BSB663
Image Processing

Pinar Duygulu

Revisit Texture

- Texture depicts spatially repeating patterns
- Many natural phenomena are textures

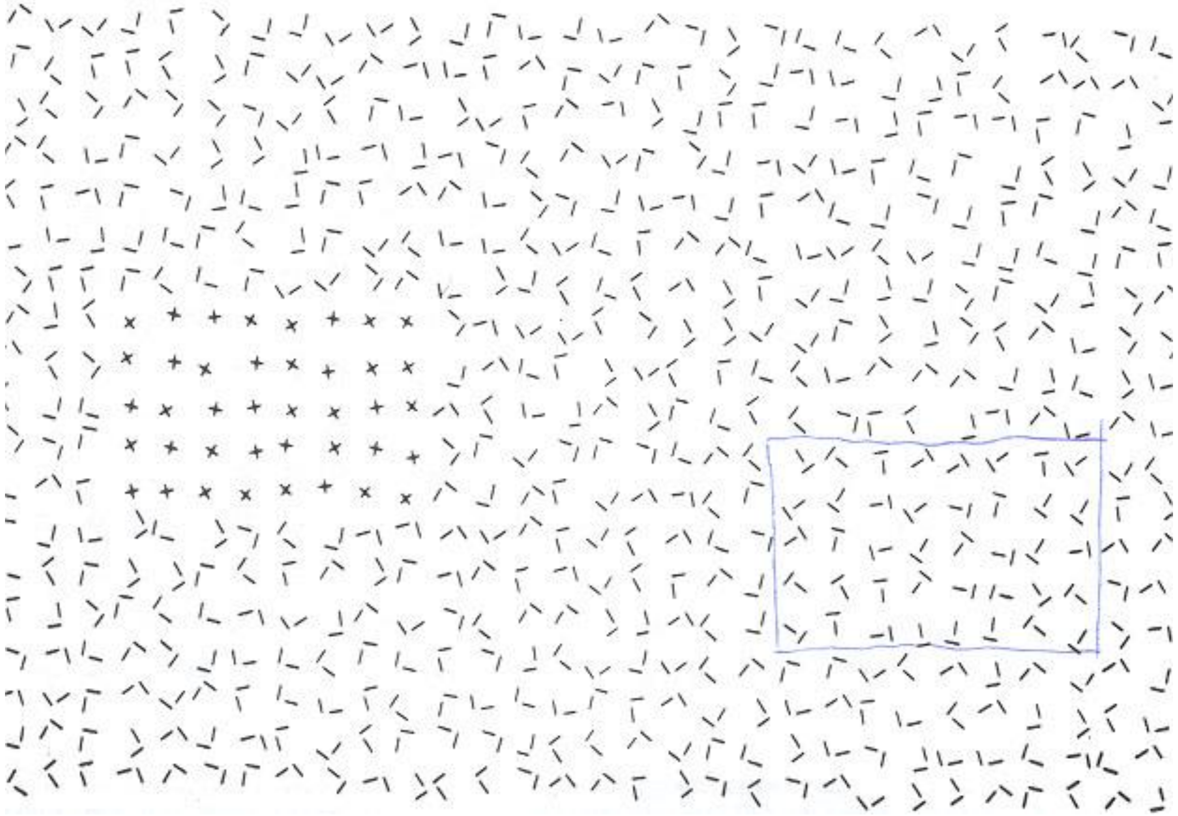
radishes

rocks

yogurt

Alyosha Efros, CMU

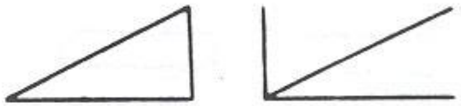
Texton Discrimination (Julesz)



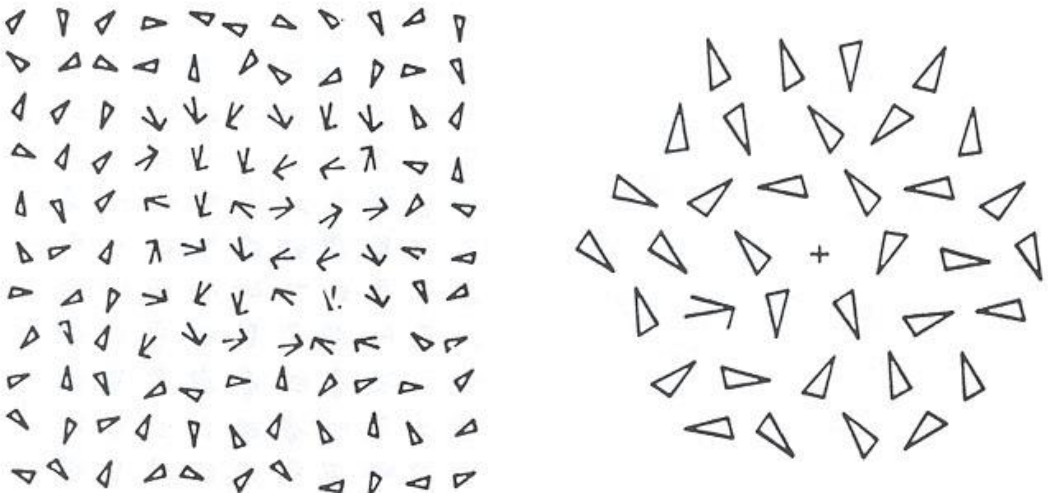
Human vision is sensitive to the difference of some types of elements and appears to be "numb" on other types of differences.

Alyosha Efros, CMU

Search Experiment I



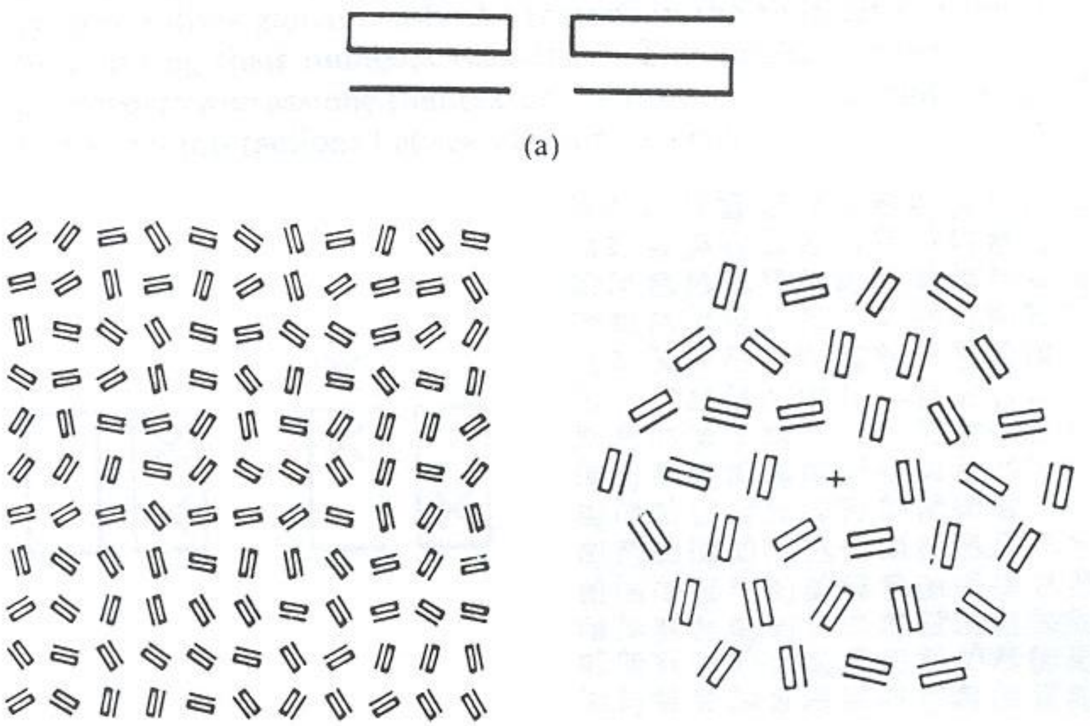
(a)



The subject is told to detect a target element in a number of background elements.
In this example, the detection time is independent of the number of background elements.

Alyosha Efros, CMU

Search Experiment II



In this example, the detection time is proportional to the number of background elements,
And thus suggests that the subject is doing element-by-element scrutiny.

Alyosha Efros, CMU

Heuristic (Axiom) I

Julesz then conjectured the following axiom:

Human vision operates in two distinct modes:

1. Preattentive vision

   parallel, instantaneous *(~100--200ms)*, without scrutiny,
   independent of the number of patterns, covering a large visual field.

2. Attentive vision

   serial search by focal attention in 50ms steps limited to small aperture.

Then what are the basic elements?

Alyosha Efros, CMU

Heuristic (Axiom) II

Julesz's second heuristic answers this question:

Textons are the fundamental elements in preattentive vision, including

1. Elongated blobs

   rectangles, ellipses, line segments with attributes

   color, orientation, width, length, flicker rate.

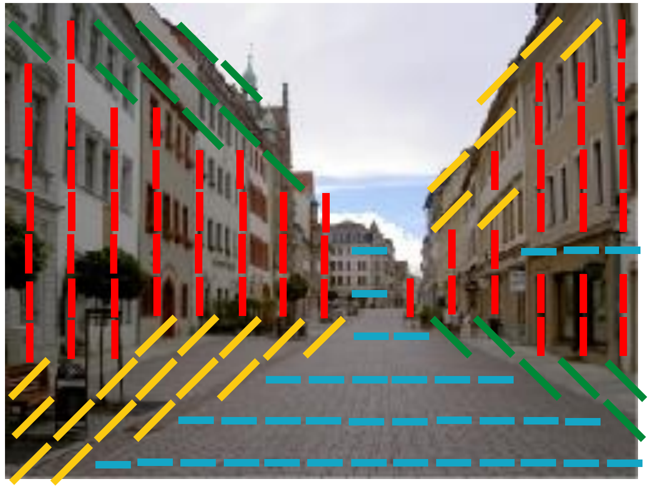
2. Terminators

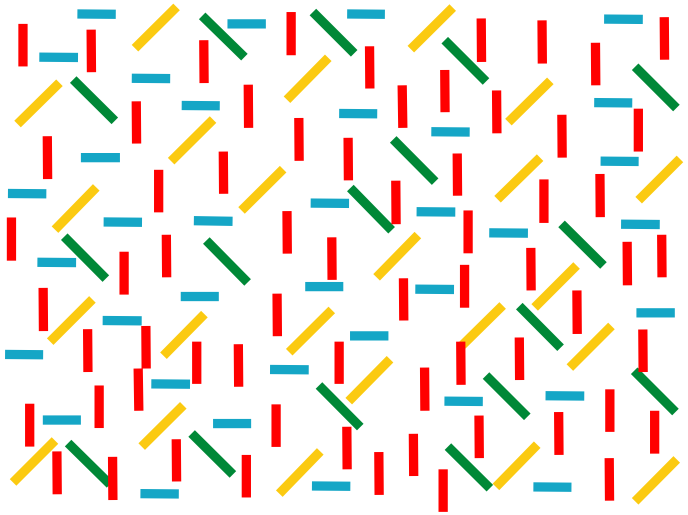
   ends of line segments.

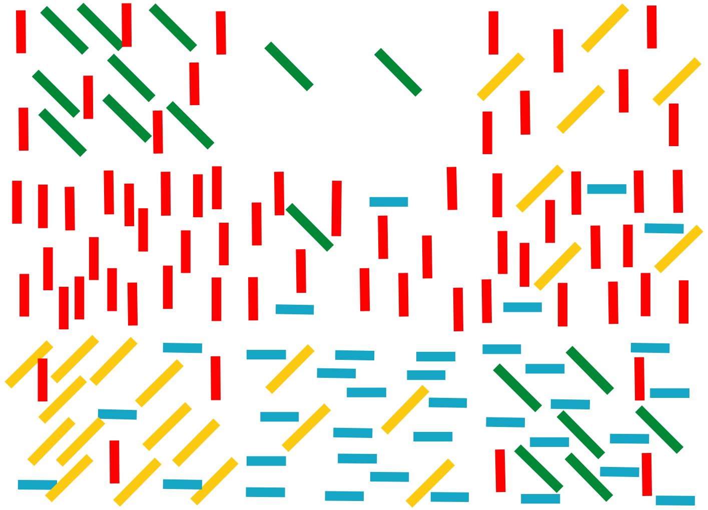
3. Crossings of line segments.

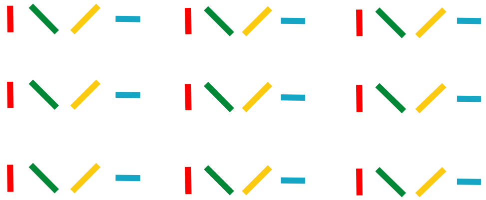
But it is worth noting that Julesz's conclusions are largely based by ensemble of artificial texture patterns. It was infeasible to synthesize natural textures for controlled experiments at that time.

Bag of words



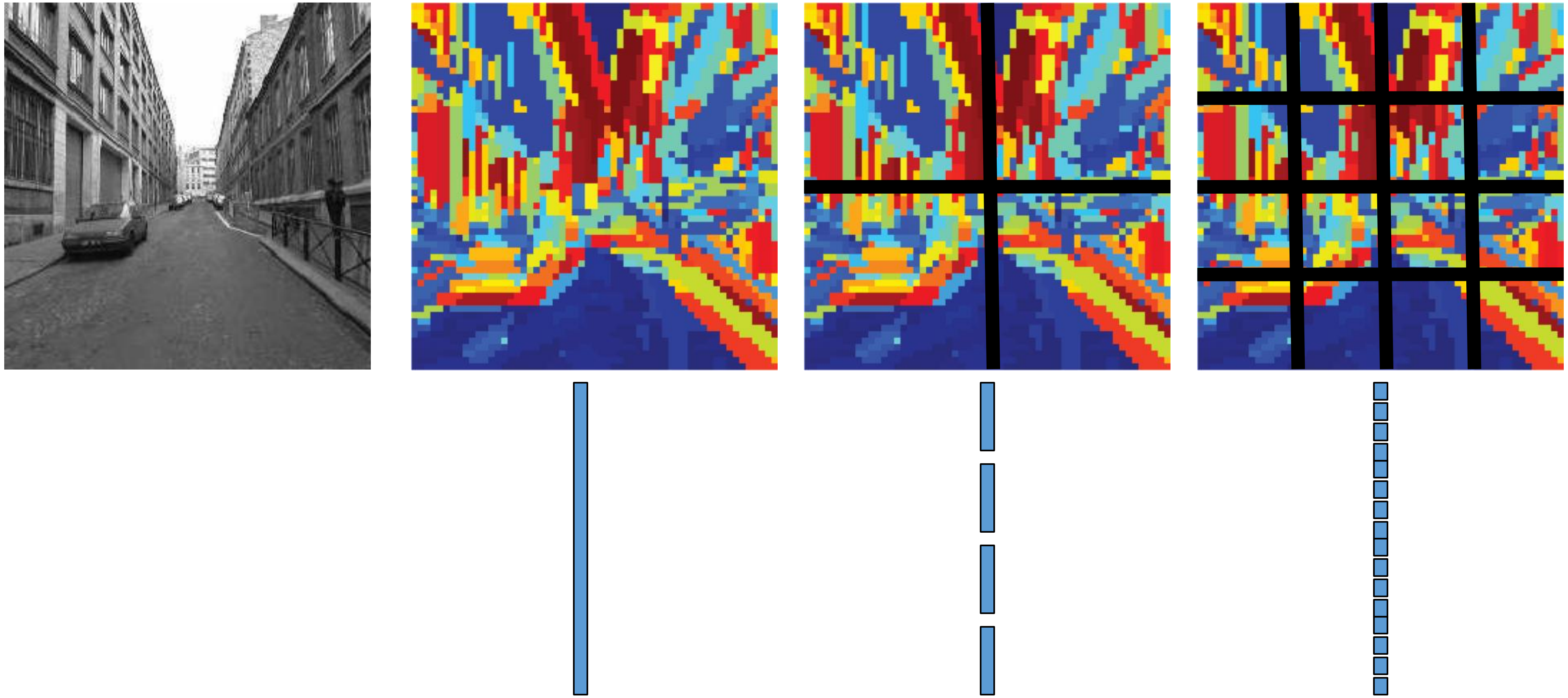






Torralba, MIT

Bag of words &
spatial pyramid matching



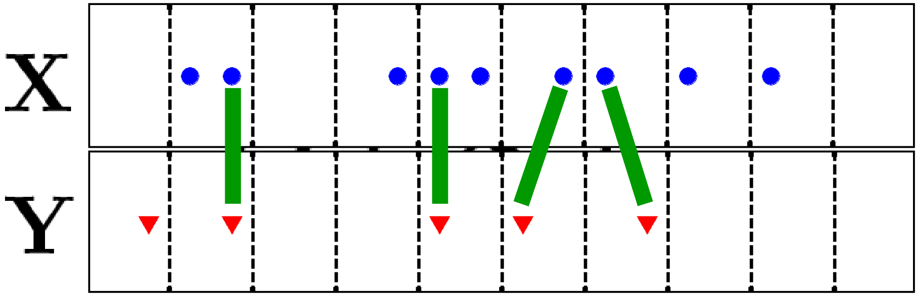
Grauman & Darell,
S. Lazebnik, et al, CVPR 2006

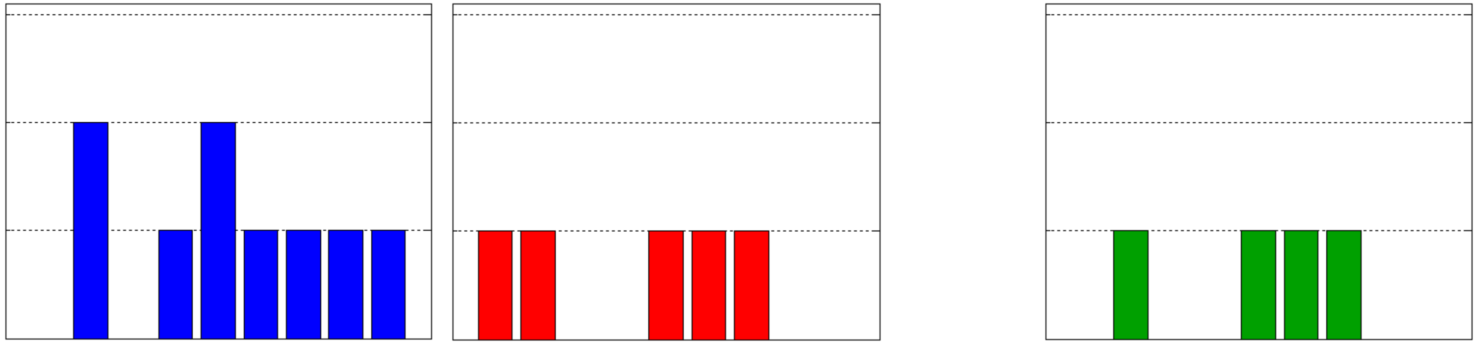
Torralba, MIT

Histogram Intersection

Histogram intersection

$$\mathcal{I}\left(H(\mathbf{X}), H(\mathbf{Y})\right) = \sum_{j=1}^{r} \min\left(H(\mathbf{X})_j, H(\mathbf{Y})_j\right)$$





$H(\mathbf{X})$ $\quad$ $H(\mathbf{Y})$ $\quad$ $\mathcal{I}\left(H(\mathbf{X}), H(\mathbf{Y})\right) = 4$

Slide credit: Kristen Grauman

Histogram based distances

Given two histograms: h1, h2, such that sum(h1)=sum(h2)=1

- Euclidean

D(h1, h2) = sum ((h1 – h2).^2)

- Histogram intersection

D(h1, h2) = 1-sum (min (h1, h2))

- X^2

D(h1, h2) = sum((h1-h2).^2 ./ (h1+h2))

(using Matlab notation)

Torralba, MIT

Capturing the “essence” of texture

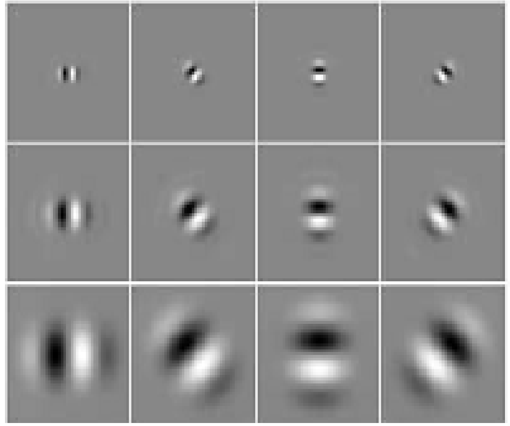
- …for real images

- We don’t want an actual texture realization, we want a texture invariant
- What are the tools for capturing <u>statistical</u> properties of some signal?

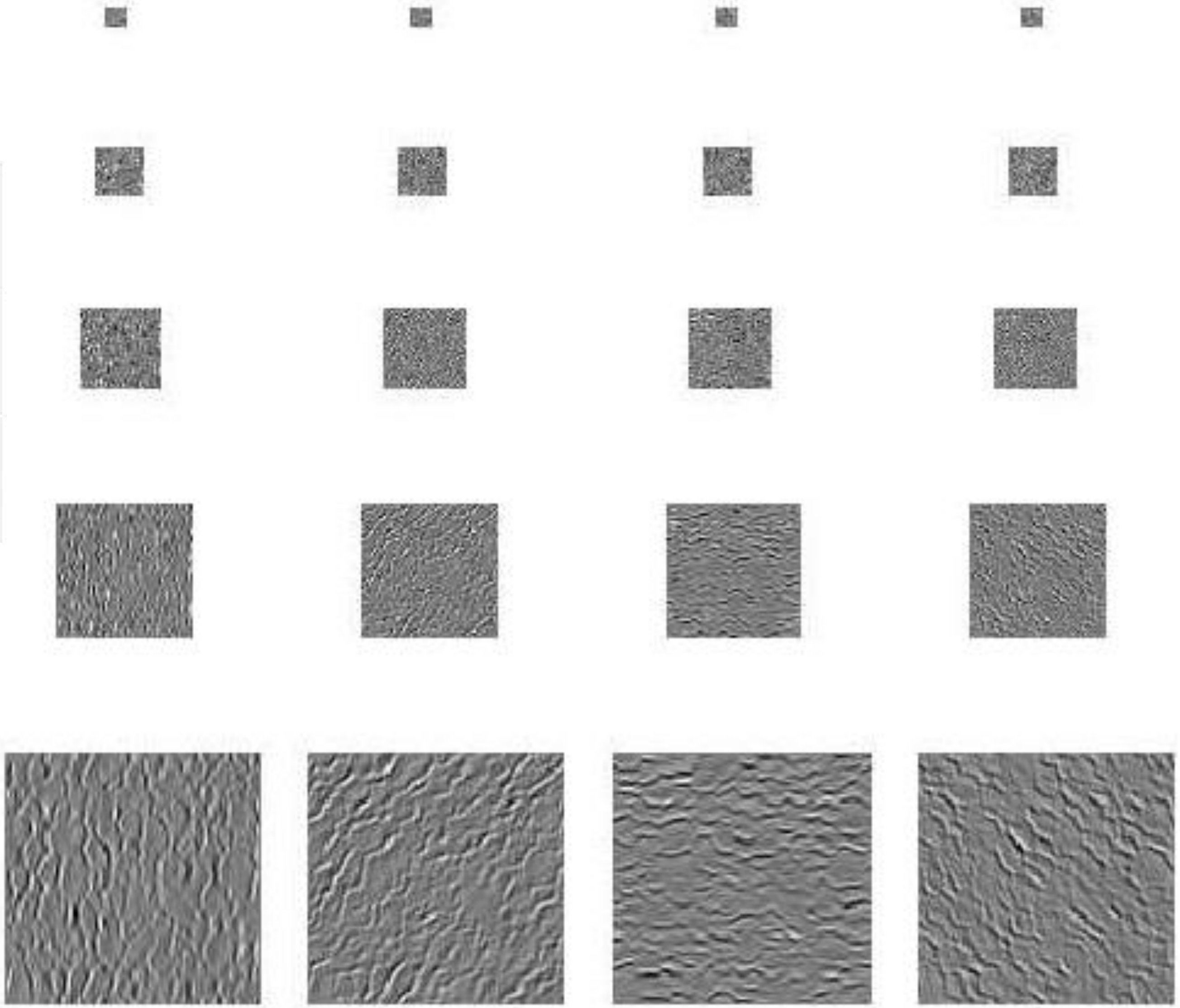
Alyosha Efros, CMU

Multi-scale filter decomposition

Filter bank

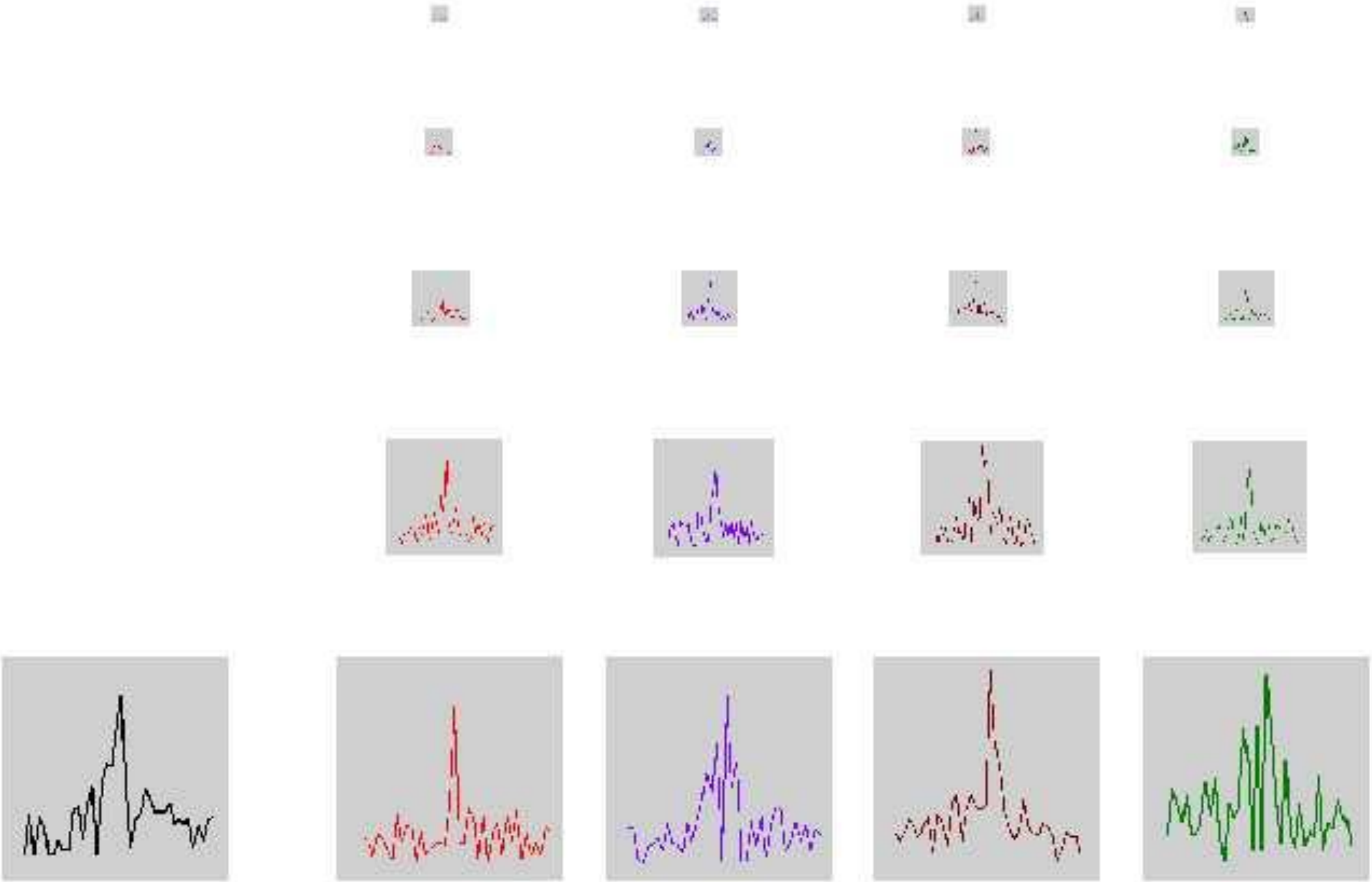


Input image



Alyosha Efros, CMU

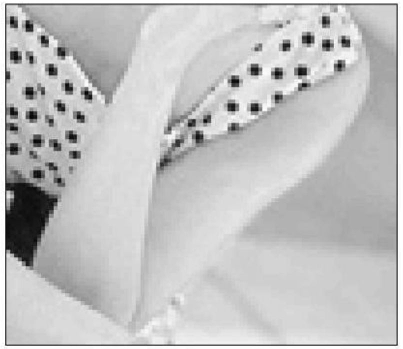
Filter response histograms

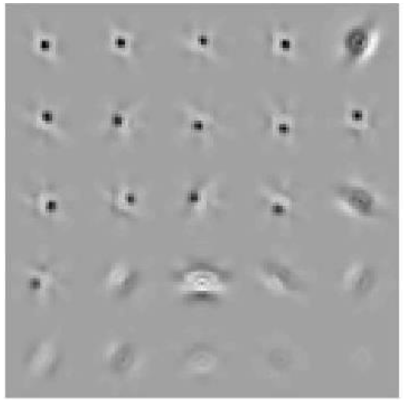


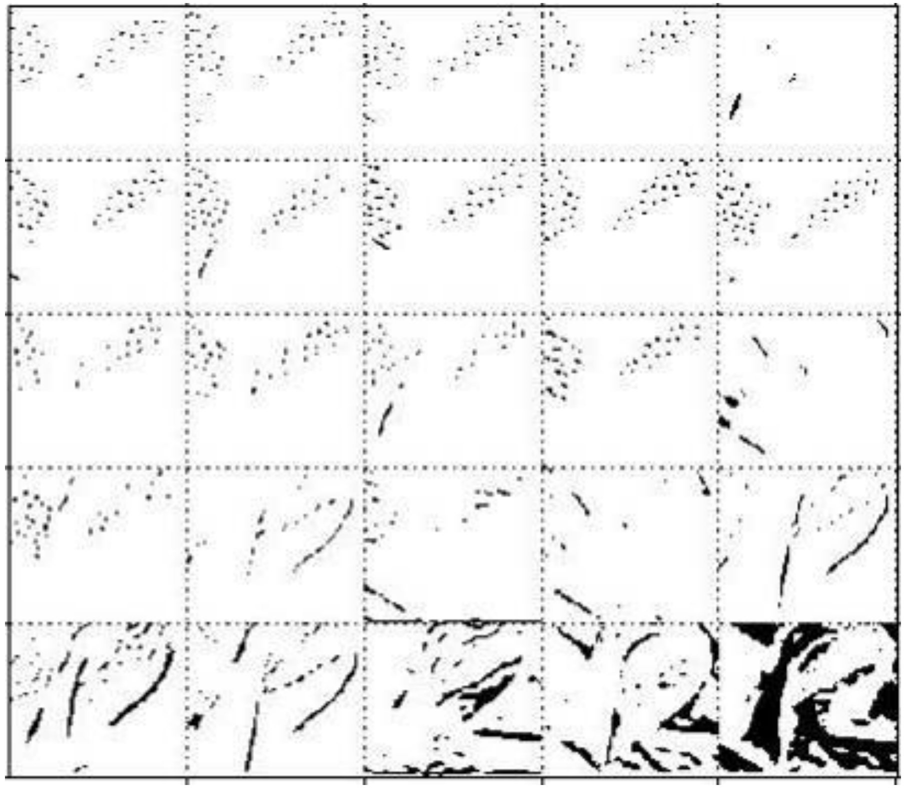
Alyosha Efros, CMU

Textons (Malik et al, IJCV 2001)

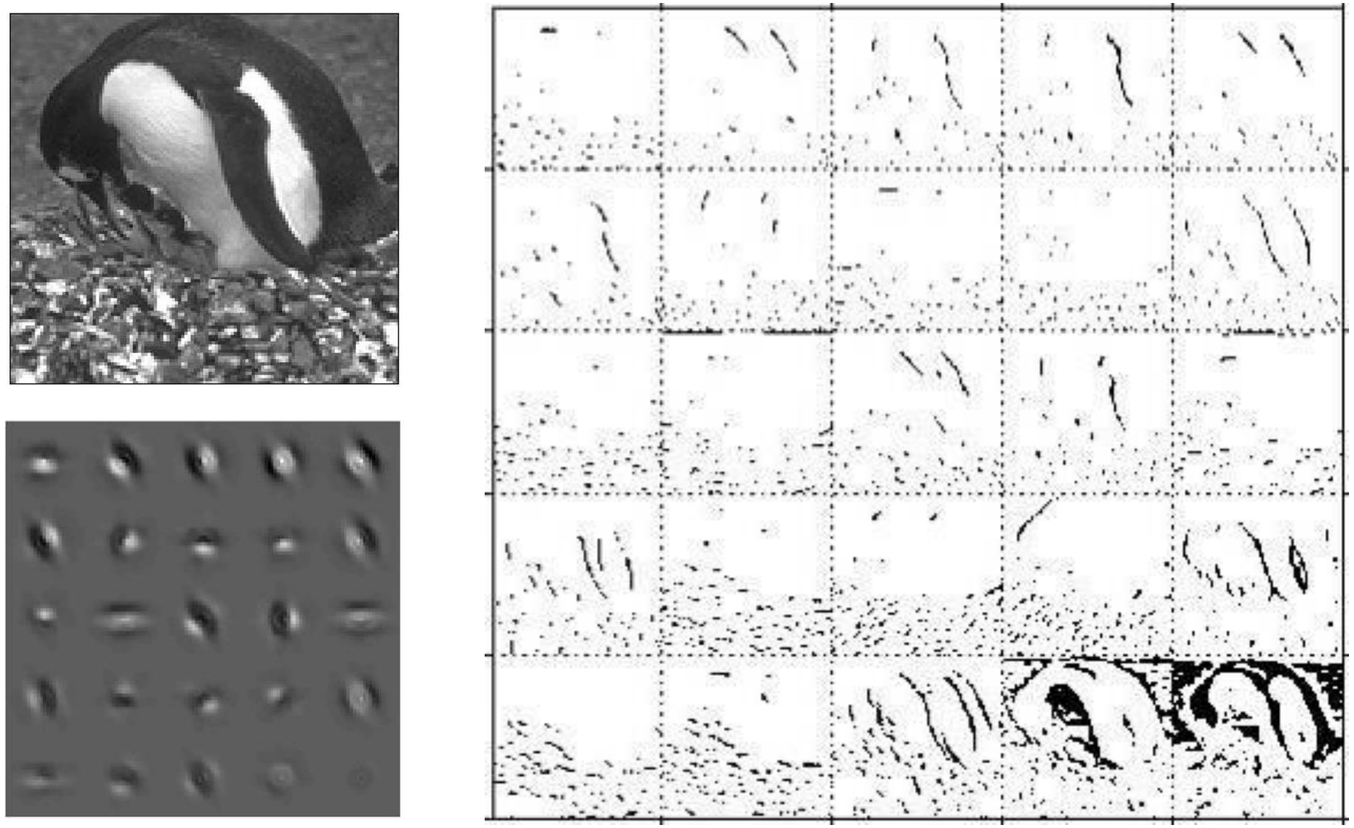
- K-means on vectors of filter responses



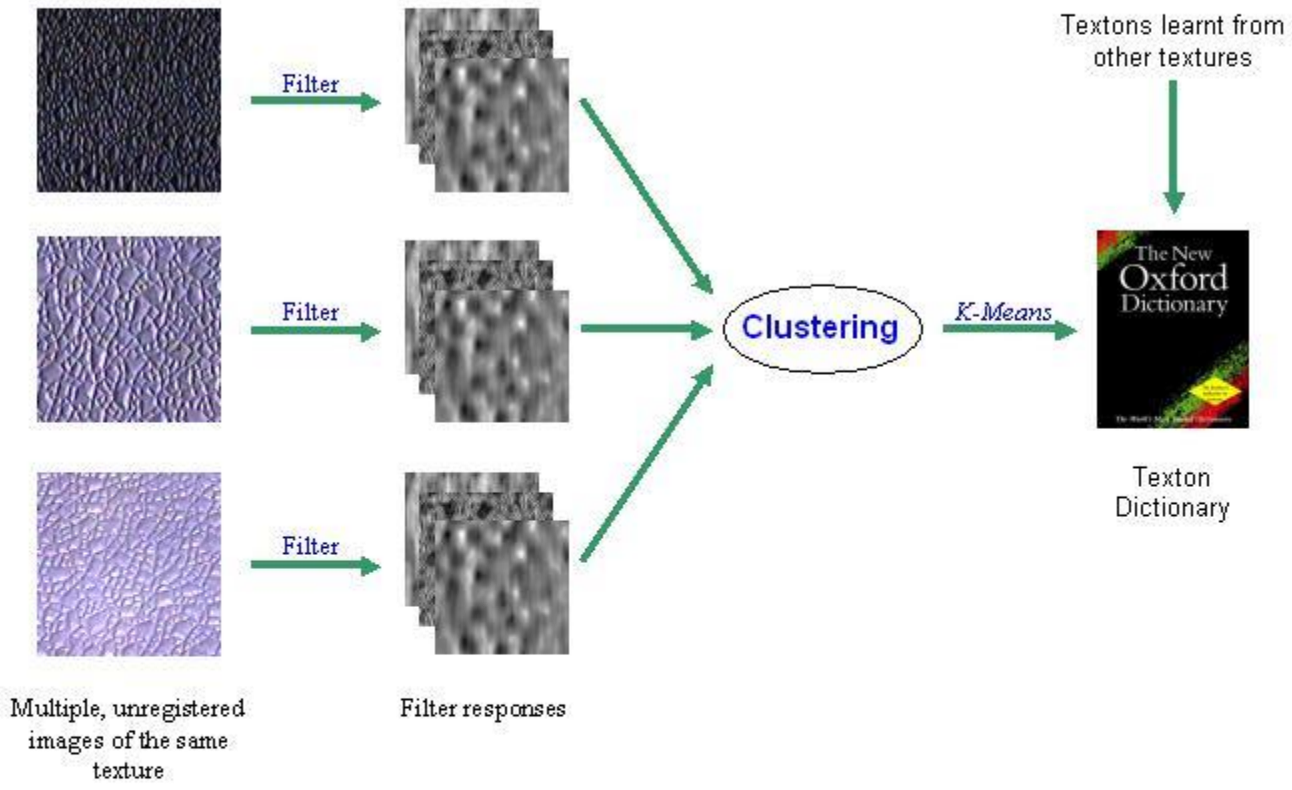




Textons (cont.)

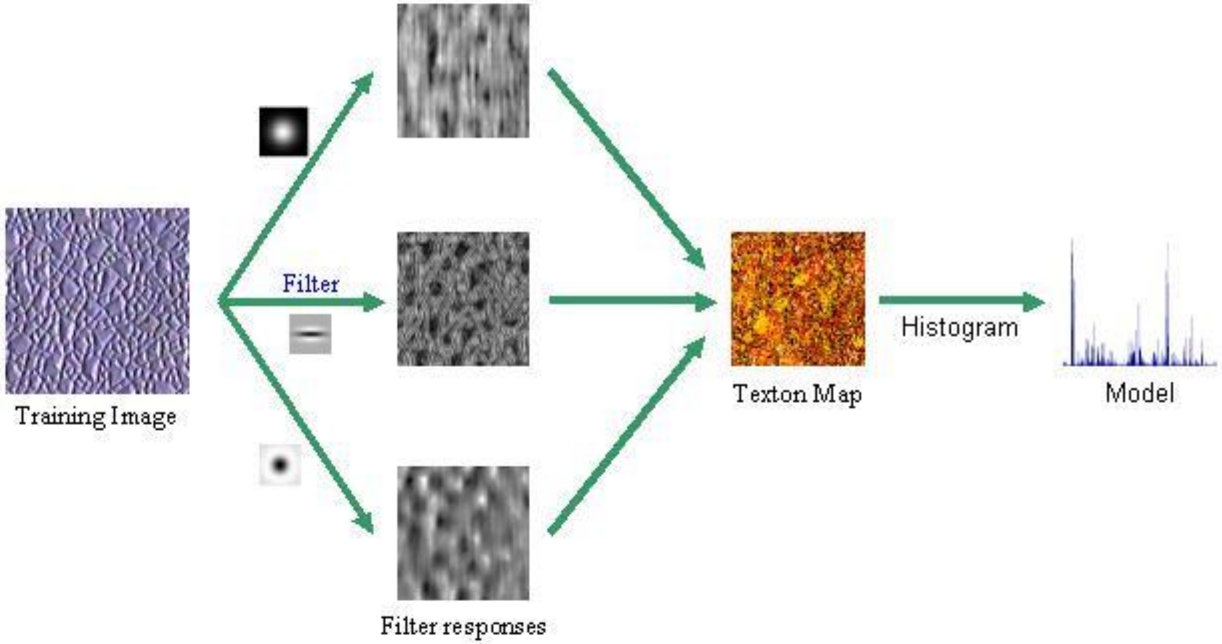


Modelling I – Learning the Texton Dictionary



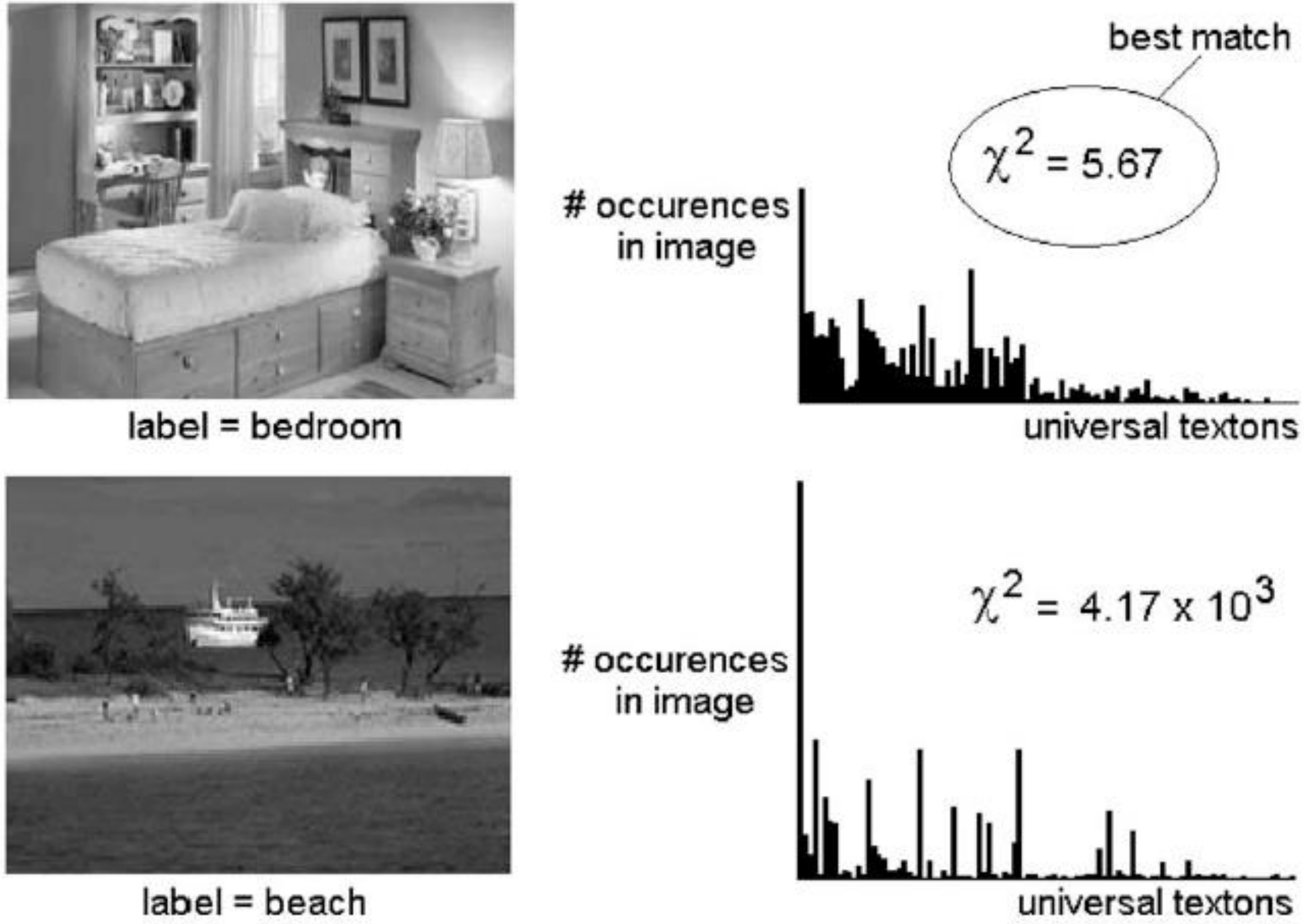
Varma, M. and Zisserman, A., IJCV 2005

Modelling II – Multiple Models Per Texture



Varma, M. and Zisserman, A., IJCV 2005

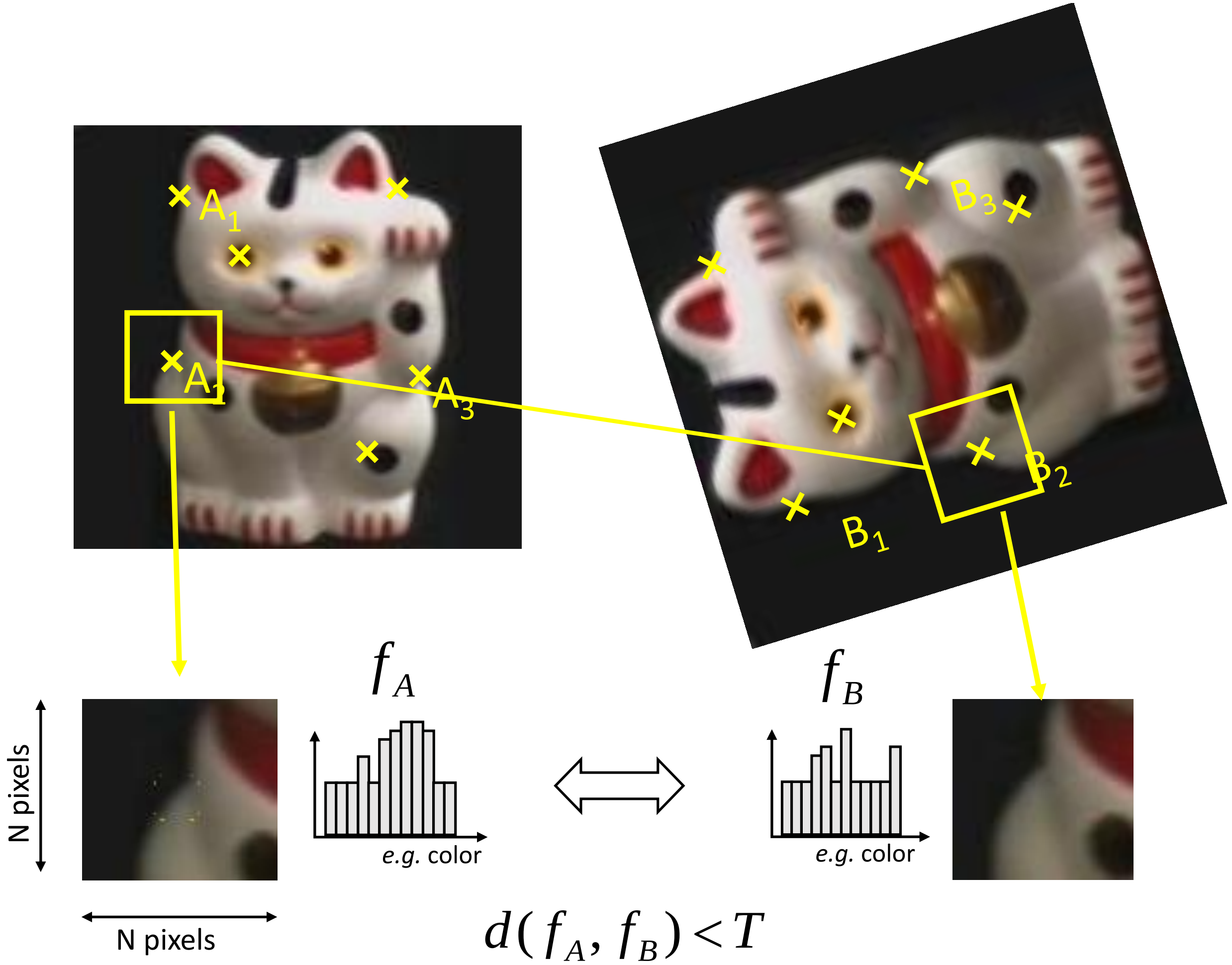
Textons



Walker, Malik, 2004

Torralba, MIT

Revisit Keypoint Matching



1. Find a set of distinctive key-points

2. Define a region around each keypoint

3. Extract and normalize the region content

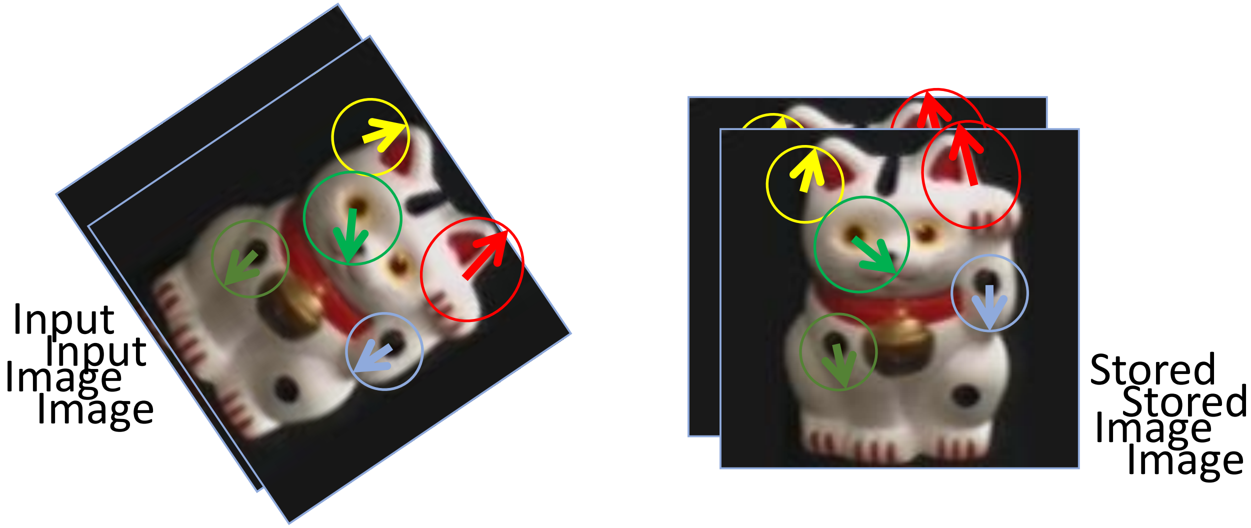
4. Compute a local descriptor from the normalized region

5. Match local descriptors

K. Grauman, B. Leibe

Hayes, Brown

Finding the objects (overview)



1. Match interest points from input image to database image
2. Matched points vote for rough position/orientation/scale of object
3. Find triplets of position/orientation/scale that have at least three votes
4. Compute affine registration and matches using iterative least squares with outlier check
5. Report object if there are at least T matched points

Hayes, Brown

Matching Keypoints

- Want to match keypoints between:
    1. Query image
    2. Stored image containing the object

- Given descriptor x_0, find two nearest neighbors x_1, x_2 with distances d_1, d_2

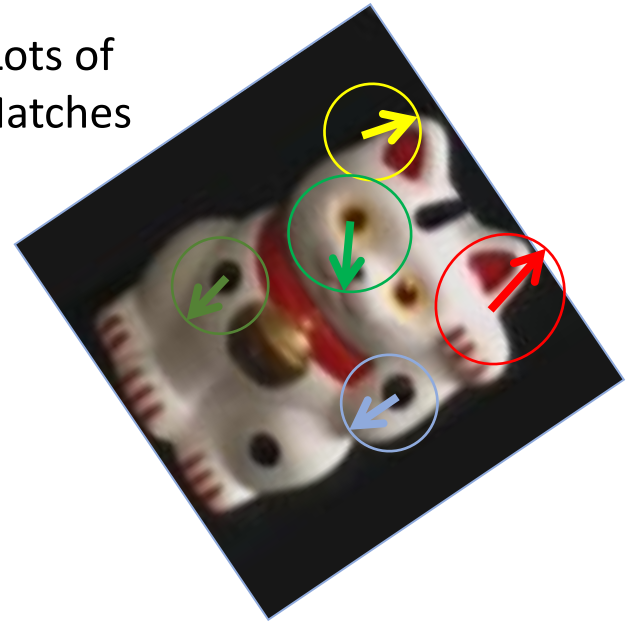
- x_1 matches x_0 if $d_1/d_2 < 0.8$
    - This gets rid of 90% false matches, 5% of true matches in Lowe's study

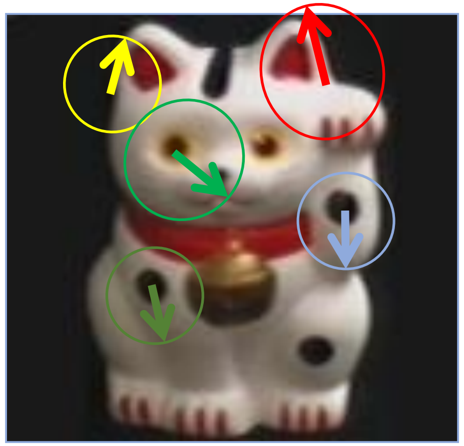
Hayes, Brown

Simple idea

See how many keypoints are close to keypoints in each other image

Lots of Matches





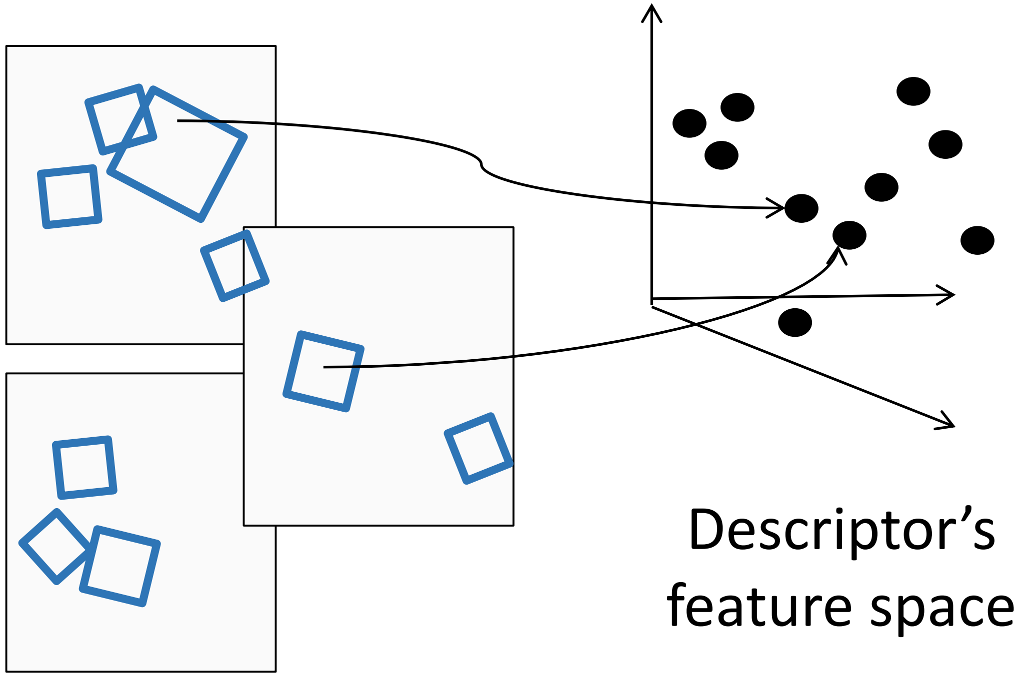
Few or No Matches

But this will be really, really slow!

Hayes, Brown

Indexing local features

- Each patch / region has a descriptor, which is a point in some high-dimensional feature space (e.g., SIFT)

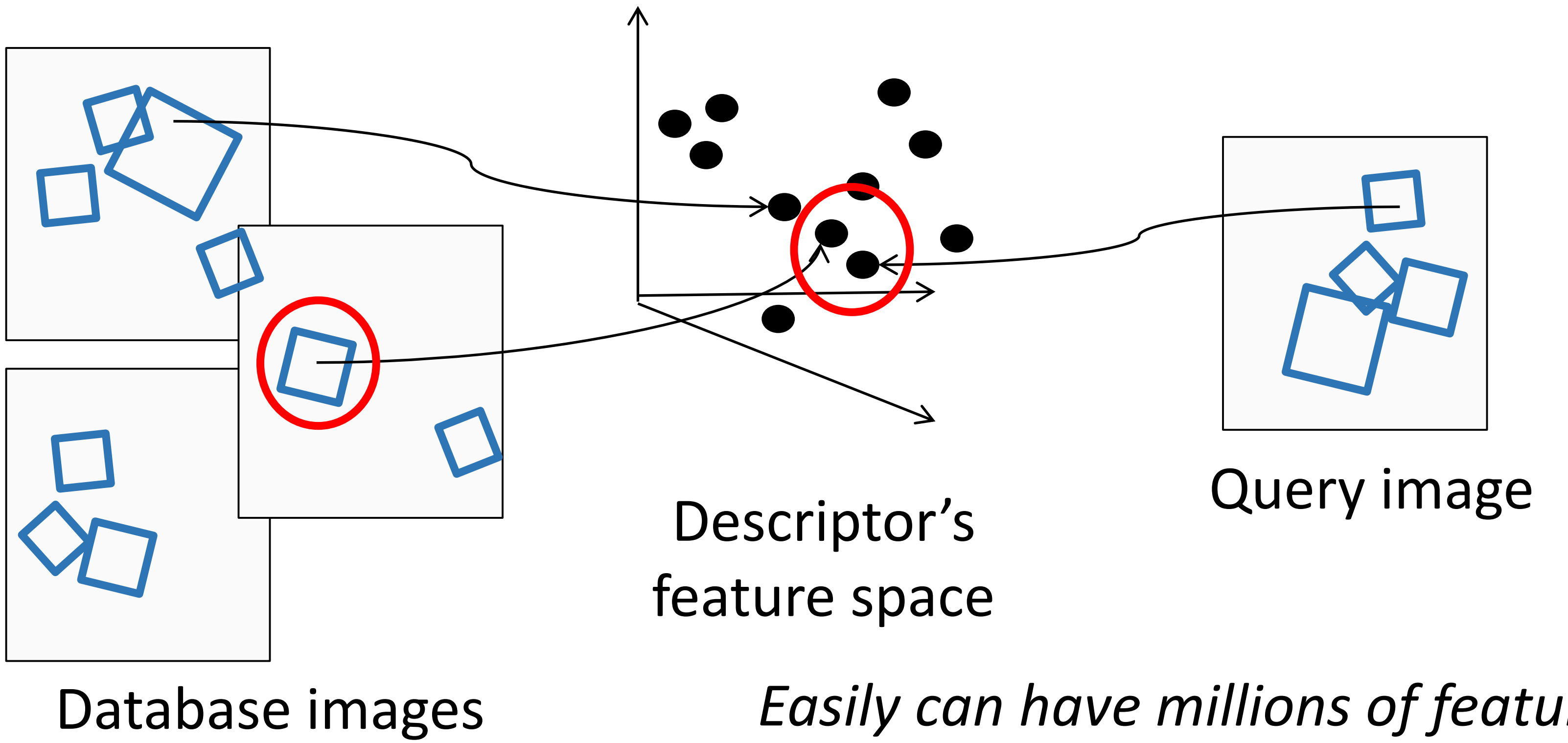


Descriptor's feature space

Kristen Grauman

Indexing local features

- When we see close points in feature space, we have similar descriptors, which indicates similar local content.



Easily can have millions of features to search!

Kristen Grauman

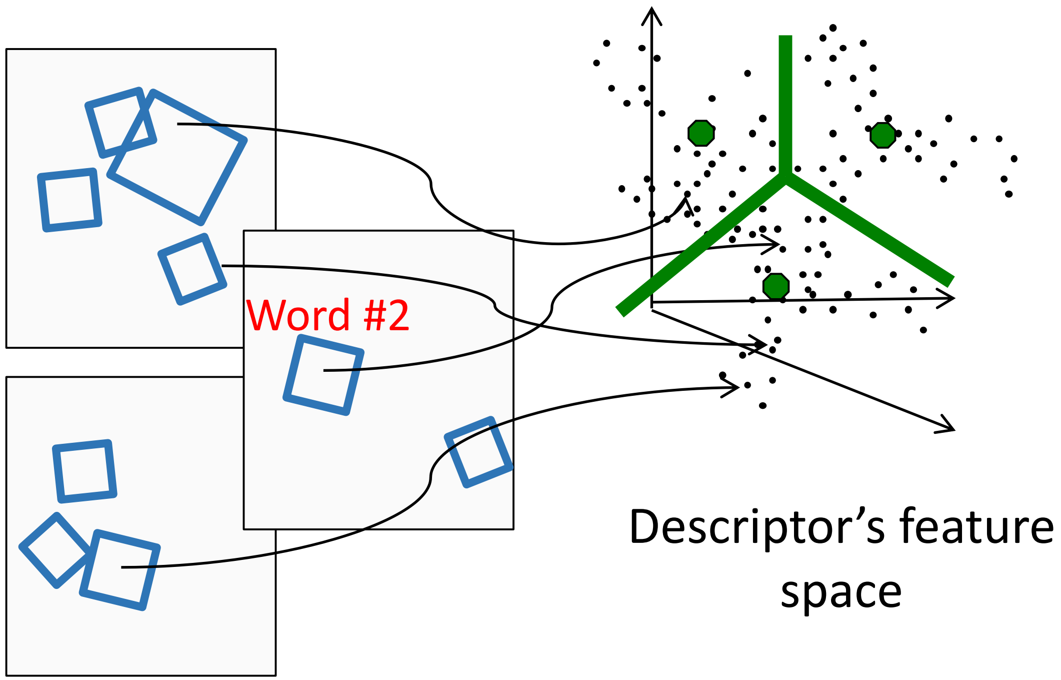
Indexing local features: inverted file index

- For text documents, an efficient way to find all *pages* on which a *word* occurs is to use an index…
- We want to find all *images* in which a *feature* occurs.
- To use this idea, we'll need to map our features to "visual words".

Kristen Grauman

Visual words

- Map high-dimensional descriptors to tokens/words by quantizing the feature space

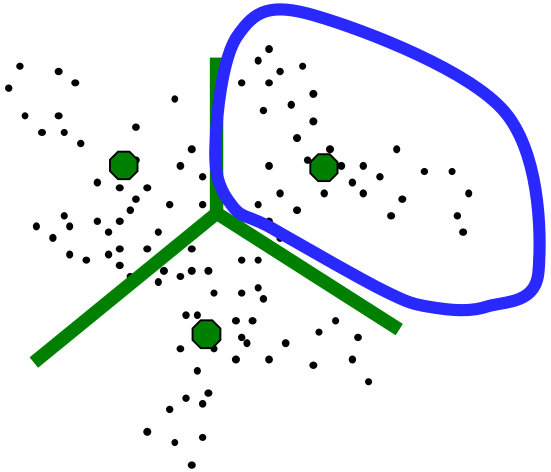


- Quantize via clustering, let cluster centers be the prototype "words"
- Determine which word to assign to each new image region by finding the closest cluster center.

Kristen Grauman

Visual words

- Example: each group of patches belongs to the same visual word



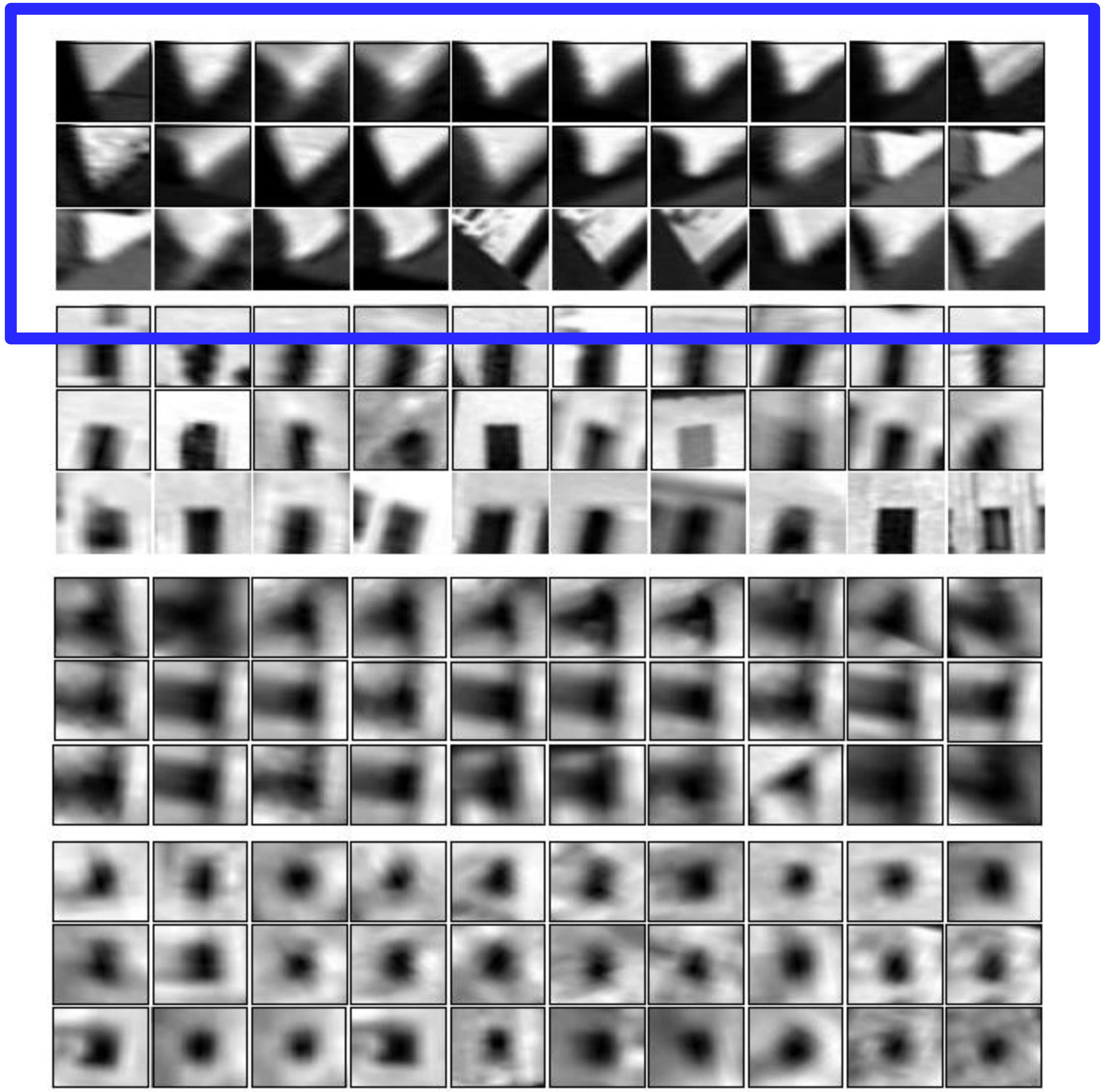
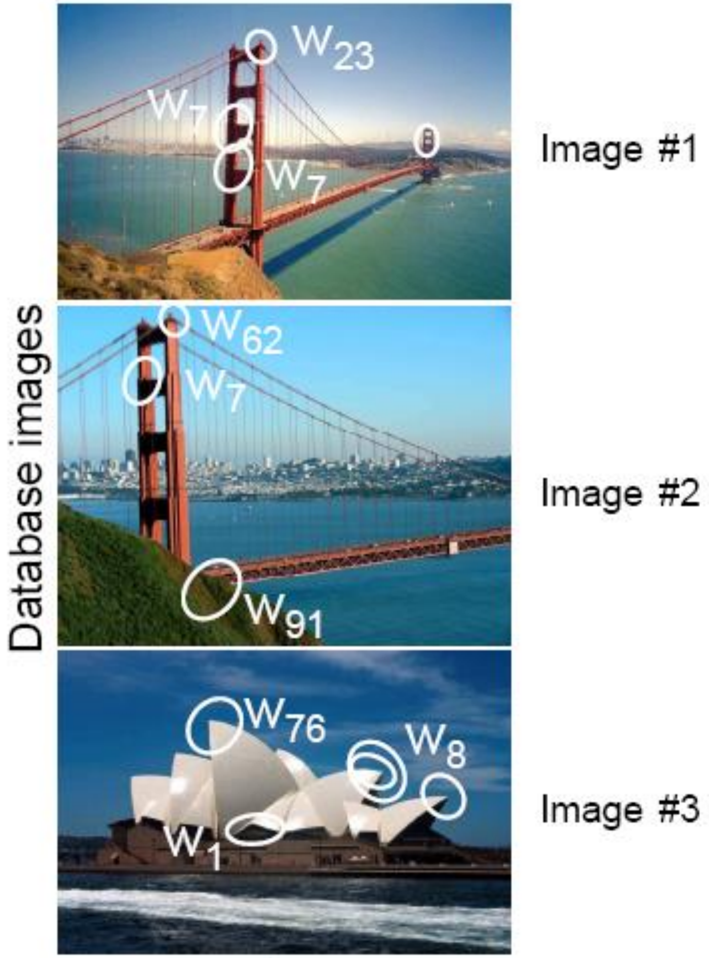


Figure from Sivic & Zisserman, ICCV 2003

Kristen Grauman

Inverted file index



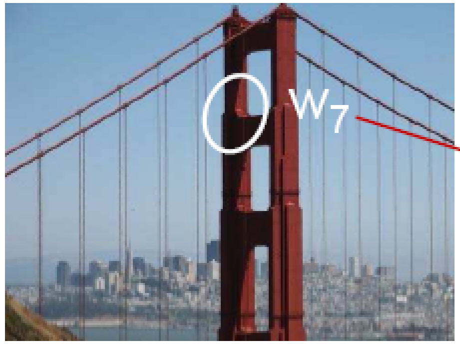
Word #	Image #
1	3
2 …	
7	1, 2
8	3
9	
10 …	
91	2

…

- Database images are loaded into the index mapping words to image numbers

Kristen Grauman

Inverted file index



New query image

Word #	Image #
1	3
2 ...	
7	1, 2
8	3
9	
10 ...	
91	2

...

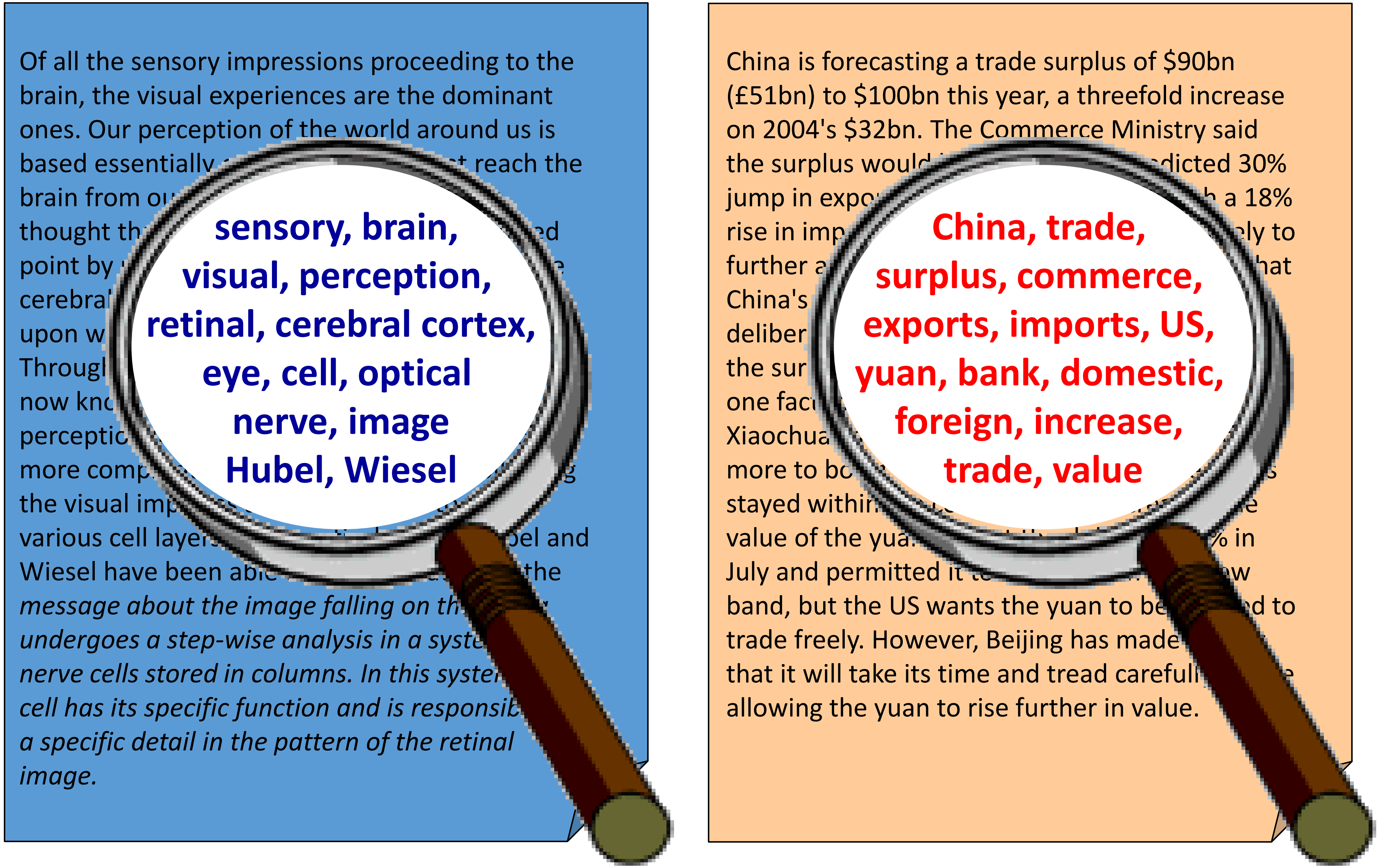
Image #1

Image #2

- New query image is mapped to indices of database images that share a word.

Kristen Grauman

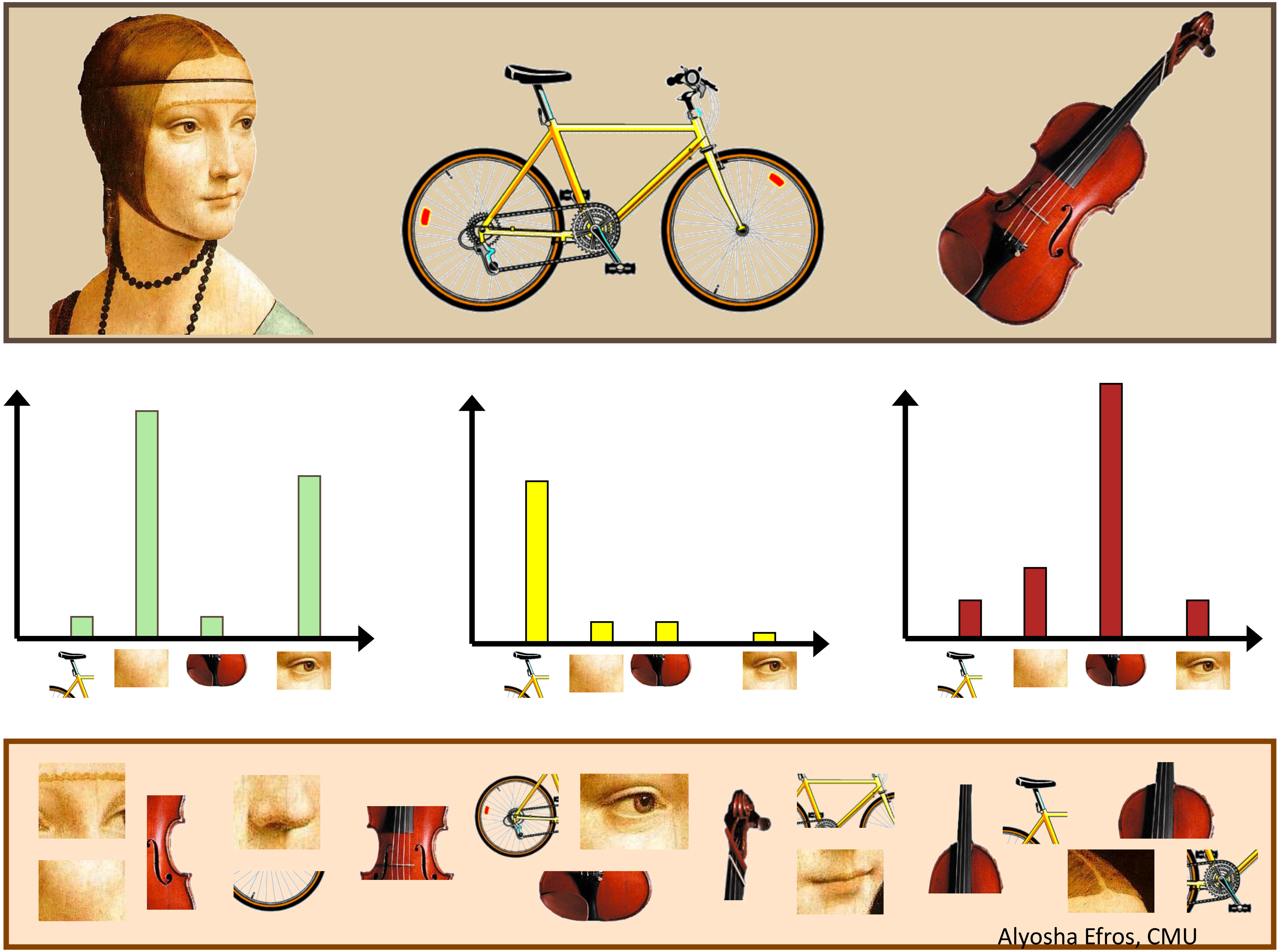
Analogy to documents

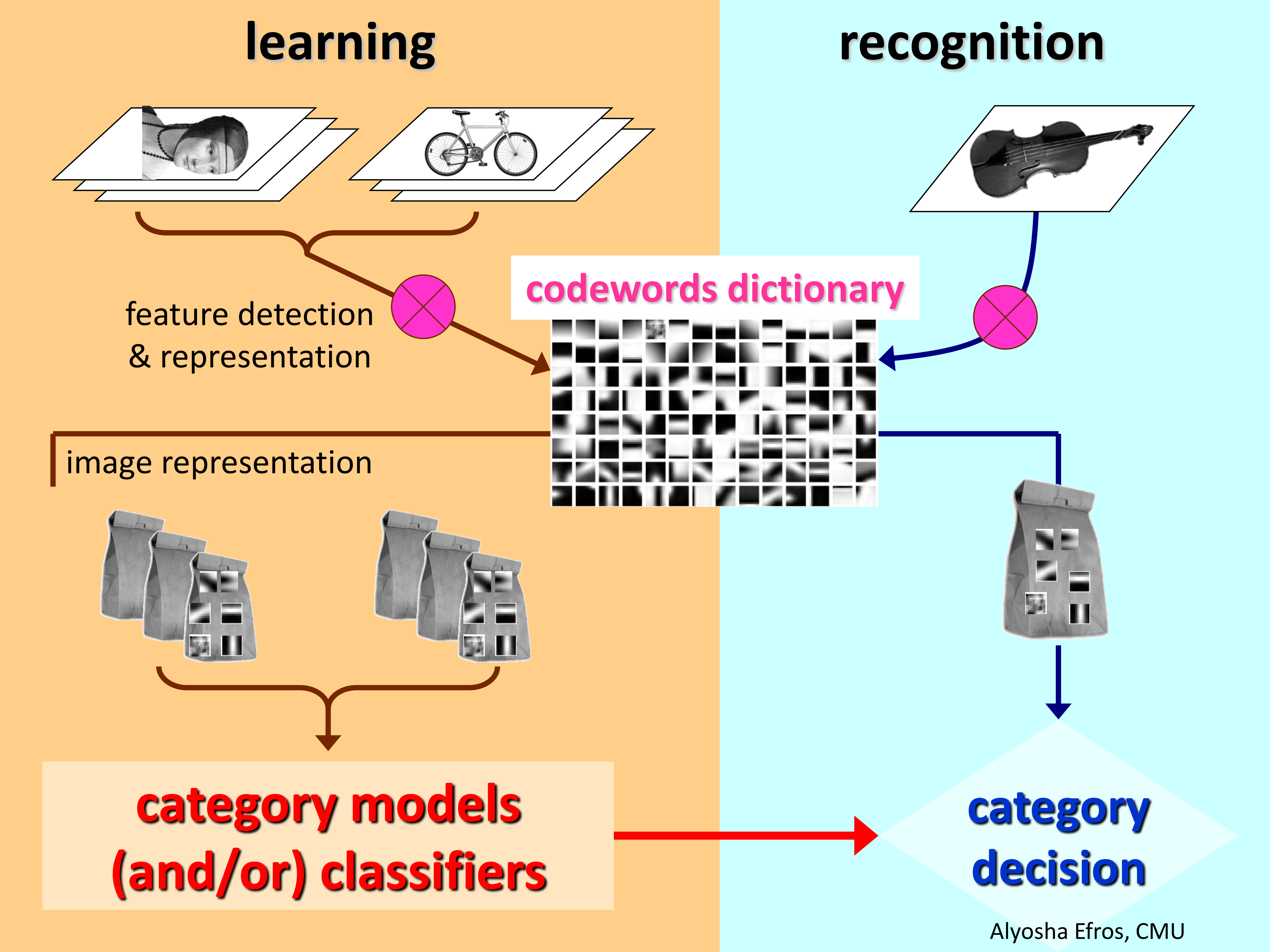


ICCV 2005 short course, L. Fei-Fei

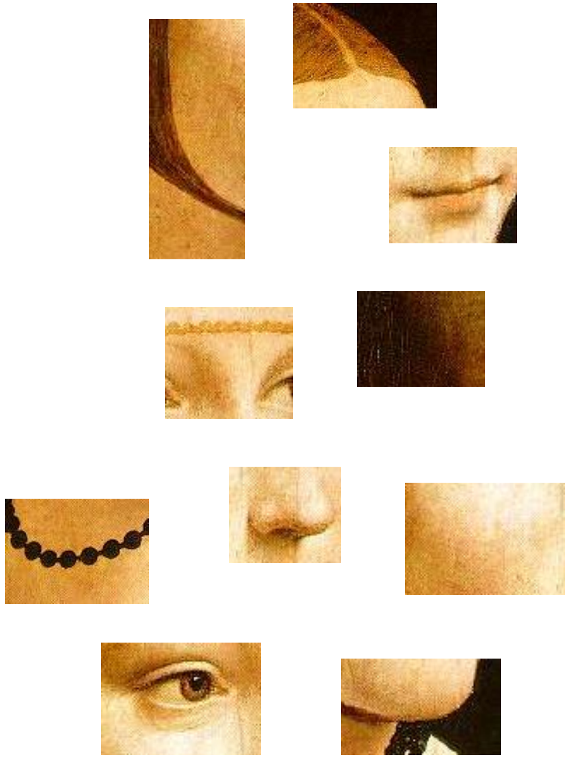
Object → **Bag of ‘words’**

Alyosha Efros, CMU


Alyosha Efros, CMU


learning
recognition
feature detection
& representation
codewords dictionary
image representation
category models
(and/or) classifiers
category
decision
Alyosha Efros, CMU

1.Feature detection and representation



Alyosha Efros, CMU

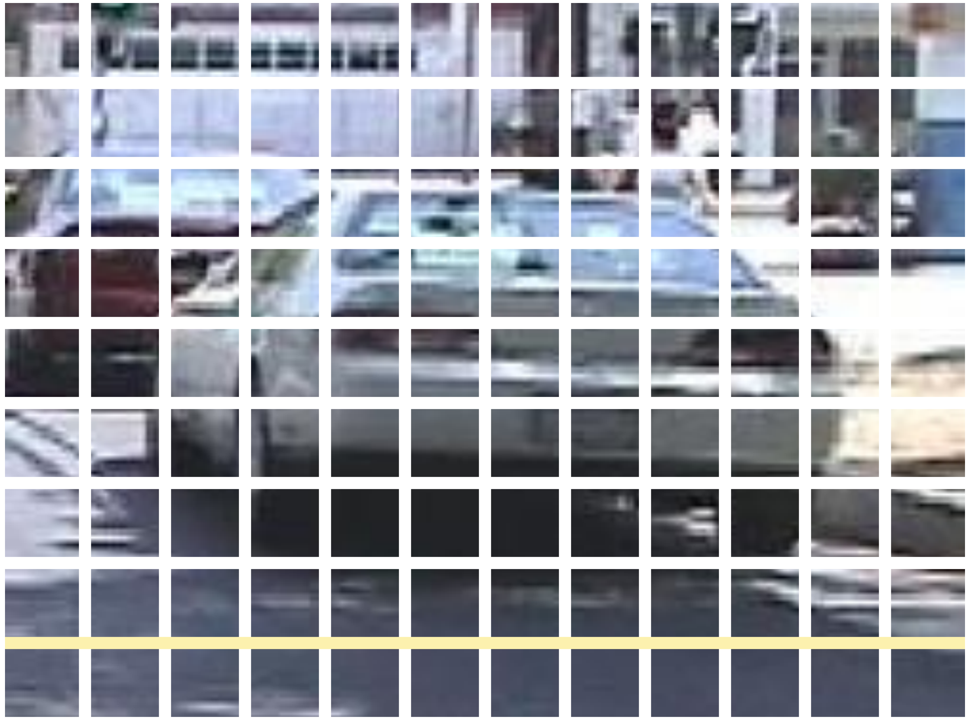
Feature detection

- Sliding Window
  - Leung et al, 1999
  - Viola et al, 1999
  - Renninger et al 2002

Alyosha Efros, CMU

Feature detection

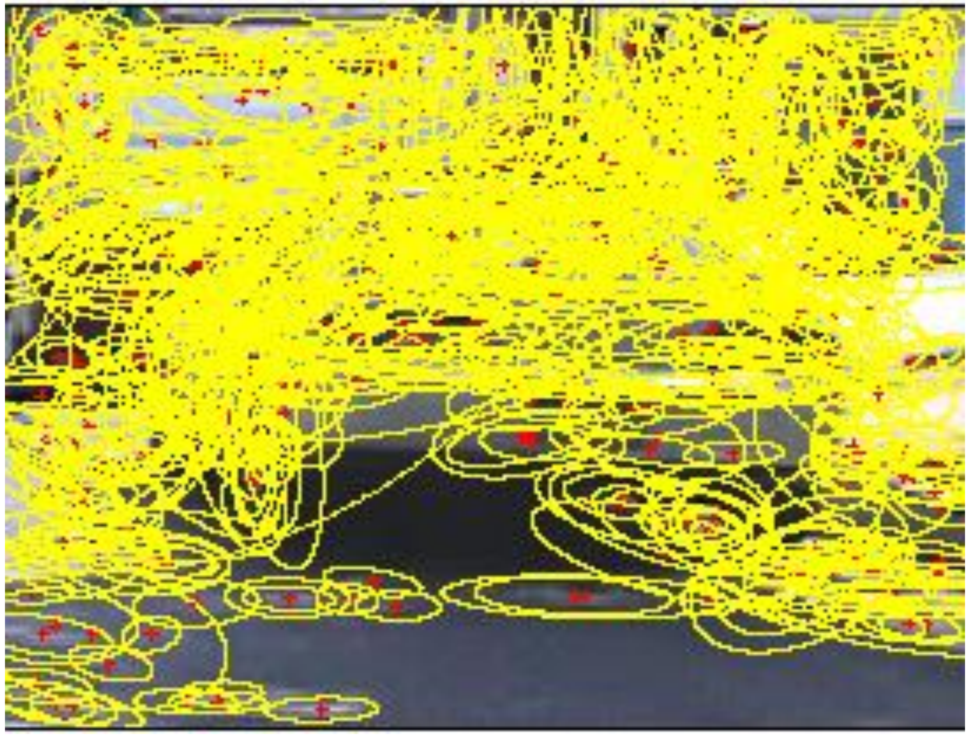
- Sliding Window
  - Leung et al, 1999
  - Viola et al, 1999
  - Renninger et al 2002
- Regular grid
  - Vogel et al. 2003
  - Fei-Fei et al. 2005



Alyosha Efros, CMU

Feature detection

- Sliding Window
  - Leung et al, 1999
  - Viola et al, 1999
  - Renninger et al 2002
- Regular grid
  - Vogel et al. 2003
  - Fei-Fei et al. 2005
- Interest point detector
  - Csurka et al. 2004
  - Fei-Fei et al. 2005
  - Sivic et al. 2005



Alyosha Efros, CMU

Feature detection

- Sliding Window
  - Leung et al, 1999
  - Viola et al, 1999
  - Renninger et al 2002
- Regular grid
  - Vogel et al. 2003
  - Fei-Fei et al. 2005
- Interest point detector
  - Csurka et al. 2004
  - Fei-Fei et al. 2005
  - Sivic et al. 2005
- Other methods
  - Random sampling (Ullman et al. 2002)
  - Segmentation based patches
    - Barnard et al. 2003, Russell et al 2006, etc.)

Alyosha Efros, CMU

Feature Representation

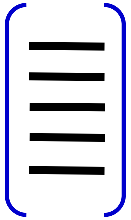
Visual words, aka textons, aka keypoints:

K-means clustered pieces of the image

- Various Representations:
  - Filter bank responses
  - Image Patches
  - SIFT descriptors

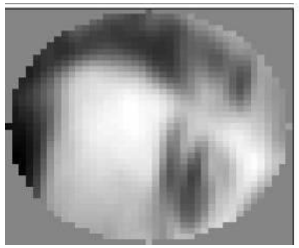
All encode more-or-less the same thing…

Interest Point Features

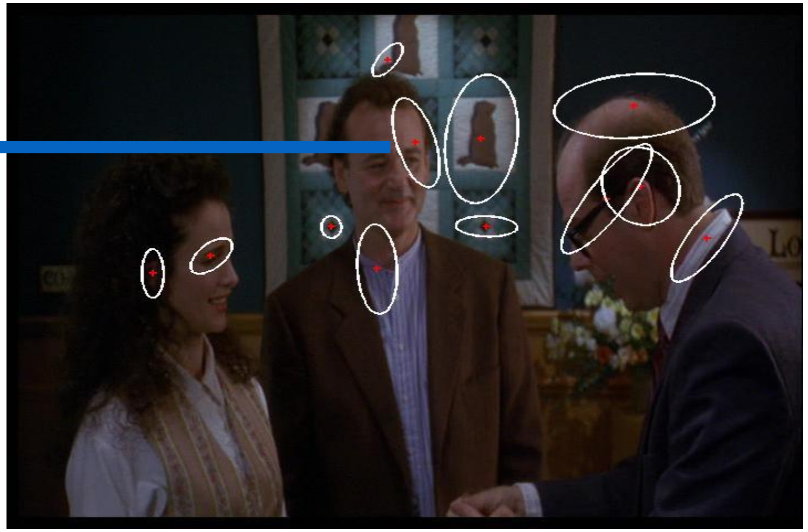


Compute SIFT descriptor

[Lowe’99]



Normalize patch



Detect patches

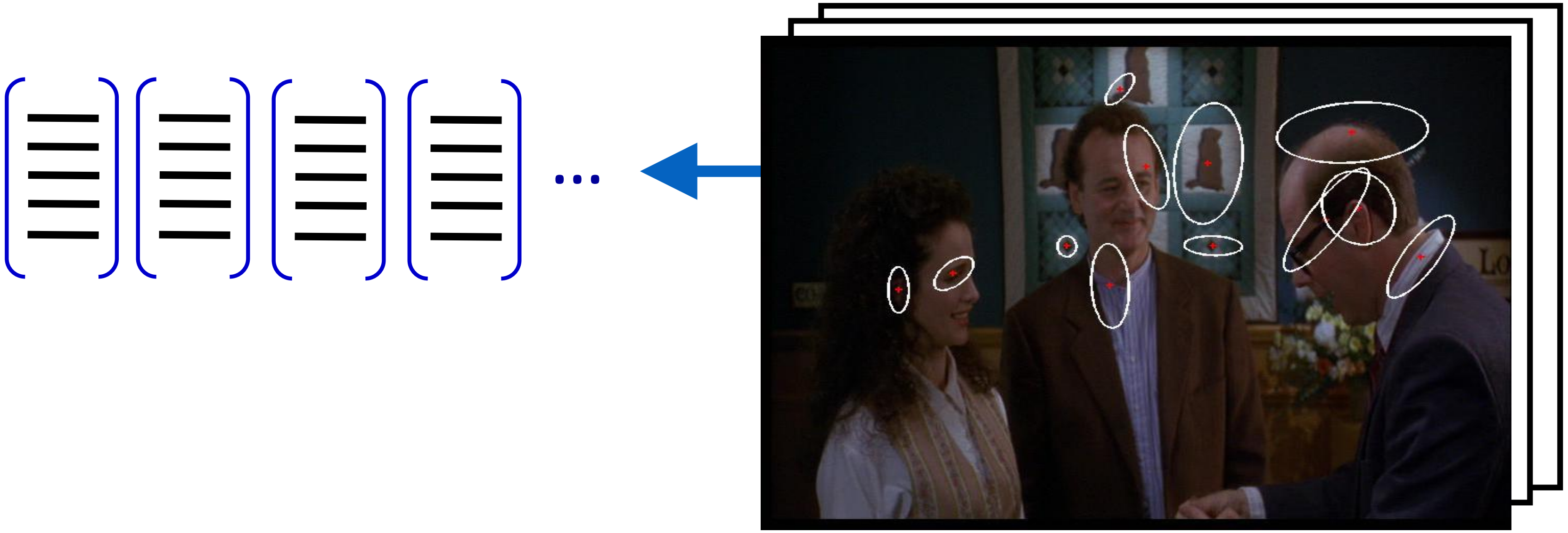
[Mikojaczyk and Schmid ’02]

[Matas et al. ’02]

[Sivic et al. ’03]

Slide credit: Josef Sivic

Interest Point Features

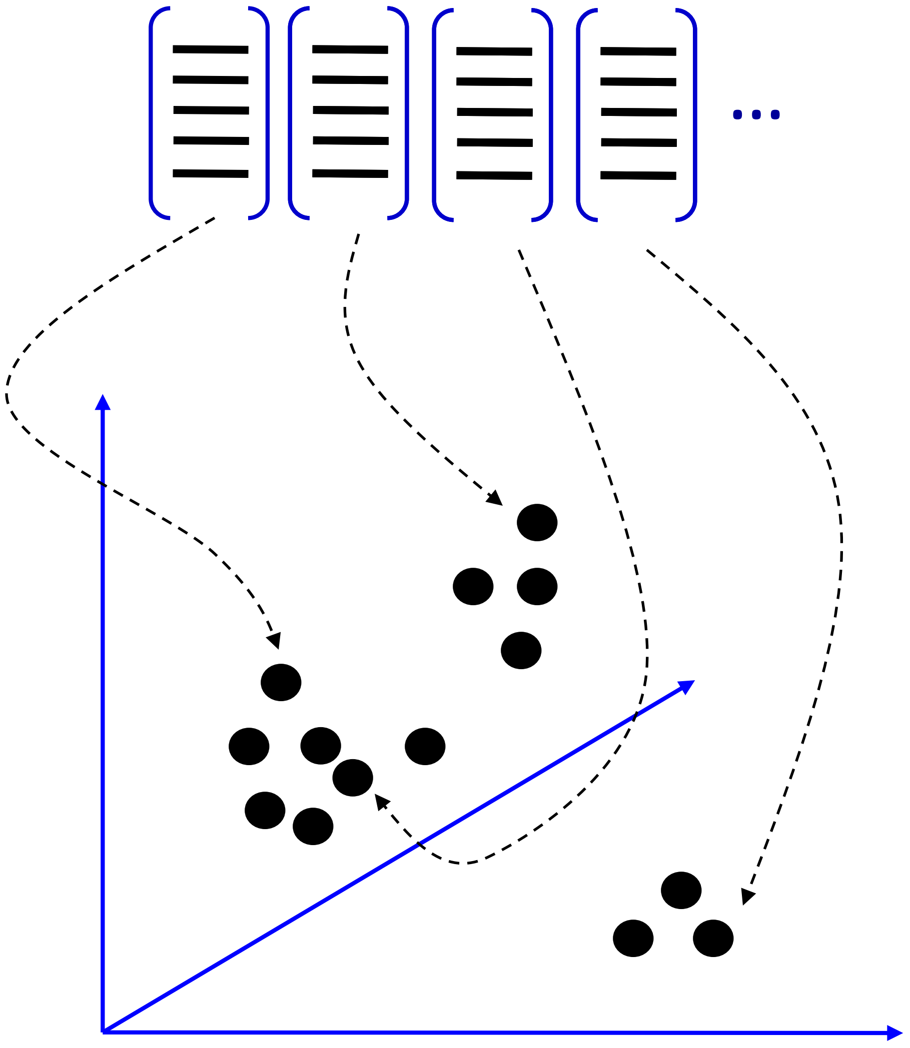


Alyosha Efros, CMU

Patch Features

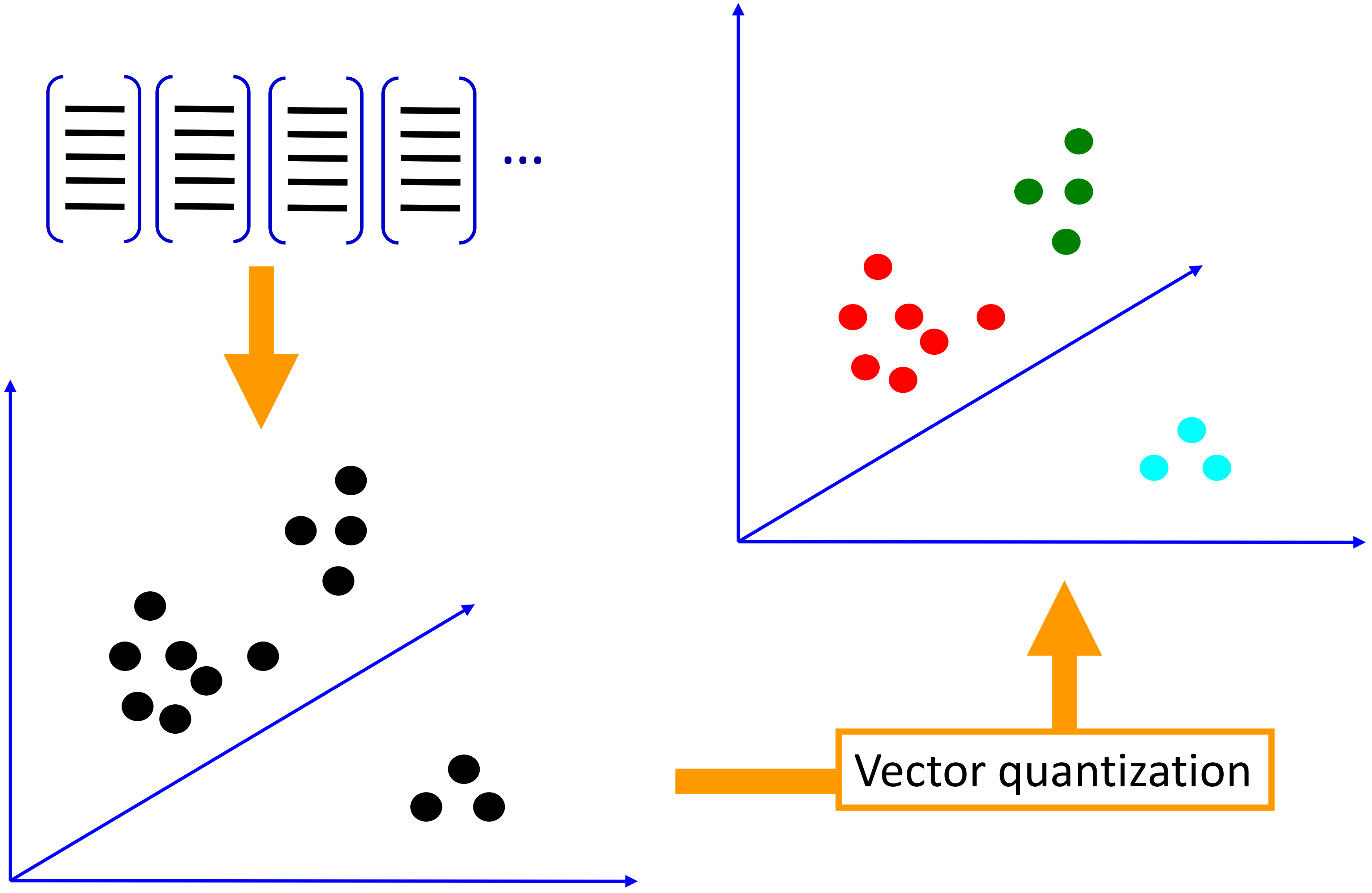
Alyosha Efros, CMU

dictionary formation



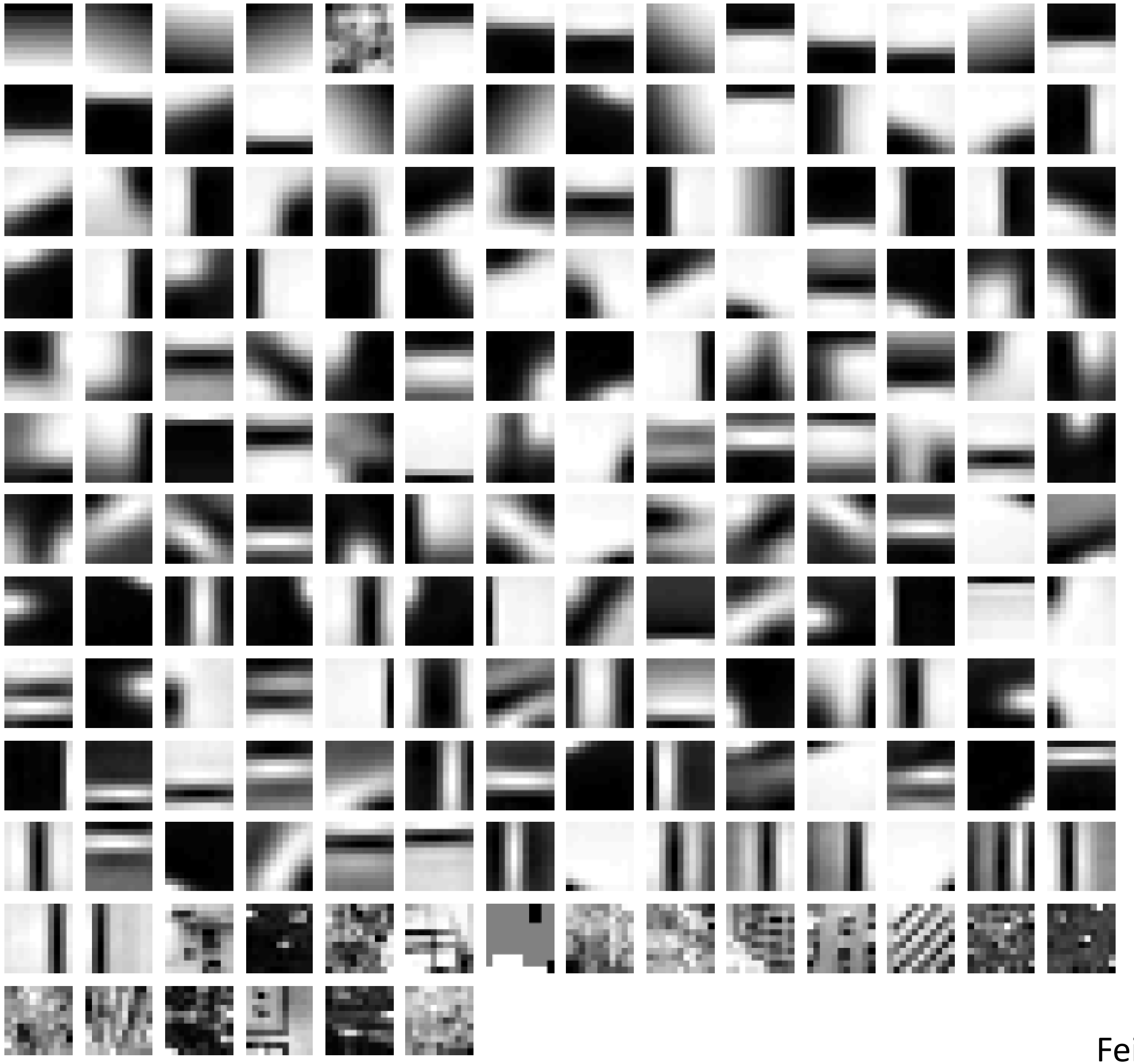
Alyosha Efros, CMU

Clustering (usually k-means)



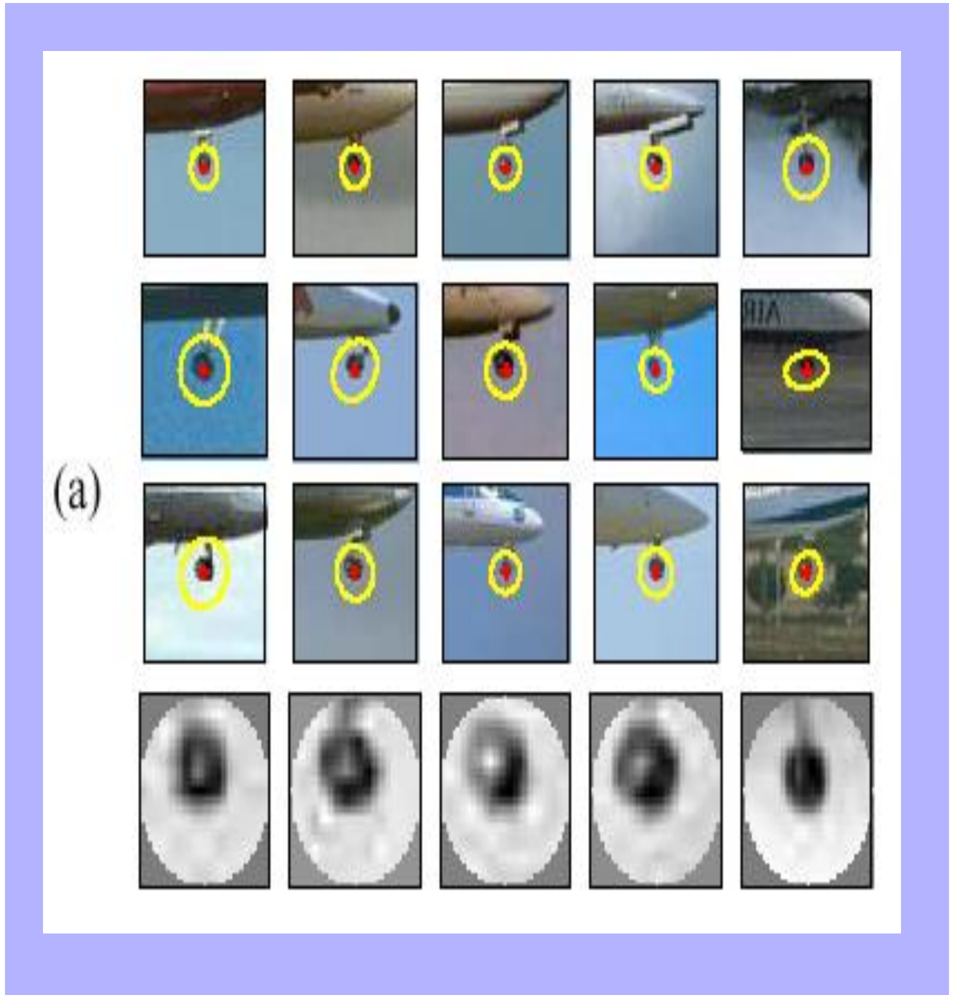
Slide credit: Josef Sivic

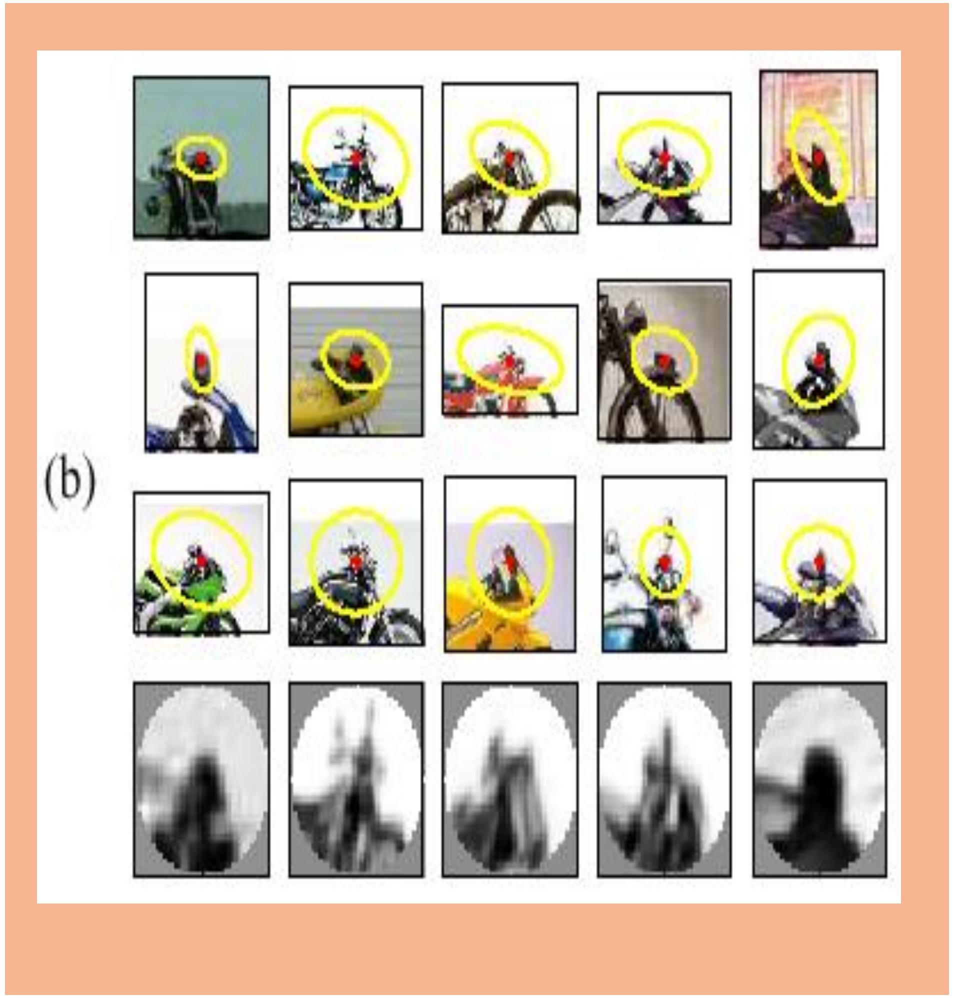
Clustered Image Patches



Fei-Fei et al. 2005

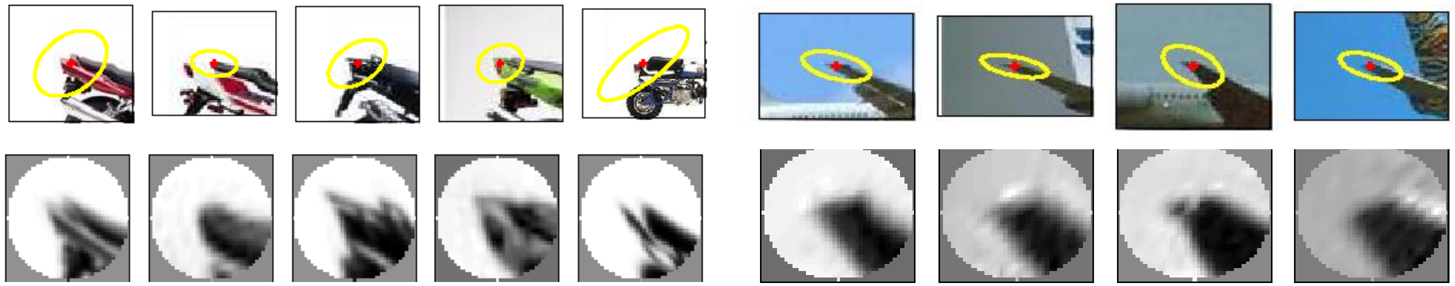
Image patch examples of codewords



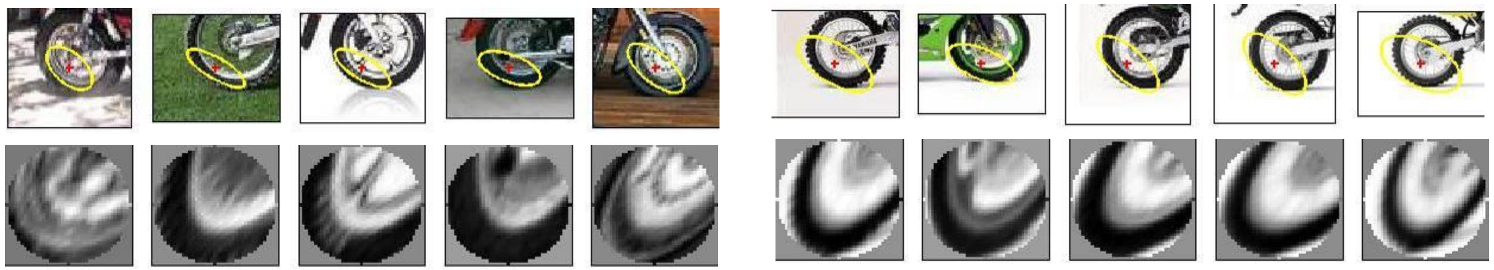


Sivic et al. 2005

Visual synonyms and polysemy



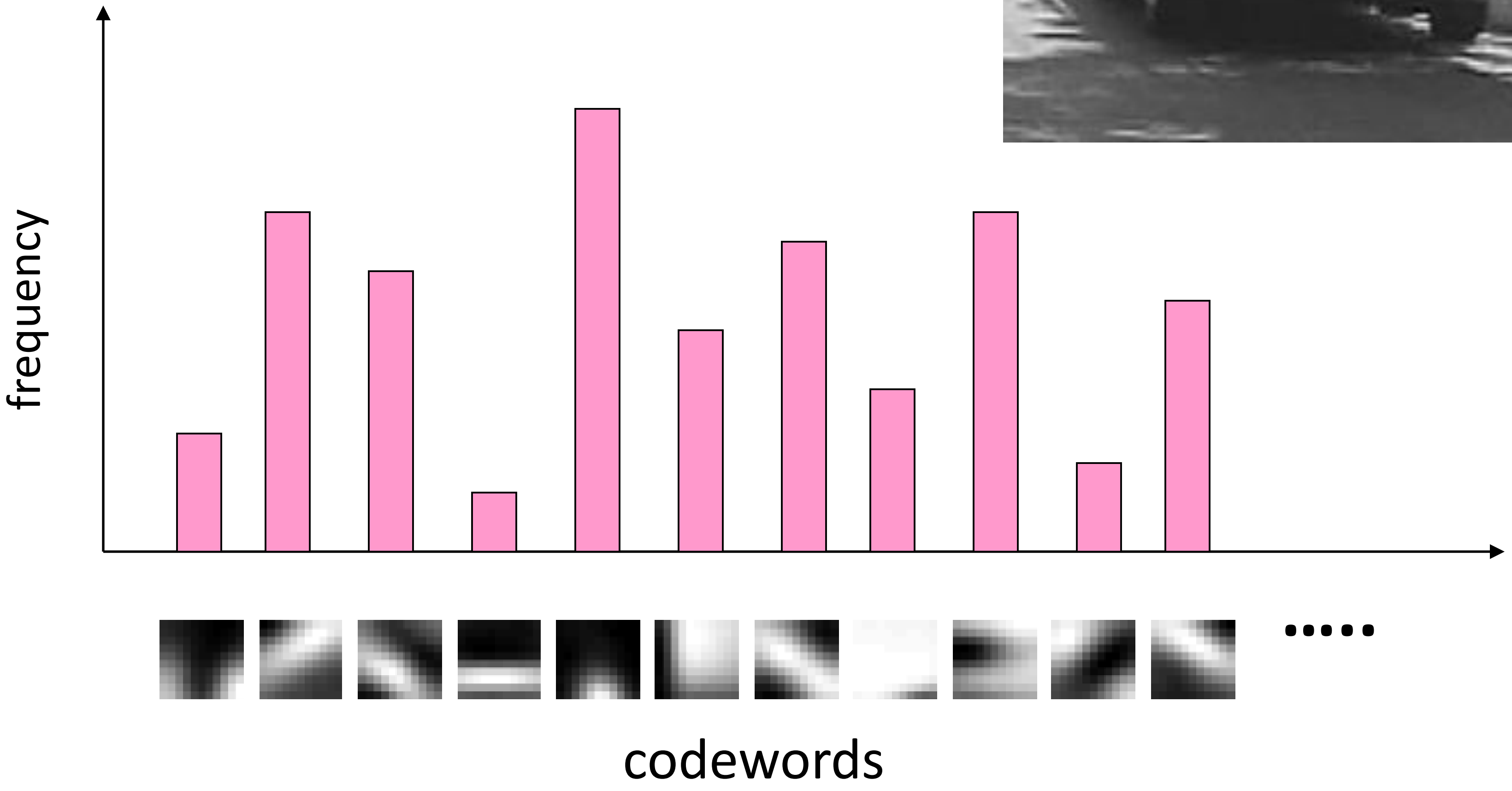
Visual Polysemy. Single visual word occurring on different (but locally similar) parts on different object categories.



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

Alyosha Efros, CMU

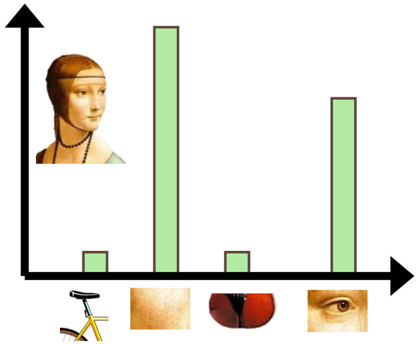
Image representation

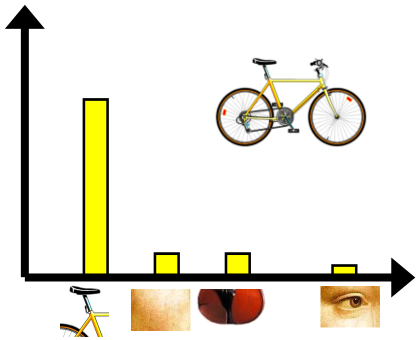


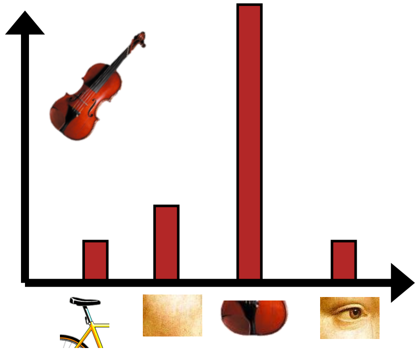
Alyosha Efros, CMU

Bags of visual words

- Summarize entire image based on its distribution (histogram) of word occurrences.
- Analogous to bag of words representation commonly used for documents.





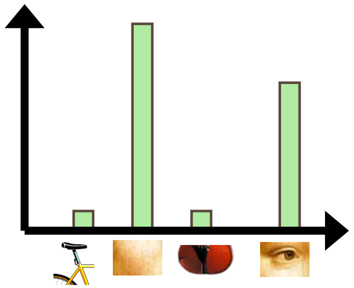


Alyosha Efros, CMU

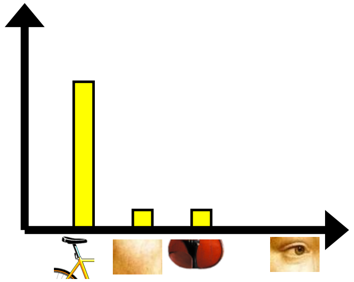
Comparing bags of words

- Rank frames by normalized scalar product between their (possibly weighted) occurrence counts---*nearest neighbor* search for similar images.

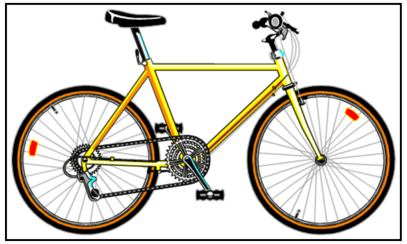
[1 8 1 4]



[5 1 1 0]



$\vec{d}_j$



$\vec{q}$

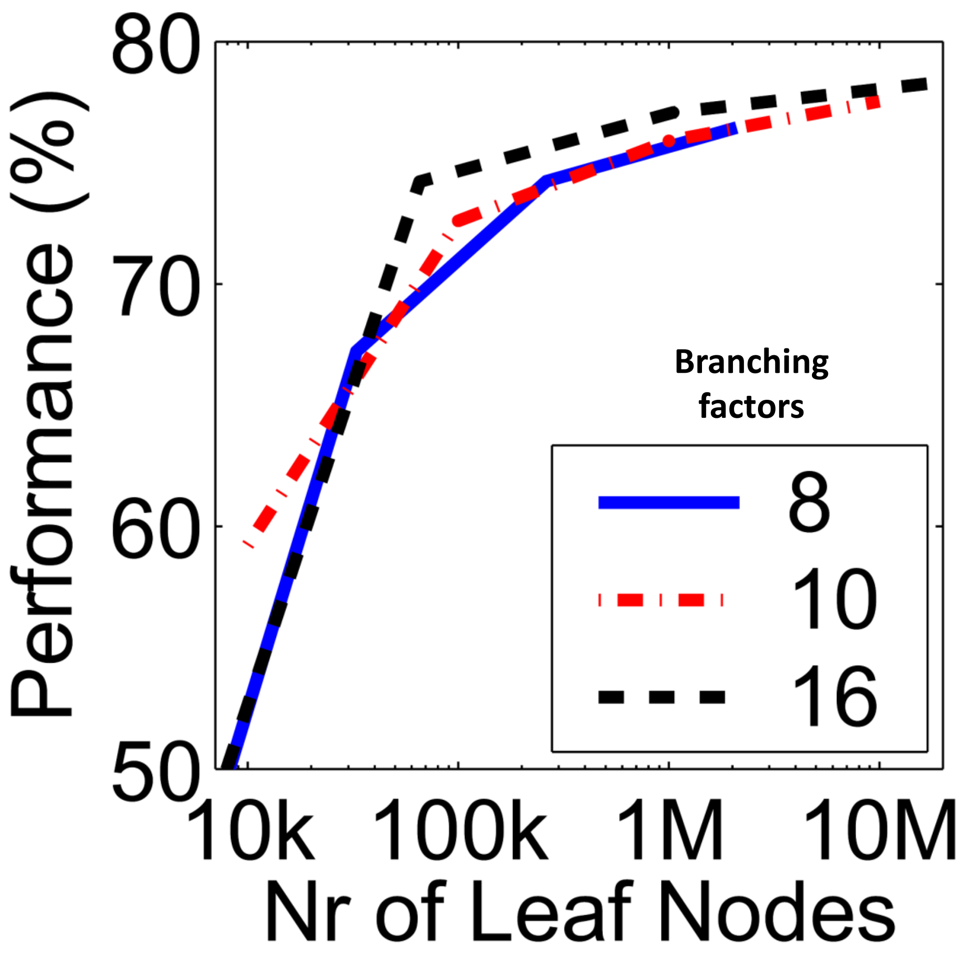
$$sim(d_j, q) = \frac{\langle d_j, q \rangle}{\|d_j\| \|q\|}$$

$$= \frac{\sum_{i=1}^{V} d_j(i) * q(i)}{\sqrt{\sum_{i=1}^{V} d_j(i)^2} * \sqrt{\sum_{i=1}^{V} q(i}$$

for vocabulary of *V* words

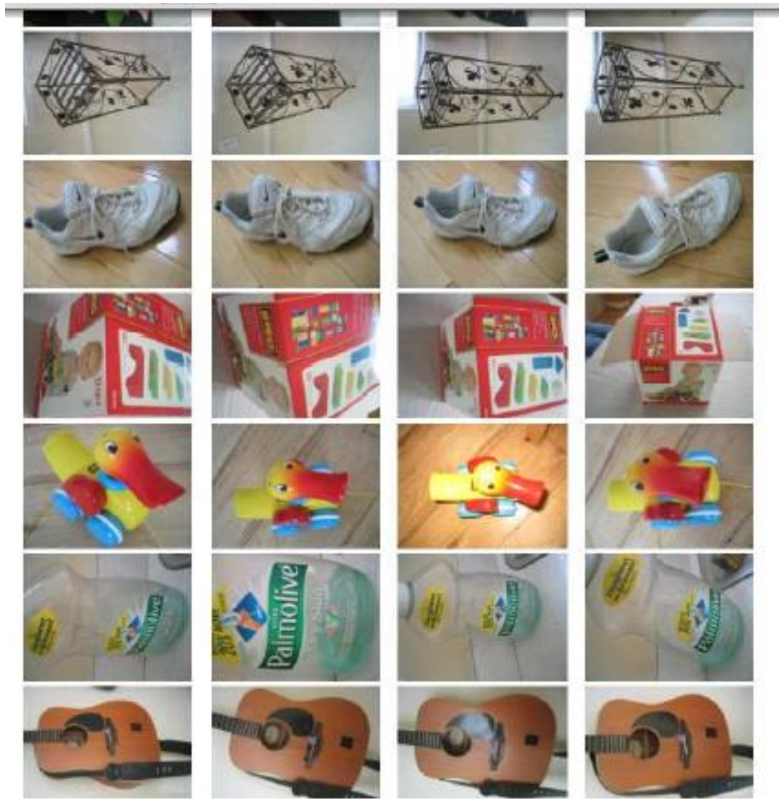
Kristen Grauman

Vocabulary size



Influence on performance, sparsity

Results for recognition task with 6347 images



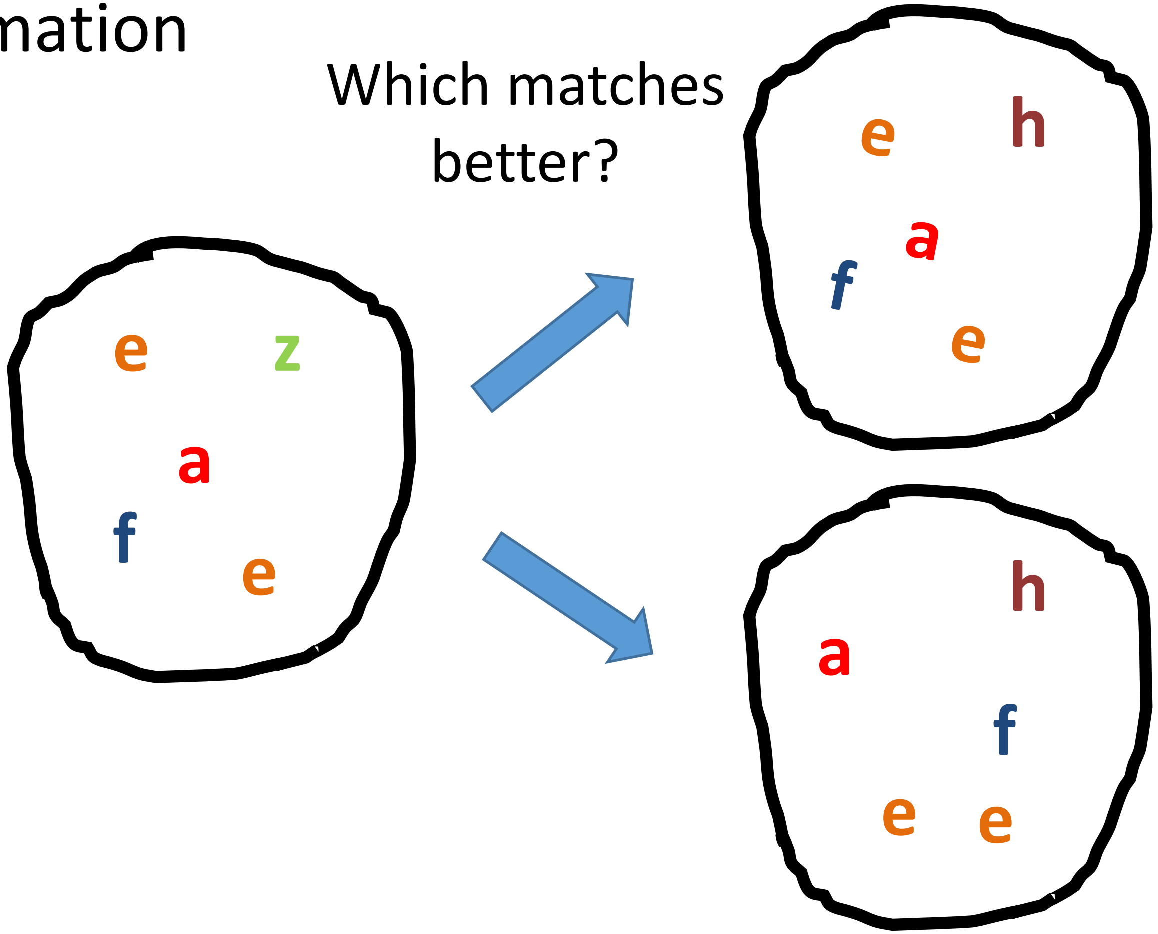
Nister & Stewenius, CVPR 2006

Kristen Grauman

Can we be more accurate?

So far, we treat each image as containing a “bag of words”, with no spatial information

Which matches better?



Hays, Brown

Can we be more accurate?

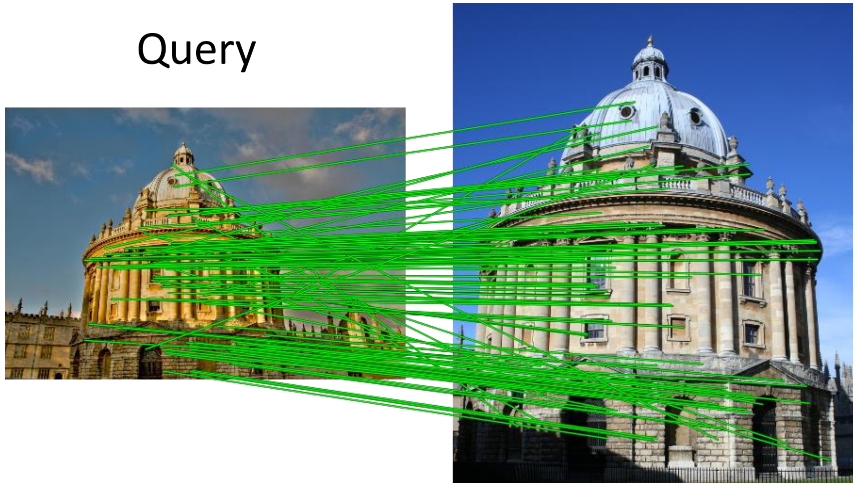
So far, we treat each image as containing a “bag of words”, with no spatial information

Real objects have consistent geometry

Hays, Brown

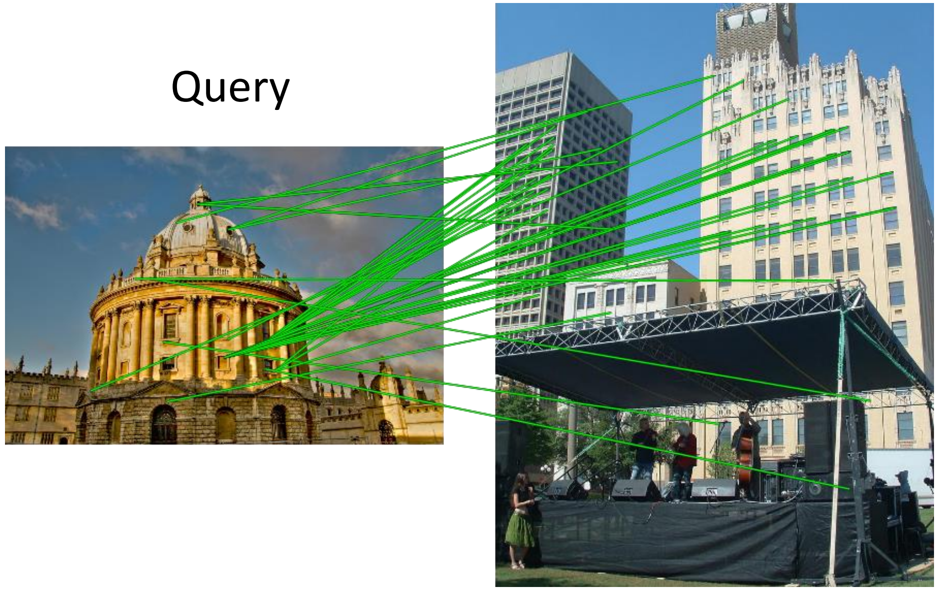
Spatial Verification

Query



DB image with high BoW similarity

Query



DB image with high BoW similarity

Both image pairs have many visual words in common.

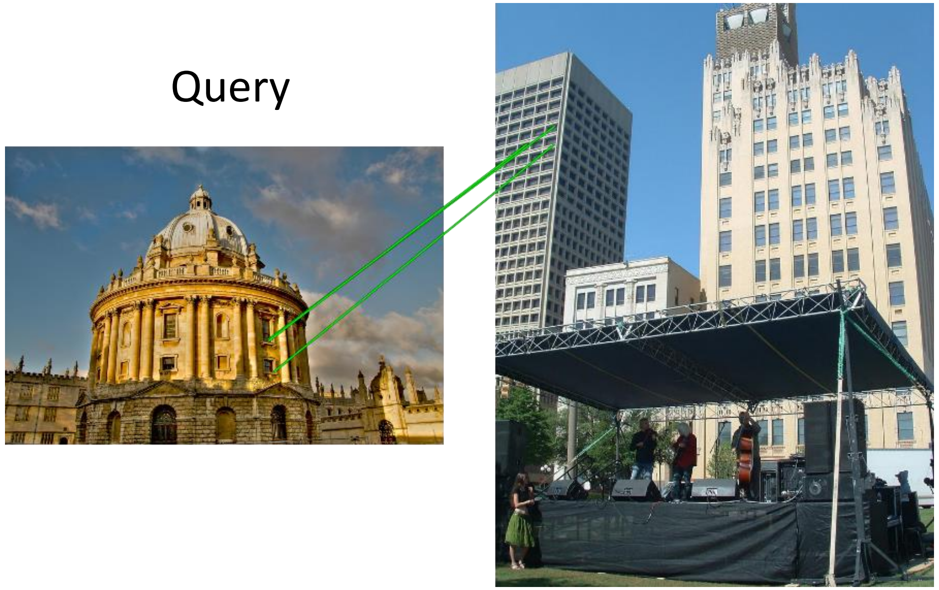
Slide credit: Ondrej Chum

Spatial Verification

Query

DB image with high BoW similarity

Query

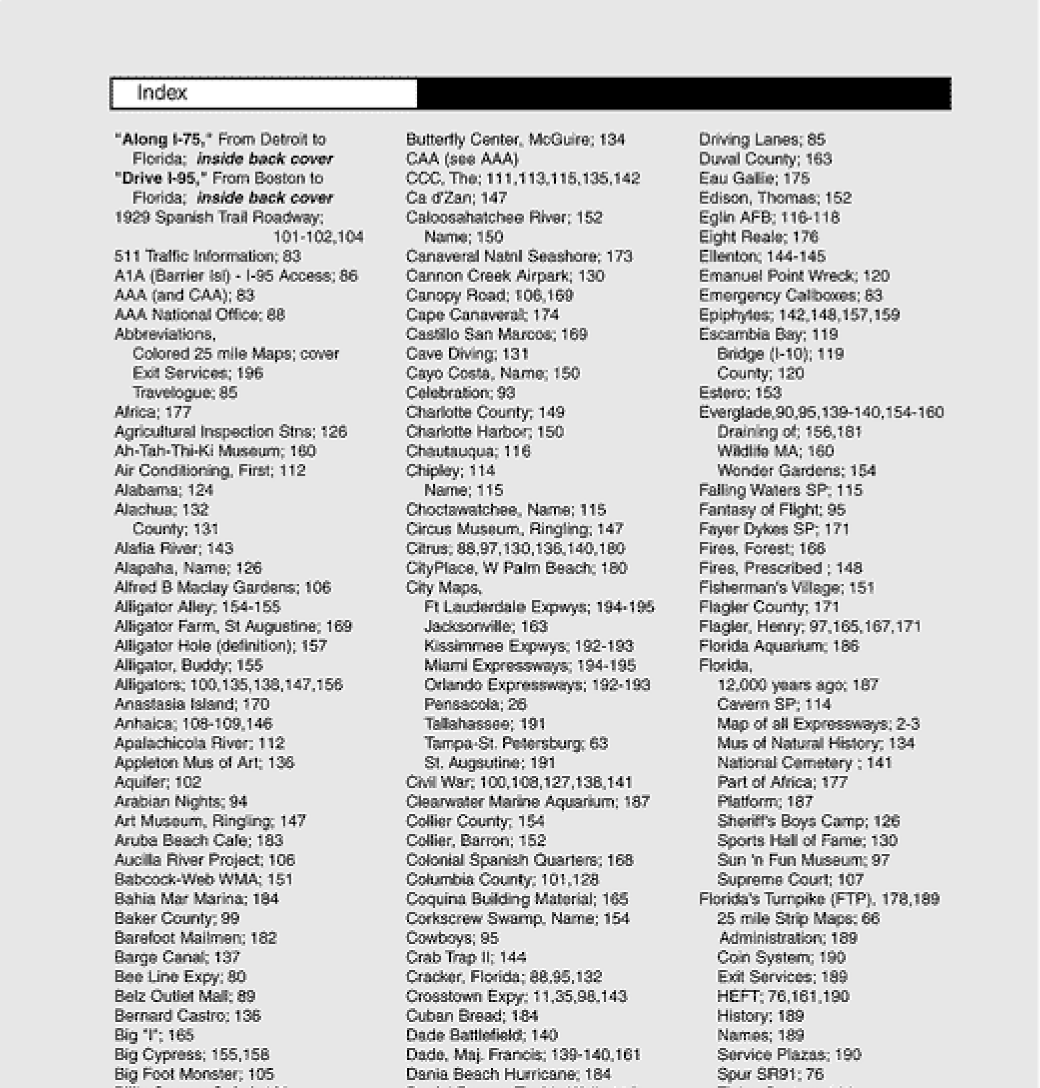


DB image with high BoW similarity

Only some of the matches are mutually consistent

Slide credit: Ondrej Chum

What else can we borrow from text retrieval?



Index

tf-idf weighting

- **T**erm **f**requency – **i**nverse **d**ocument **f**requency
- Describe frame by frequency of each word within it, downweight words that appear often in the database
- (Standard weighting for text retrieval)

Number of occurrences of word i in document d

Number of words in document d

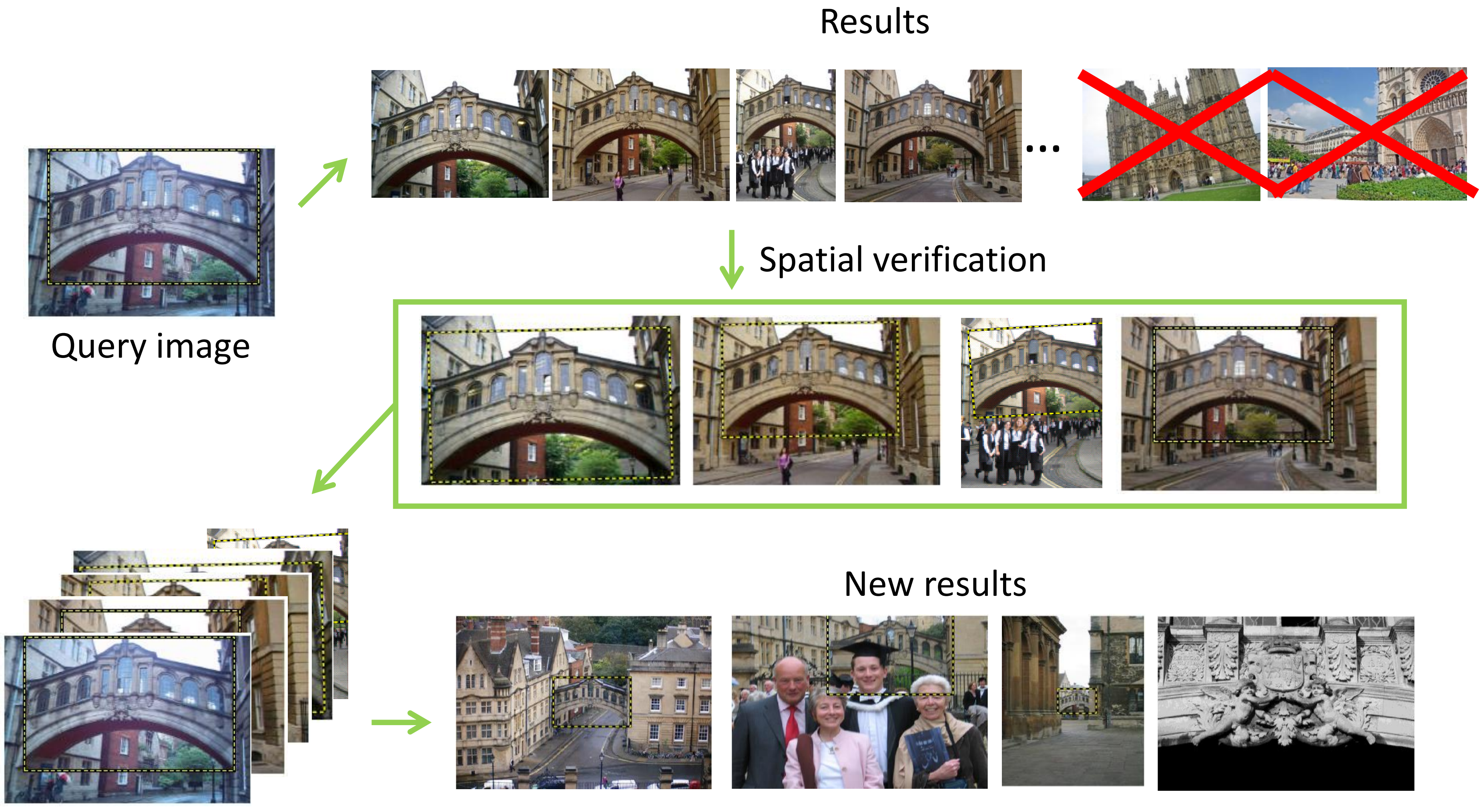
$$t_i = \frac{n_{id}}{n_d} \log \frac{N}{n_i}$$

Total number of documents in database

Number of documents word i occurs in, in whole database

Kristen Grauman

Query Expansion



New query

Chum, Philbin, Sivic, Isard, Zisserman: Total Recall..., ICCV 2007

Slide credit: Ondrej Chum