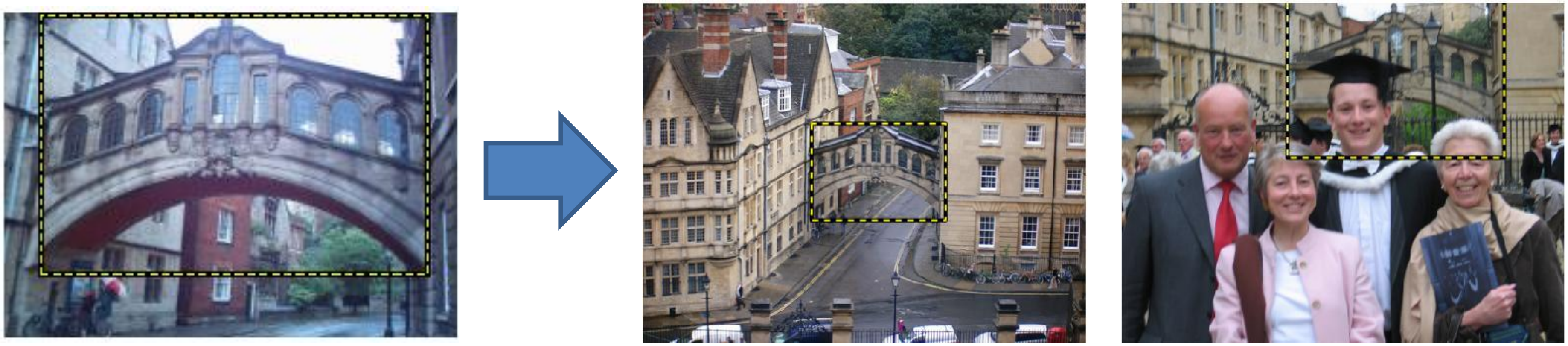
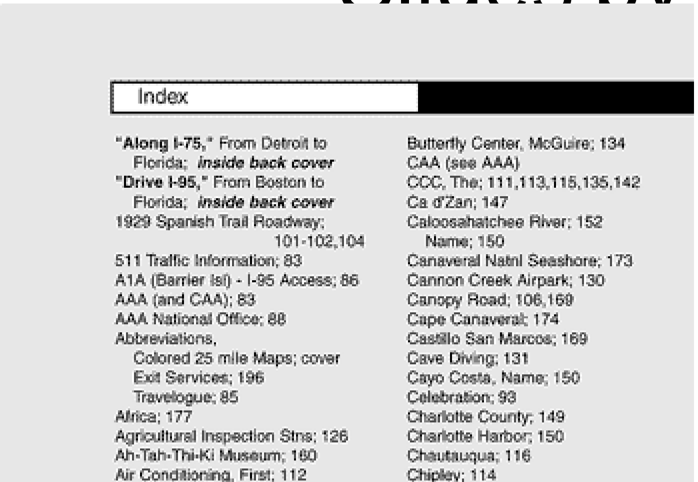


TP14 - Indexing local features

Computer Vision, FCUP, 2014

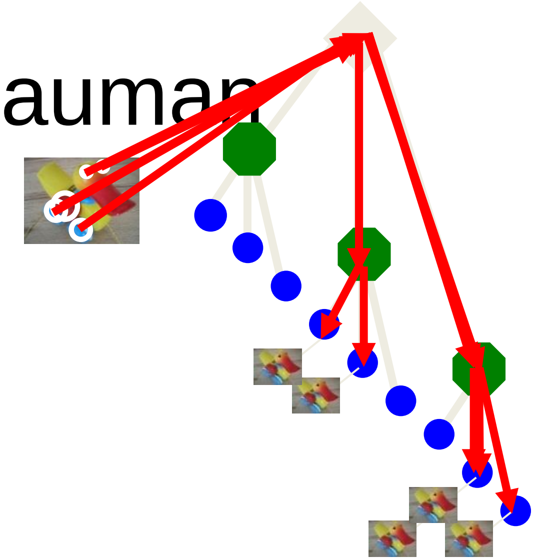
Miguel Coimbra

Slides by Prof. Kristen Grauman



Index

"**Along I-75,**" From Detroit to Florida; ***inside back cover***
"**Drive I-95,**" From Boston to Florida; ***inside back cover***
1929 Spanish Trail Roadway; 101-102,104
511 Traffic Information; 83
A1A (Barrier Isl) - I-95 Access; 86
AAA (and CAA); 83
AAA National Office; 88
Abbreviations,
Colored 25 mile Maps; cover
Exit Services; 196
Travelogue; 85
Africa; 177
Agricultural Inspection Stns; 126
Ah-Tah-Thi-Ki Museum; 160
Air Conditioning, First; 112
Butterfly Center, McGuire; 134
CAA (see AAA)
CCC, The; 111,113,115,135,142
Ca d'Zan; 147
Caloosahatchee River; 152
Name; 150
Canaveral Natnl Seashore; 173
Cannon Creek Airpark; 130
Canopy Road; 106,169
Cape Canaveral; 174
Castillo San Marcos; 169
Cave Diving; 131
Cayo Costa, Name; 150
Celebration; 93
Charlotte County; 149
Charlotte Harbor; 150
Chautauqua; 116
Chipley; 114

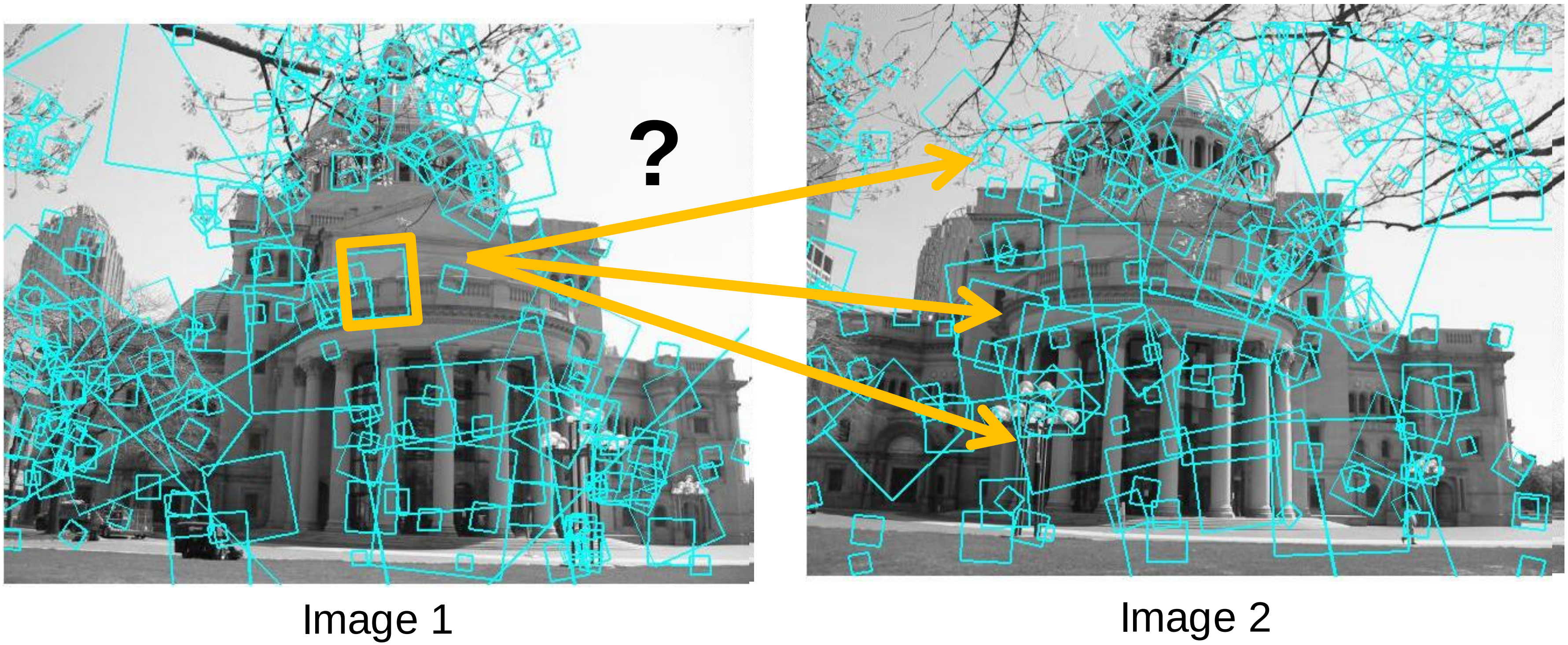


Matching local features

Kristen Grauman

Matching local features



To generate **candidate matches**, find patches that have the most similar appearance (e.g., lowest SSD)

Simplest approach: compare them all, take the closest (or closest k, or within a thresholded distance)

Kristen Grauman

Matching local features

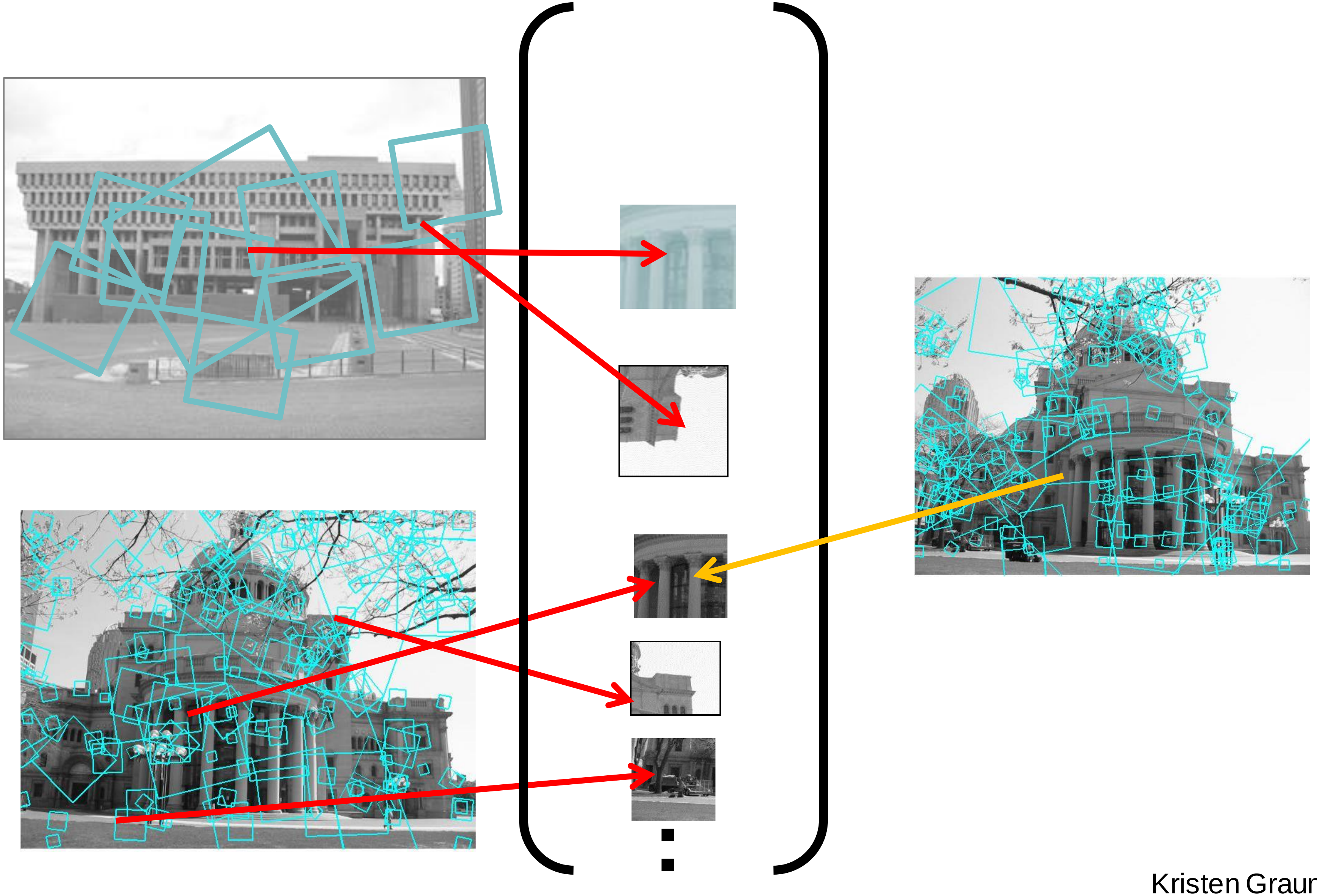
Image 1

Image 2

In stereo case, may constrain by proximity if we make assumptions on max disparities.

Kristen Grauman

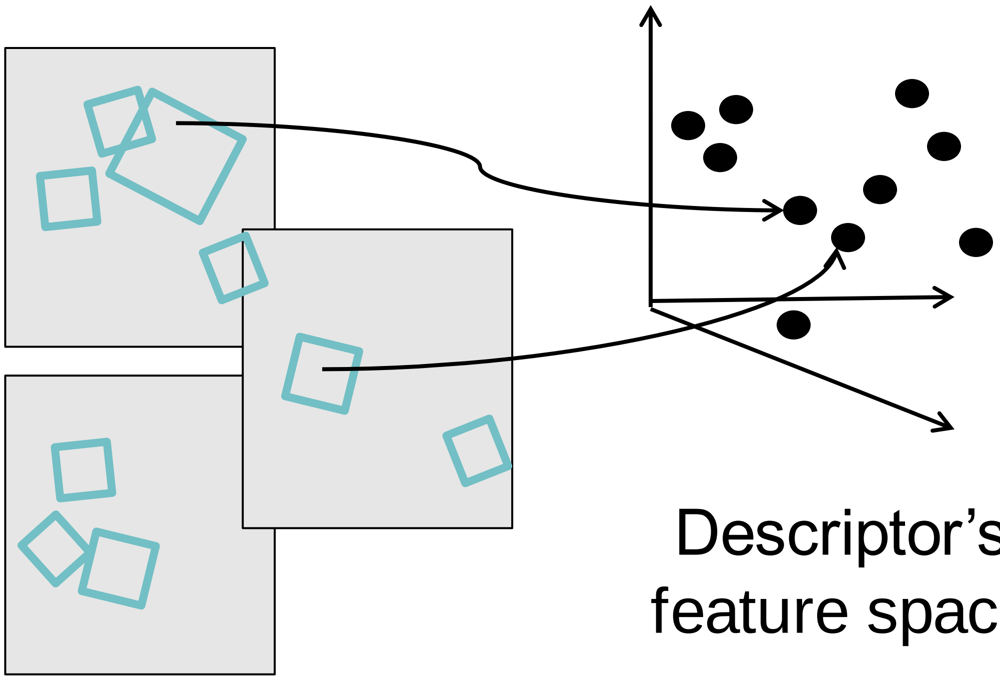
Indexing local features



Kristen Grauman

Indexing local features

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

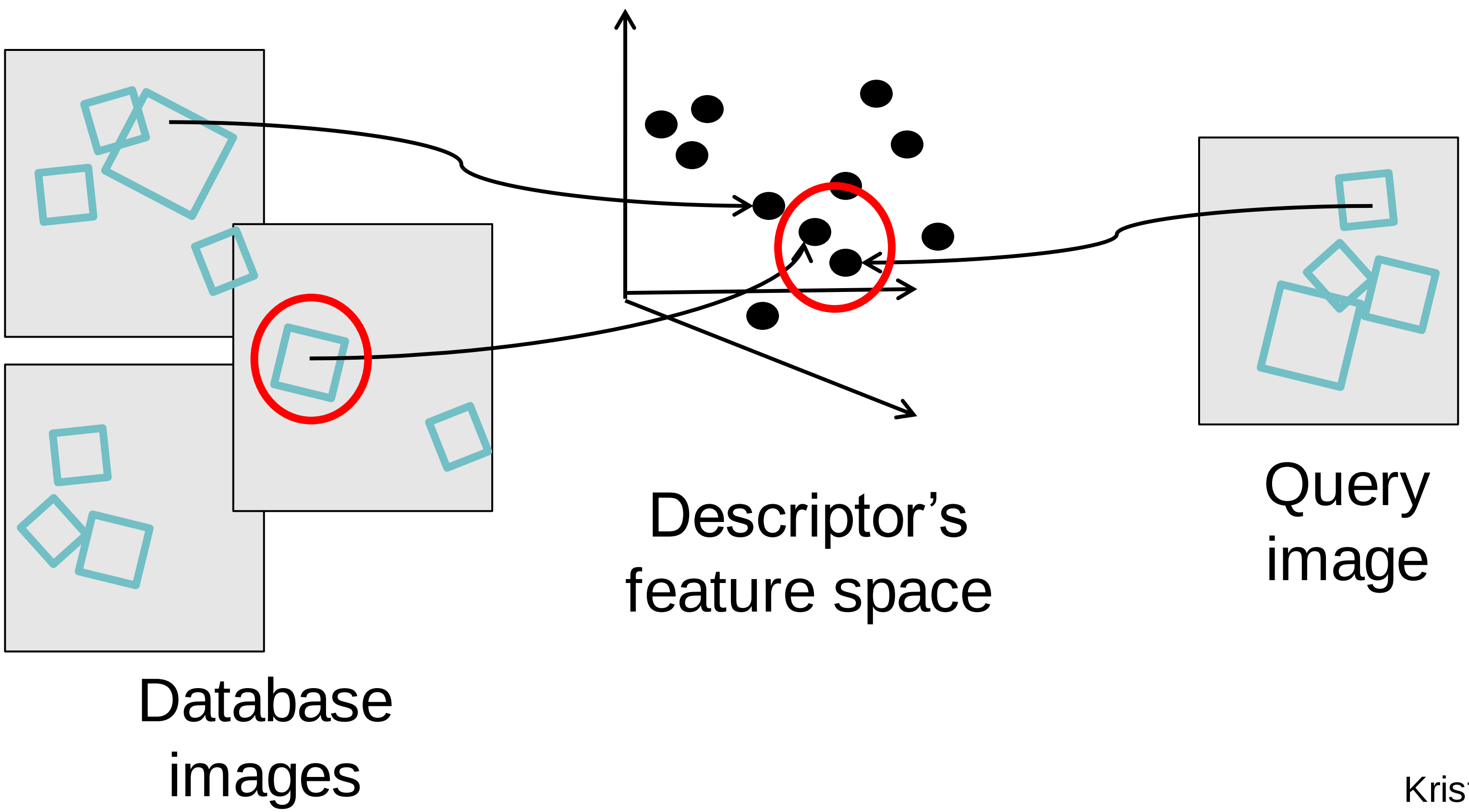


Descriptor's feature space

Kristen Grauman

Indexing local features

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



Kristen Grauman

Indexing local features

- With potentially thousands of features per image, and hundreds to millions of images to search, how to efficiently find those that are relevant to a new image?

Kristen Grauman

Indexing local features: inverted file index

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

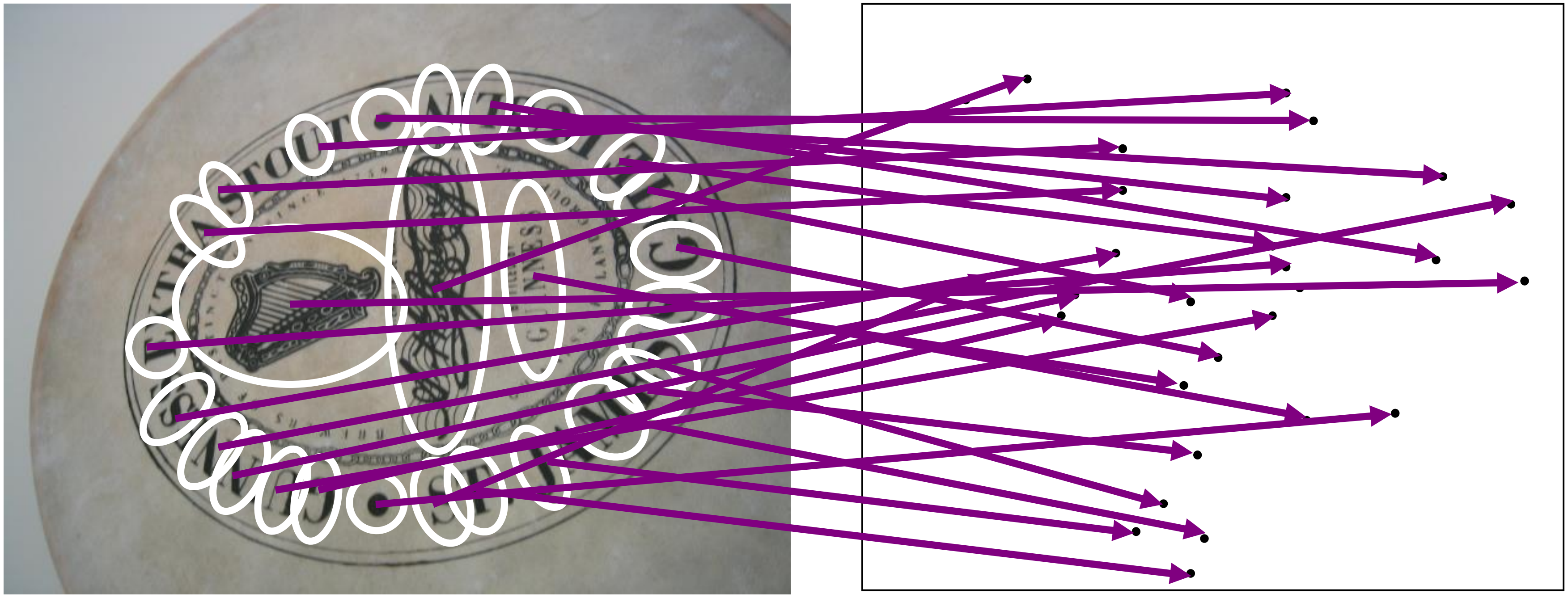
Text retrieval vs. image search

- What makes the problems similar, different?

Kristen Grauman

Visual words: main idea

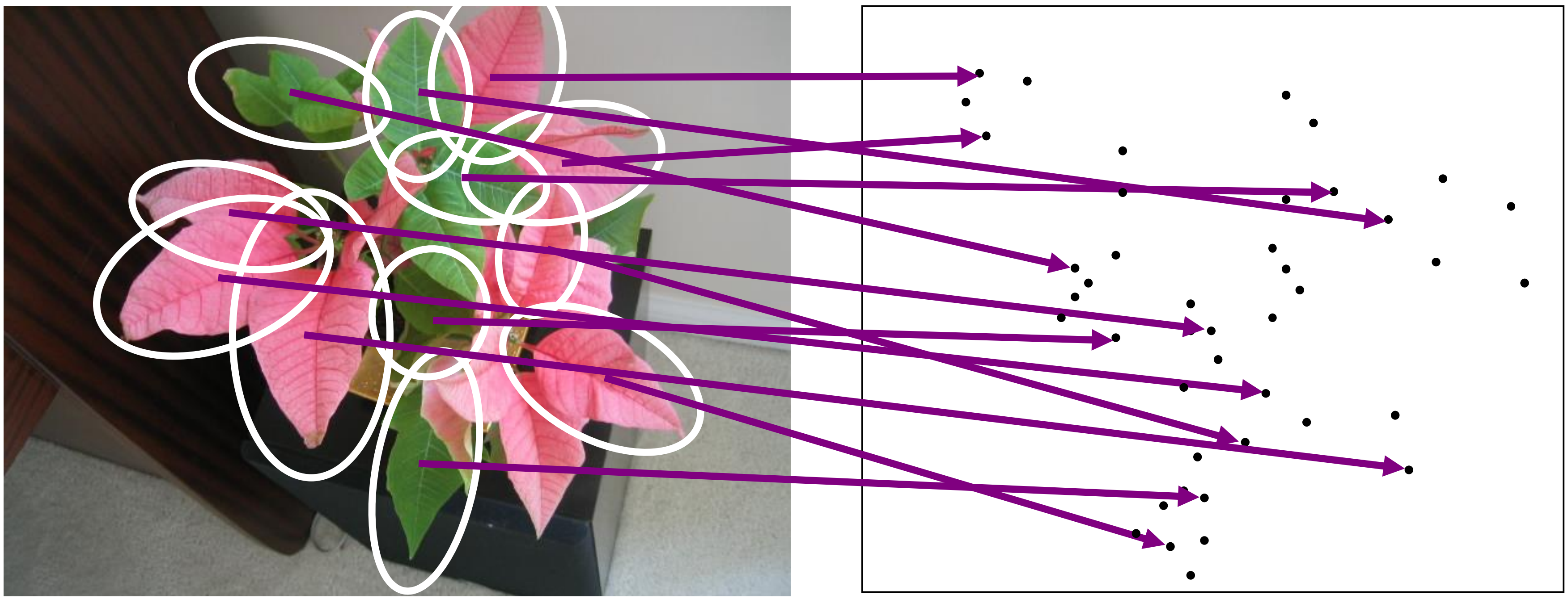
- Extract some local features from a number of images …



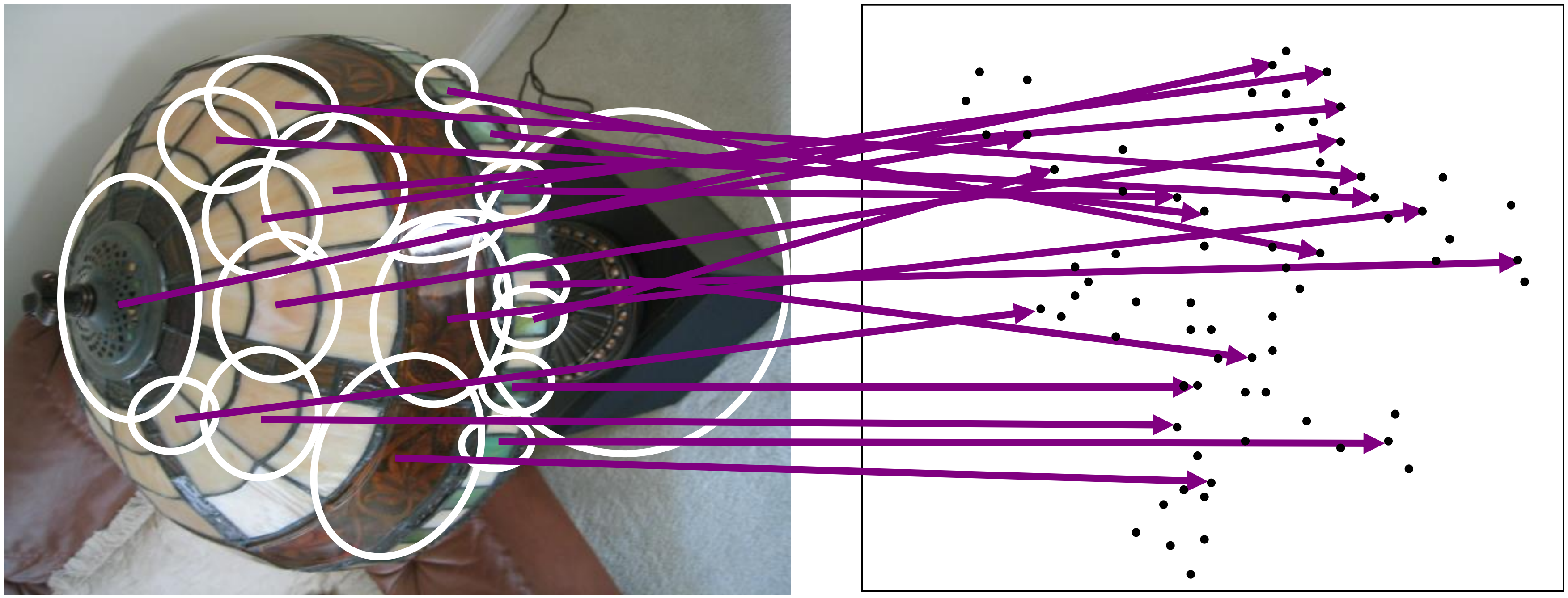
e.g., SIFT descriptor space: each point is 128-dimensional

Slide credit: D. Nister, CVPR 2006

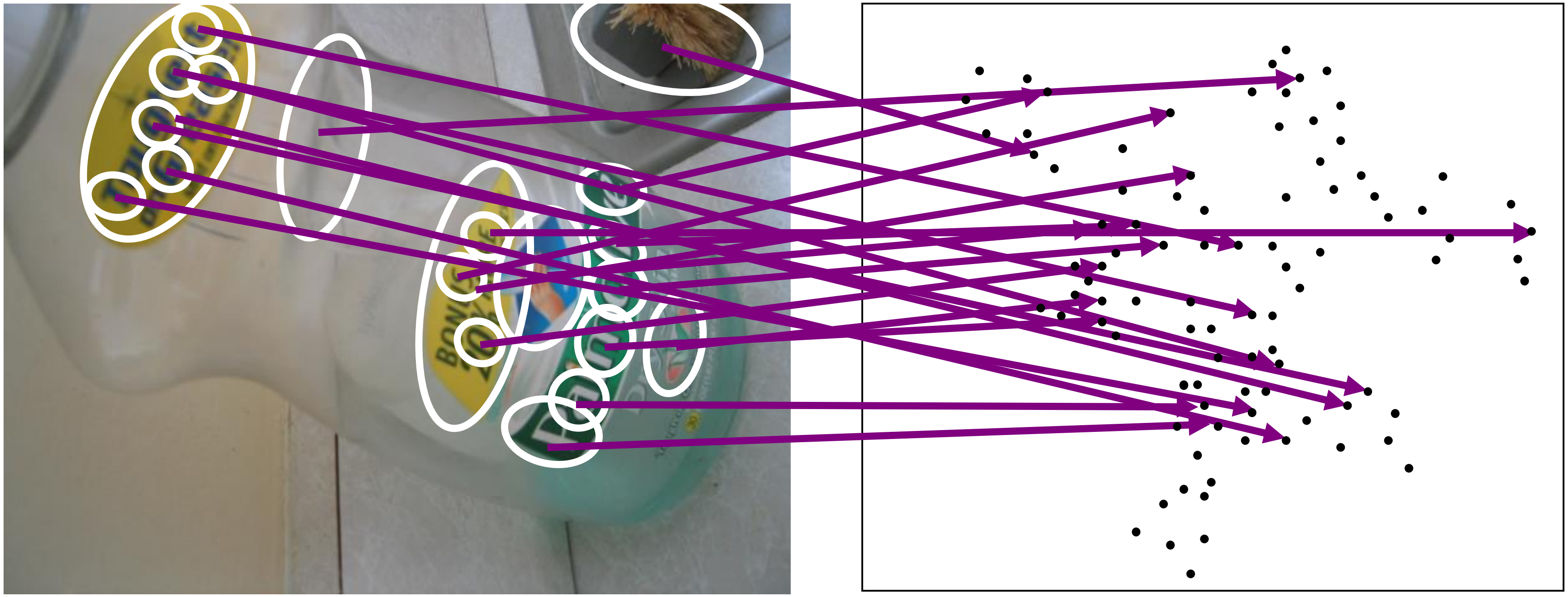
Visual words: main idea

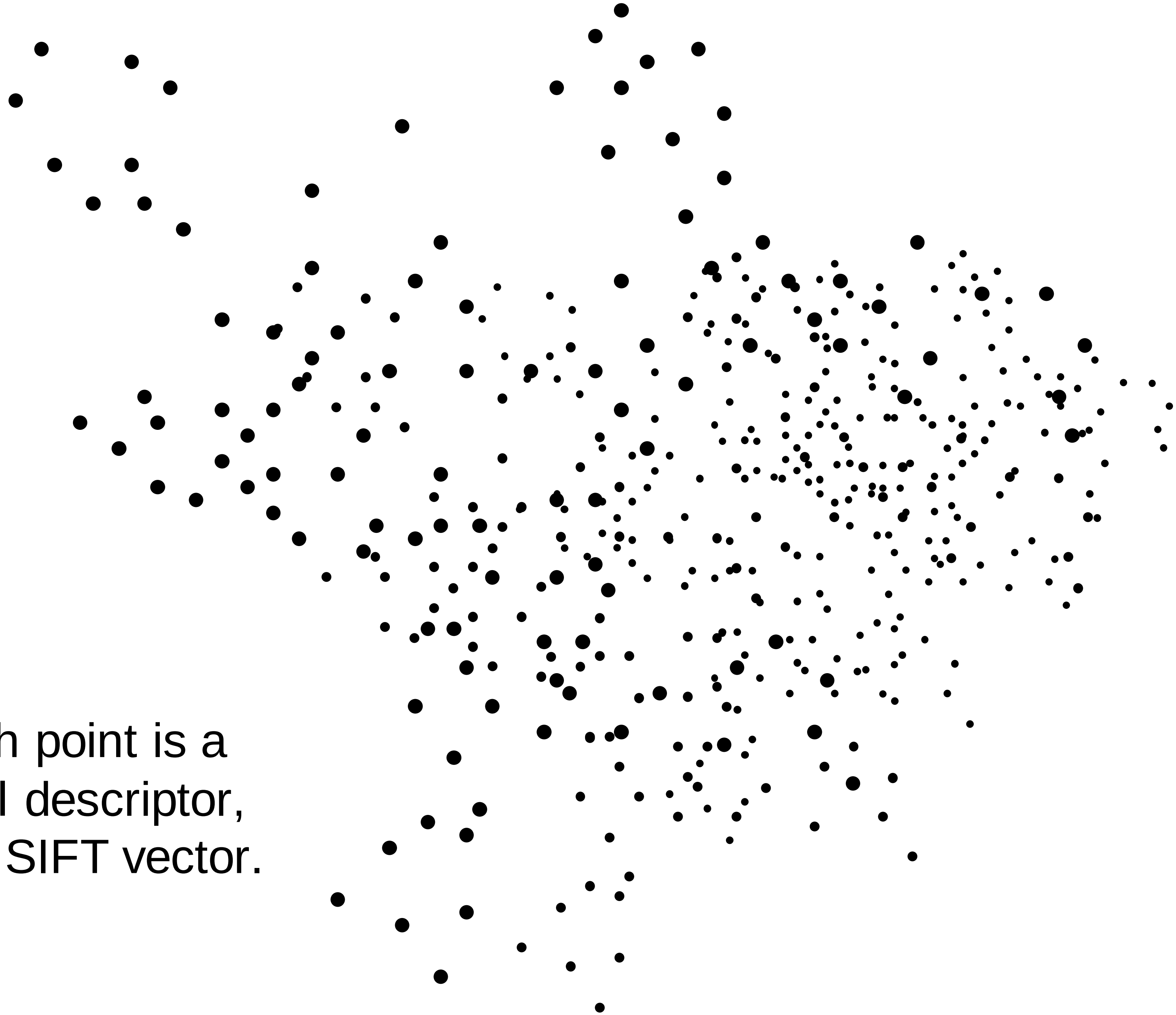


Visual words: main idea

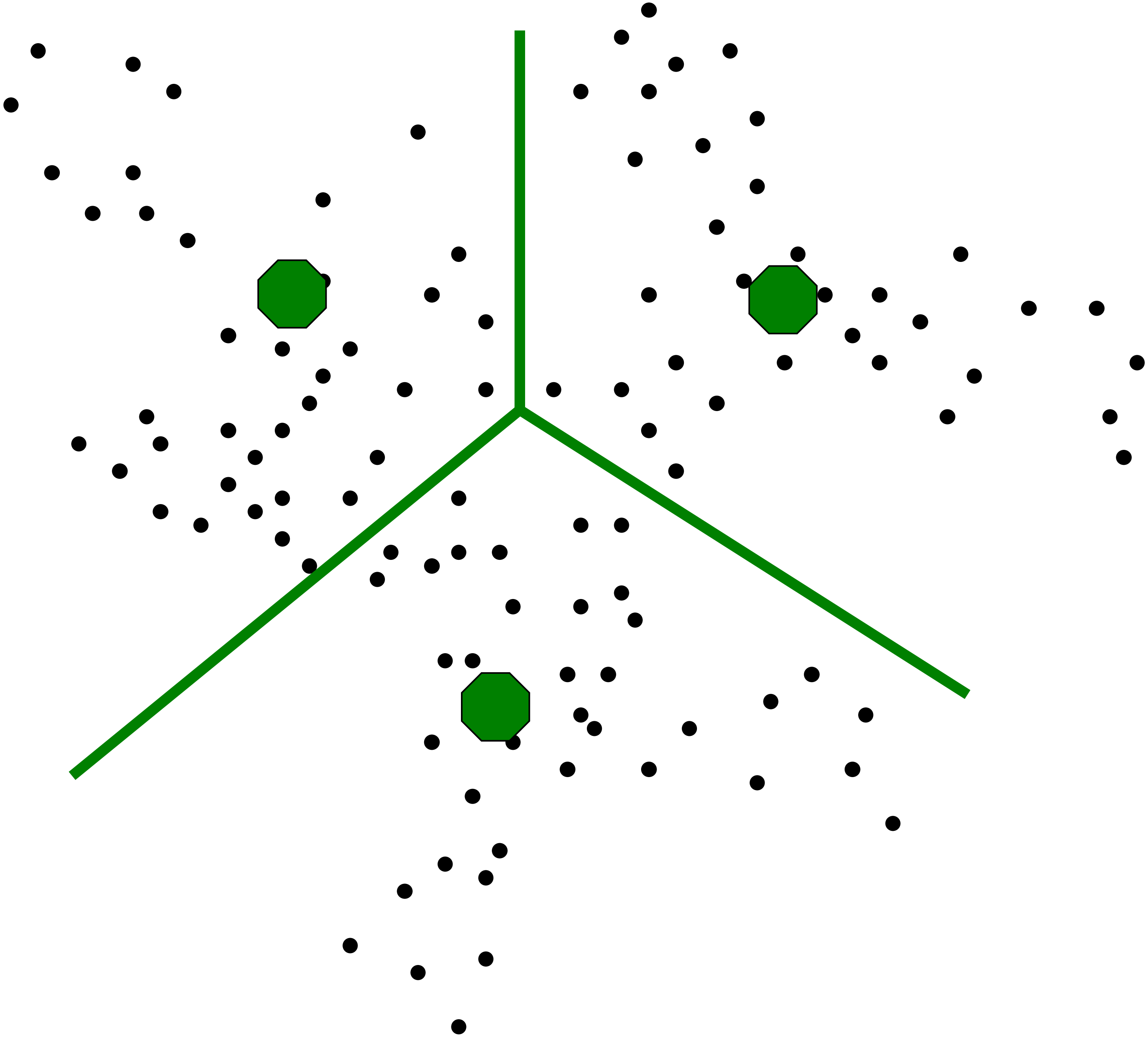


Visual words: main idea



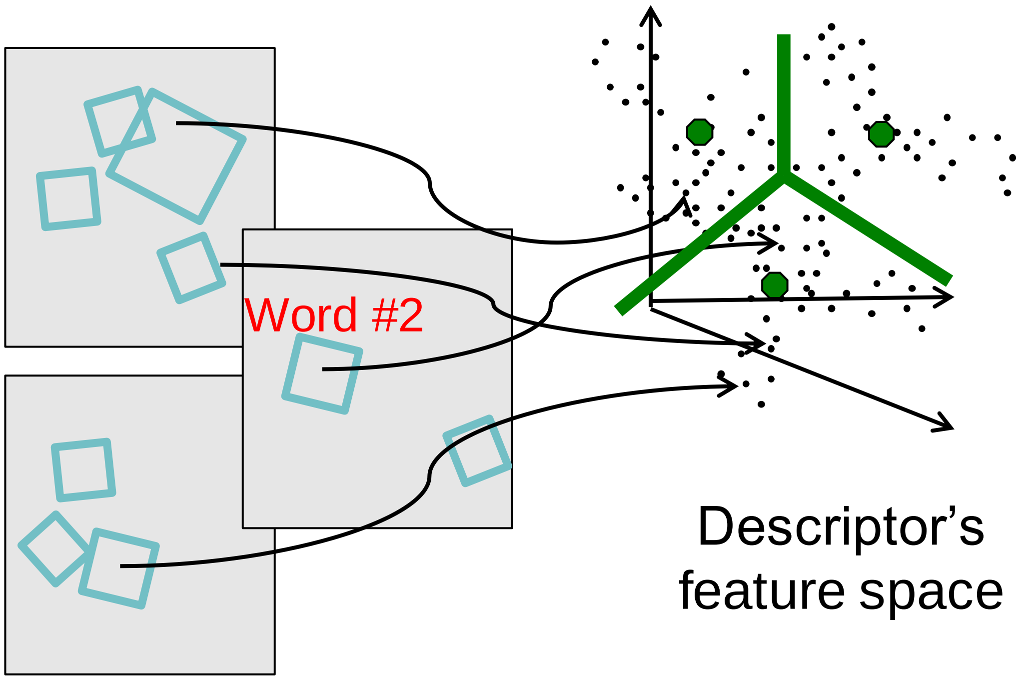


Each point is a local descriptor, e.g. SIFT vector.



Visual words

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

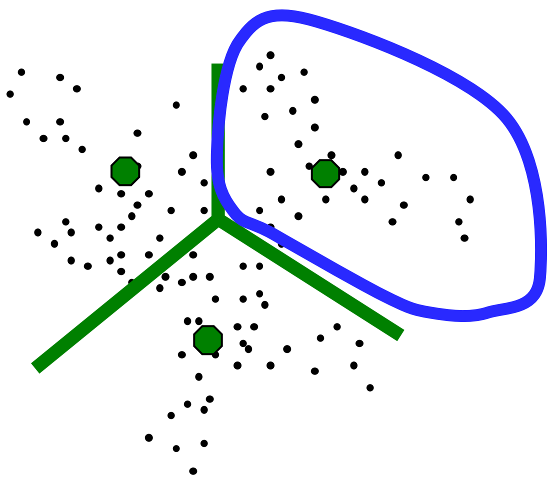


- Quantize via clustering, let cluster centers be the prototype "words"
- Determine which word to assign to each new image region by finding the closest cluster center.

Kristen Grauman

Visual words

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



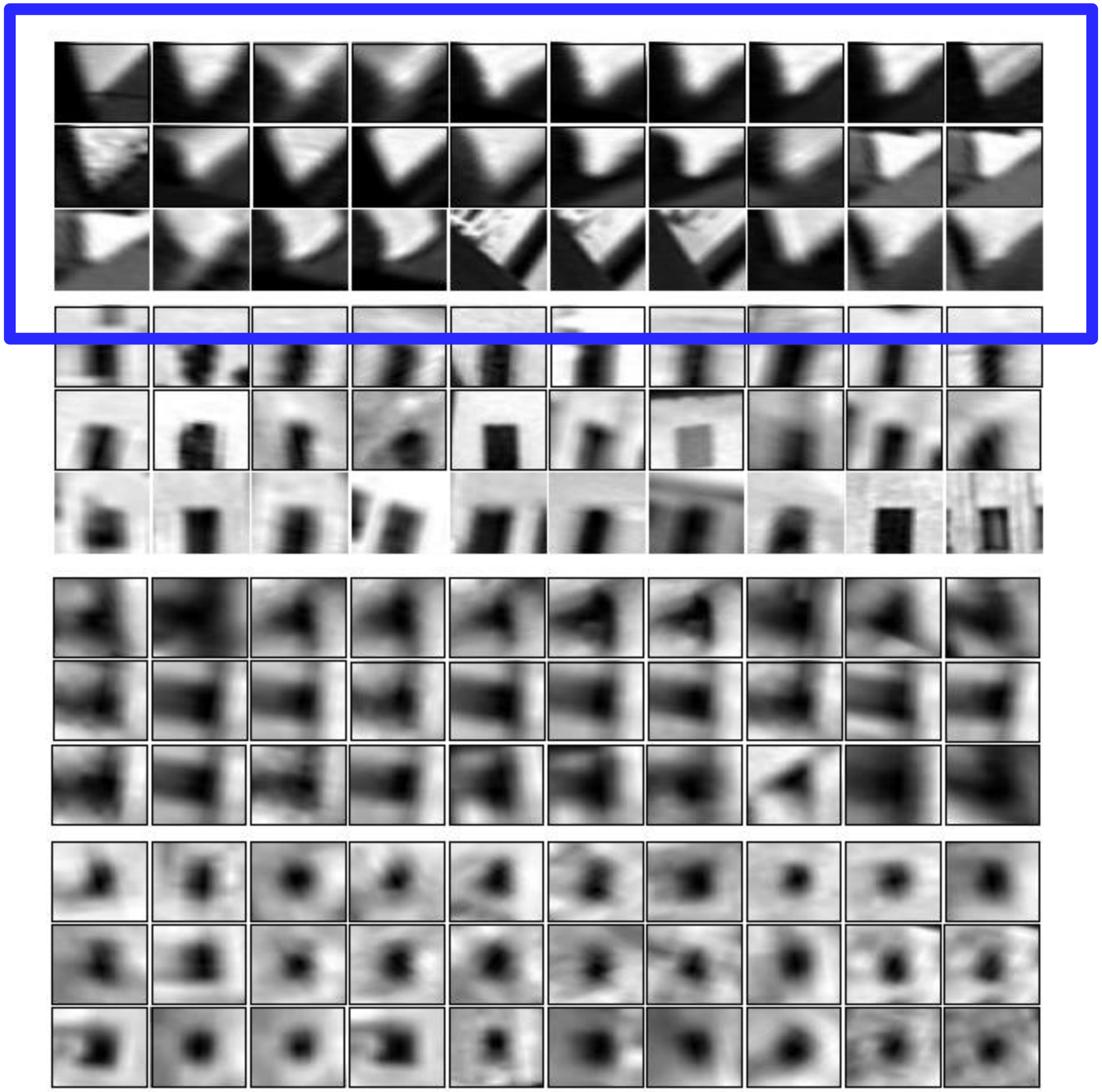
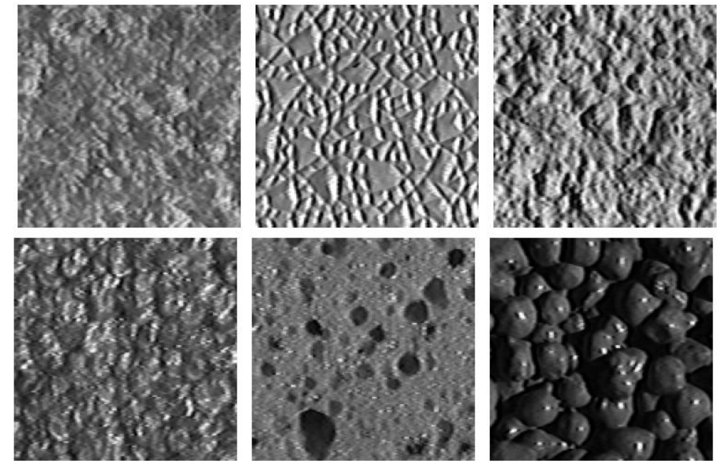


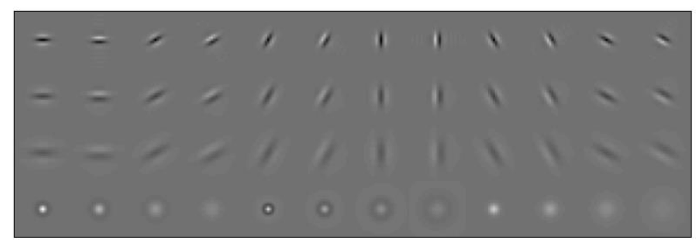
Figure from Sivic & Zisserman, ICCV 2003

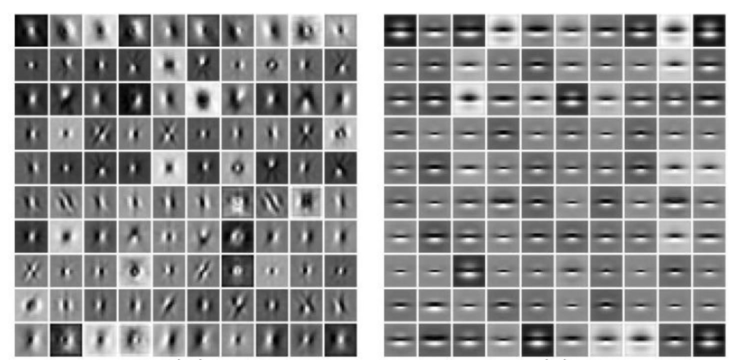
Kristen Grauman

Visual words and textons

- First explored for texture and material representations
- *Texton* = cluster center of filter responses over collection of images
- Describe textures and materials based on distribution of prototypical texture elements.



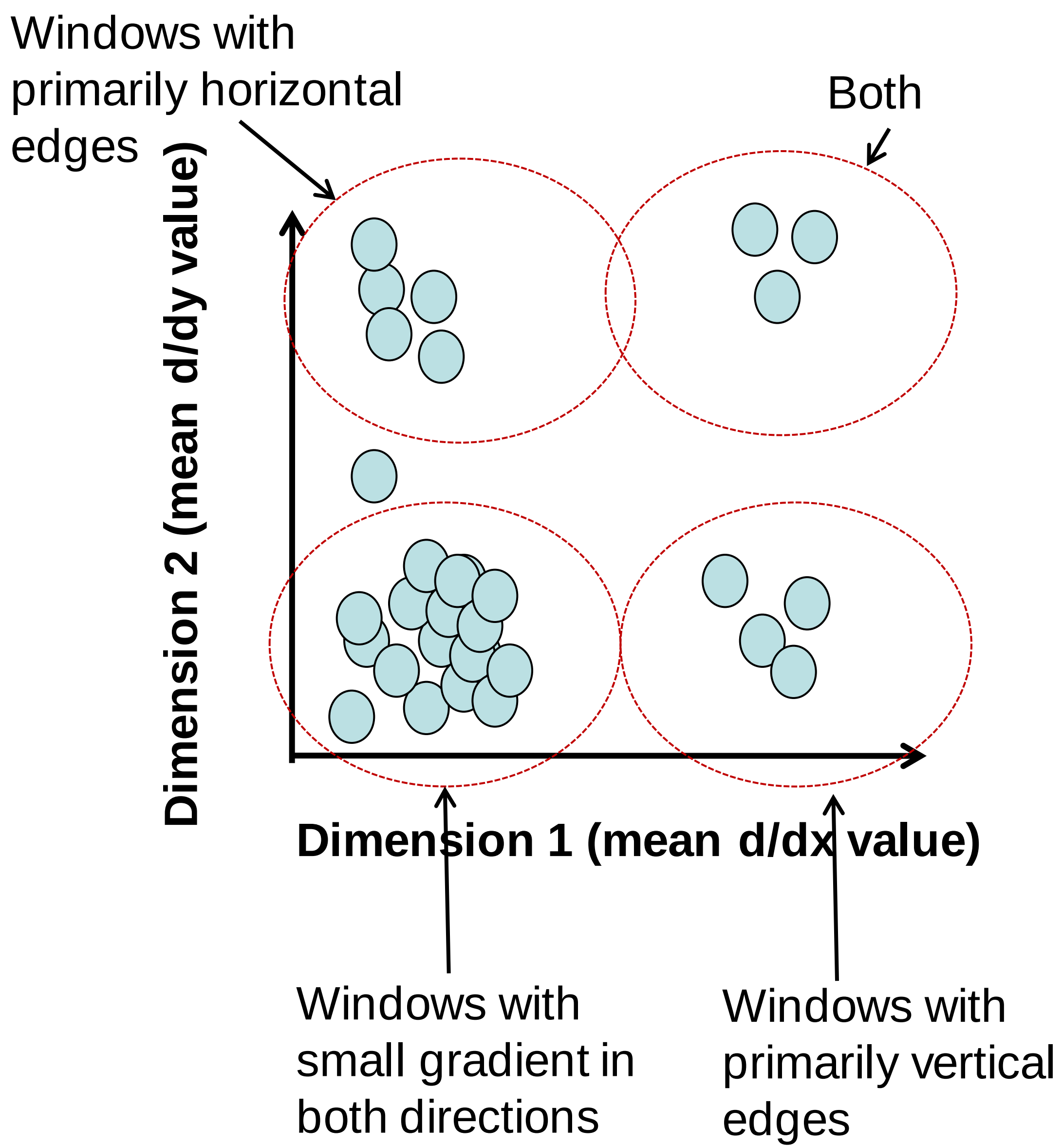




Leung & Malik 1999; Varma & Zisserman, 2002

Kristen Grauman

Recall: Texture representation example



	mean d/dx value	**mean d/dy value**
Win. #1	4	10
Win.#2	18	7
⋮		
Win.#9	20	20

⋮

statistics to summarize patterns in small windows

Kristen Grauman

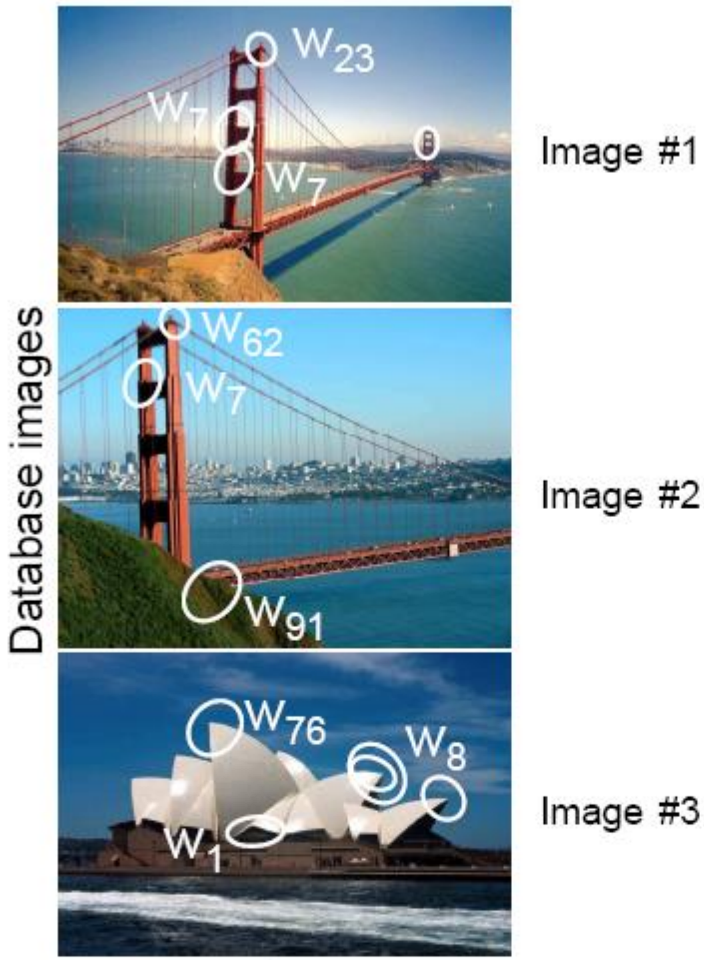
Visual vocabulary formation

Issues:

- Sampling strategy: where to extract features?
- Clustering / quantization algorithm
- Unsupervised vs. supervised
- What corpus provides features (universal vocabulary?)
- Vocabulary size, number of words

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

When will this give us a significant gain in efficiency?

New query image

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

…

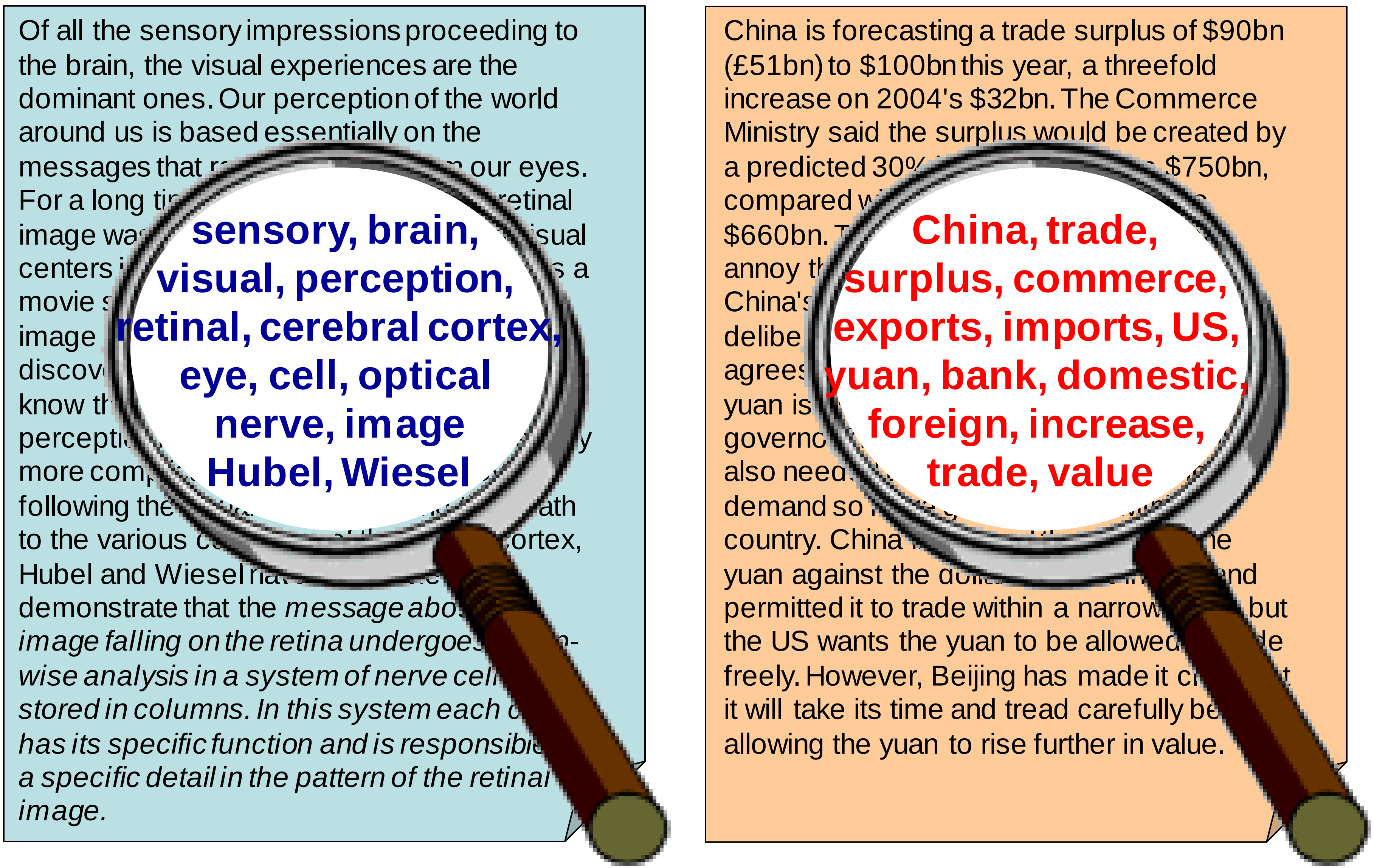
⋮ ⋮

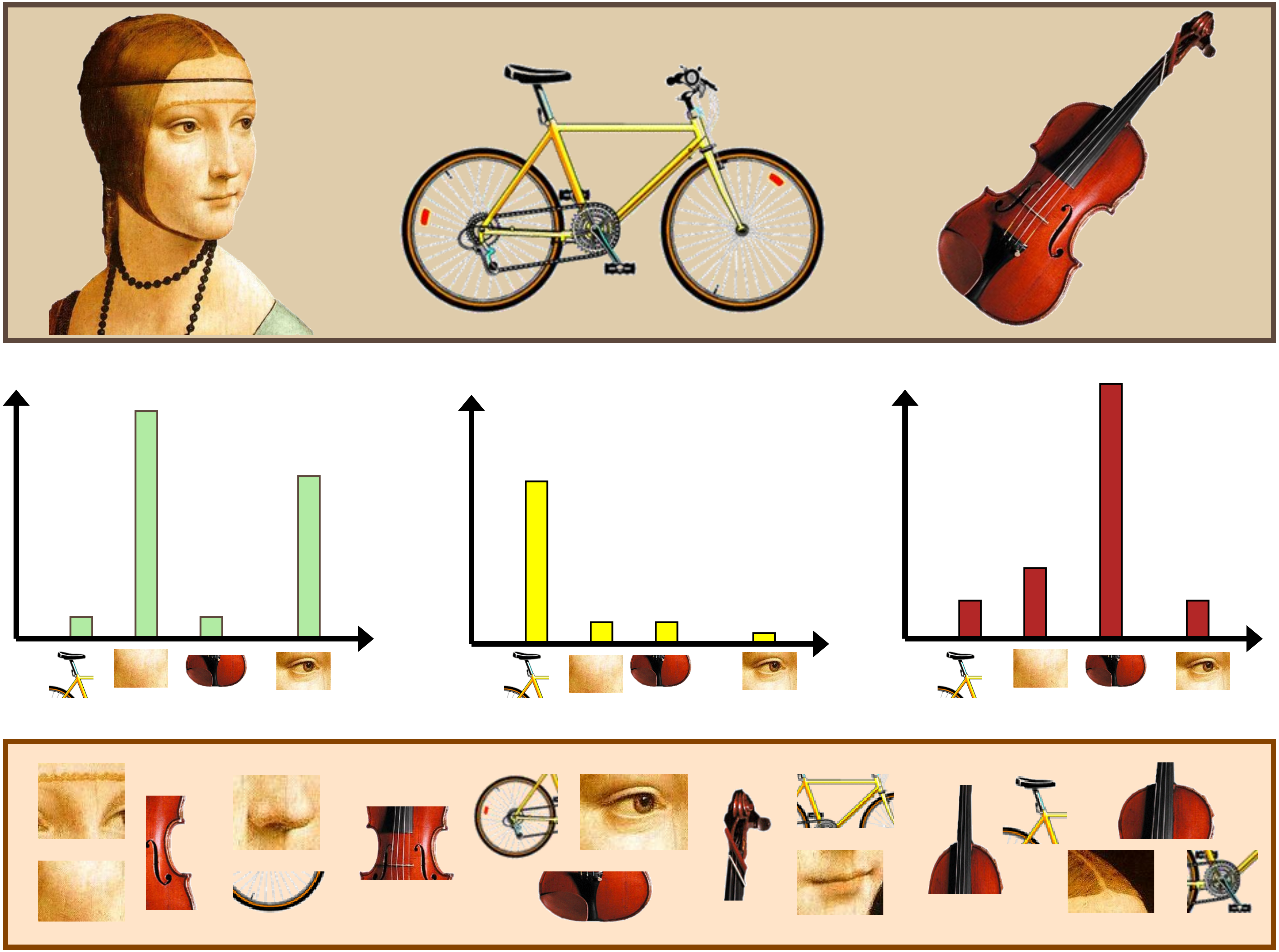
- New query image is mapped to indices of database images that share a word.

Kristen Grauman

- If a local image region is a visual word, how can we summarize an image (the document)?

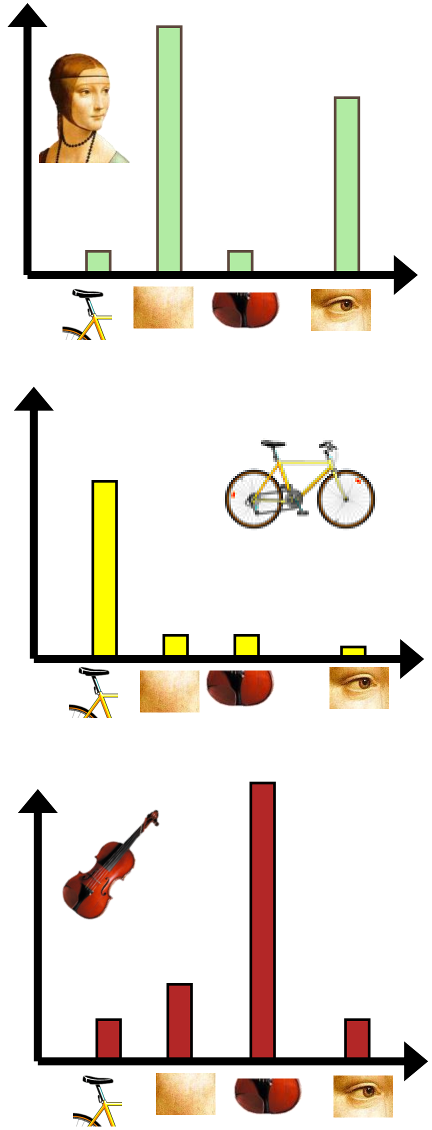
Analogy to documents





Bags of visual words

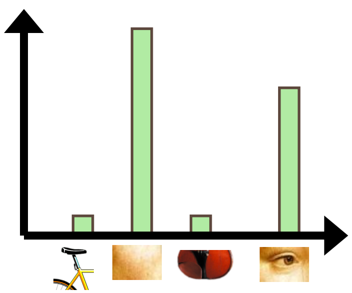
- Summarize entire image based on its distribution (histogram) of word occurrences.
- Analogous to bag of words representation commonly used for documents.



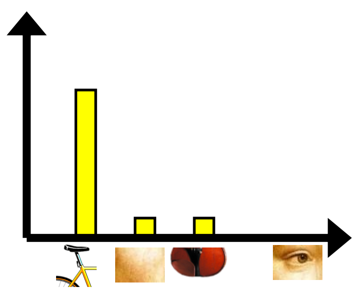
Comparing bags of words

- 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]



$\vec{d}_j$

$\vec{q}$

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

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

for vocabulary of *V* words

Kristen Grauman

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

Bags of words for content-based image retrieval

Visually defined query

"Groundhog Day" [Rammis, 1993]

"Find this clock"

"Find this place"

Slide from Andrew Zisserman
Sivic & Zisserman, ICCV 2003

Example

retrieved shots

Slide from Andrew Zisserman
Sivic & Zisserman, ICCV 2003

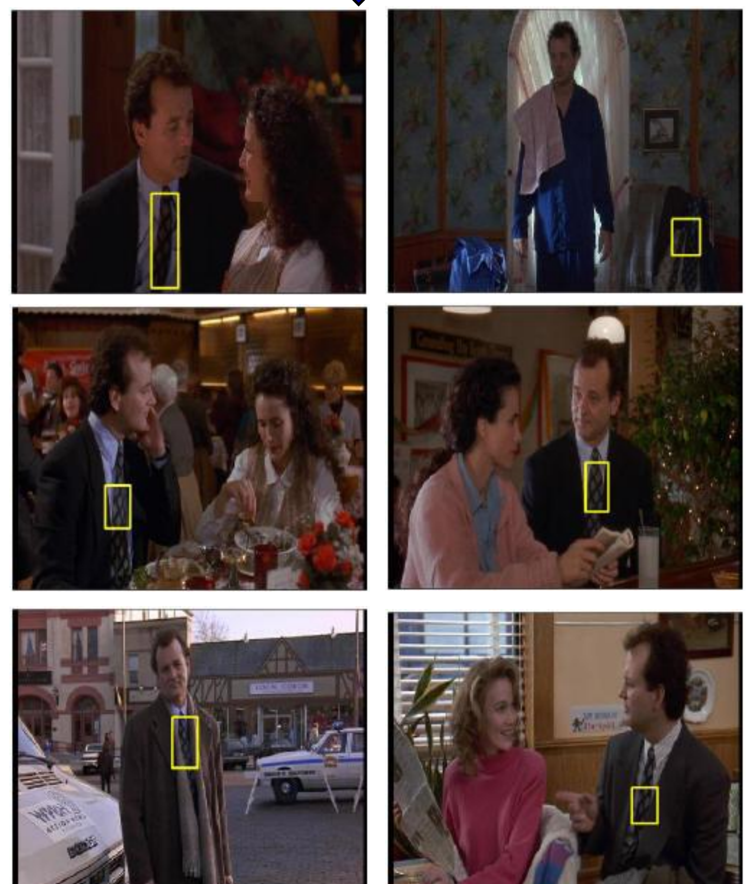
Video Google System

1. Collect all words within query region
2. Inverted file index to find relevant frames
3. Compare word counts
4. Spatial verification

Sivic & Zisserman, ICCV 2003

- **Demo online at :**
  http://www.robots.ox.ac.uk/~vgg/research/vgoogle/index.html

Query region



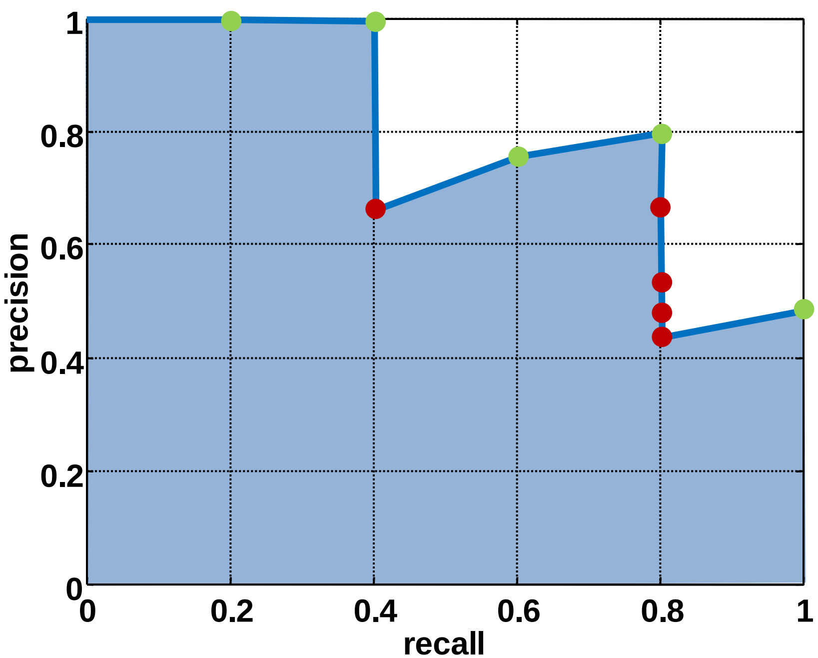
Retrieved frames

Scoring retrieval quality

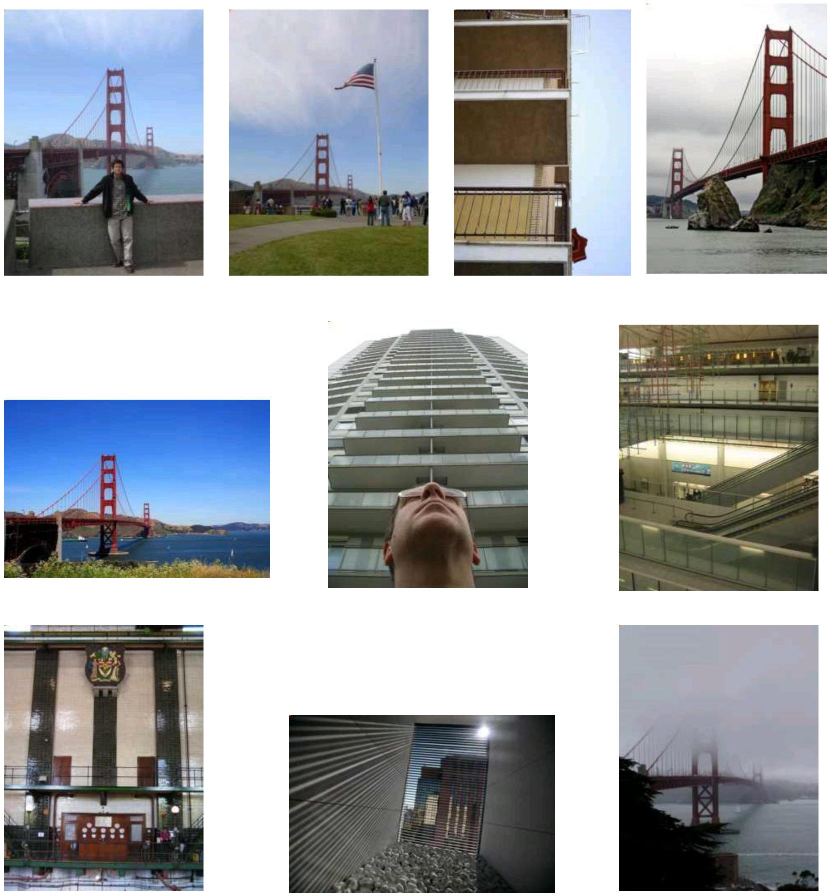
Query

Database size: 10 images
Relevant (total): 5 images

precision = #relevant / #returned
recall = #relevant / #total relevant



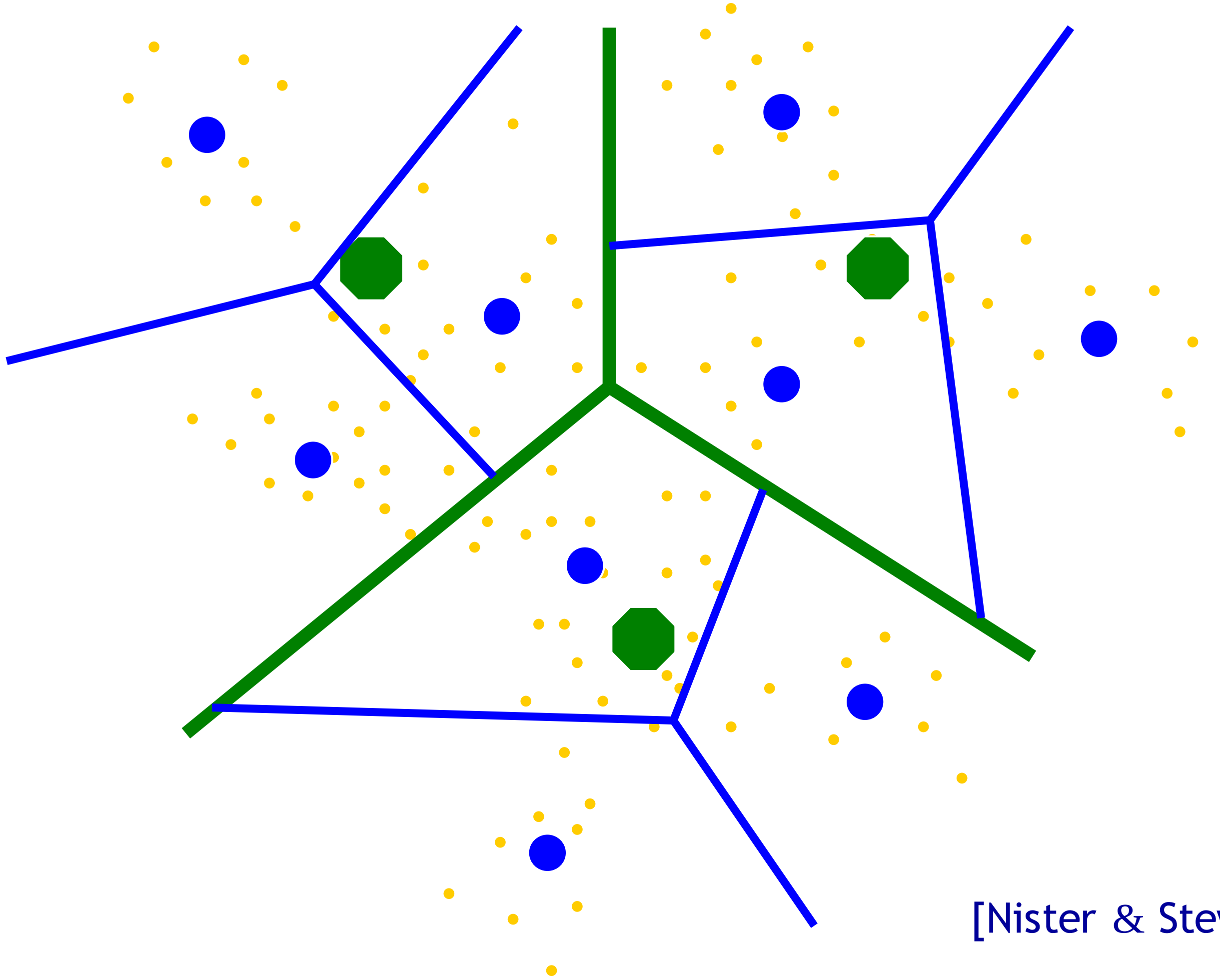
Results (ordered):



Slide credit: Ondrej Chum

Vocabulary Trees: hierarchical clustering for large vocabularies

- Tree construction:

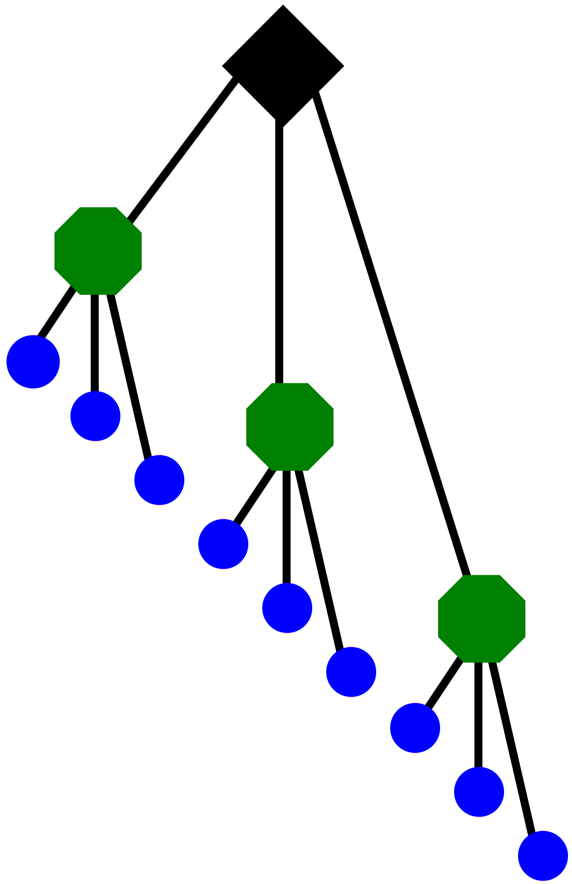


[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

Vocabulary Tree

- **Training: Filling the tree**

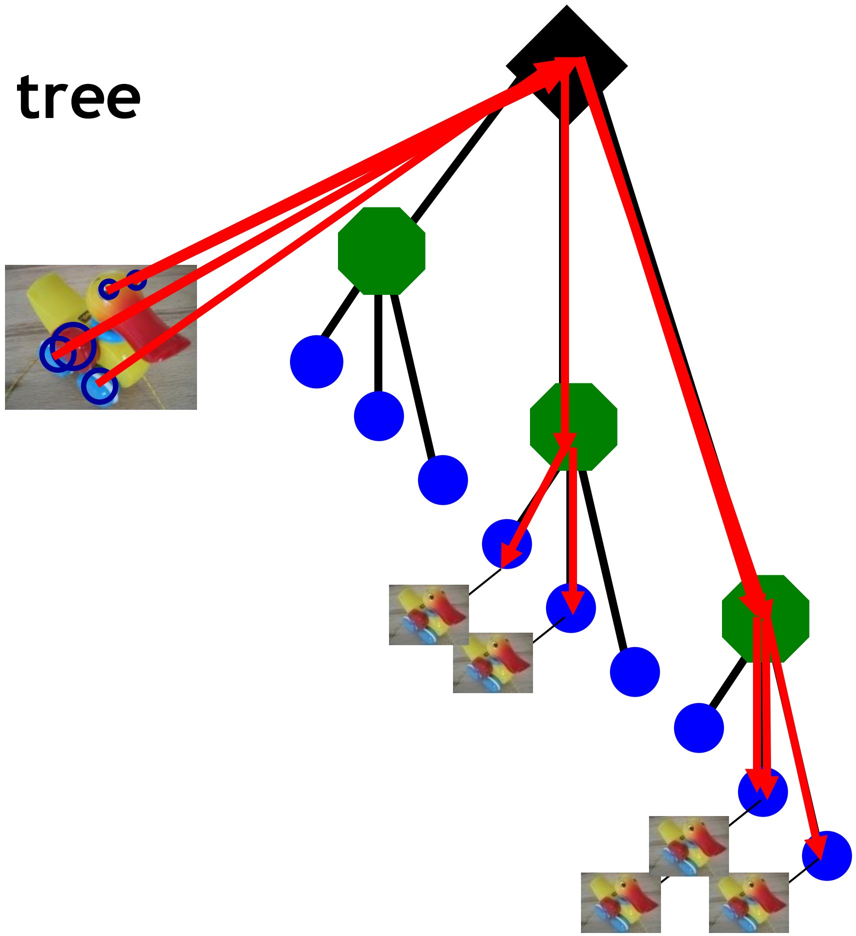


[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

Vocabulary Tree

- **Training: Filling the tree**

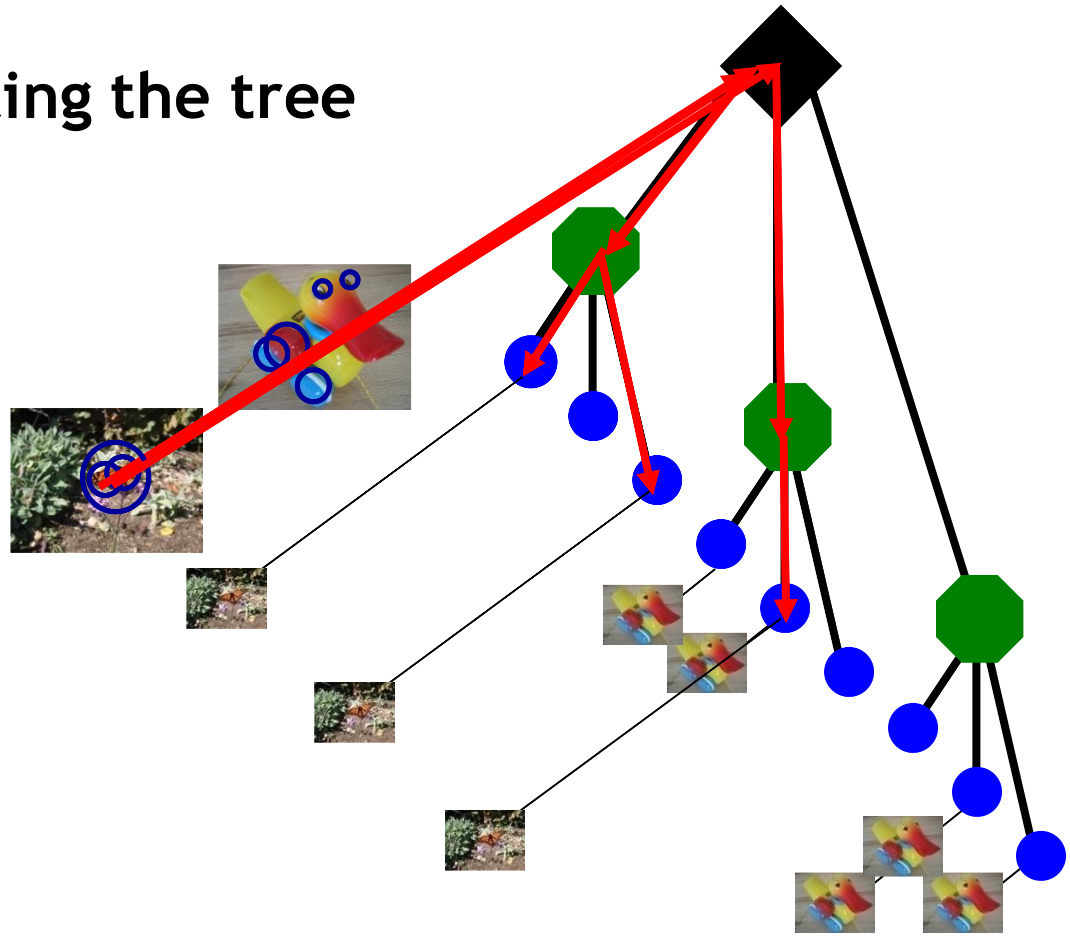


[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

Vocabulary Tree

- **Training: Filling the tree**

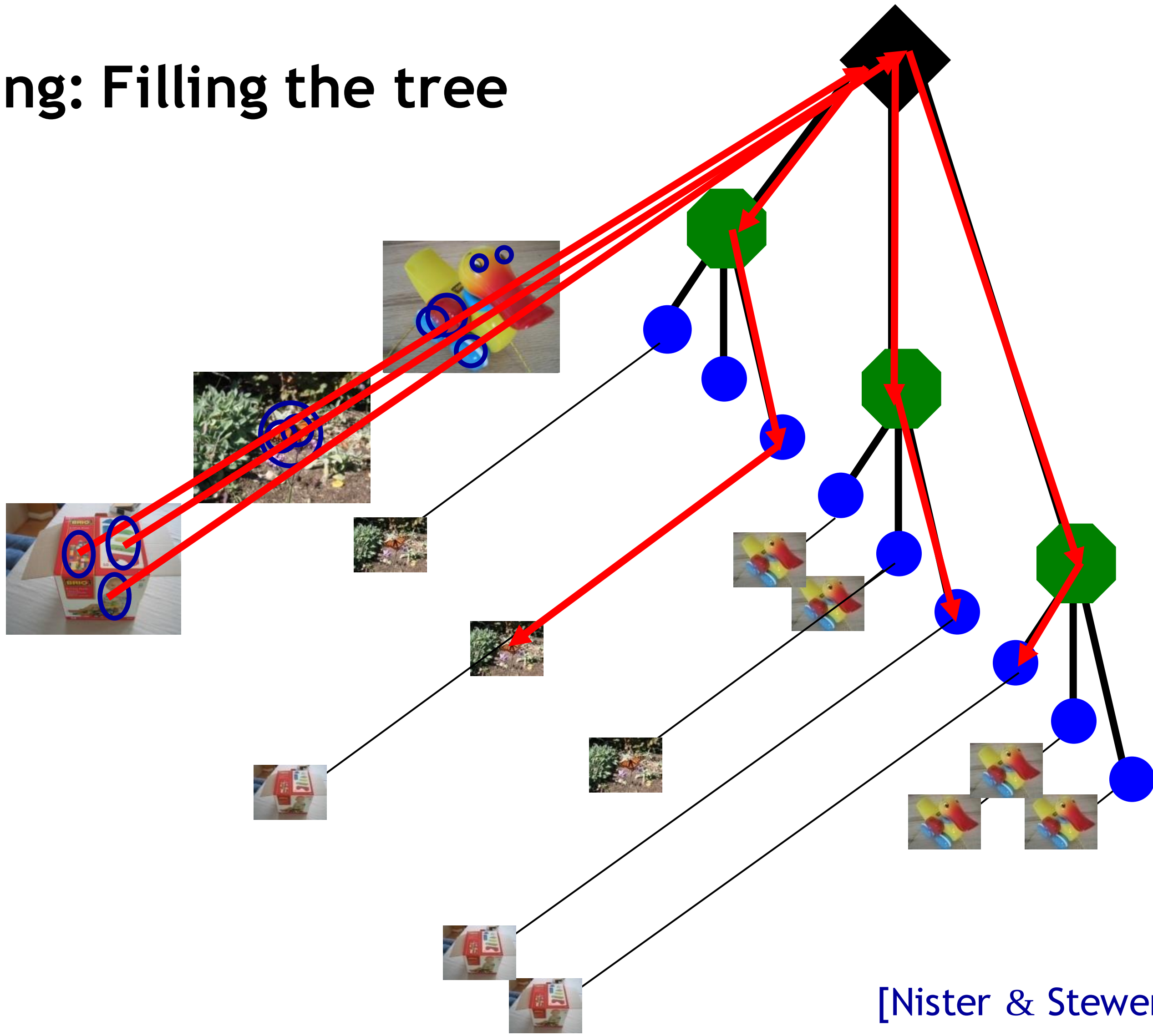


[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

Vocabulary Tree

- **Training: Filling the tree**

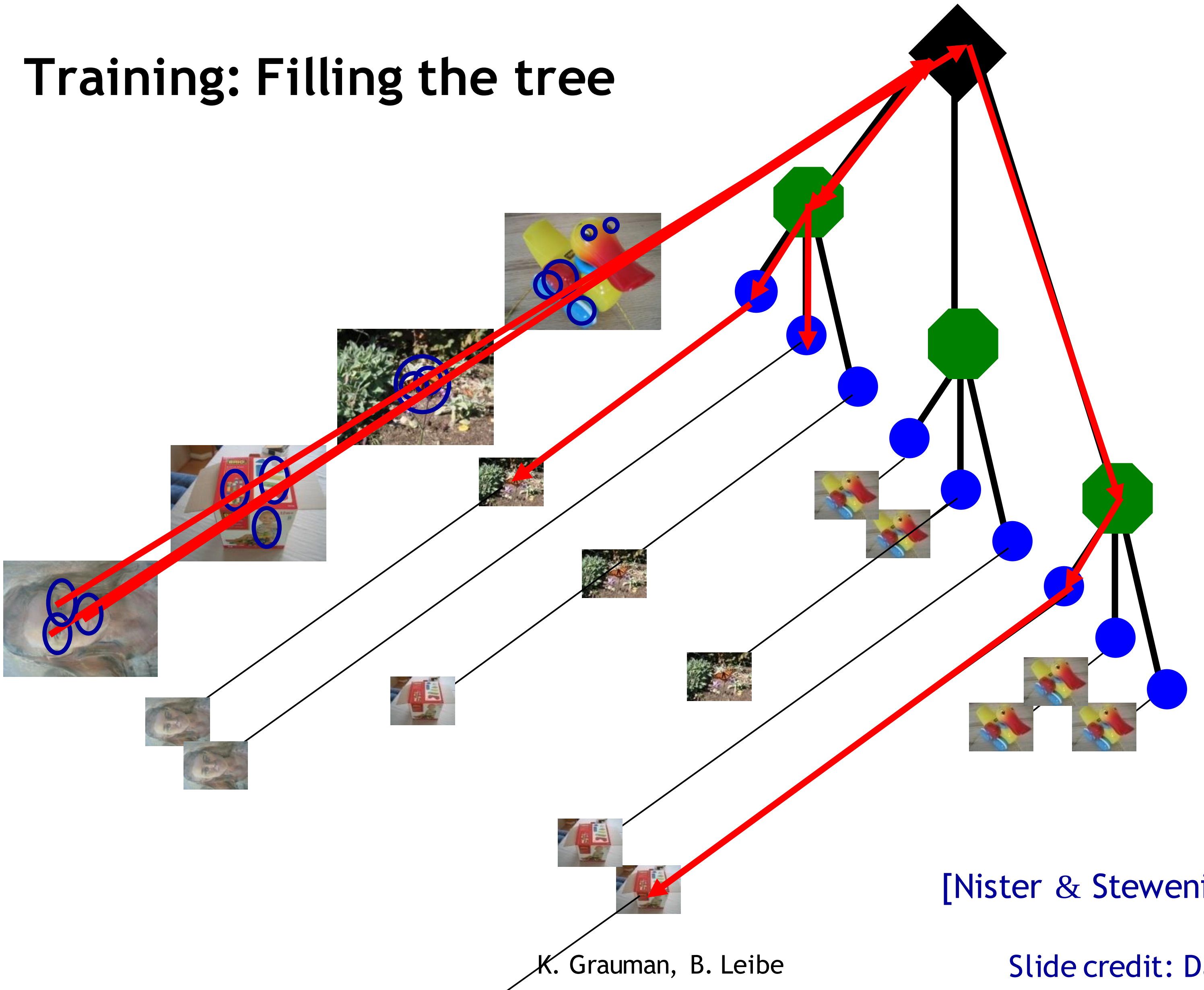


[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

Vocabulary Tree

- **Training: Filling the tree**



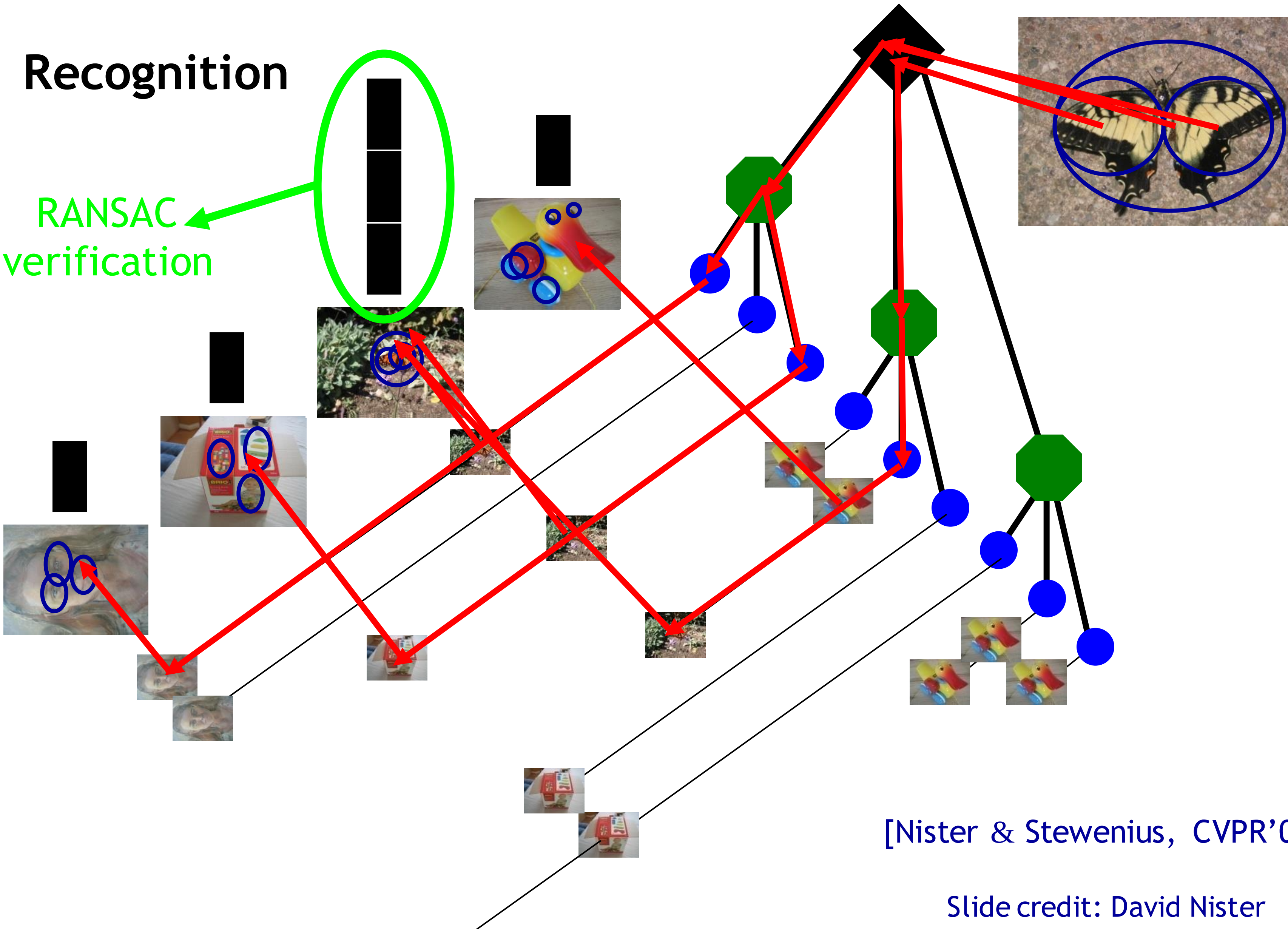
[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

What is the computational advantage of the hierarchical representation bag of words, vs. a flat vocabulary?

Vocabulary Tree

- **Recognition**



[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

Bags of words: pros and cons

+ flexible to geometry / deformations / viewpoint
+ compact summary of image content
+ provides vector representation for sets
+ very good results in practice

- basic model ignores geometry – must verify afterwards, or encode via features
- background and foreground mixed when bag covers whole image
- optimal vocabulary formation remains unclear

Summary

- **Matching local invariant features**: useful not only to provide matches for multi-view geometry, but also to find objects and scenes.
- **Bag of words** representation: quantize feature space to make discrete set of visual words
  - Summarize image by distribution of words
  - Index individual words
- **Inverted index**: pre-compute index to enable faster search at query time