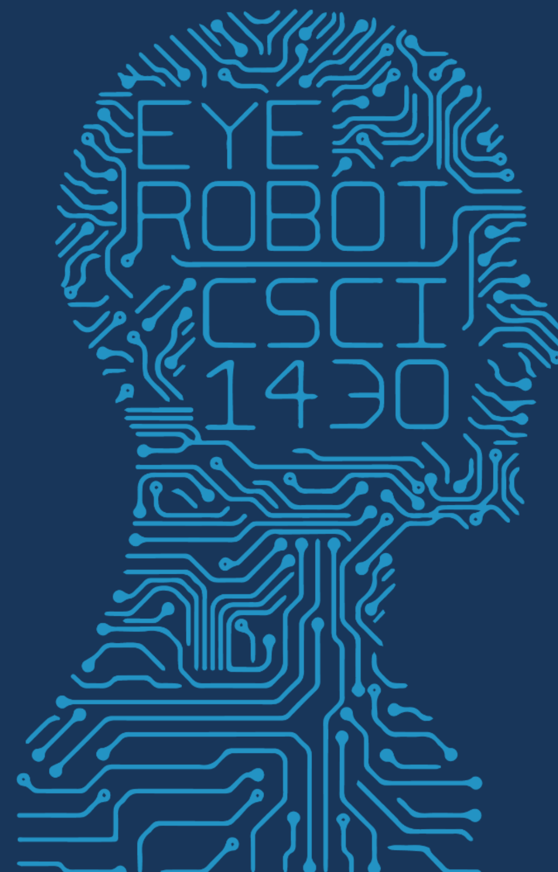
1950

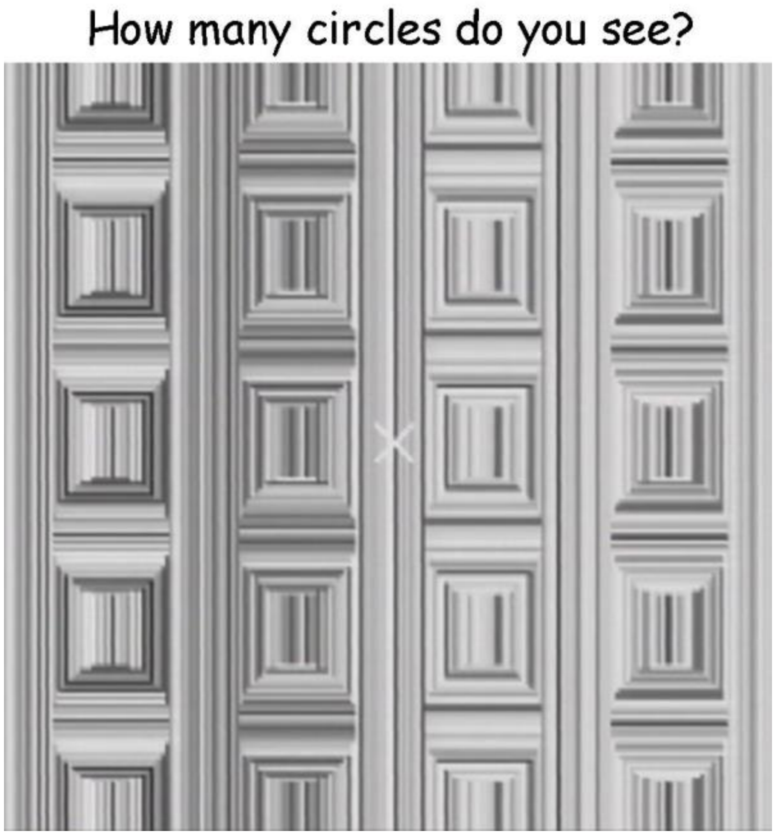
Future Vision



2020

Computer Vision

Coffer Illusion



Coffer Illusion

Machine Learning Problems

	Supervised Learning	*Unsupervised Learning*
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

Supervised learning

$$\mathbf{f(x) = y}$$

Prediction function   Image feature   Output (label)

Training: Given a *training set* of labeled examples:

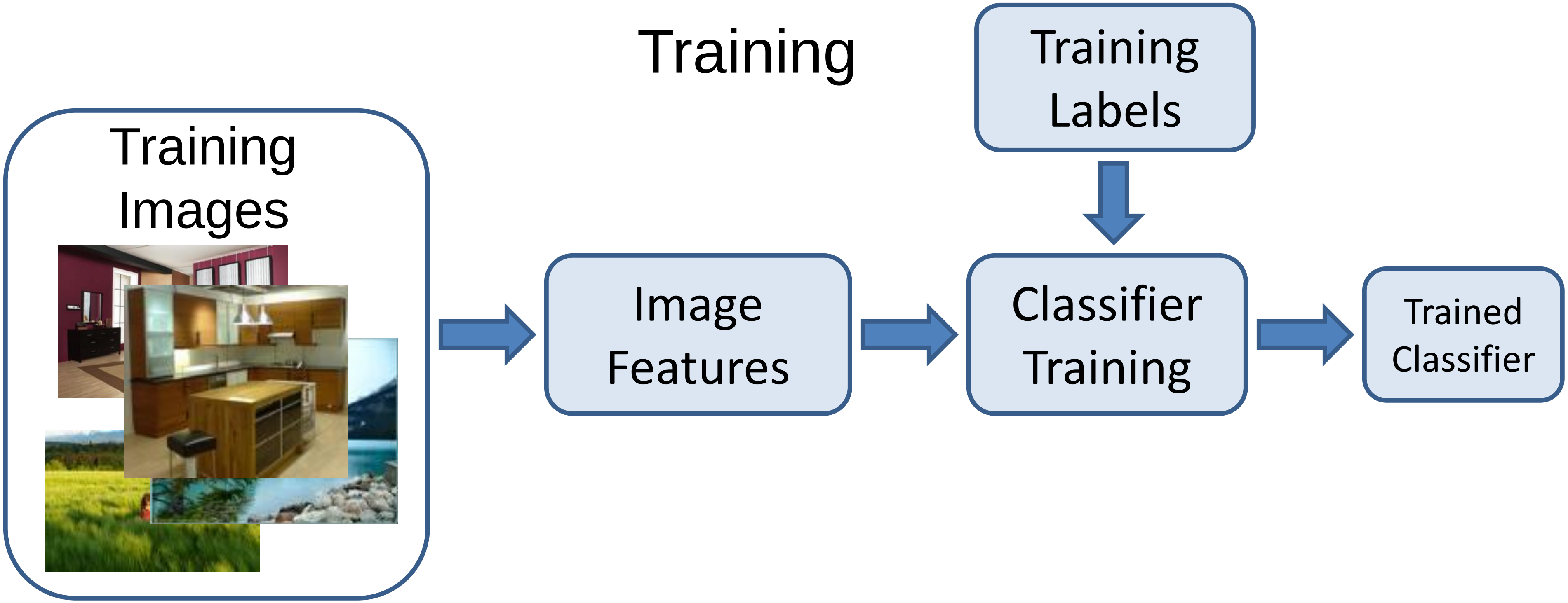
$$\{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_N,y_N)\}$$

Estimate the prediction function f by minimizing the prediction error on the training set.

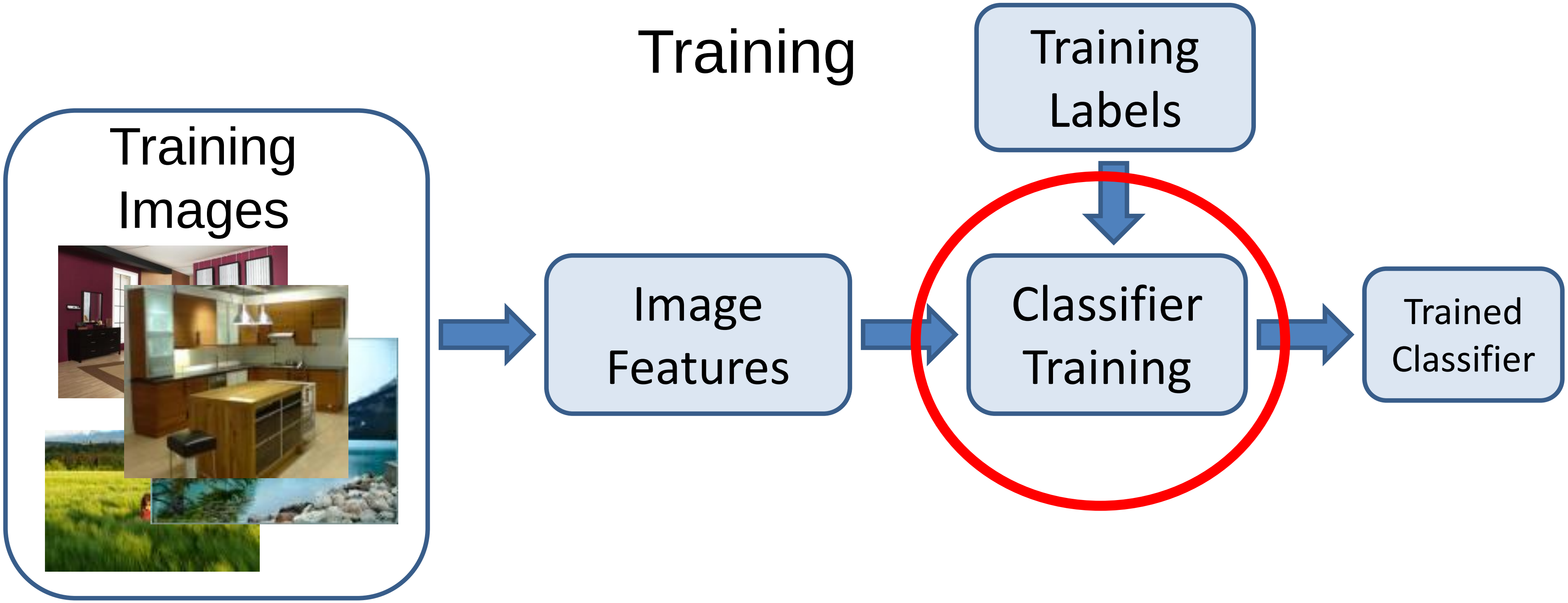
Testing: Apply f to a unseen *test example* $\mathbf{x}$ and output the predicted value $y = f(\mathbf{x})$ to *classify* $\mathbf{x}$.

Slide credit: L. Lazebnik

Image Categorization



Classifiers



Learning a classifier

Given a set of features with corresponding labels, learn a function to predict the labels from the features.

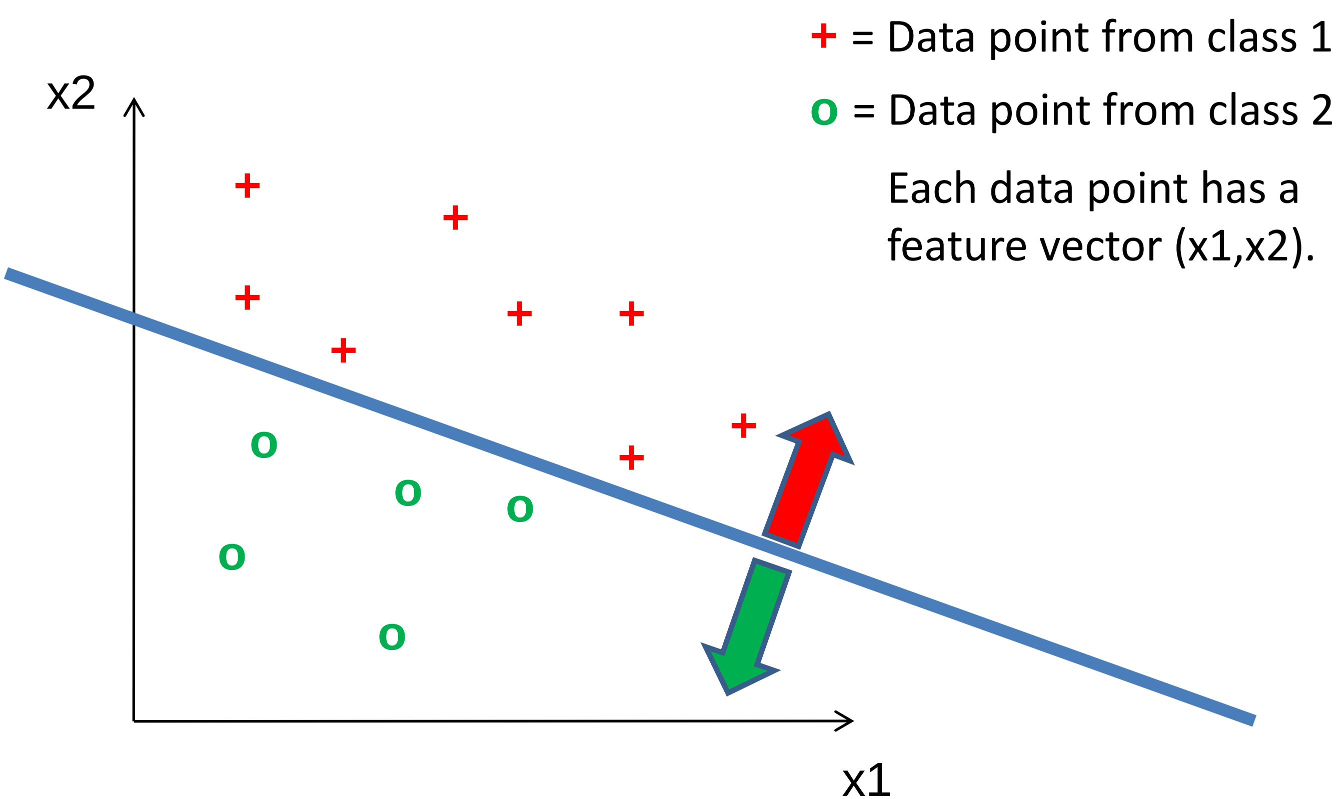
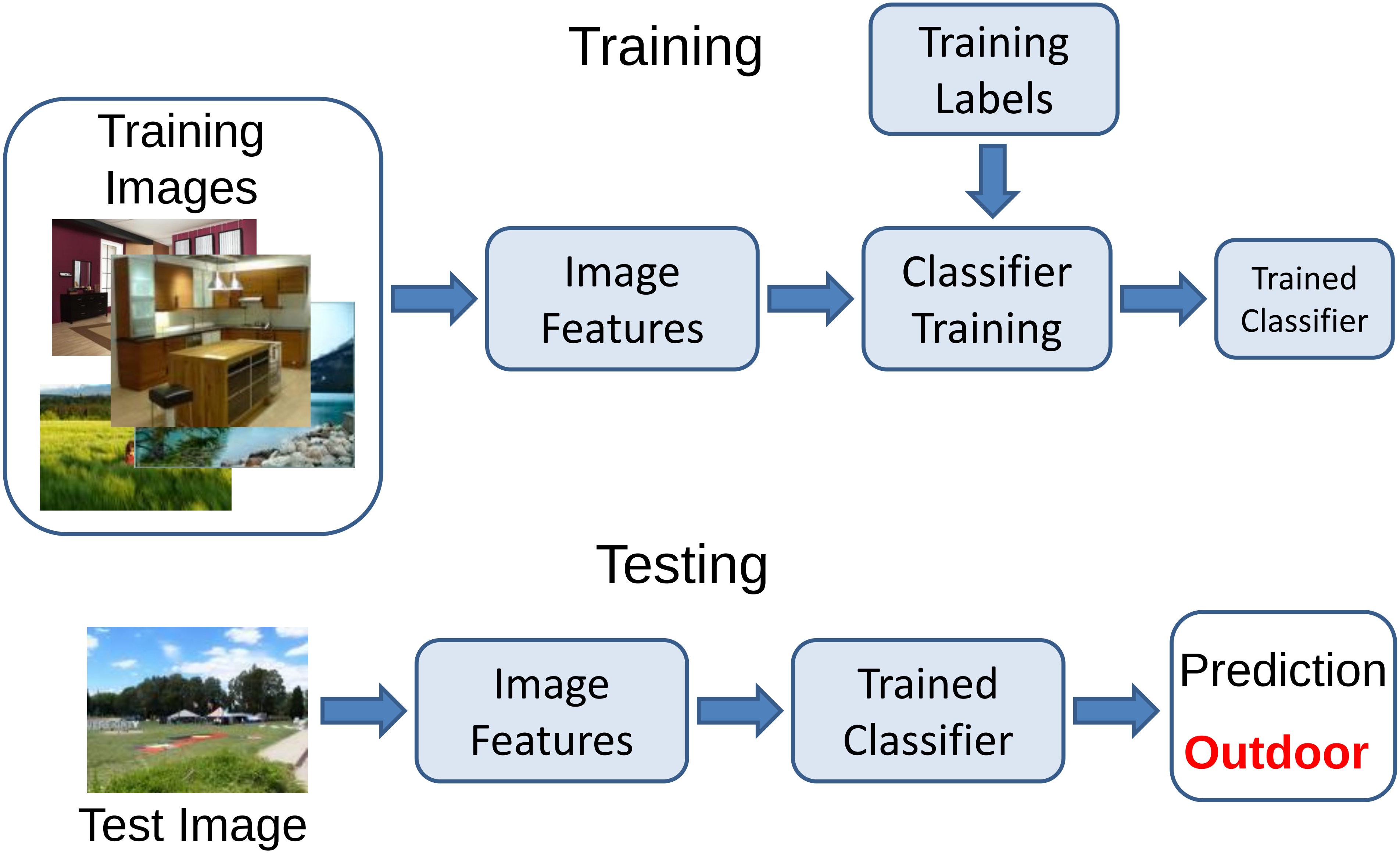


Image Categorization



Example: Scene Categorization

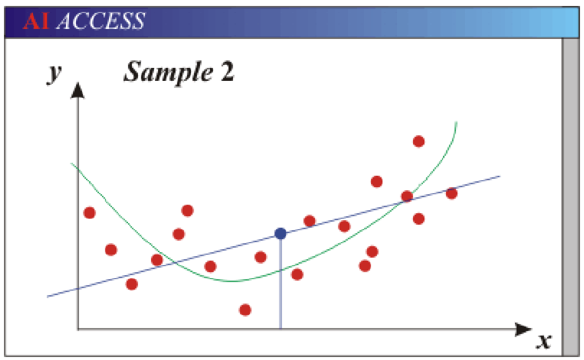
- Is this a kitchen?



Bias-Variance Trade-off

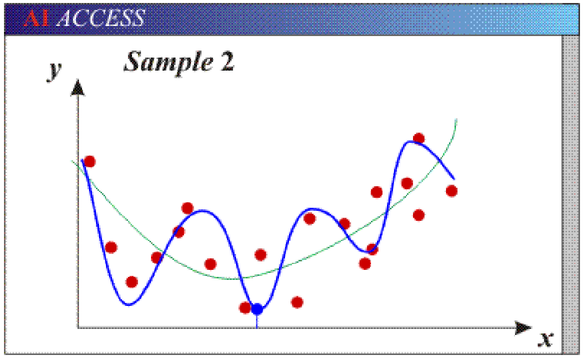
Bias: *error in model assumptions*; how much the average model over all training sets differs from the true model.

Variance: how much models estimated from different training sets differ from each other.



Models with too few parameters are inaccurate because of a large bias.

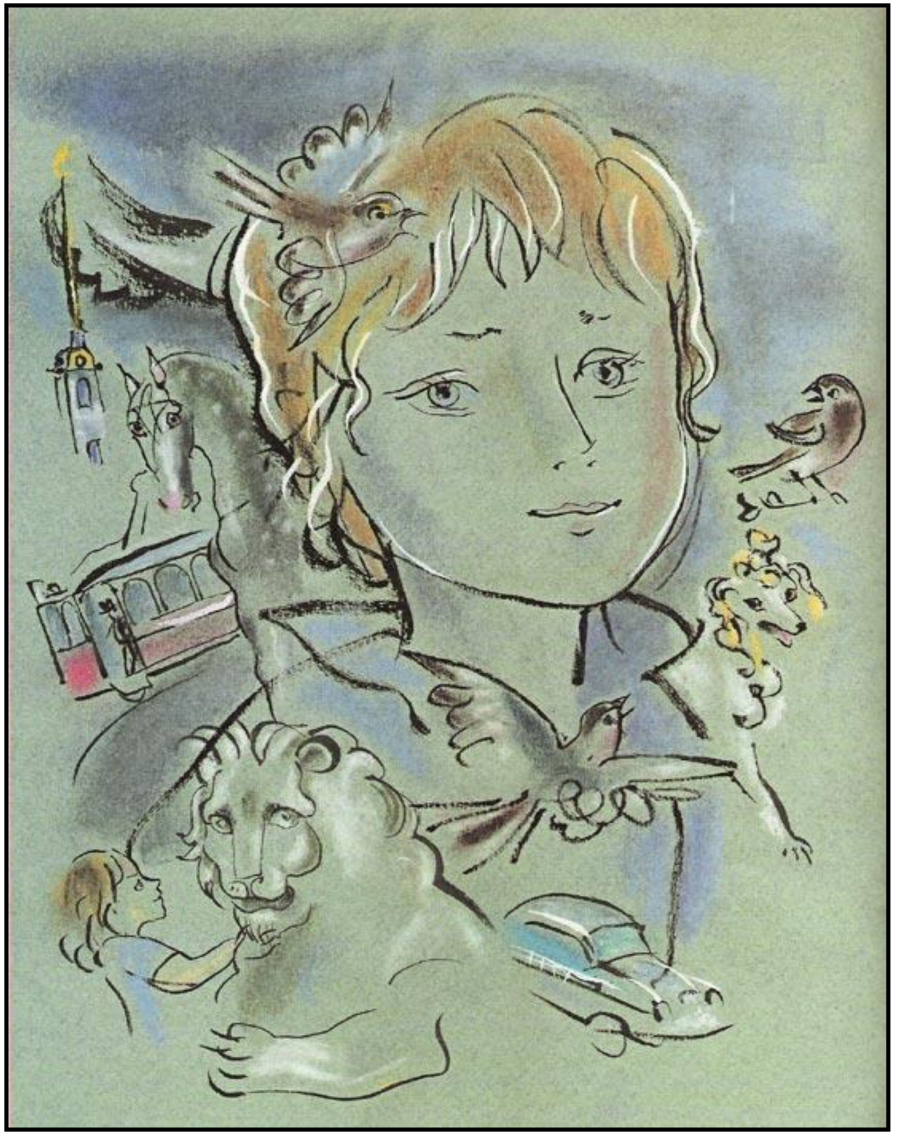
- Not enough flexibility!



Models with too many parameters are inaccurate because of a large variance.

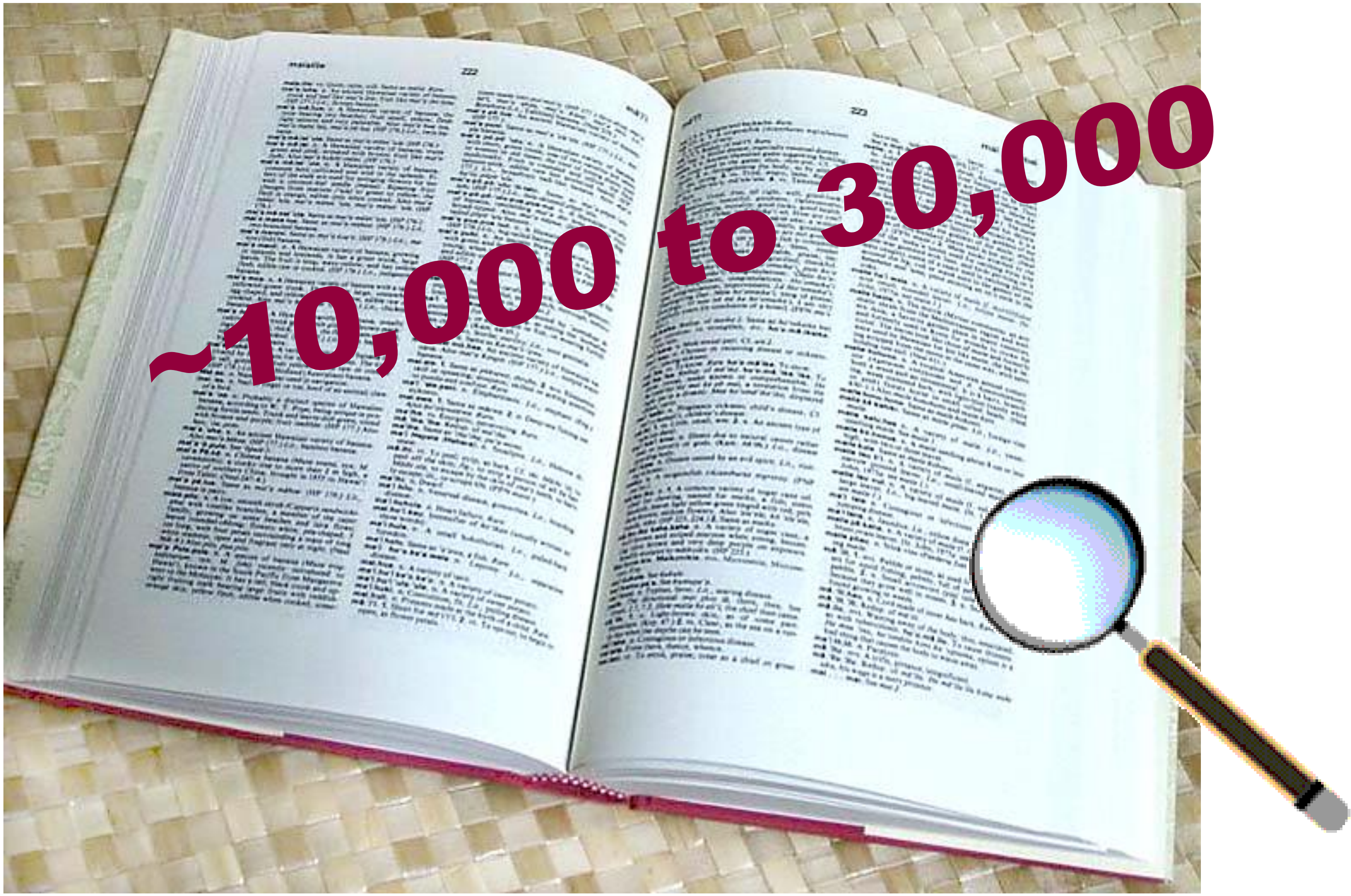
- Too much sensitivity to the sample.

Recognition: Overview and History



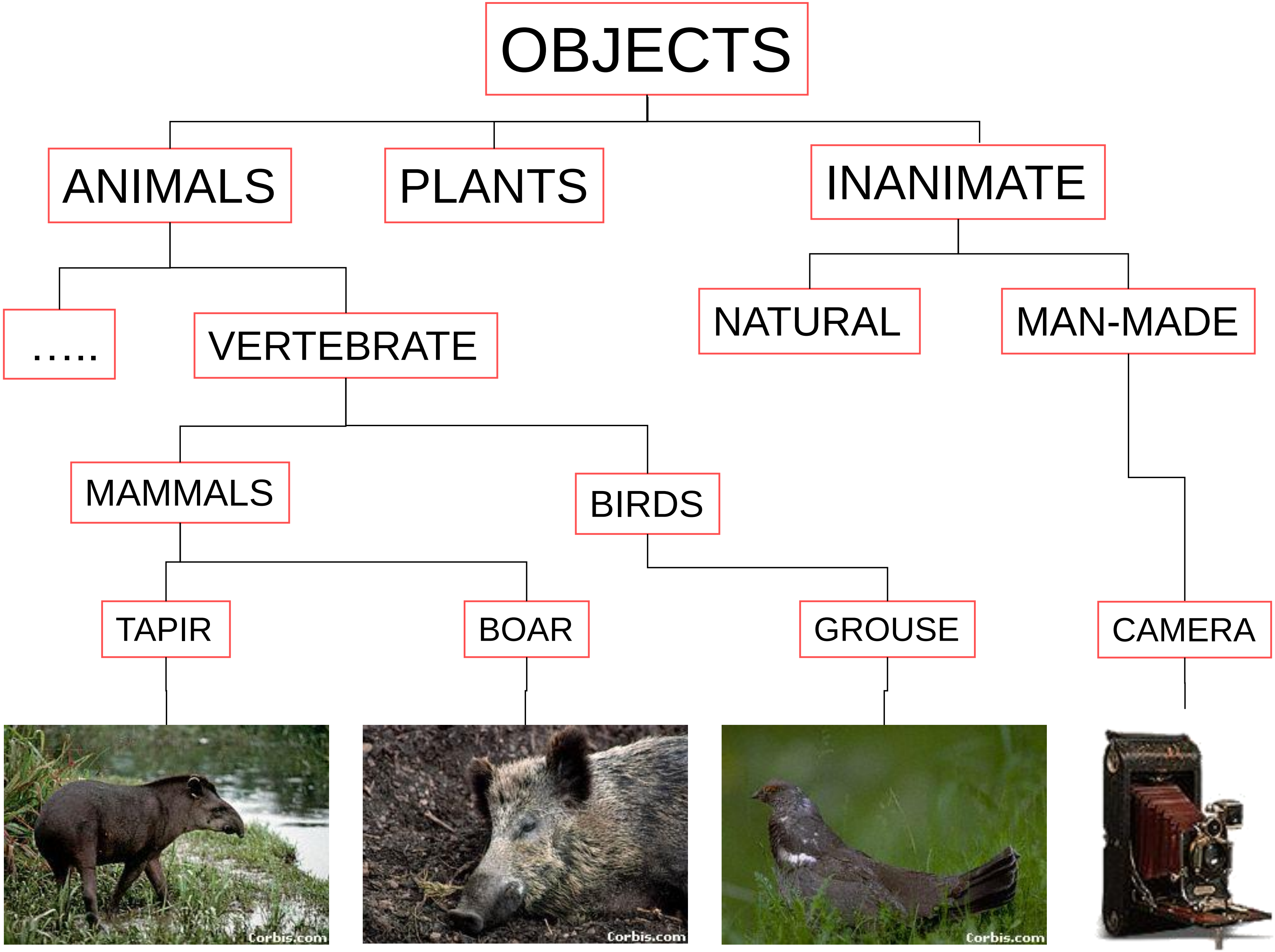
Slides from James Hays, Lana Lazebnik, Fei-Fei Li, Rob Fergus, Antonio Torralba, and Jean Ponce

How many visual object categories are there?



Biederman 1987


~10,000 to 30,000


OBJECTS
ANIMALS
PLANTS
INANIMATE
.....
VERTEBRATE
NATURAL
MAN-MADE
MAMMALS
BIRDS
TAPIR
BOAR
GROUSE
CAMERA
Corbis.com
Corbis.com
Corbis.com

Specific recognition tasks

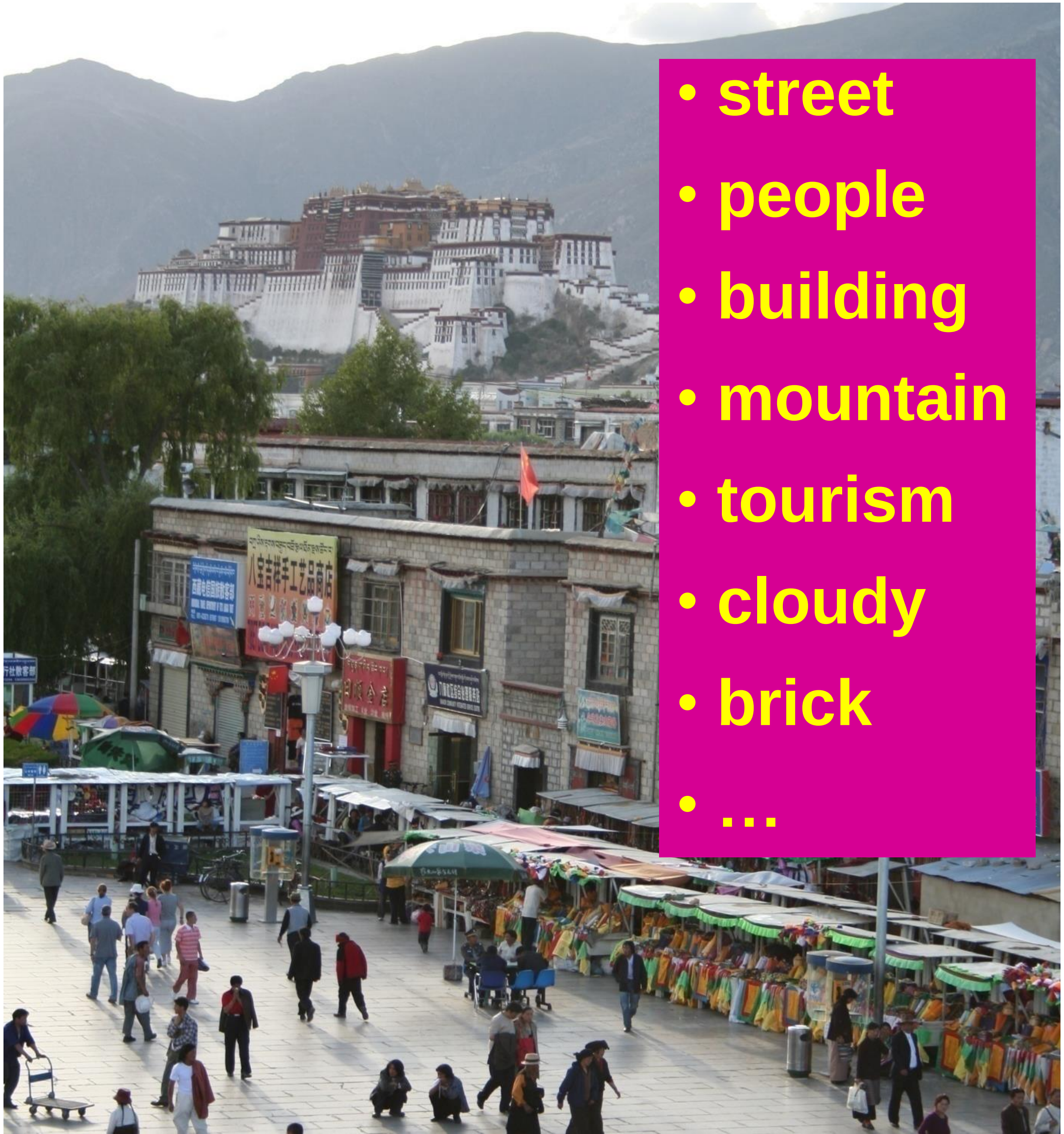
Svetlana Lazebnik

Scene categorization or classification

- **outdoor/indoor**
- **city/forest/factory/etc.**

Svetlana Lazebnik

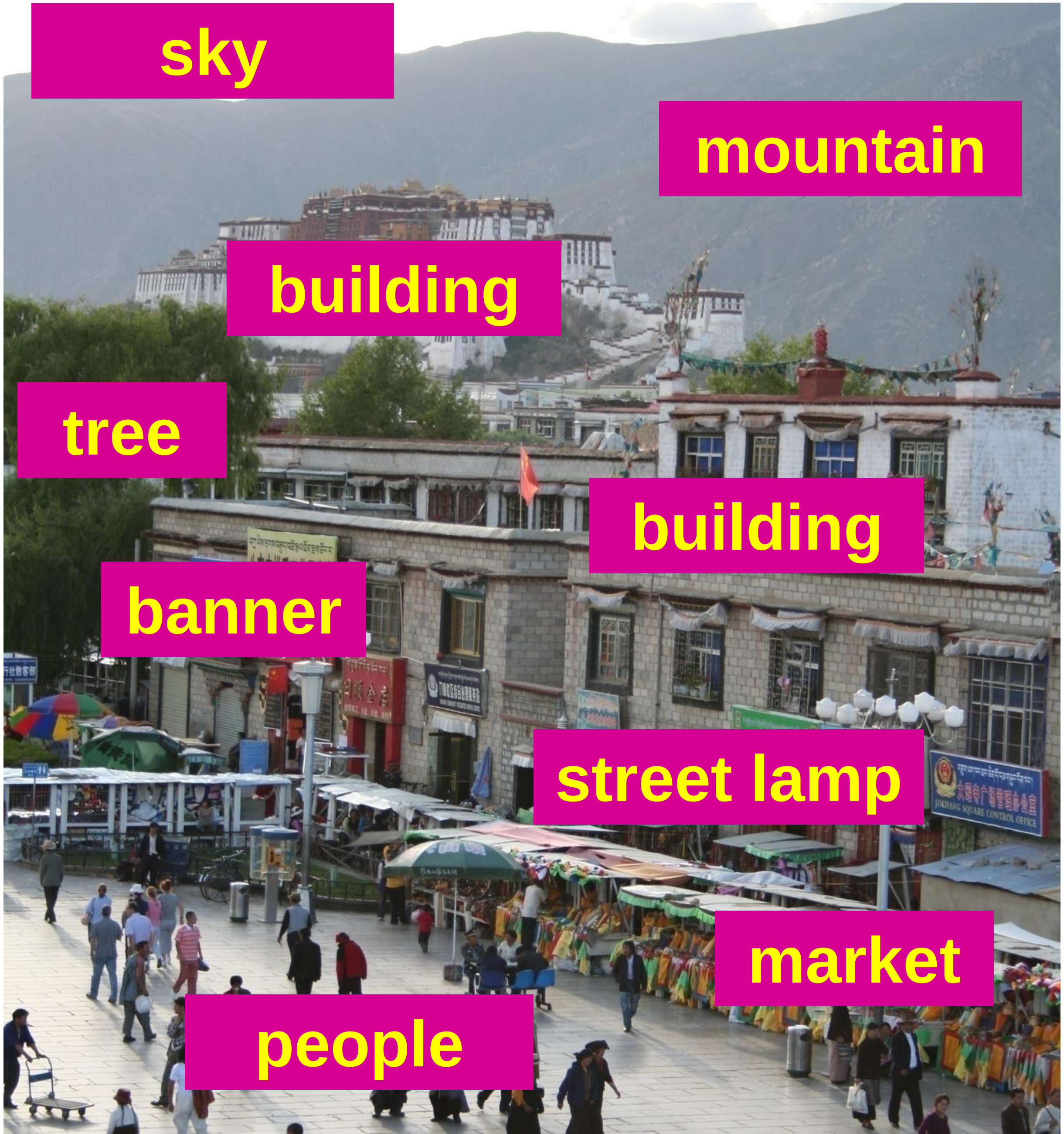
Image annotation / tagging / attributes



- **street**
- **people**
- **building**
- **mountain**
- **tourism**
- **cloudy**
- **brick**
- **…**

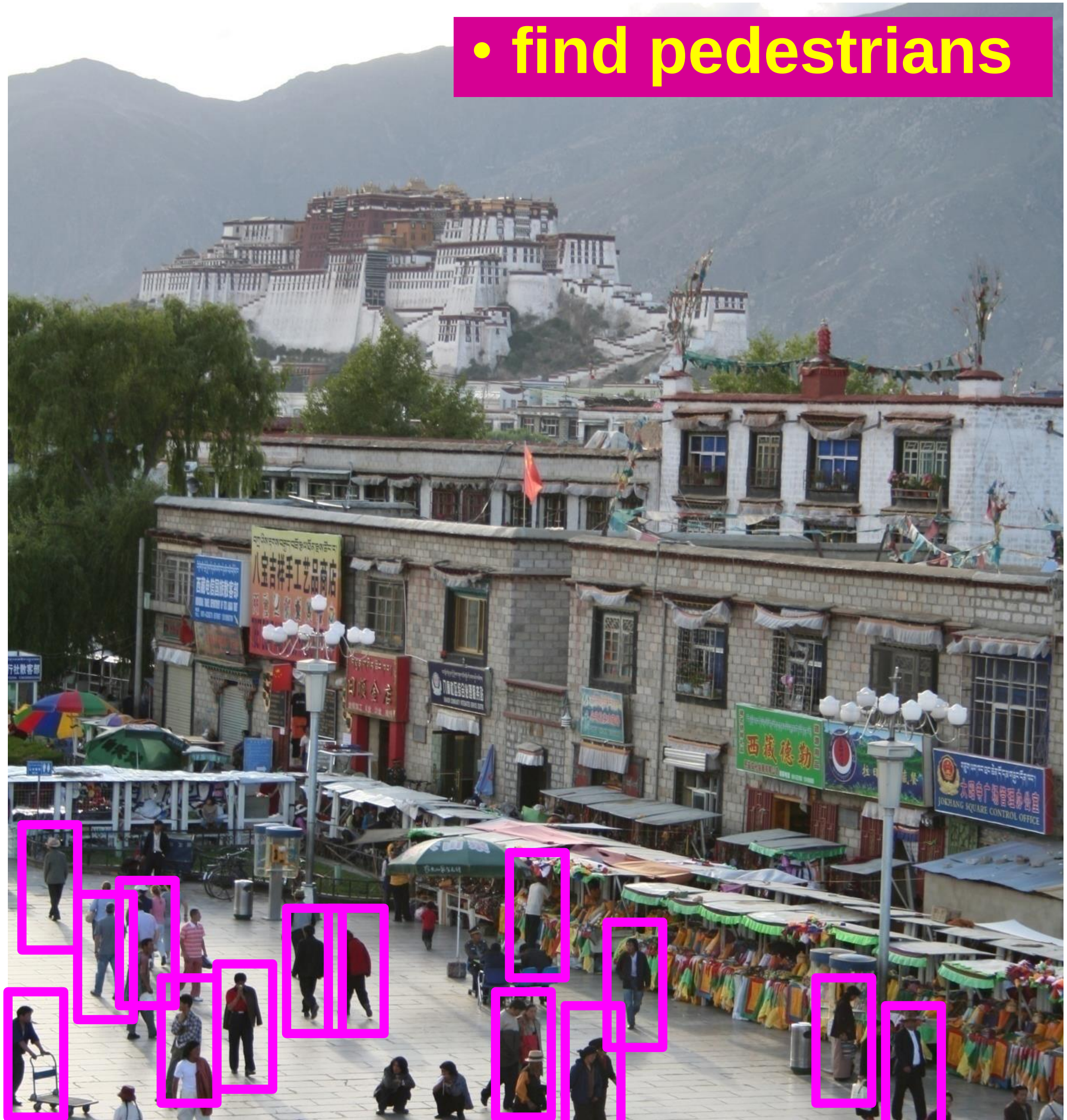
Svetlana Lazebnik

Image parsing / semantic segmentation



Svetlana Lazebnik

Object detection



Svetlana Lazebnik

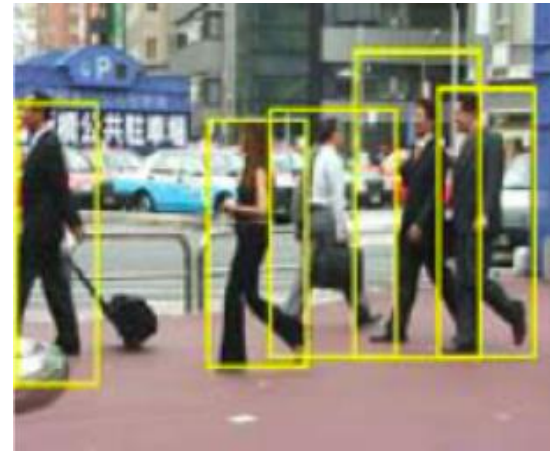
Scene understanding?

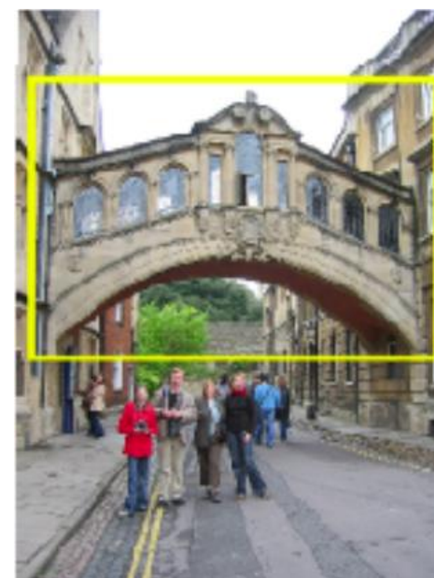
Svetlana Lazebnik

Category vs. instance recognition

Category:

- Find all the people
- Find all the buildings
- Often within a single image
- Often ‘sliding window’

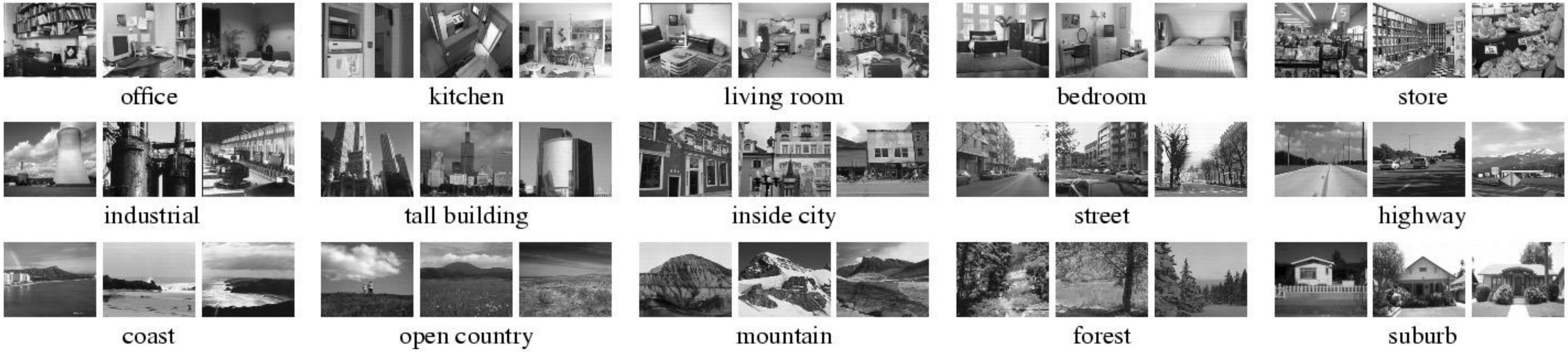




Instance:

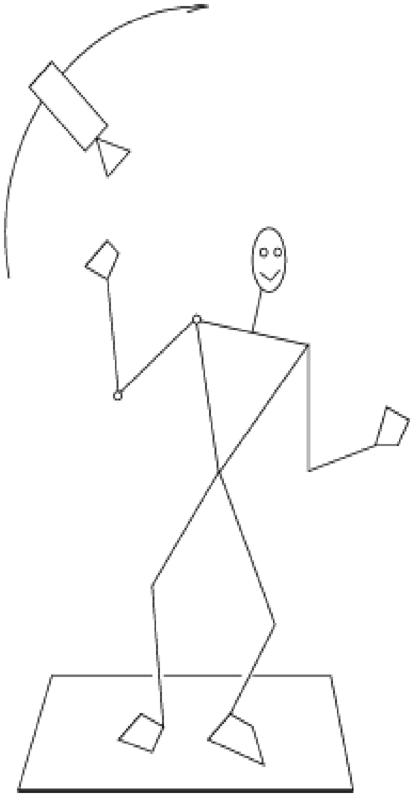
- Is this face James?
- Find this specific famous building
- Often within a database of images

Scene recognition dataset



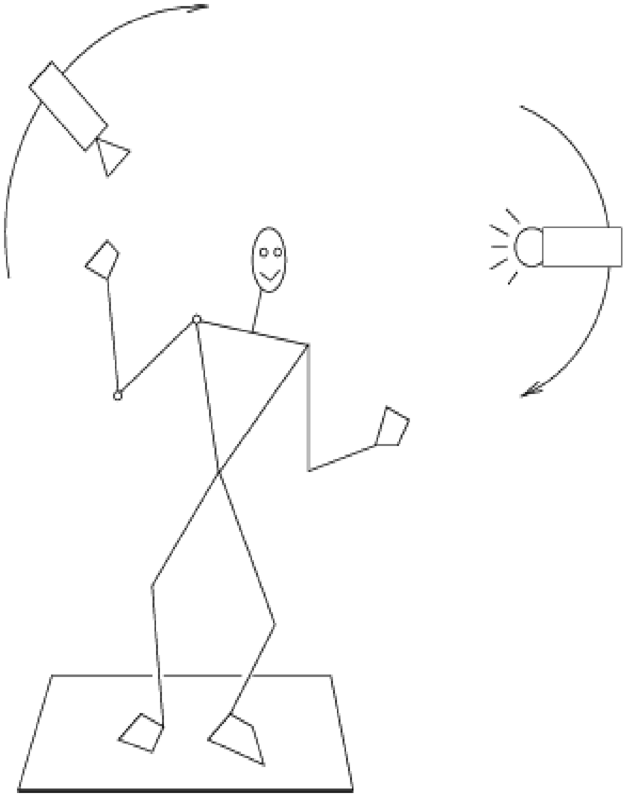
Instance or category?

Recognition is all about modeling variability



Variability: Camera position

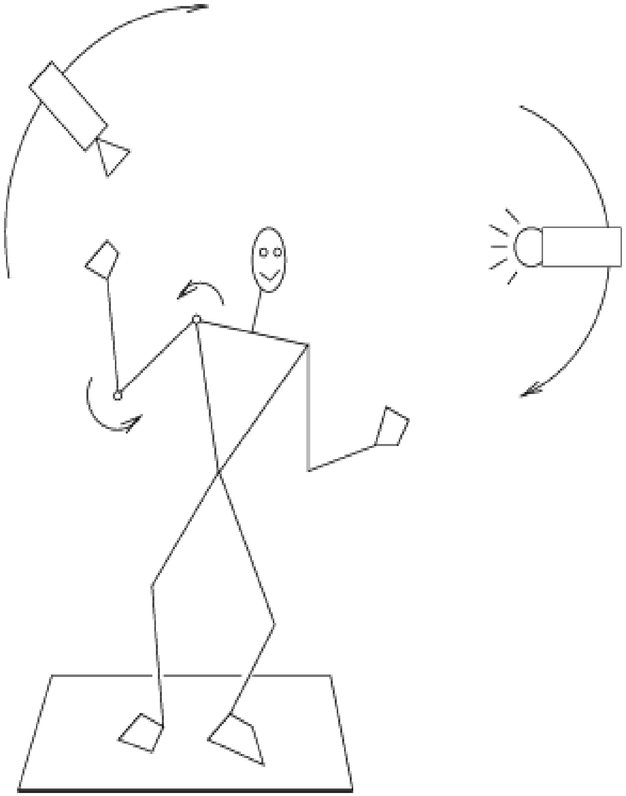
Recognition is all about modeling variability



Variability: Camera position
Illumination

Svetlana Lazebnik

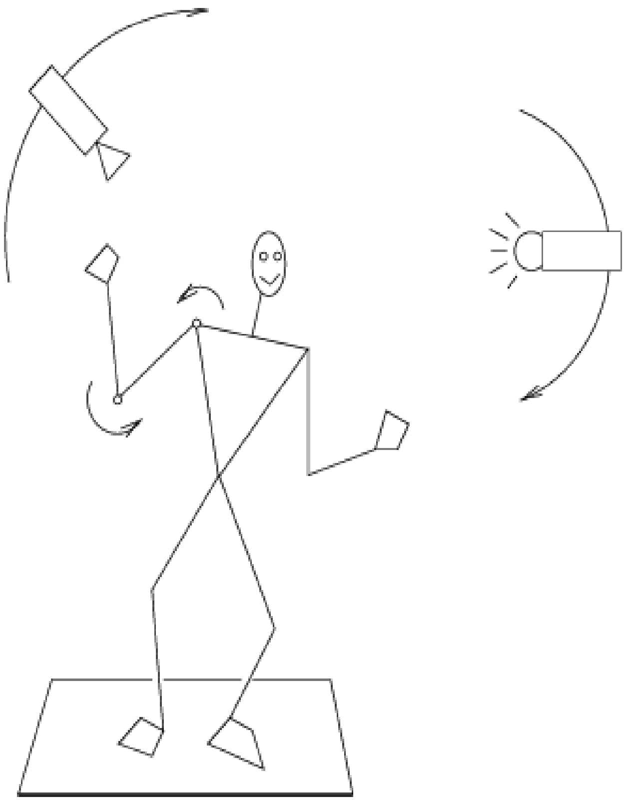
Recognition is all about modeling variability



Variability: Camera position
Illumination
Pose/shape parameters

Svetlana Lazebnik

Recognition is all about modeling variability



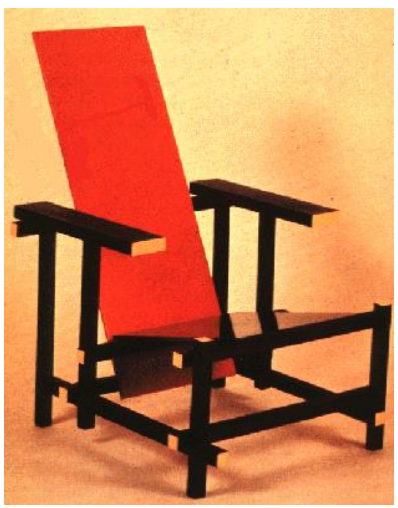
Variability: Camera position
Illumination
Pose/shape parameters

Within-class variations?

Svetlana Lazebnik

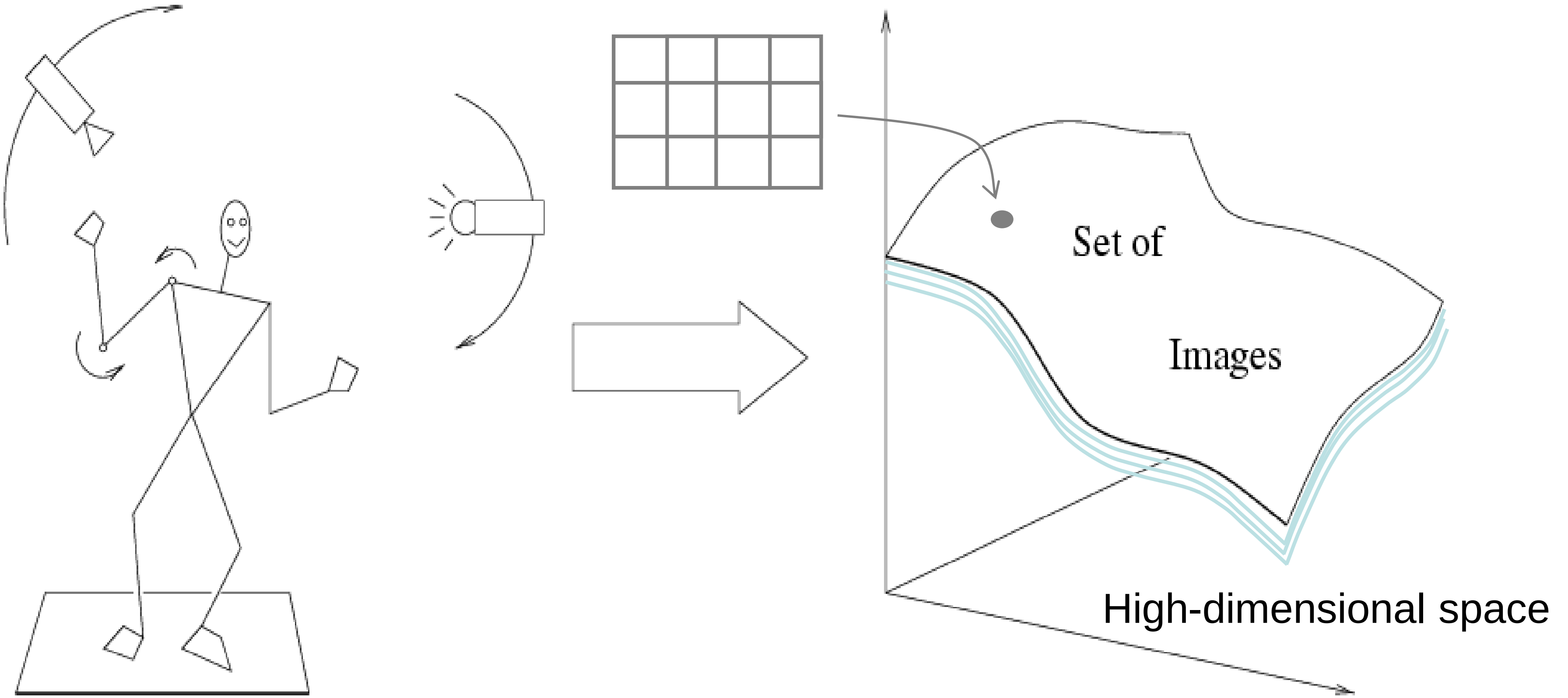
Within-class variations



Svetlana Lazebnik

Recognition is all about modeling variability



Variability: Camera position
Illumination
Pose/shape parameters
Within-class variation

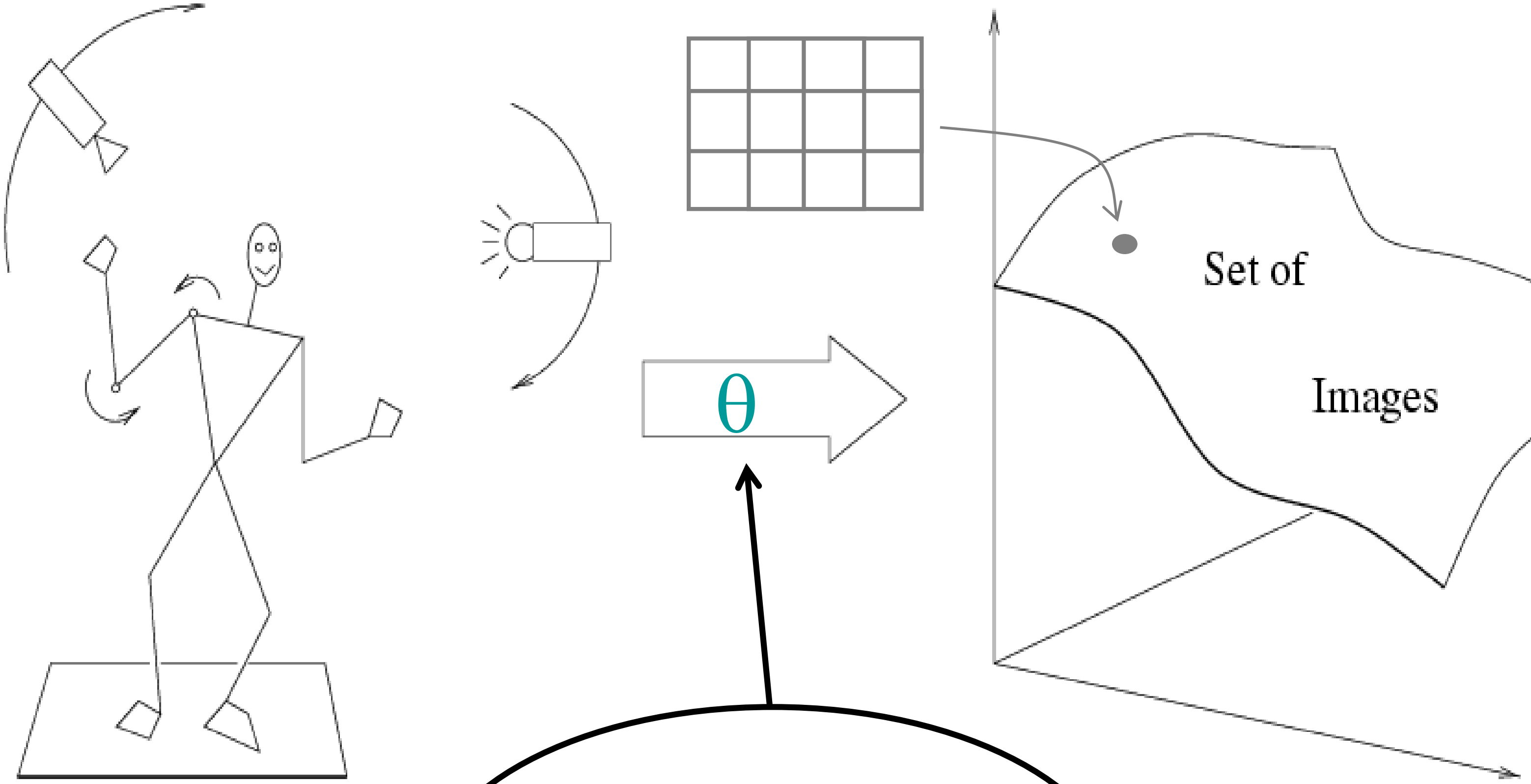
Svetlana Lazebnik

History of ideas in recognition

- 1960s – early 1990s: the geometric era

No digital cameras!
Slow compute!

Svetlana Lazebnik



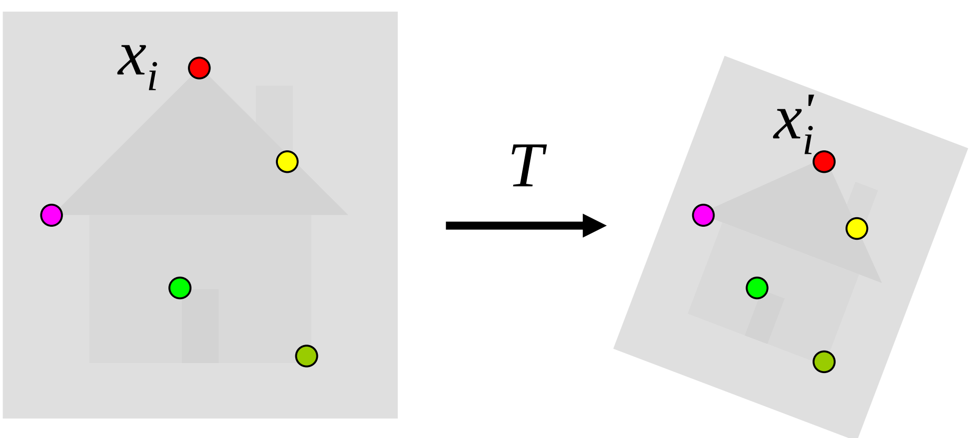
Variability: Camera position
Illumination

Shape is known

Roberts (1965); Lowe (1987); Faugeras & Hebert (1986); Grimson & Lozano-Perez (1986); Huttenlocher & Ullman (1987) Svetlana Lazebnik

Alignment

- Alignment: fitting a model to a transformation between pairs of features (*matches*) in two images

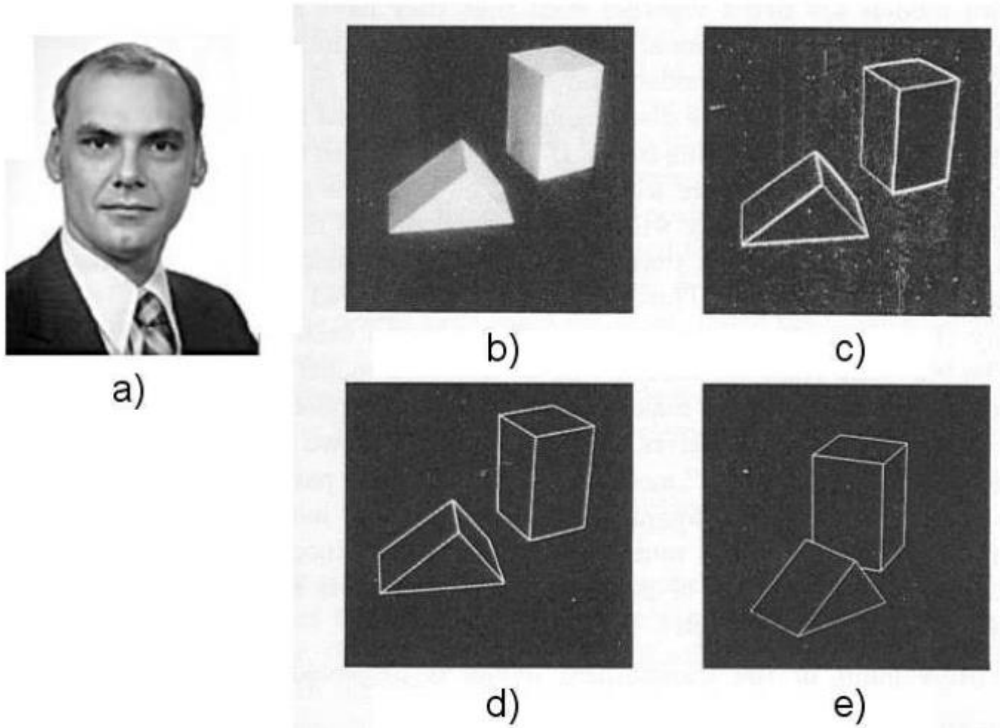


Find transformation T that minimizes

$$\sum_i \text{residual}(T(x_i), x_i')$$

Svetlana Lazebnik

Recognition as an alignment problem: Block world

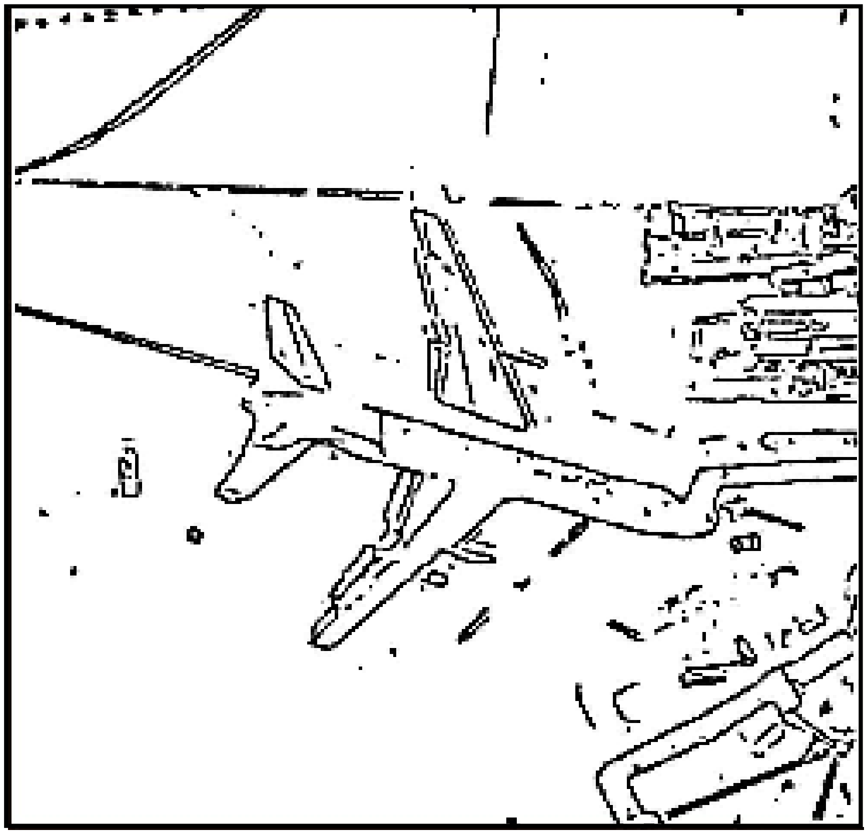


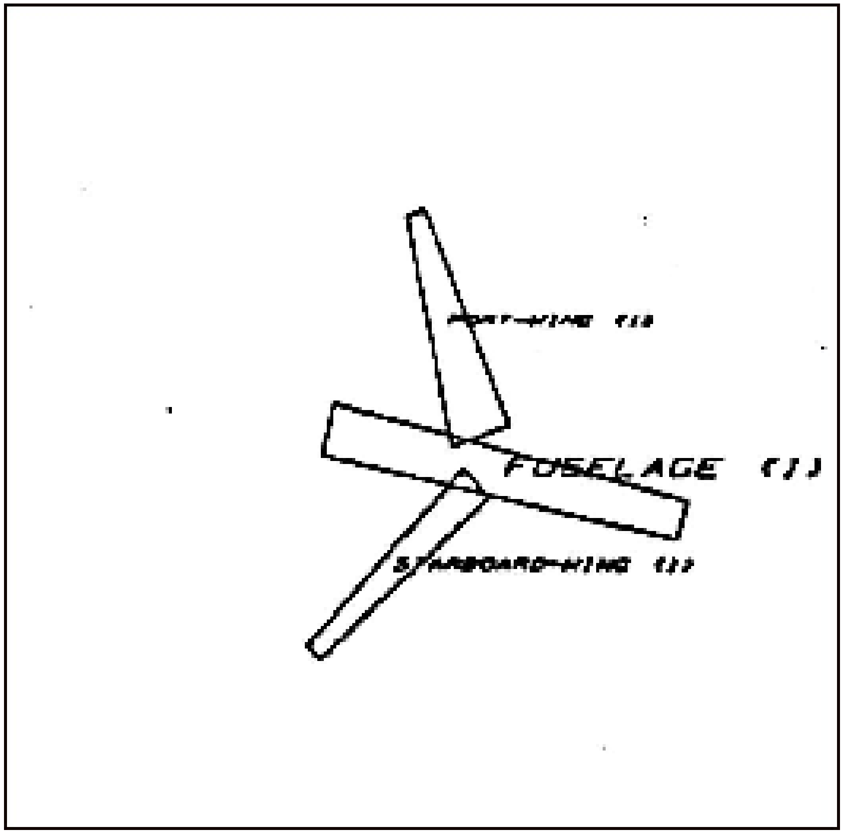
L. G. Roberts
Machine Perception of Three Dimensional Solids,
Ph.D. thesis, MIT Department of Electrical Engineering, 1963.

Fig. 1. A system for recognizing 3-d polyhedral scenes. a) L.G. Roberts. b)A blocks world scene. c)Detected edges using a 2x2 gradient operator. d) A 3-d polyhedral description of the scene, formed automatically from the single image. e) The 3-d scene displayed with a viewpoint different from the original image to demonstrate its accuracy and completeness. (b) - e) are taken from [64] with permission MIT Press.)

J. Mundy, Object Recognition in the Geometric Era: a Retrospective, 2006

Representing and recognizing object categories is harder...

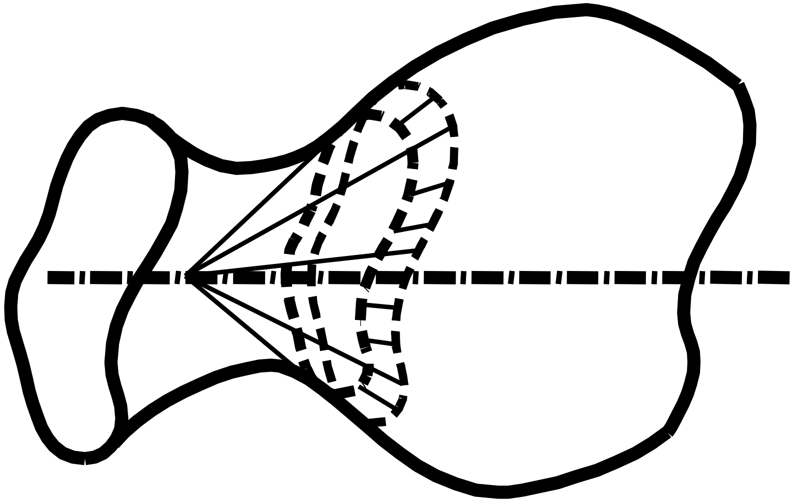




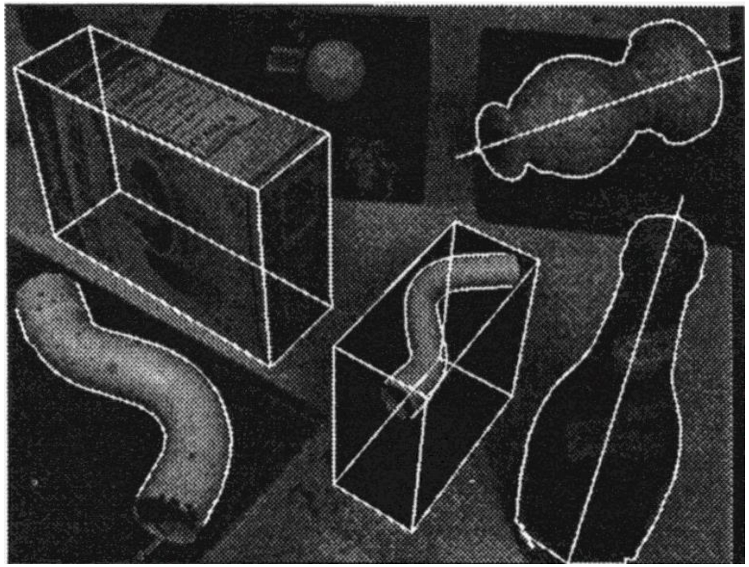
ACRONYM (Brooks and Binford, 1981)

Binford (1971), Nevatia & Binford (1972), Marr & Nishihara (1978)

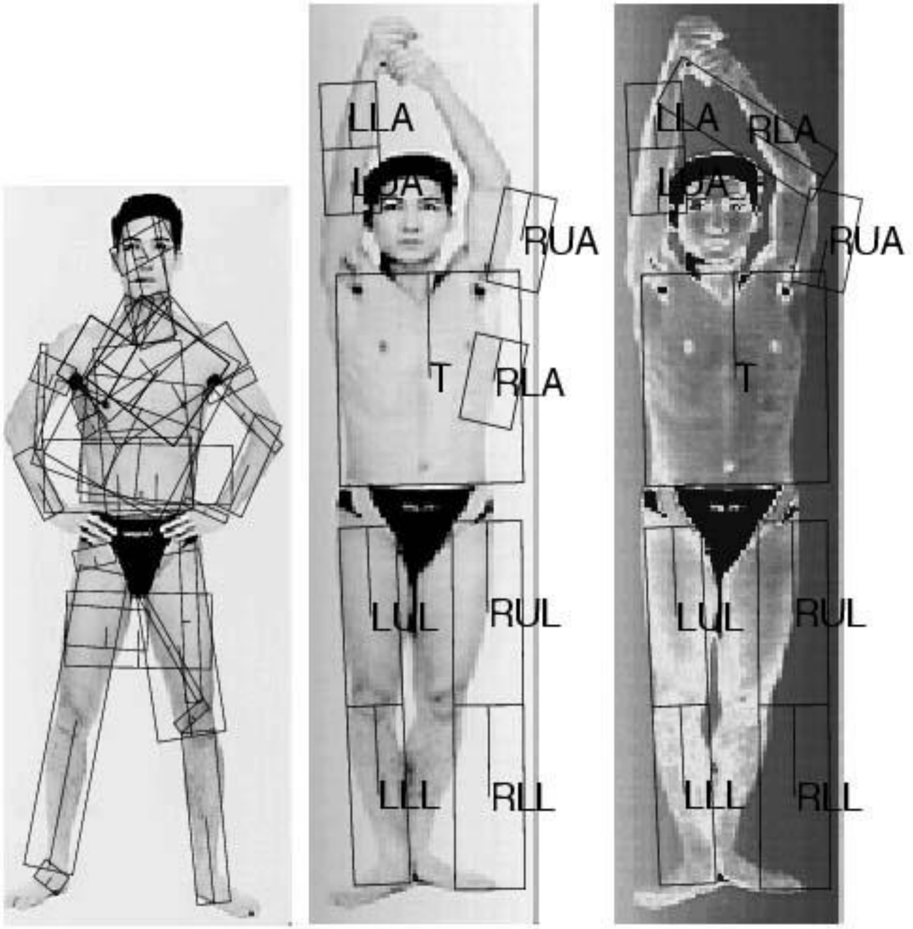
General shape primitives?



Generalized cylinders
Ponce et al. (1989)



Zisserman et al. (1995)



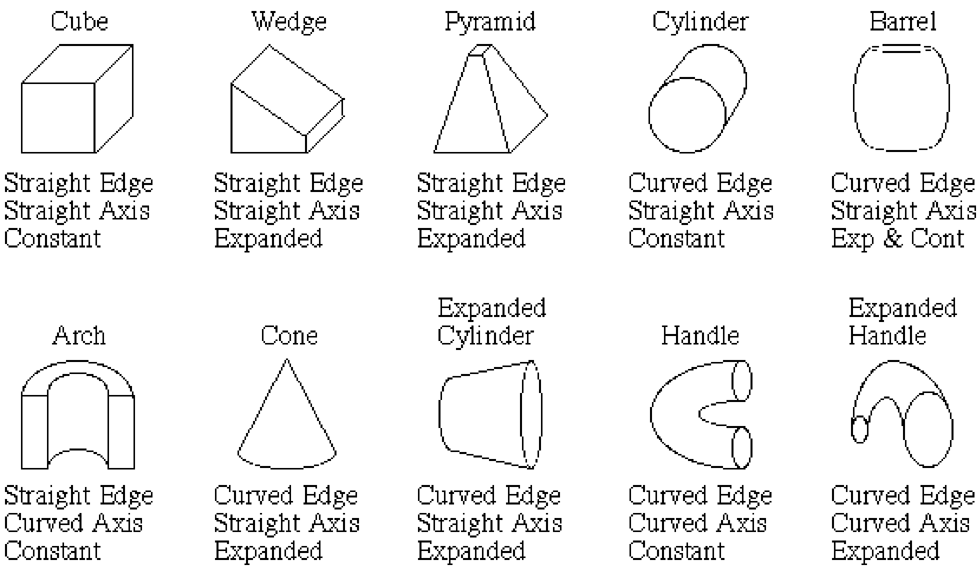
Forsyth (2000)

Svetlana Lazebnik

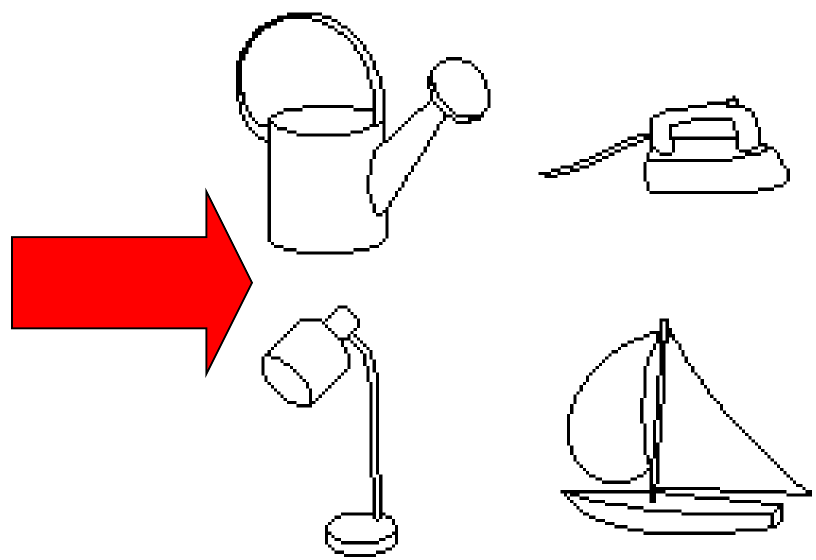
Recognition by components

Biederman (1987)

Primitives (geons)



Objects



http://en.wikipedia.org/wiki/Recognition_by_Components_Theory

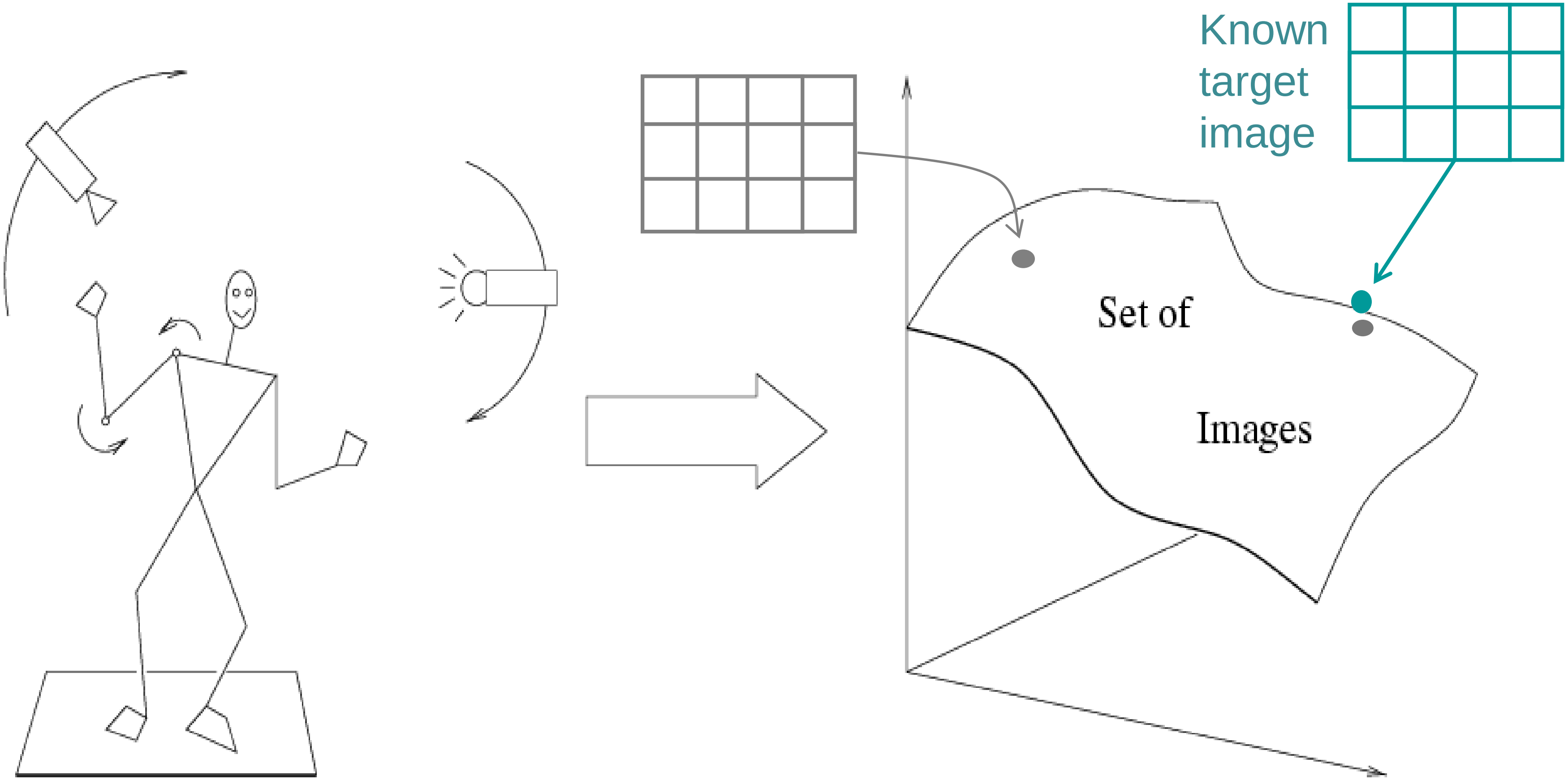
Svetlana Lazebnik

History of ideas in recognition

- 1960s – early 1990s: the geometric era
- 1990s: appearance-based models

No digital cameras!
Slow compute!

Slow compute!



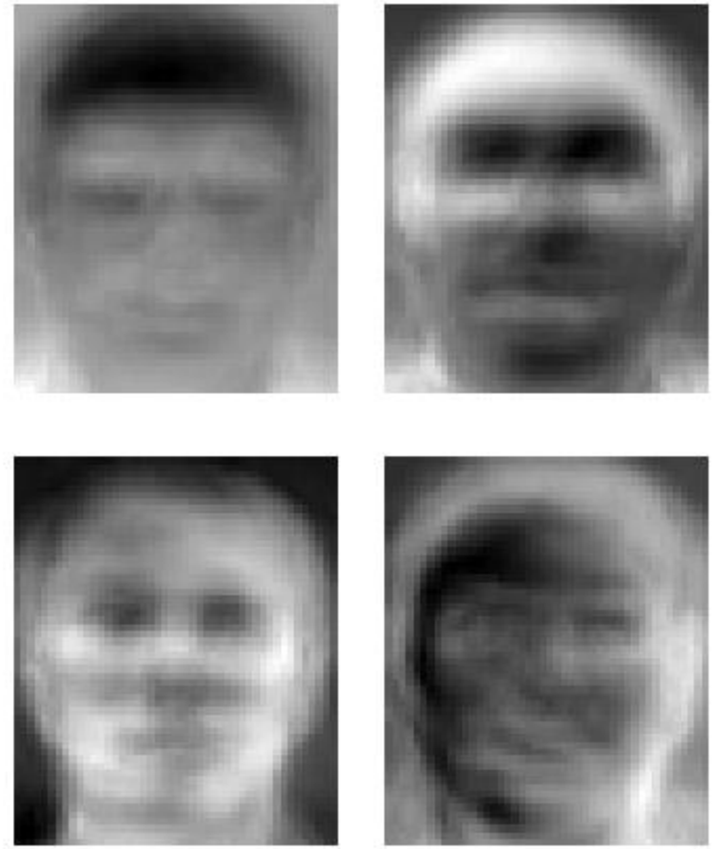
Empirical models of image variability

Appearance-based techniques

Turk & Pentland (1991); Murase & Nayar (1995); etc.

Svetlana Lazebnik

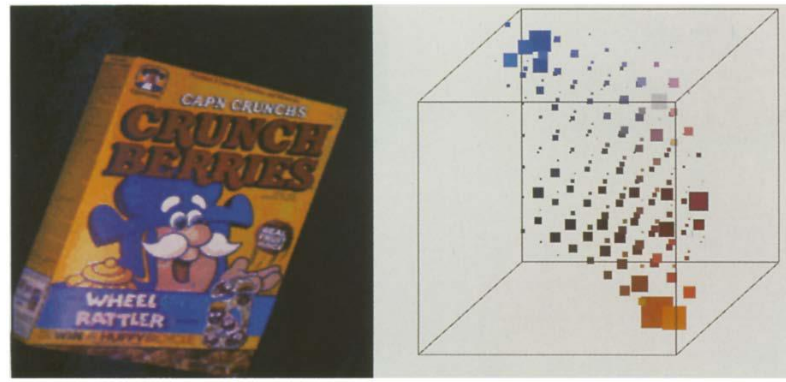
Eigenfaces (Turk & Pentland, 1991)

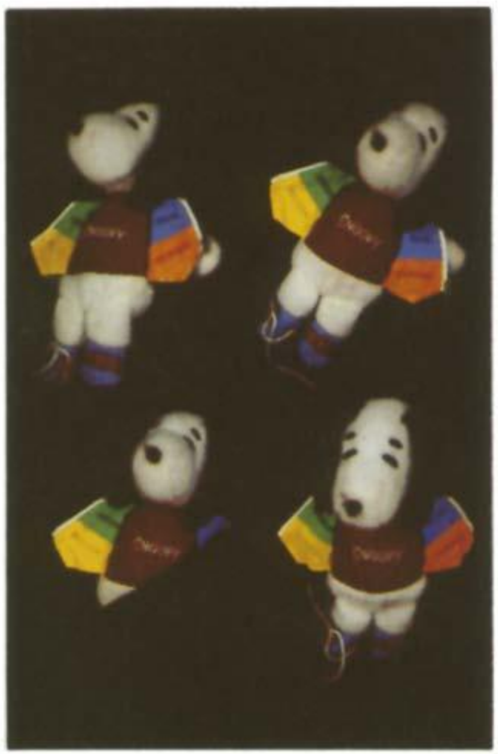


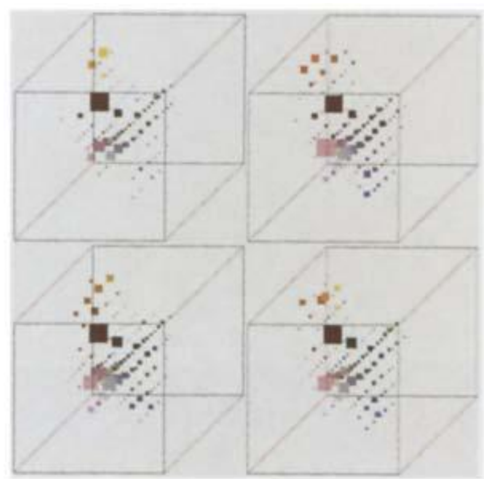
Experimental	Correct/Unknown Recognition Percentage		
Condition	Lighting	Orientation	Scale
Forced classification	96/0	85/0	64/0
Forced 100% accuracy	100/19	100/39	100/60
Forced 20% unknown rate	100/20	94/20	74/20

Svetlana Lazebnik

Color Histograms







Swain and Ballard, [Color Indexing](), IJCV 1991.

Svetlana Lazebnik

History of ideas in recognition

- 1960s – early 1990s: the geometric era

No digital cameras!
Slow compute!

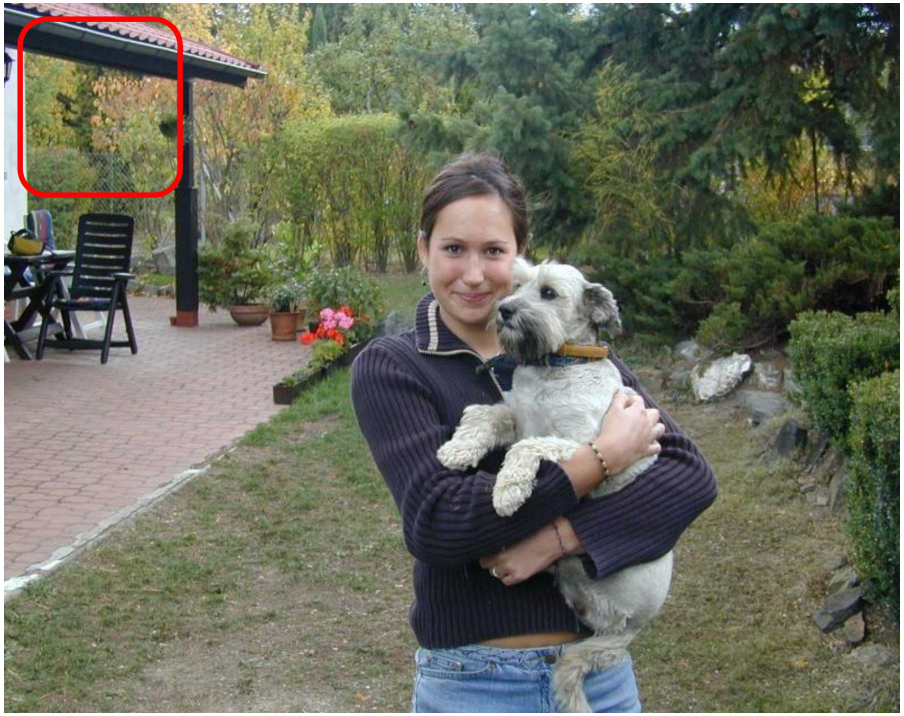
- 1990s: appearance-based models

Slow compute!

- 1990s – present: sliding window approaches

Svetlana Lazebnik

Sliding window approaches



Sliding window approaches

- Turk and Pentland, 1991
- Belhumeur, Hespanha, & Kriegman, 1997
- Schneiderman & Kanade 2004
- Viola and Jones, 2000

- Schneiderman & Kanade, 2004
- Argawal and Roth, 2002
- Poggio et al. 1993

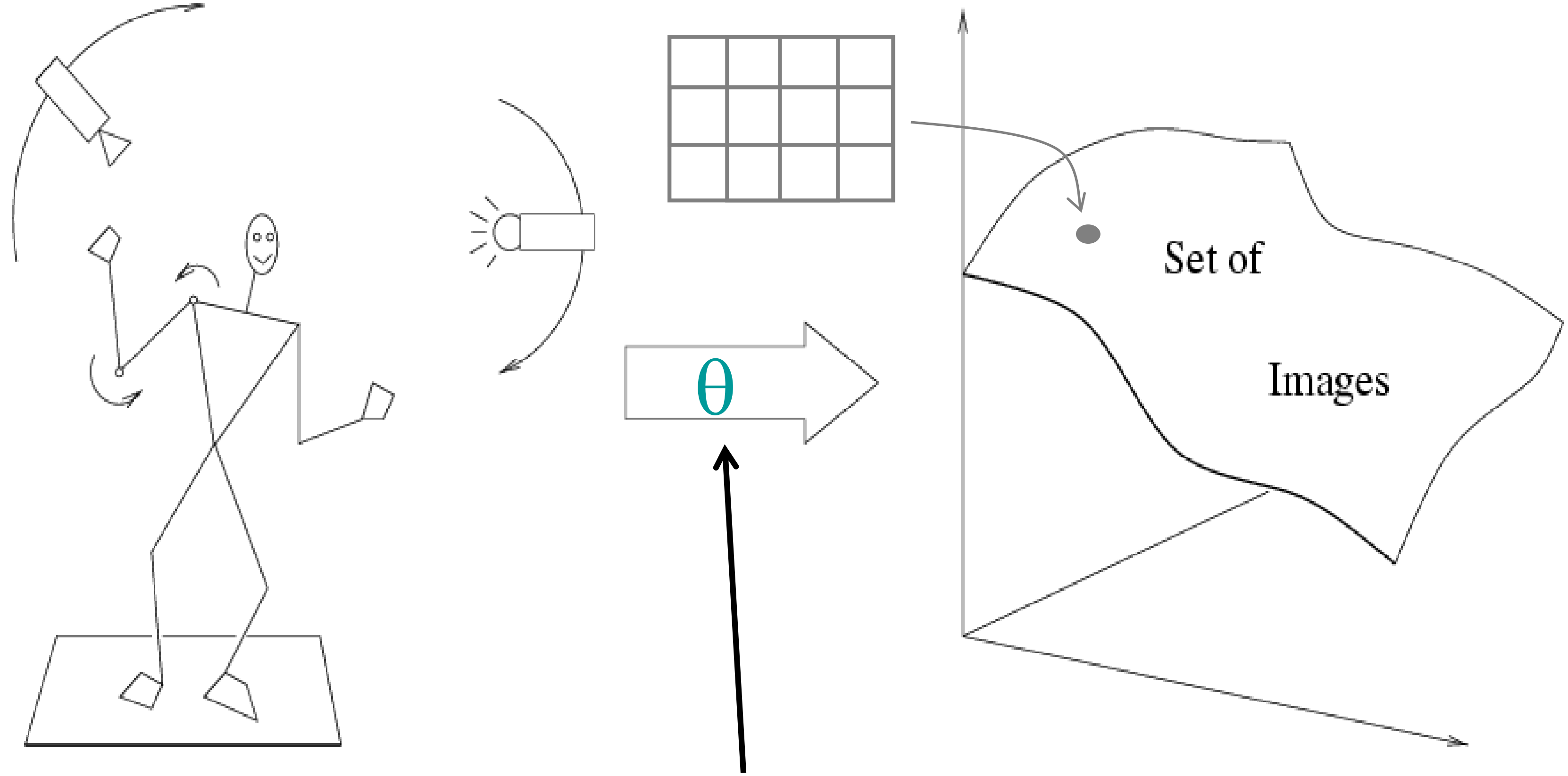
History of ideas in recognition

- 1960s – early 1990s: the geometric era
- 1990s: appearance-based models
- Mid-1990s: sliding window approaches
- Late 1990s: local features

No digital cameras!
Slow compute!

Slow compute!

Svetlana Lazebnik



Variability:

Camera position
Illumination
Shape is partially known

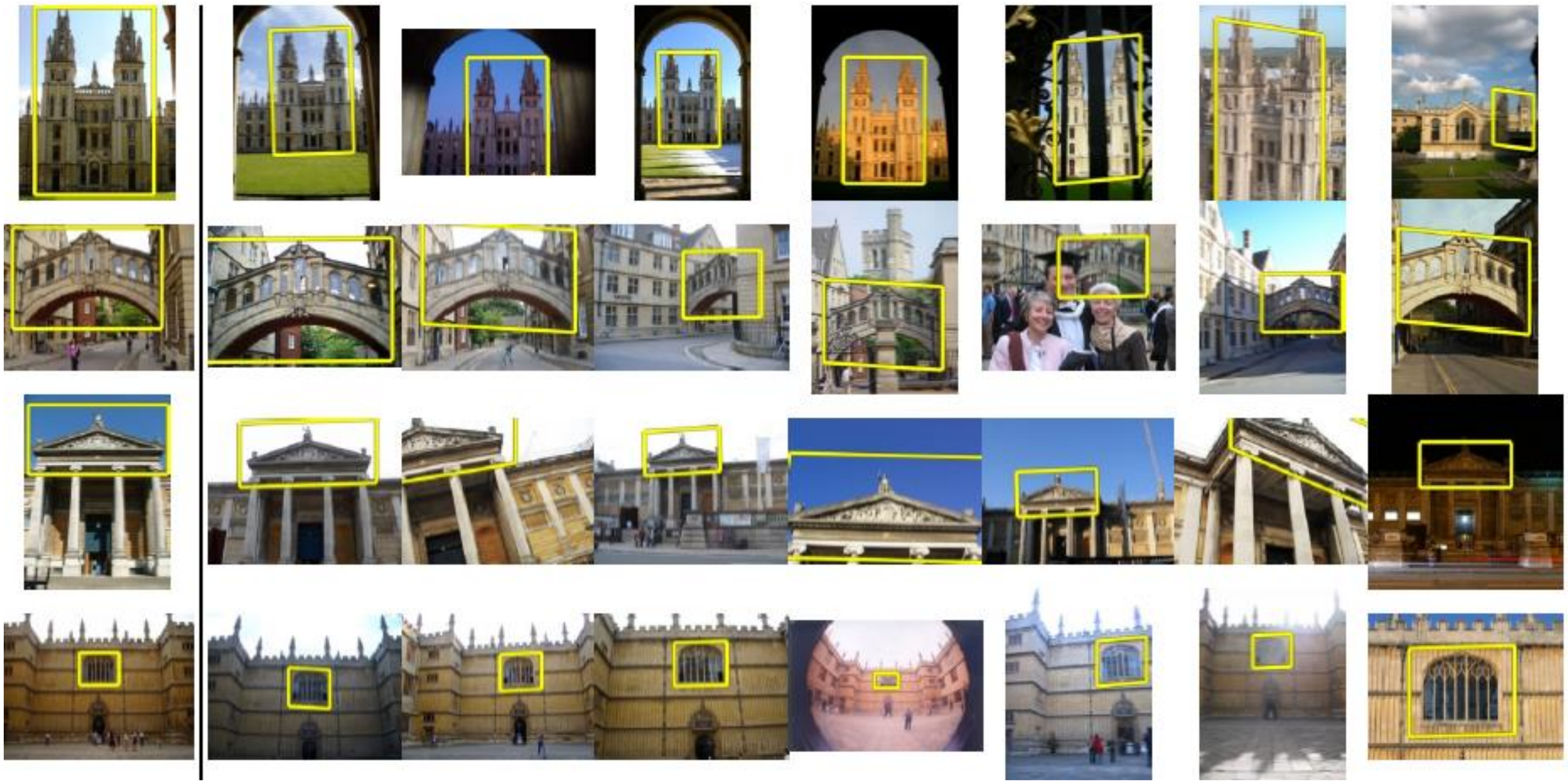
Roberts (1965); Lowe (1987); Faugeras & Hebert (1986); Grimson & Lozano-Perez (1986); Huttenlocher & Ullman (1987)

Local features for object instance recognition

D. Lowe (1999, 2004)

Large-scale image search

Combining local features, indexing, and spatial constraints



Philbin et al. '07

Large-scale image search

Combining local features, indexing, and spatial constraints

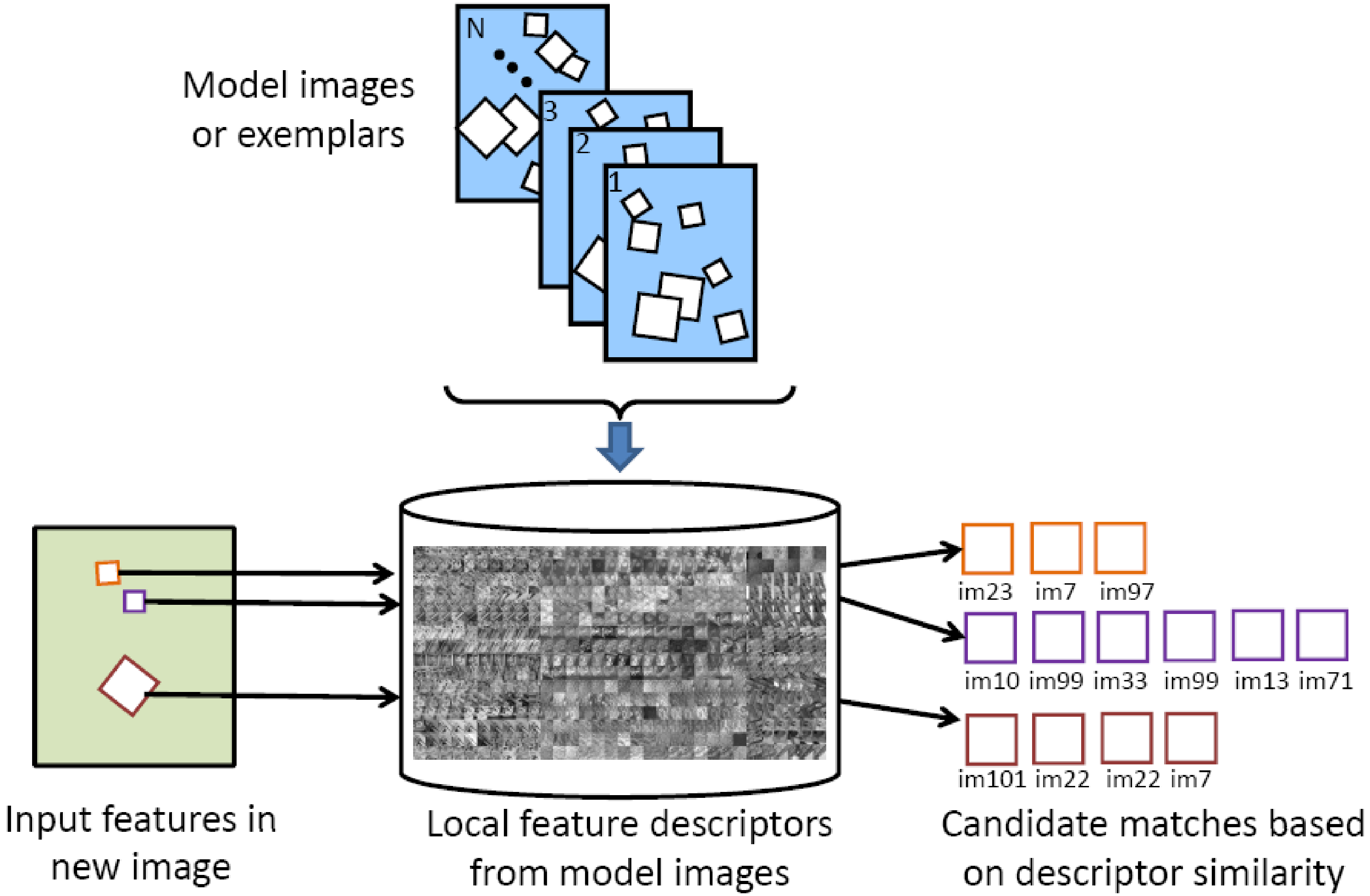


Image credit: K. Grauman and B. Leibe

Large-scale image search

Combining local features, indexing, and spatial constraints

Available on phones that run Android 1.6+ (i.e. Donut or Eclair)

Svetlana Lazebnik

History of ideas in recognition

- 1960s – early 1990s: the geometric era
- 1990s: appearance-based models
- Mid-1990s: sliding window approaches
- Late 1990s: local features
- Early 2000s: parts-and-shape models

Parts-and-shape models

- Model:
  - Object as a set of parts
  - Relative locations between parts
  - Appearance of part

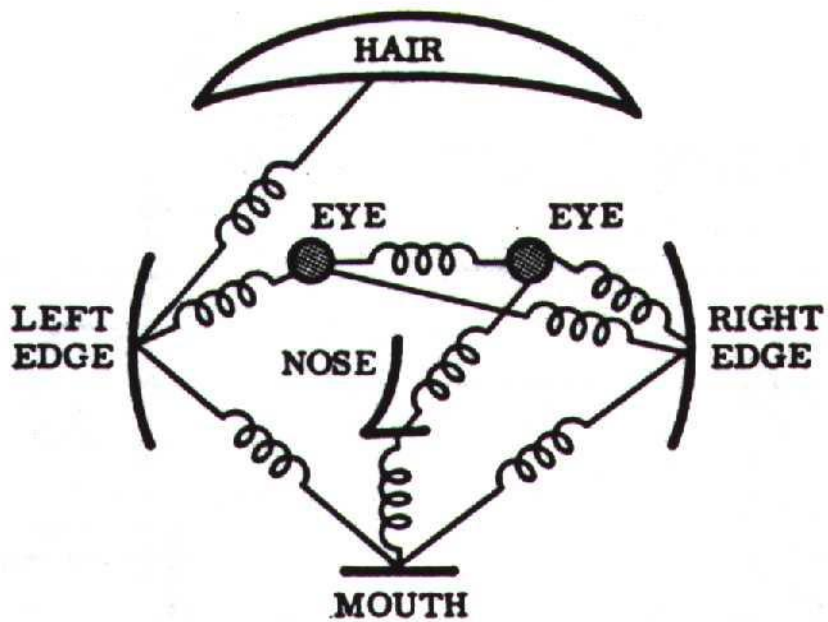


Figure from [Fischler & Elschlager 73]

Constellation models

Weber, Welling & Perona (2000), Fergus, Perona & Zisserman (2003)

History of ideas in recognition

- 1960s – early 1990s: the geometric era — No digital cameras! Slow compute!
- 1990s: appearance-based models — Slow compute!
- Mid-1990s: sliding window approaches
- Late 1990s: local features
- Early 2000s: parts-and-shape models
- Mid-2000s: bags of features (next!) — Early GPU compute.

Svetlana Lazebnik

History of ideas in recognition

- 1960s – early 1990s: the geometric era
  No digital cameras!
  Slow compute!
- 1990s: appearance-based models
  Slow compute!
- Mid-1990s: sliding window approaches
- Late 1990s: local features
- Early 2000s: parts-and-shape models
- Mid-2000s: bags of features (next!)
  Early GPU compute.
- *Present trends:*
  Combined local and global methods, context, deep learning
  GPU/cloud compute.

Recognition Issues

How to summarize the content of an entire image?
How to gauge overall similarity?

How large should the vocabulary be?
How to perform quantization efficiently?

How to score the retrieval results?

How might we add more spatial verification?

Kristen Grauman

Recognition Issues

How to summarize the content of an entire image?
How to gauge overall similarity?

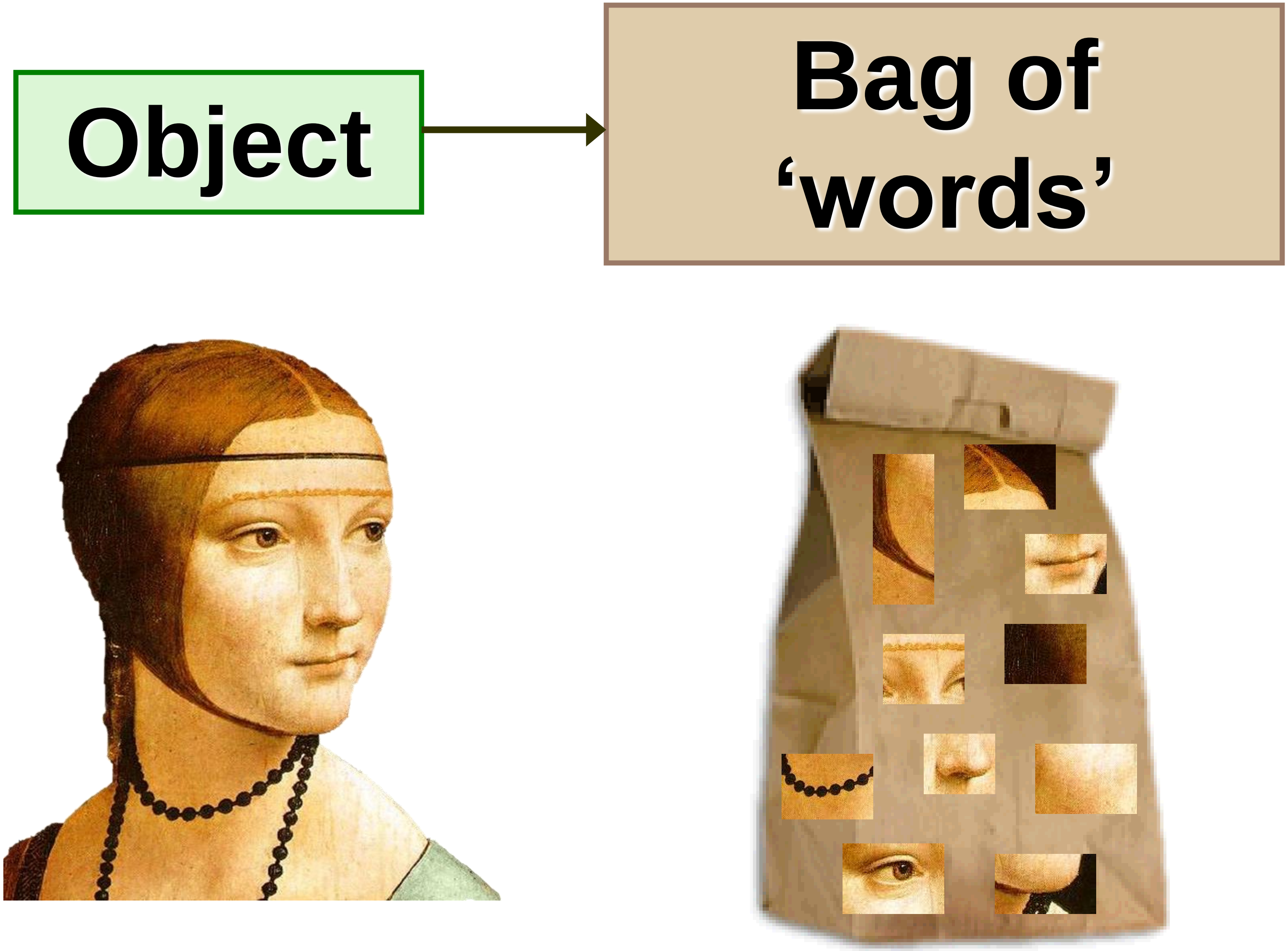
How large should the vocabulary be?
How to perform quantization efficiently?

How to score the retrieval results?

How might we add more spatial verification?

Kristen Grauman

Bag-of-features models



Svetlana Lazebnik

Origin 1: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Origin 1: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

US Presidential Speeches Tag Cloud
`http://chir.ag/phernalia/preztags/`

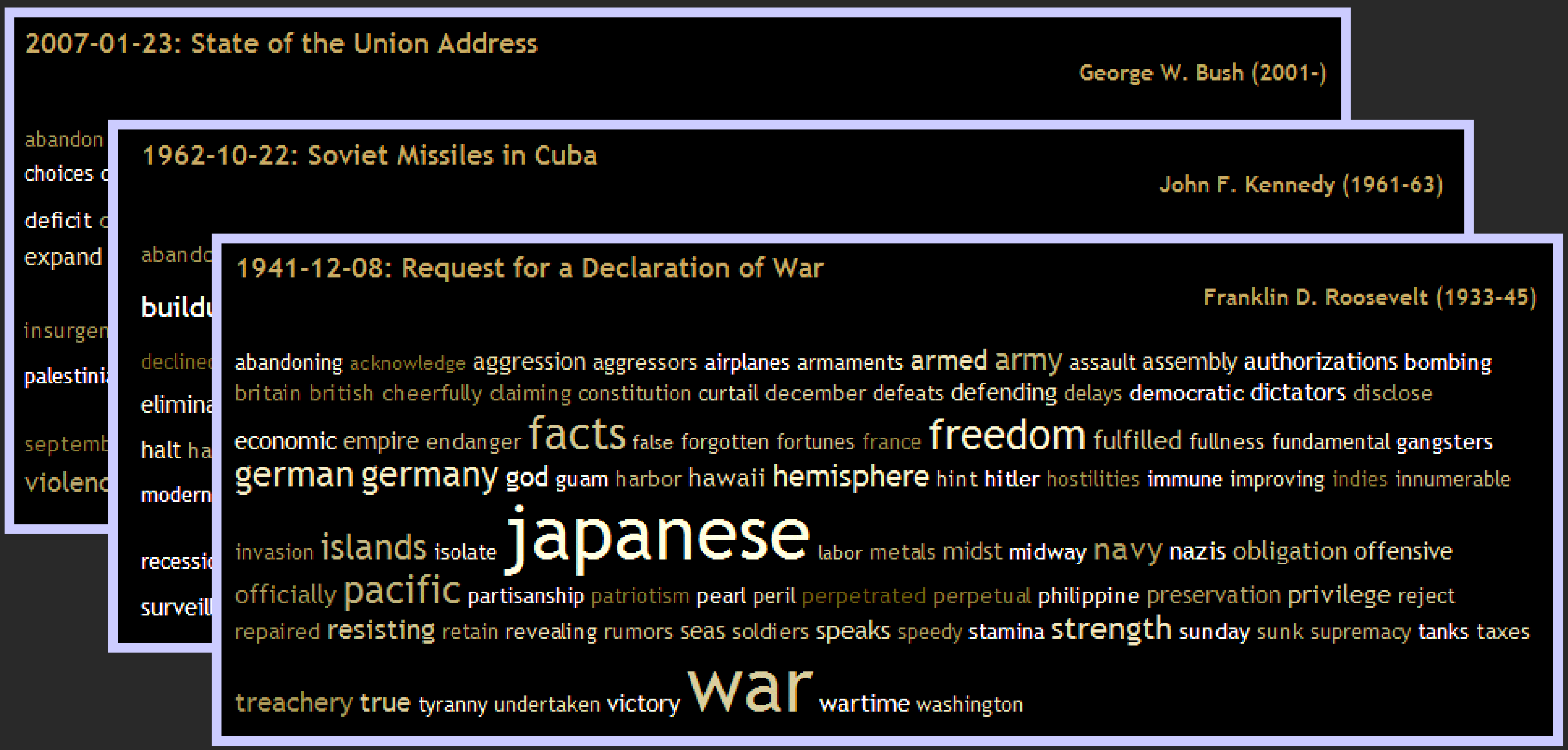
Origin 1: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

US Presidential Speeches Tag Cloud
`http://chir.ag/phernalia/preztags/`

Origin 1: Bag-of-words models

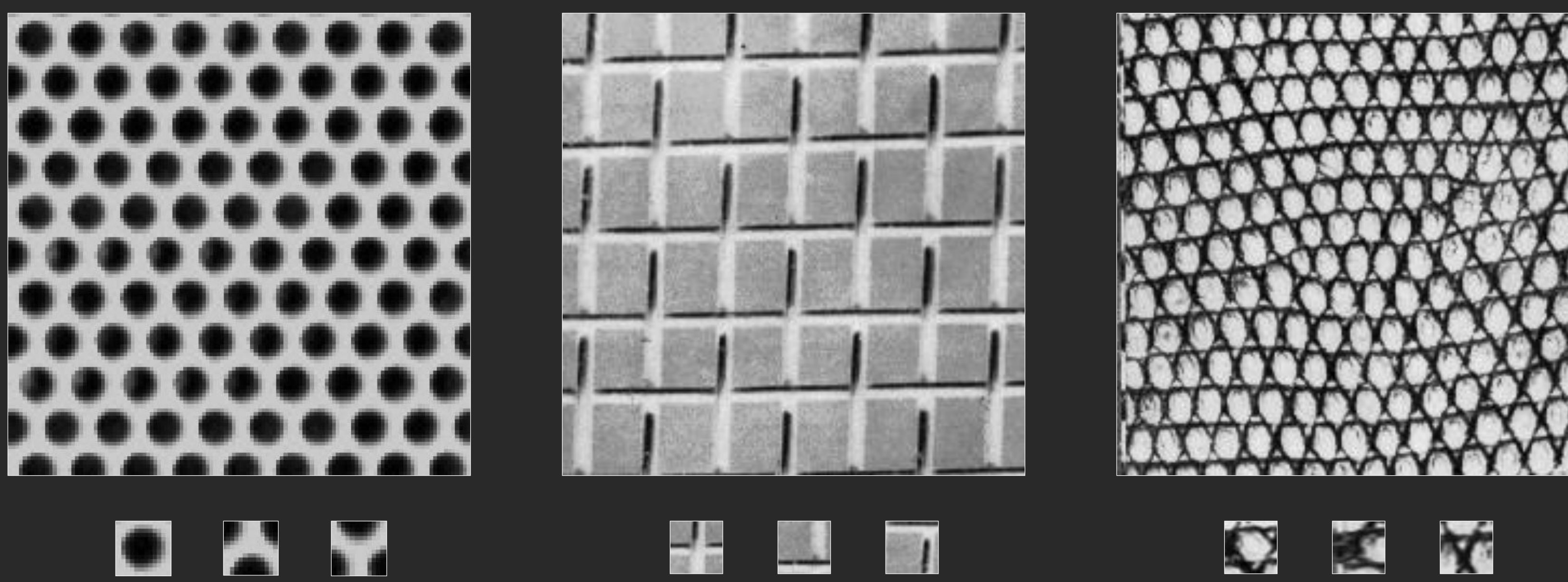
- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)



US Presidential Speeches Tag Cloud
`http://chir.ag/phernalia/preztags/`

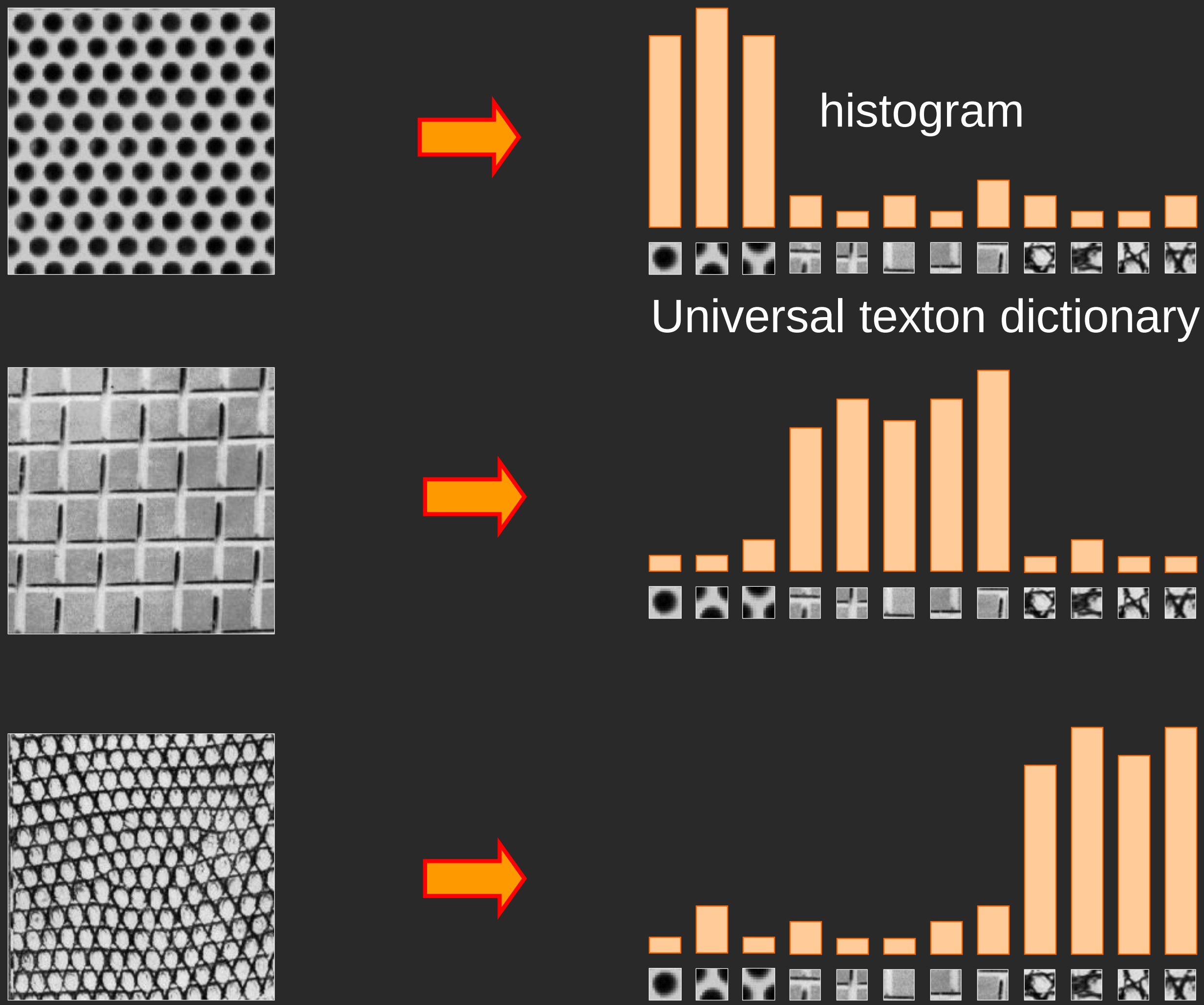
Origin 2: Texture recognition

- Characterized by repetition of basic elements or *textons*
- For stochastic textures, the identity of textons matters, not their spatial arrangement



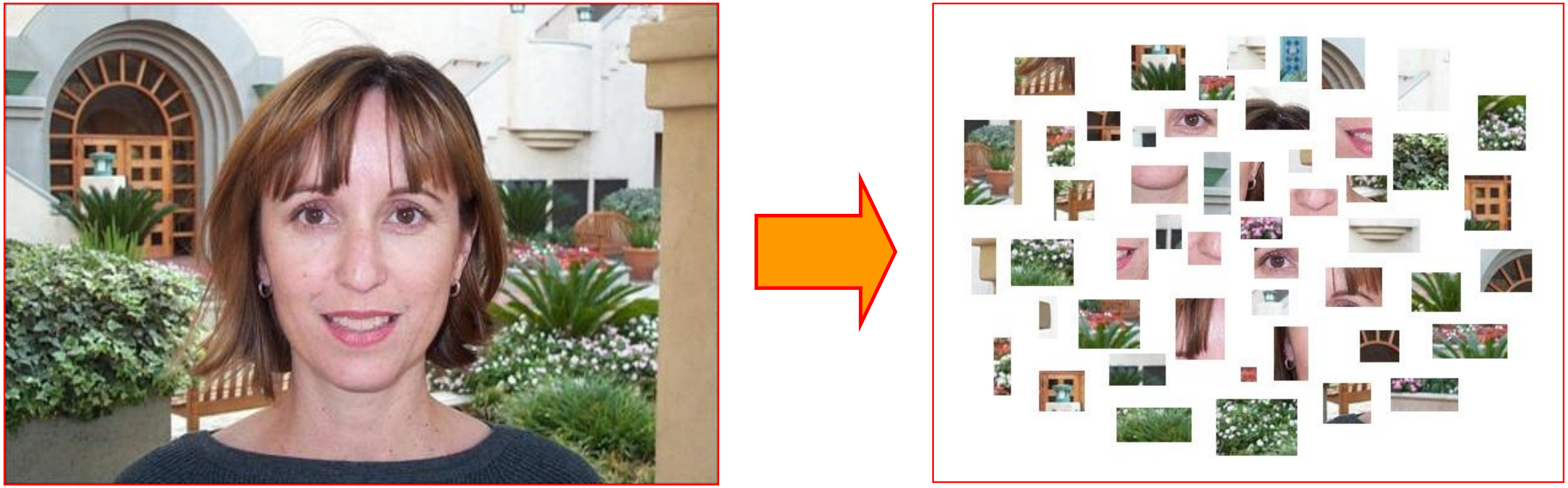
Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Origin 2: Texture recognition



Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Bag-of-features models



Svetlana Lazebnik

Objects as texture

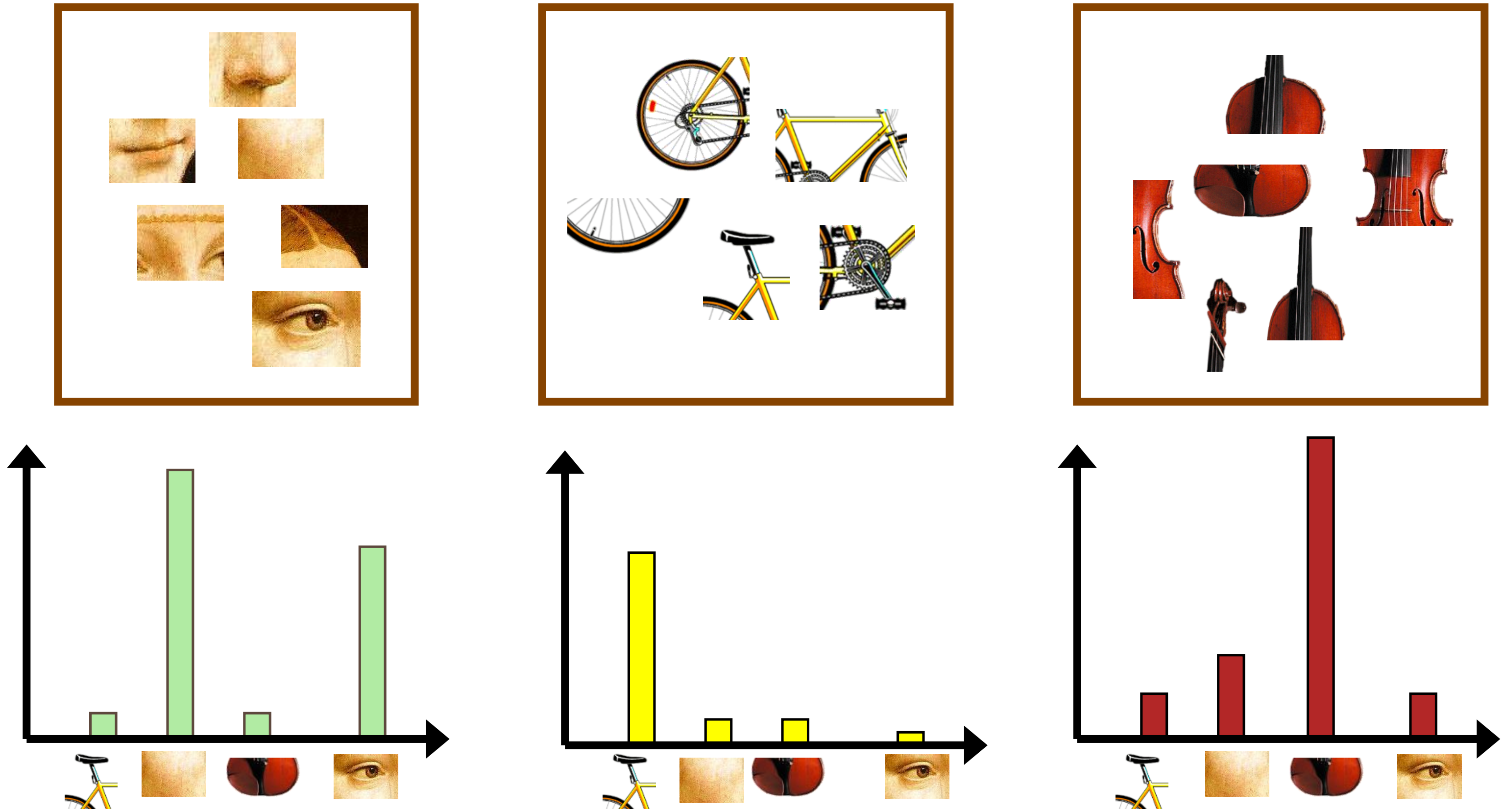
- All of these are treated as being the same

- No distinction between foreground and background: scene recognition?

Svetlana Lazebnik

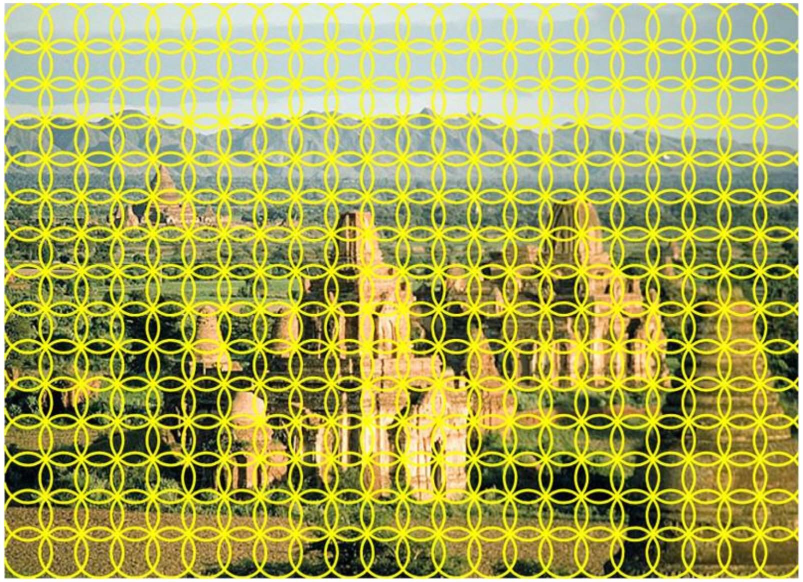
Bag-of-features steps

1. Feature extraction
2. Learn “visual vocabulary”
3. Quantize features using visual vocabulary
4. Represent images by frequencies of “visual words”

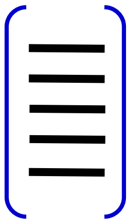


1. Feature extraction

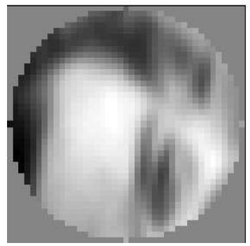
- Regular grid or interest regions



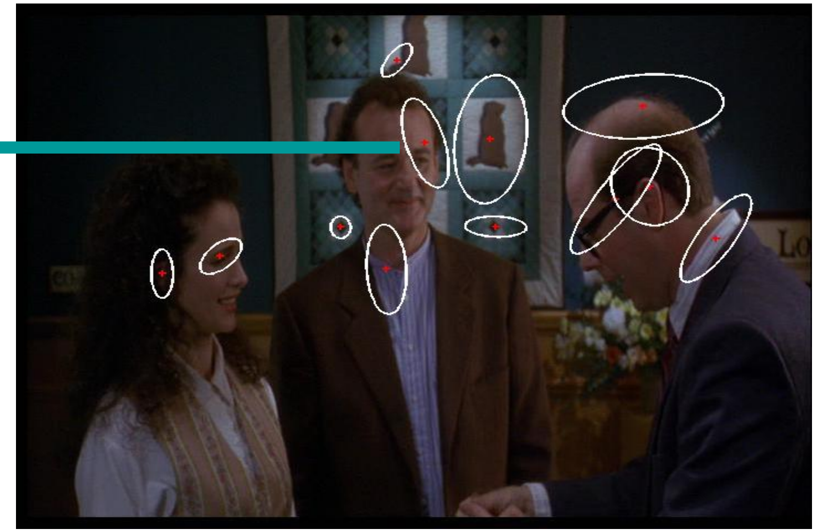
1. Feature extraction



Compute descriptor



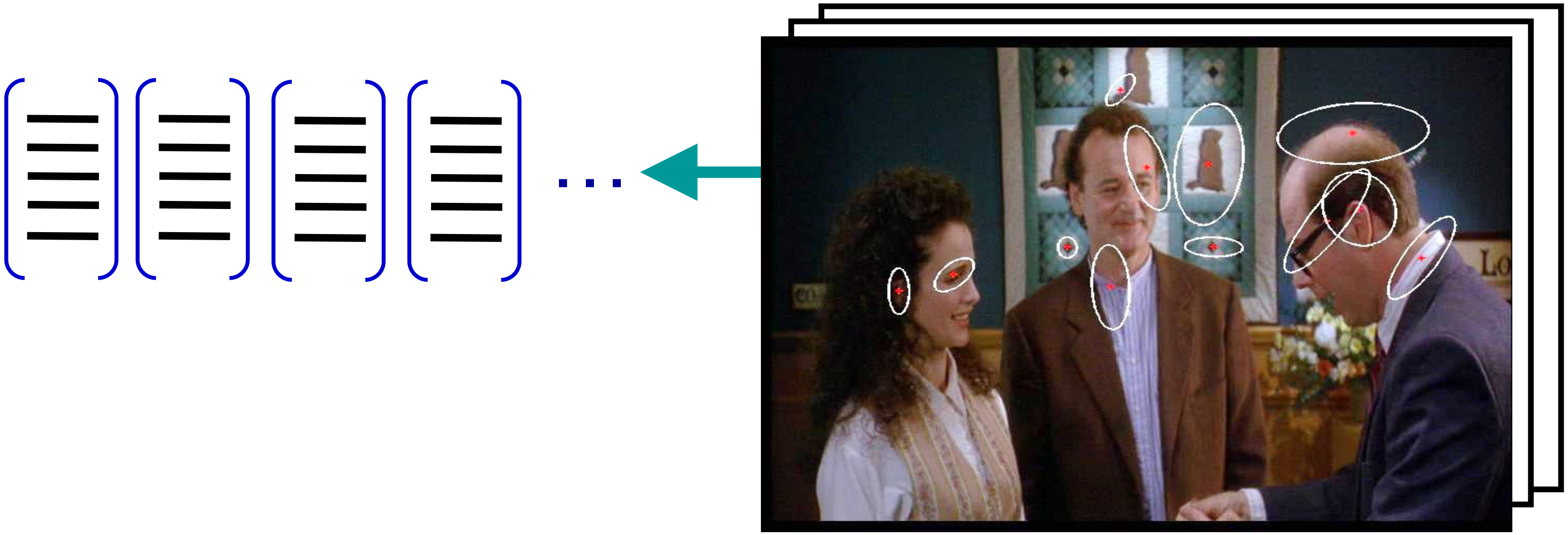
Extract patch



Detect patches

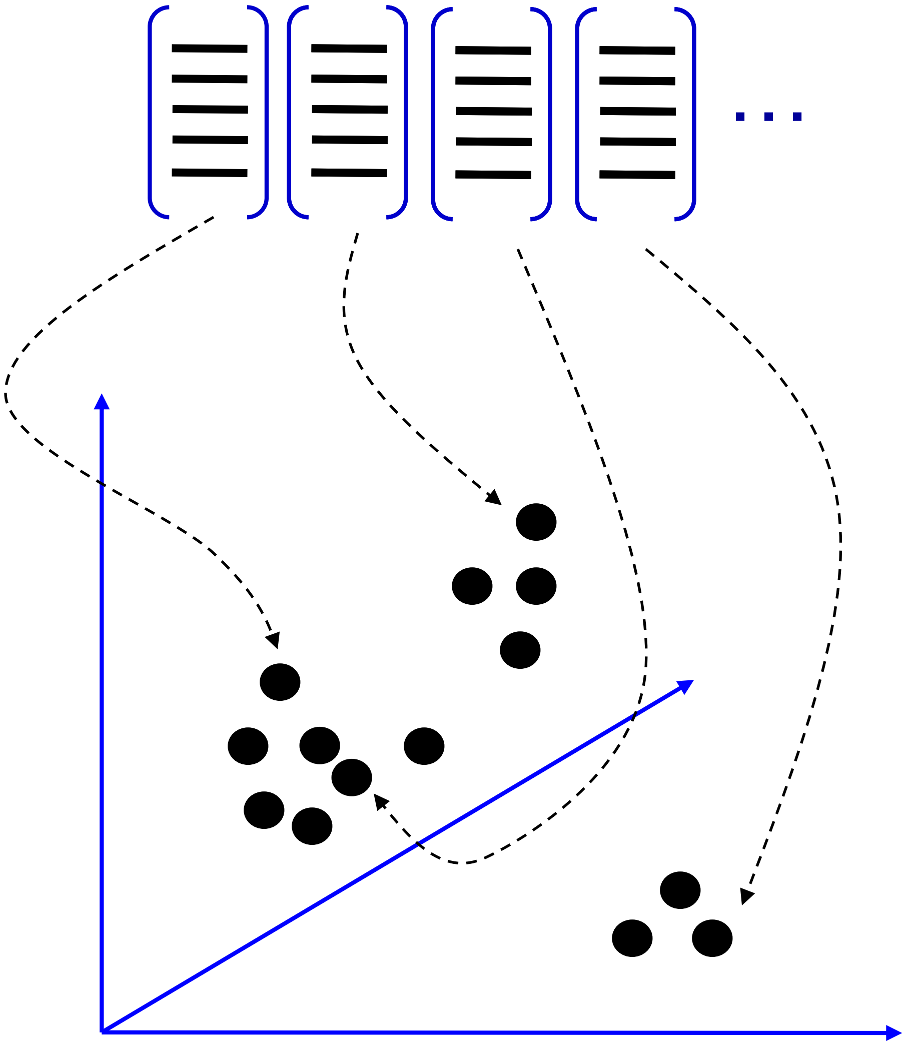
Slide credit: Josef Sivic

1. Feature extraction



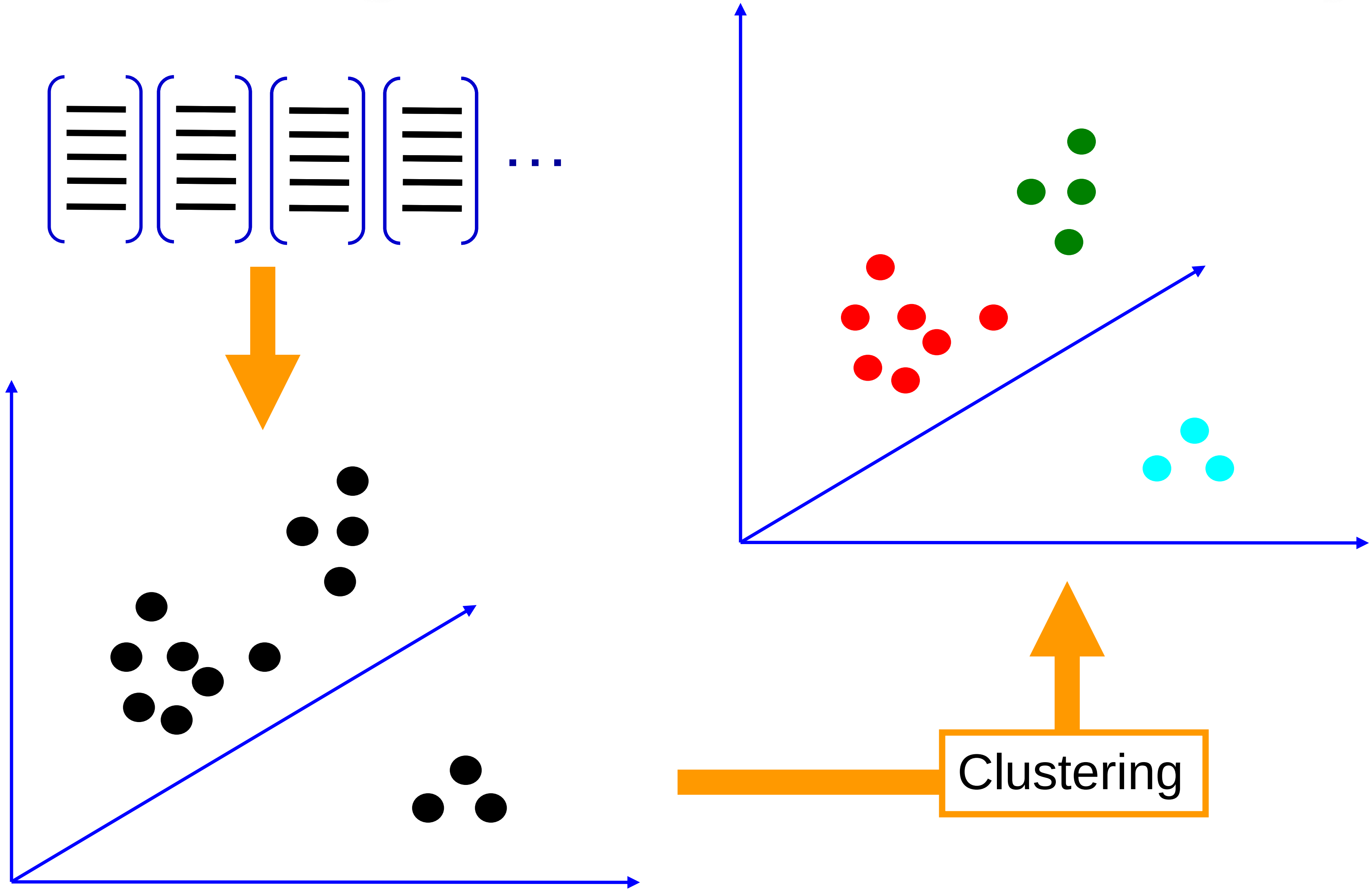
Slide credit: Josef Sivic

2. Learning the visual vocabulary



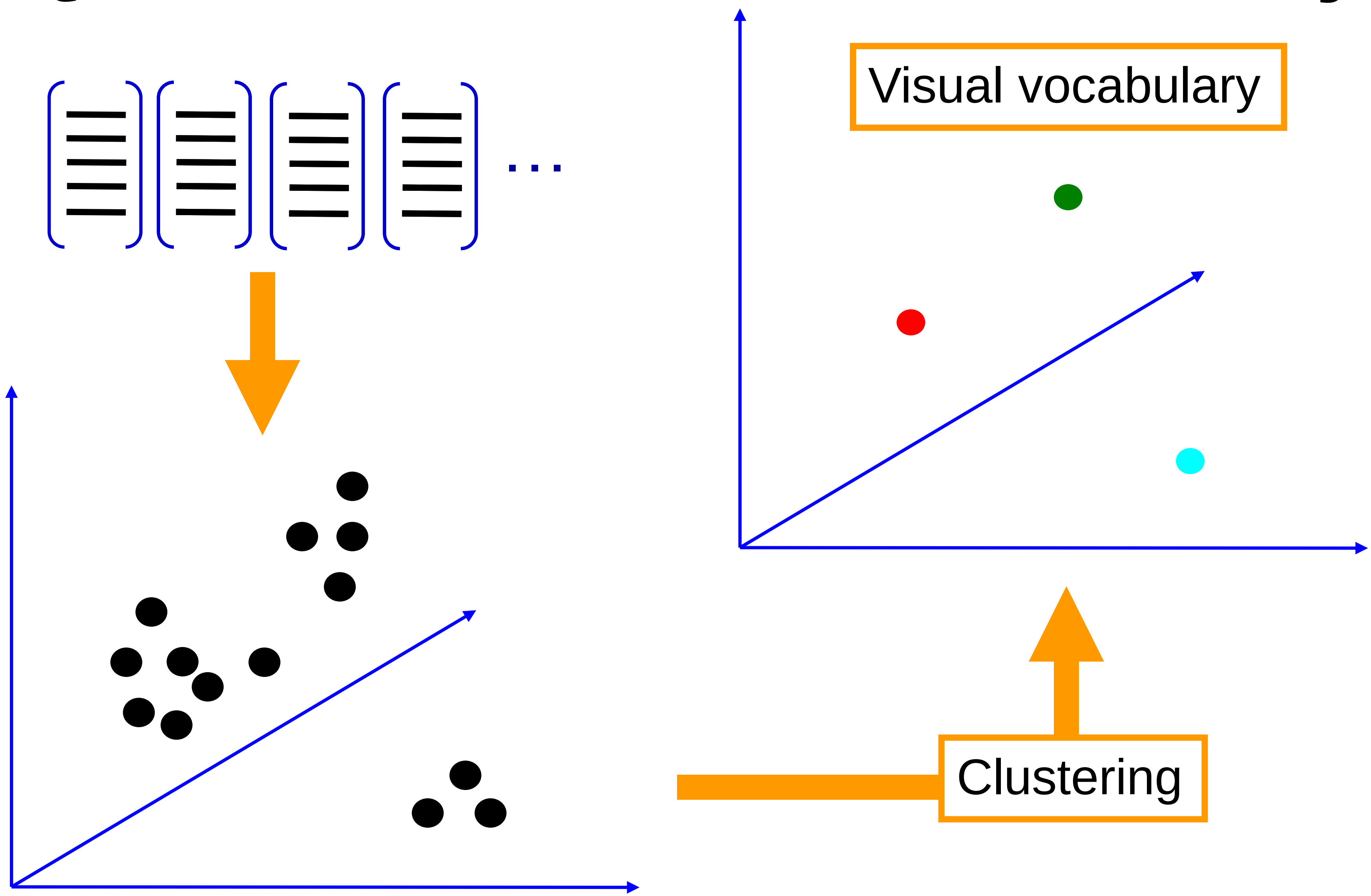
Slide credit: Josef Sivic

2. Learning the visual vocabulary

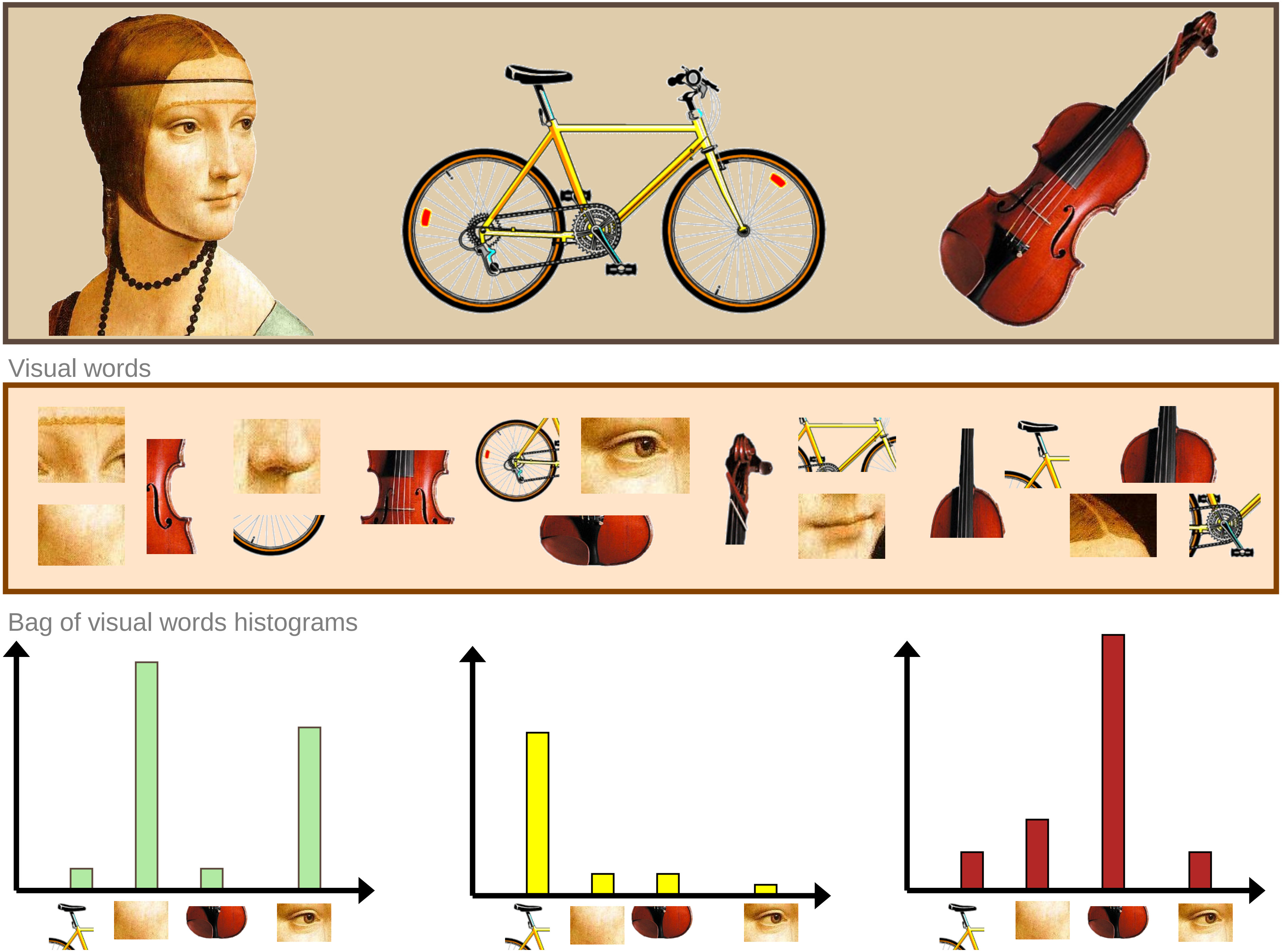


Slide credit: Josef Sivic

3. Quantize the visual vocabulary

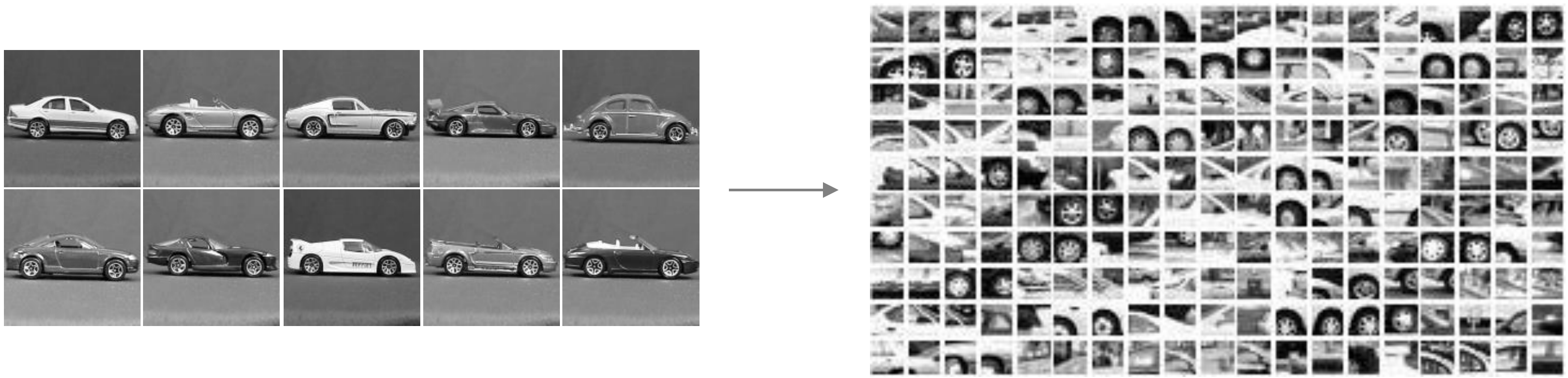


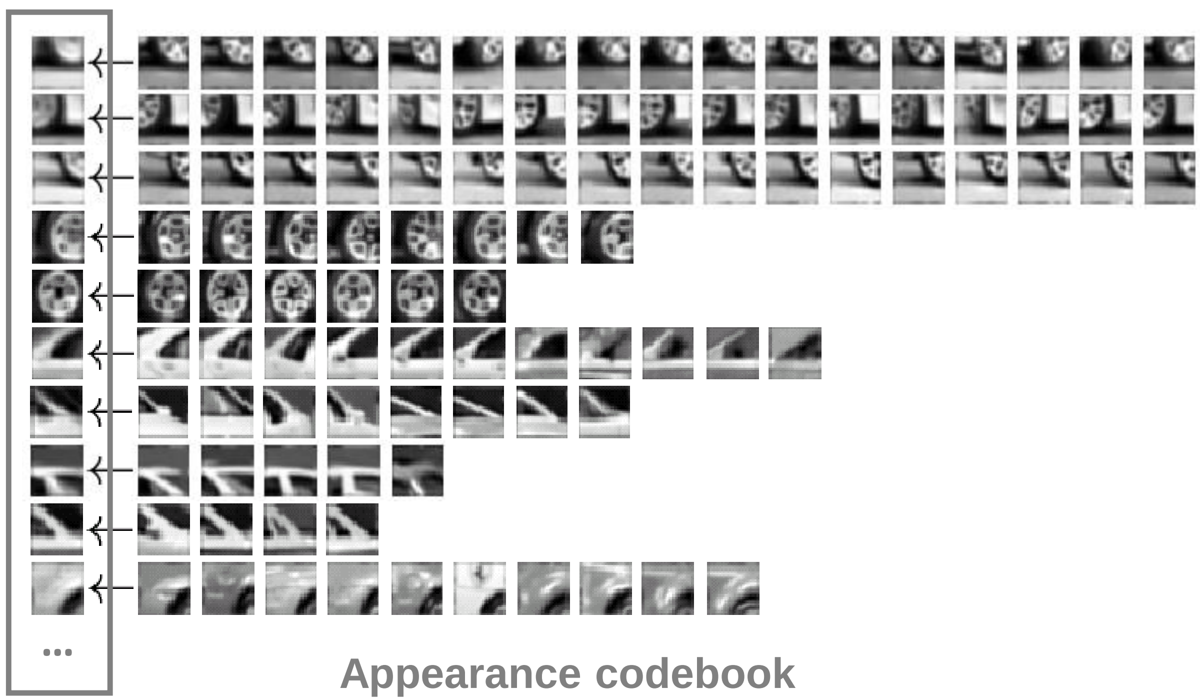
Slide credit: Josef Sivic



Visual words
Bag of visual words histograms

Example real codebook



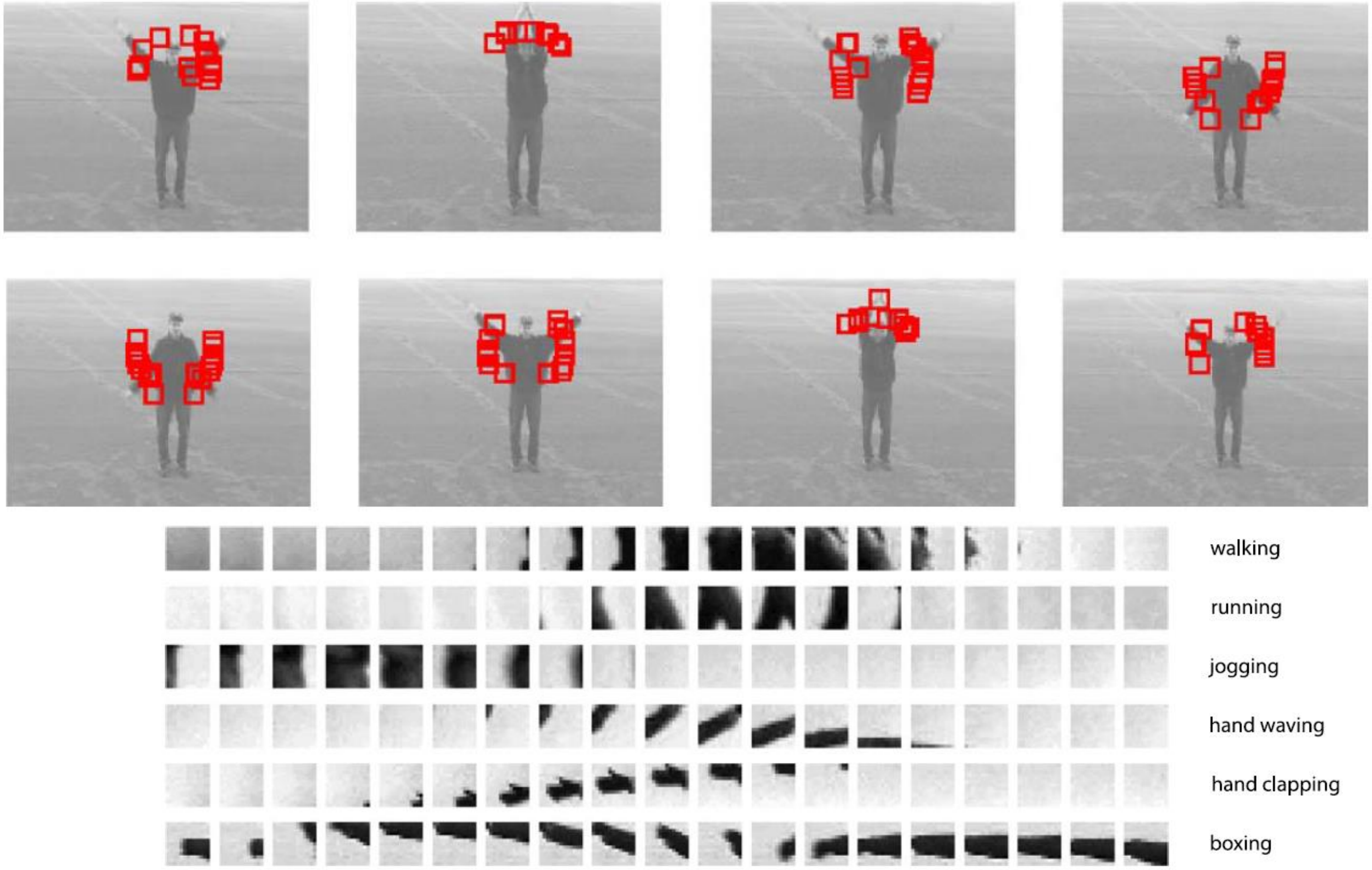


Appearance codebook

Source: B. Leibe

Bags of features for action recognition

Space-time interest points



Juan Carlos Niebles, Hongcheng Wang and Li Fei-Fei, **Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words**, IJCV 2008.

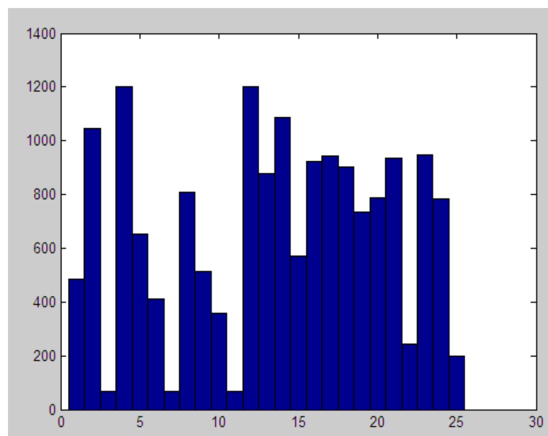
Visual words/bags of words

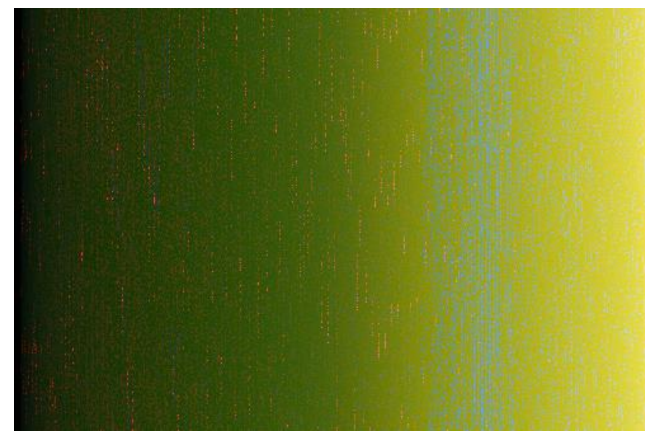
+ flexible to geometry / deformations / viewpoint
+ compact summary of image content
+ provides fixed dimensional vector representation for sets
+ very good results in practice

- background and foreground mixed when bag covers whole image -> *is it really instance recognition?*
- optimal vocabulary formation remains unclear
- basic model ignores geometry – must verify afterwards, or encode via features

Kristen Grauman

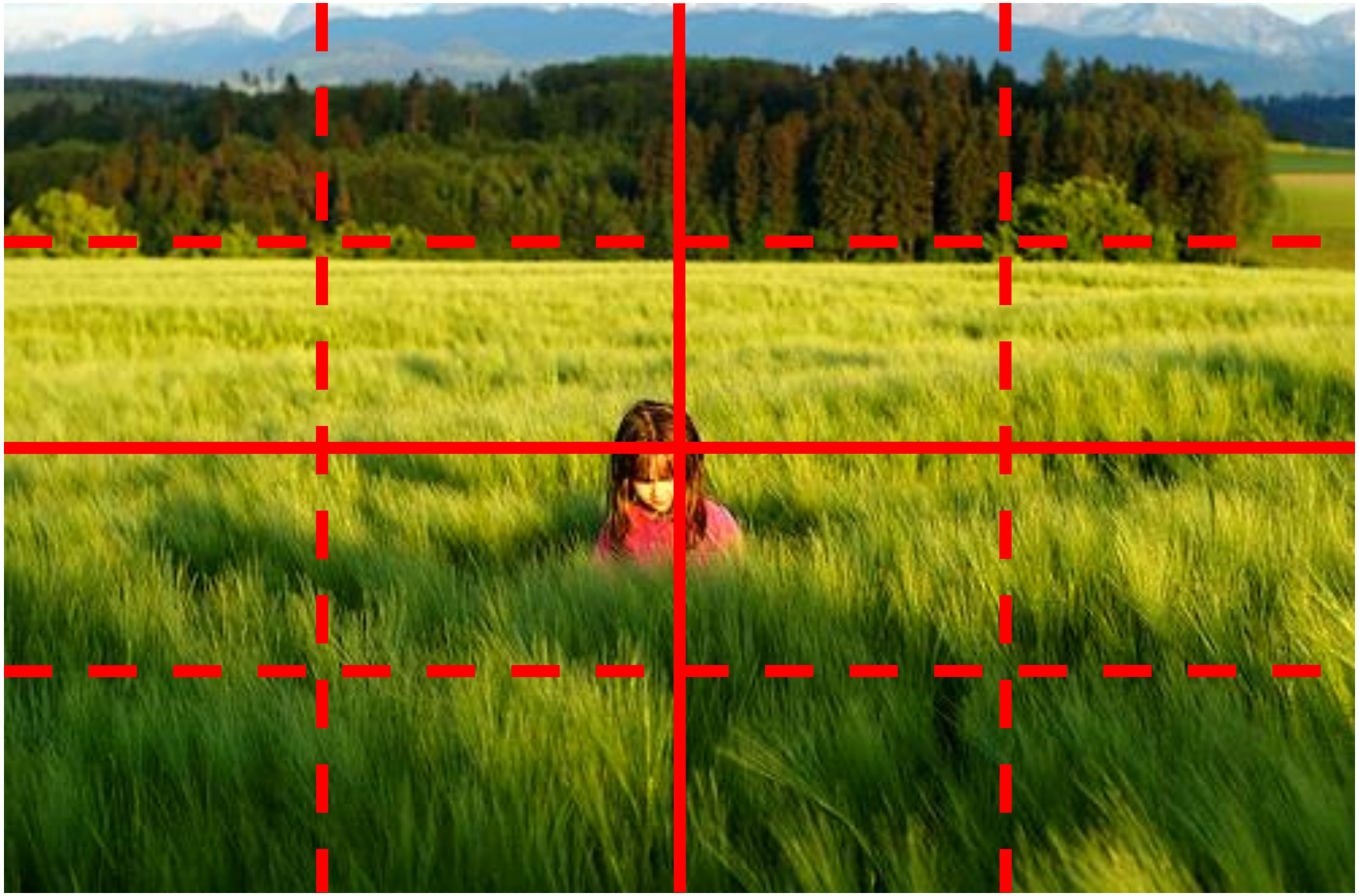
But what about layout?





All of these images have the same color histogram.
How to extend bag of words?

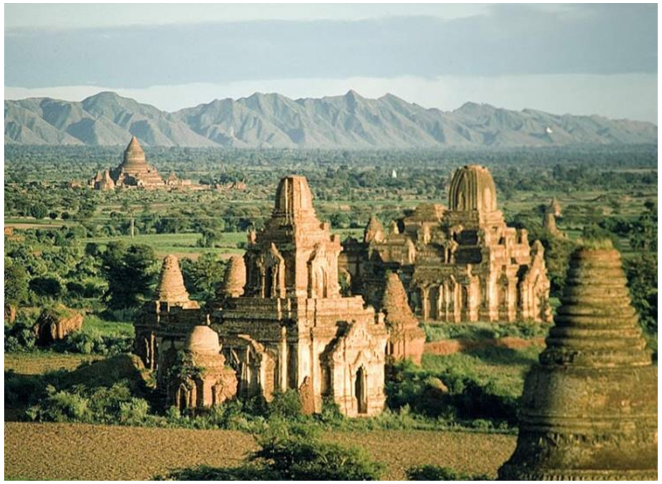
Spatial pyramid

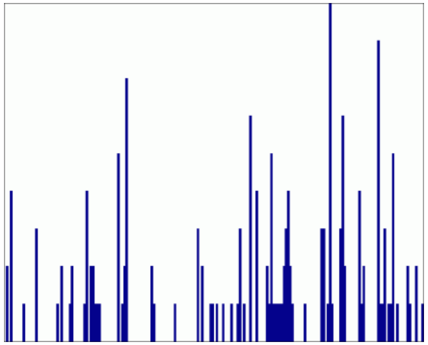


Compute histogram in each spatial bin

Spatial pyramid representation

- Extension of a bag of features
- Locally orderless representation at several levels of resolution



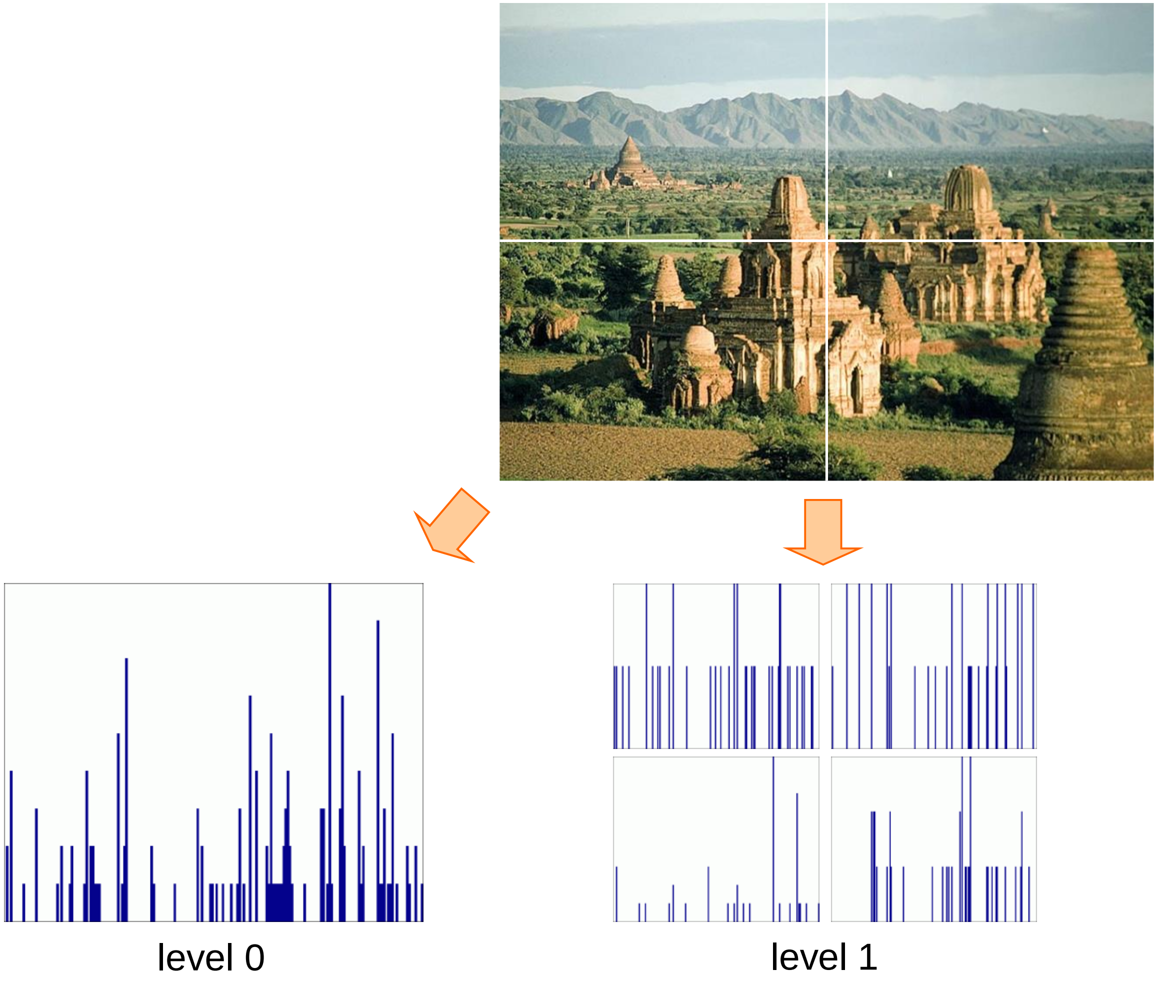


level 0

Lazebnik, Schmid & Ponce (CVPR 2006)

Spatial pyramid representation

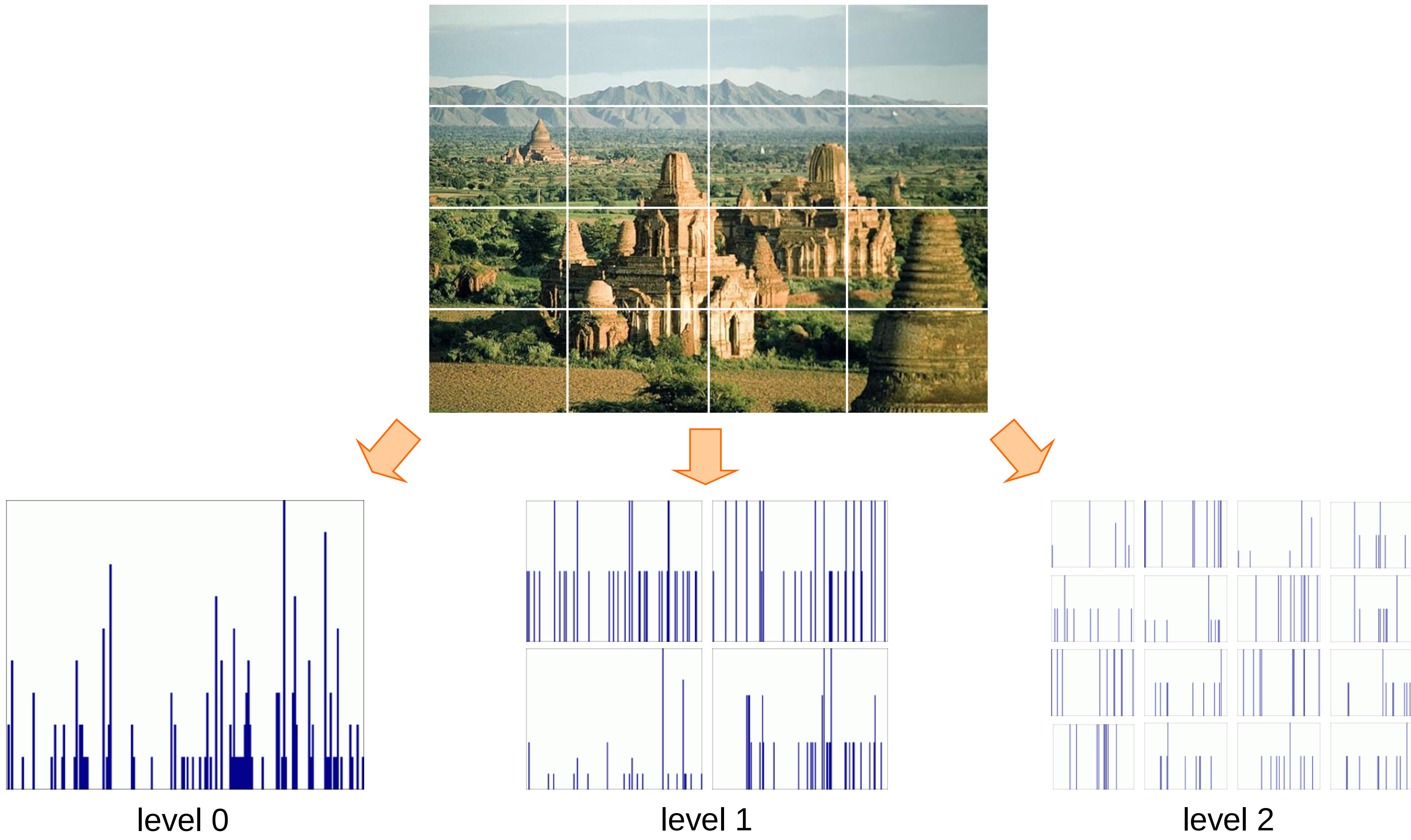
- Extension of a bag of features
- Locally orderless representation at several levels of resolution



Lazebnik, Schmid & Ponce (CVPR 2006)

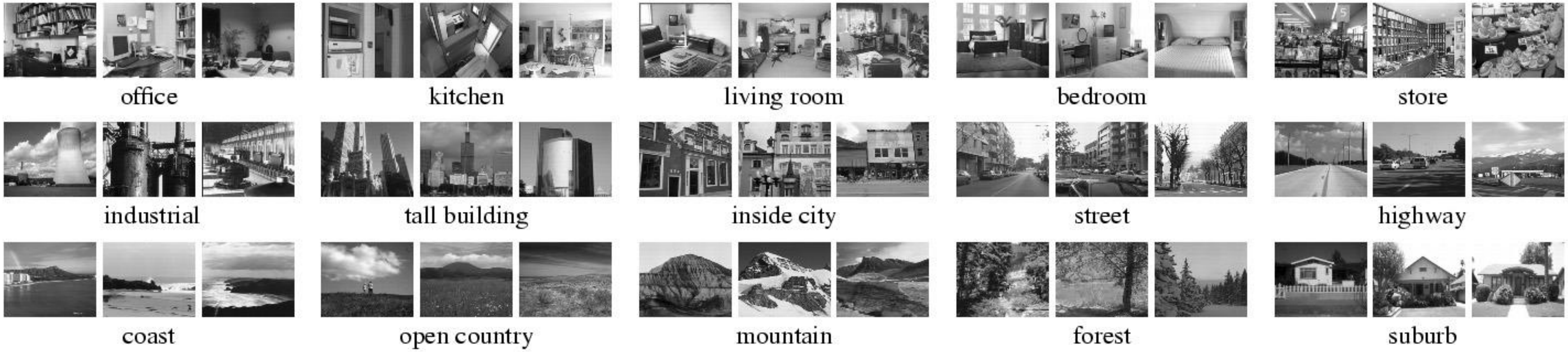
Spatial pyramid representation

- Extension of a bag of features
- Locally orderless representation at several levels of resolution



Lazebnik, Schmid & Ponce (CVPR 2006)

Scene category dataset



Multi-class classification results
(100 training images per class)

	Weak features (vocabulary size: 16)		Strong features (vocabulary size: 200)	
Level	Single-level	Pyramid	Single-level	Pyramid
0 (1×1)	45.3 ±0.5		72.2 ±0.6	
1 (2×2)	53.6 ±0.3	56.2 ±0.6	77.9 ±0.6	79.0 ±0.5
2 (4×4)	61.7 ±0.6	64.7 ±0.7	79.4 ±0.3	**81.1** ±0.3
3 (8×8)	63.3 ±0.8	**66.8** ±0.6	77.2 ±0.4	80.7 ±0.3

Recognition Issues

How to summarize the content of an entire image?
How to gauge overall similarity?

How large should the vocabulary be?
How to perform quantization efficiently?

How to score the retrieval results?

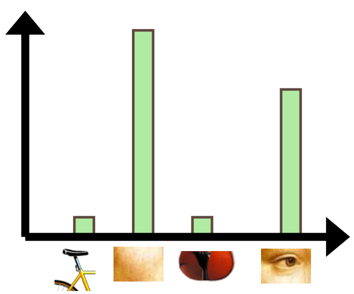
How might we add more spatial verification?

Comparing bags of words

Compute cosine similarity (normalized scalar (dot) product) between their occurrence counts, then rank and pick smallest. *Nearest neighbor* search for similar images.

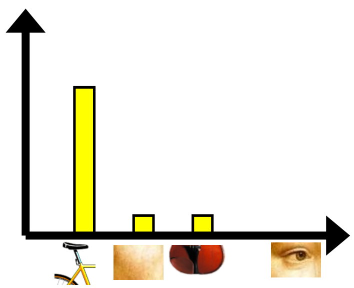
Database image

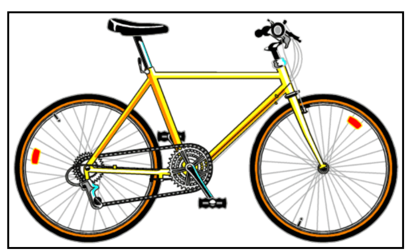
$$\vec{d}_j = [1 \quad 8 \quad 1 \quad 4]$$



Query

$$\vec{q} = [5 \quad 1 \quad 1 \quad 0]$$





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

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

for vocabulary of *V* words

Kristen Grauman

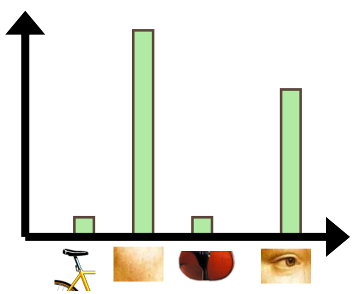
Comparing bags of words

Why might we use cosine similarity here?

What 'intuitive' effect does this provide?

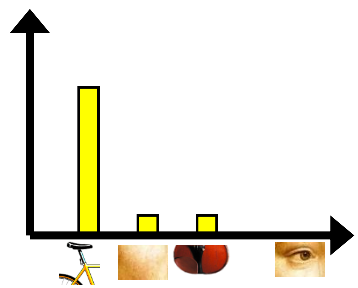
Database image

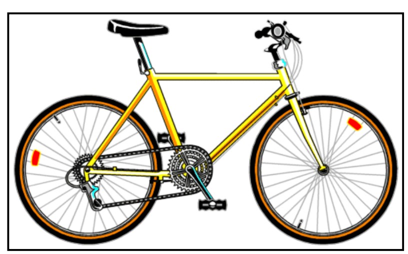
$$\vec{d}_j = [1 \quad 8 \quad 1 \quad 4]$$



Query

$$\vec{q} = [5 \quad 1 \quad 1 \quad 0]$$





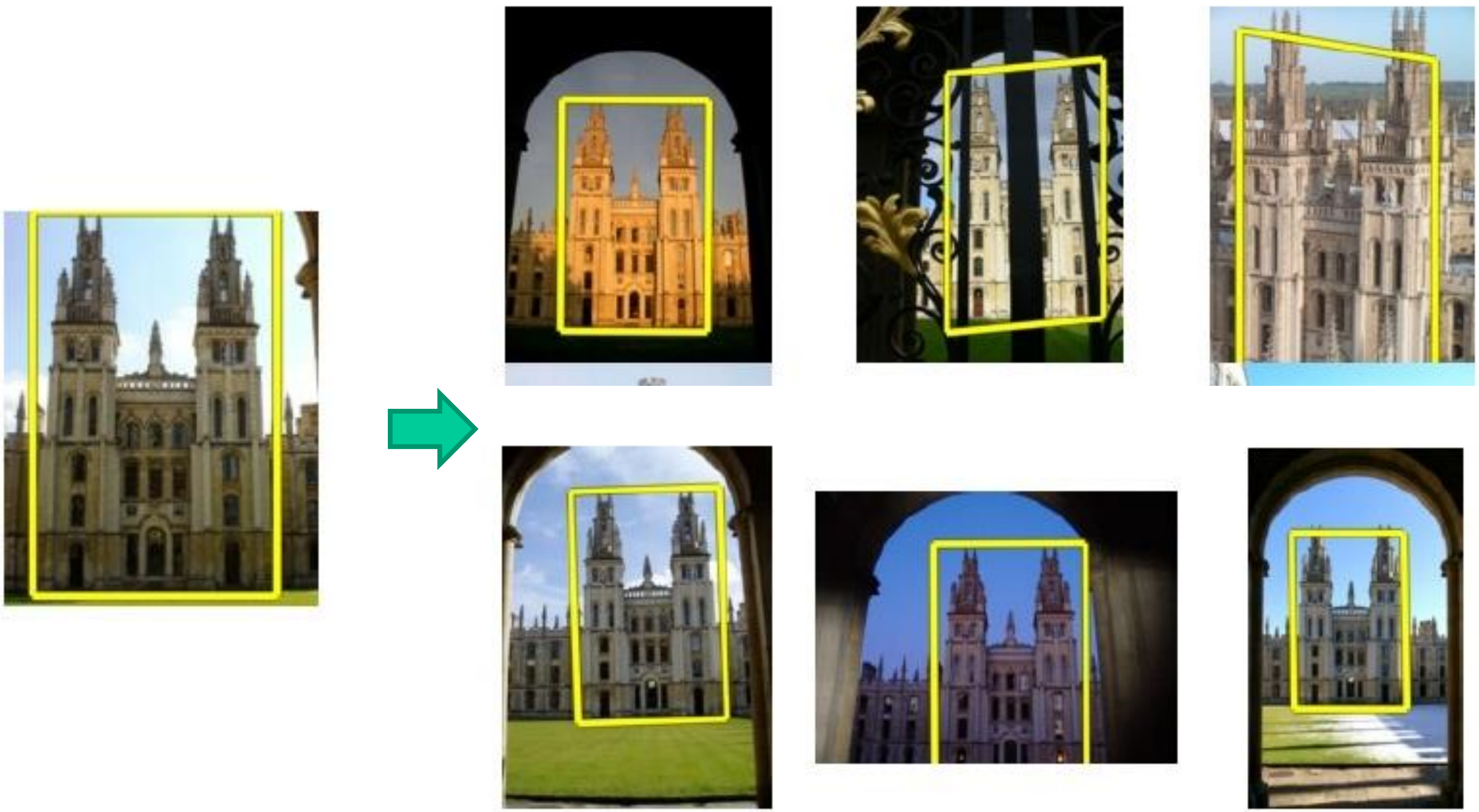
$$sim(d_j, q) = \frac{\langle d_j, q \rangle}{\|d_j\| \|q\|}$$

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

for vocabulary of *V* words

Kristen Grauman

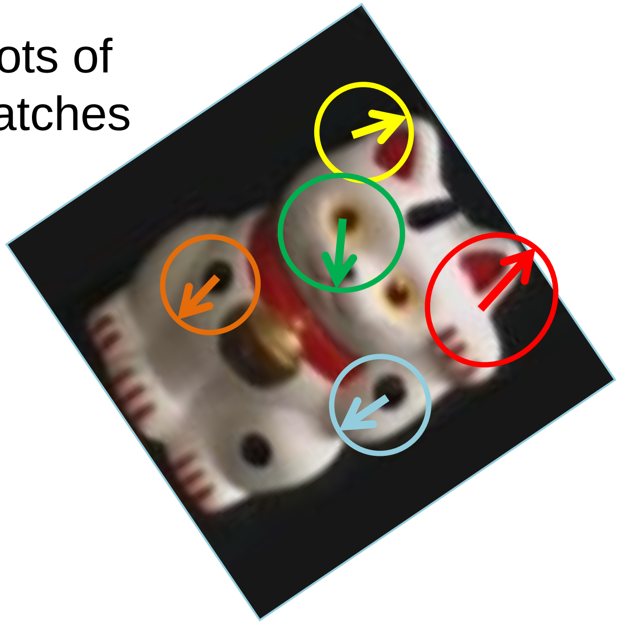
How can we quickly find images in a large database that match a given image region?



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!

Fast lookup: inverted 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.

Kristen Grauman

Build Inverted Index from Database

Database images

Image #1

Image #2

Image #3

⋮

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

…

Kristen Grauman

Query Inverted Index

New query image

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

…

Candidate matches

Image #1

Image #2

Kristen Grauman

Query Inverted Index

New query image

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

…

Candidate matches

Image #1

Image #2

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

Kristen Grauman

Inverted index

Key requirement: *sparsity*.

If most images contain most words, then we’re not better off than exhaustive search.

- Exhaustive search would mean comparing the visual word distribution of a query versus every page.

Recognition Issues

How to summarize the content of an entire image? And gauge overall similarity?

How large should the vocabulary be? How to perform quantization (clustering) efficiently?

How to score the retrieval results?

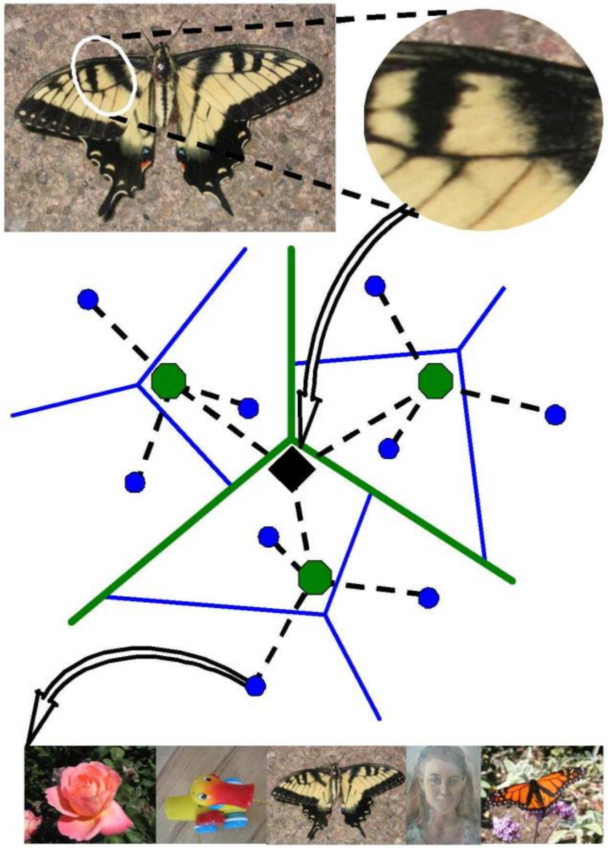
How might we add more spatial verification?

Following slides by David Nister (CVPR 2006)

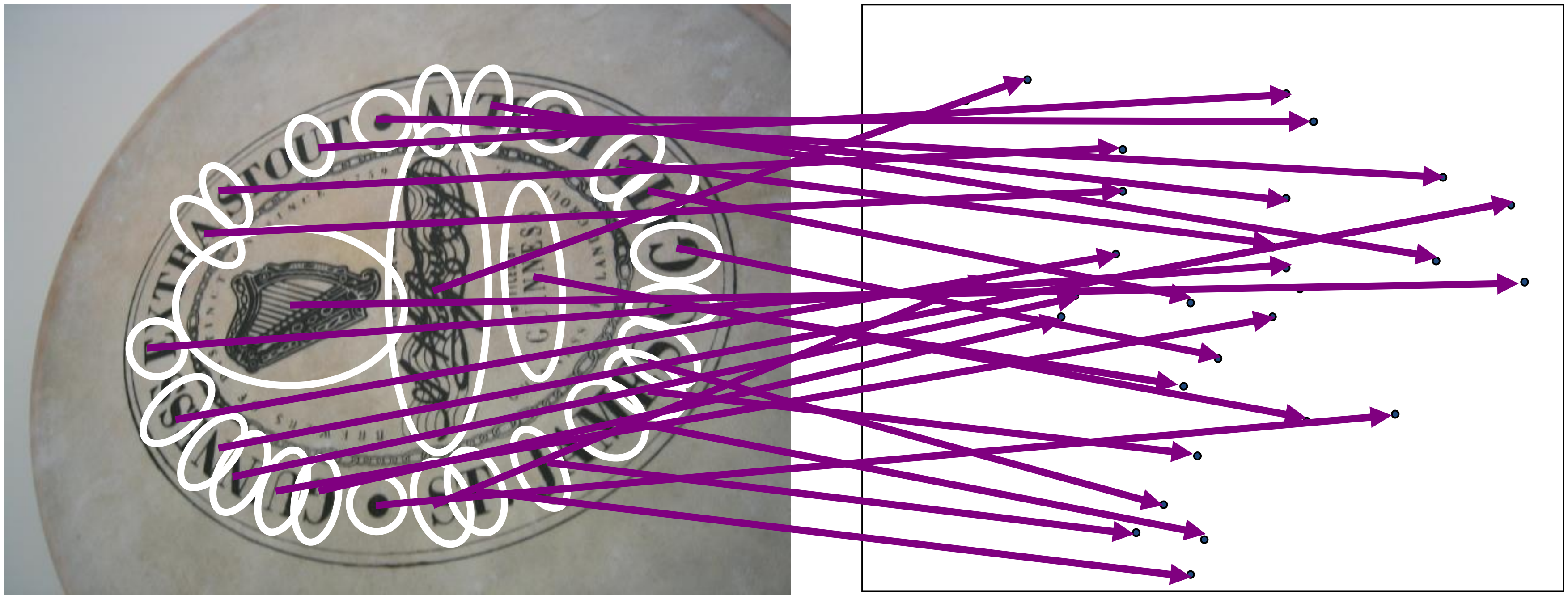
Visual vocabularies: Issues

- How to choose vocabulary size?
  - Too small: visual words not representative of all patches
  - Too large: quantization artifacts, overfitting

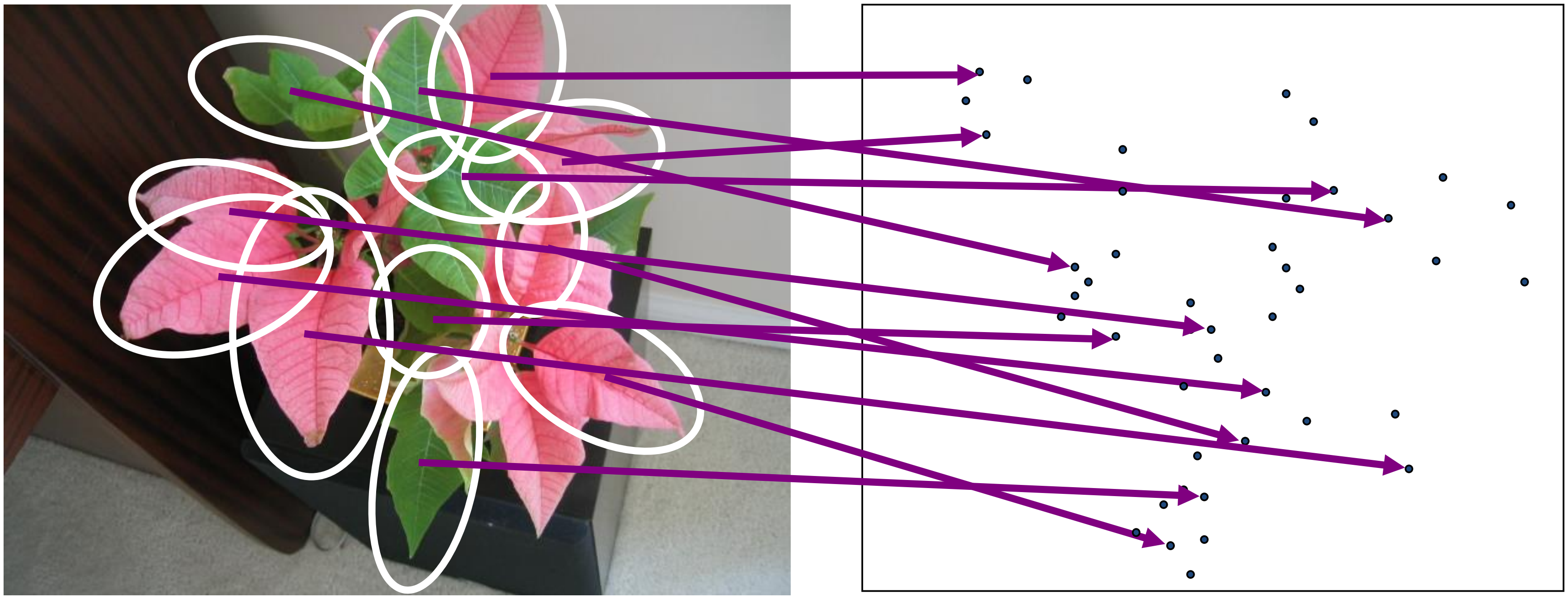
- Computational efficiency
  - Vocabulary trees
    (Nister & Stewenius, 2006)



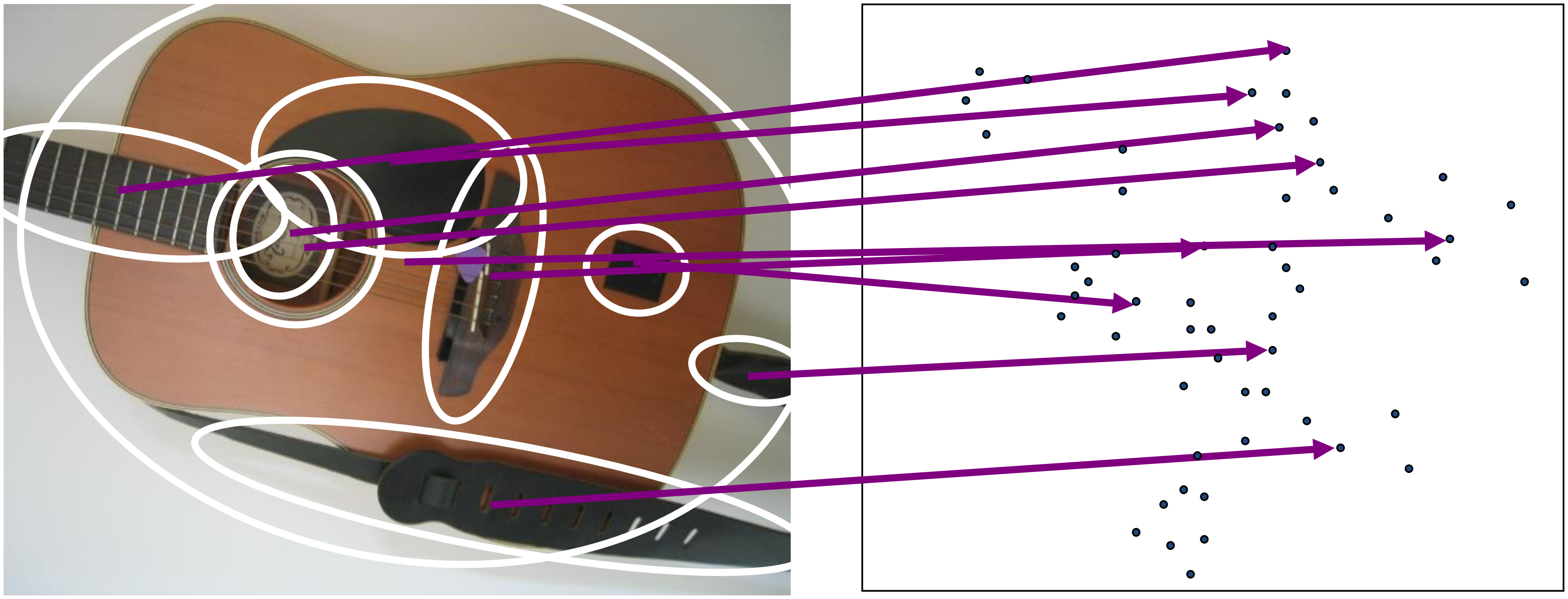
Training the vocabulary tree



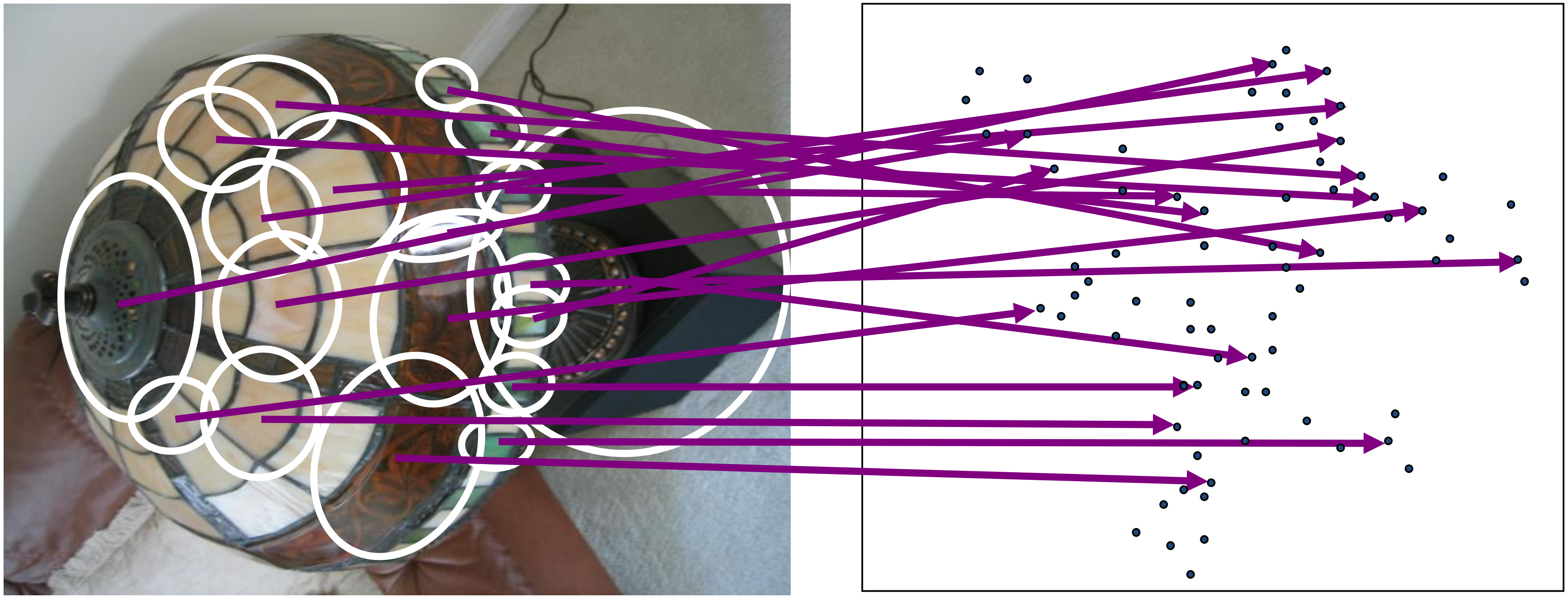
Training the vocabulary tree



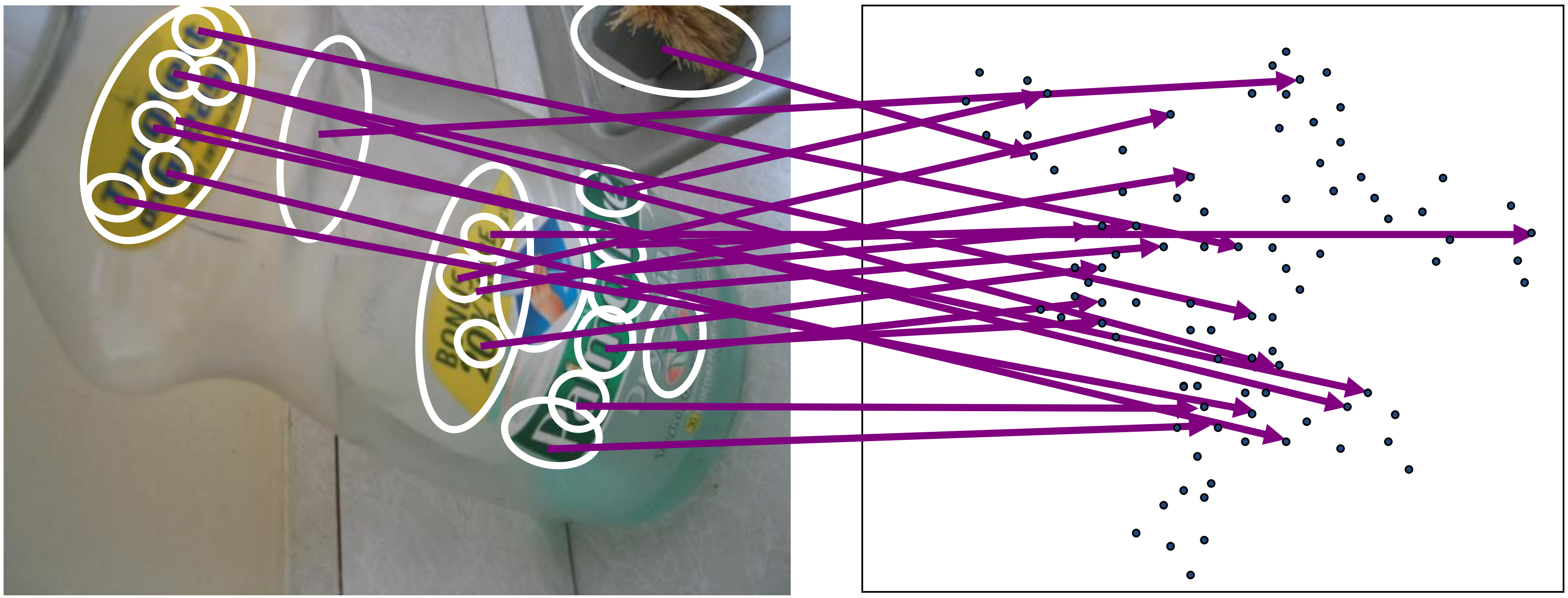
Training the vocabulary tree



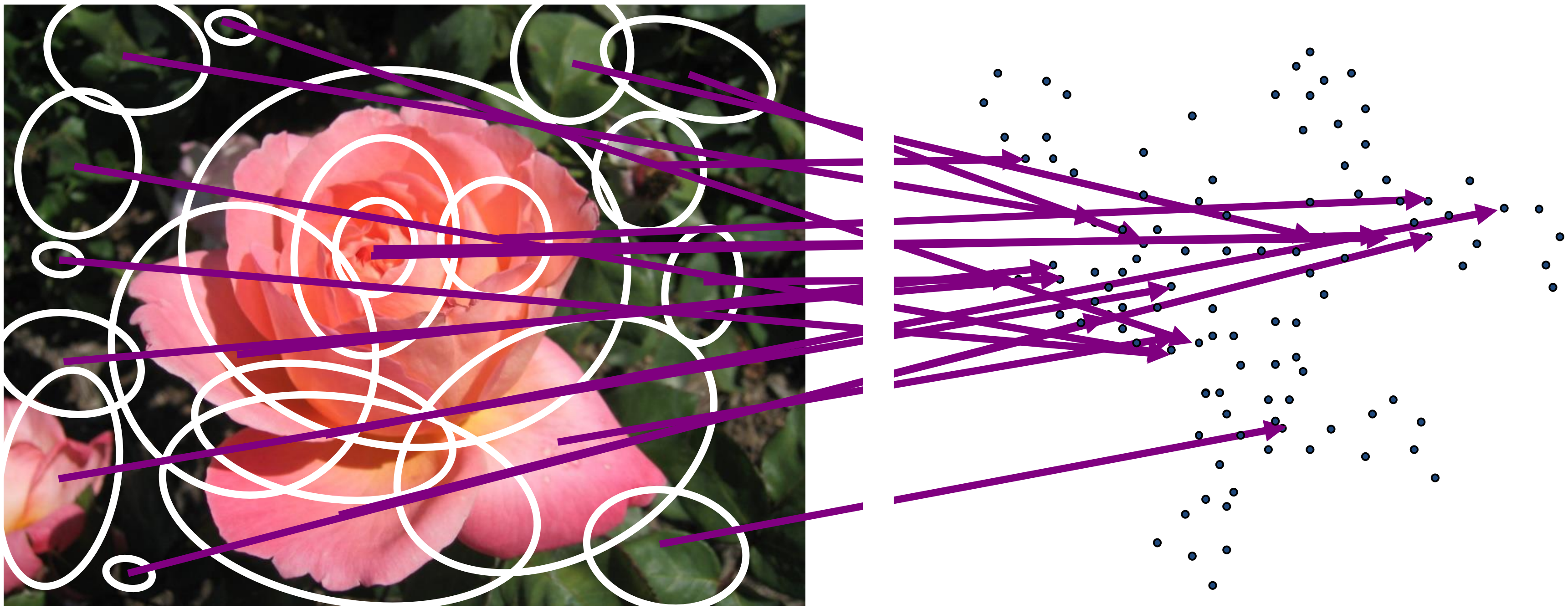
Training the vocabulary tree

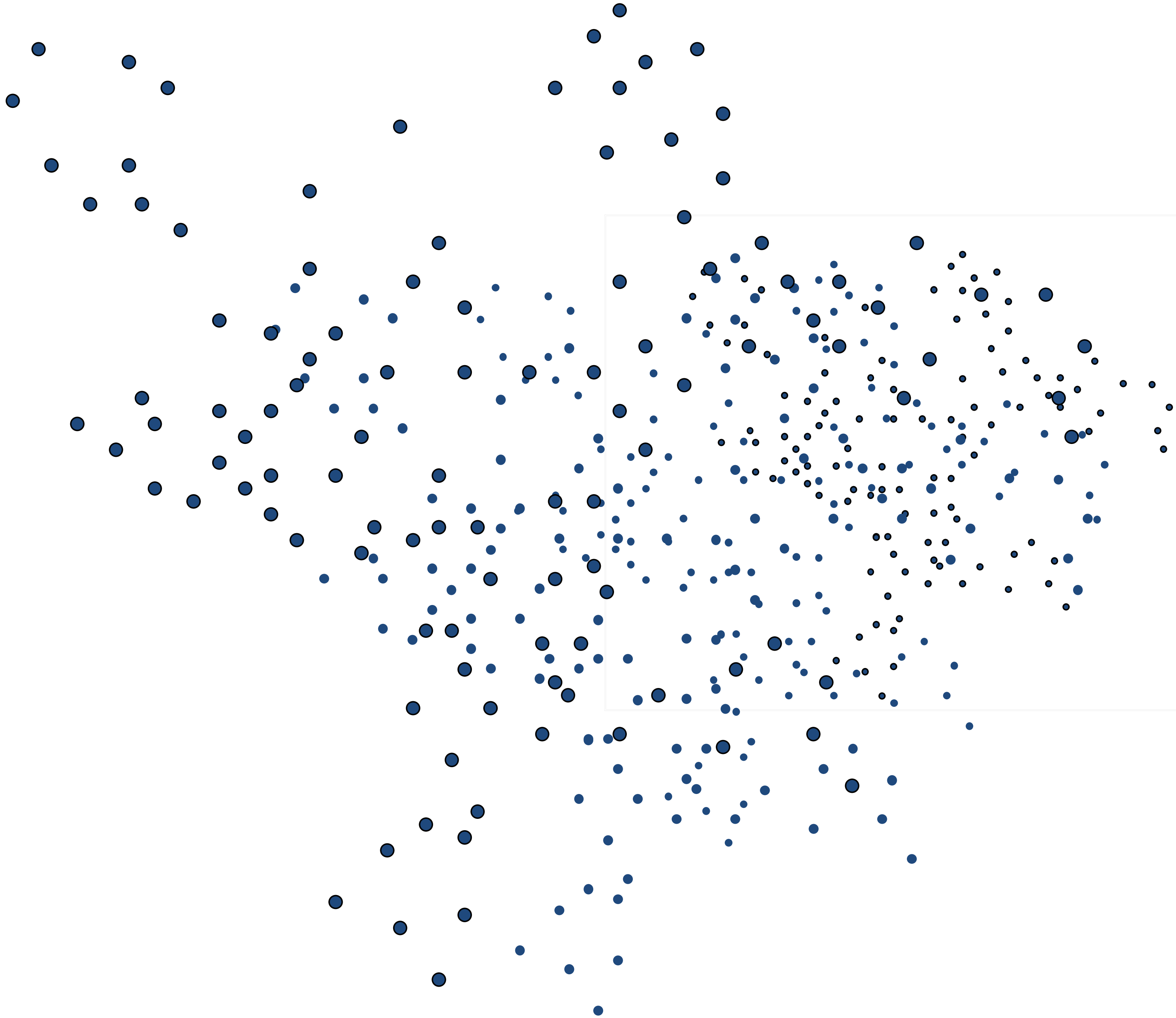


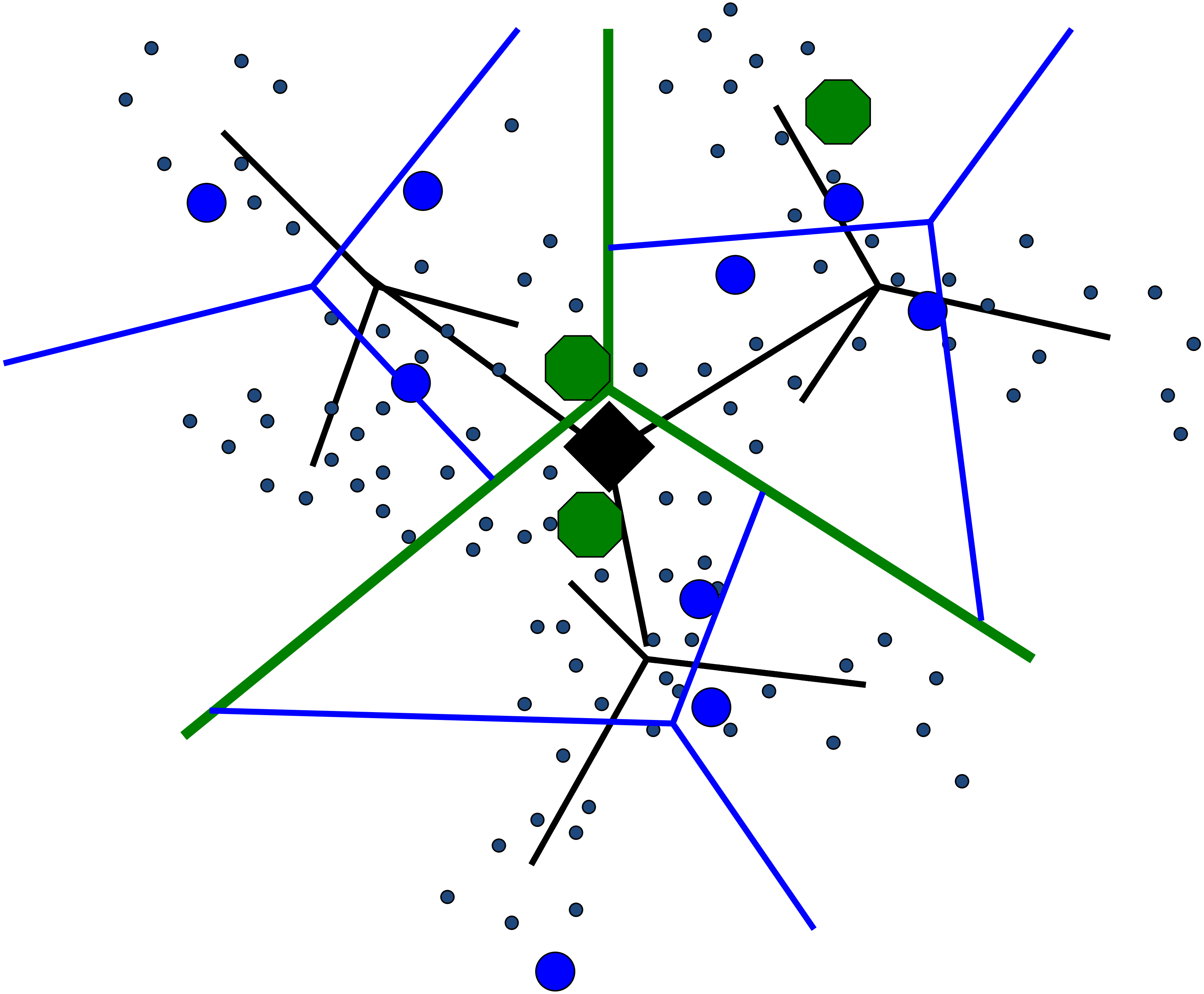
Training the vocabulary tree

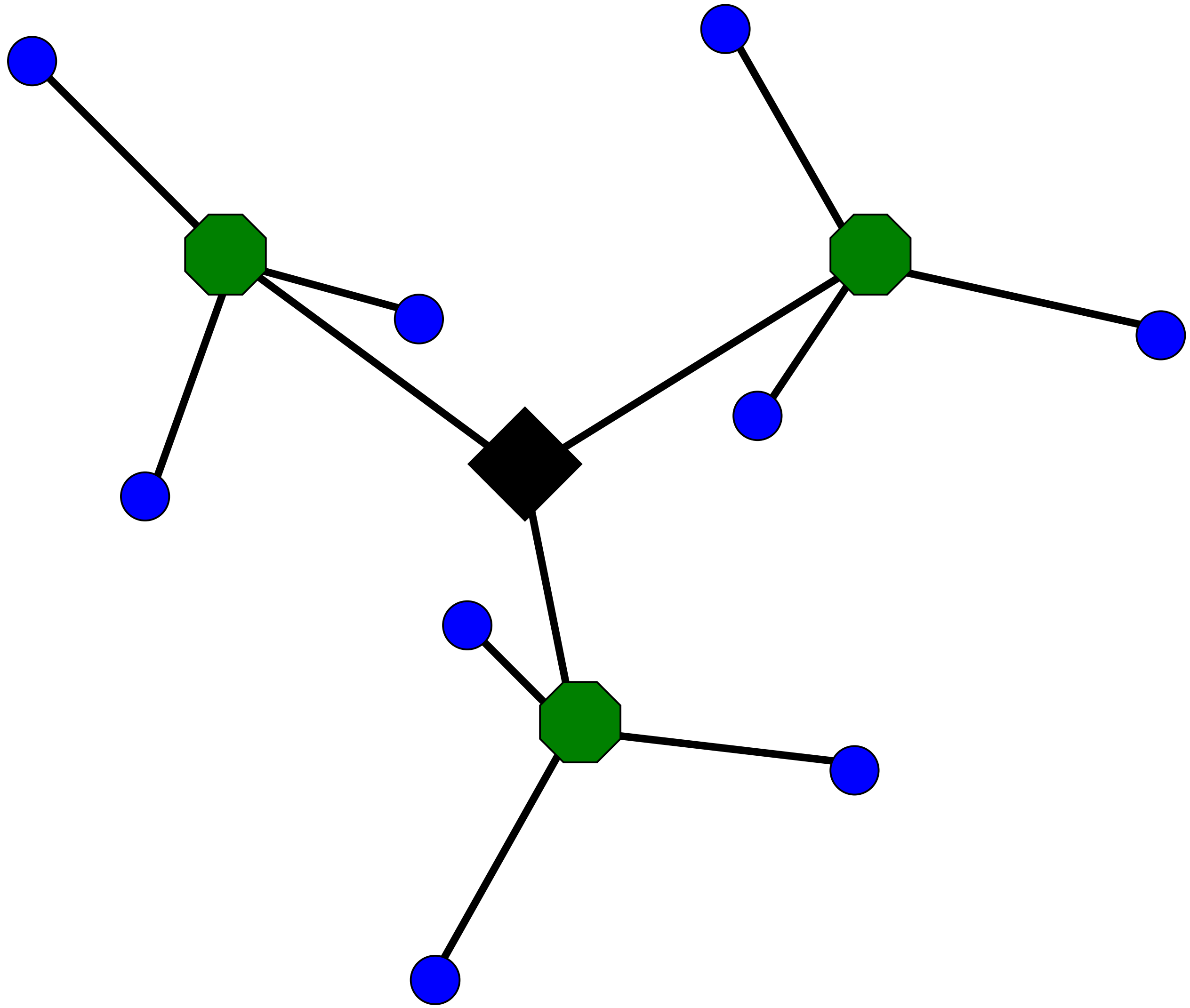


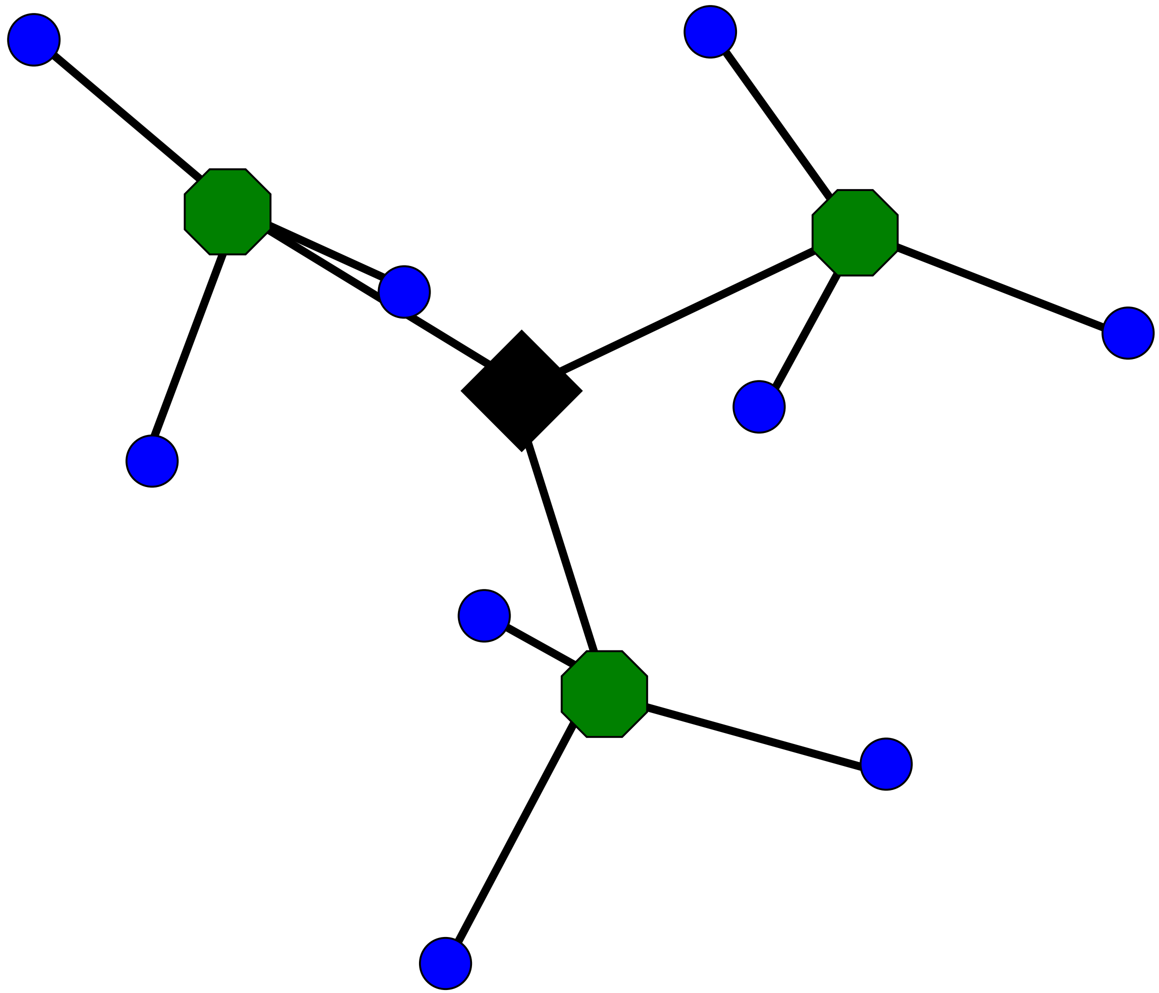
Training the vocabulary tree

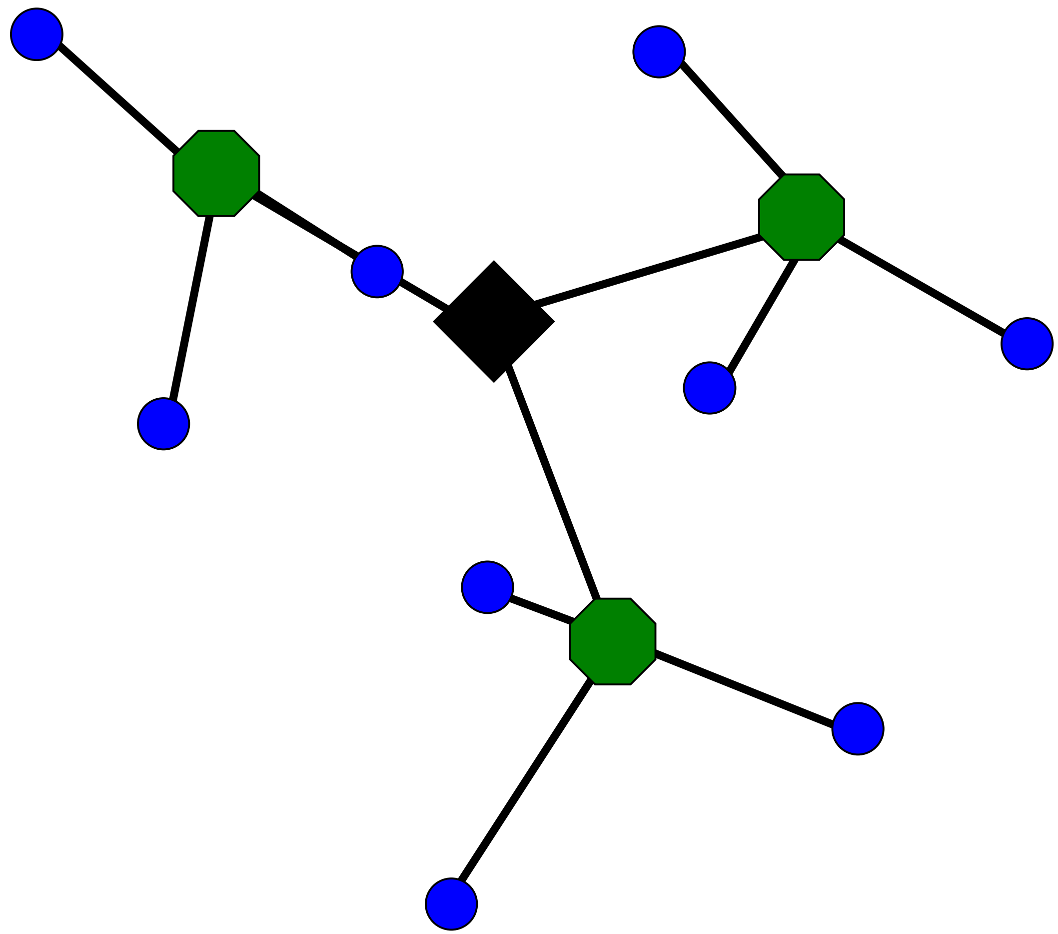


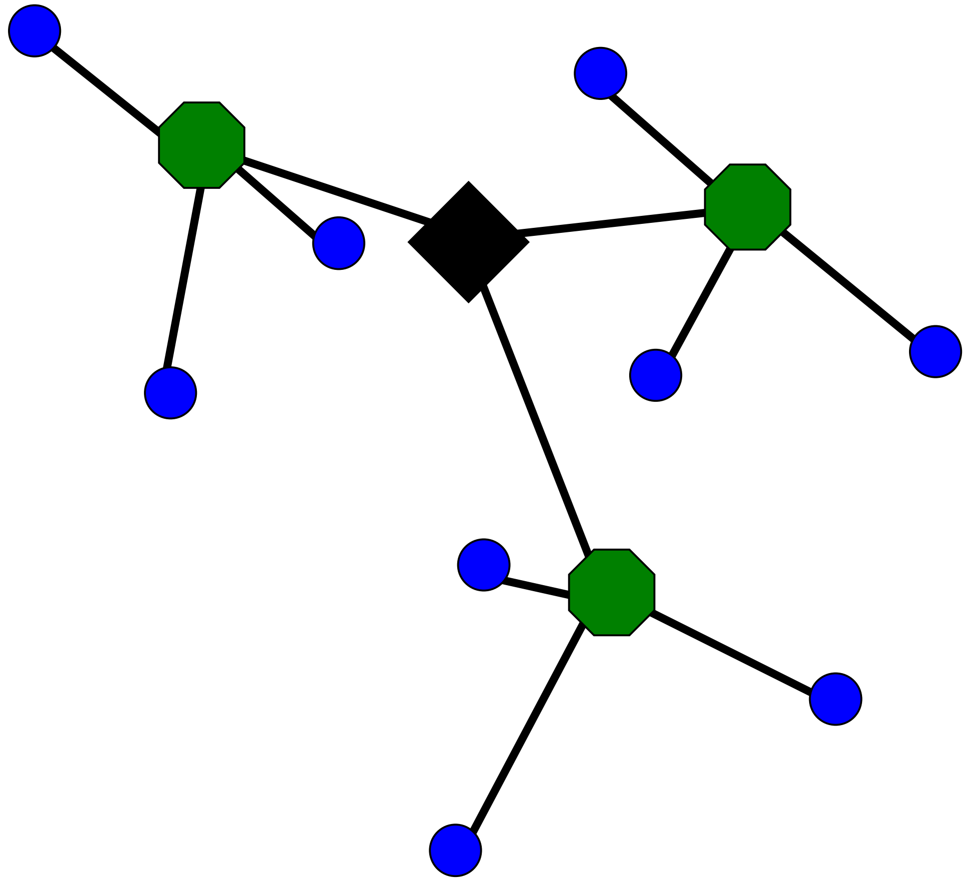


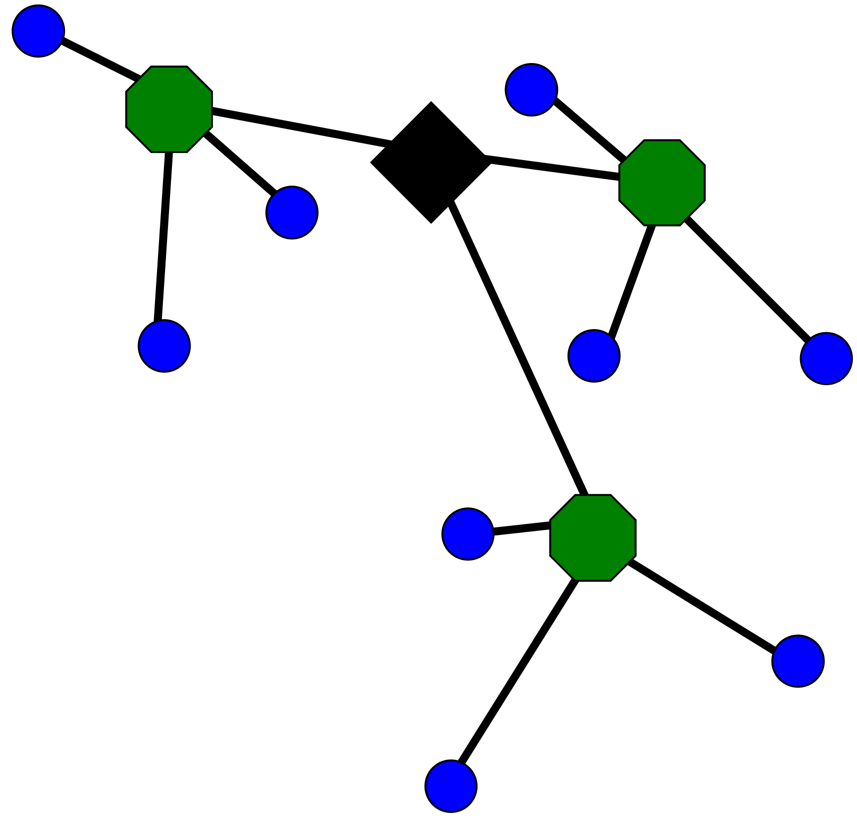


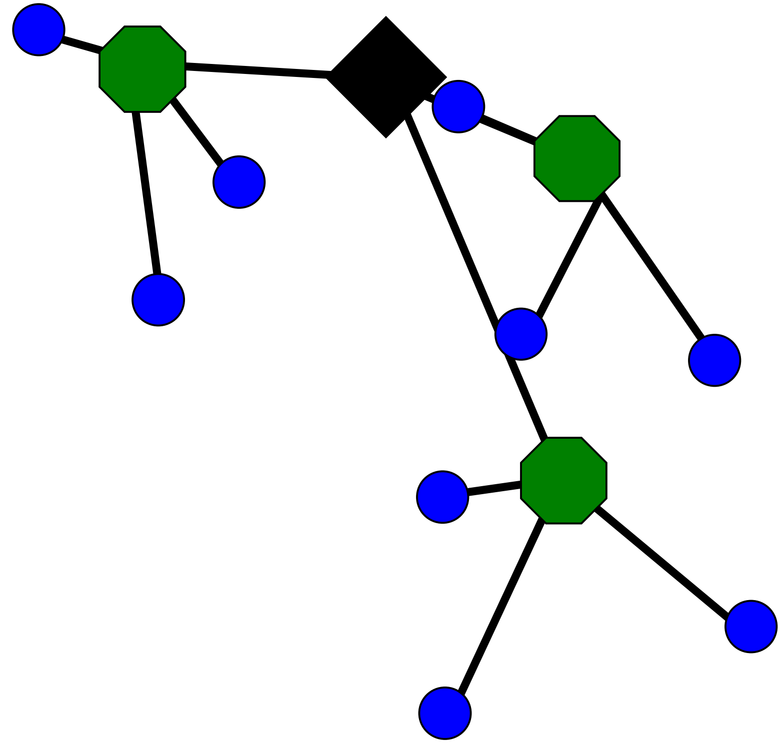


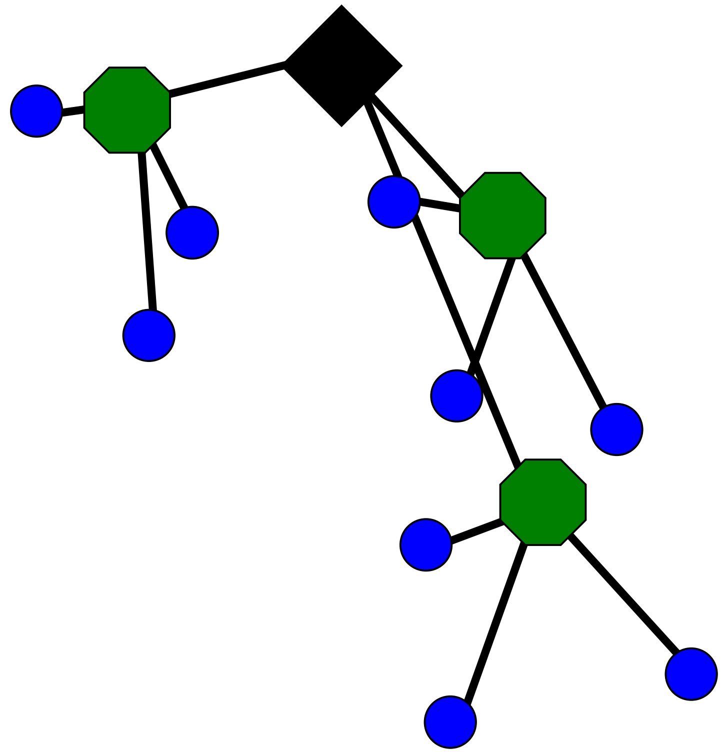


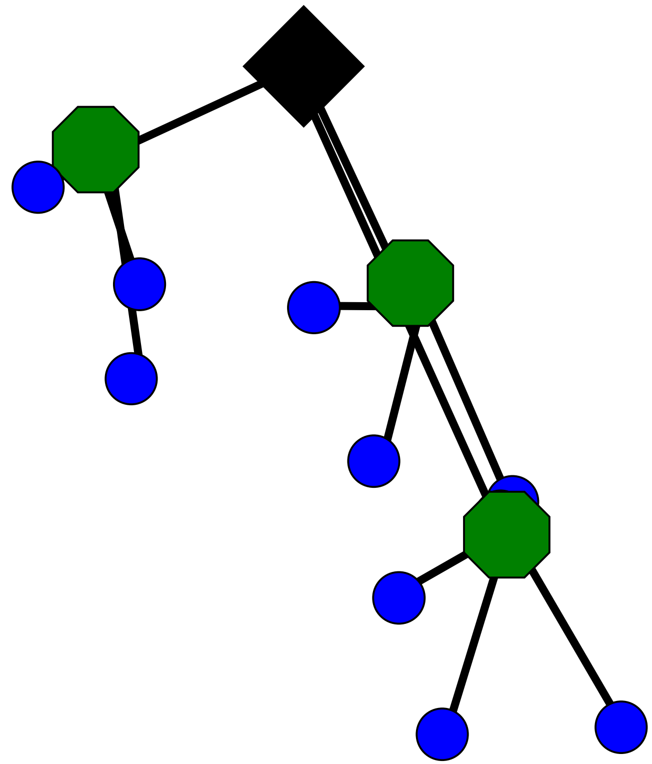


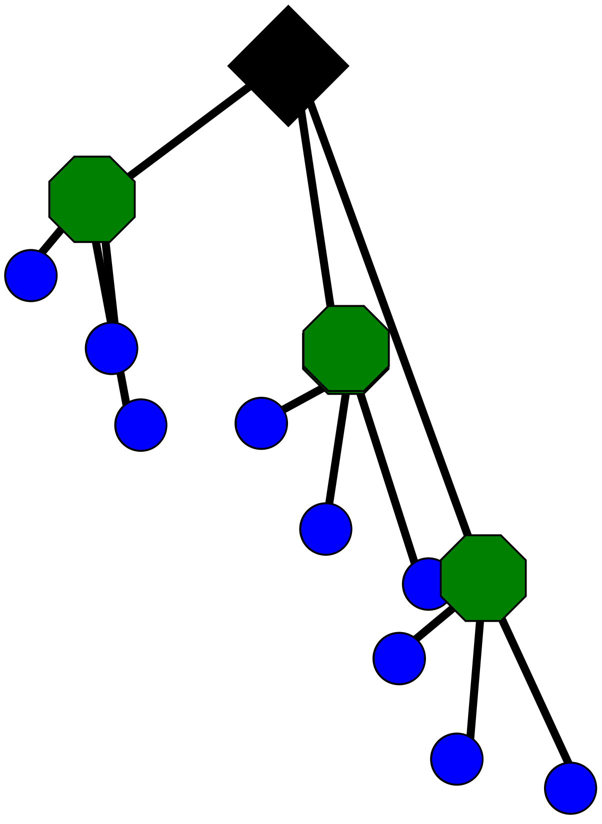




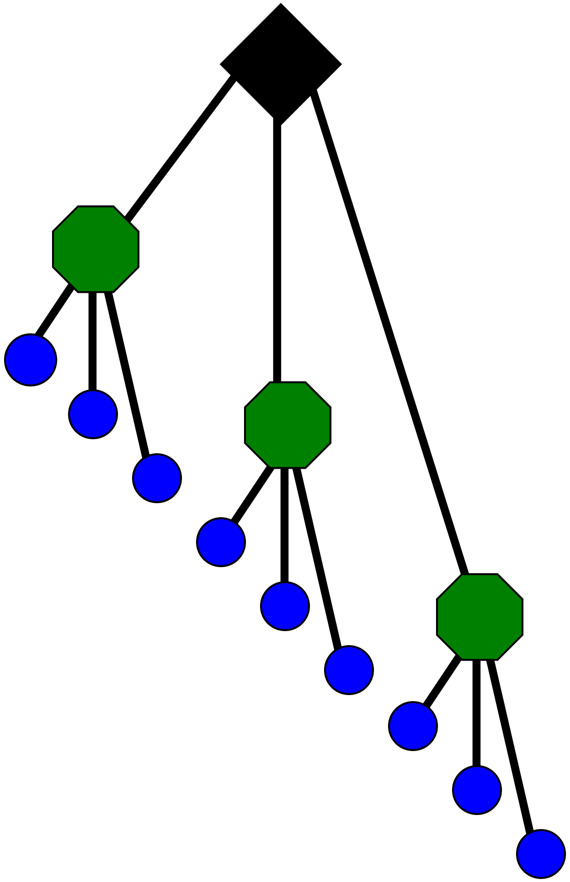




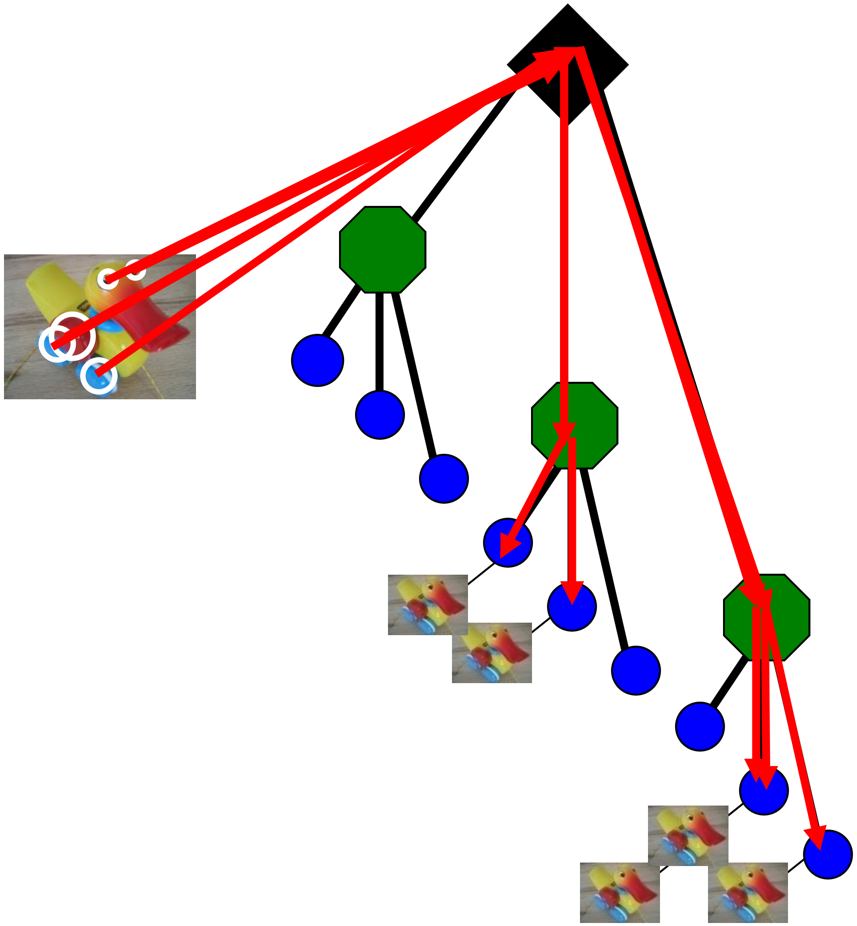


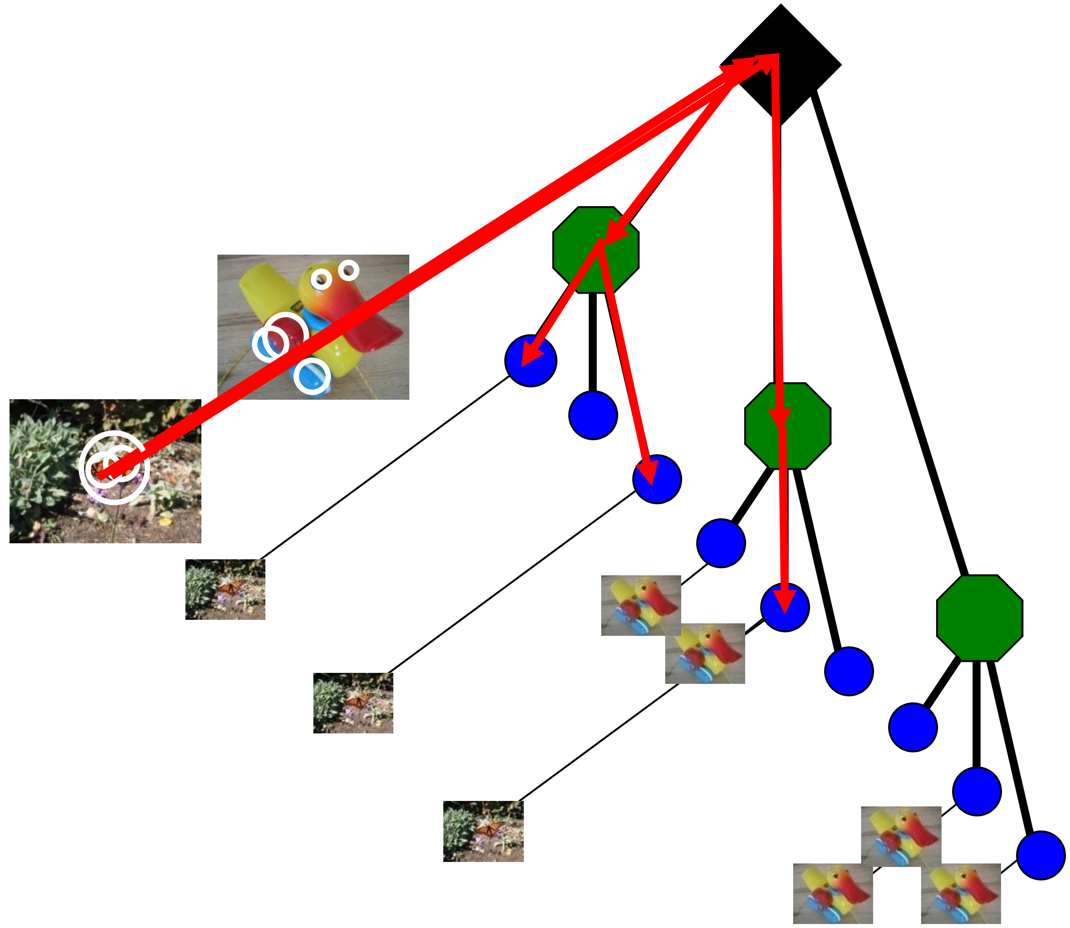


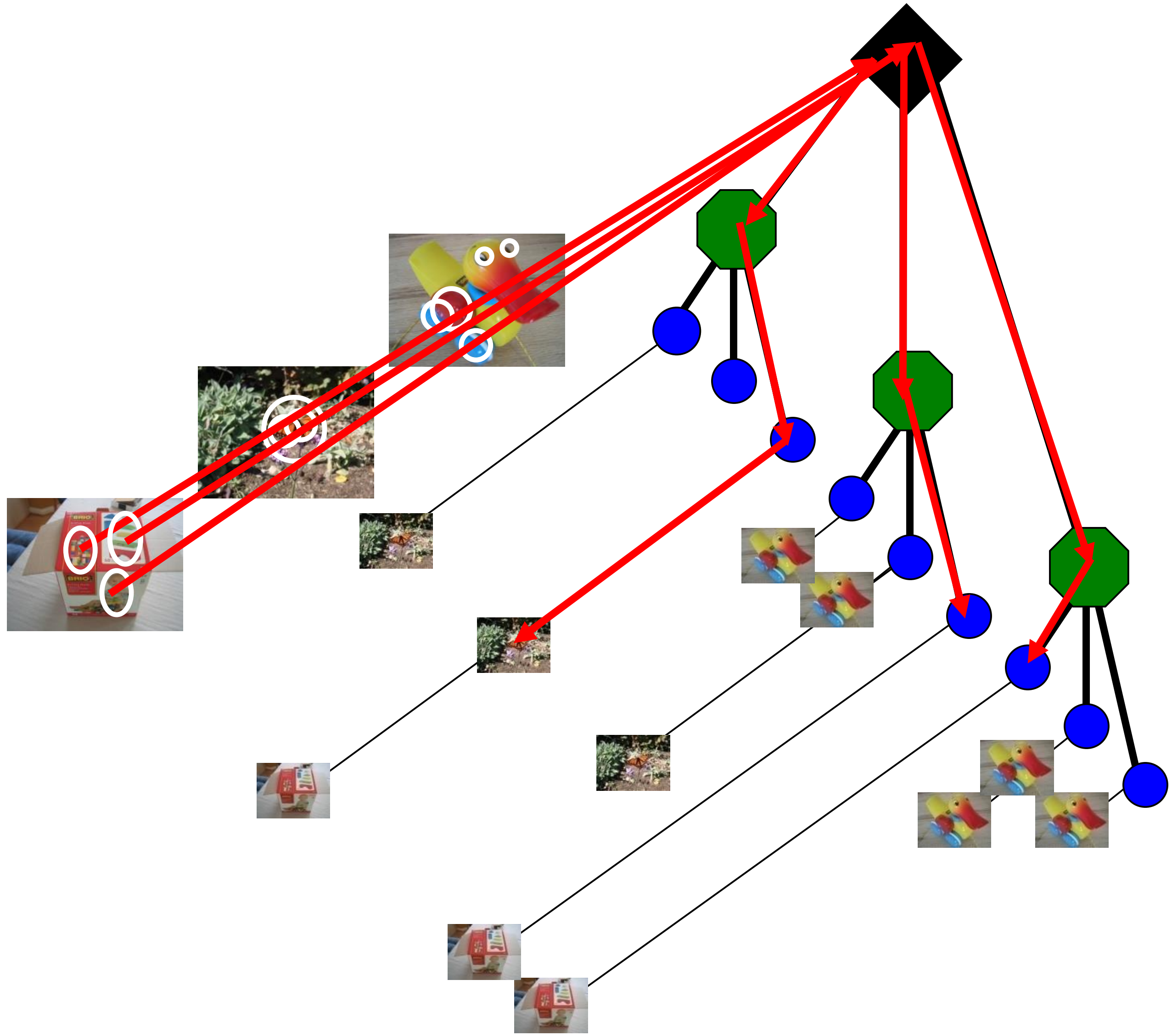
Vocabulary tree built recursively



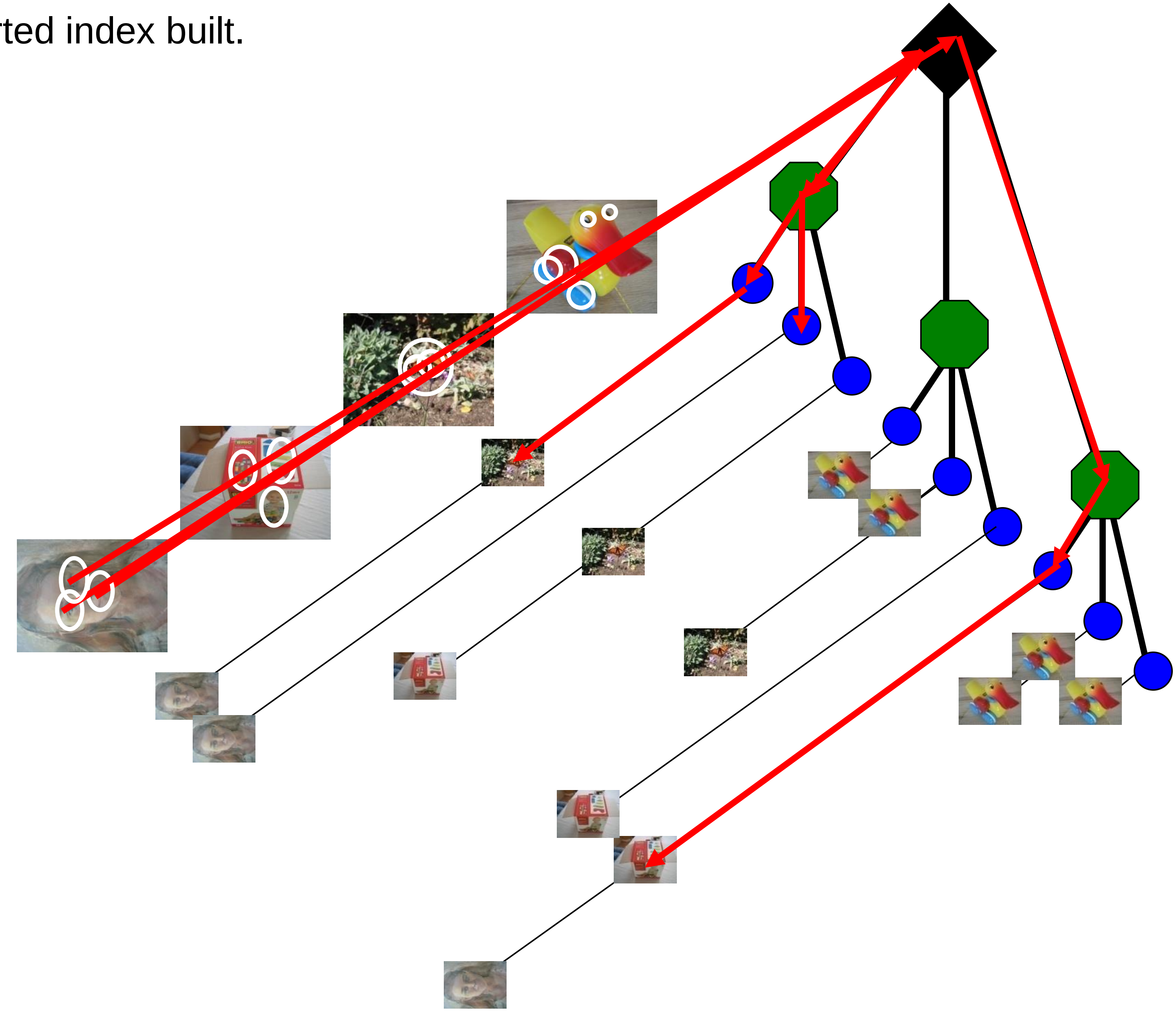
Each leaf has inverted index

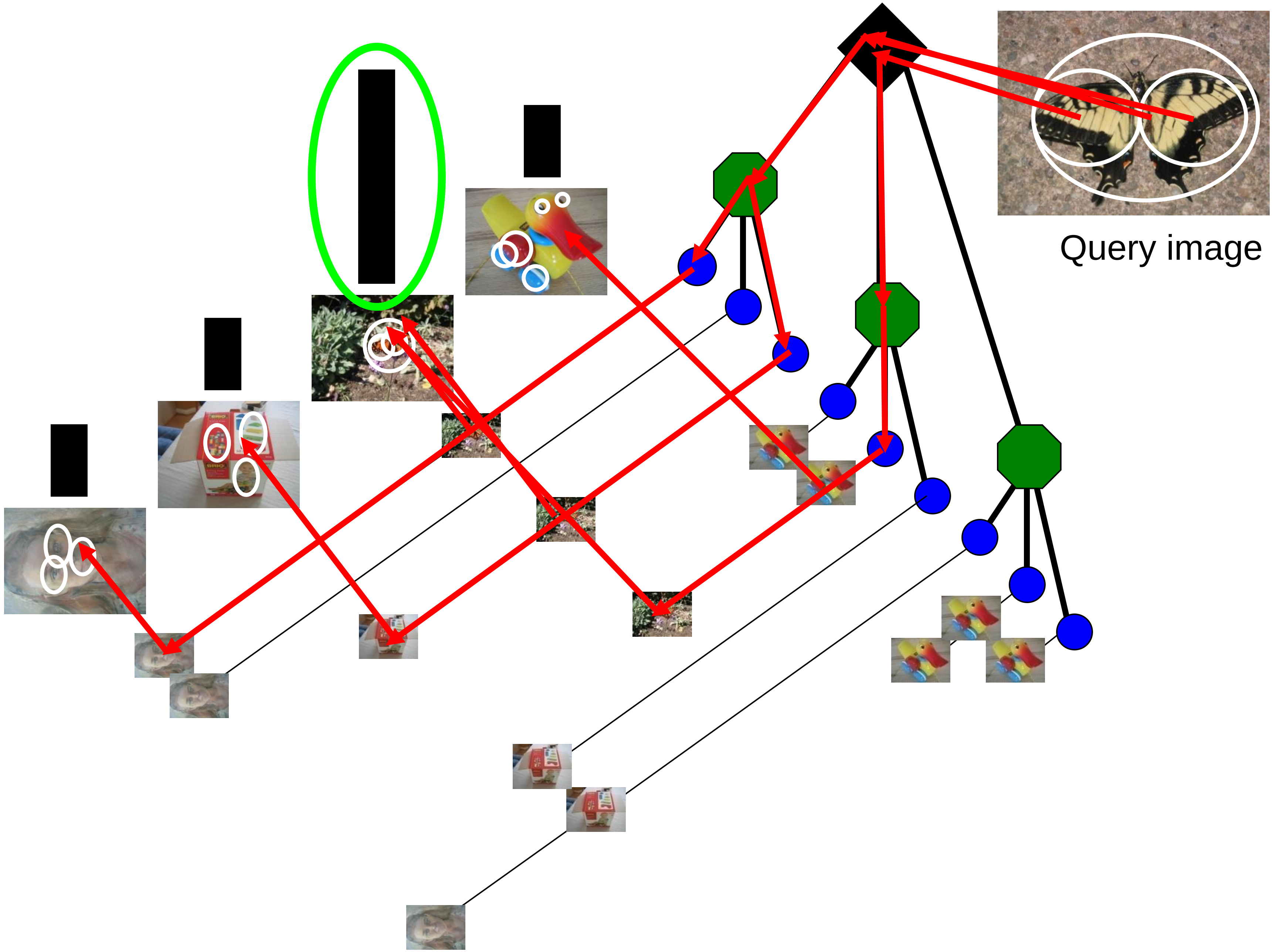




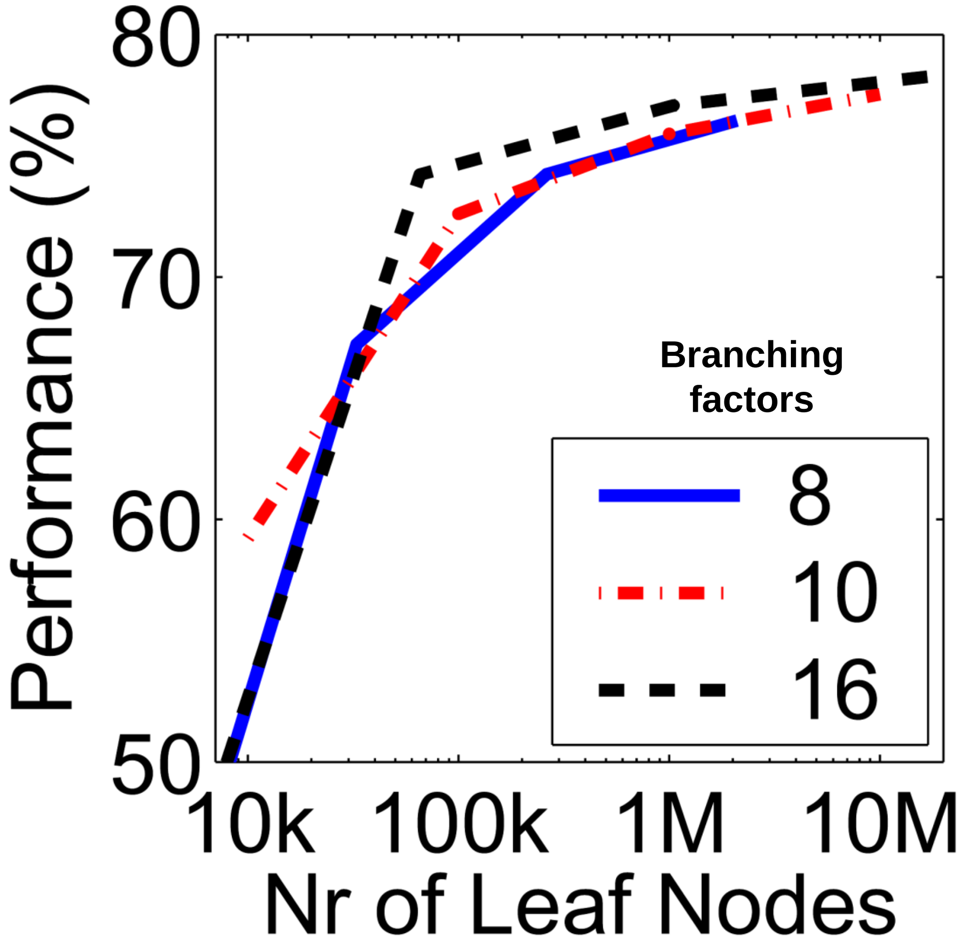


Inverted index built.

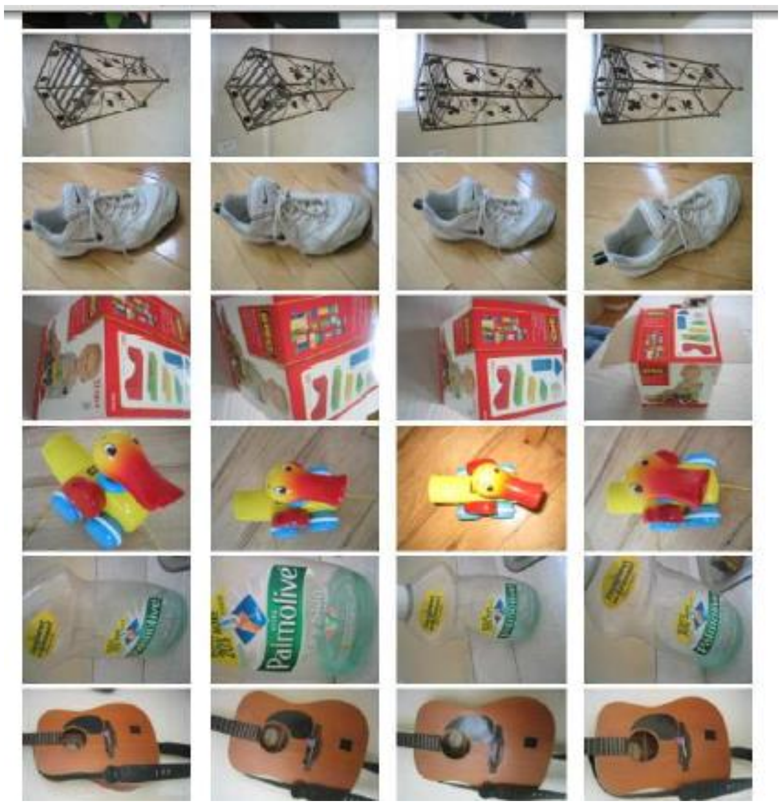



Query image

Vocabulary size



Recognition with 6347 images

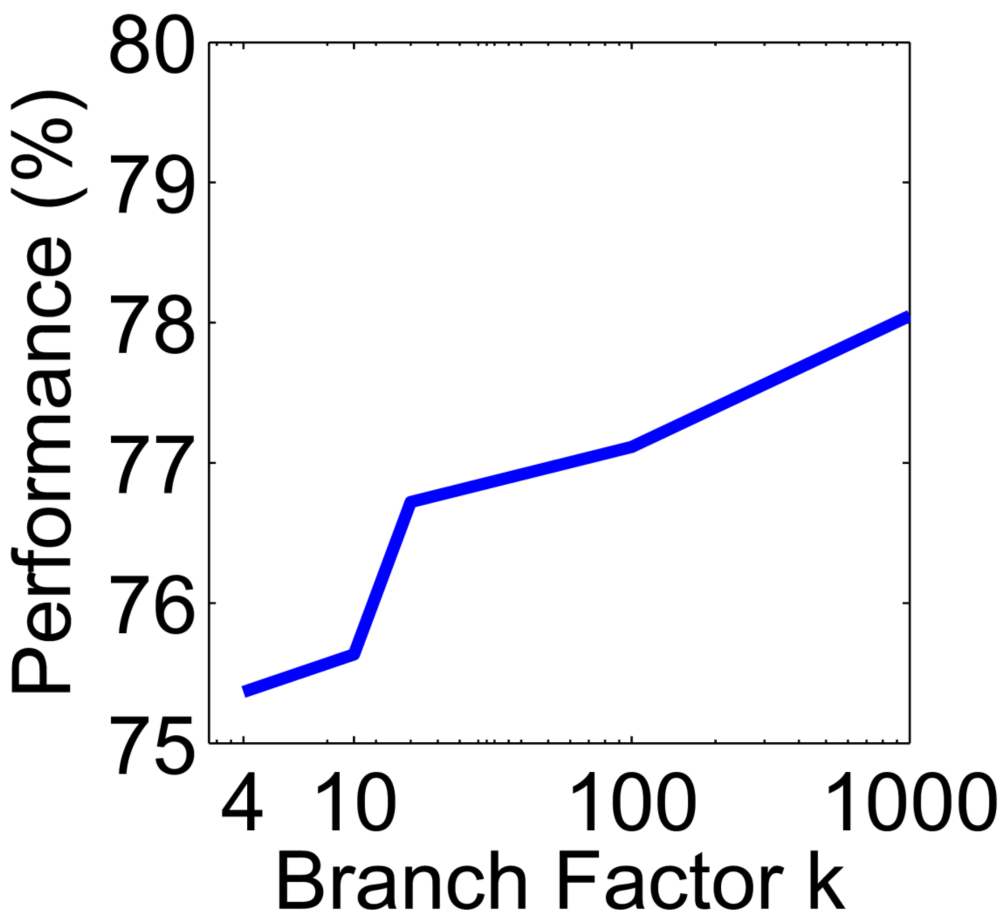


Nister & Stewenius, CVPR 2006

Influence on performance, sparsity

Kristen Grauman

Higher branch factor works better
(but slower)



(2006) 110,000,000 images in 5.8 Seconds

On a 50k image index

UK

Center for Visualization & Virtual Environments

David Nister

David Nister

David Nister

David Nister

Recognition 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?

How to score the retrieval results?

How might we add more spatial verification?

Kristen Grauman

Precision and Recall

True positive (tp) – correct attribution
True negative (tn) – correct rejection

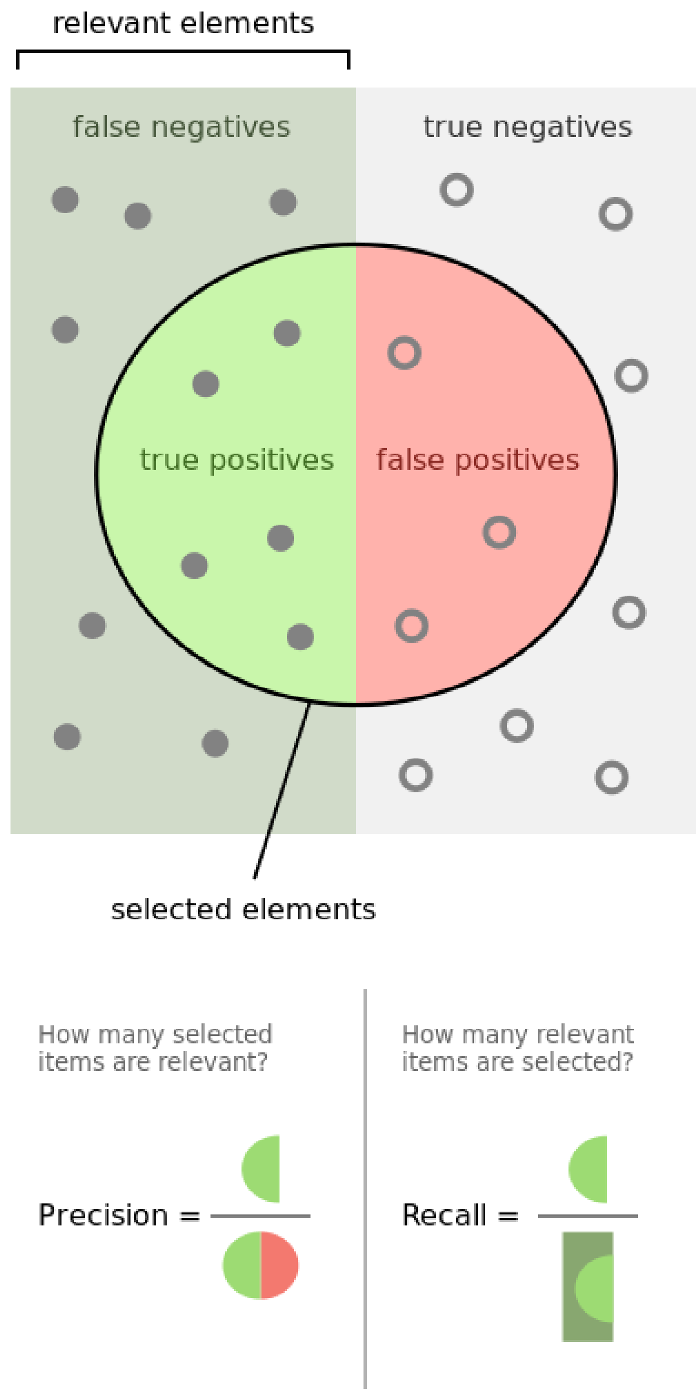
False positive (fp) – incorrect attribution
False negative (fn) – incorrect rejection

$$\text{Precision} = \frac{tp}{tp + fp}$$

Precision = #relevant / #returned

$$\text{Recall} = \frac{tp}{tp + fn}$$

Recall = #relevant / #total relevant

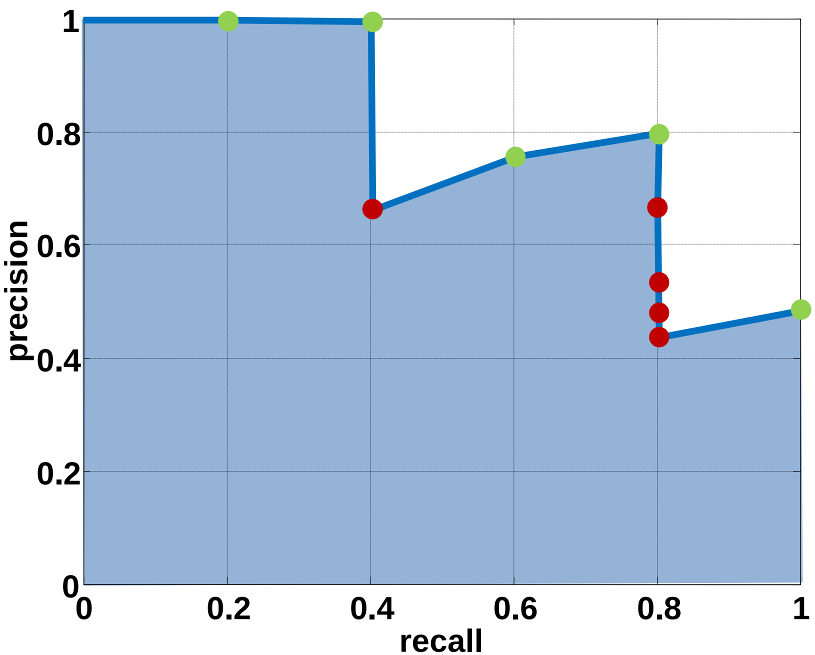


Scoring retrieval quality

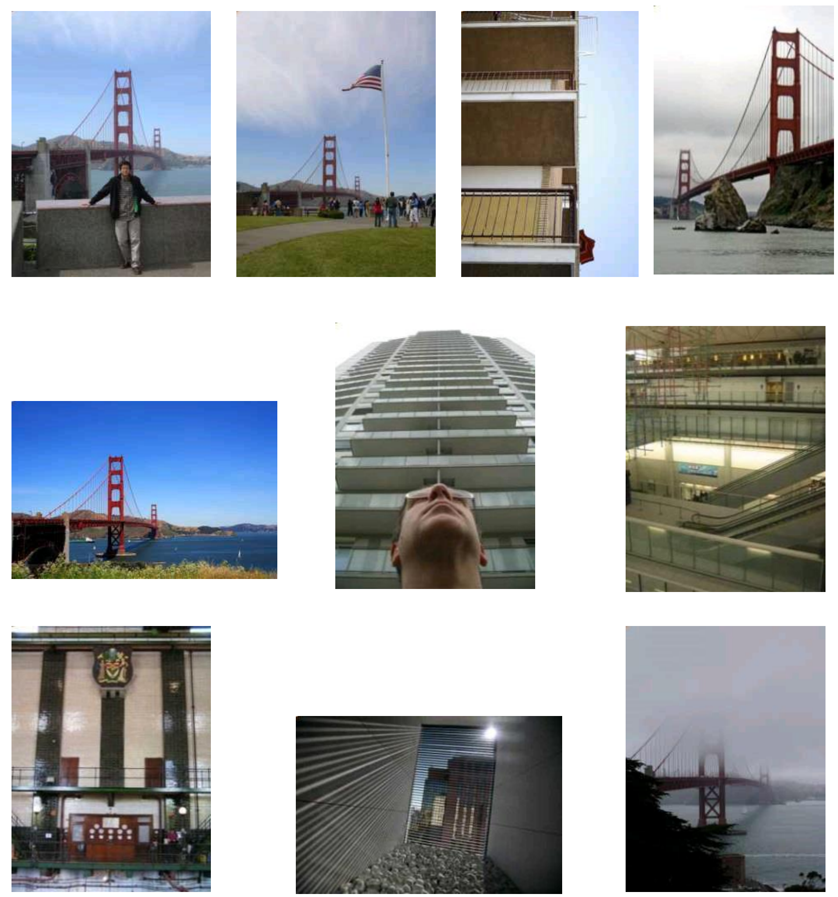
Query

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

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

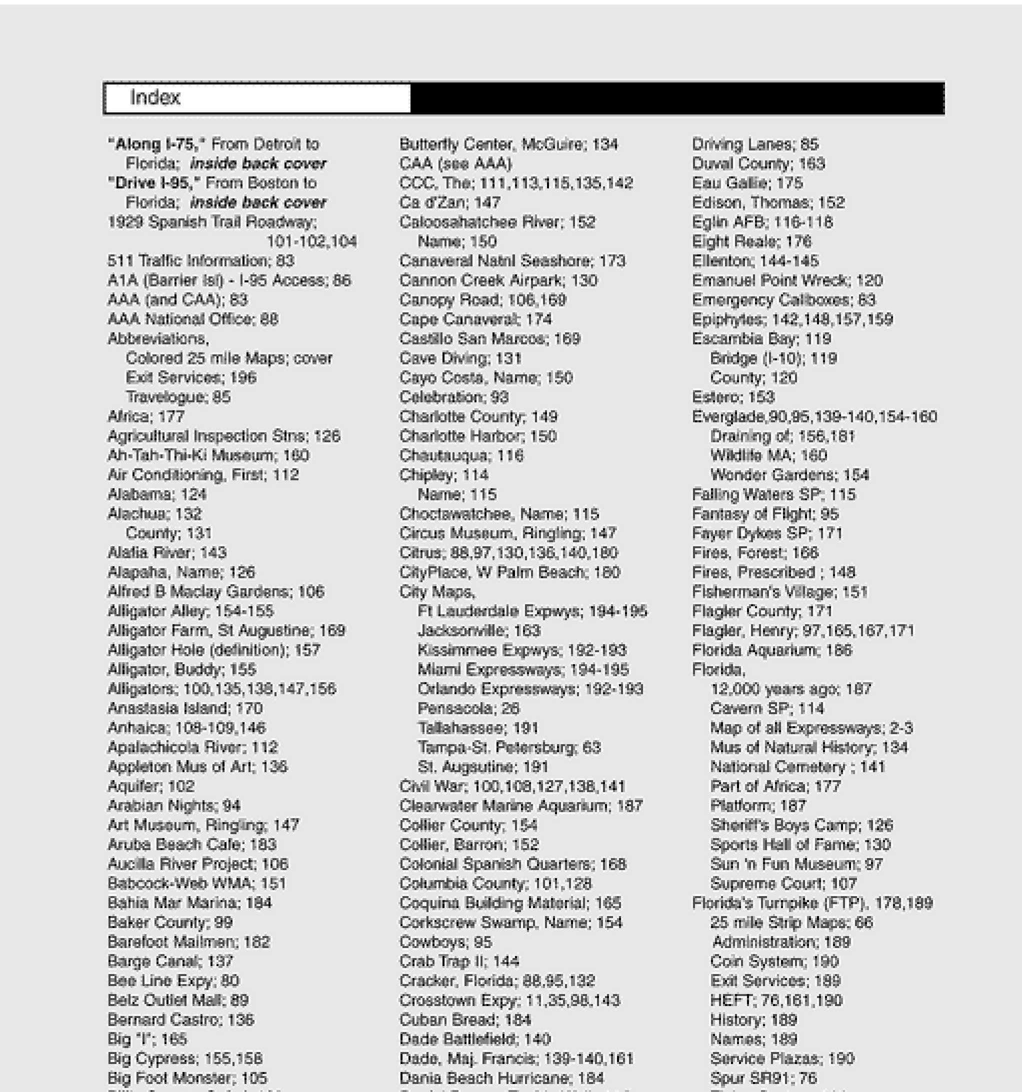


Results (ordered):



[Ondrej Chum]

What else can we borrow from text retrieval?



Index

China is forecasting a trade surplus of $90bn
(£51bn) to $100bn this year, a threefold
increase on 2004's $32bn. The Commerce
Ministry said the surplus would be created by
a predicted 30% … $750bn,
compared w…
$660bn. T…
annoy th…
China's …
deliber…
agrees…
yuan is …
governo…
also need…
demand so …
country. China … e
yuan against the do… nd
permitted it to trade within a narrow … but
the US wants the yuan to be allowed … e
freely. However, Beijing has made it c… t
it will take its time and tread carefully be…
allowing the yuan to rise further in value.

tf-idf weighting

- **T**erm **f**requency – **i**nverse **d**ocument **f**requency
- Describe image 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

Total number of documents in database

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

Number of words in document d

Number of documents word i occurs in, in whole database

Kristen Grauman

Query expansion

Use good retrieved results as new inputs.
Increase recall possibly at the expense of precision.

Example 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.

Good new queries: grass, golf course, par-four hole, putting, etc.

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

Bad new queries: petrol, hatchback, ABS, etc.

[Ondrej Chum]

Query expansion

Chum, Philbin, Sivic, Isard, Zisserman: Total Recall…, ICCV 2007 Ondrej Chum

Recognition Issues

How to summarize the content of an entire image? And gauge overall similarity?

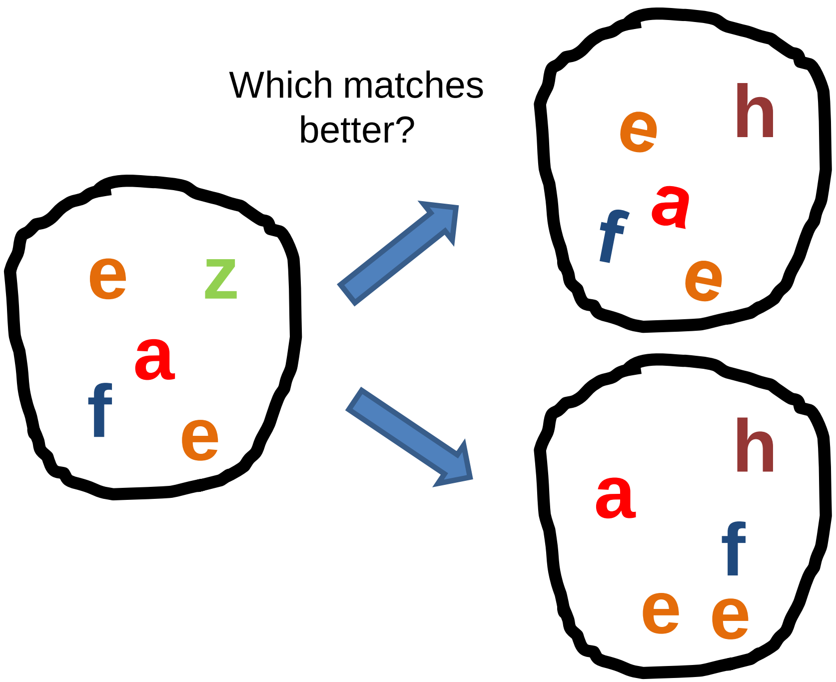
How large should the vocabulary be? How to perform quantization efficiently?

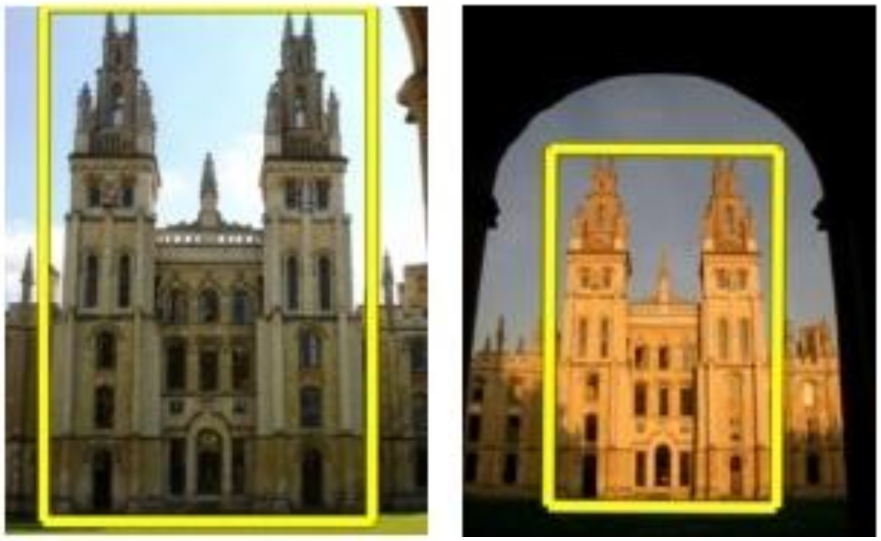
How to score the retrieval results?

How might we add more spatial verification?

Can we be more accurate?

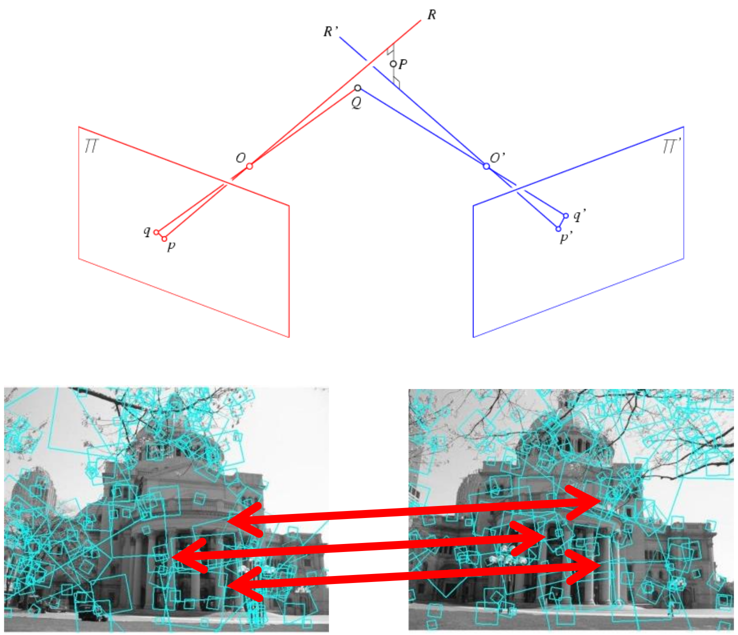
So far, we treat each image as containing a "bag of words", with no spatial information



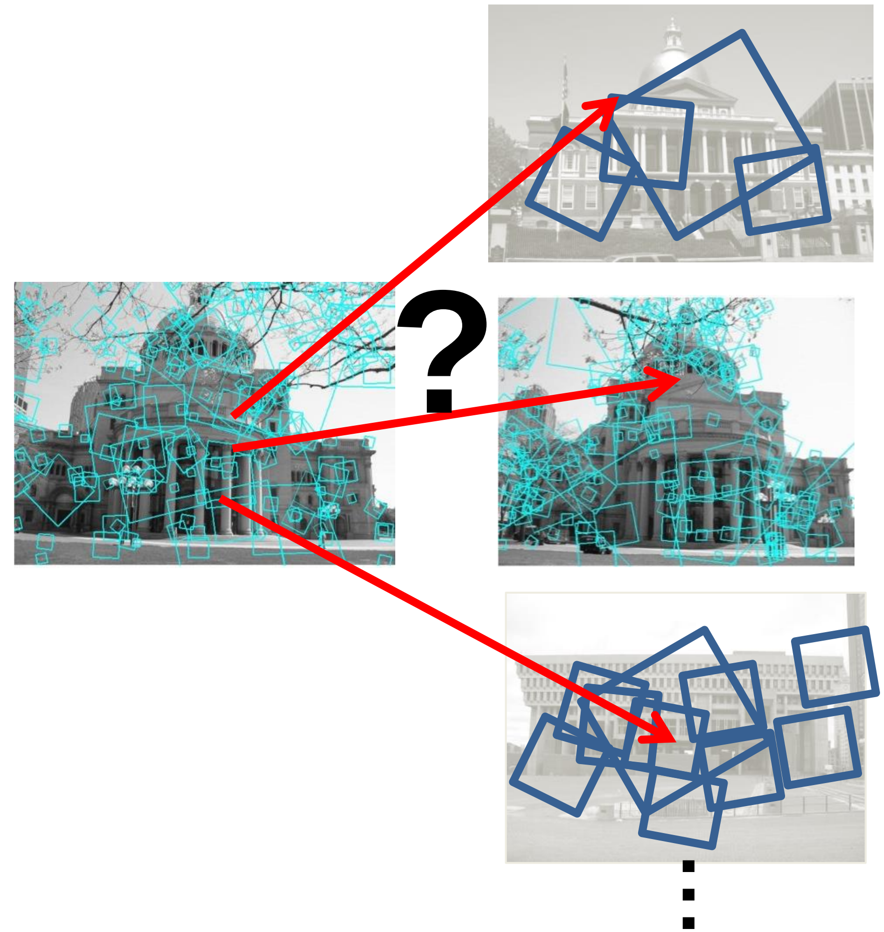


Real objects have consistent geometry

Multi-view matching



vs



Matching two given views for depth

Search for a matching view for recognition

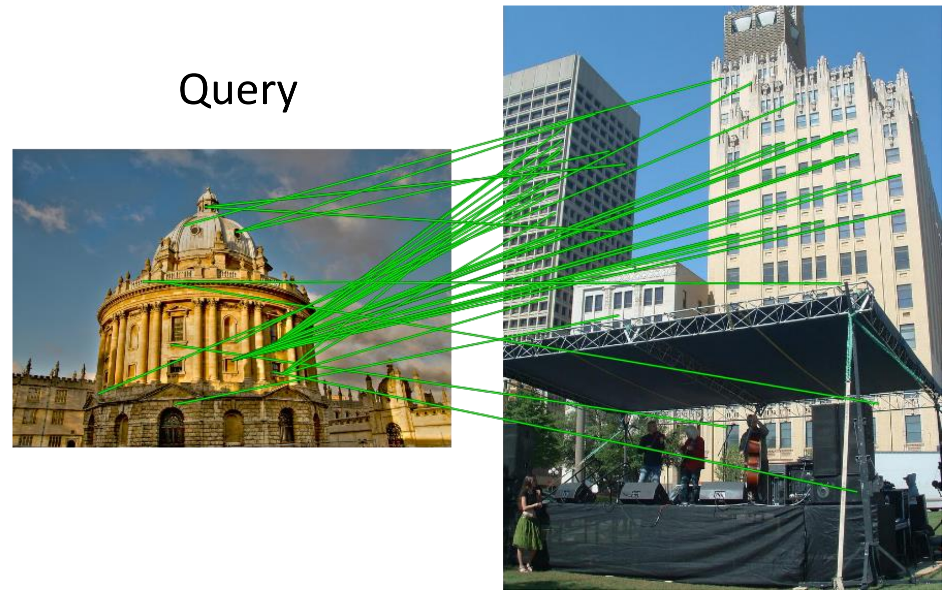
Kristen Grauman

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.

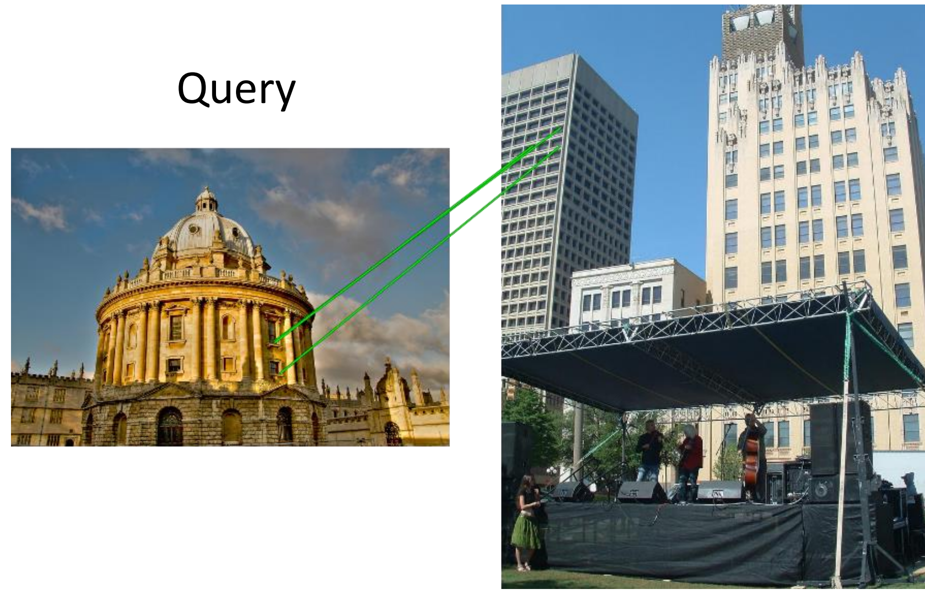
Slide credit: Ondrej Chum

Spatial Verification

Query

DB image with high BoW similarity

Query



DB image with high BoW similarity

Only some of the matches are mutually consistent with real-world geometry imaged by a camera.

Ondrej Chum

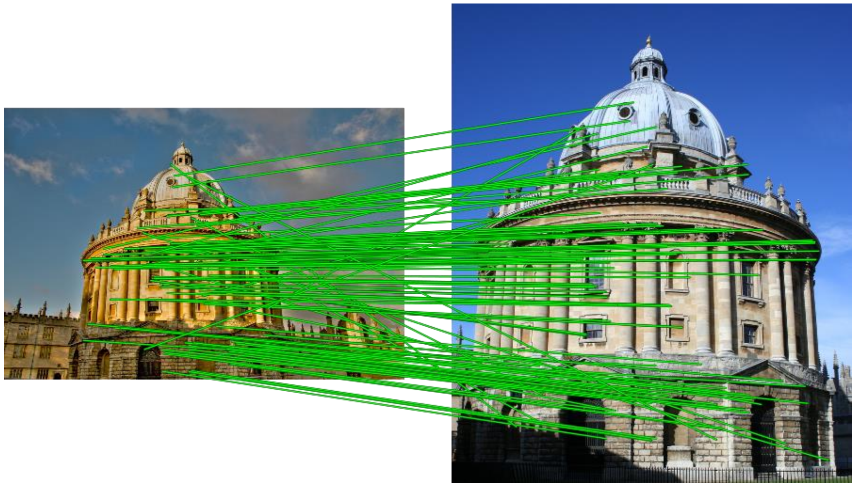
Spatial Verification: two basic strategies

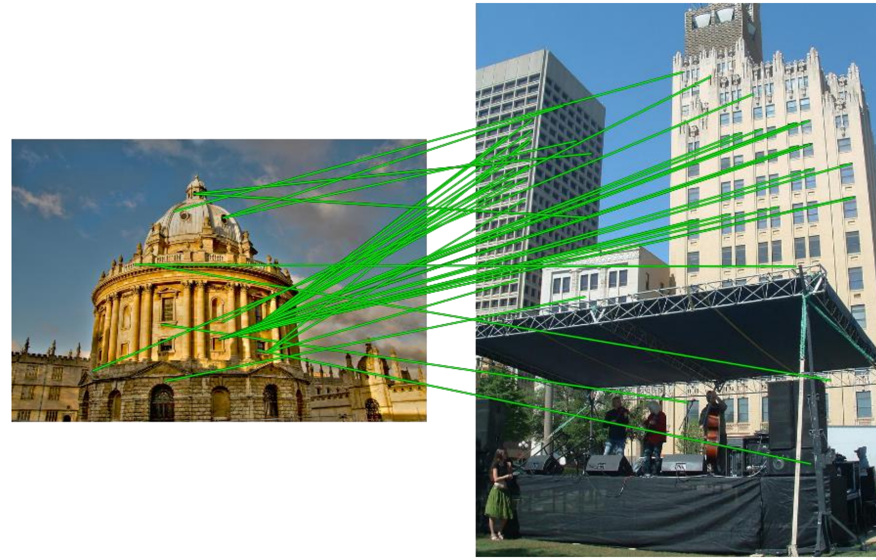
- RANSAC
  - Typically sort by BoW similarity as initial filter
  - Verify by checking support (inliers) for possible transformations
    - e.g., “success” if find a transformation with > N inlier correspondences

- Generalized Hough Transform
  - Let each matched feature cast a vote on location, scale, orientation of the model object
  - Verify parameters with enough votes

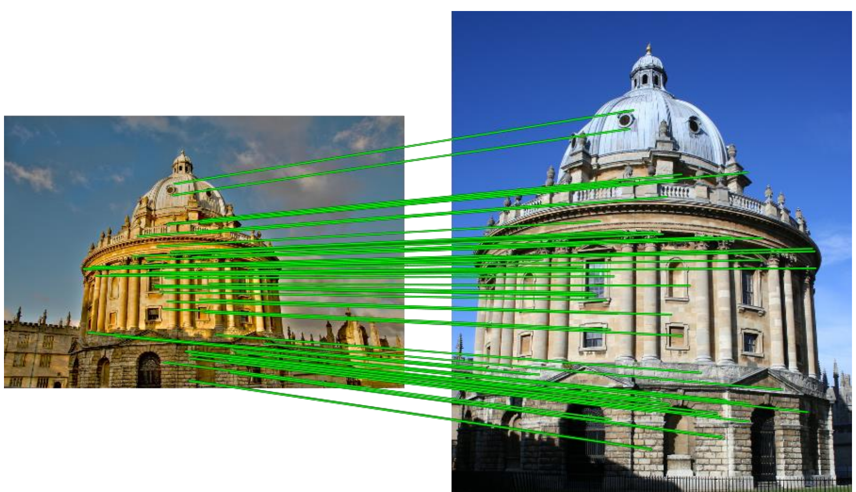
Kristen Grauman

No verification





RANSAC verification



Fails to meet threshold
on # inliers! Good!

Recognition via alignment

Pros:

– Effective for reliable features within clutter

– Great for matching specific instances

Cons:

– Expensive post-process (how long for proj3?!)

– Not suited for category recognition

Kristen Grauman

Summary

- **Bag of words**: quantize feature space into discrete visual words
  - Summarize image by distribution of words
- **Inverted index**: visual word index for faster query time
- **Evaluation:**
- **Additional spatial verification alignment:**
  - Robust fitting : RANSAC, Generalized Hough Transform
  - We will do this in detail later on in the course

Kristen Grauman

Lessons from a decade later

For *Category* recognition (project 3)

- Bag of Feature models remained the state of the art until Deep Learning.
- Spatial layout either isn't that important or its too difficult to encode.
- Quantization error is, in fact, the bigger problem. Advanced feature encoding methods address this.
- Bag of feature models are nearly obsolete. At best they seem to be inspiring tweaks to deep models e.g., NetVLAD.

Lessons from a decade later

For *instance* retrieval (this lecture):

- deep learning is taking over.
- learn better local features (replace SIFT)
  e.g., MatchNet 2015
- learn better image embeddings (replace visual word histograms)
  e.g., Vo and Hays 2016.
- learn spatial verification
  e.g., DeTone, Malisiewicz, and Rabinovich 2016.
- learn a monolithic deep network to recognition all locations
  e.g., Google's PlaNet 2016.

James Hays