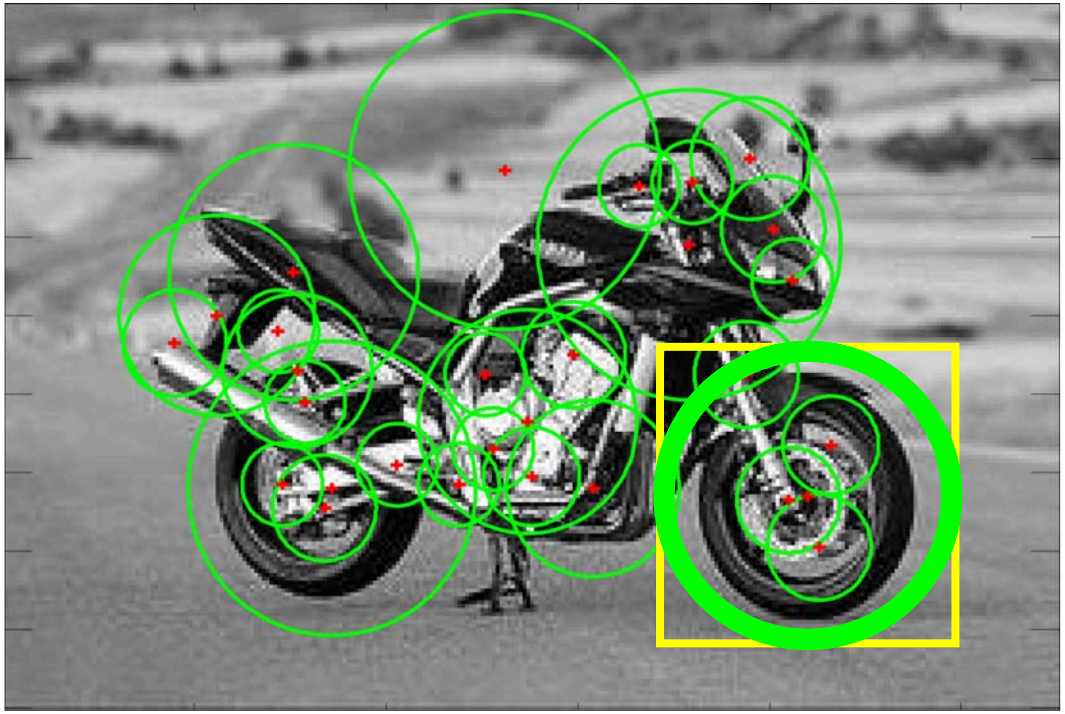
Constellation model

- Explicit structure: joint Gaussian over all part positions
  - dates back to Weber, Welling & Perona 2000 and earlier
- Also, explicit appearance model – Gaussian
- Simultaneous learning of parts and structure



Fergus, Perona & Zisserman 2003

Representation of regions



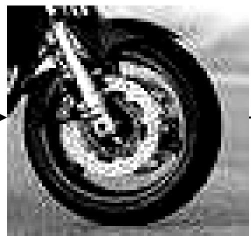
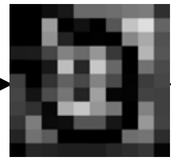
Location

(x,y) coords. of region centre

Scale

Radius of region (pixels)

Appearance (monochrome)

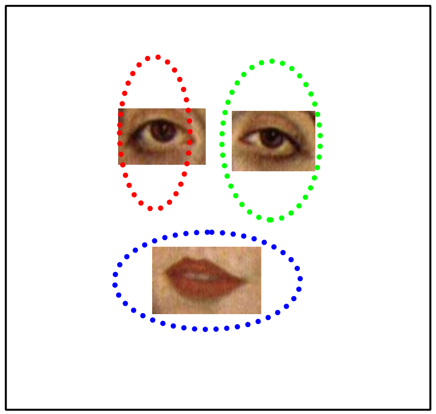
 Normalize →  11x11 patch →  Projection onto PCA basis → $\begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{15} \end{pmatrix}$

Gives representation of appearance in low-dimensional vector space

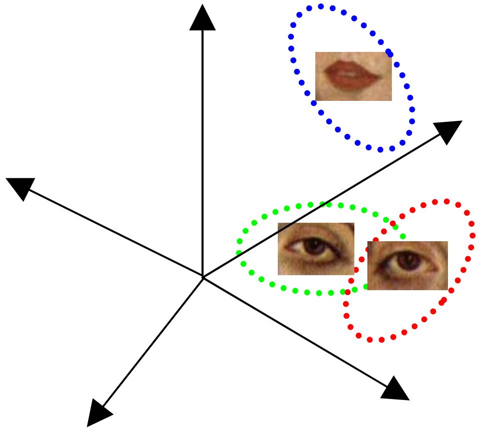
Generative probabilistic model

Foreground model

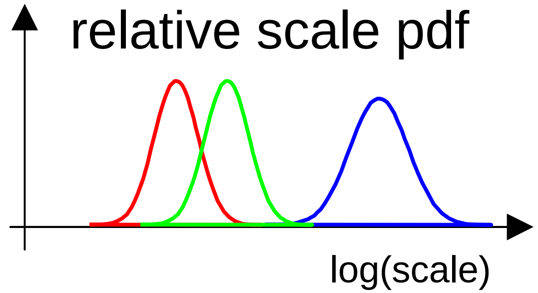
Gaussian shape pdf



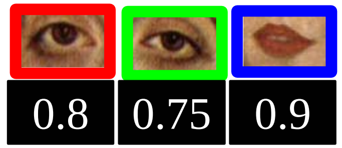
Gaussian part appearance pdf



Gaussian relative scale pdf

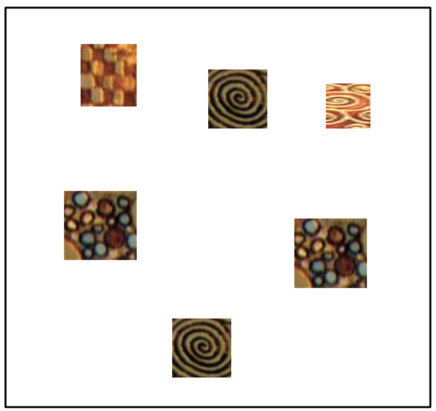


Prob. of detection

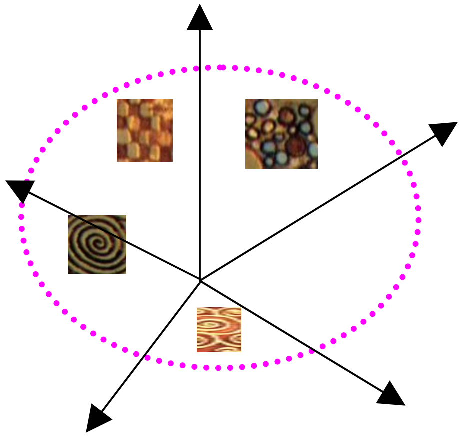


Clutter model

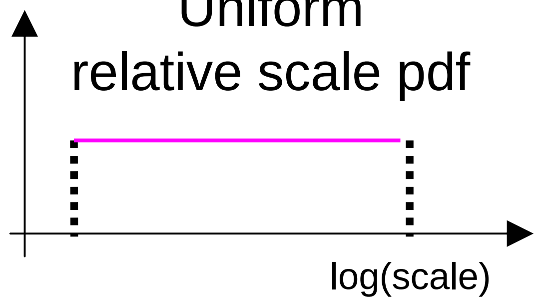
Uniform shape pdf



Gaussian background appearance pdf



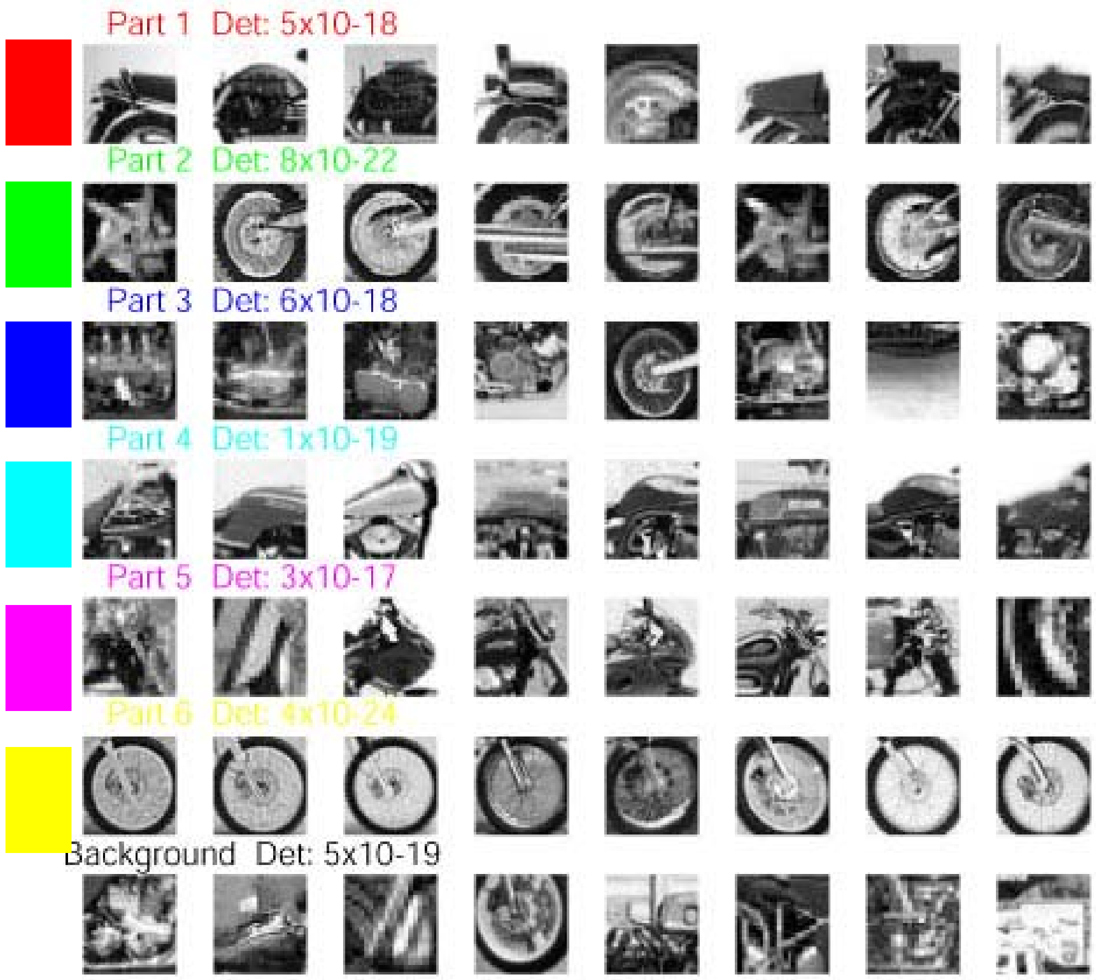
Uniform relative scale pdf



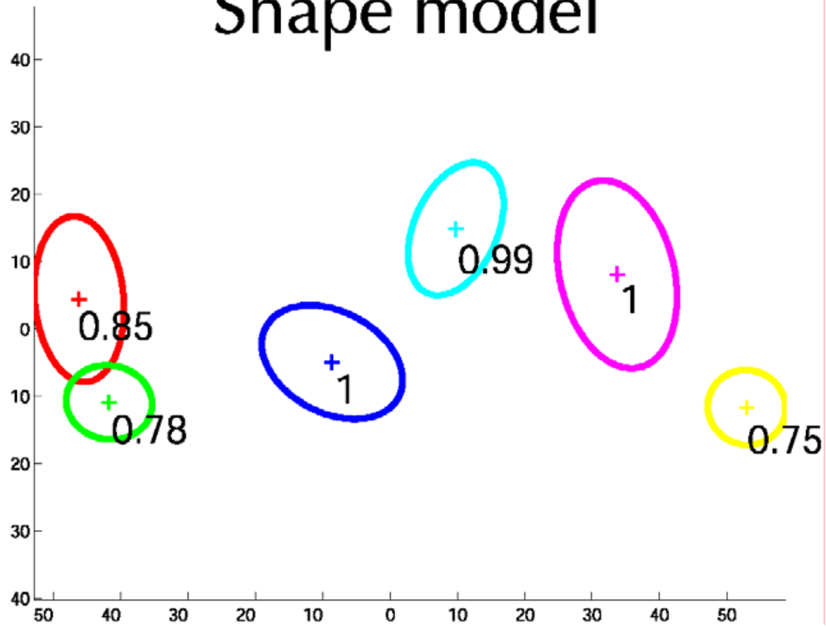
Poission pdf on # detections

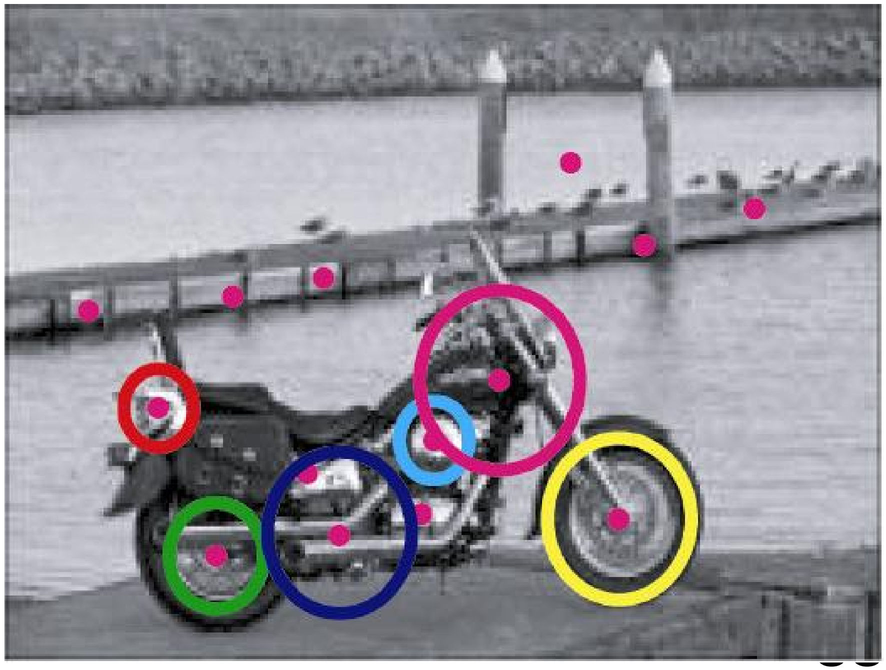
Example – Learnt Motorbike Model

Samples from appearance model



Shape model

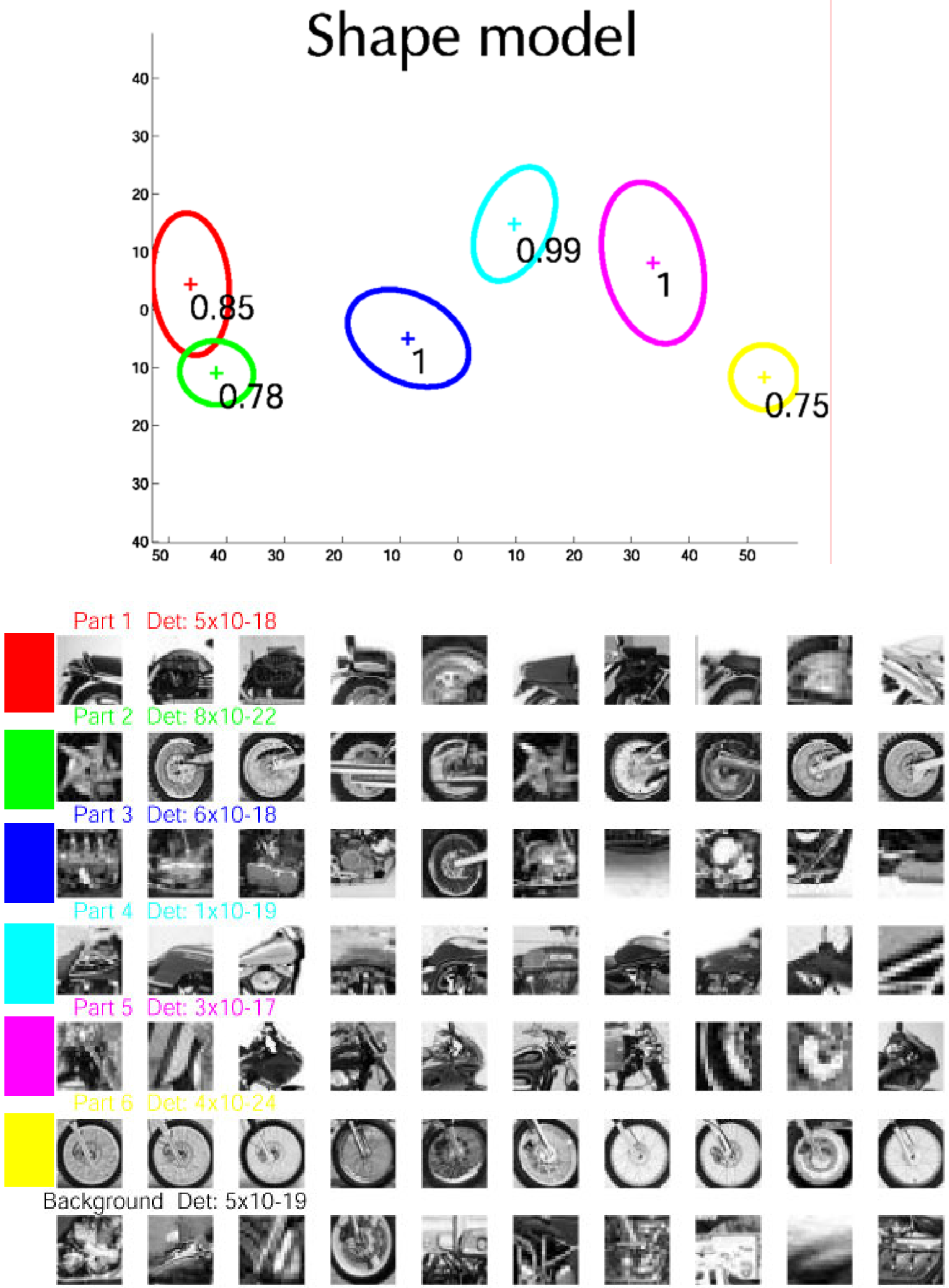




Recognition

- Detect regions in target image
- Evaluate the likelihood of the model (a search over assignments of parts to regions)
- Threshold on the likelihood ratio

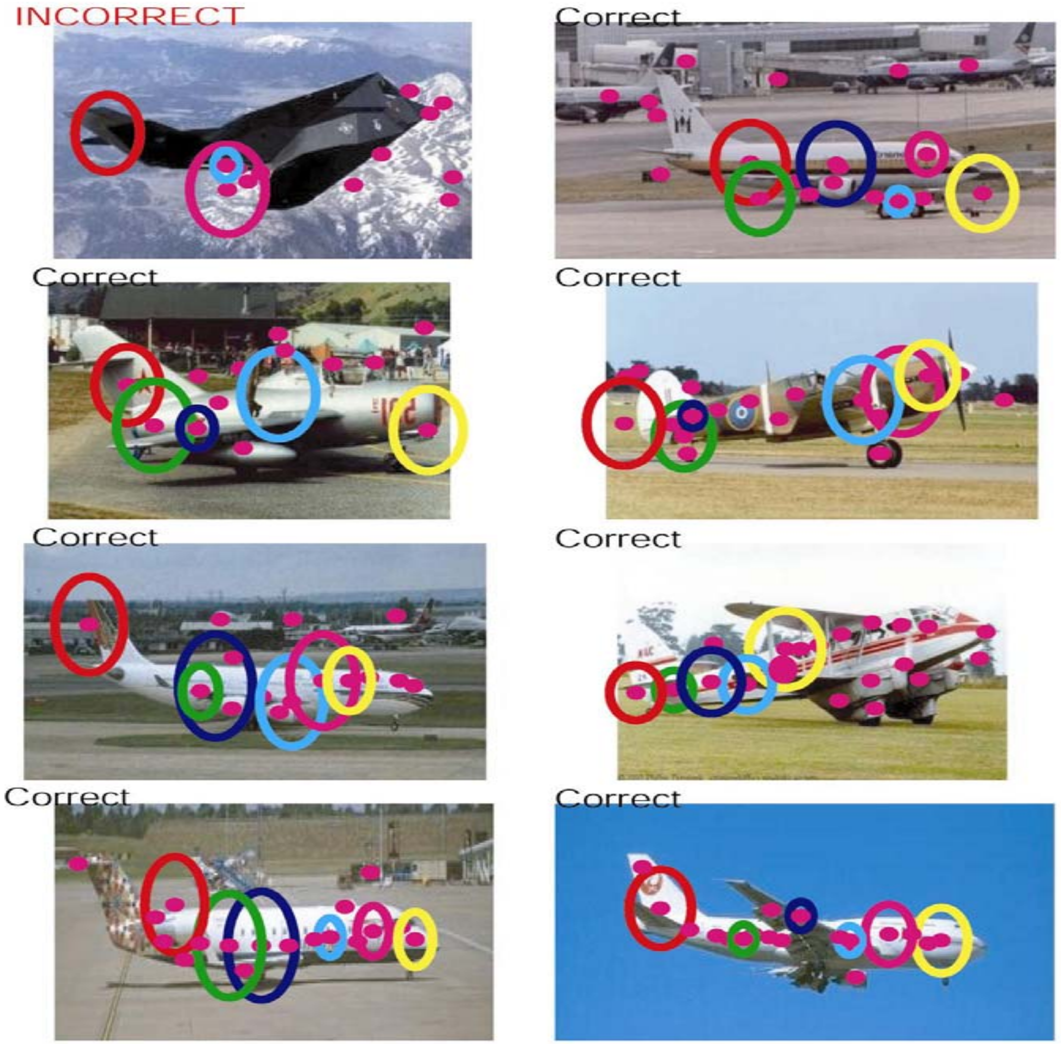
Recognized Motorbikes



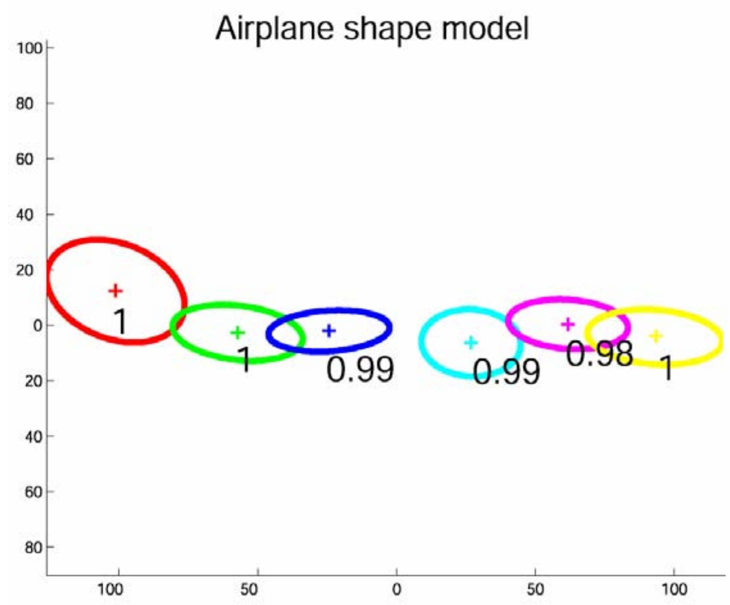
position of object determined

Background images evaluated with motorbike model

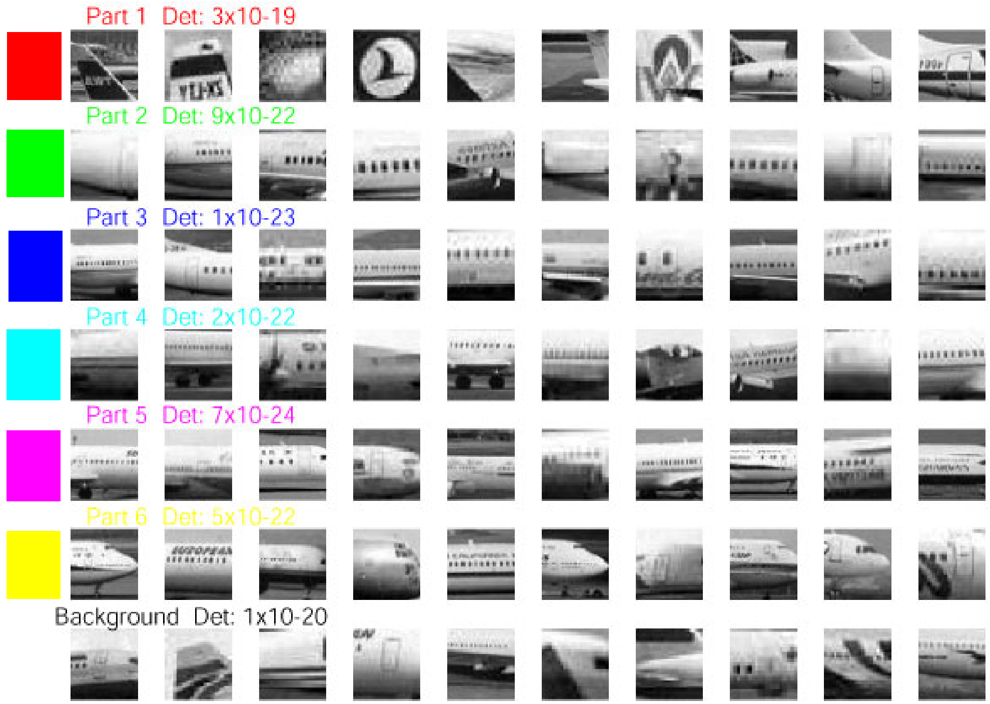
Airplanes



INCORRECT
Correct
Correct
Correct
Correct
Correct
Correct
Correct



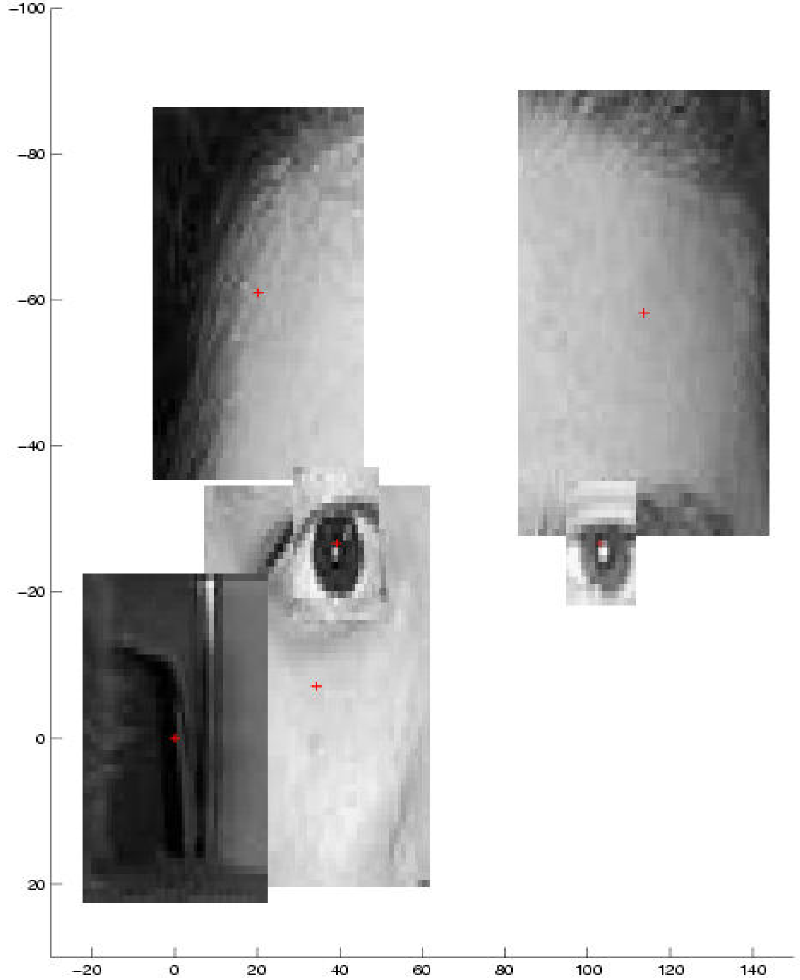
Airplane shape model
100
80
60
40
20
0
20
40
60
80
100
50
0
50
100
1
1
0.99
0.99
0.98
1



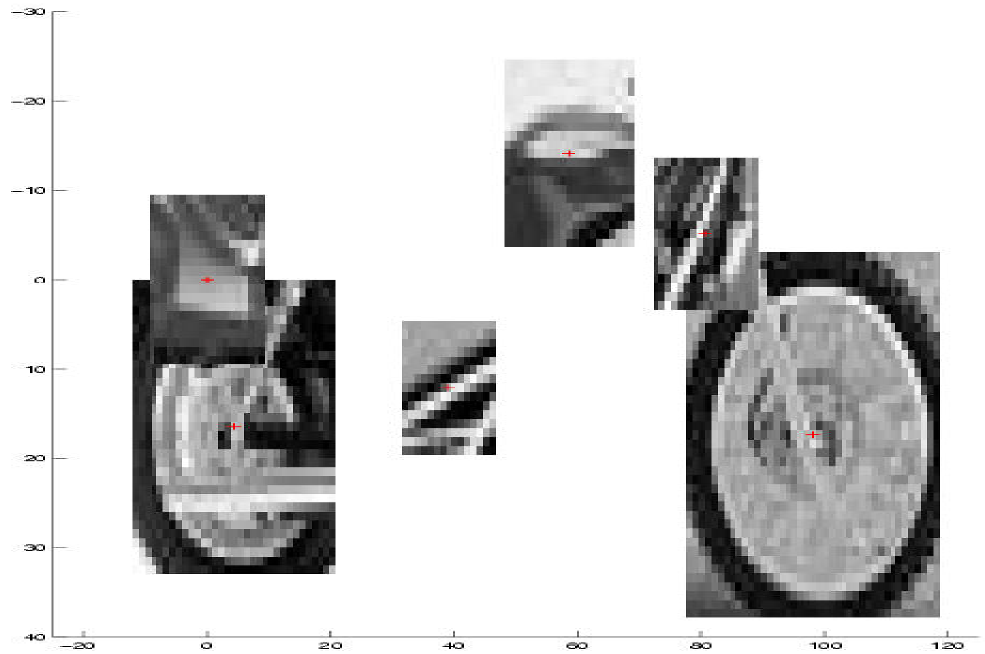
Part 1 Det: 3x10-19
Part 2 Det: 9x10-22
Part 3 Det: 1x10-23
Part 4 Det: 2x10-22
Part 5 Det: 7x10-24
Part 6 Det: 5x10-22
Background Det: 1x10-20

Sampling from models

- generative model



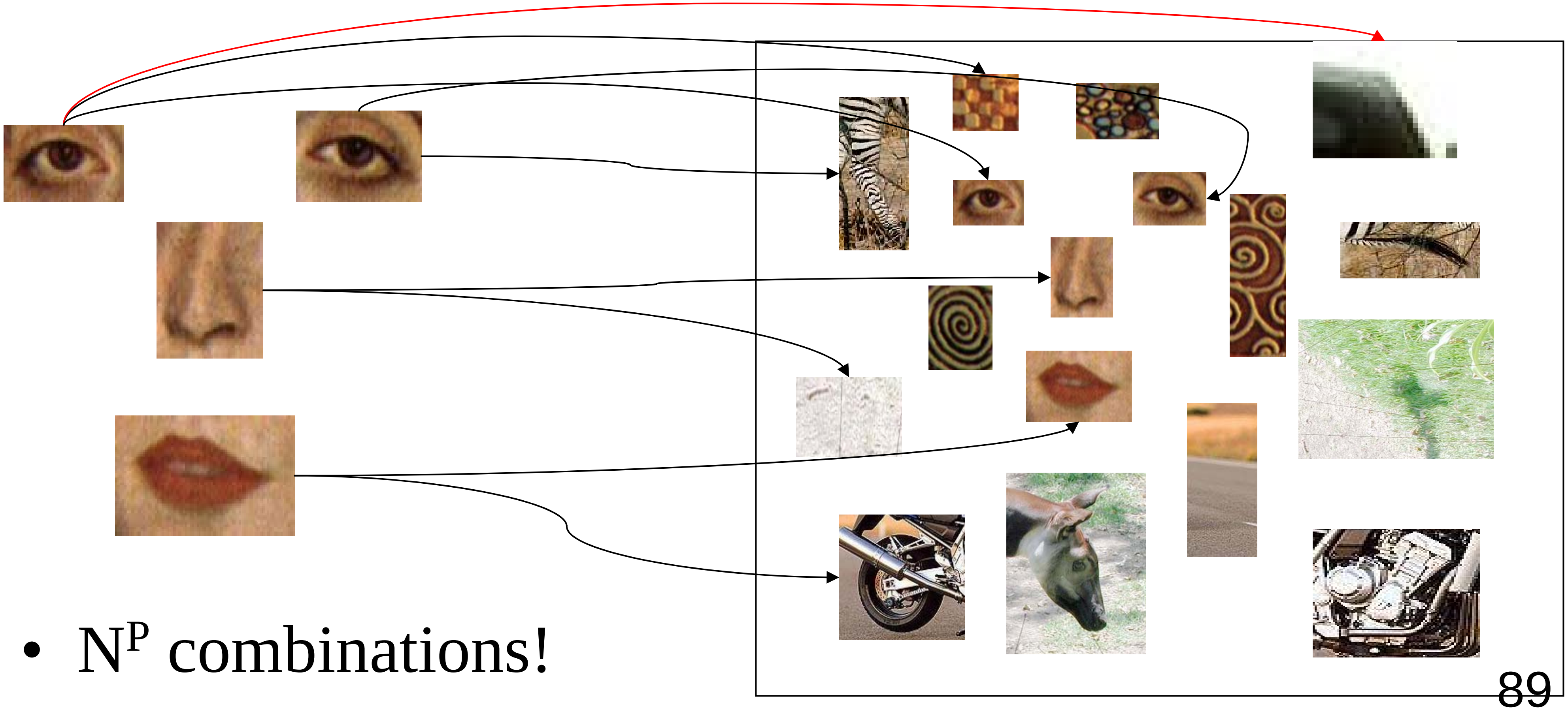
Faces



Motorbikes

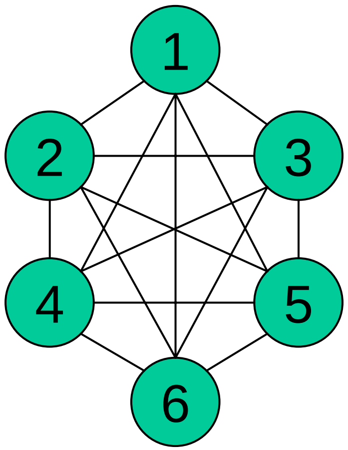
The correspondence problem

- Model with P parts
- Image with N possible locations for each part



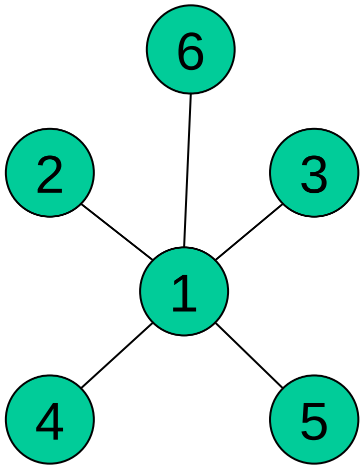
- N^P combinations!

Different graph structures



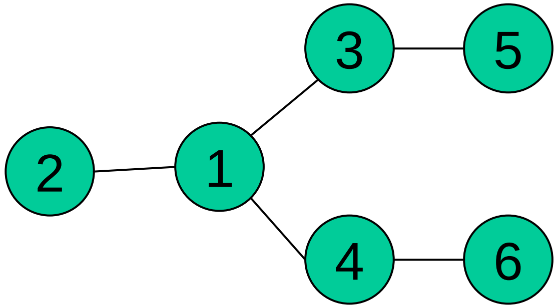
Fully connected

$O(N^6)$



Star structure

$O(N^2)$



Tree structure

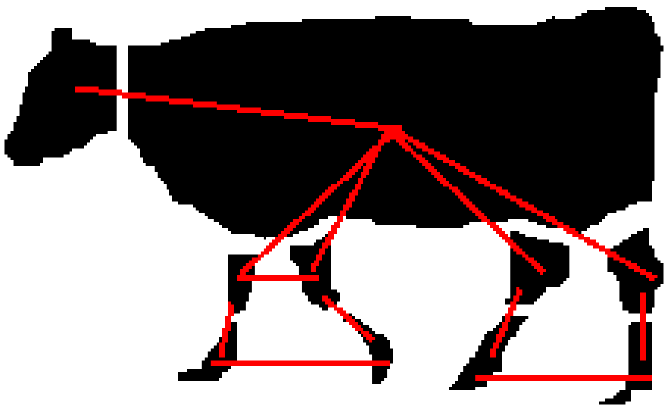
$O(N^2)$

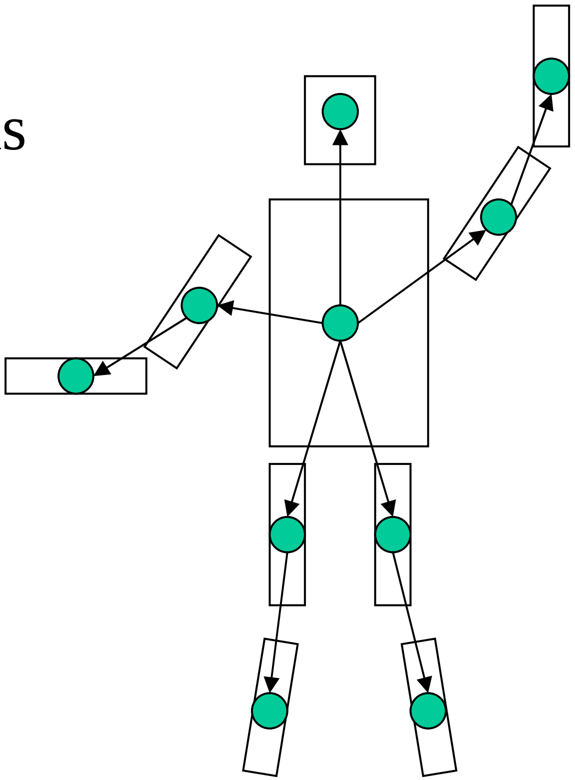
- Sparser graphs cannot capture all interactions between parts,
- but far cheaper to recognize (and learn)

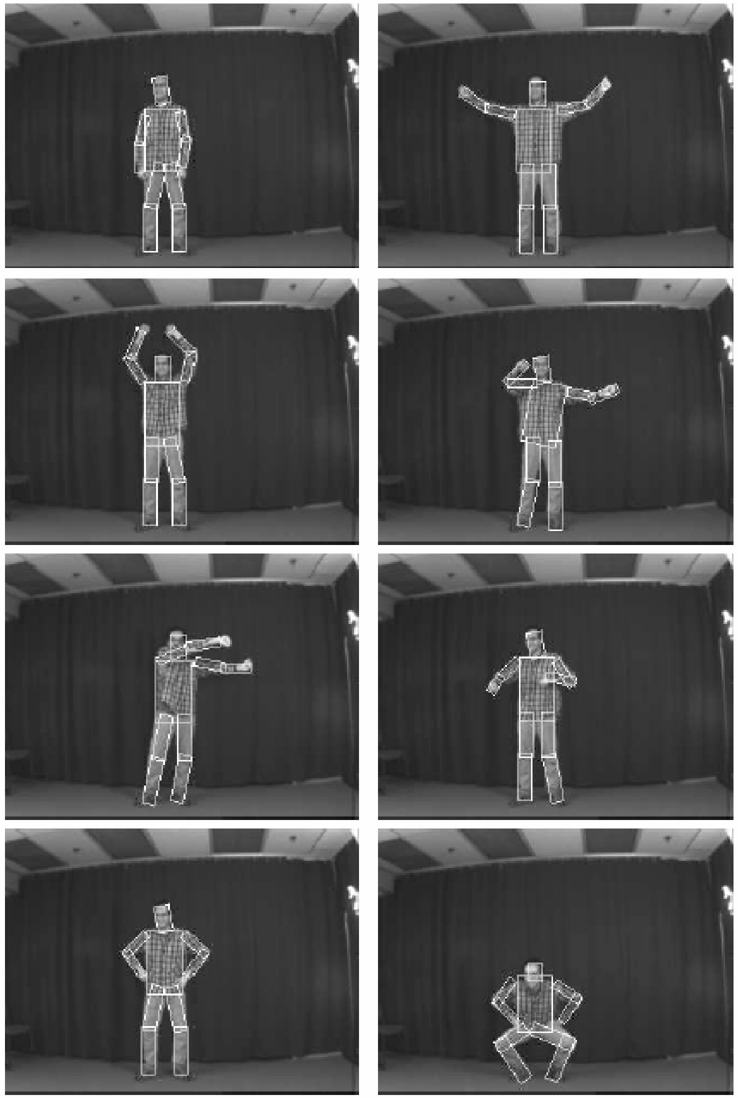
Some class-specific graphs

- Articulated motion
  - People
  - Animals

- Special parameterisations
  - Limb angles



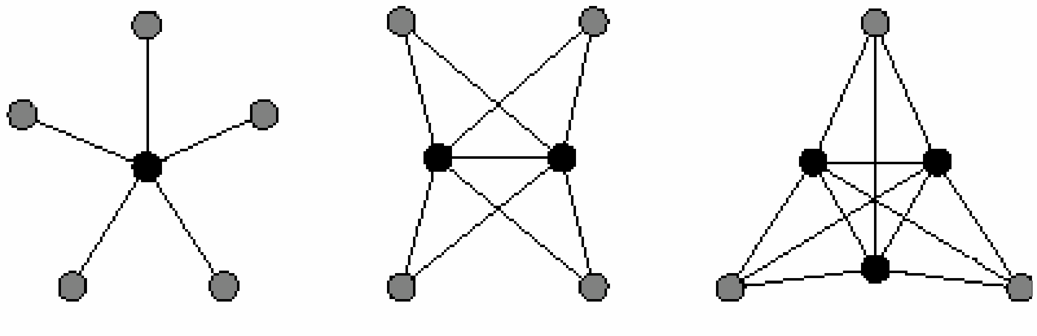




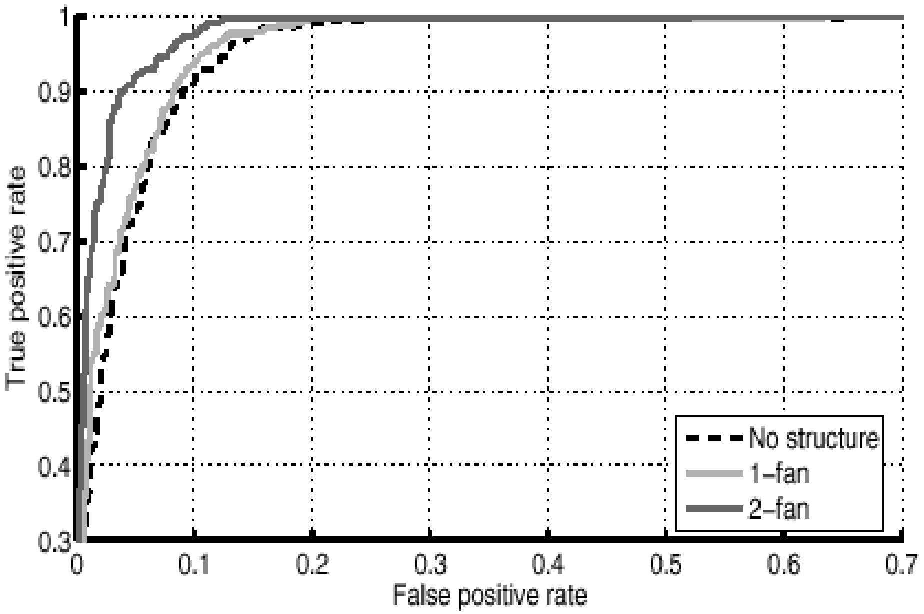
Images from [Kumar05, Felzenszwalb05]

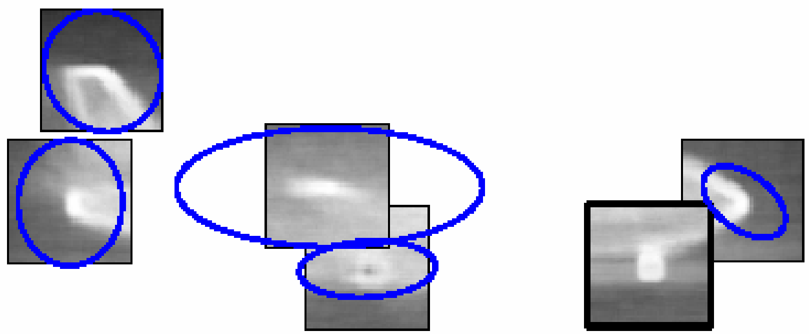
How much does shape help?

- Crandall, Felzenszwalb, Huttenlocher CVPR’05
- Shape variance increases with increasing model complexity

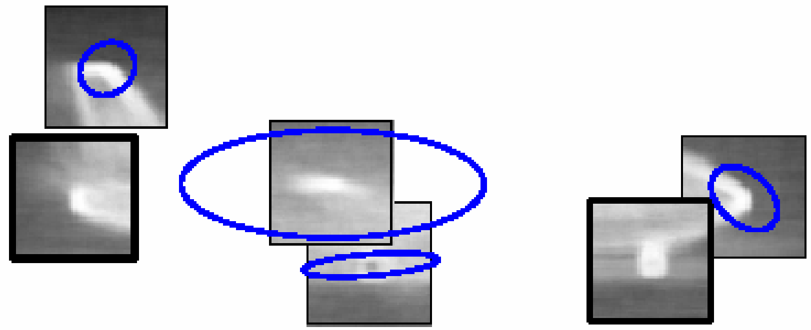


1-fan 2-fan 3-fan





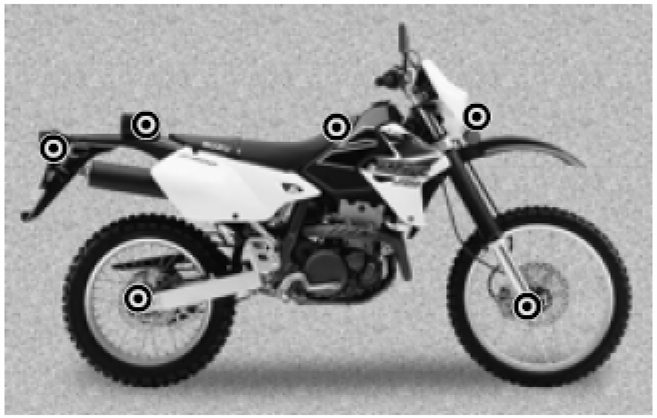
(a) Airplane, 1-fan



(b) Airplane, 2-fan

6 part models

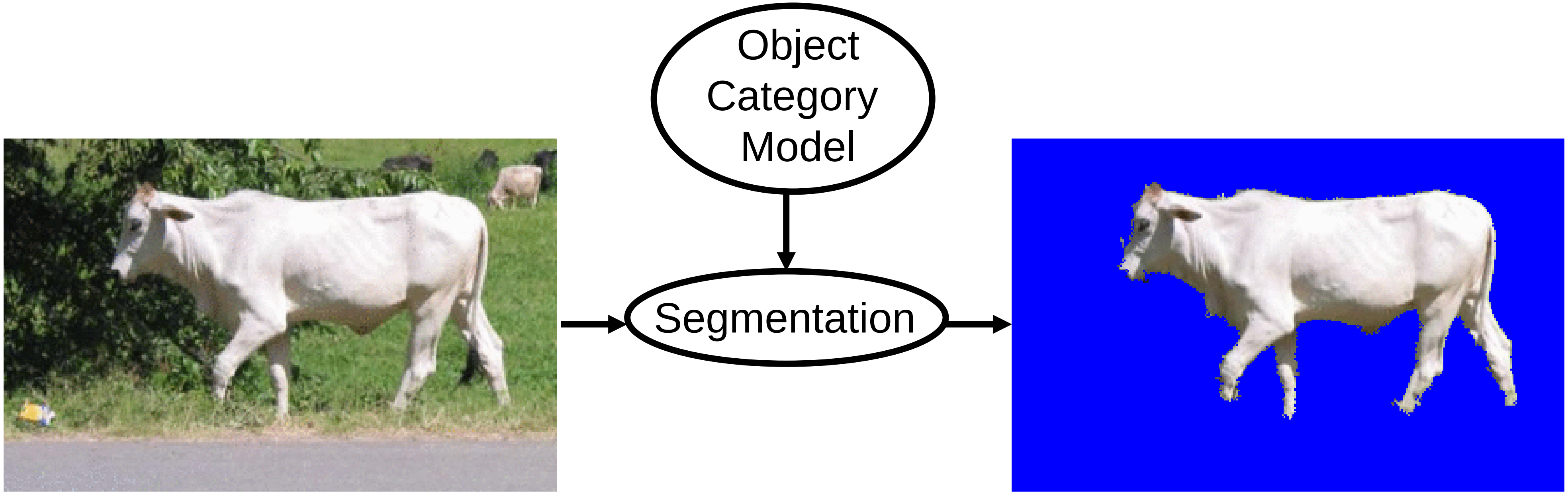
	Planes	Bikes	Faces
0-fans	90.5%	96.5%	98.2%
1-fans	91.3**%**	**97.0%**	98.2%
2-fans	93.3**%**	97.0%	98.2%


4. Class based segmentation

Objective

- Given an image, to recognize and segment the object

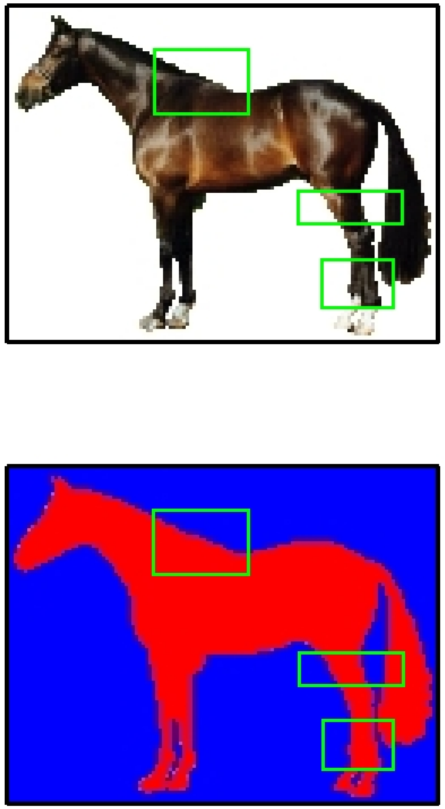


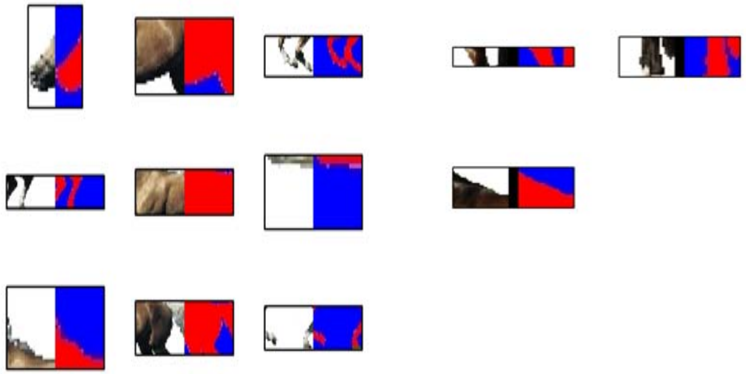
Cow Image Segmented Cow

- Combine object detection with segmentation
  - Borenstein and Ullman, ECCV ’02
  - Leibe and Schiele, BMVC ’03

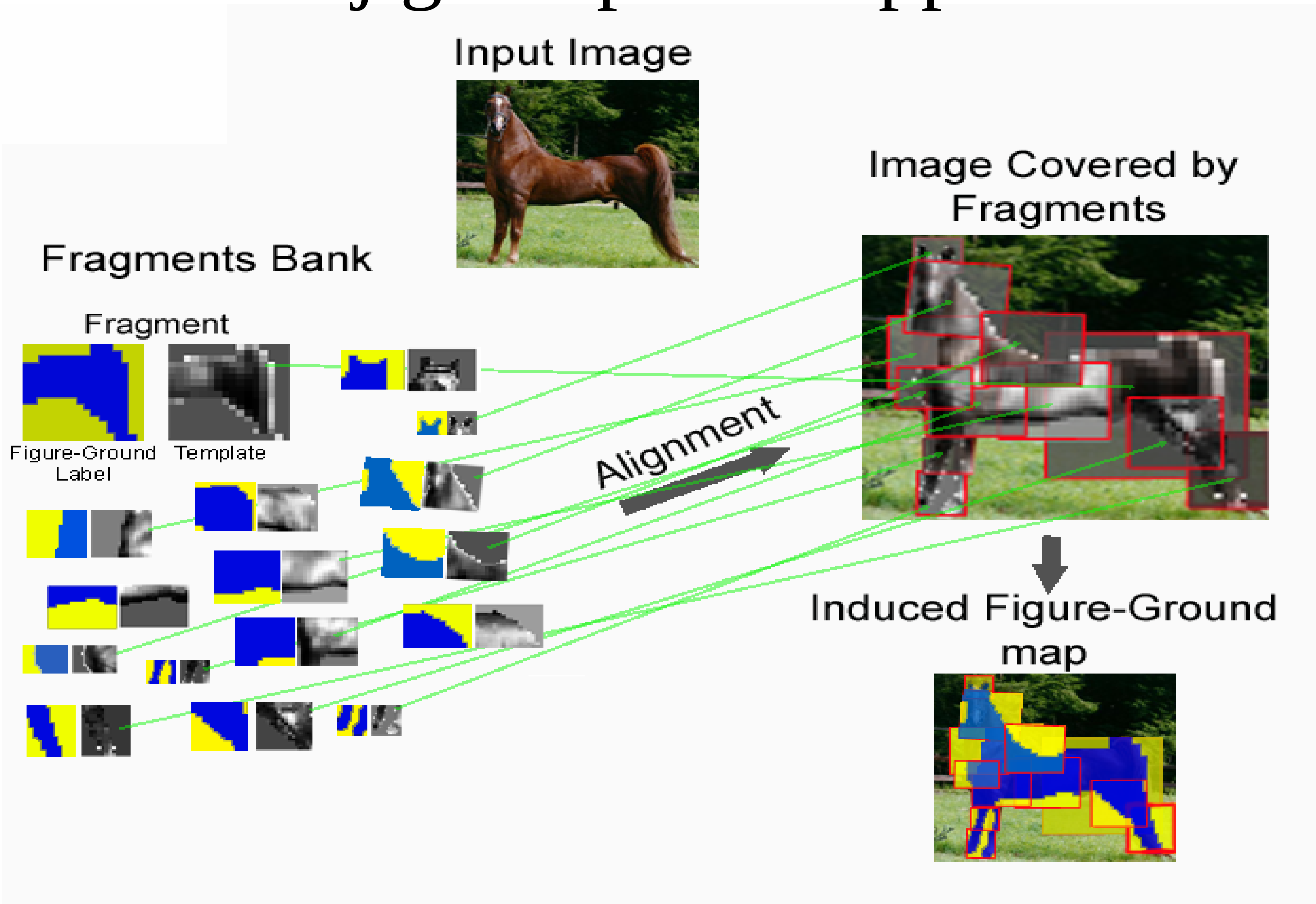
Background: Borenstein & Ullman 2002

- Training
- Learn fragments from segmented images

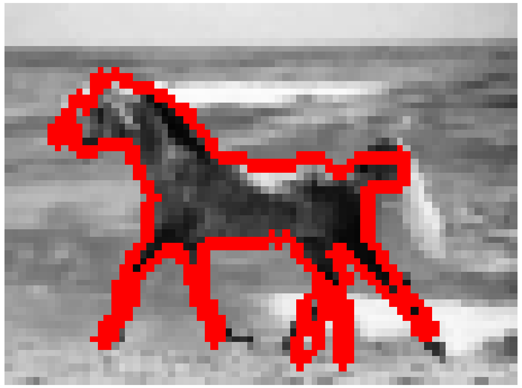


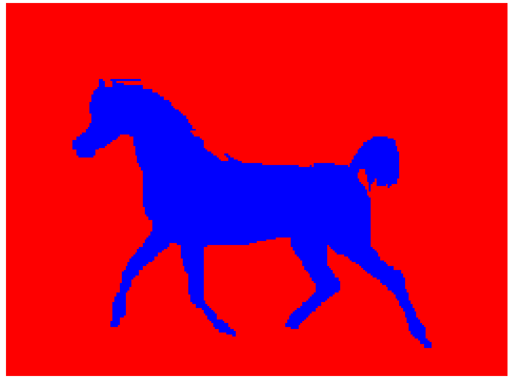


Structure: jigsaw puzzle approach



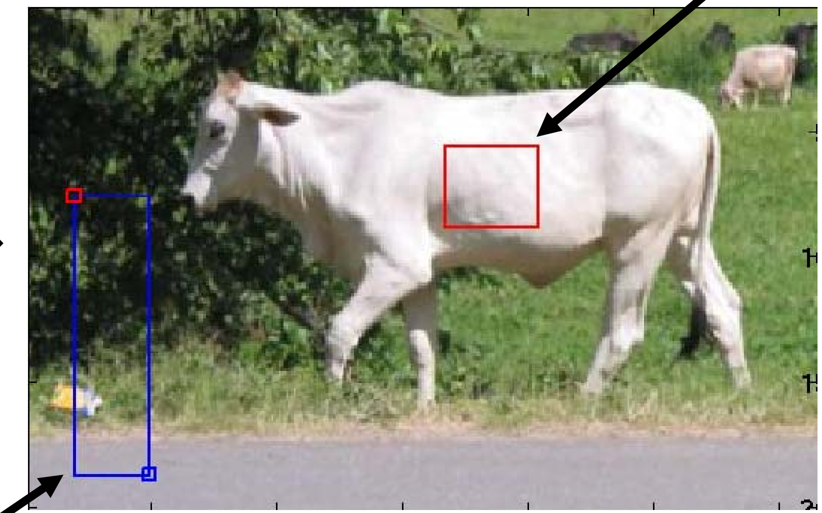
1. Obtain approximate foreground segmentation using parts
2. Refine using bottom up segmentation



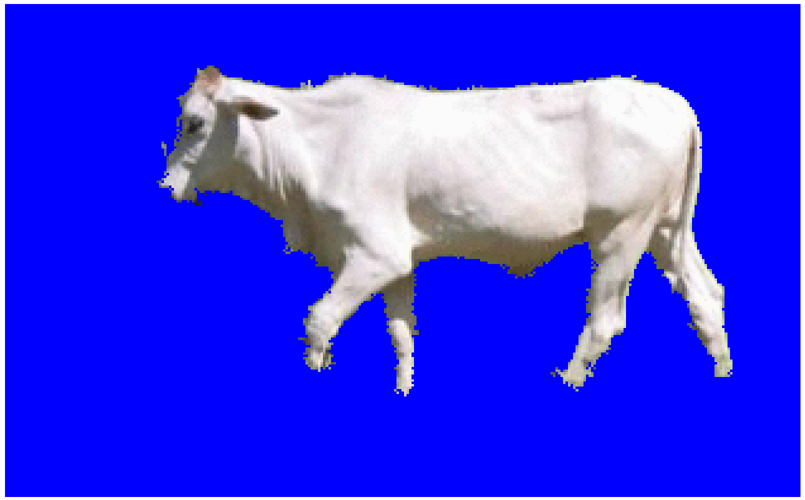


Object segmentation using graph cuts

foreground Seed Pixels



Background Seed Pixels

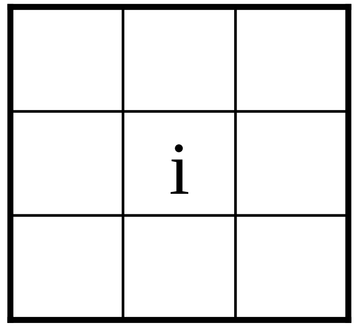


Boykov and Jolly, ICCV 01

user provides
foreground/background regions

Binary Markov Random Field

$$f(\mathbf{x}) = \sum_{i=1}^{n} \{m_i(x_i) + \sum_{j \in \mathcal{N}(i)} \phi_i(x_i, x_j)\}$$



$\mathcal{N}(i)$

- $x_i = 1$ for foreground pixels, $x_i = 0$ for background
- $m_i(x_i)$ is likelihood that pixel at i is foreground (if $x_i = 1$), or background (if $x_i = 0$), e.g. using colour histogram of seed regions
- $\phi(x_i, x_j)$ penalizes a change of state:

$$\phi(x_i, x_j) = \begin{cases} 0 & \text{if} x_i = x_j \\ \gamma e^{-\beta(I_i - I_j)^2} & \text{if} x_i \neq x_j. \end{cases}$$

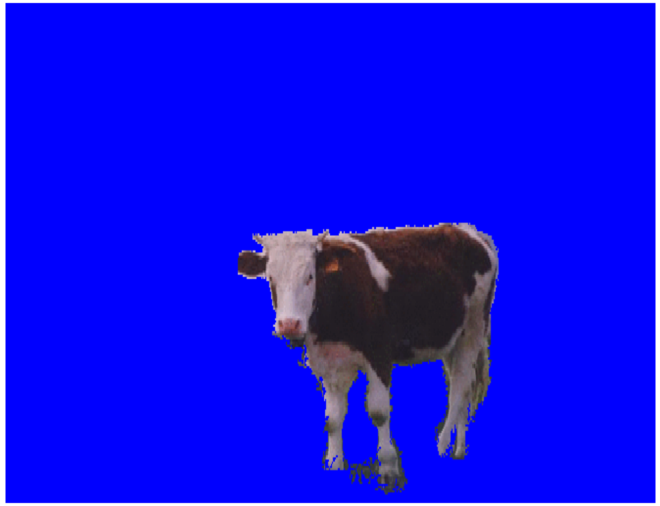
Can be optimized globally with graph cuts algorithm

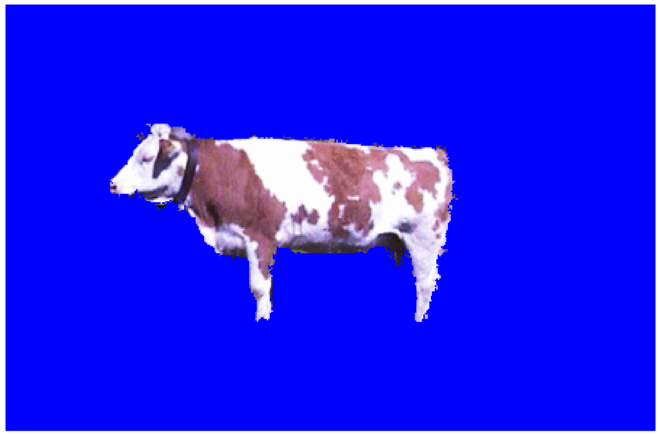
ObjCut

- Recognize object using category model (LPS)
- Provides foreground/background for colour and texture
- Apply graph cuts segmentation

Image

Segmentation





Kumar et al CVPR 05

Using LPS Model for Horse

Image

Segmentation

5. Summary and open challenges

- ☺ Single visual aspects (e.g. car rear/front)
  - Can learn and recognize from unsegmented images
  - Translation and scale invariance
  - Partial occlusion tolerated
  - Background clutter tolerated
  - Heterogeneous models

- ☹ Multiple visual aspects (e.g. car from any viewpoint)
  - Multiple 2D models ?
  - 3D models ?

Open Research Areas

- Structure model
  - tight parametric model (e.g. complete Gaussian)
  - loose model (e.g. pairwise relations)
- Greater viewpoint invariance
  - scale invariant → similarity invariant → affine invariant
- Multiple class/Hierarchical class models
- Ease of learning
  - learn from ‘contaminated’ data sets
  - learn multiple object classes simultaneously
- Difficulty of training/testing sets

Datasets and software

- All image datasets:
  - http://www.pascal-network.org/challenges/VOC/
- Caltech image datasets:
  - http://www.robots.ox.ac.uk/~vgg/data.html, and
  - http://www.vision.caltech.edu/html-files/archive.html
- Feature detectors (scale and affine covariant)
  - http://www.robots.ox.ac.uk/~vgg/research/affine