Large-scale Instance Retrieval

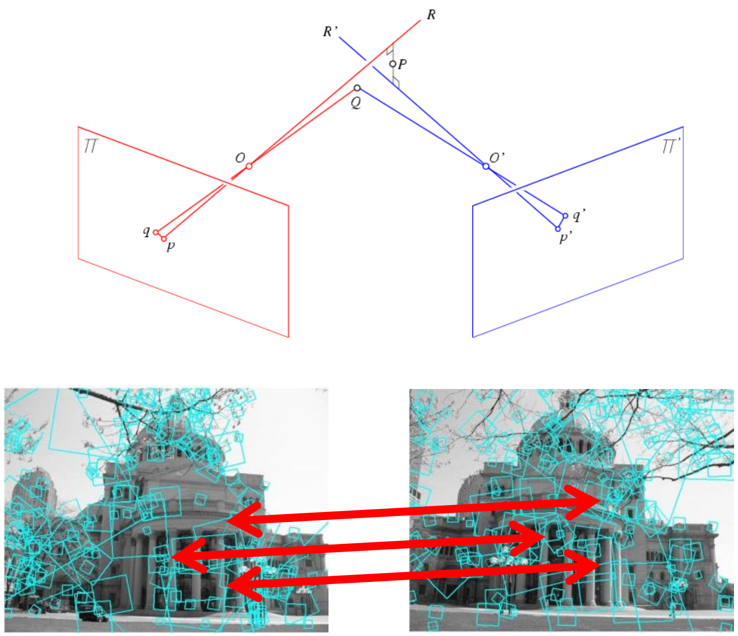
Computer Vision

CS 143, Brown

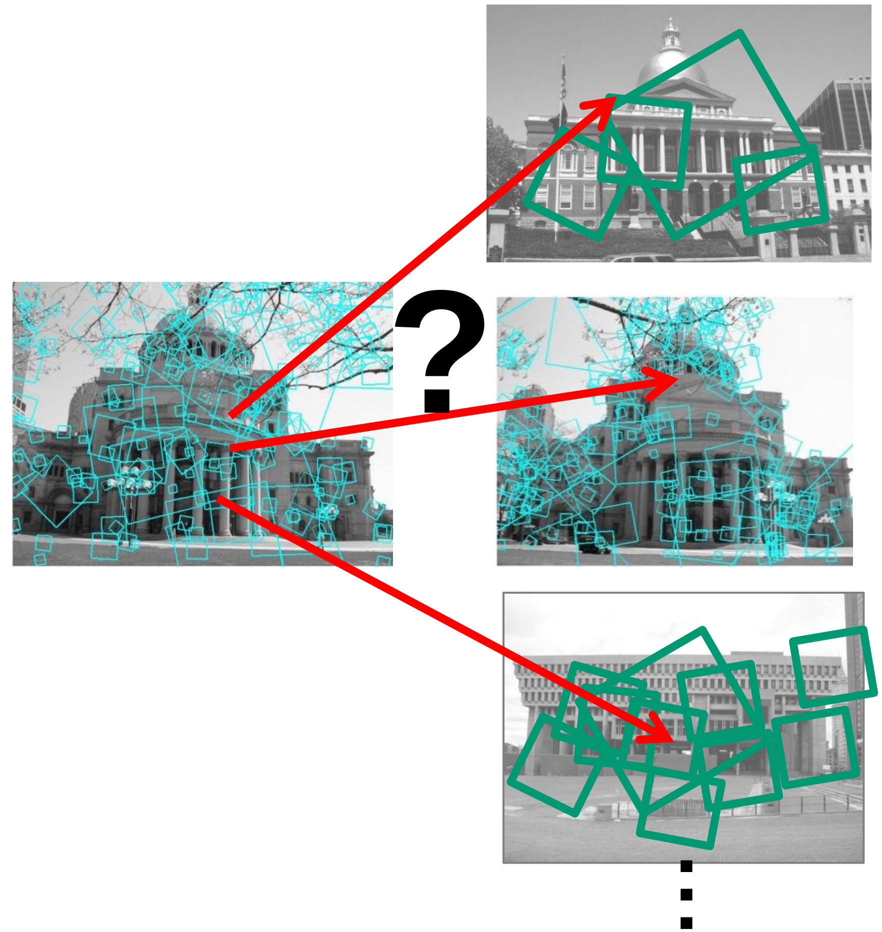
James Hays

Many slides from Derek Hoiem and Kristen Grauman

Multi-view matching



vs

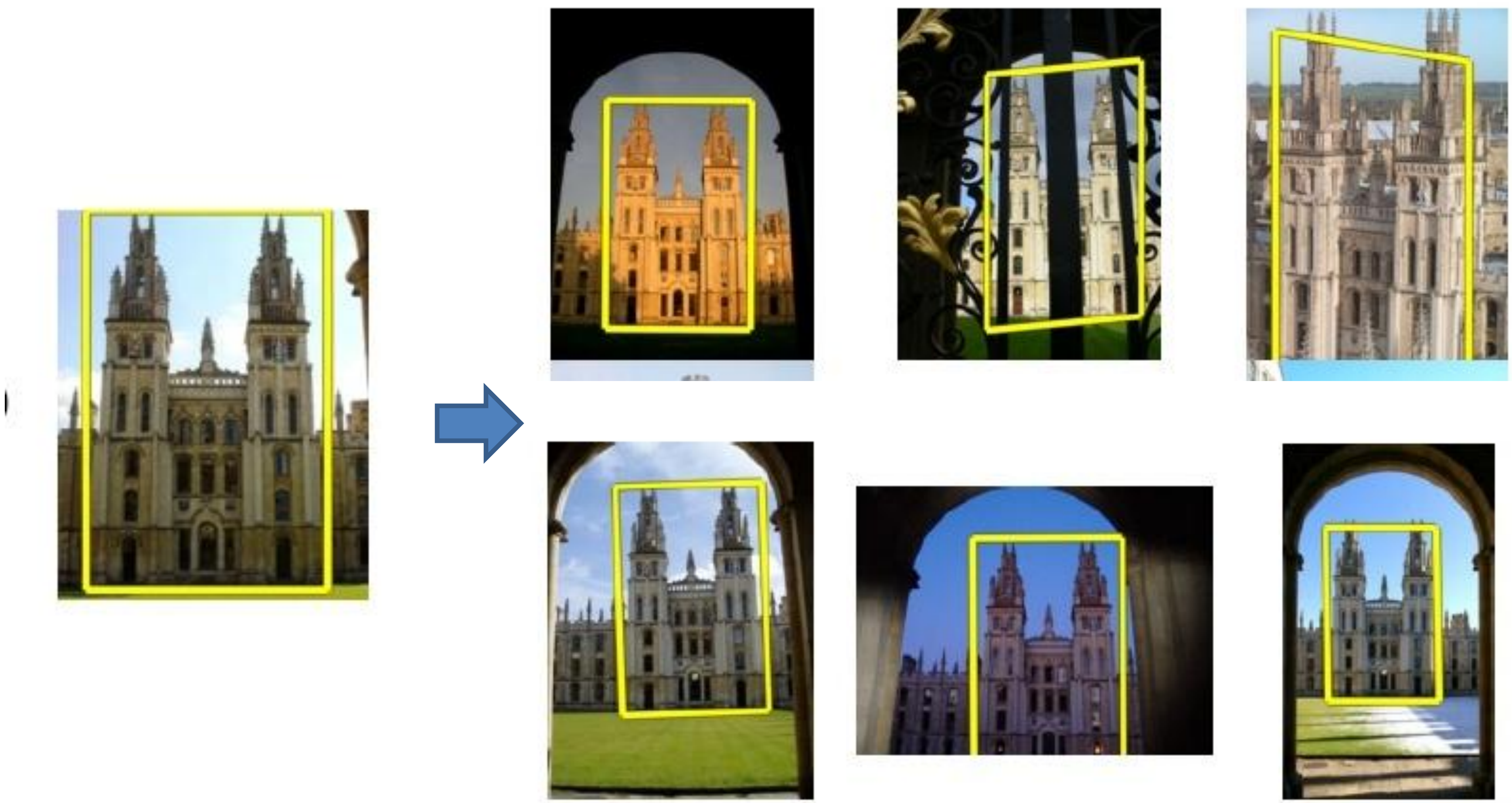


Matching two given views for depth

Search for a matching view for recognition

Kristen Grauman

How to quickly find images in a large database that match a given image region?

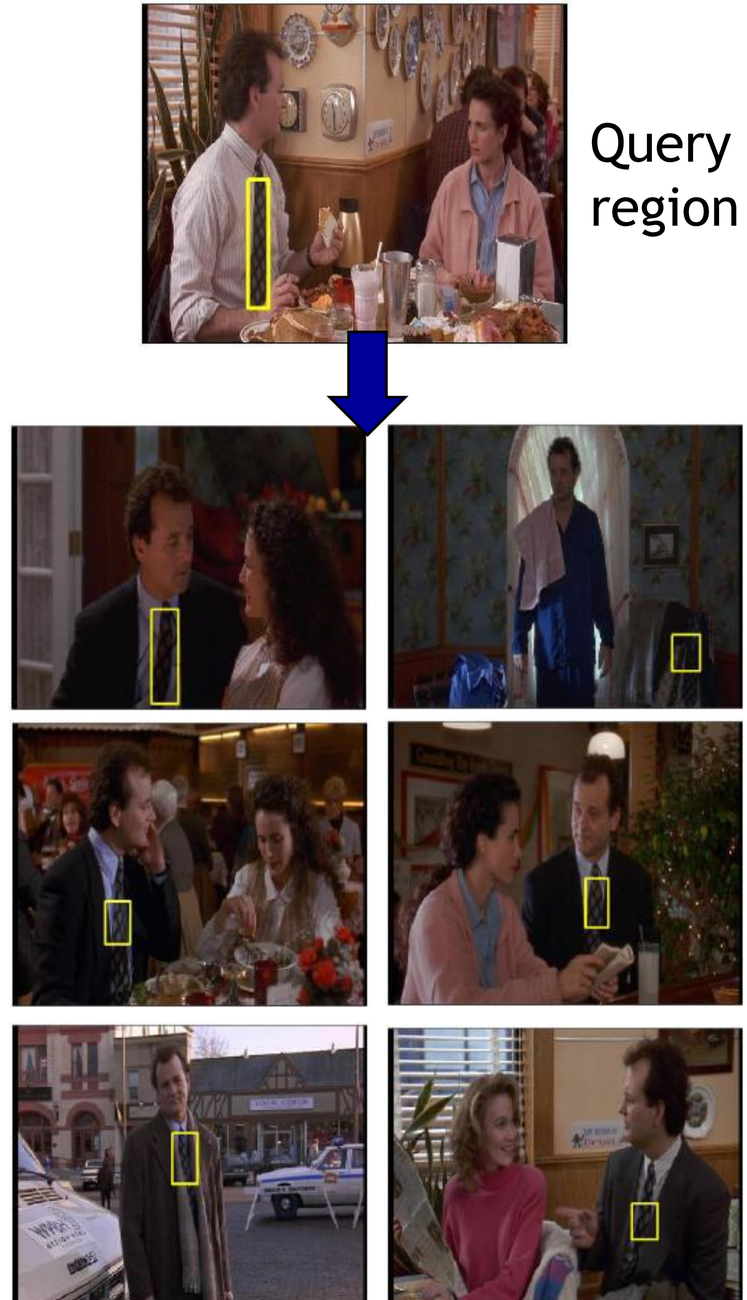


Video Google System

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

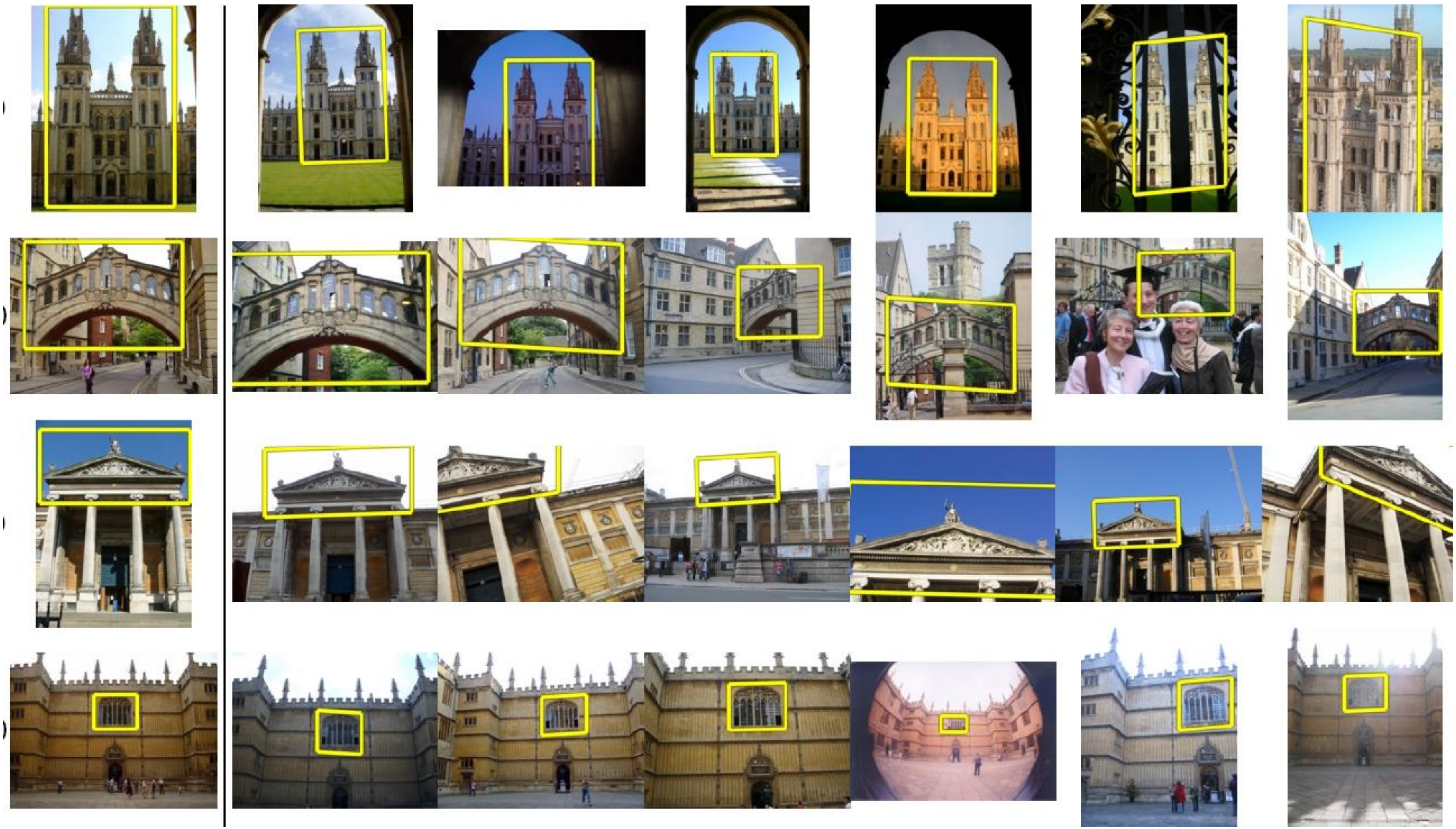
Kristen Grauman

Example Applications

Mobile tourist guide
- Self-localization
- Object/building recognition
- Photo/video augmentation

[Quack, Leibe, Van Gool, CIVR'08]

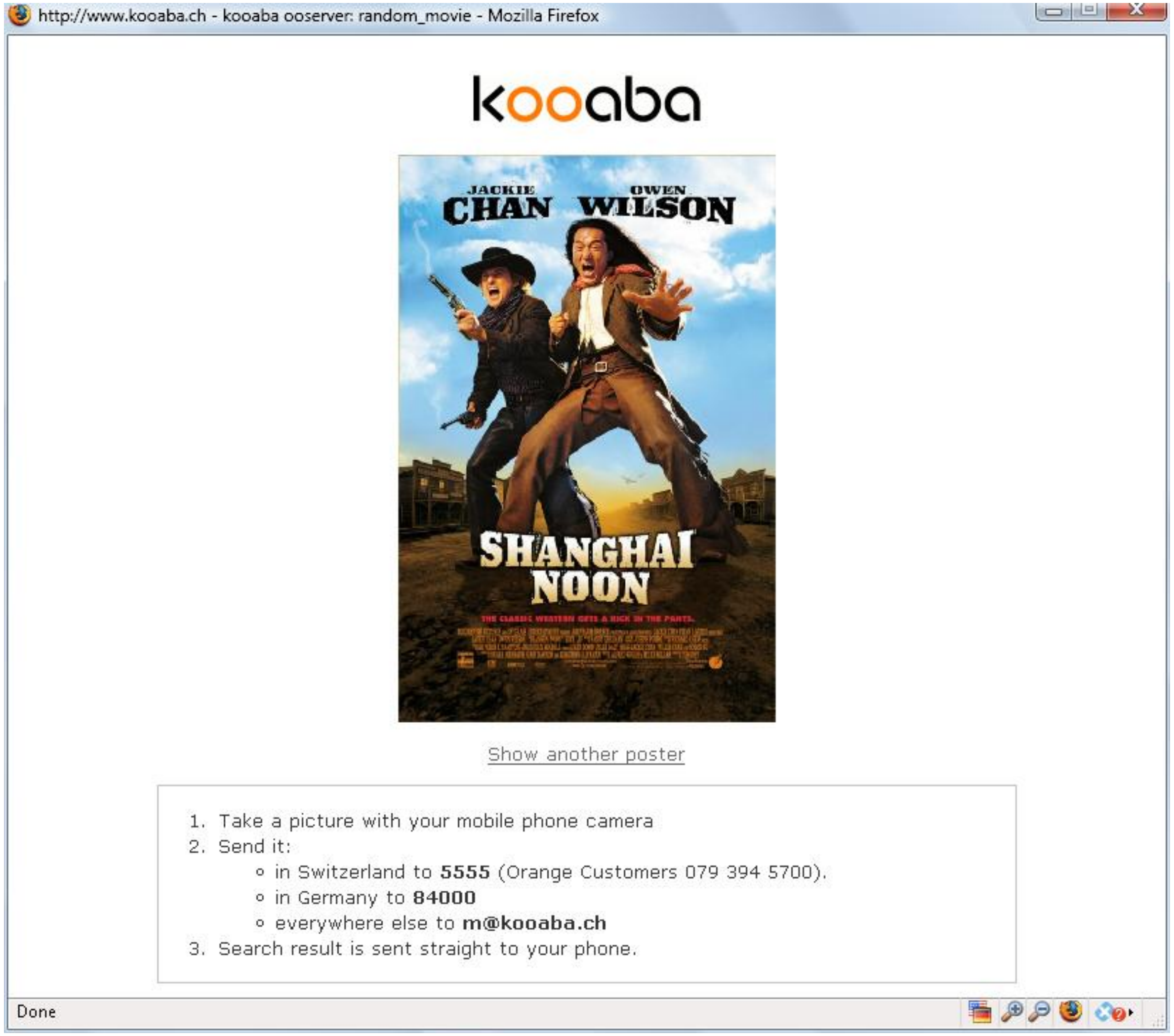
Application: Large-Scale Retrieval



Query Results from 5k Flickr images (demo available for 100k set)

[Philbin CVPR'07]

Web Demo: Movie Poster Recognition

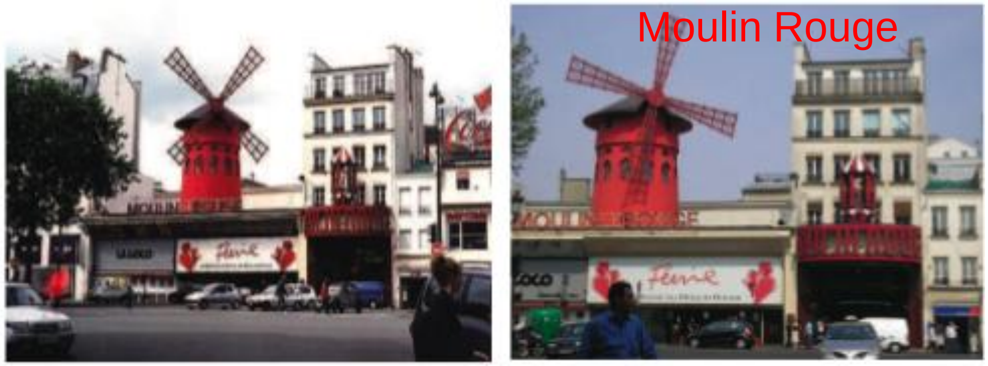


50’000 movie posters indexed

Query-by-image from mobile phone available in Switzer-land

http://www.kooaba.com/en/products_engine.html#

Application: Image Auto-Annotation

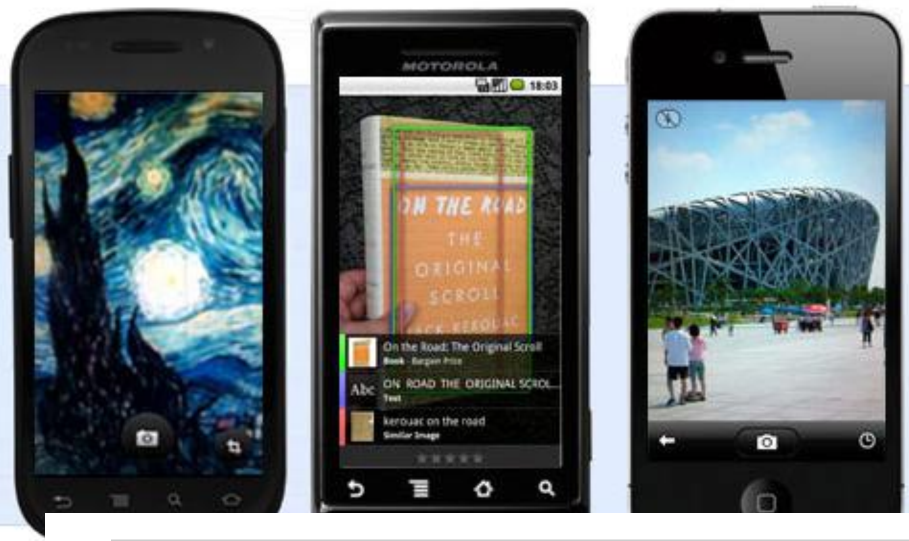


Left: Wikipedia image
Right: closest match from Flickr

[Quack CIVR'08]

Google Goggles

Use pictures to search the web. ▷ Watch a video



Get Google Goggles

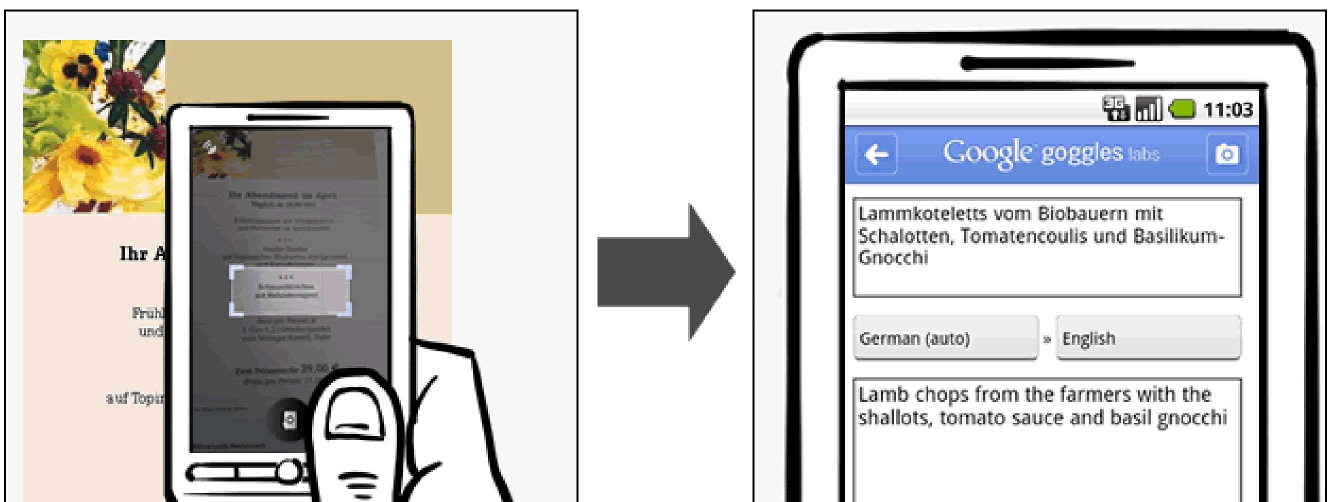
Android (1.6+ required)
Download from Android Market.

Send Goggles to Android phone

New! **iPhone (iOS 4.0 required)**
Download from the App Store.

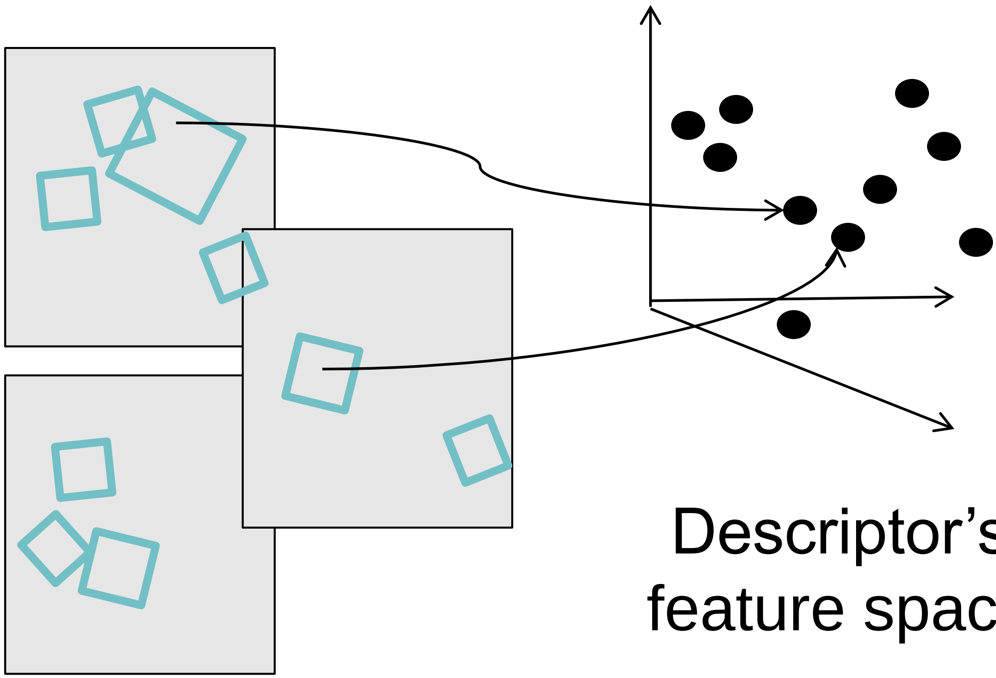
Send Goggles to iPhone

Text | Landmarks | Books | Contact Info | Artwork | Wine | Logos



Indexing local features

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

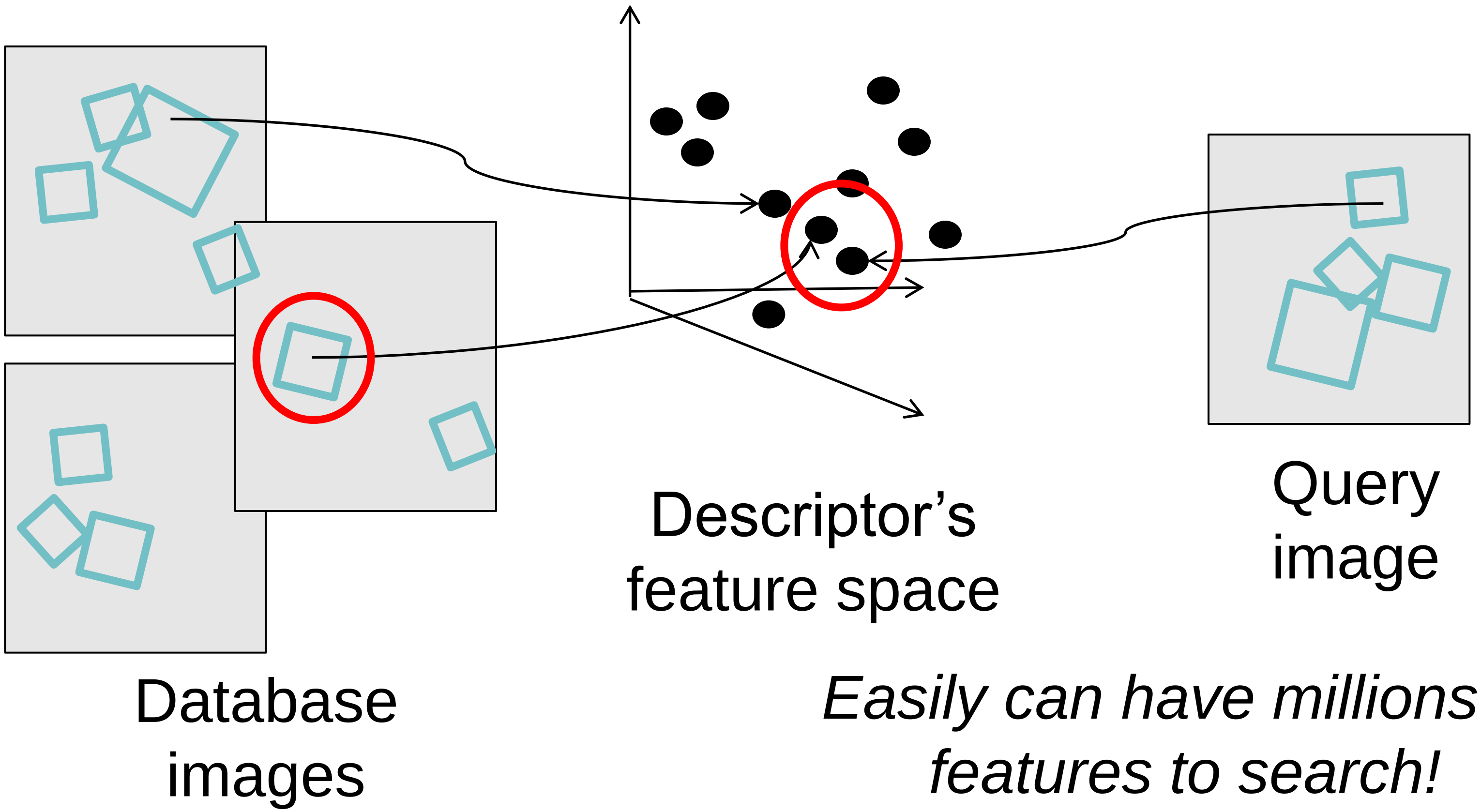


Descriptor's feature space

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 Grau

Indexing local features: inverted file index

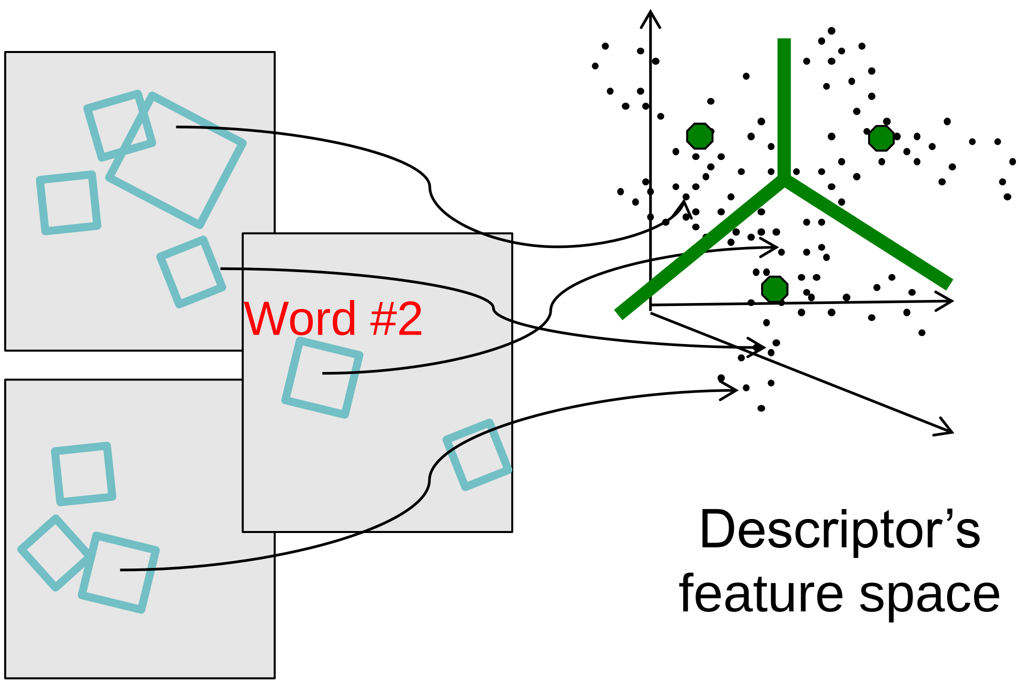
Index

- 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

Visual words

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

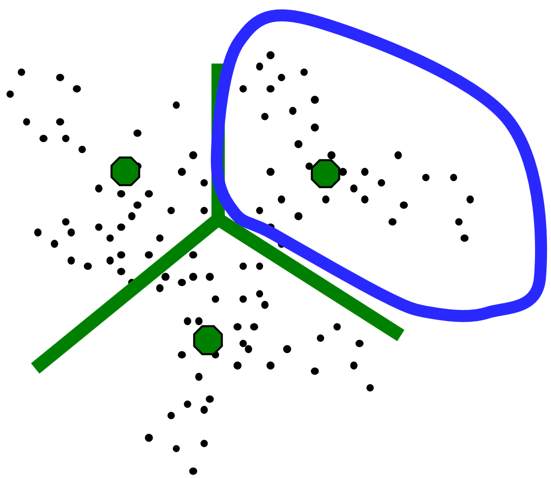


- Quantize via clustering, let cluster centers be the prototype "words"
- Determine which word to assign to each new image region by finding the closest cluster center.

Kristen Grauman

Visual words

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



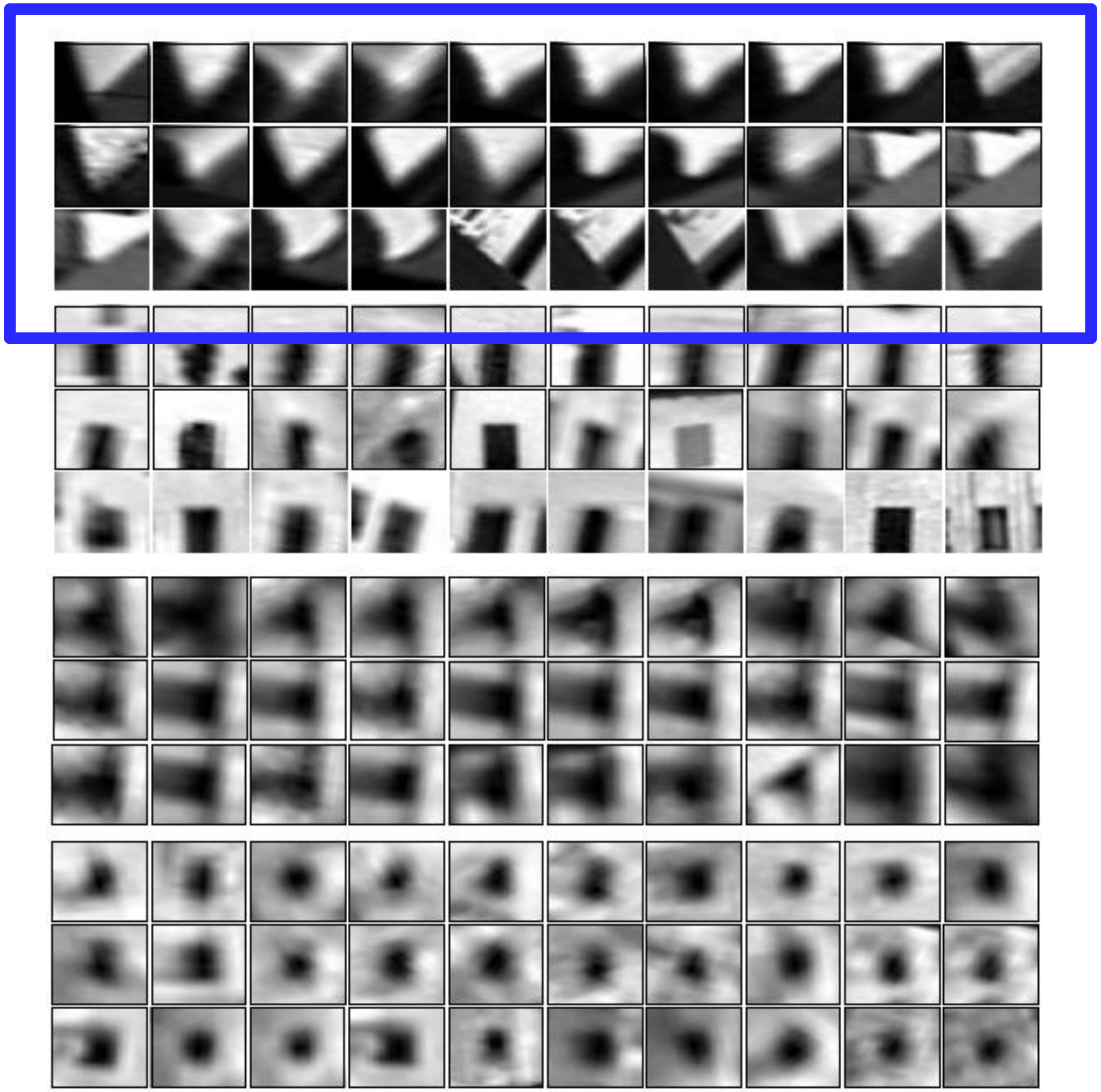


Figure from Sivic & Zisserman, ICCV 2003

Kristen Grauman

Visual vocabulary formation

Issues:

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

Kristen Grauman

Inverted file index

Database images

Image #1

Image #2

Image #3

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

…

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

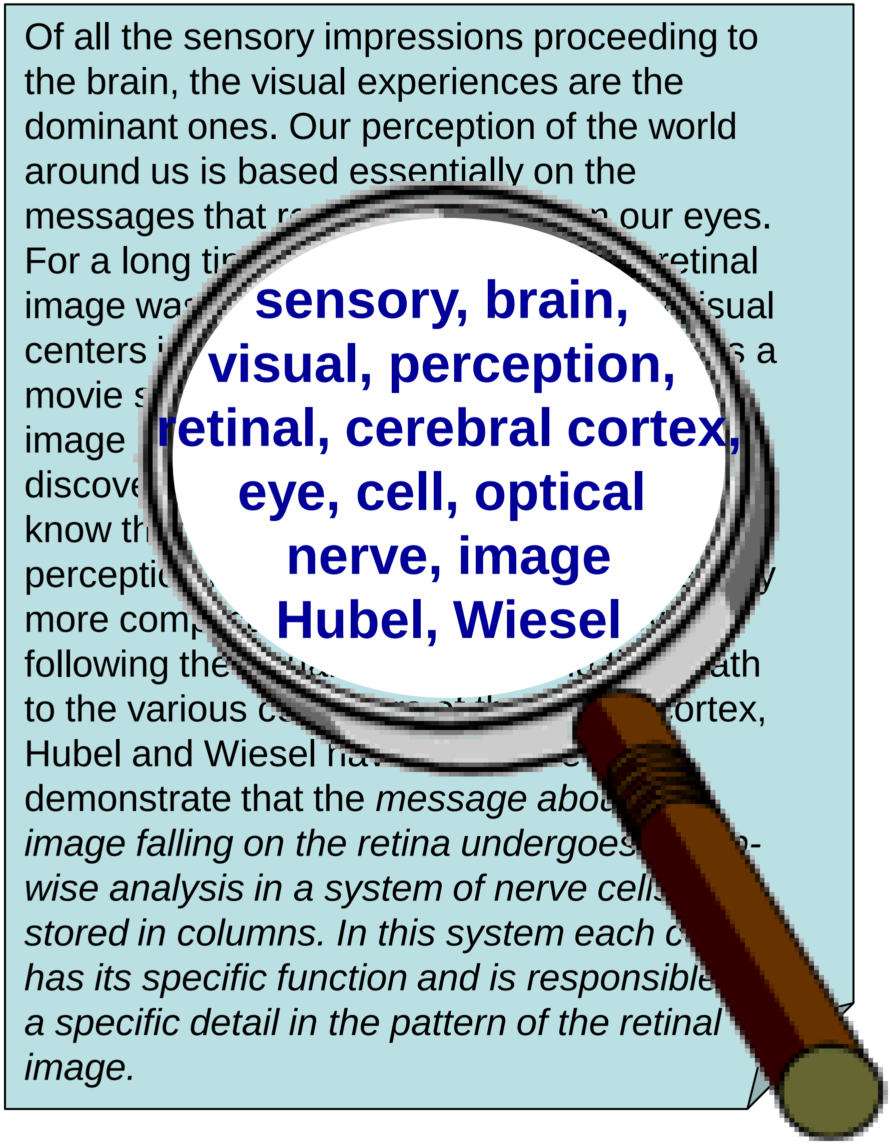
Kristen Grauman

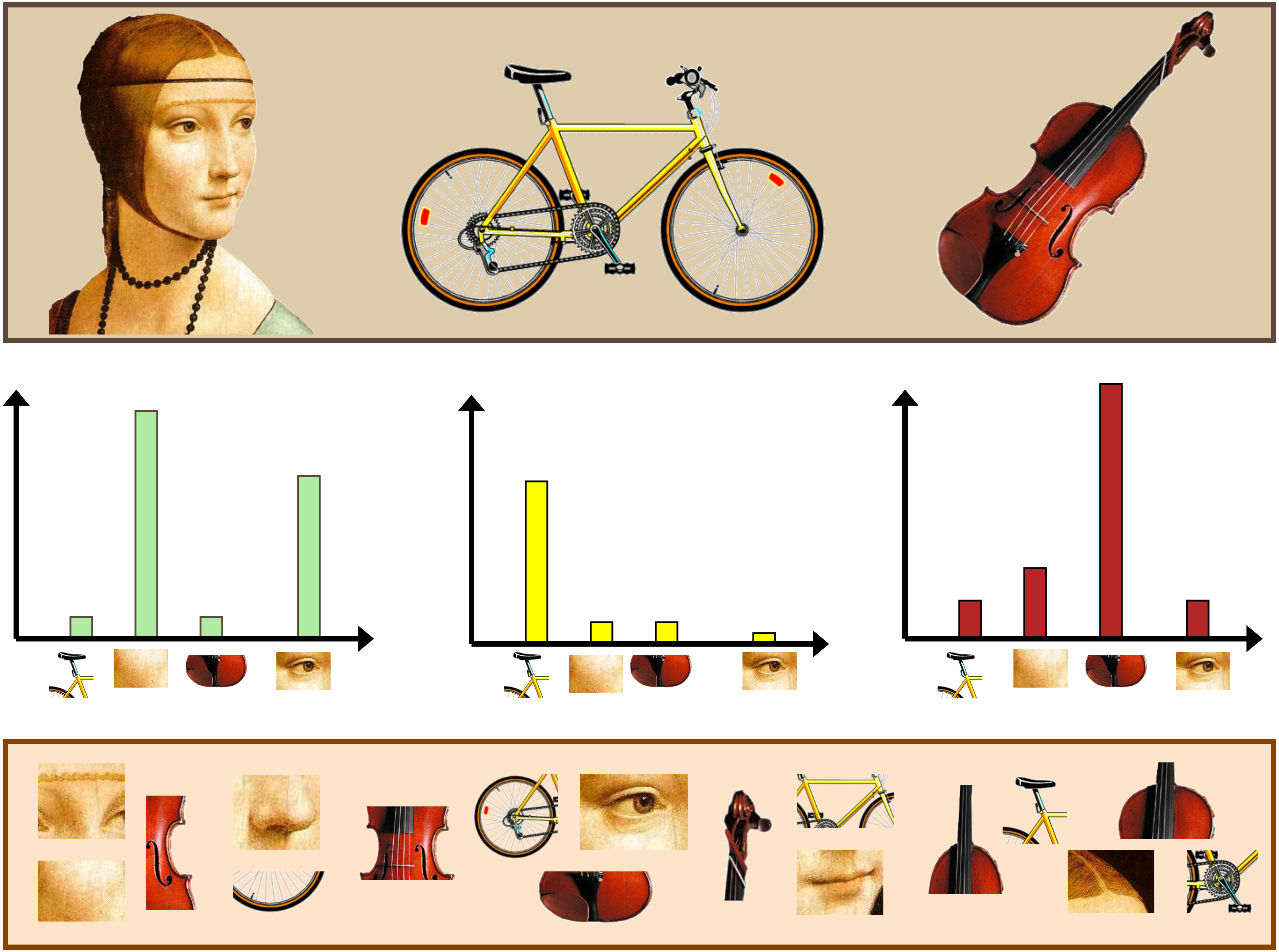
Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

Kristen Grauman

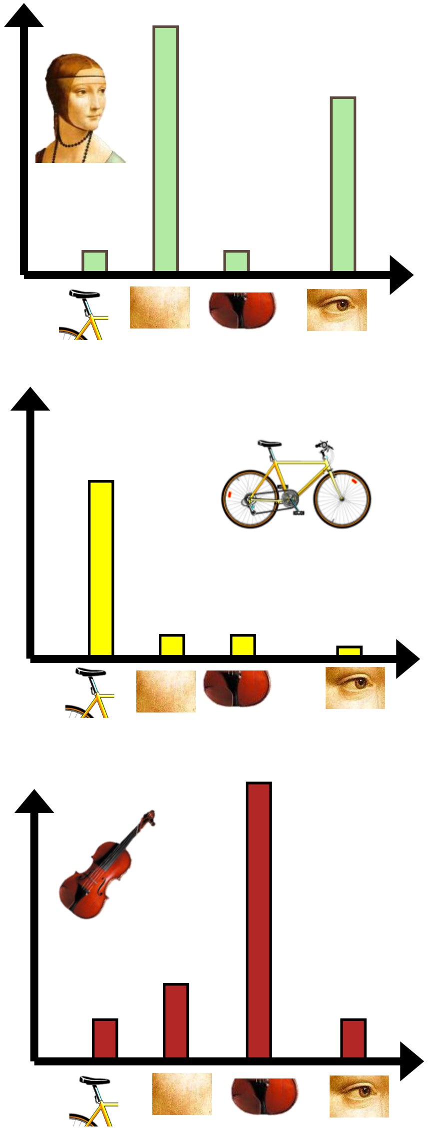
Analogy to documents





Bags of visual words

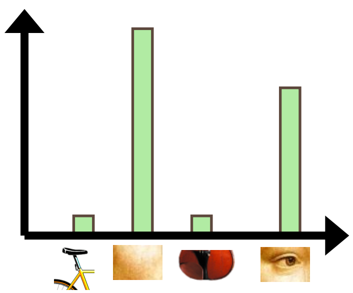
- Summarize entire image based on its distribution (histogram) of word occurrences.
- Analogous to bag of words representation commonly used for documents.



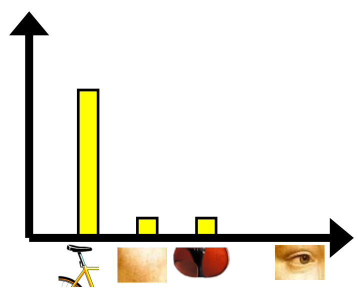
Comparing bags of words

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

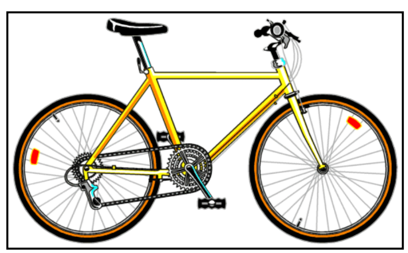
[1 8 1 4]



[5 1 1 0]



$\vec{d}_j$



$\vec{q}$

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

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

for vocabulary of *V* words

Kristen Grauman

Inverted file index and bags of words similarity

New query image

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

…

1. Extract words in query
2. Inverted file index to find relevant frames
3. Compare word counts

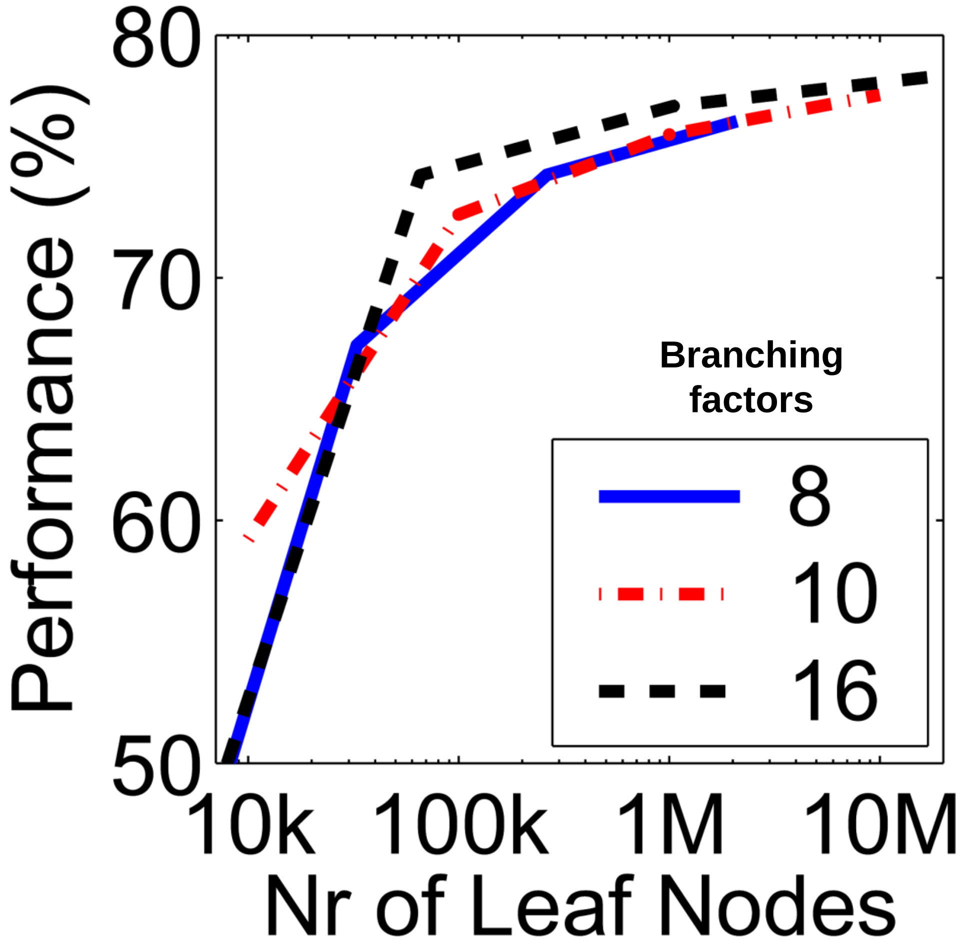
Kristen Grauman

Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

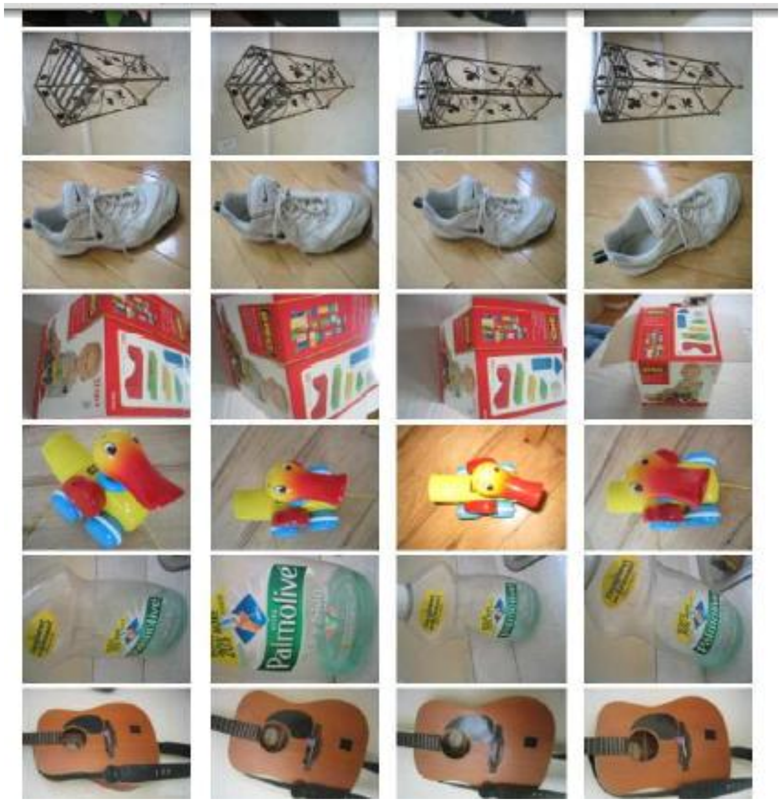
Kristen Grauman

Vocabulary size



Influence on performance, sparsity

Results for recognition task with 6347 images

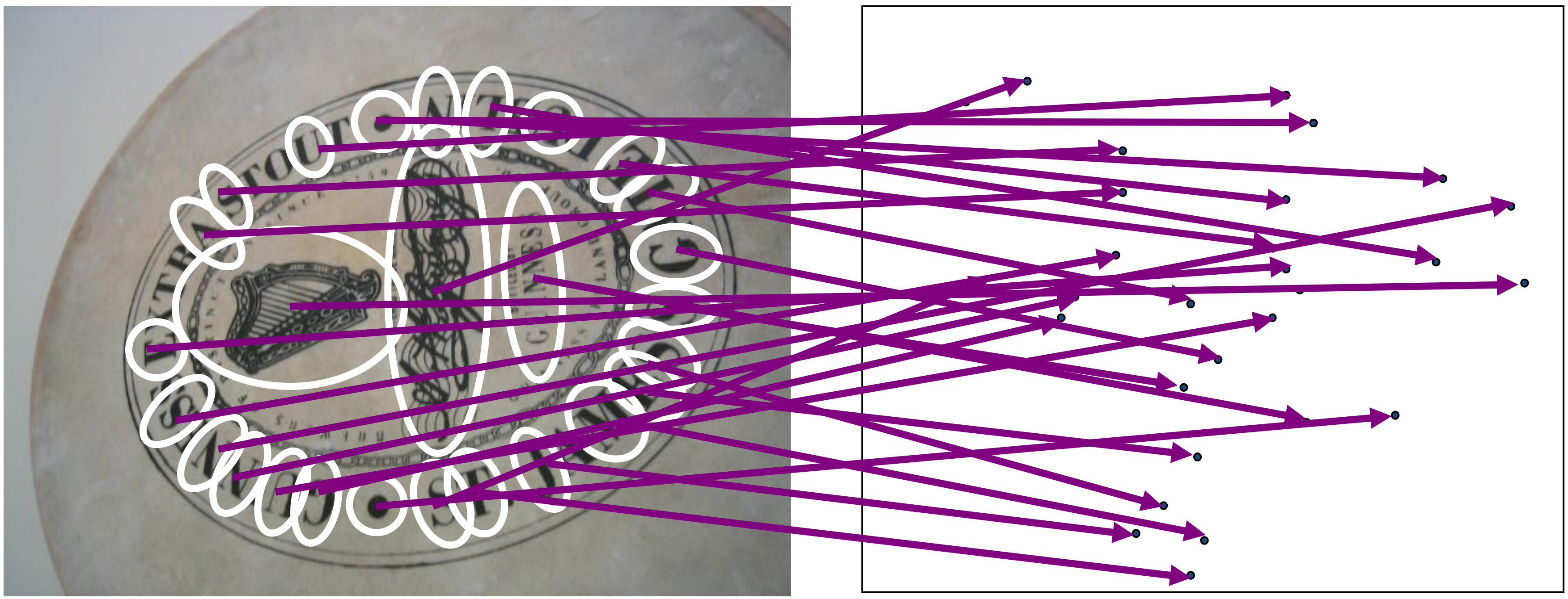


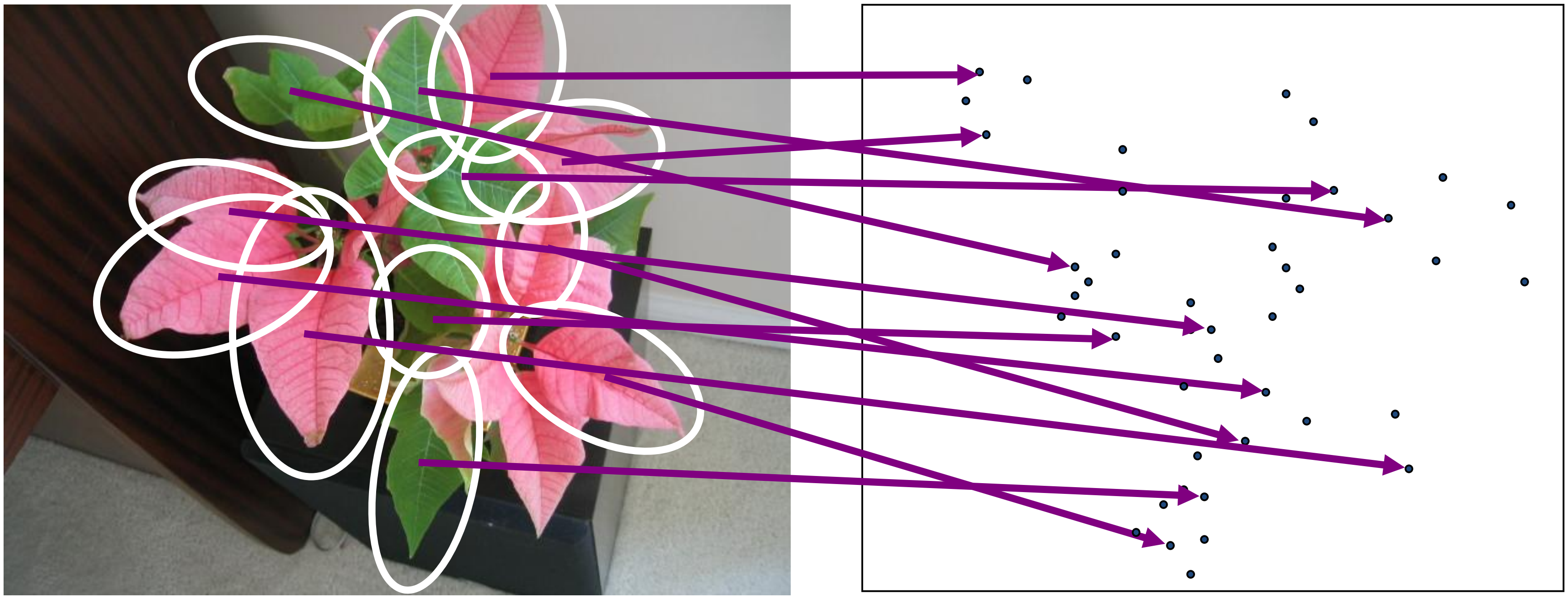
Nister & Stewenius, CVPR 2006

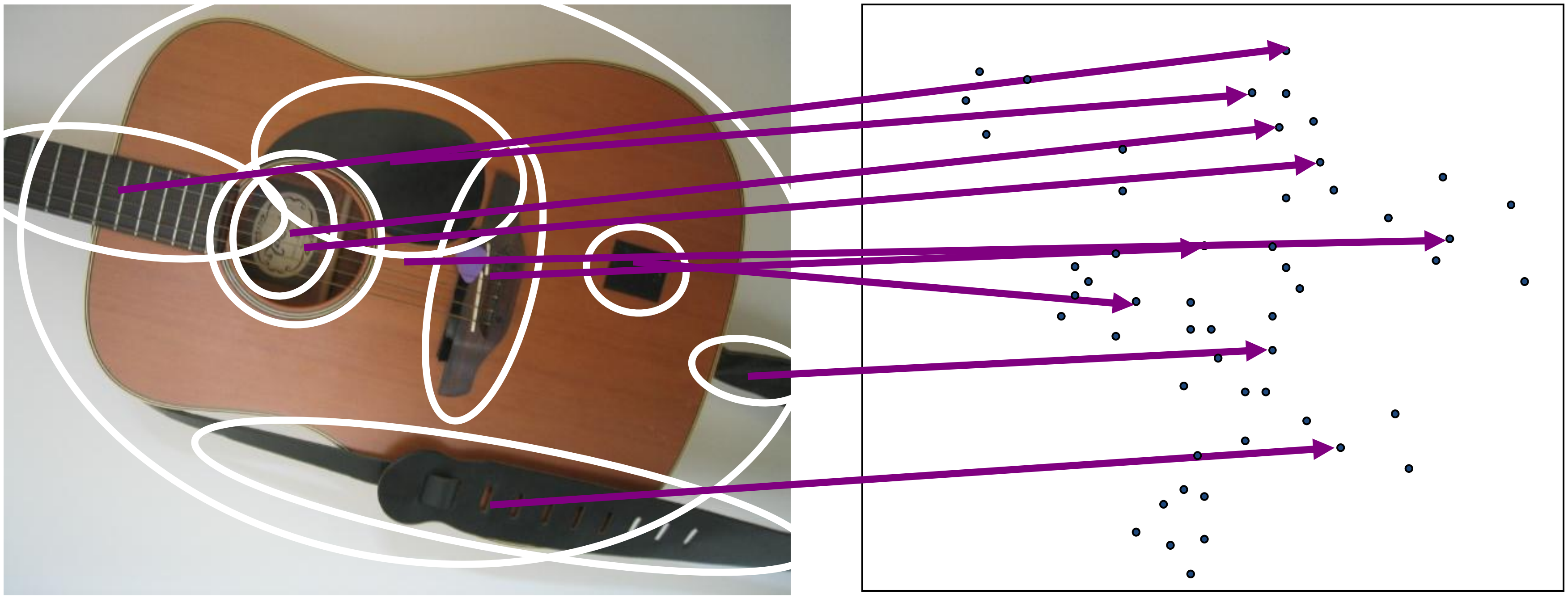
Kristen Grauman

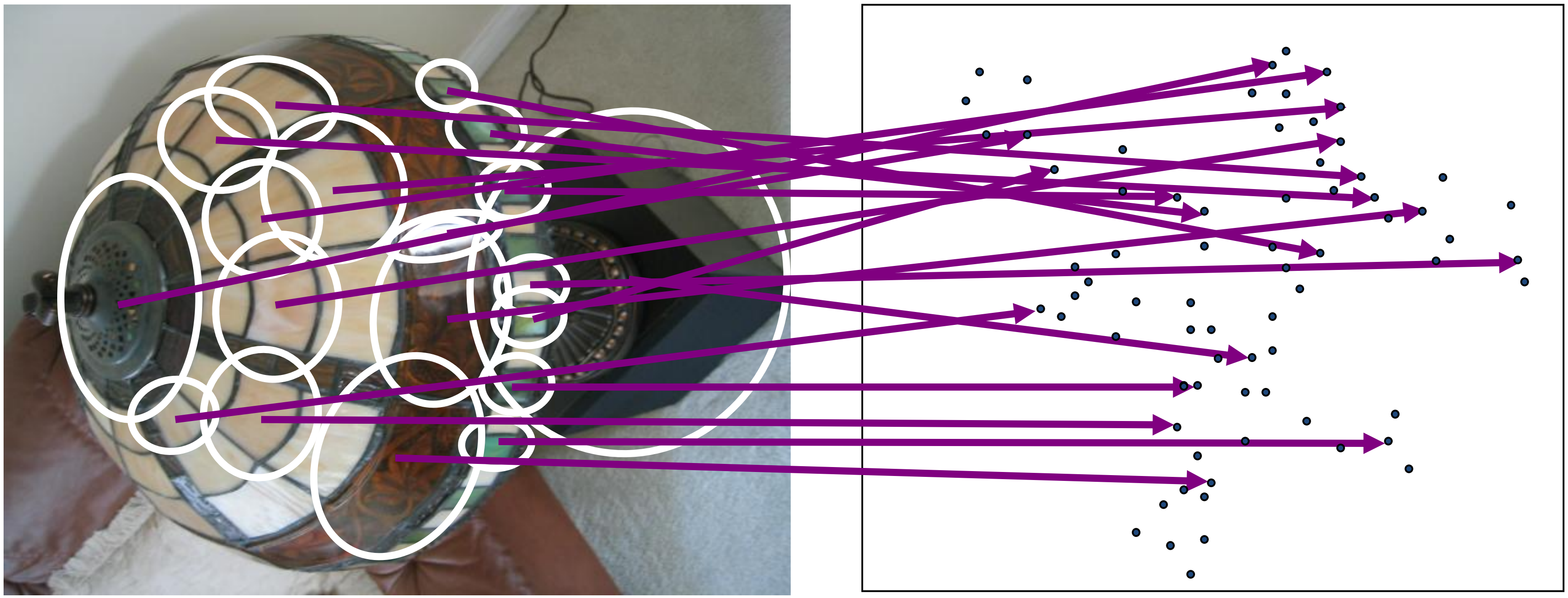
Recognition with K-tree

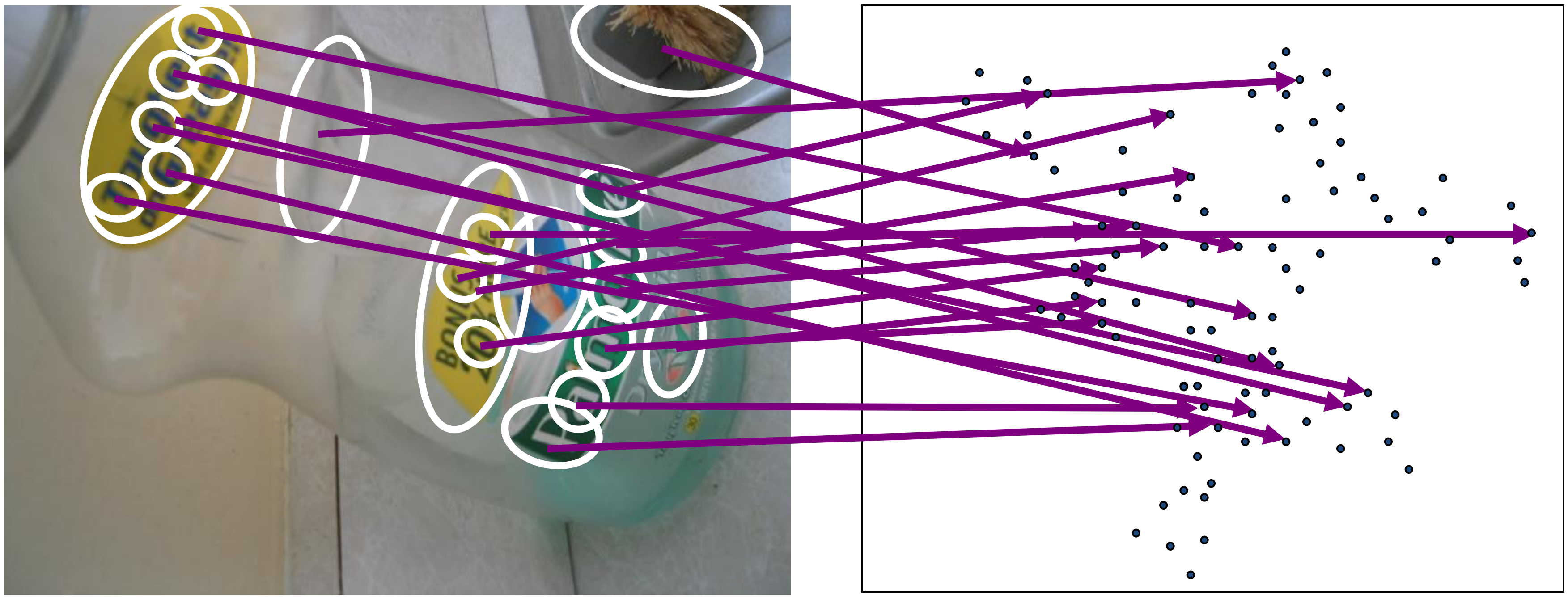
Following slides by David Nister (CVPR 2006)

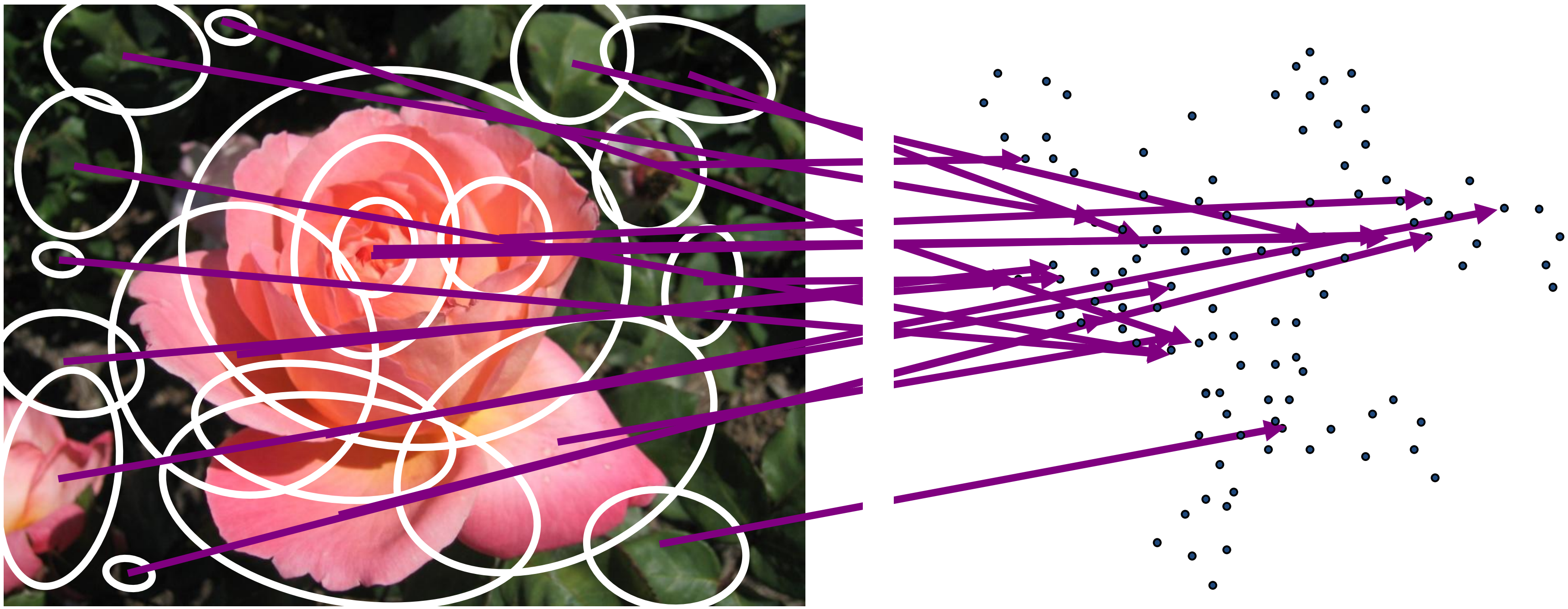


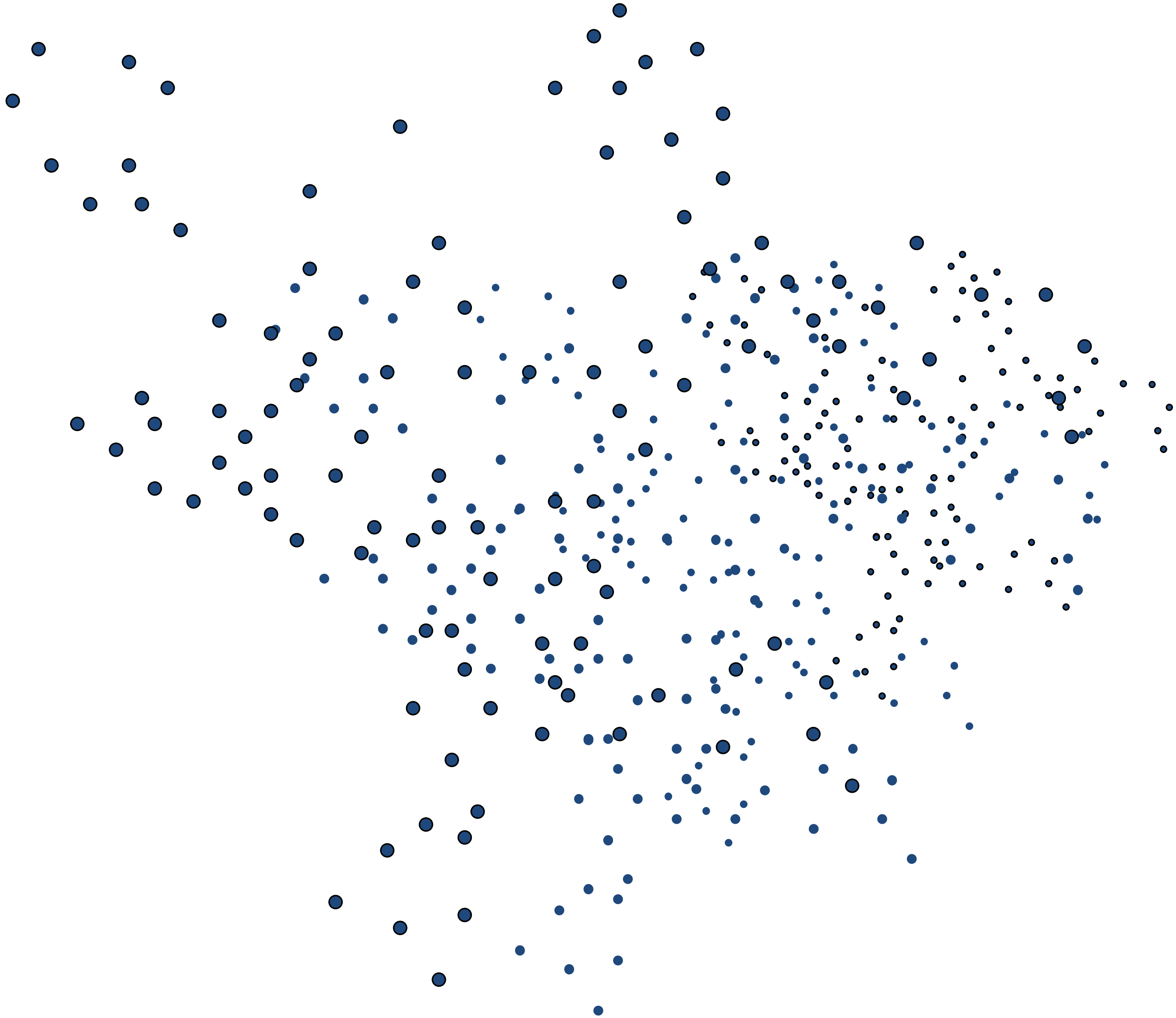


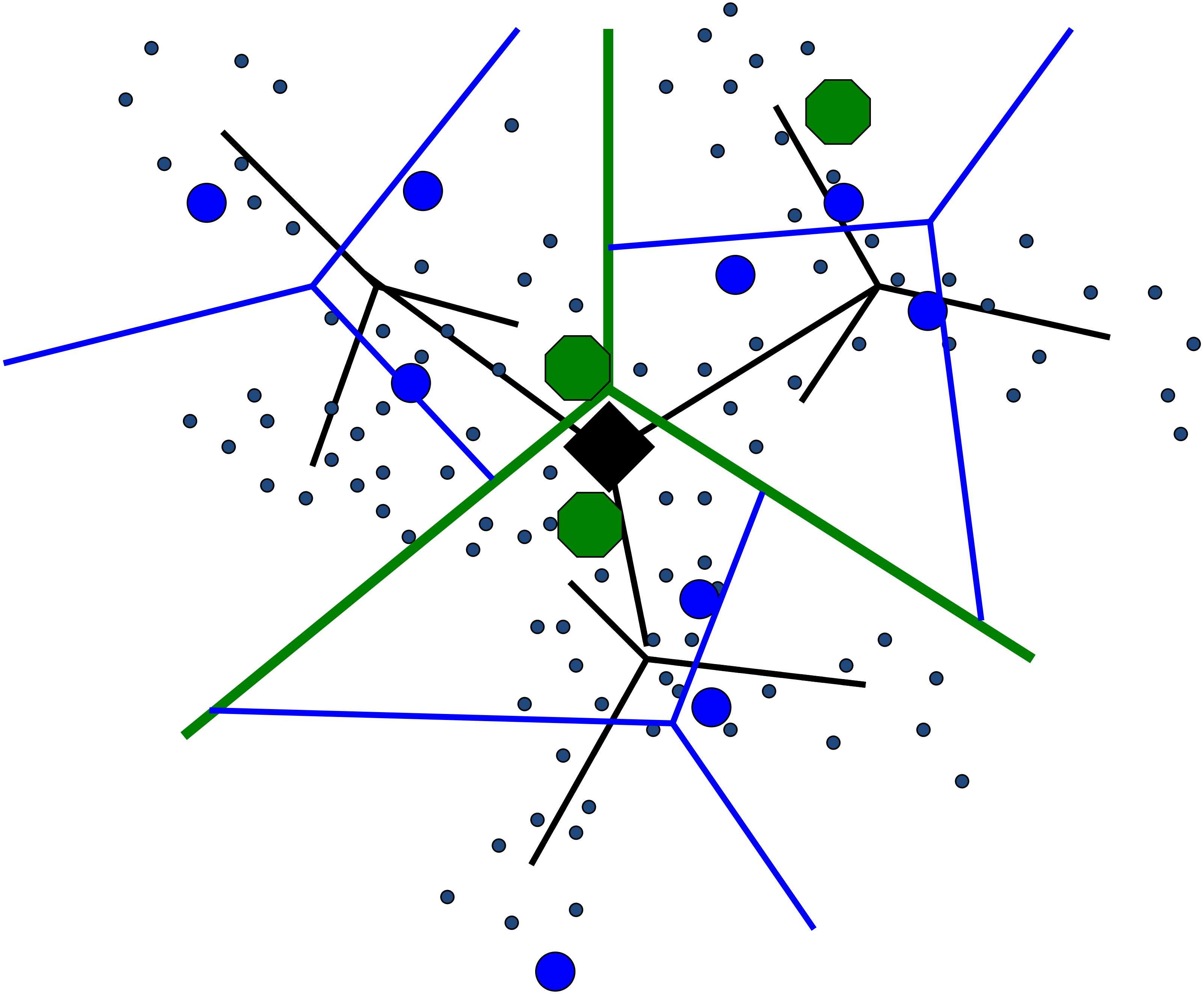


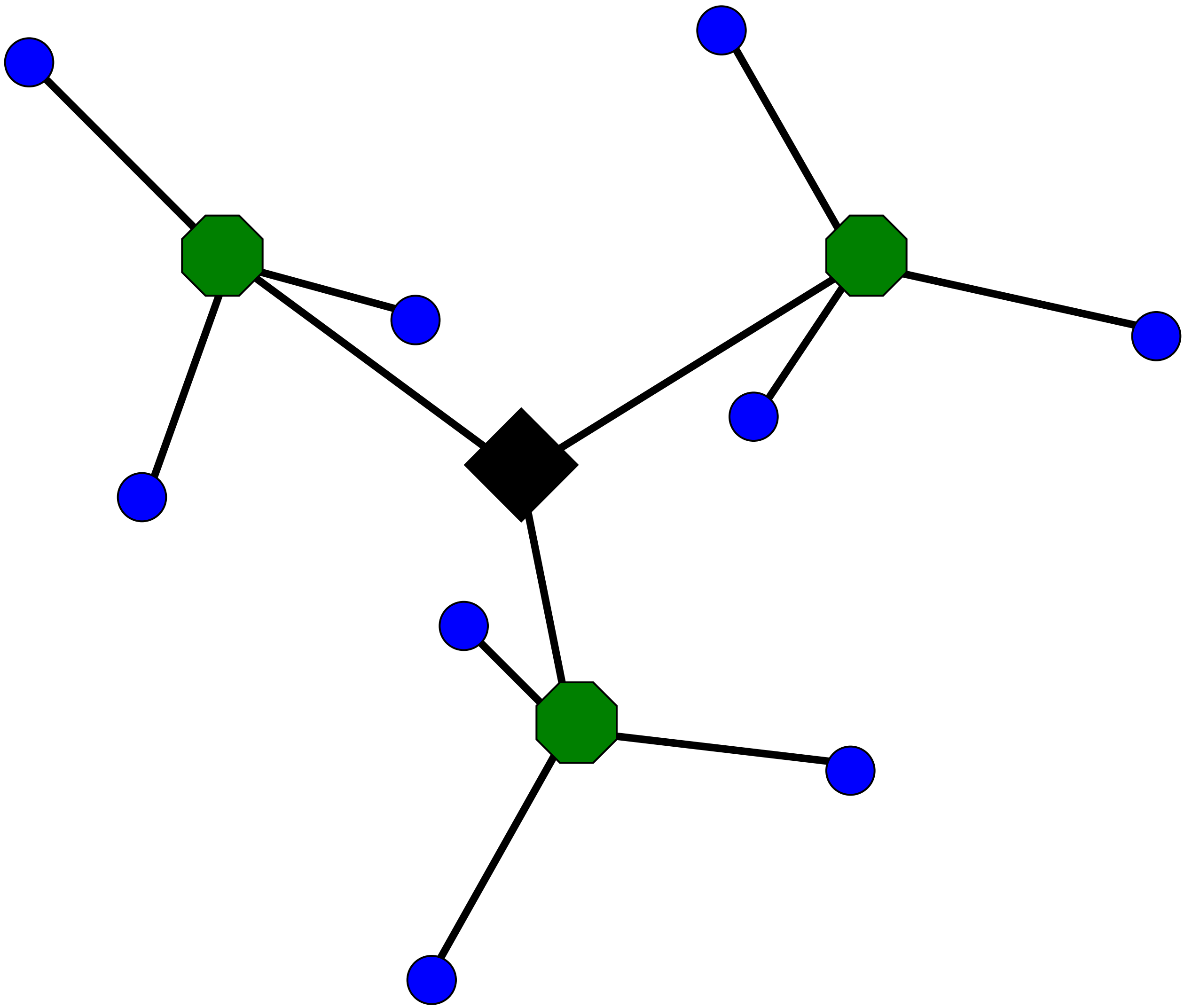


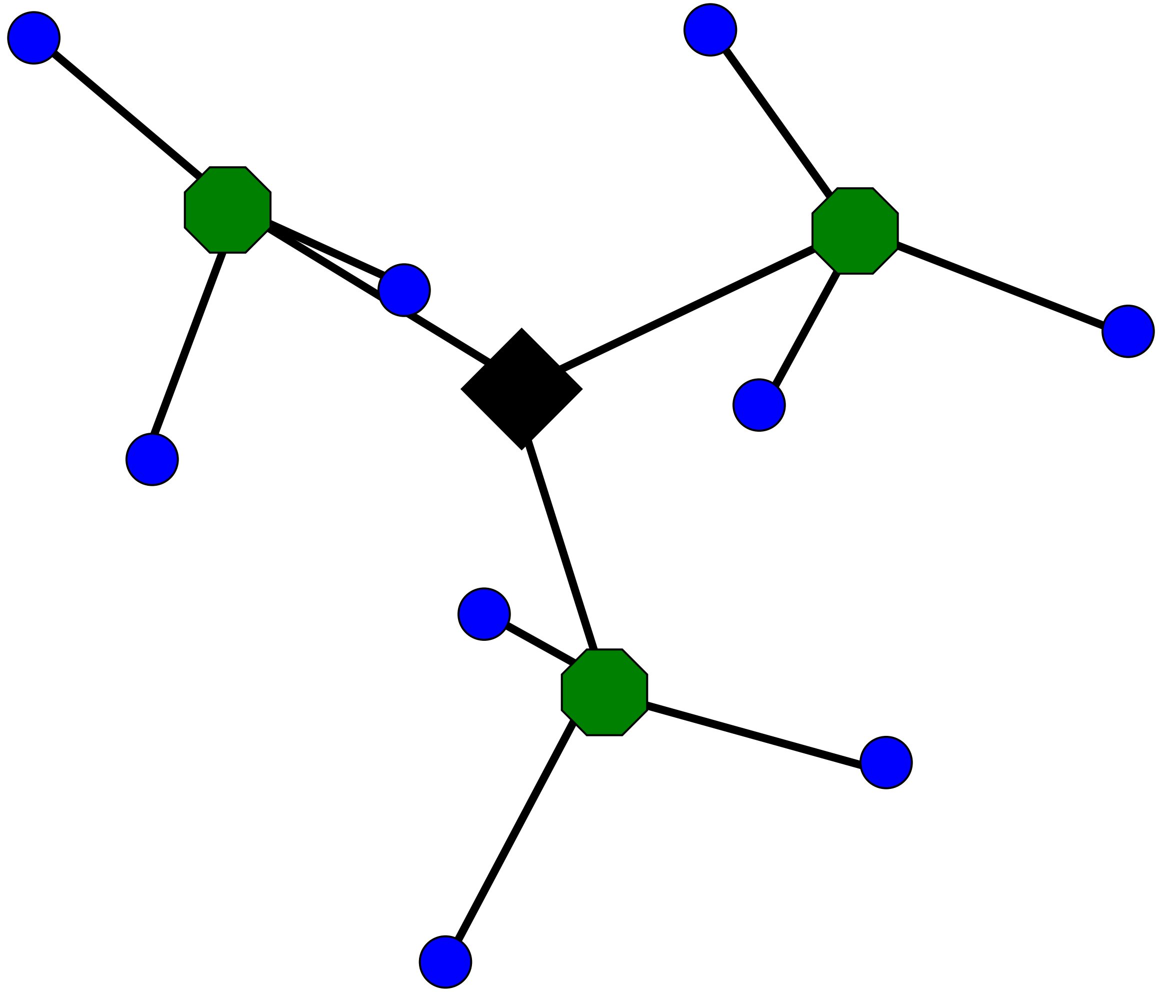


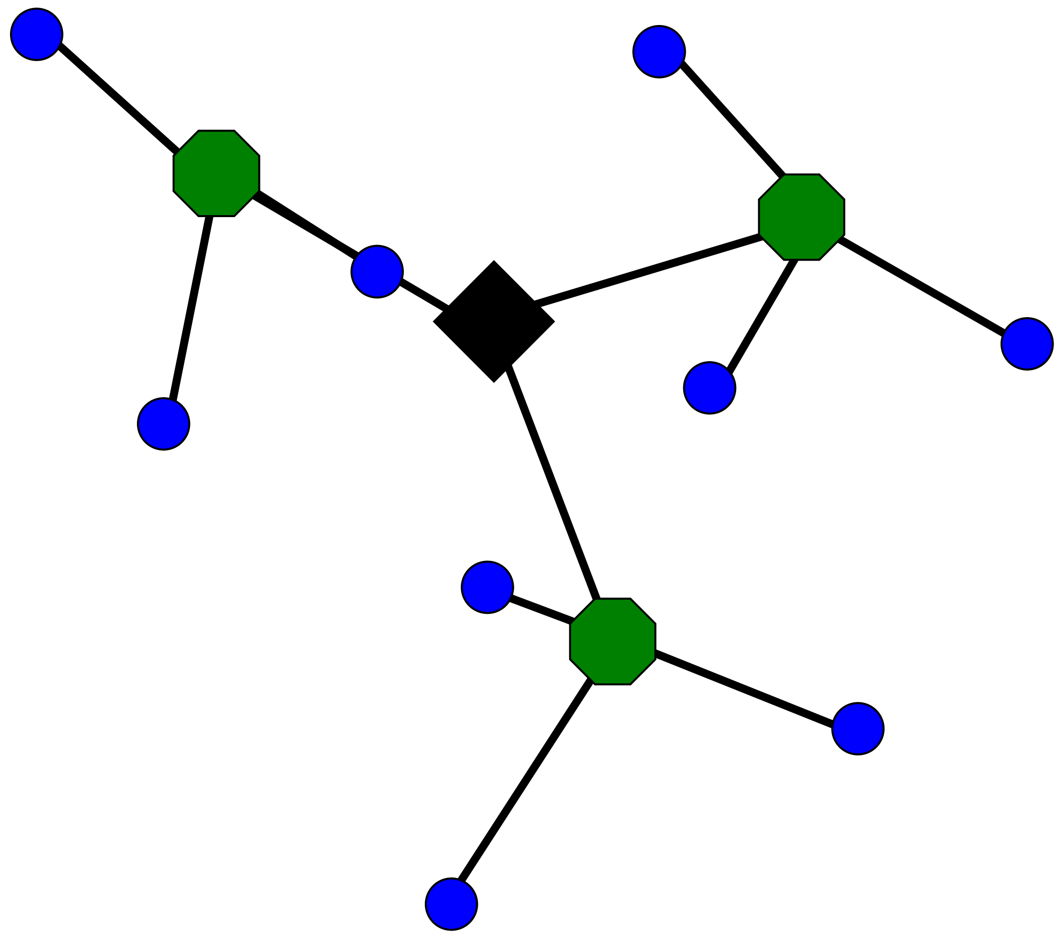


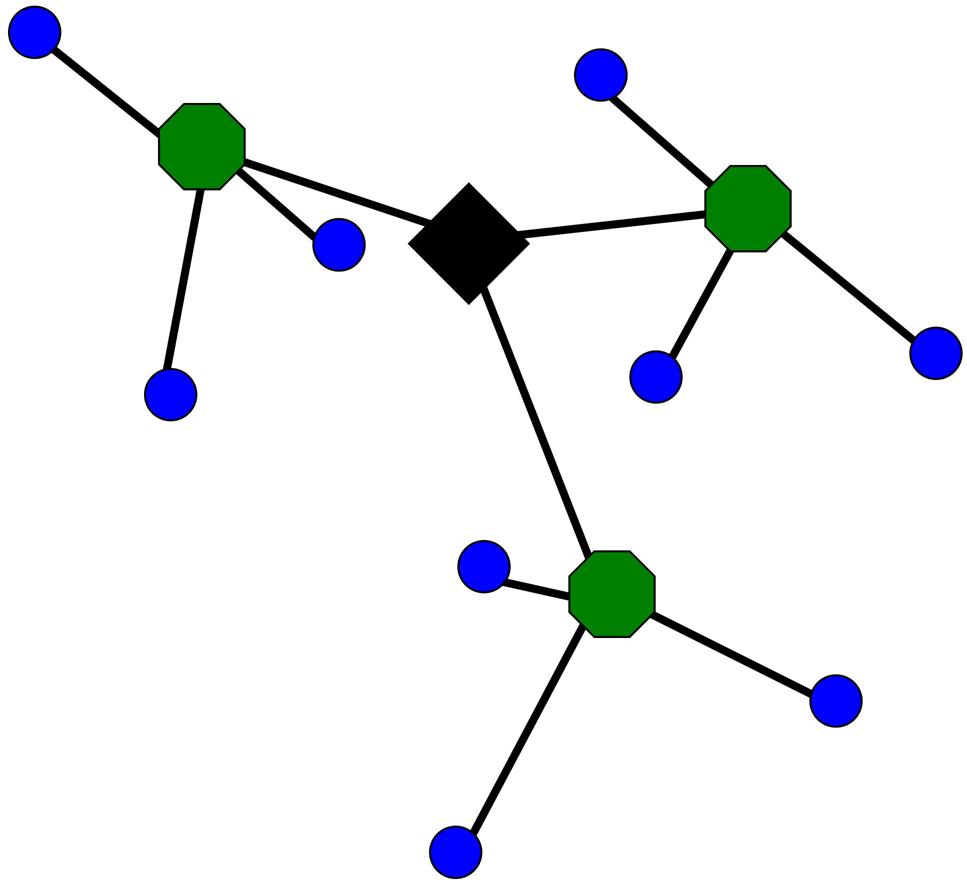


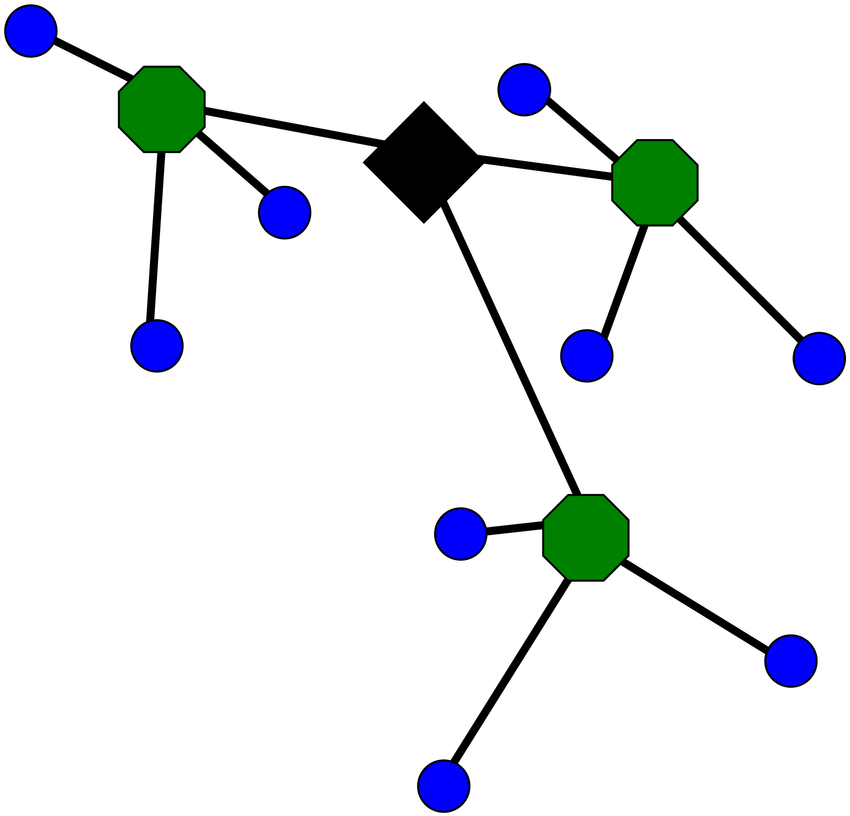


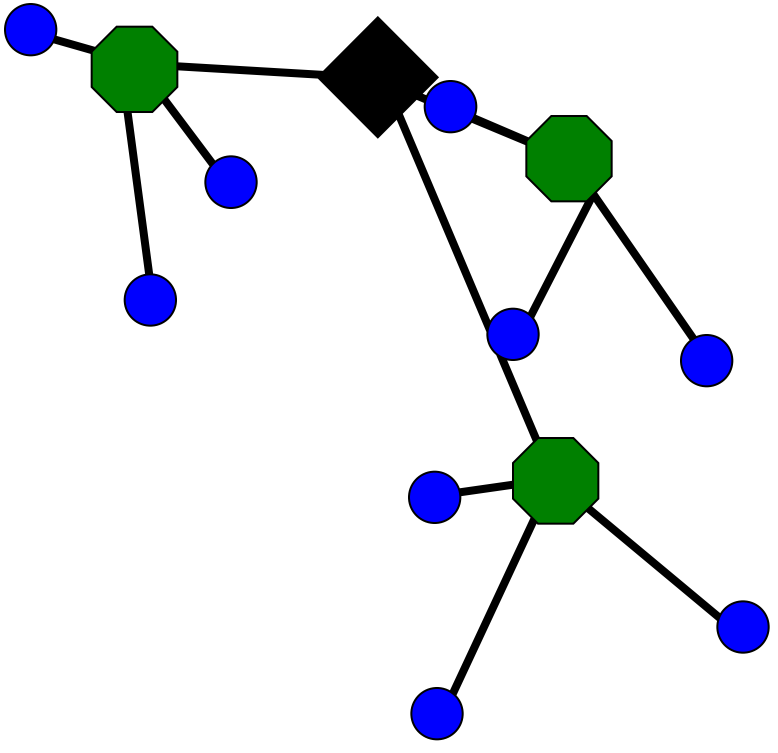


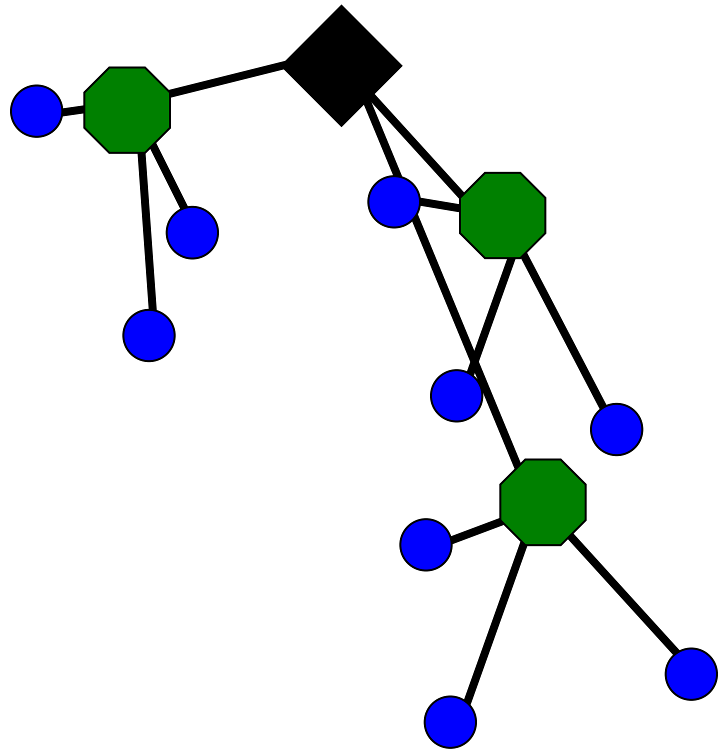


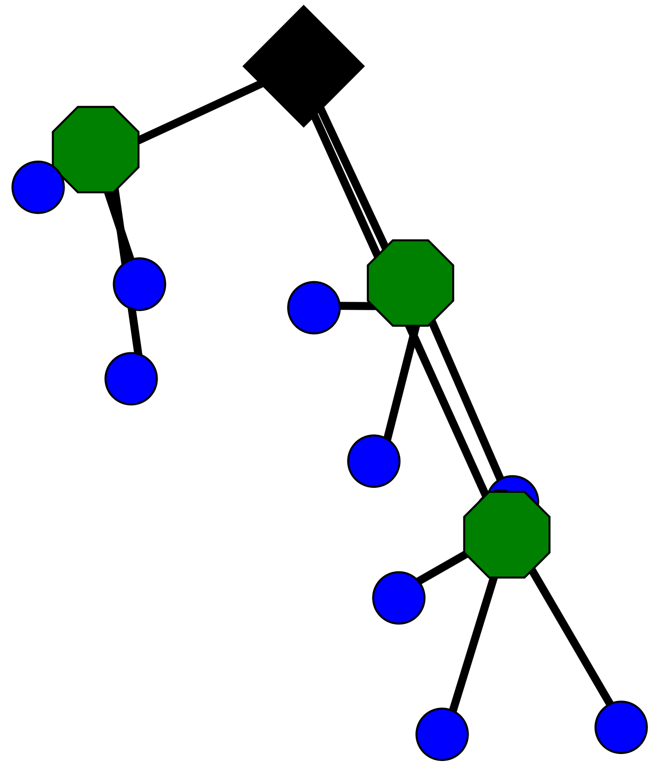


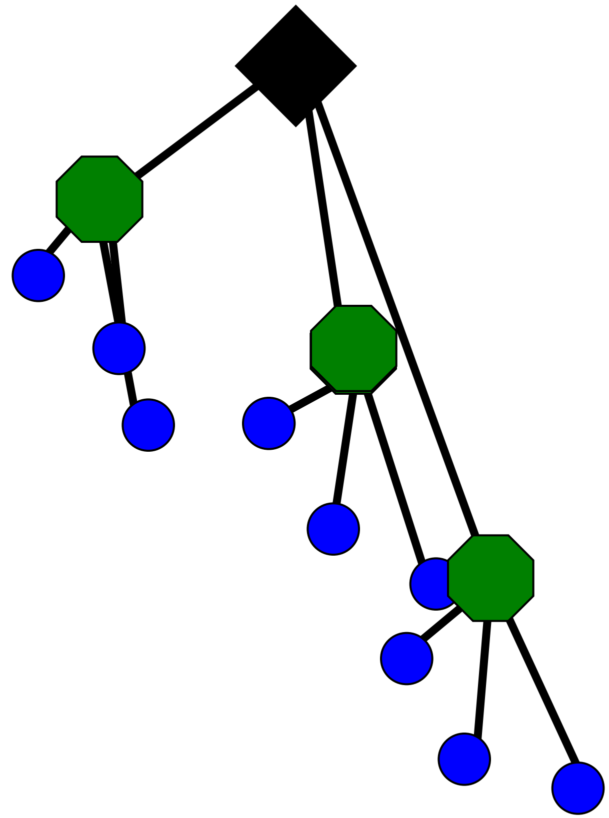


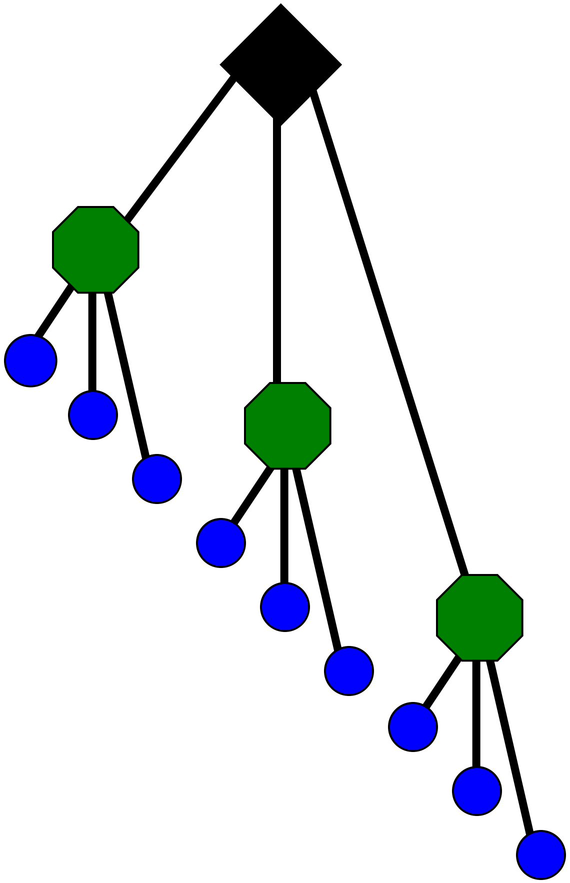


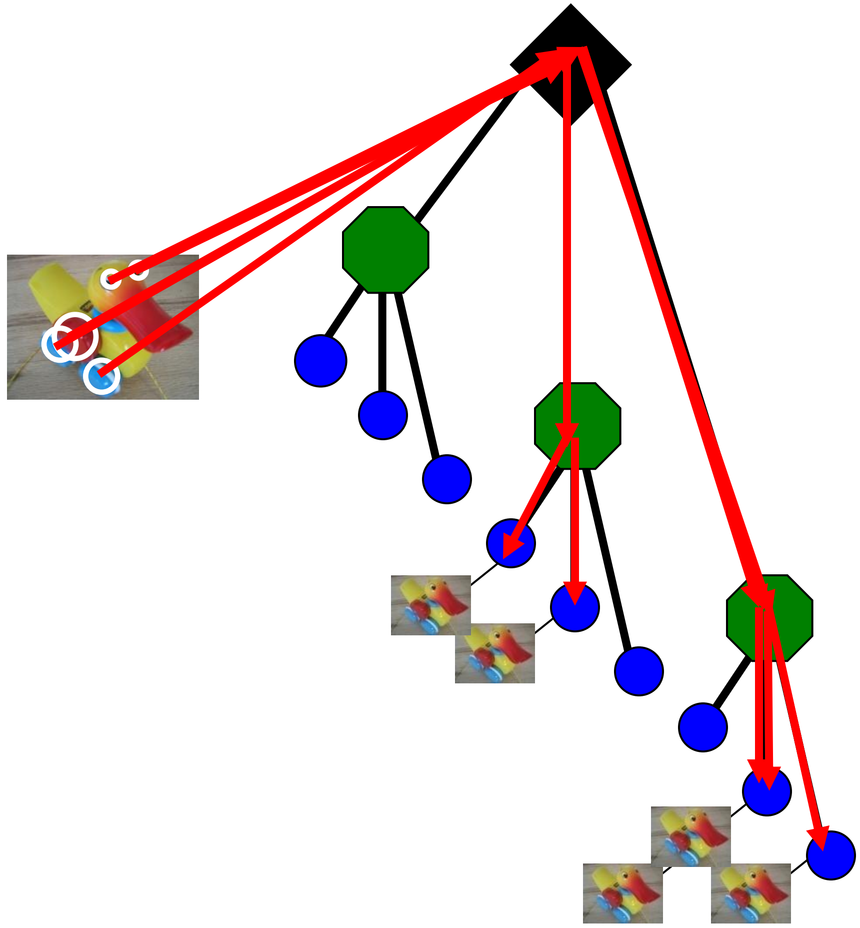


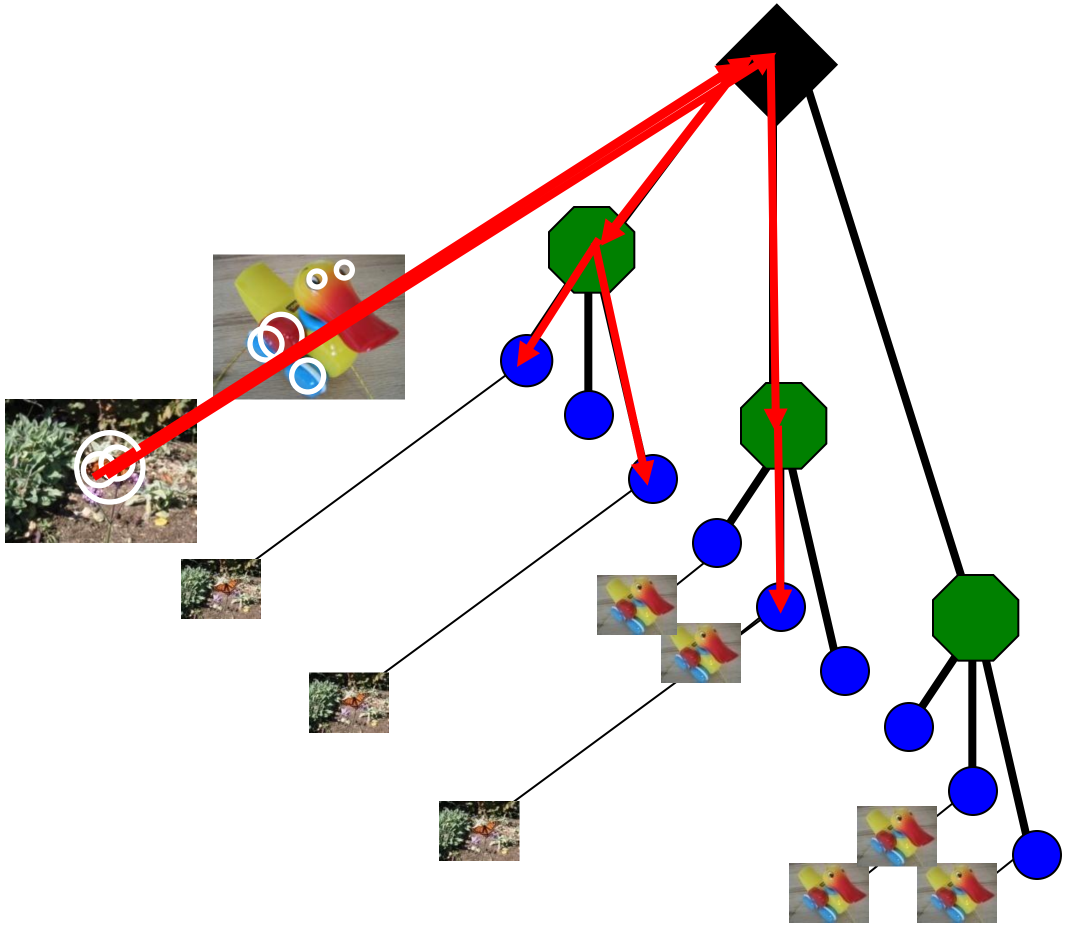


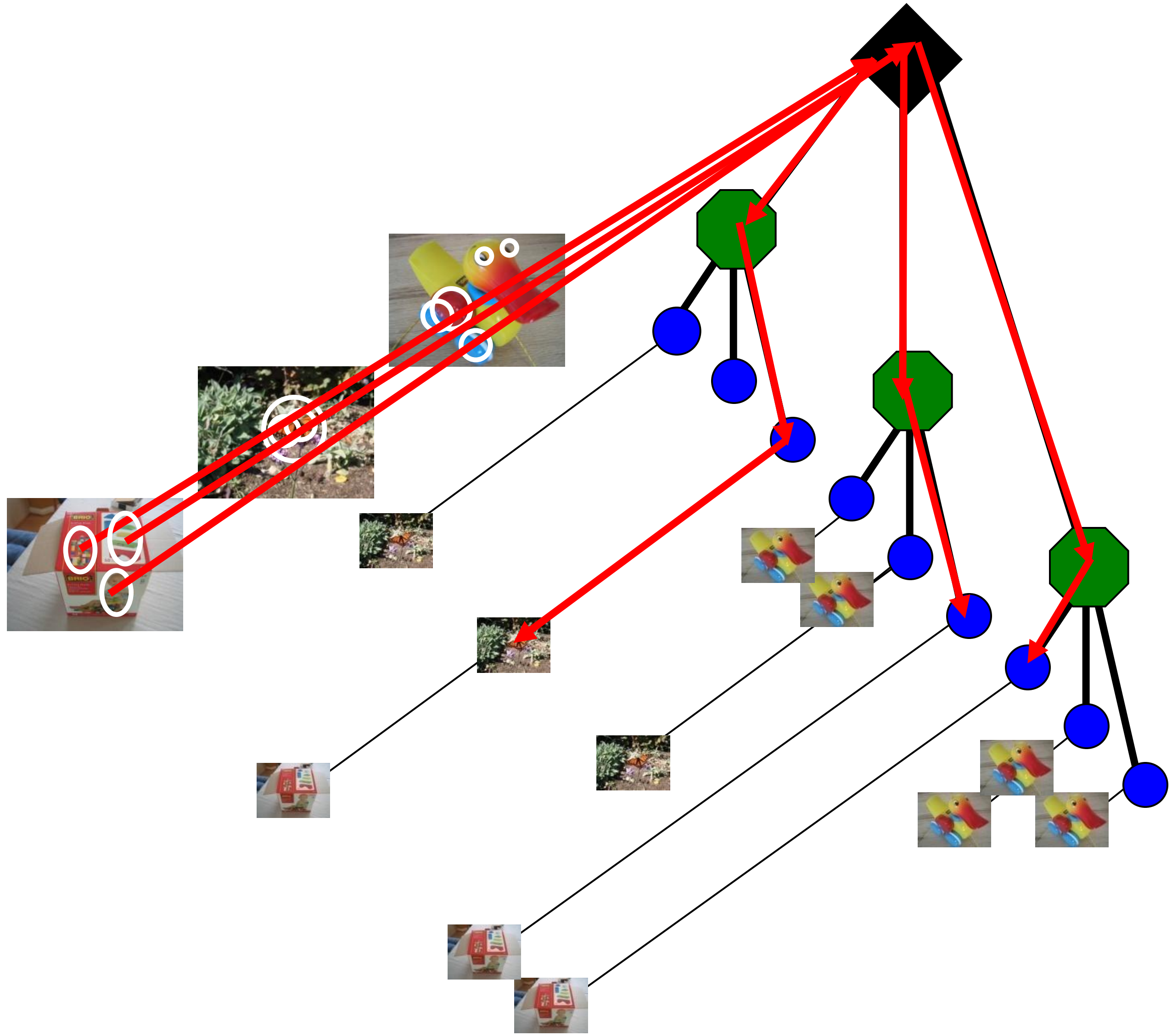


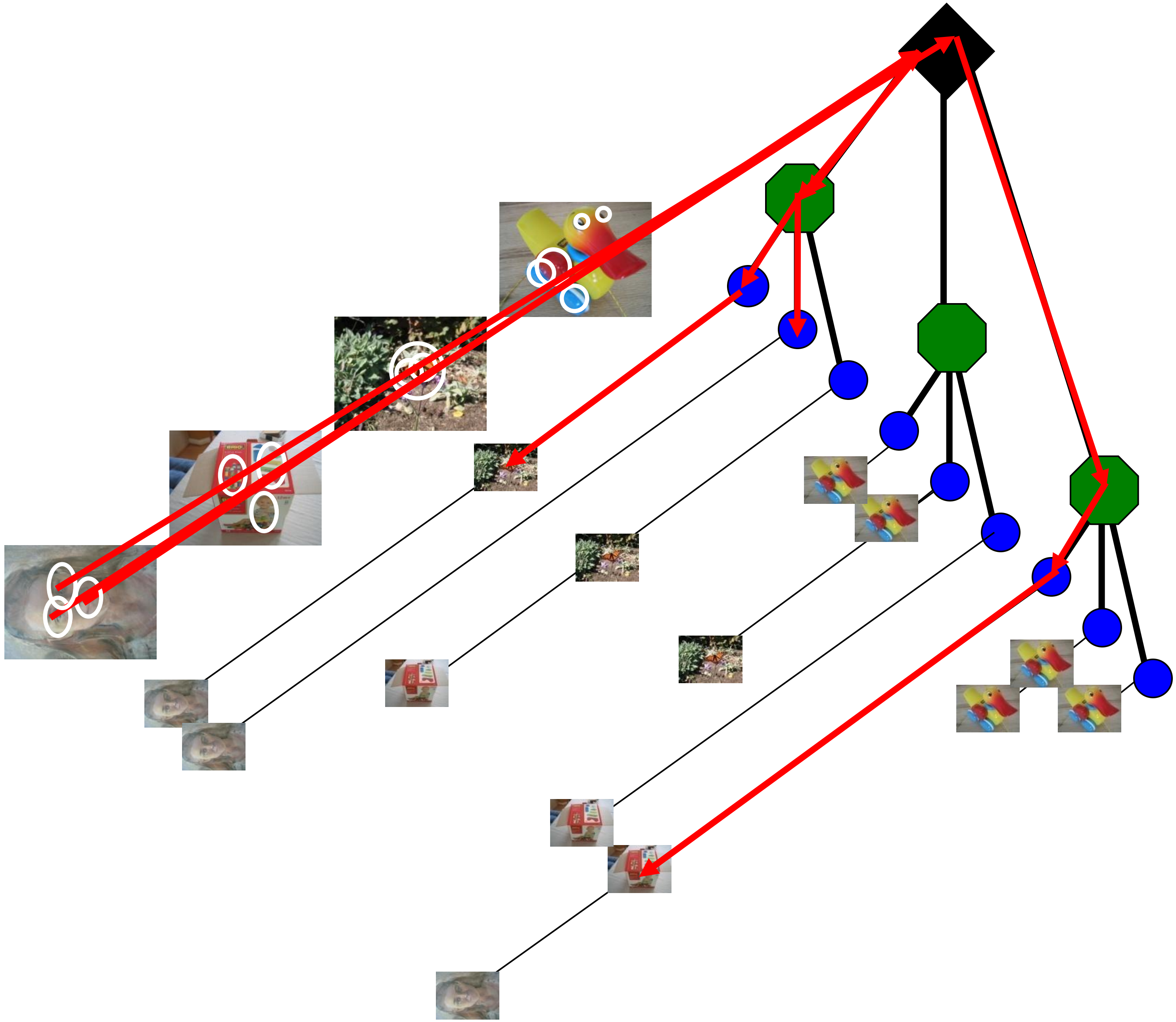


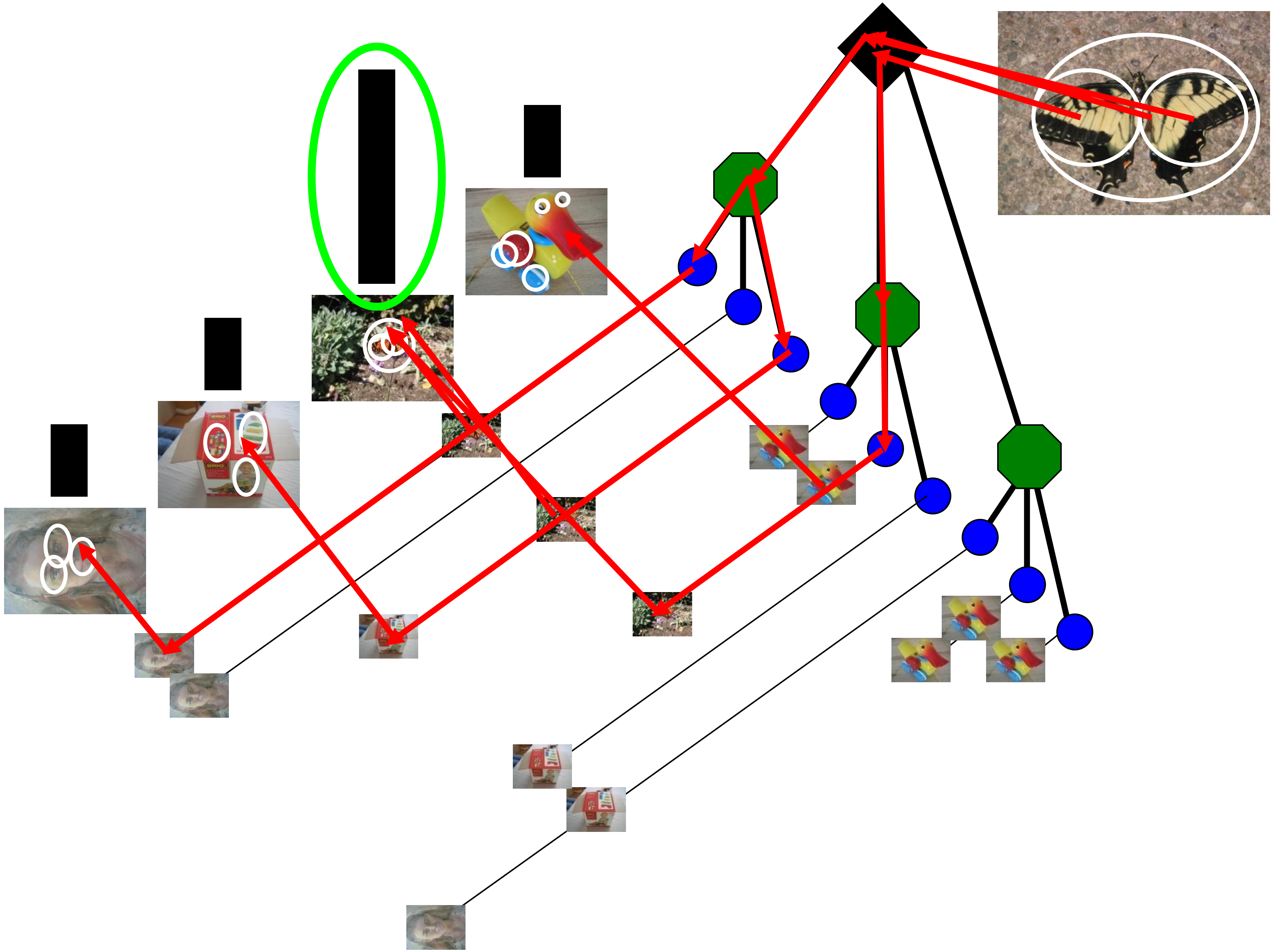












Vocabulary trees: complexity

Number of words given tree parameters:
branching factor and number of levels

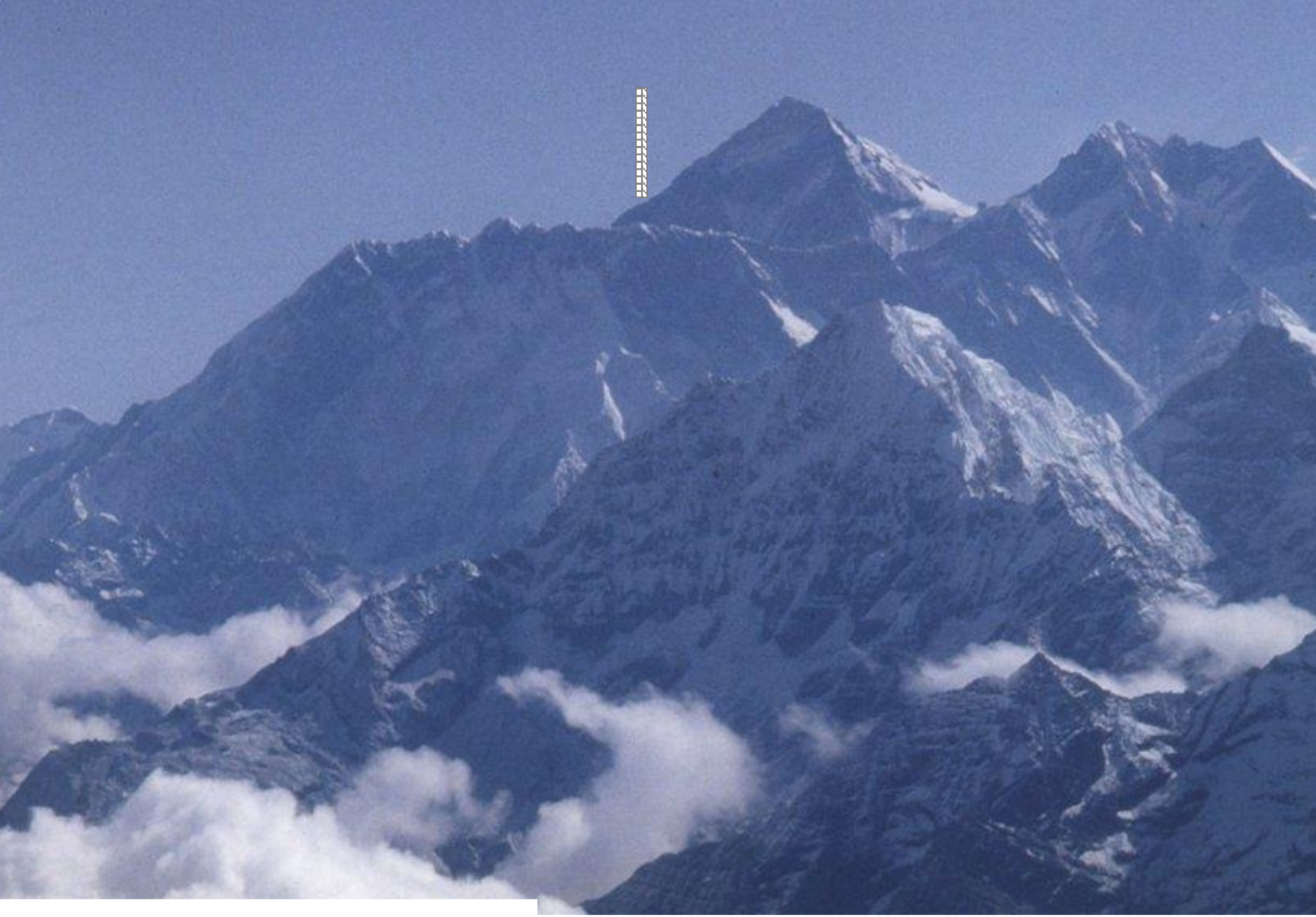
Word assignment cost vs. flat vocabulary

110,000,000 Images in 5.8 Seconds

Slide Credit: Nister

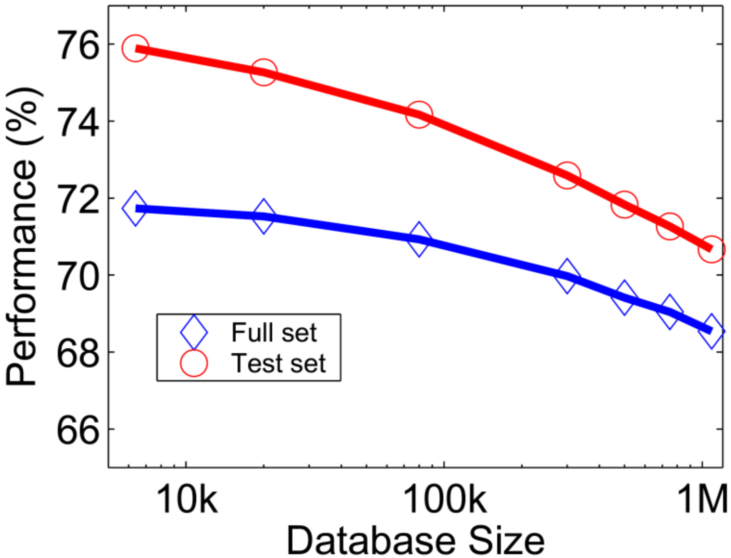
Slide Credit: Nister

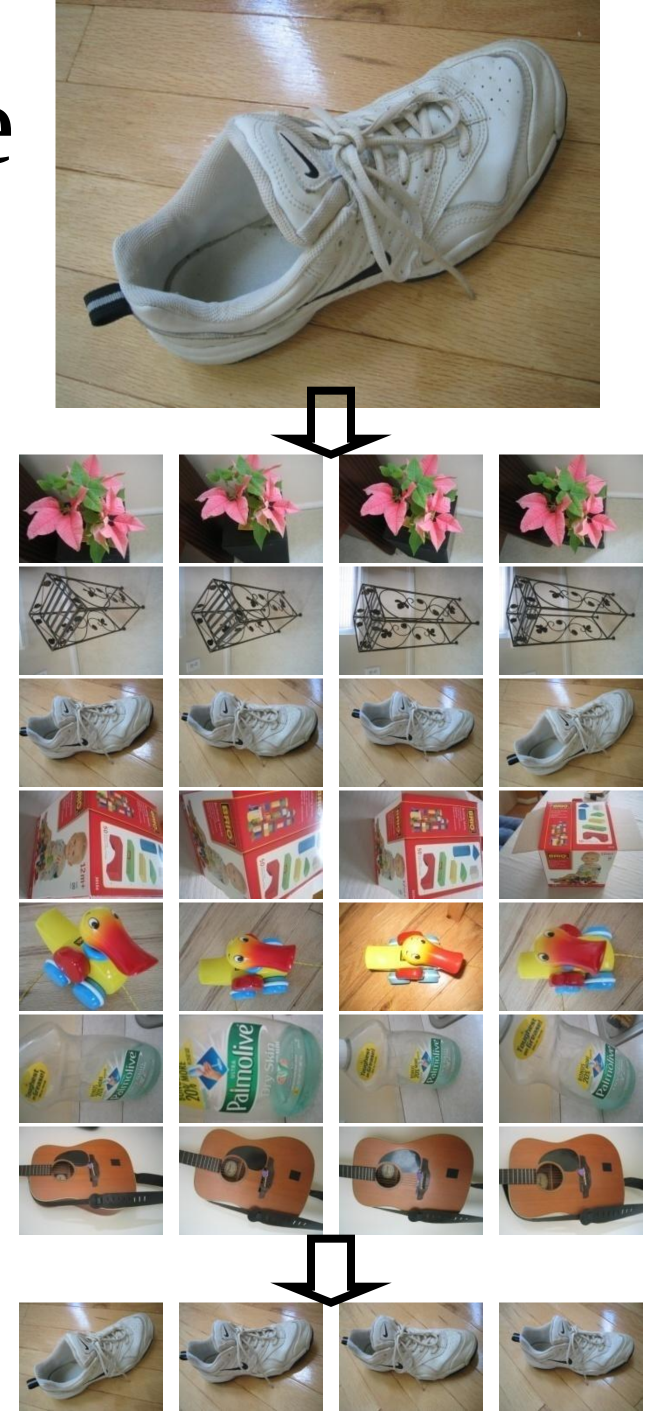
Slide Credit: Nister



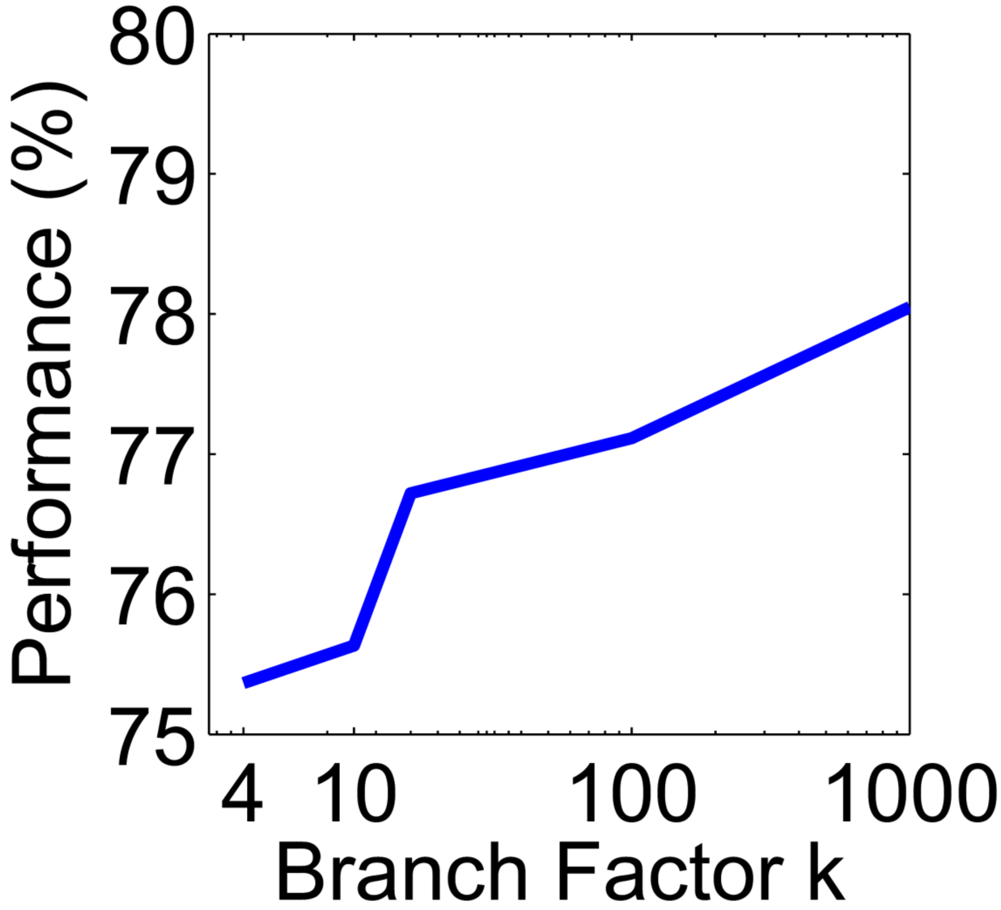
Slide Credit: Nister

Performance



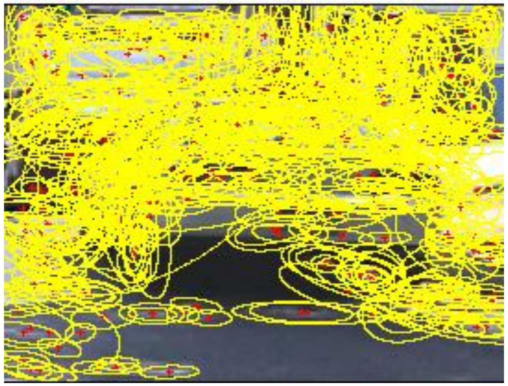


Higher branch factor works better (but slower)

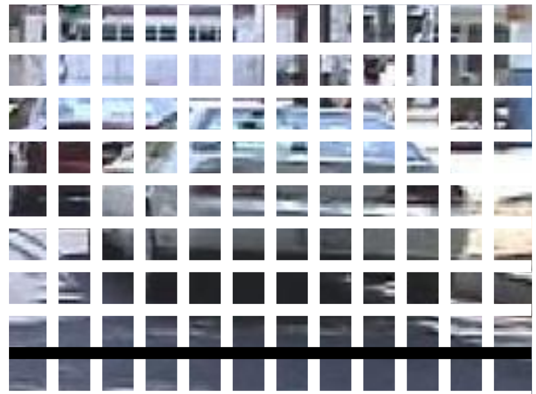


Center for Visualization & Virtual Environments

Sampling strategies

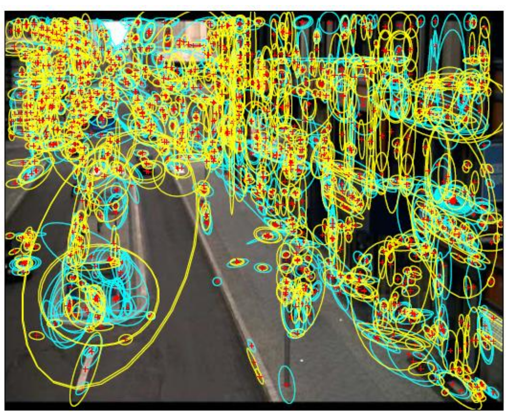


Sparse, at interest points



Dense, uniformly

Randomly



Multiple interest operators

- To find specific, textured objects, sparse sampling from interest points often more reliable.
- Multiple complementary interest operators offer more image coverage.
- For object categorization, dense sampling offers better coverage.

[See Nowak, Jurie & Triggs, ECCV 2006]

Image credits: F-F. Li, E. Nowak, J. Sivic

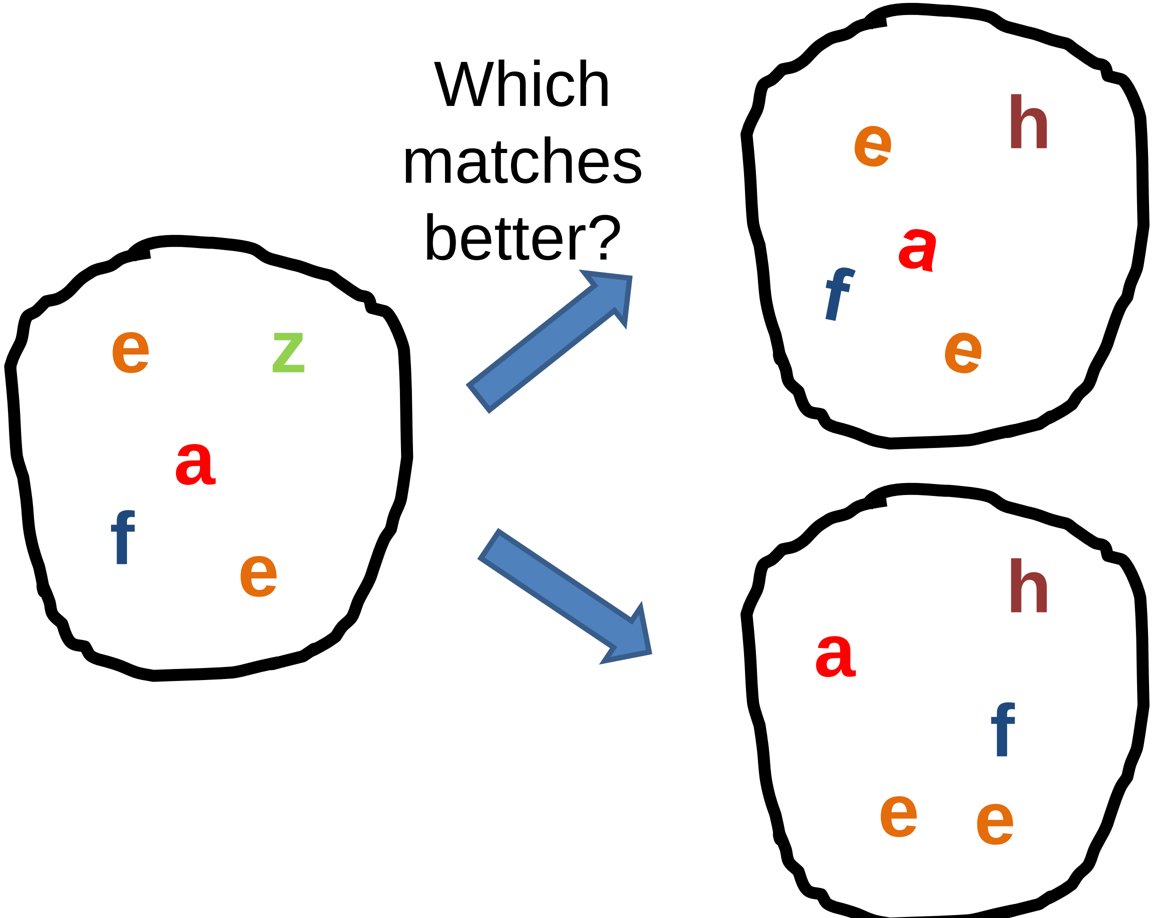
Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

Kristen Grauman

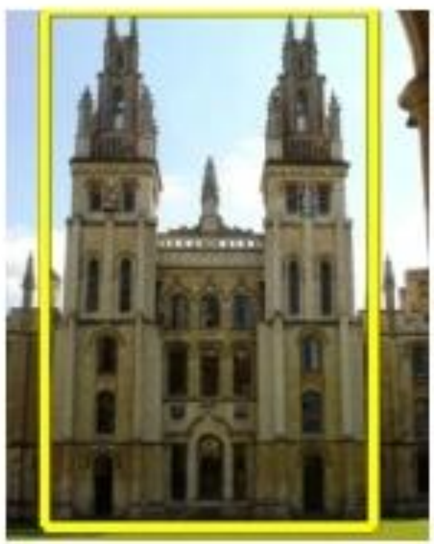
Can we be more accurate?

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



Can we be more accurate?

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



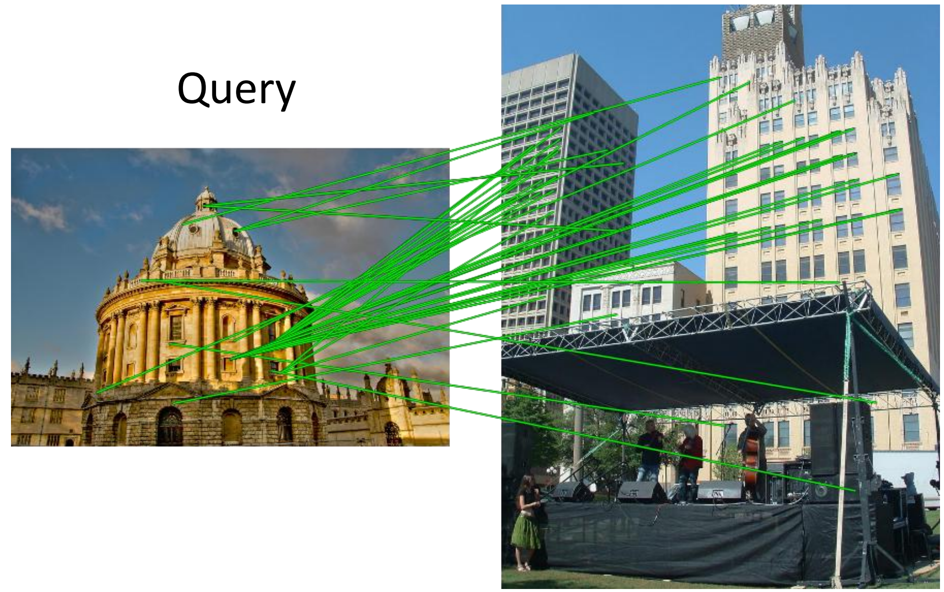

Real objects have consistent geometry

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.

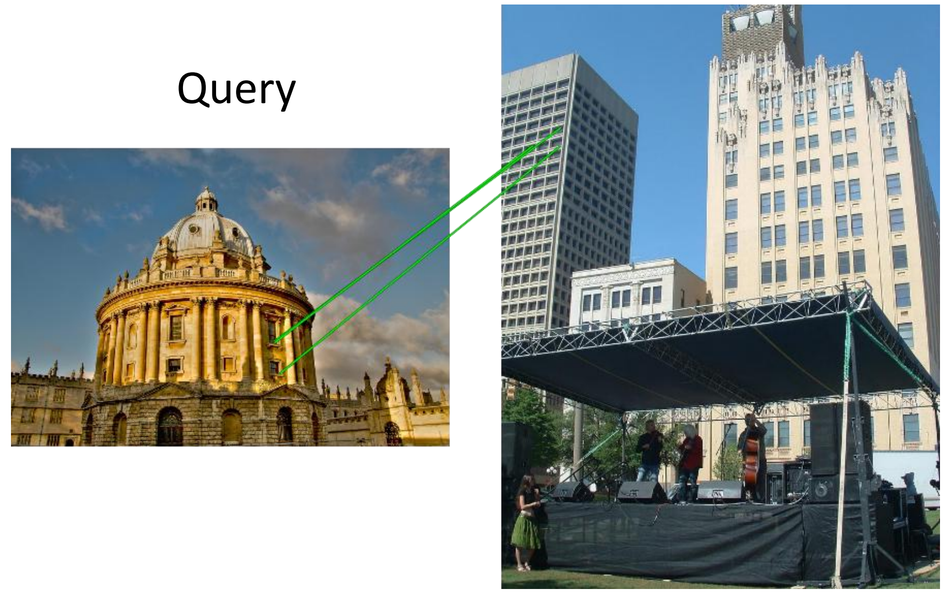
Slide credit: Ondrej Chum

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

Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

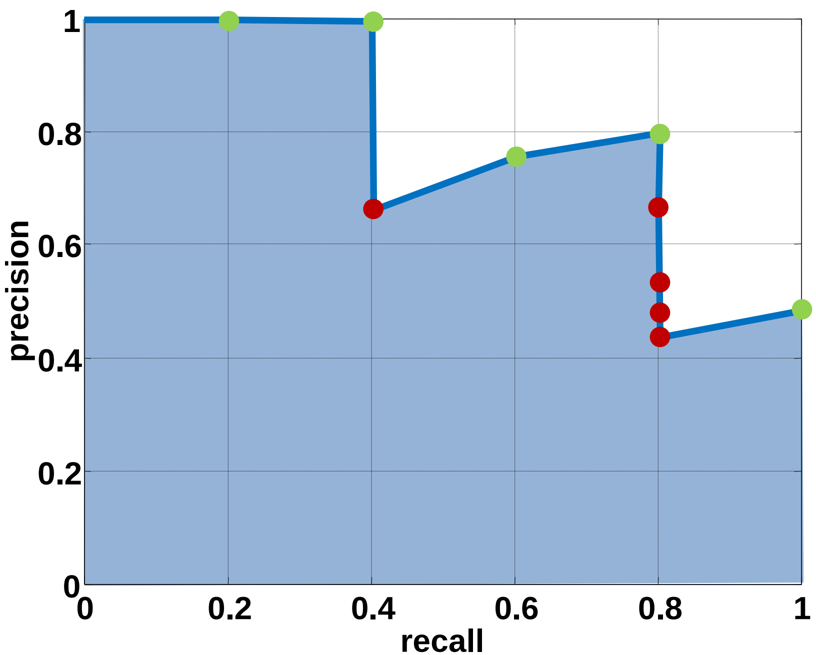
Kristen Grauman

Scoring retrieval quality

Query

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

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



Results (ordered):



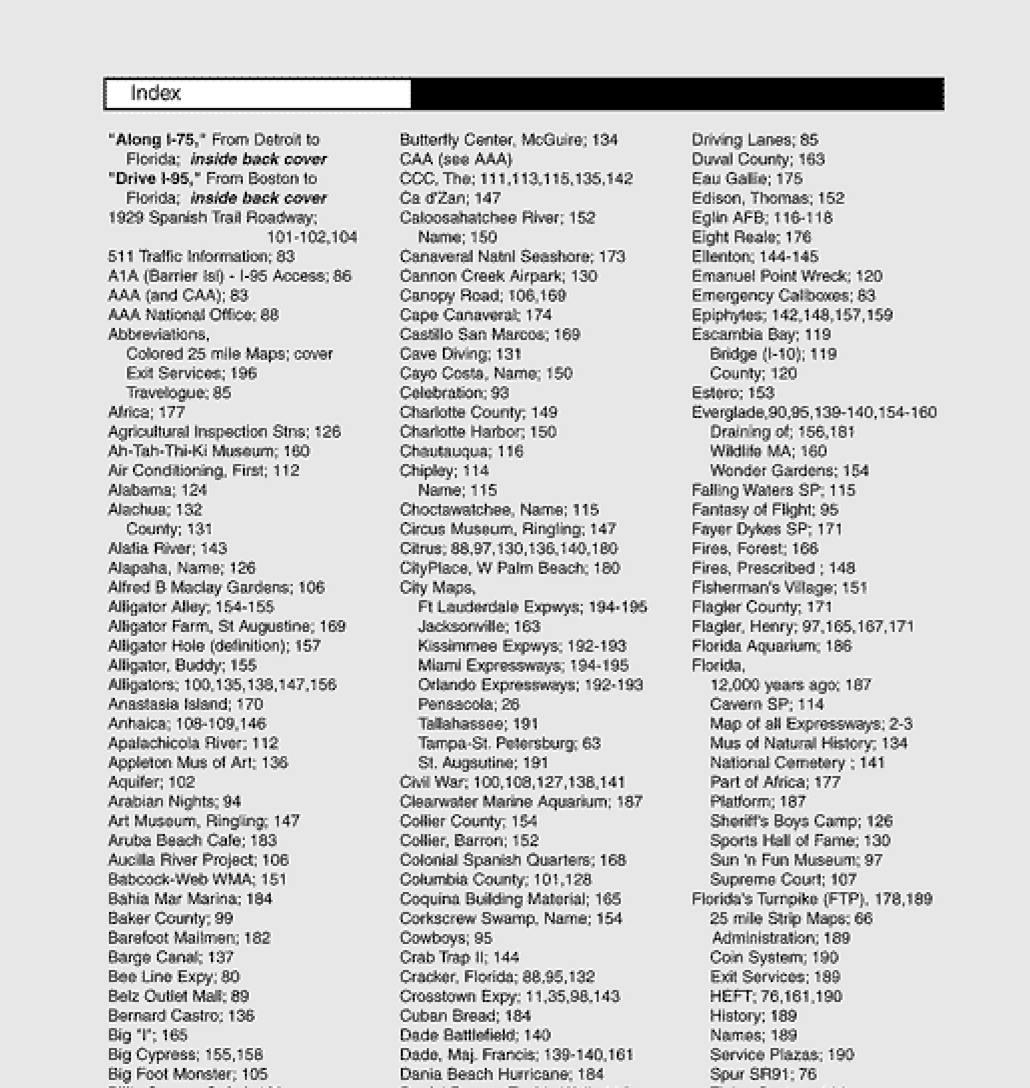







Slide credit: Ondrej Chum

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

Query: ***golf green***

Results:

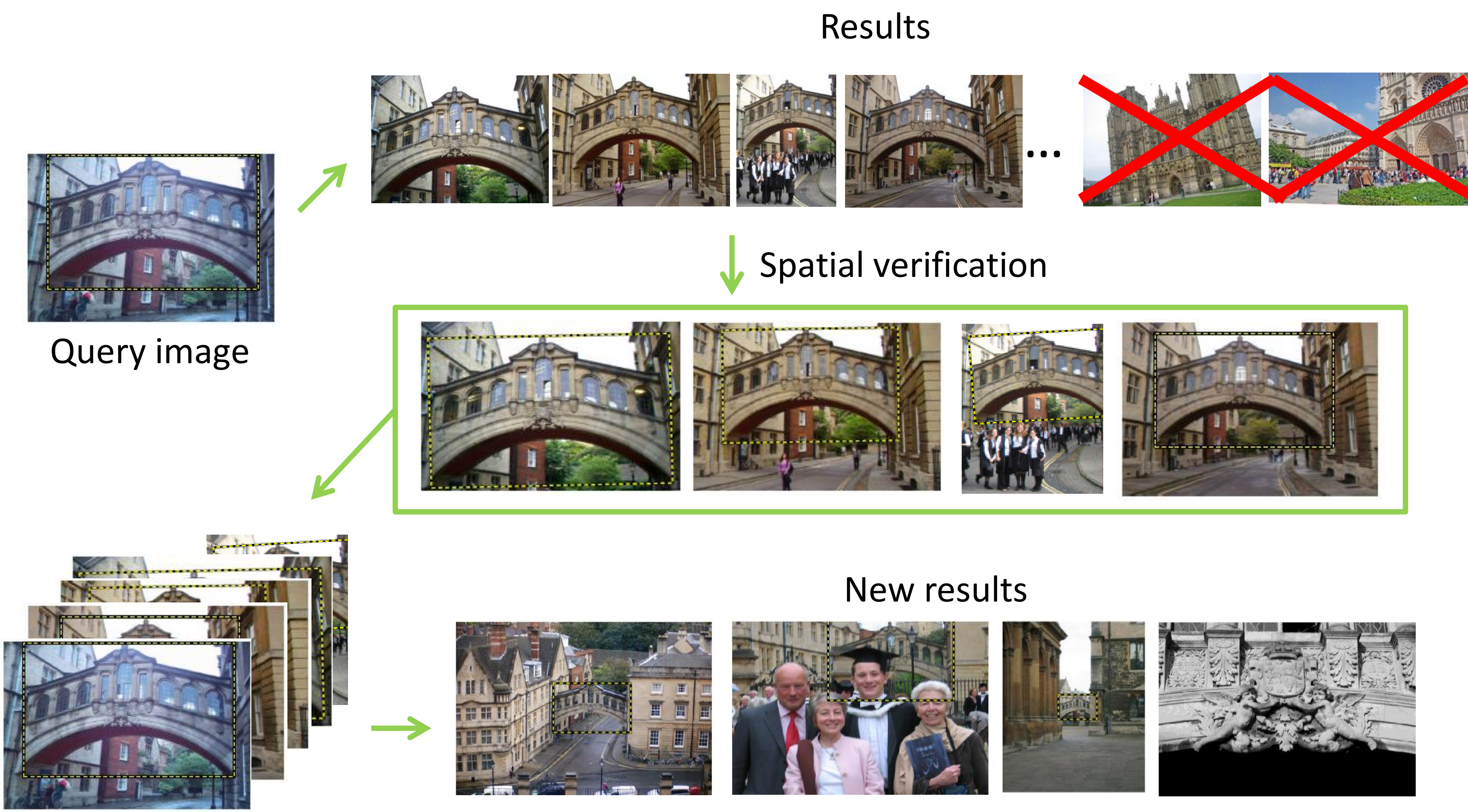
- How can the grass on the ***greens*** at a ***golf*** course be so perfect?
- For example, a skilled ***golf****er* expects to reach the ***green*** on a par-four hole in **...**
- Manufactures and sells synthetic ***golf*** putting ***green****s* and mats.

Irrelevant result can cause a `topic drift’:

- Volkswagen ***Golf***, 1999, ***Green***, 2000cc, petrol, manual, , hatchback, 94000miles, 2.0 GTi, 2 Registered Keepers, HPI Checked, Air-Conditioning, Front and Rear Parking Sensors, ABS, Alarm, Alloy

Slide credit: Ondrej Chum

Query Expansion

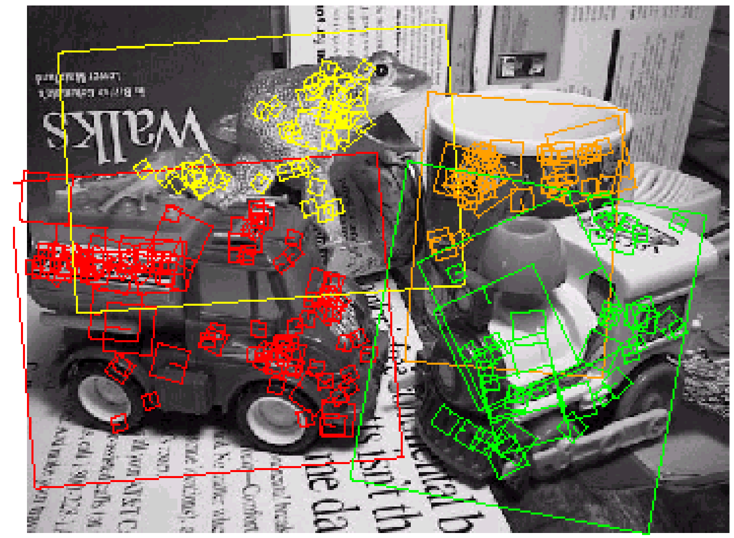


Chum, Philbin, Sivic, Isard, Zisserman: Total Recall..., ICCV 2007

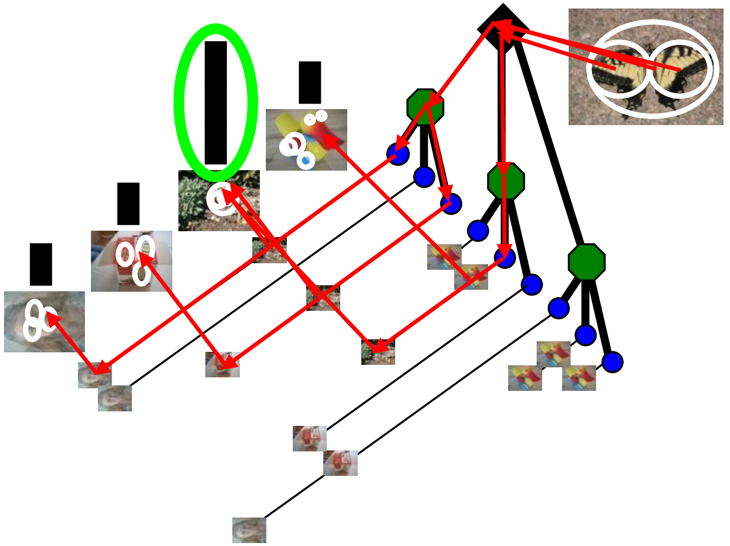
Slide credit: Ondrej Chum

Things to remember

- Object instance recognition
  - Find keypoints, compute descriptors
  - Match descriptors
  - Vote for / fit affine parameters
  - Return object if # inliers > T



- Keys to efficiency
  - Visual words
    - Used for many applications
  - Inverse document file
    - Used for web-scale search



Next Lecture

- Object category detection with sliding windows