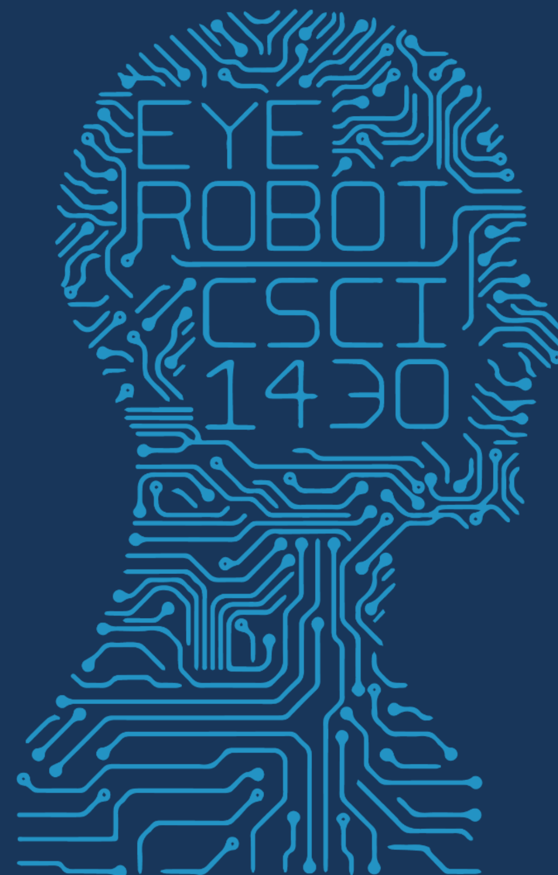
1950

Future Vision



2017 MWF 1pm 368

Computer Vision

CV as making bank

- Intel buys Mobileye!
- $15 billion

- Mobileye:
  - Spin-off from Hebrew University, Israel
  - 450 engineers
  - 15 million cars installed
  - 313 car models

Intel buys driverless car technology firm Mobileye

2 hours ago | Business | 138 Share

US chipmaker Intel is taking a big bet on driverless cars with a $15.3bn (£12.5bn) takeover of specialist Mobileye.

Intel will pay $63.54 a share in cash for the Israeli company, which develops "autonomous driving" systems.

Mobileye and Intel are already working together, along with German carmaker BMW, to put 40 test vehicles on the road in the second half of this year.

Intel expects the driverless market to be worth as much as $70bn by 2030.

Technology companies are racing to launch driverless cars.

June 2016 - Tesla left Mobileye

- Fatal crash – car ‘autopilot’ ran into a tractor trailer.

“What we know is that the vehicle was on a divided highway with Autopilot engaged when a tractor trailer drove across the highway perpendicular to the Model S. Neither Autopilot nor the driver noticed the white side of the tractor trailer against a brightly lit sky, so the brake was not applied.” – Tesla blog.

What computer vision problems does this sound like?

Tesla crash: how it happened

A preliminary investigation into 25,000 Tesla Model S cars has been opened after a driver of one of the vehicles was killed while operating in Autopilot mode in a crash in Williston, Florida. Here is how the fatal accident occurred according to authorities.

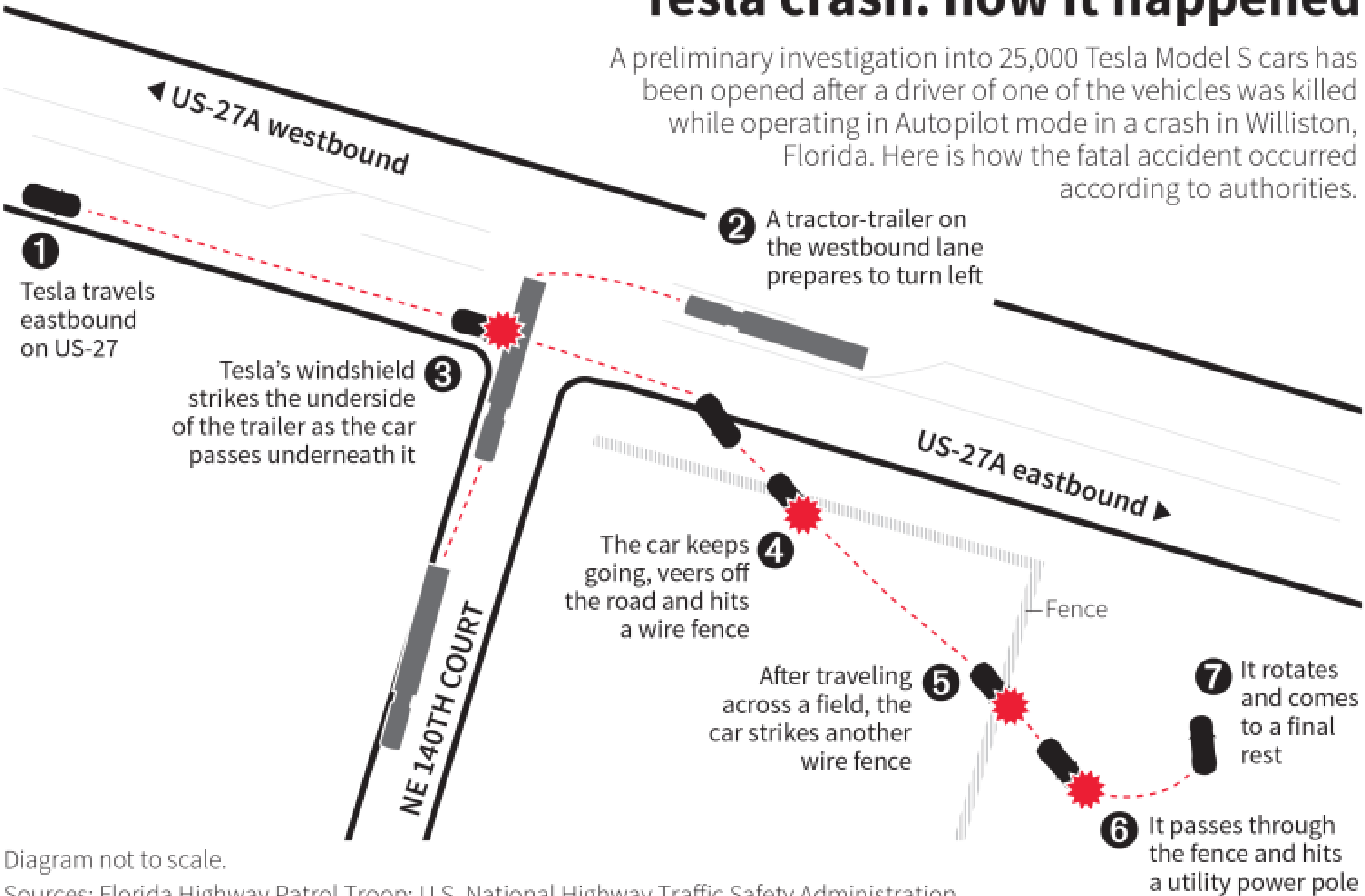


Diagram not to scale.

Sources: Florida Highway Patrol Troop; U.S. National Highway Traffic Safety Administration

C. Chan, 30/06/2016

REUTERS


EXPRESS
0:07 / 0:16
2017/02/2

June 2016 - Tesla left Mobileye

- Fatal crash – car ‘autopilot’ ran into a tractor trailer.

“What we know is that the vehicle was on a divided highway with Autopilot engaged when a tractor trailer drove across the highway perpendicular to the Model S. Neither Autopilot nor the driver noticed the white side of the tractor trailer against a brightly lit sky, so the brake was not applied.” – Tesla blog.

What computer vision problems does this sound like?

What HCI problems does this sound like?

Autosteer

U.S. Department of Transportation
National Highway Traffic Safety Administration

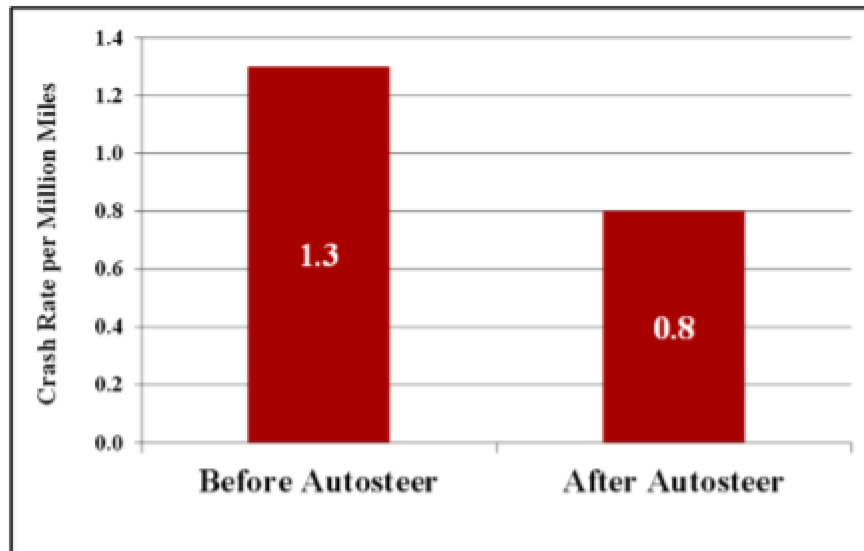


Figure 11. Crash Rates in MY 2014-16 Tesla Model S and 2016 Model X vehicles Before and After Autosteer Installation.

Machine Learning Problems

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

Supervised learning

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

Prediction function

Image feature

Output (label)

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

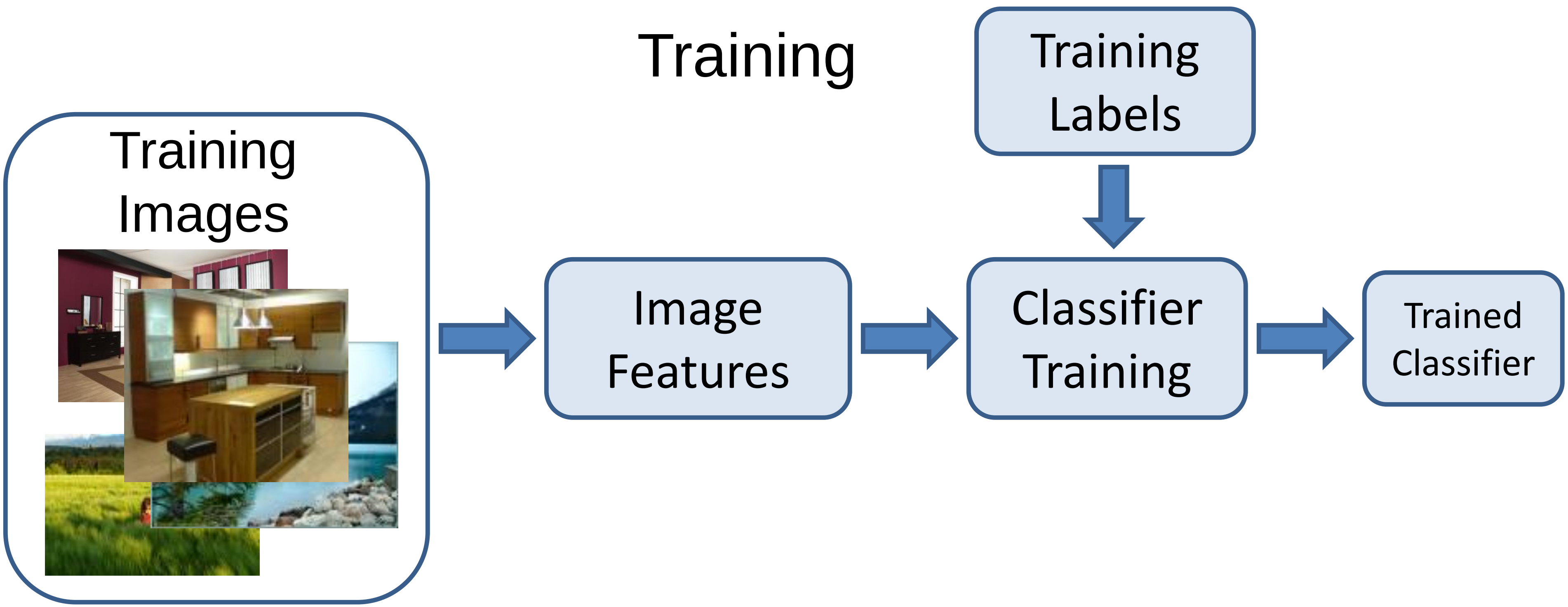
$$\{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_N,y_N)\}$$

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



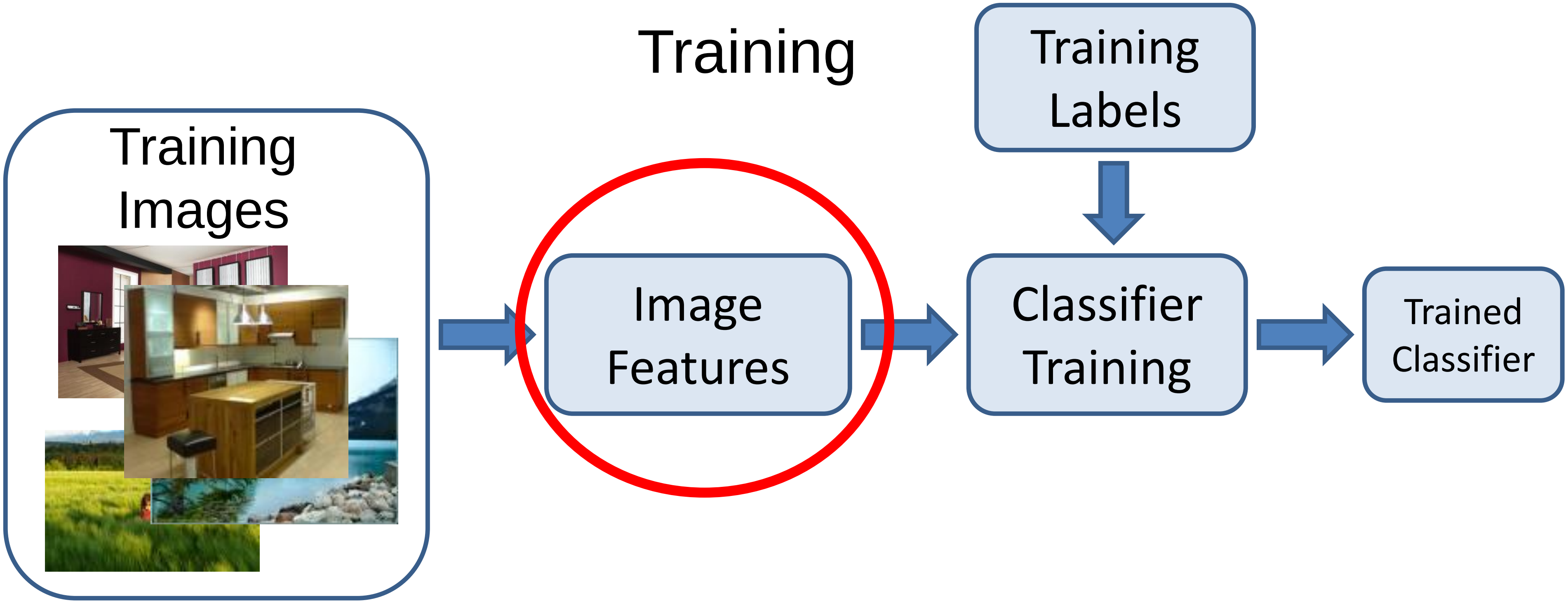
An elephant standing on top of a basket being held by a woman

MS COCO

wordseye.com

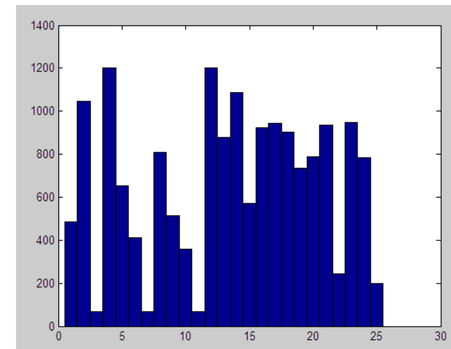
Thank you Trent Green

Image features

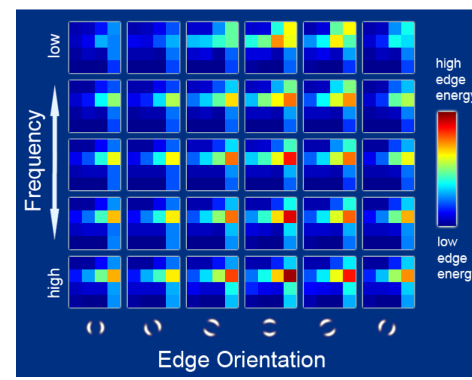


Features

- Raw pixels
- Histograms



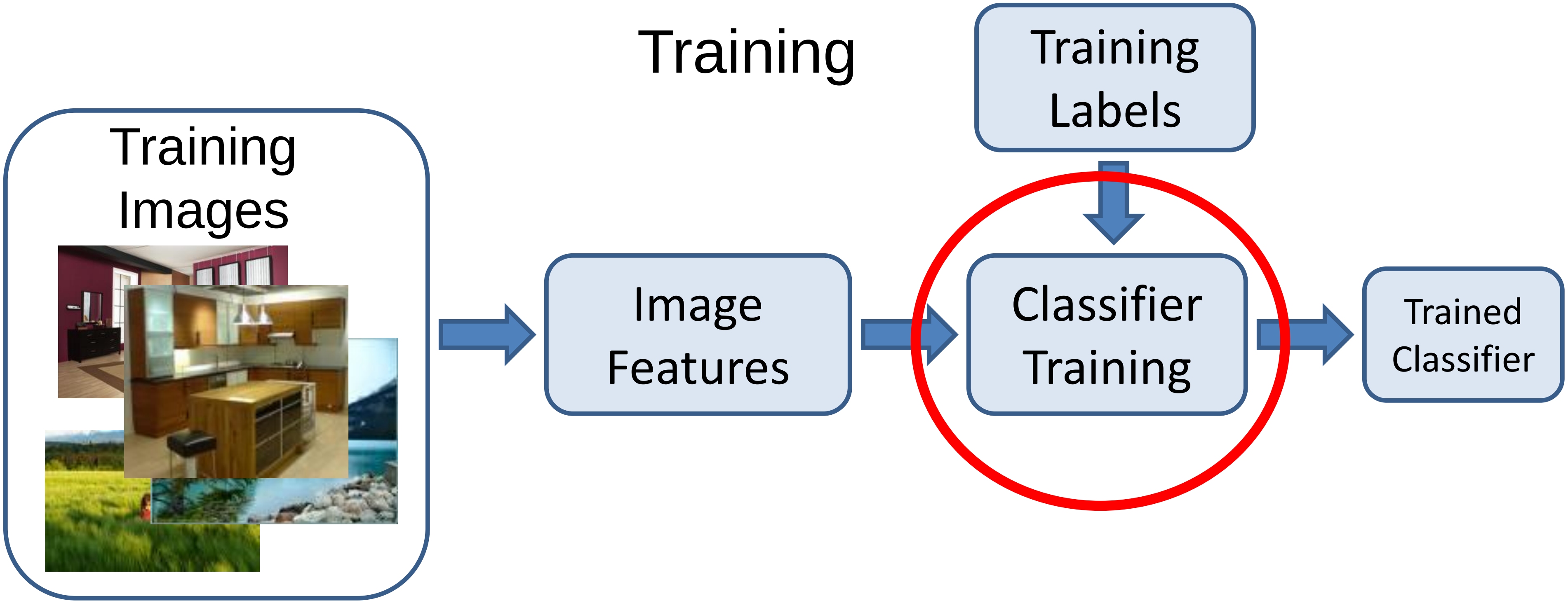
- GIST descriptors



- …

Slide credit: L. Lazebnik

Classifiers



Learning a classifier

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

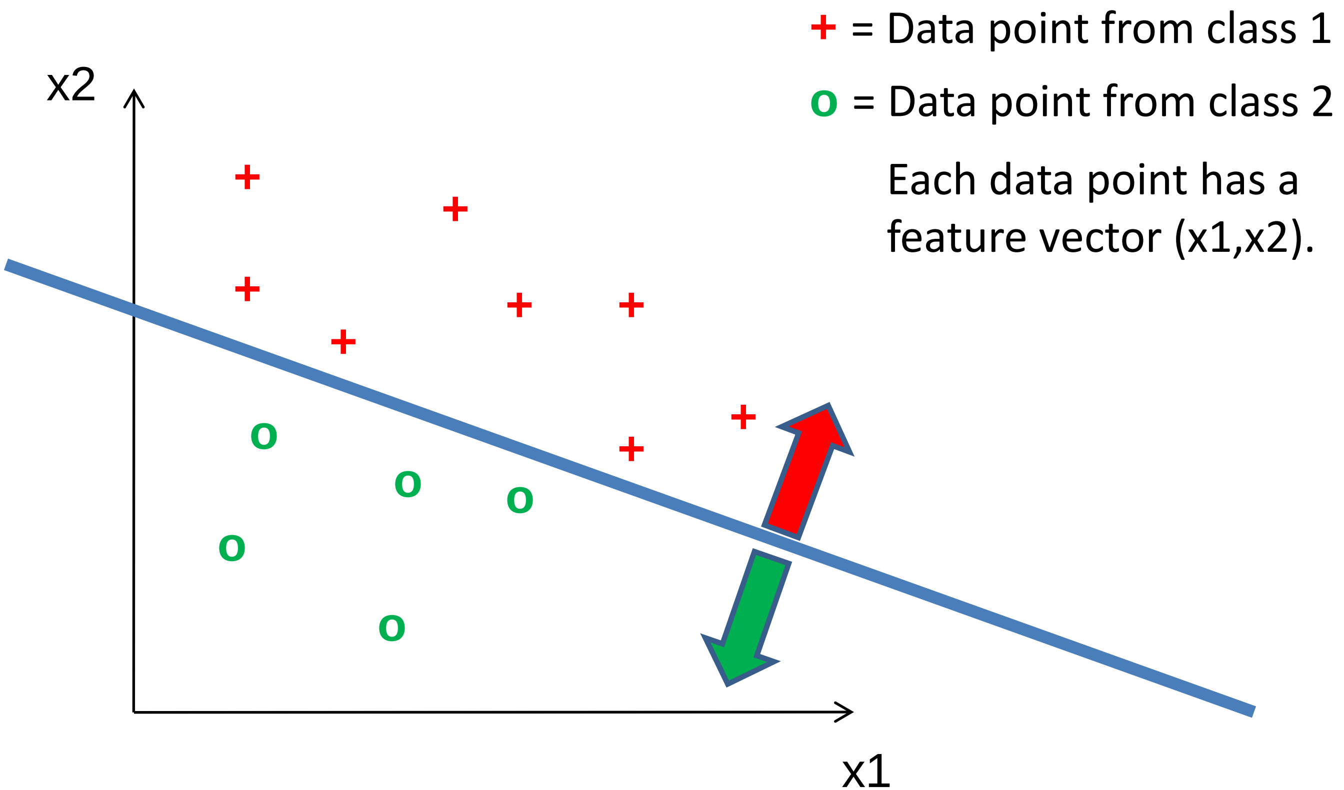
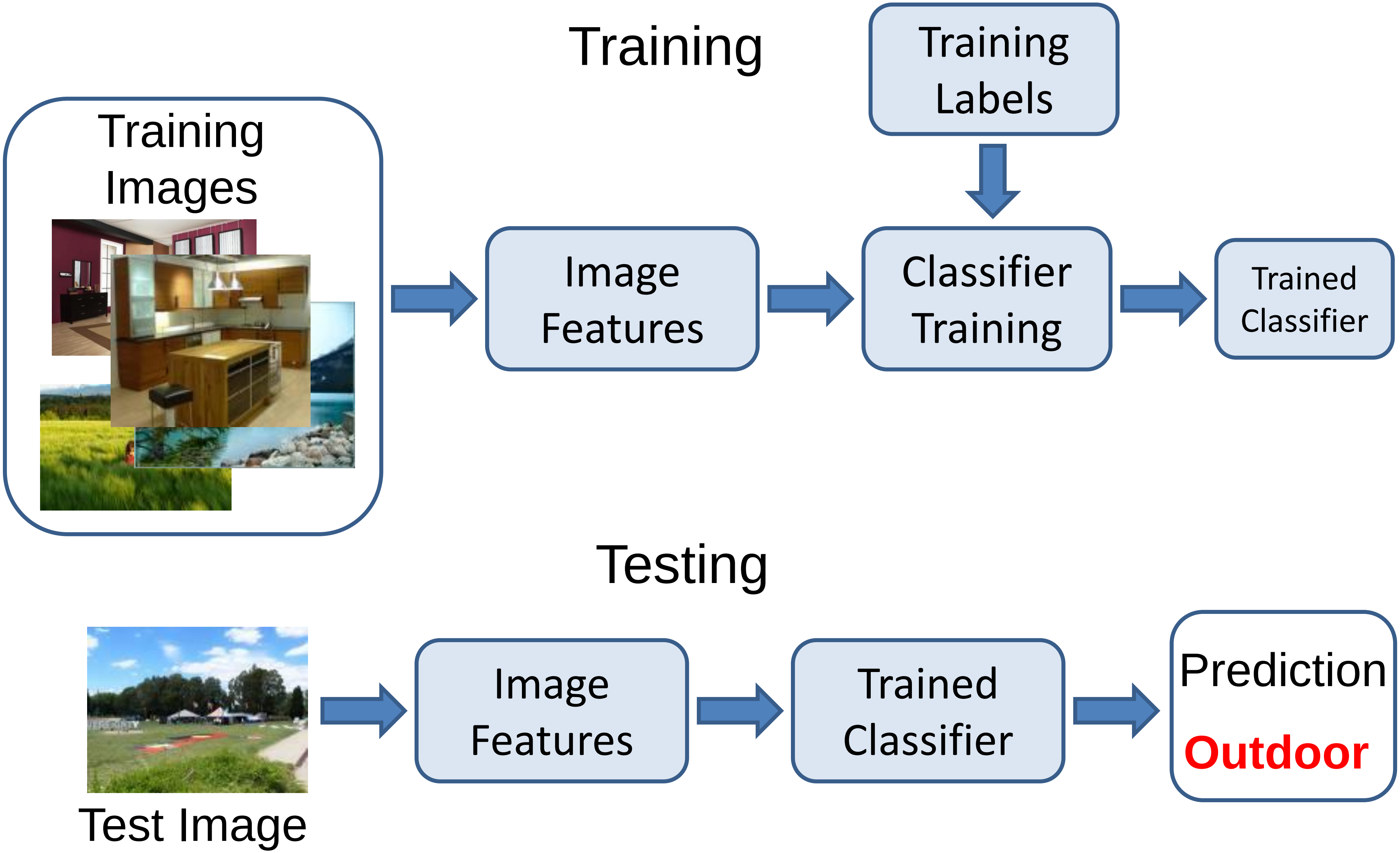


Image Categorization

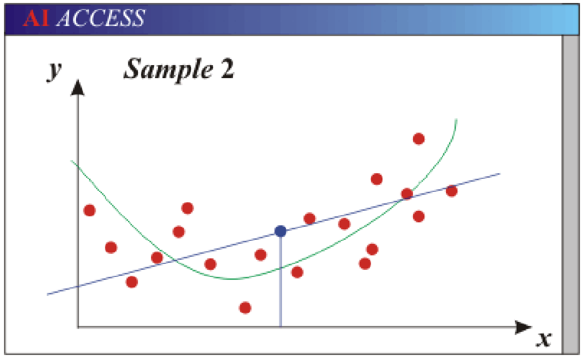


Example: Scene Categorization

- Is this a kitchen?

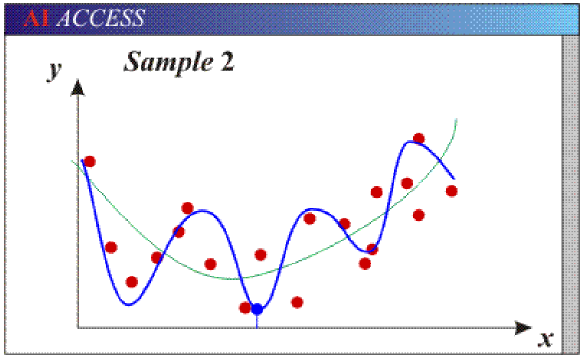
Bias-Variance Trade-off

Bias: 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!



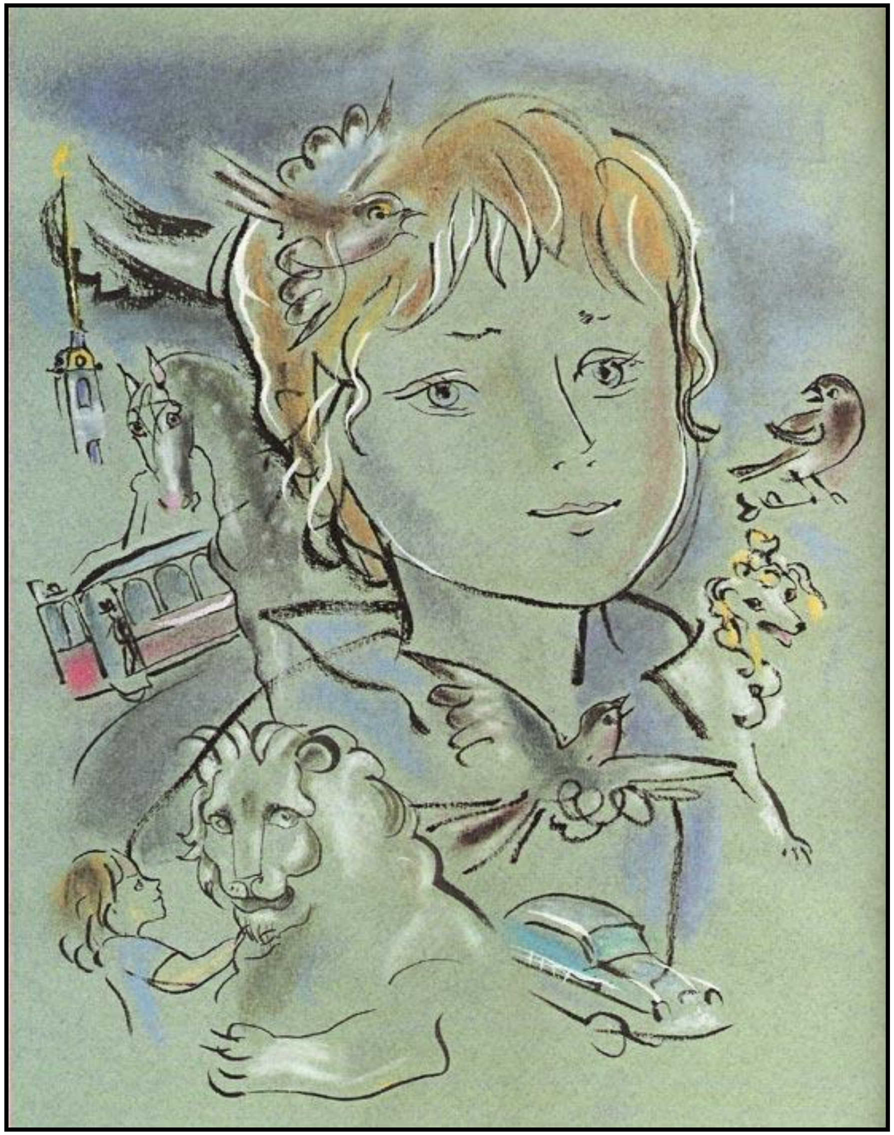
Models with too many parameters are inaccurate because of a large variance.

- Too much sensitivity to the sample.

Last week: ML crash course

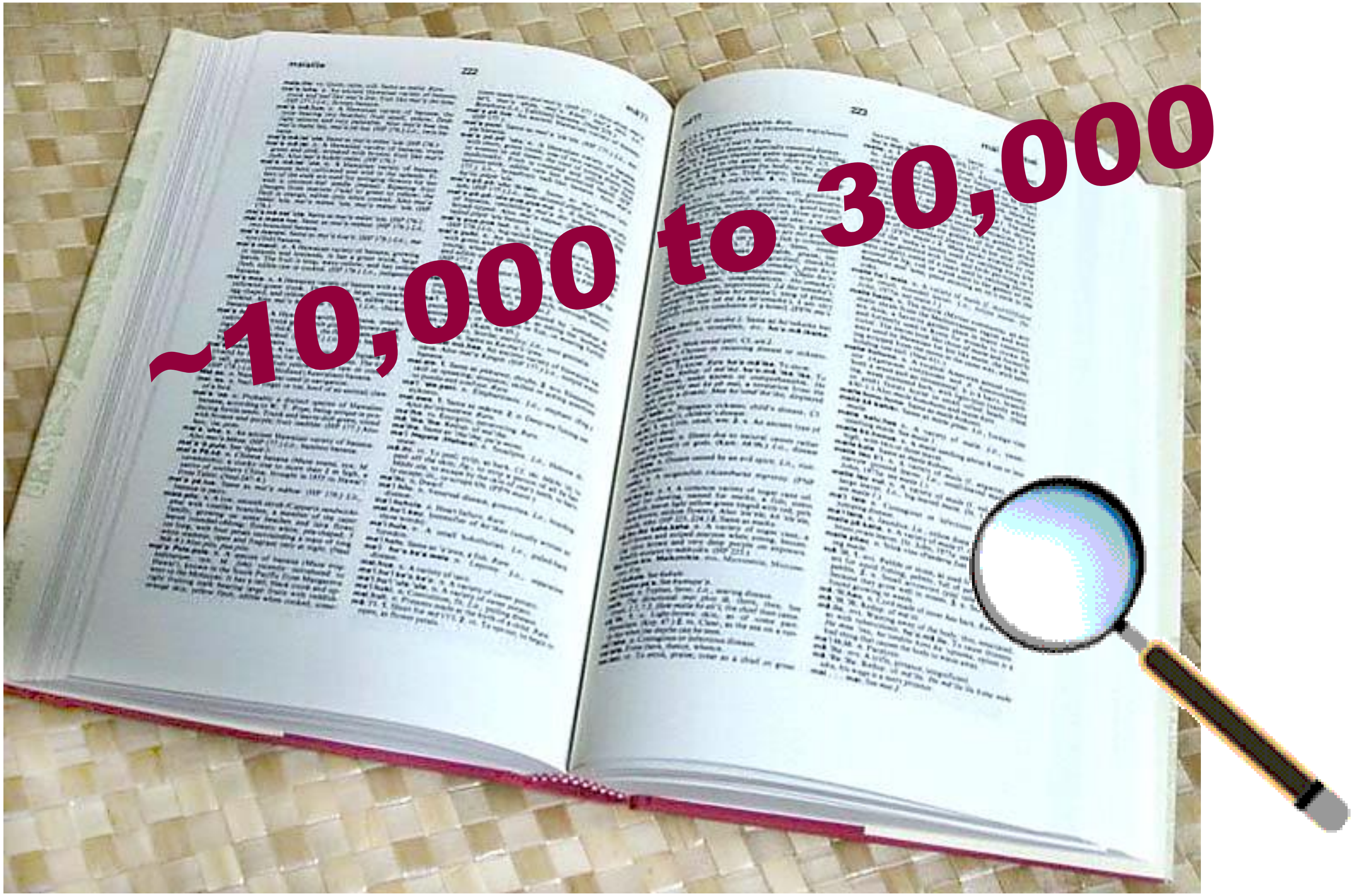
- Nice write-up of the bias-variance issues
- http://www.learnopencv.com/bias-variance-tradeoff-in-machine-learning/

Recognition: Overview and History



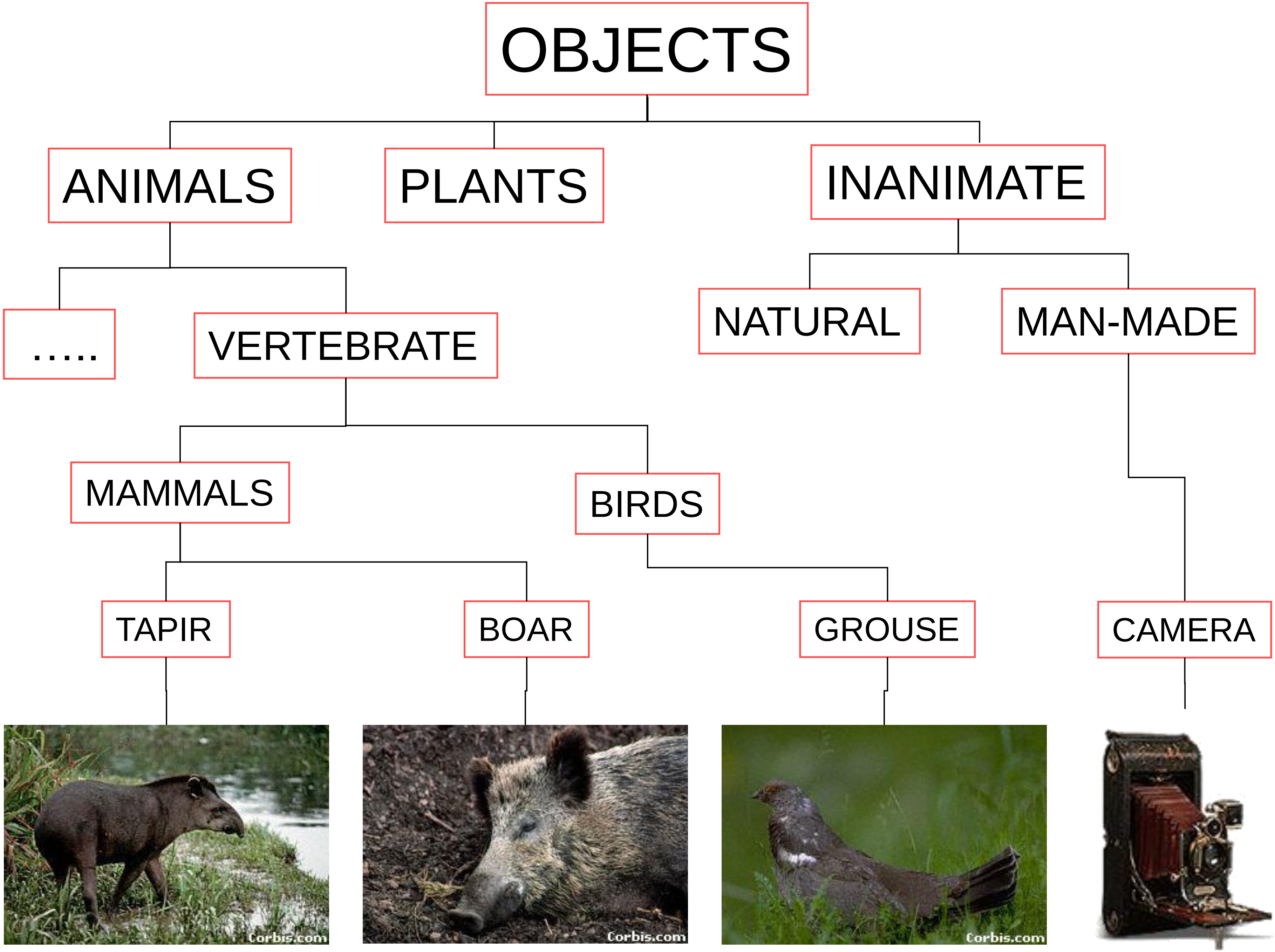
Slides from James Hays, Lana Lazebnik, Fei-Fei Li, Rob Fergus, Antonio Torralba, and Jean Ponce

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

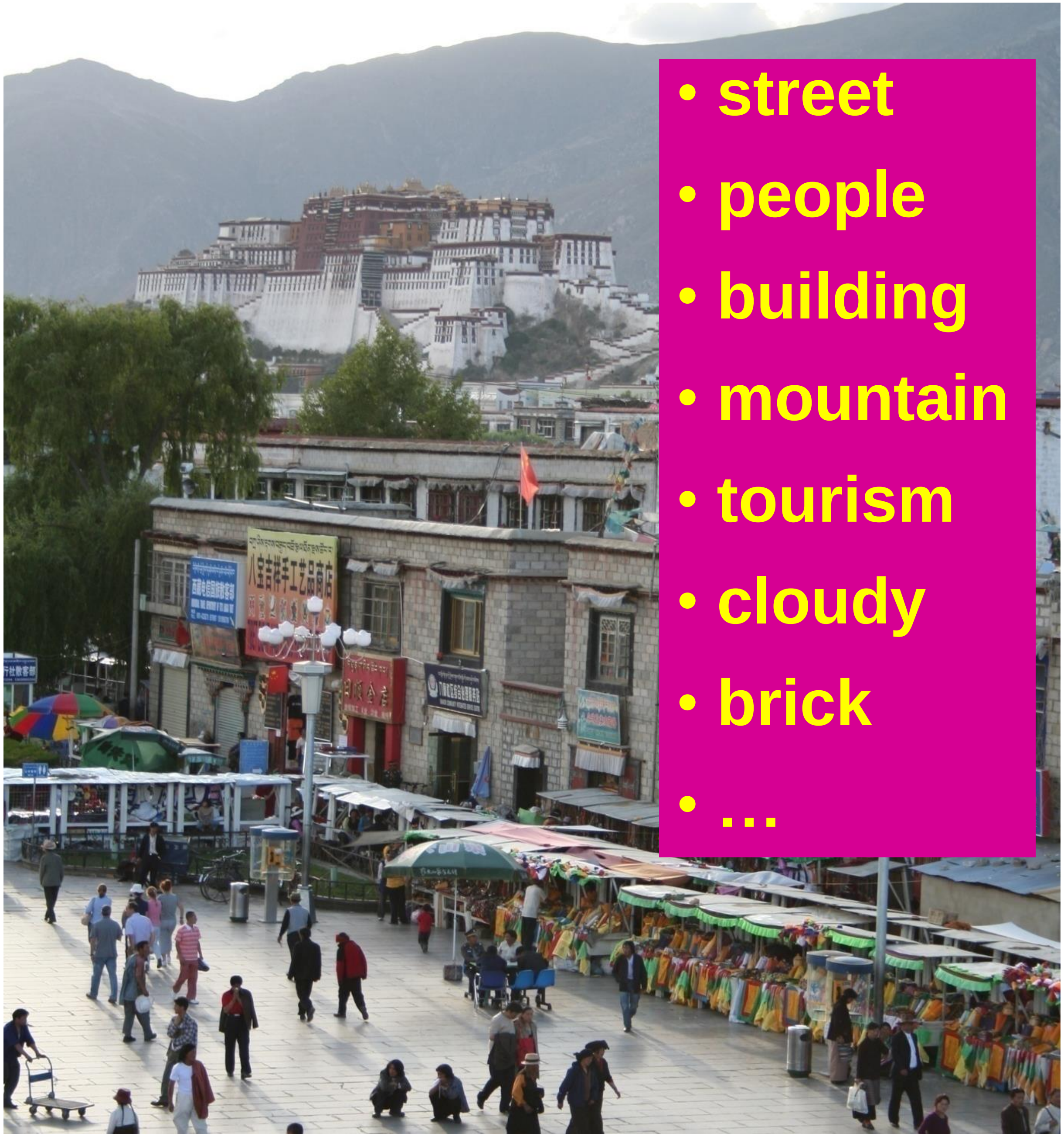
Svetlana Lazebnik

Scene categorization or classification

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

Svetlana Lazebnik

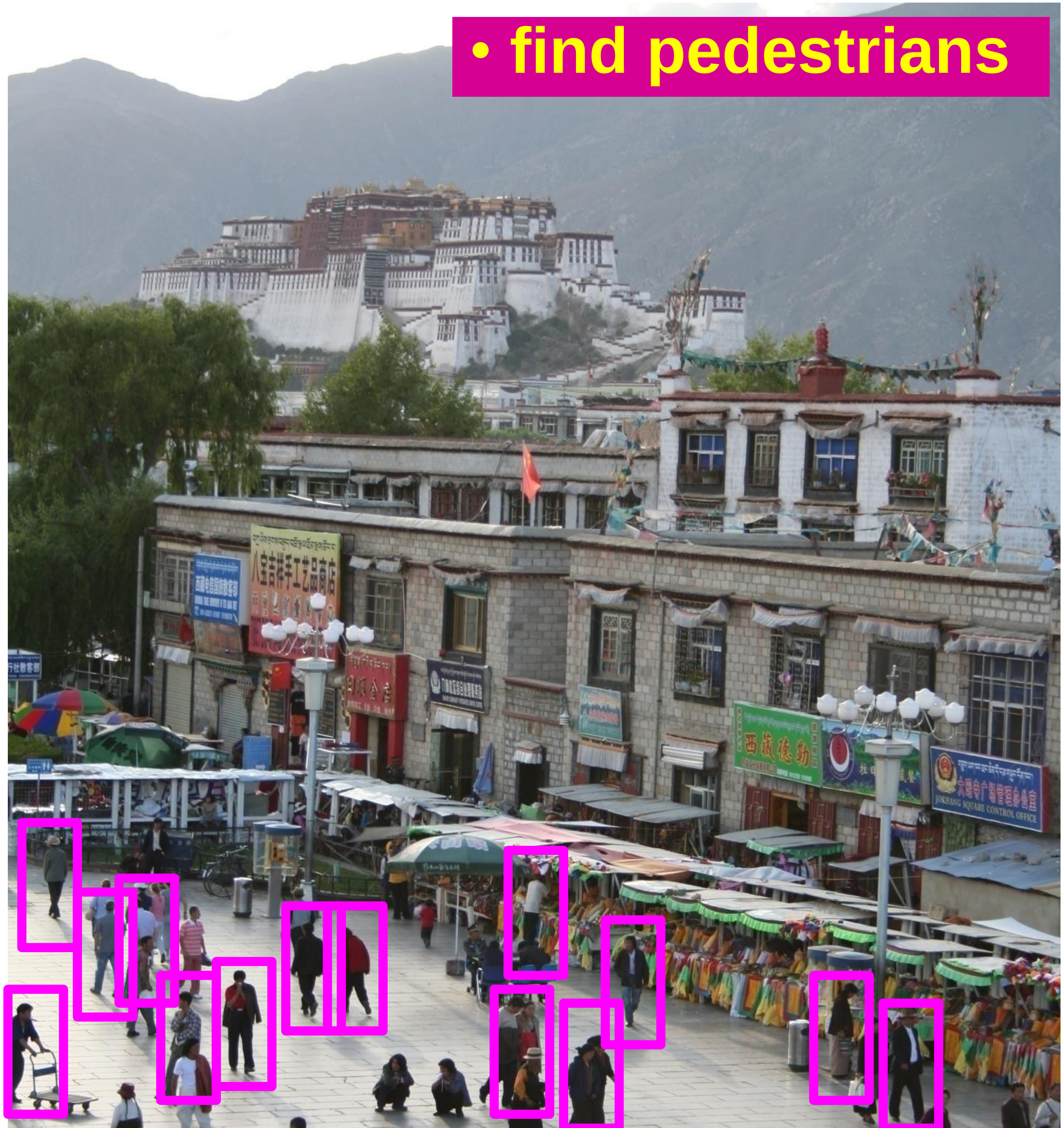
Image annotation / tagging / attributes



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

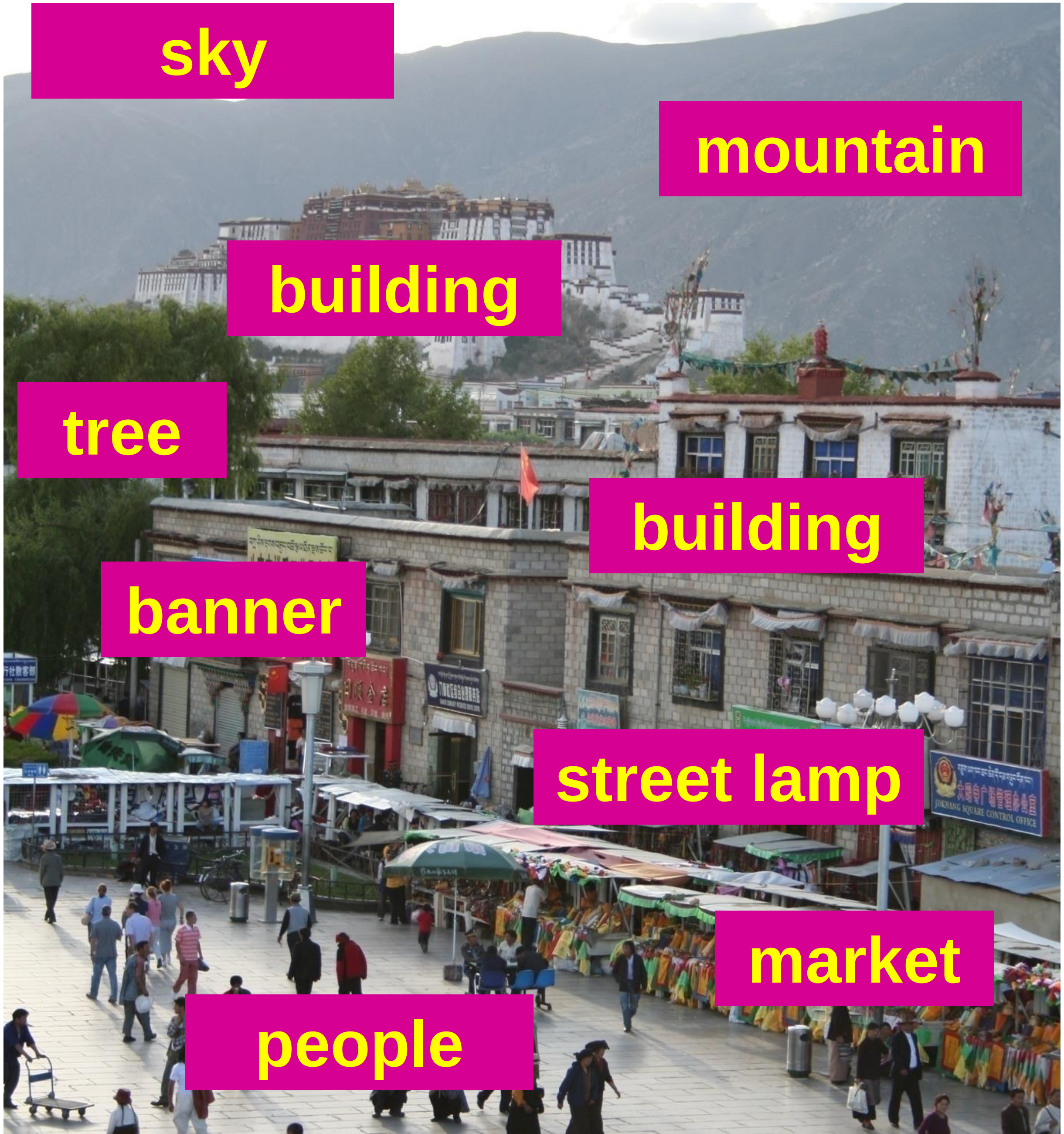
Svetlana Lazebnik

Object detection



Svetlana Lazebnik

Image parsing / semantic segmentation

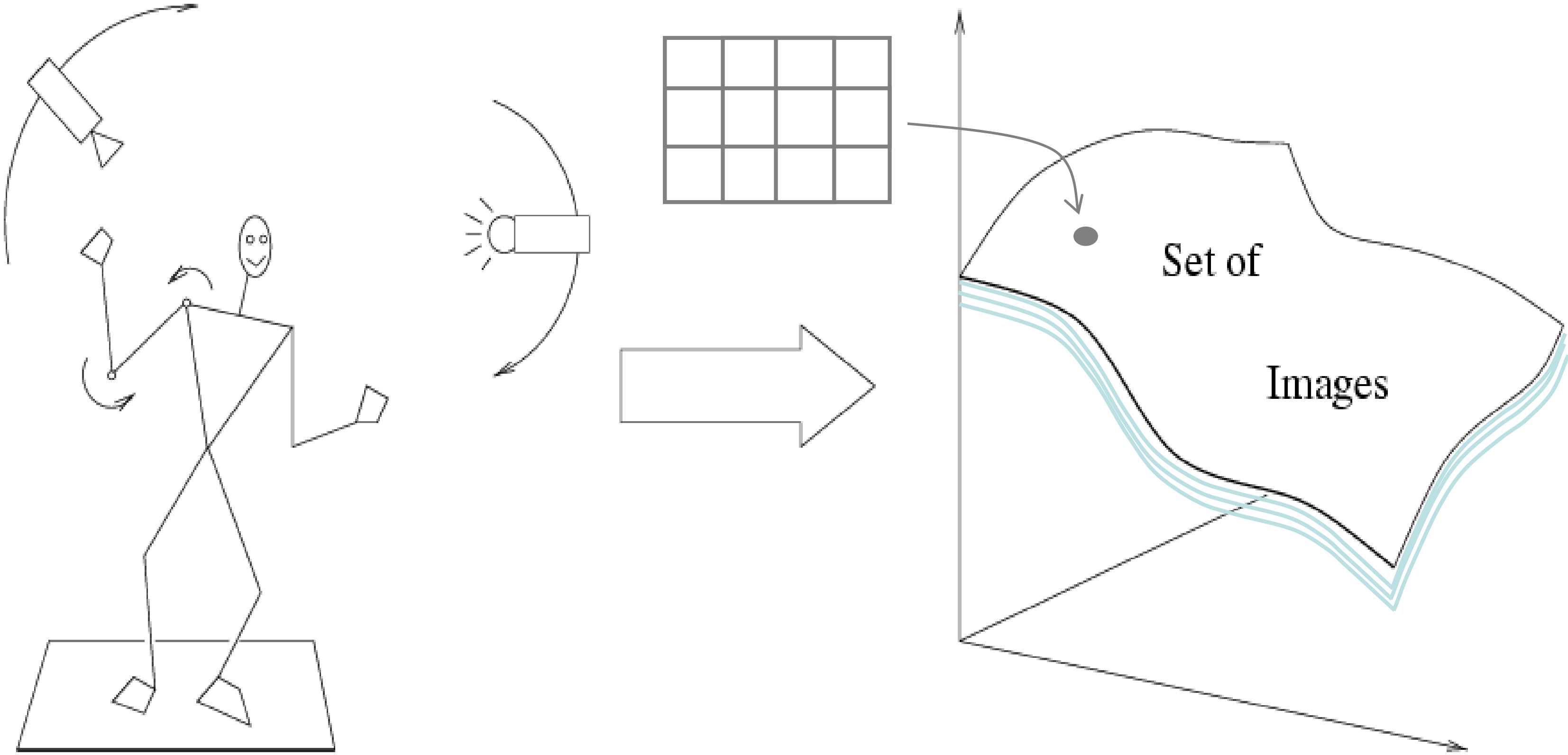


Svetlana Lazebnik

Scene understanding?

Svetlana Lazebnik

Recognition is all about modeling variability

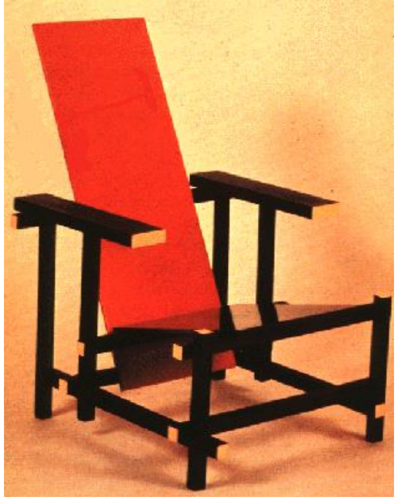


Variability: Camera position
Illumination
Shape parameters

Within-class variations?

Within-class variations



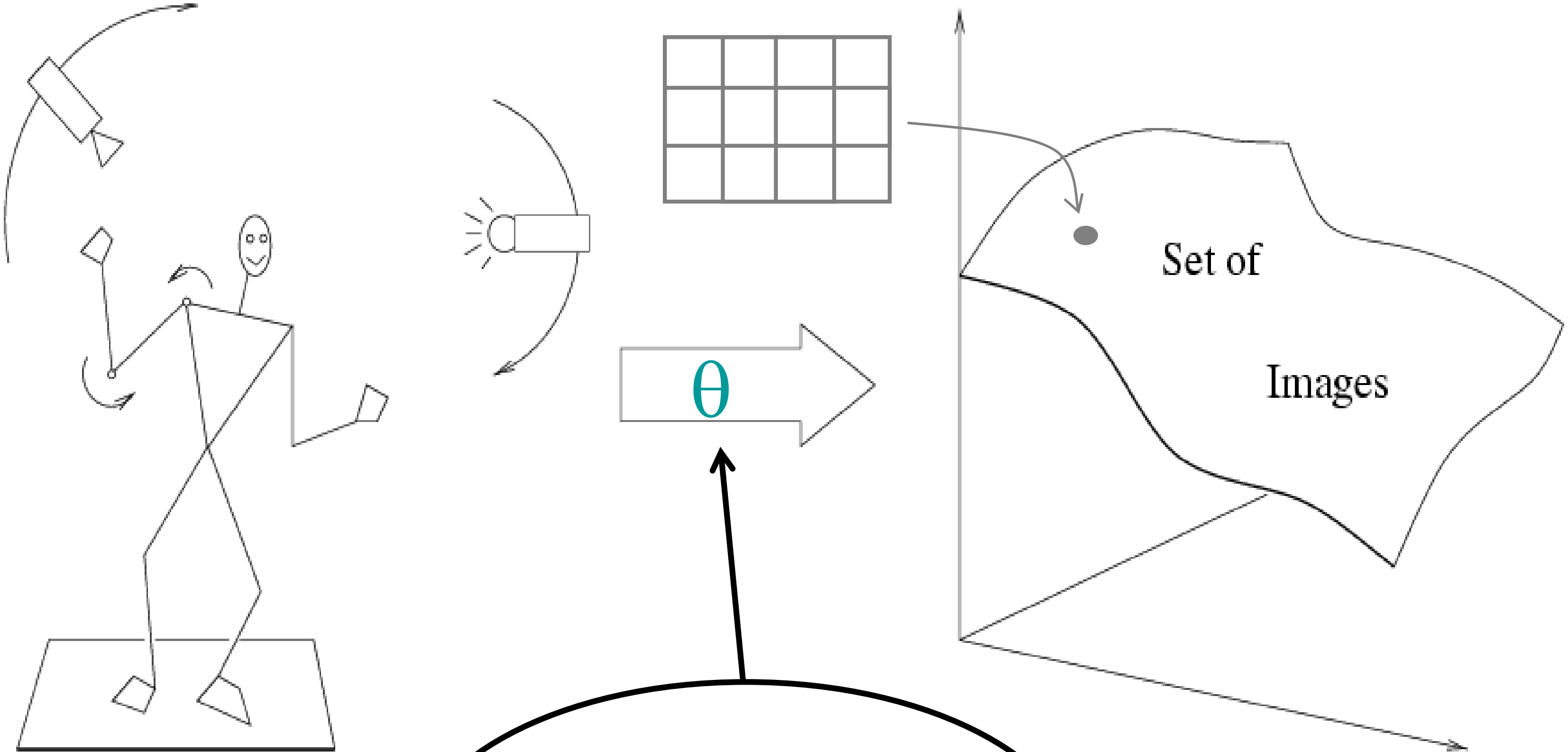
Svetlana Lazebnik

History of ideas in recognition

- 1960s – early 1990s: the geometric era

No digital cameras!
Slow compute!

Svetlana Lazebnik



Variability:

Camera position
Illumination

Alignment

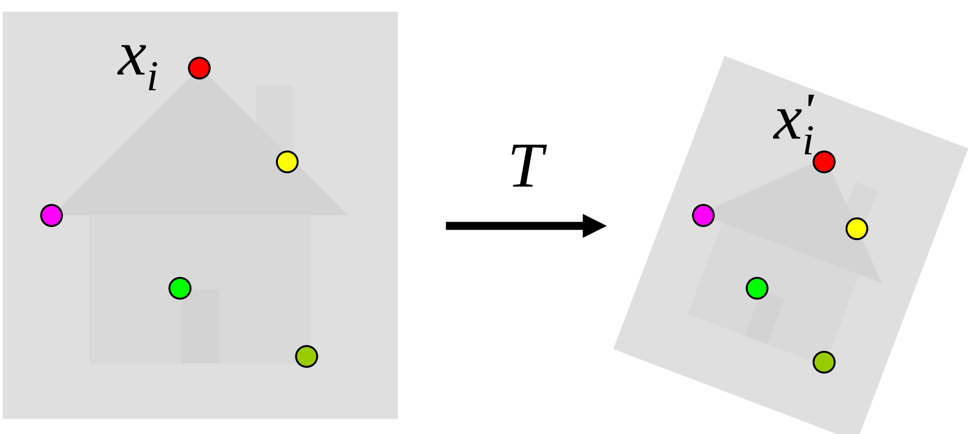
Shape: assumed known

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

Svetlana Lazebnik

Recall: Alignment

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

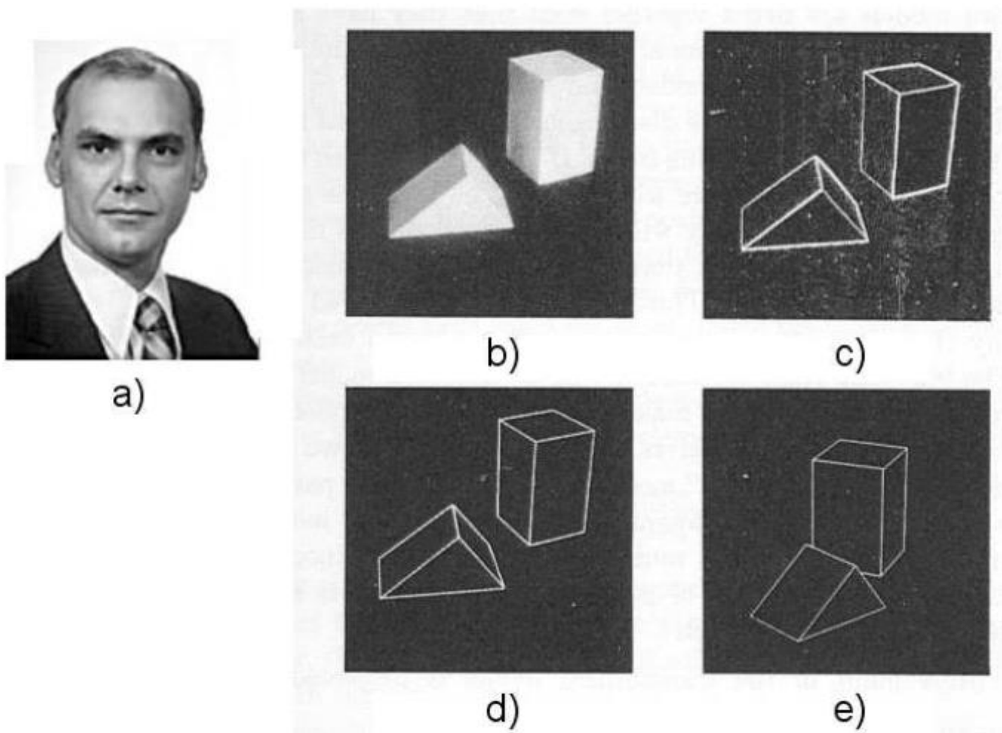


Find transformation T
that minimizes

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

Svetlana Lazebnik

Recognition as an alignment problem: Block world

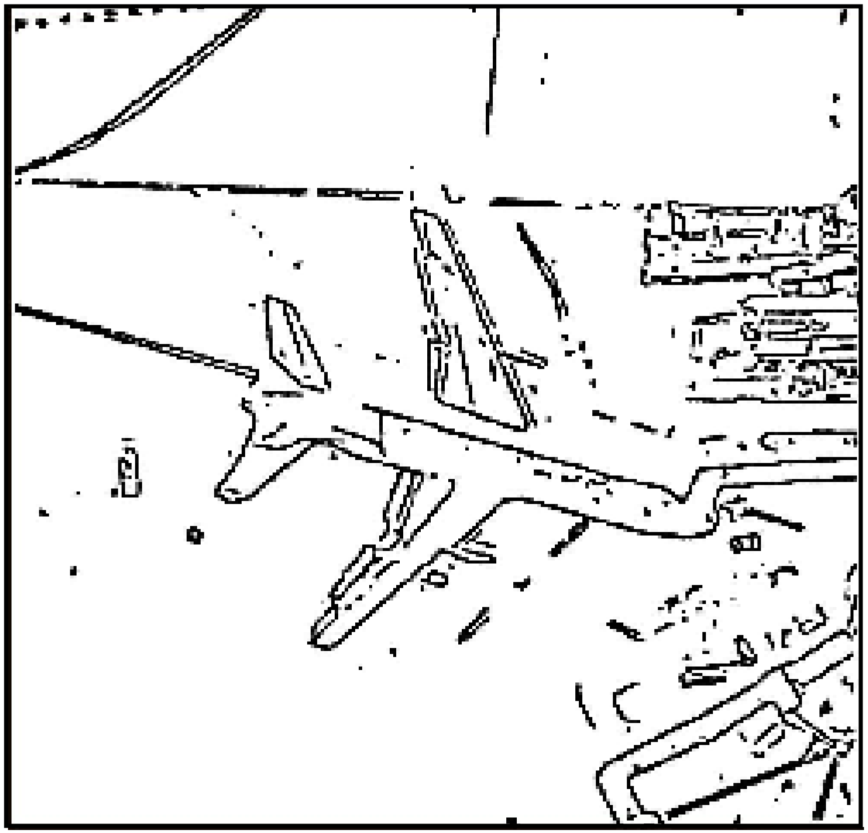


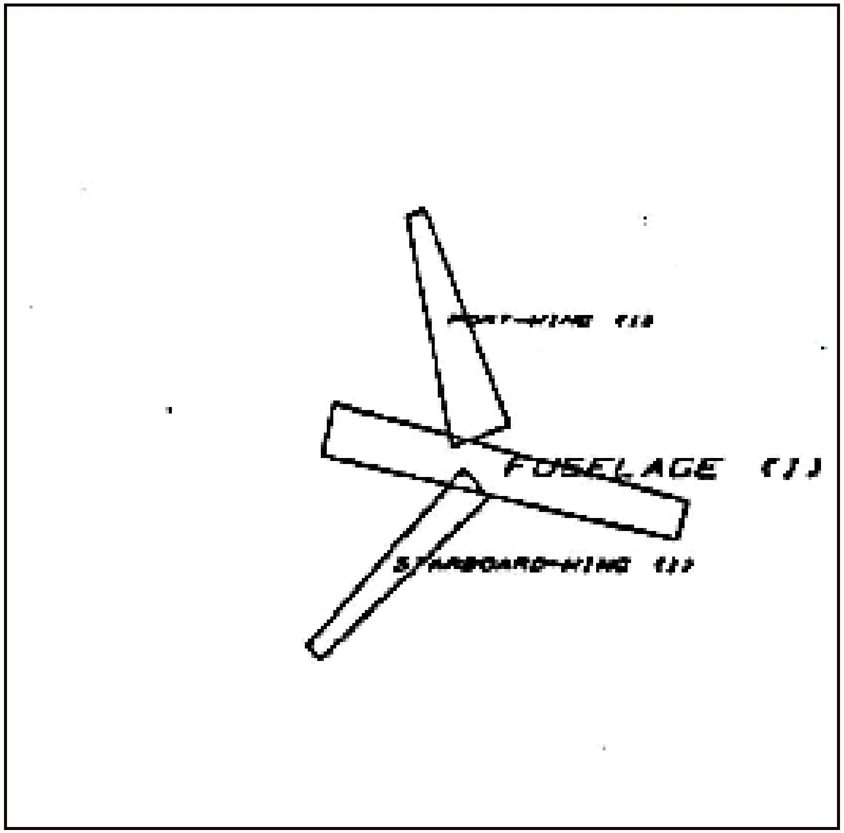
L. G. Roberts
Machine Perception of Three Dimensional Solids, Ph.D. thesis, MIT Department of Electrical Engineering, 1963.

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...





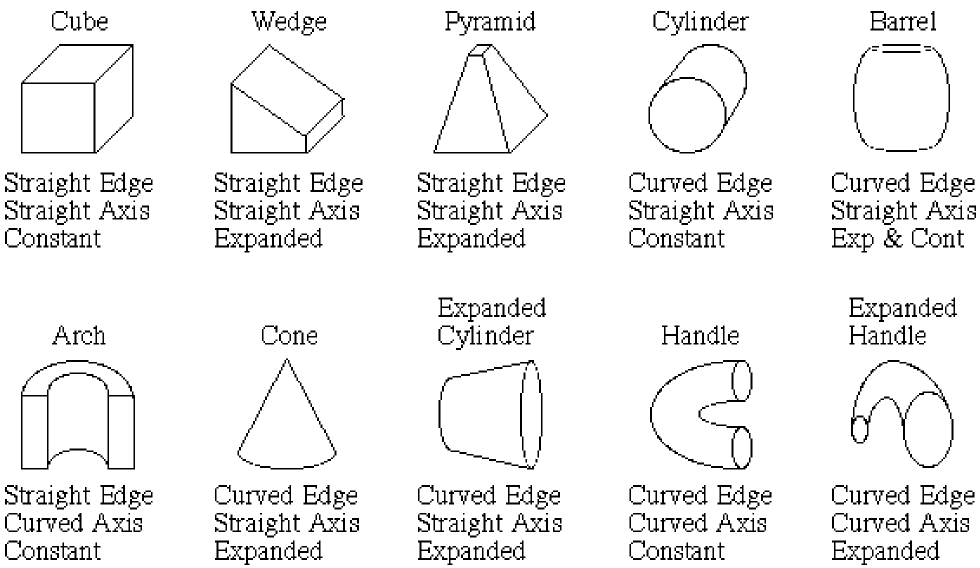
ACRONYM (Brooks and Binford, 1981)

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

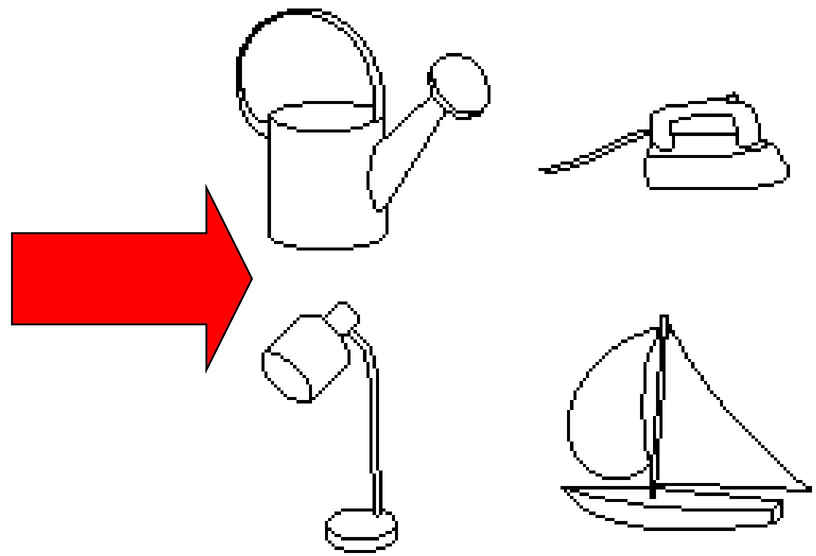
Recognition by components

Biederman (1987)

Primitives (geons)

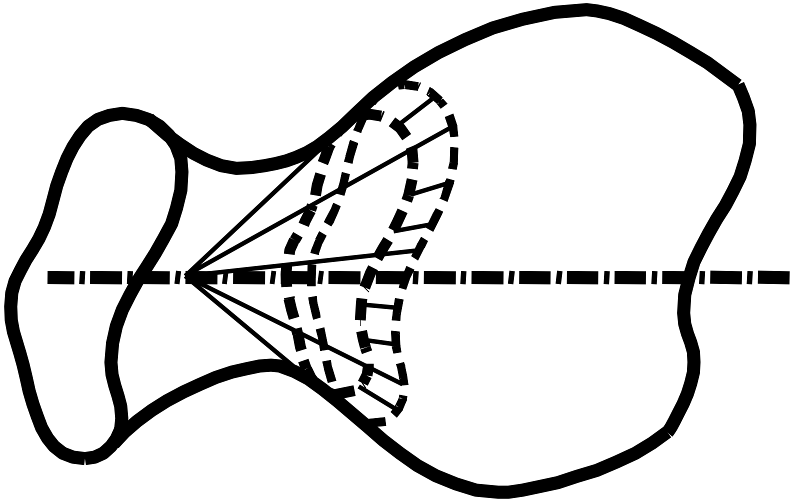


Objects

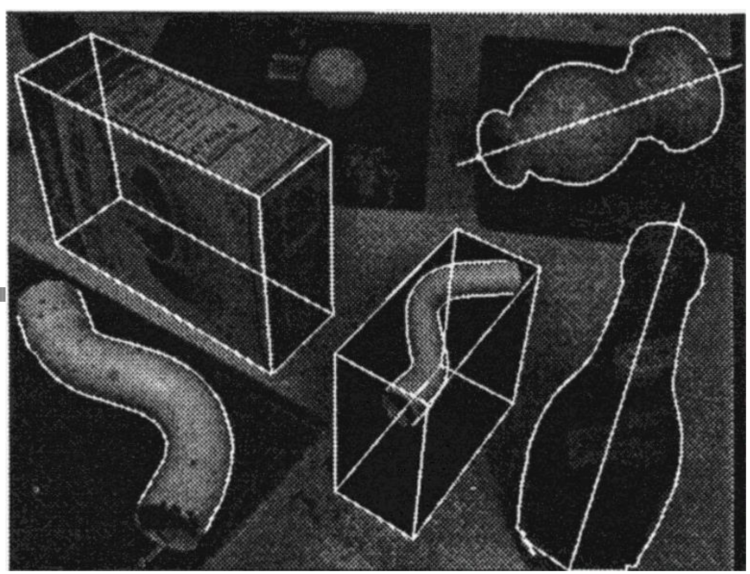


http://en.wikipedia.org/wiki/Recognition_by_Components_Theory

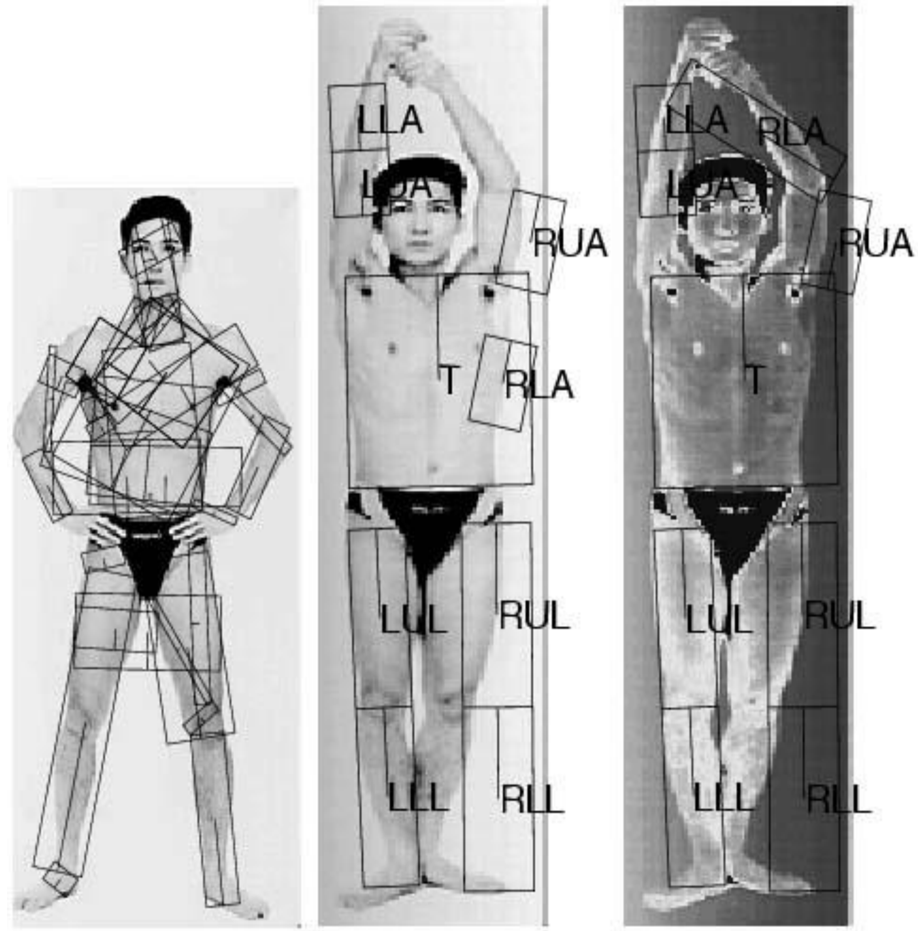
General shape primitives?



Generalized cylinders
Ponce et al. (1989)



Zisserman et al. (1995)



Forsyth (2000)

Svetlana Lazebnik

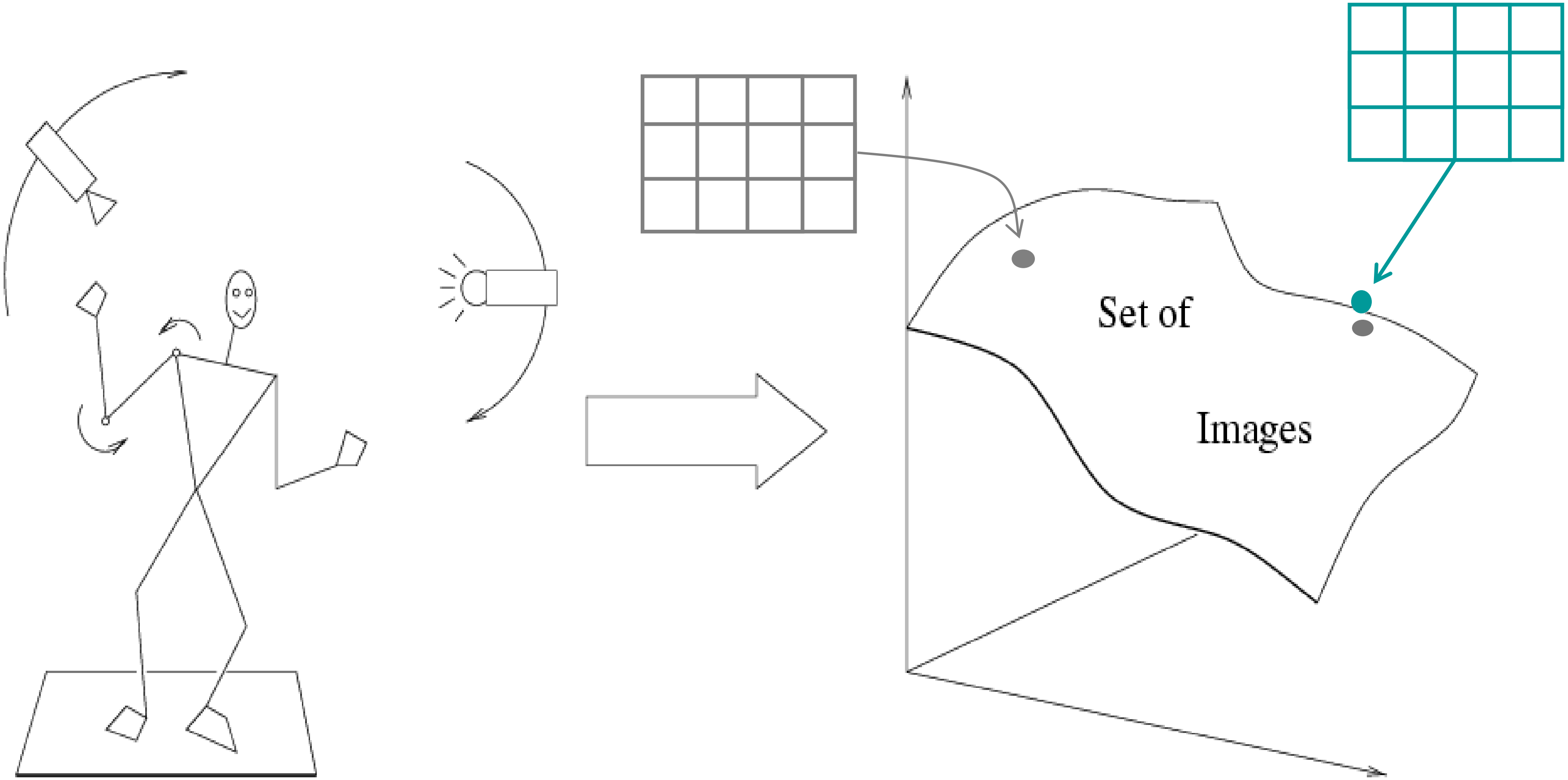
History of ideas in recognition

- 1960s – early 1990s: the geometric era

No digital cameras!
Slow compute!

- 1990s: appearance-based models

Slow compute!

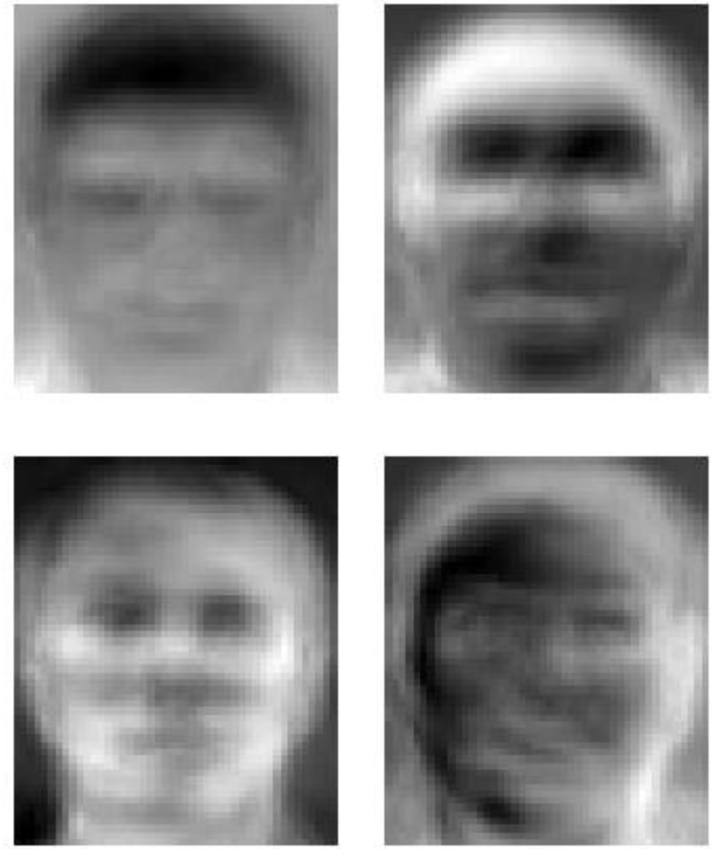


Empirical models of image variability

Appearance-based techniques

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

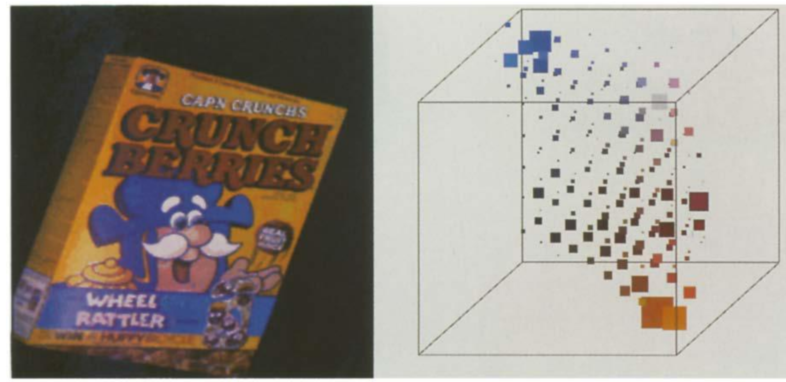
Eigenfaces (Turk & Pentland, 1991)

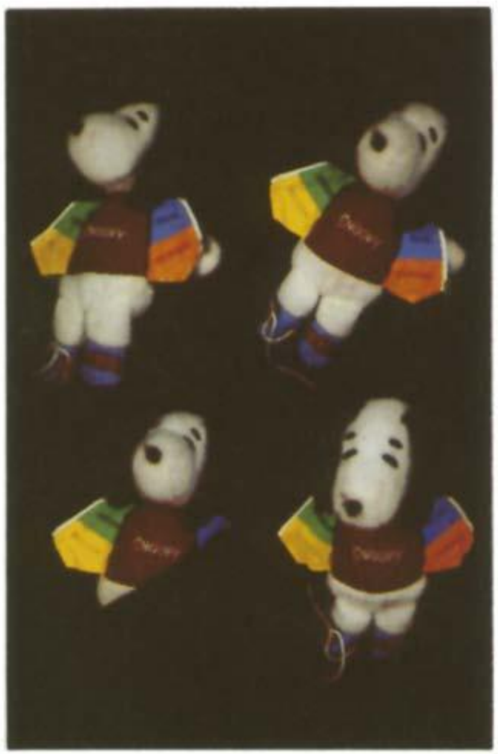


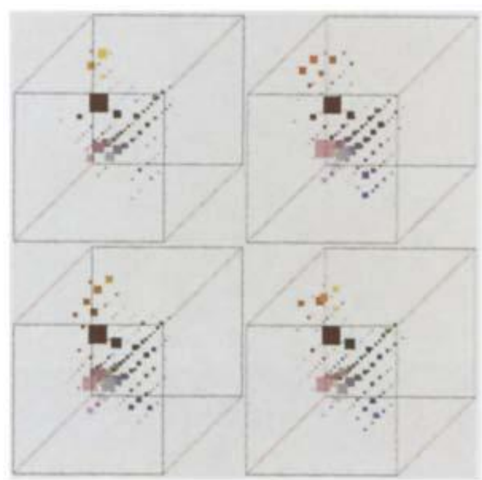
Experimental	Correct/Unknown Recognition Percentage		
Condition	Lighting	Orientation	Scale
Forced classification	96/0	85/0	64/0
Forced 100% accuracy	100/19	100/39	100/60
Forced 20% unknown rate	100/20	94/20	74/20

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!

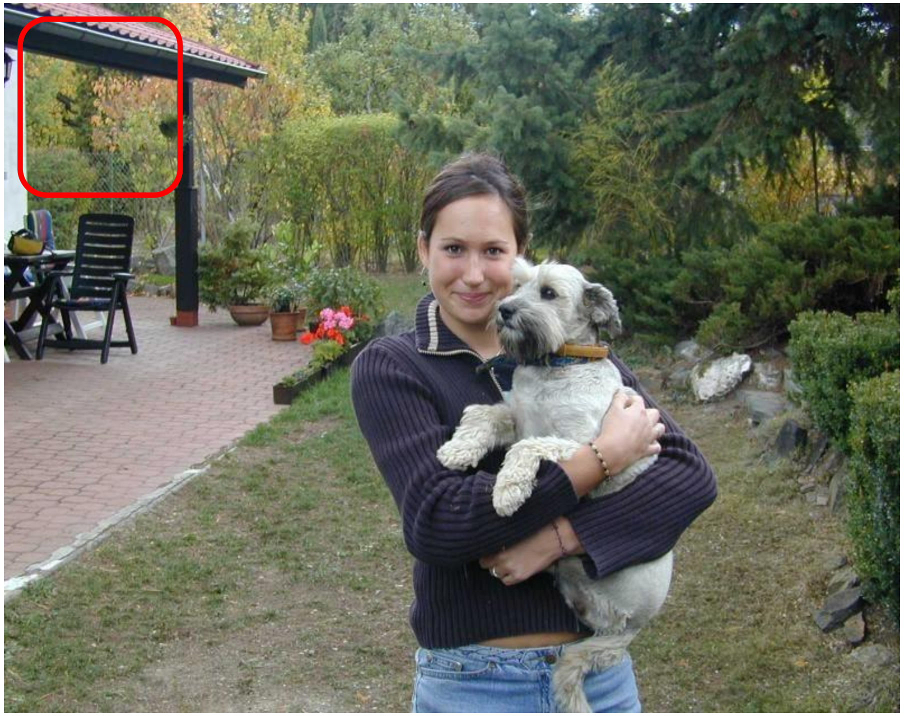
- 1990s: appearance-based models

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!

Local features for object instance recognition






D. Lowe (1999, 2004)

Large-scale image search

Combining local features, indexing, and spatial constraints

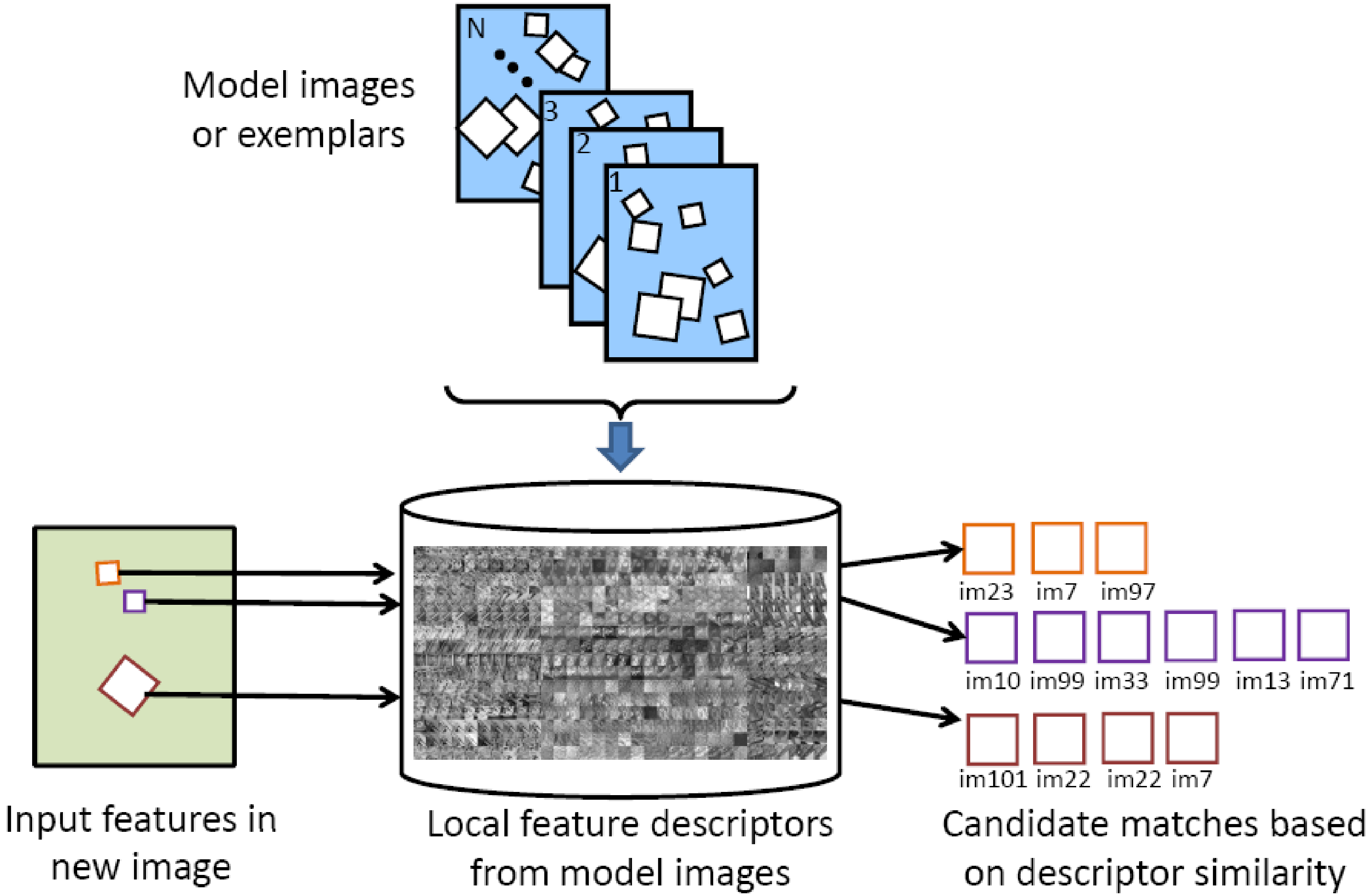
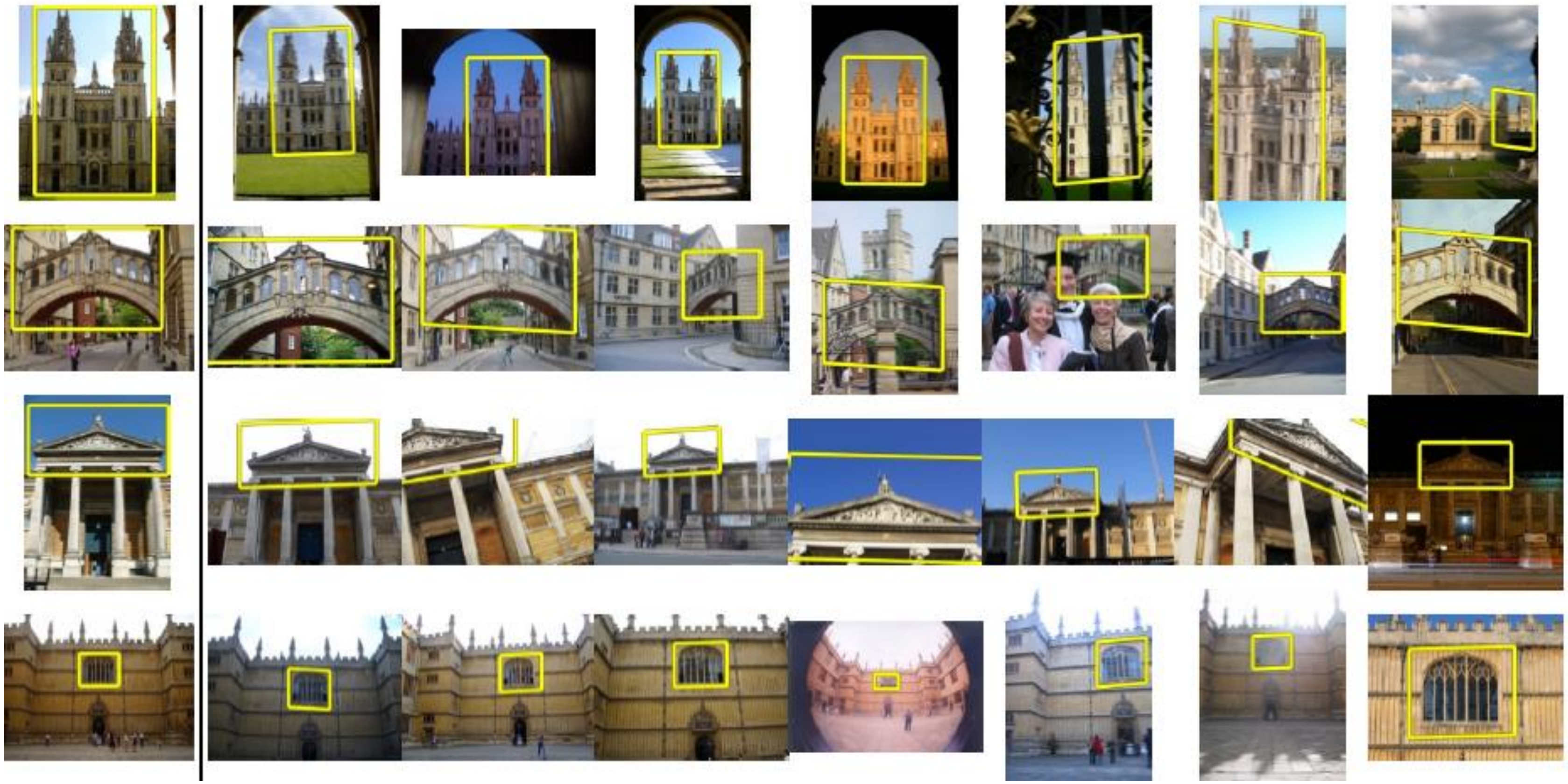


Image credit: K. Grauman and B. Leibe

Large-scale image search

Combining local features, indexing, and spatial constraints



Philbin et al. '07

Large-scale image search

Combining local features, indexing, and spatial constraints

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

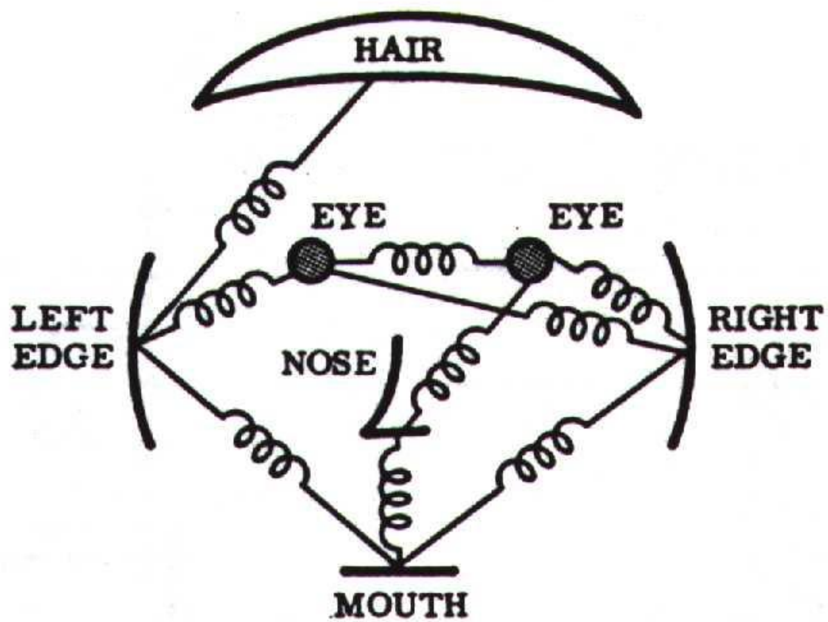


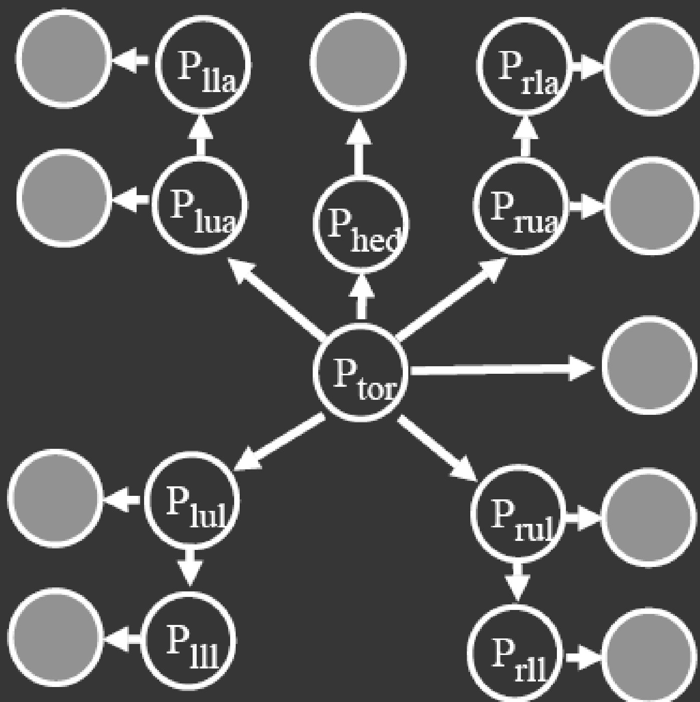
Figure from [Fischler & Elschlager 73]

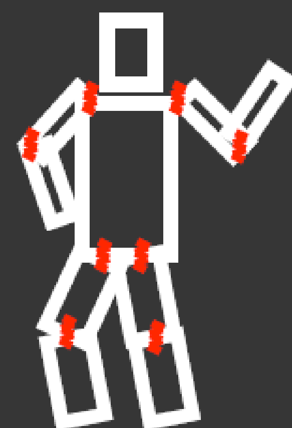
Constellation models

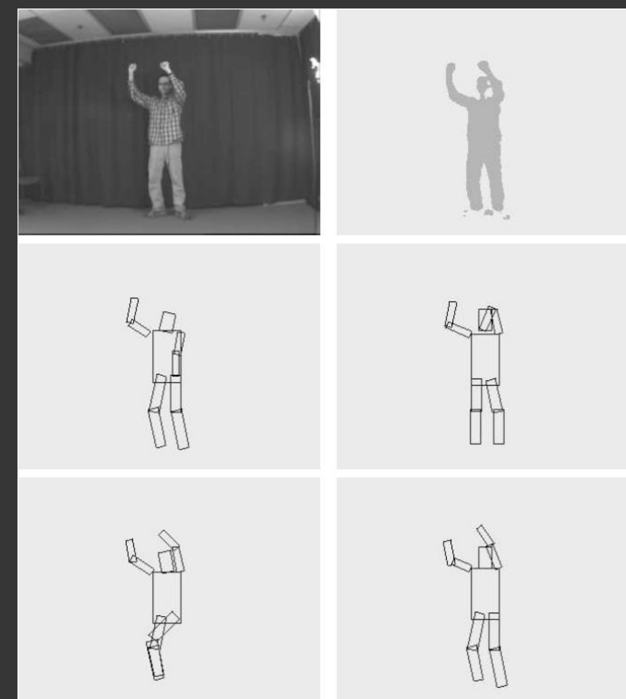
Weber, Welling & Perona (2000), Fergus, Perona & Zisserman (2003)

Pictorial structure model

Fischler and Elschlager(73), Felzenszwalb and Huttenlocher(00)





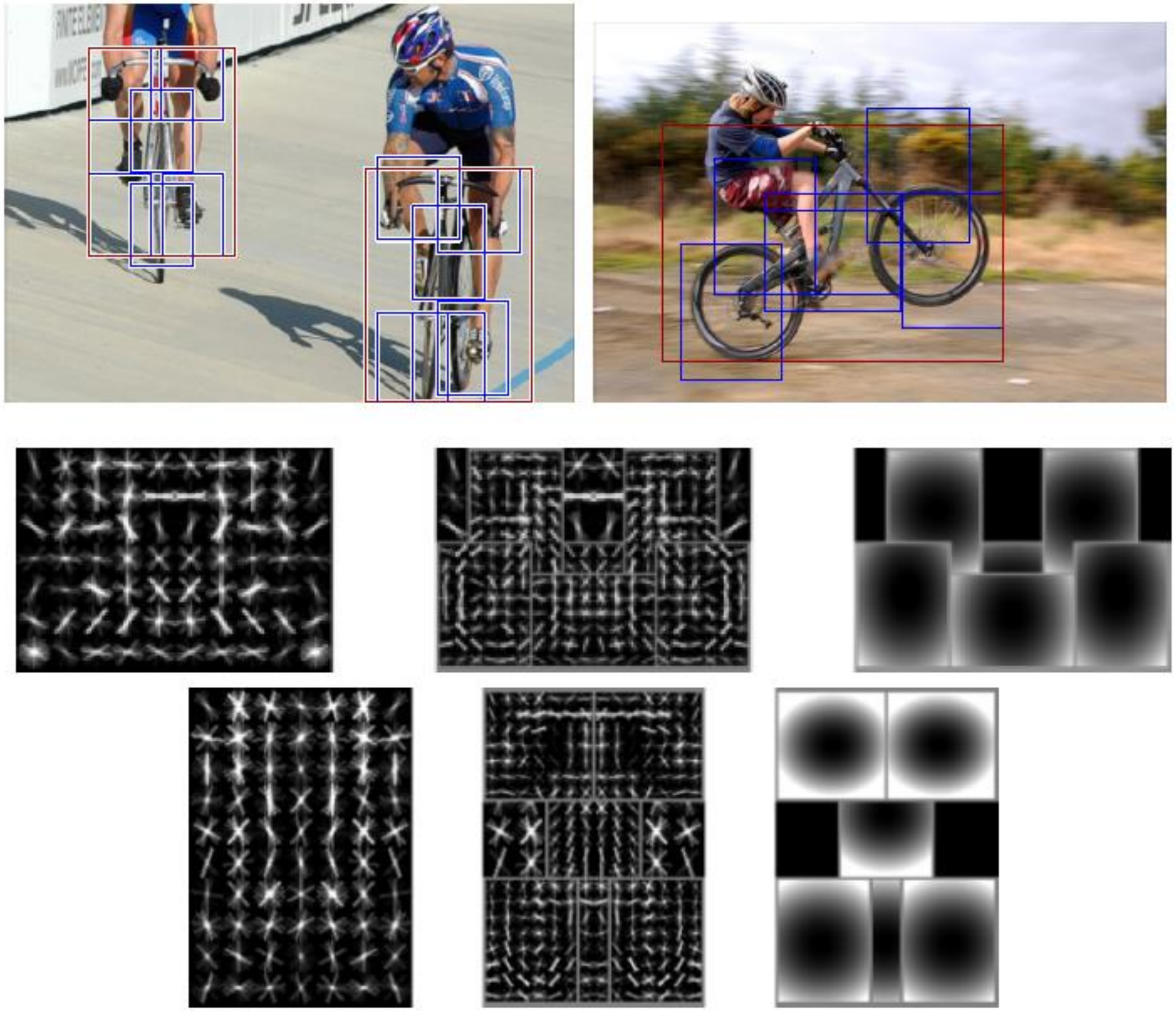


$$\Pr(P_{tor}, P_{arm}, \ldots | \mathrm{Im}) \propto \prod_{i,j} \Pr(P_i \mid P_j) \prod_i \Pr(\mathrm{Im}(P_i))$$

part geometry

part appearance

Discriminatively trained part-based models



P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan, PAMI 2009,

"**Object Detection with Discriminatively Trained Part-Based Models**"

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

Early GPU compute.

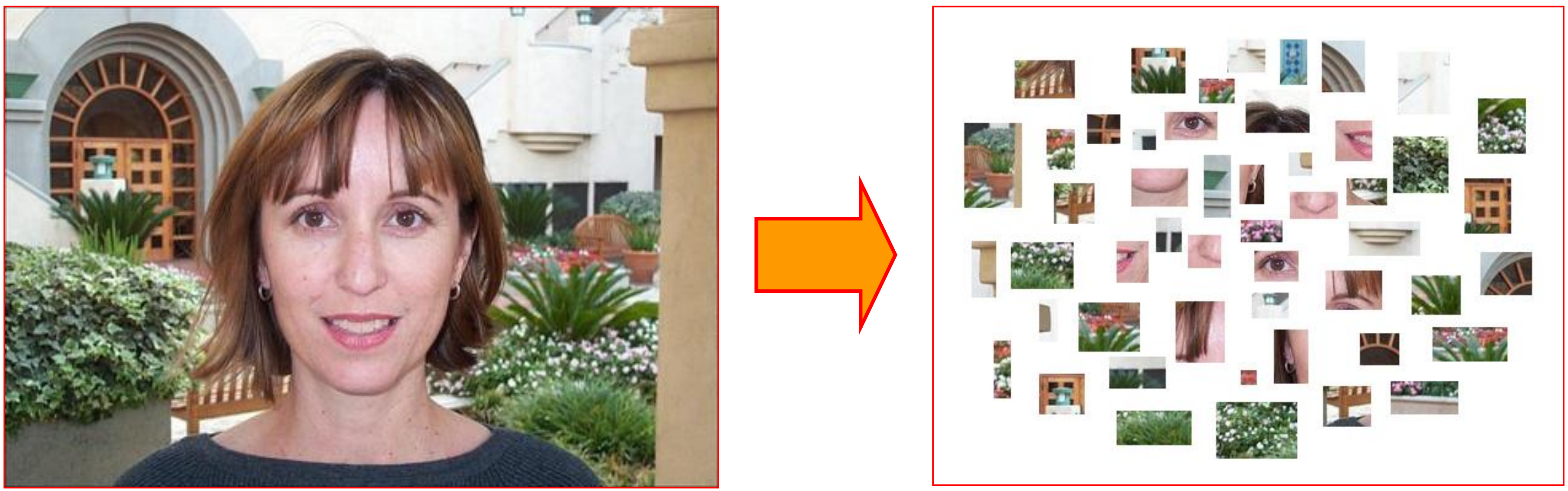
Svetlana Lazebnik

Bag-of-features models

Object → **Bag of ‘words’**

Svetlana Lazebnik

Bag-of-features models



Svetlana Lazebnik

Objects as texture

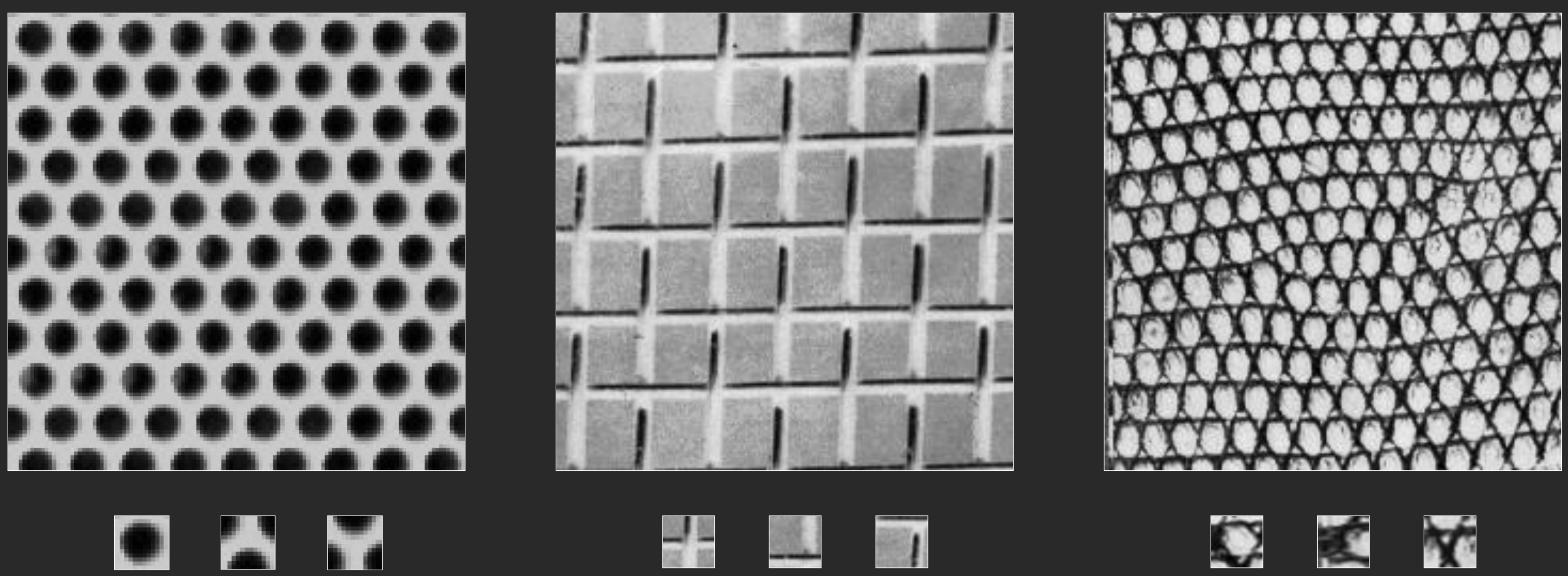
- All of these are treated as being the same

- No distinction between foreground and background: scene recognition?

Svetlana Lazebnik

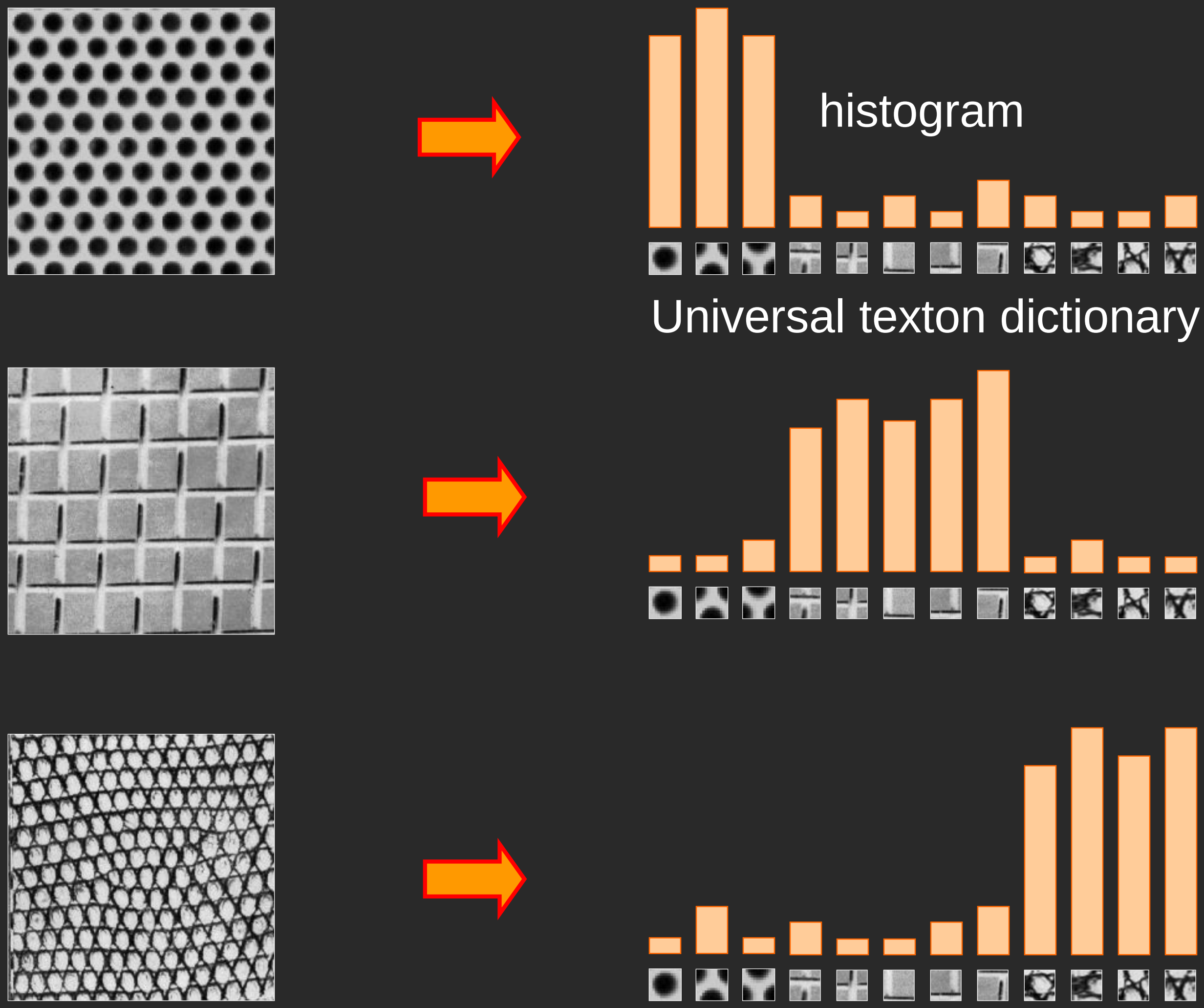
Origin 1: Texture recognition

- Texture is characterized by the repetition of basic elements or *textons*
- For stochastic textures, it is the identity of the textons, not their spatial arrangement, that matters



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

Origin 1: Texture recognition



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

Origin 2: Bag-of-words models

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

Origin 2: 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: 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/`

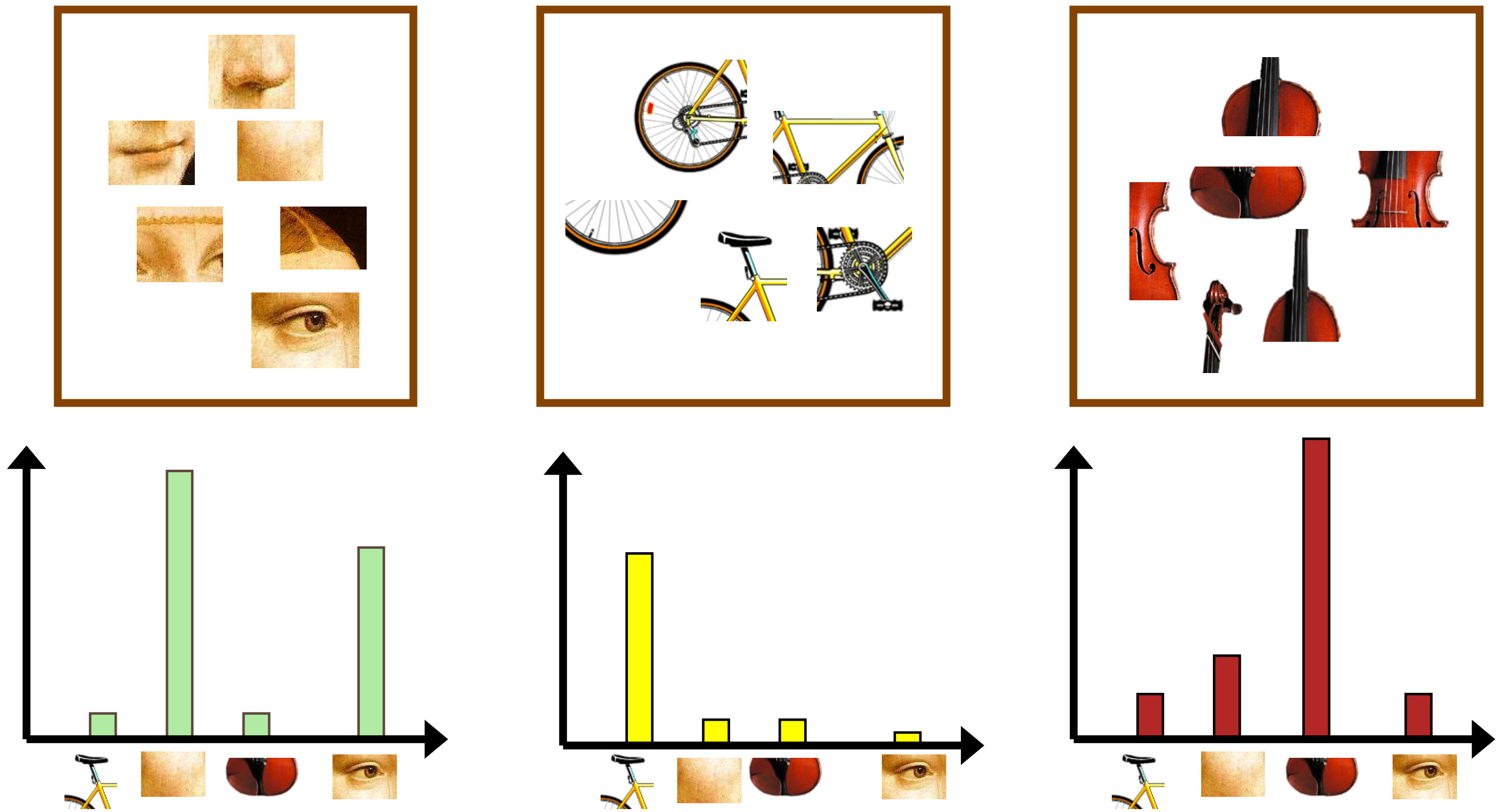
Origin 2: Bag-of-words models

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

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

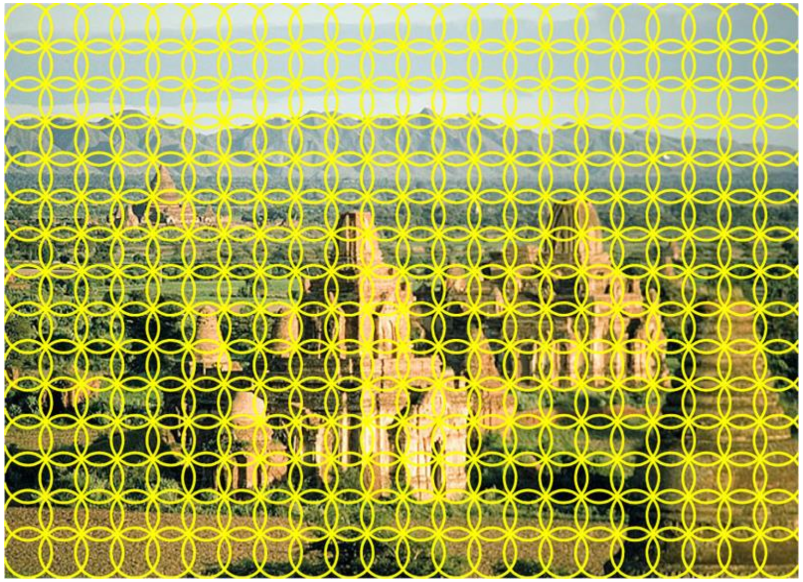
Bag-of-features steps

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

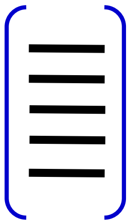


1. Feature extraction

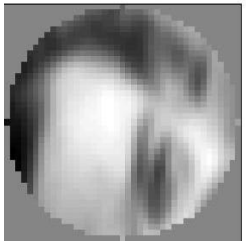
- Regular grid or interest regions



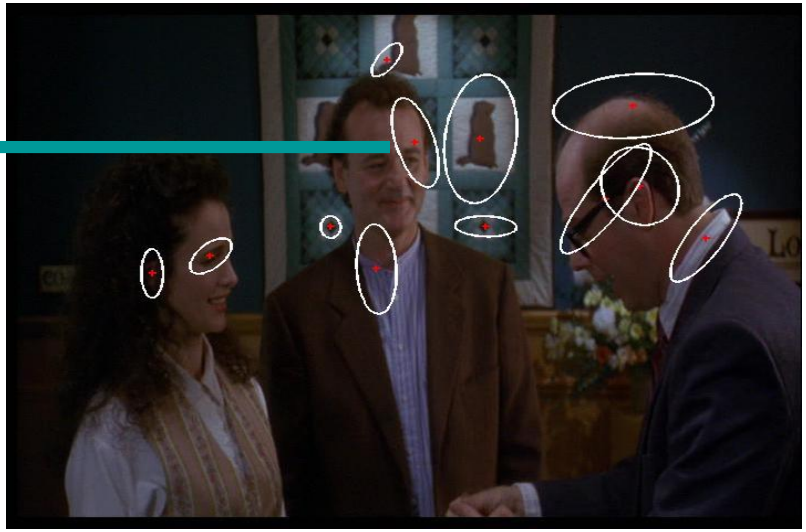
1. Feature extraction



Compute descriptor

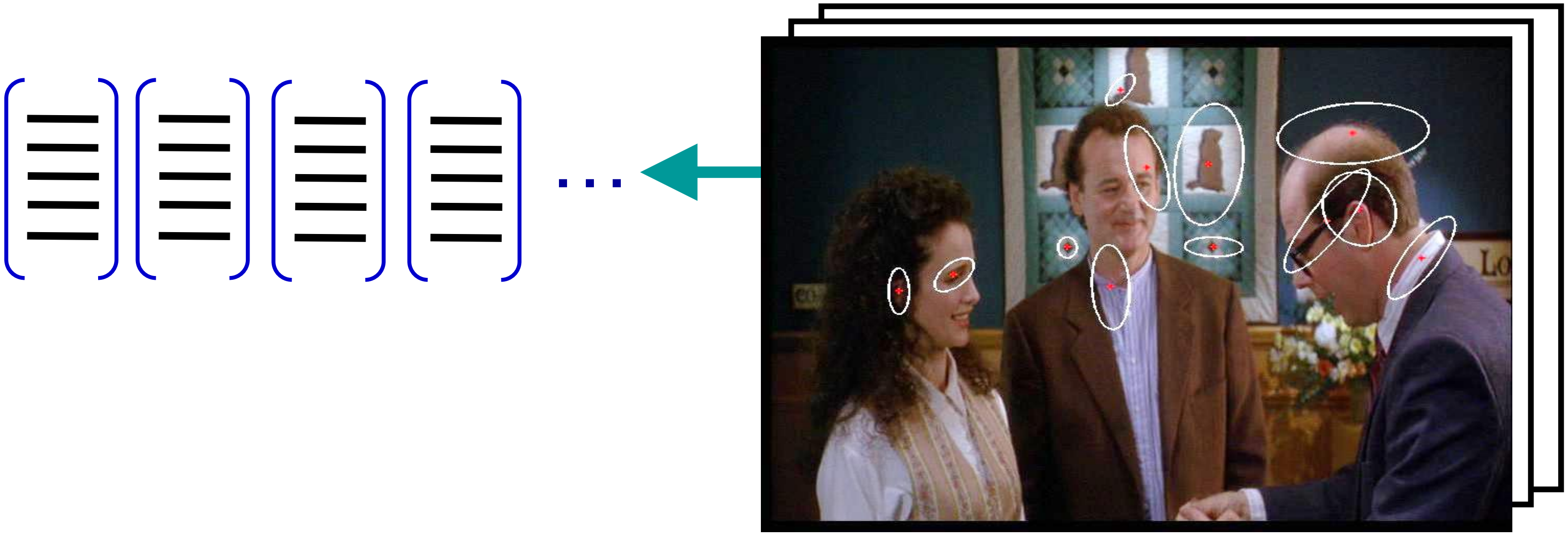


Normalize patch



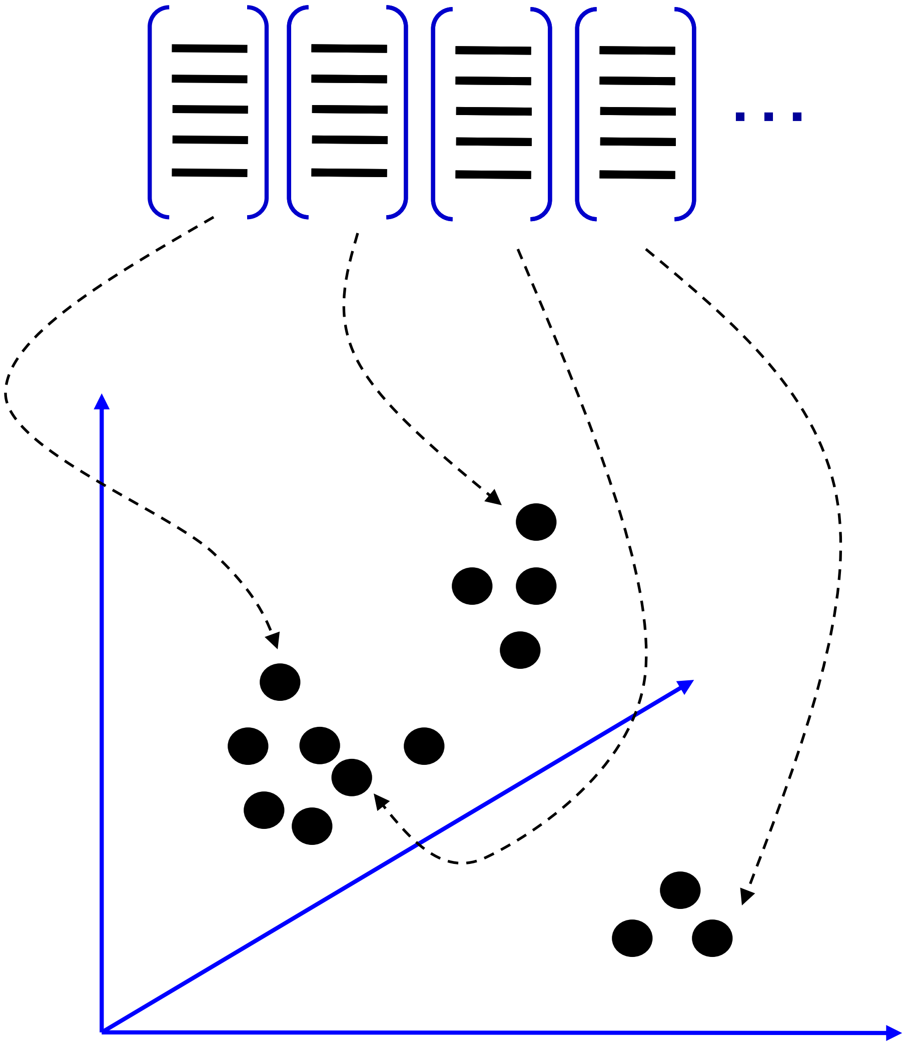
Detect patches

1. Feature extraction



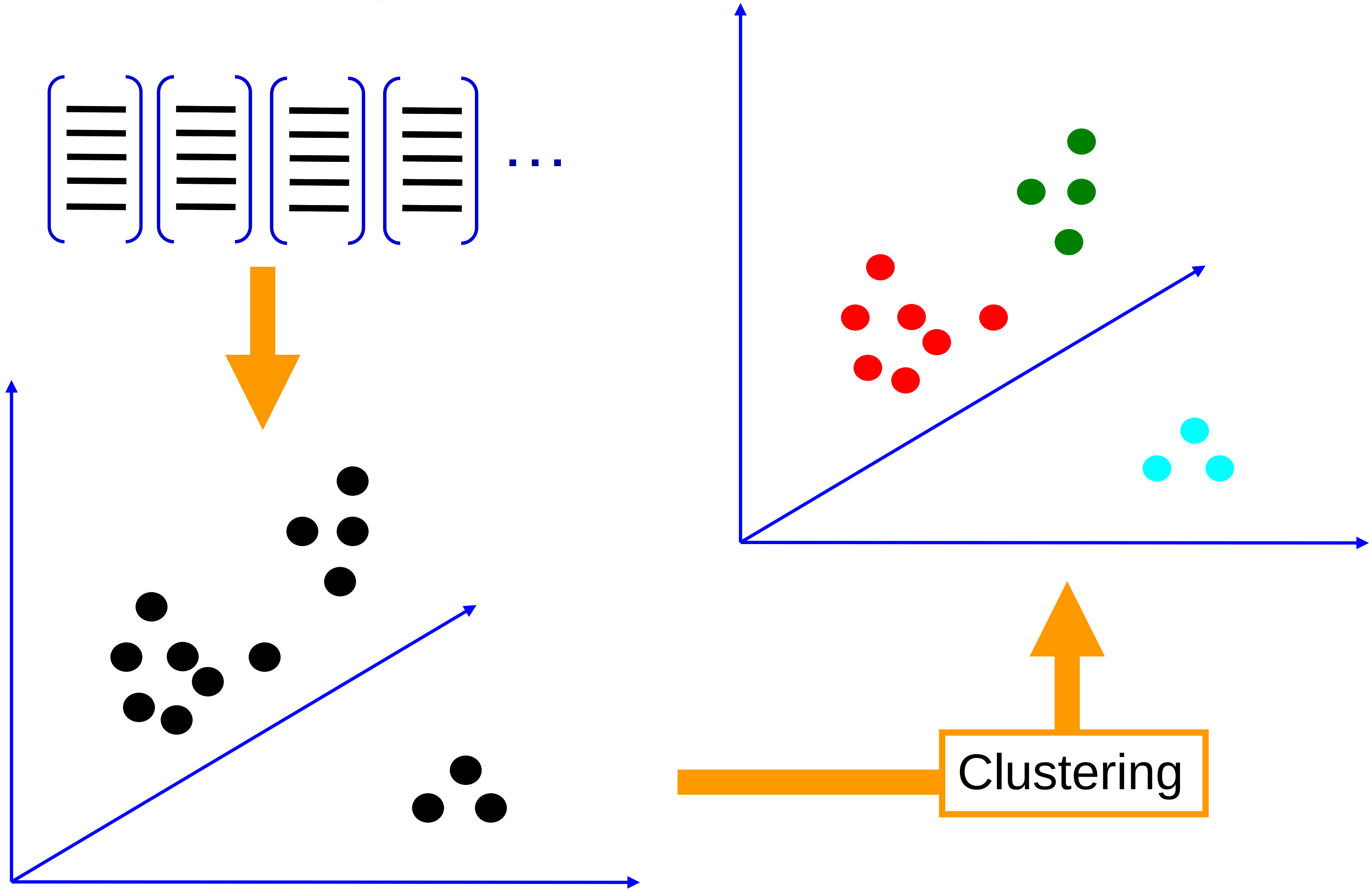
Slide credit: Josef Sivic

2. Learning the visual vocabulary



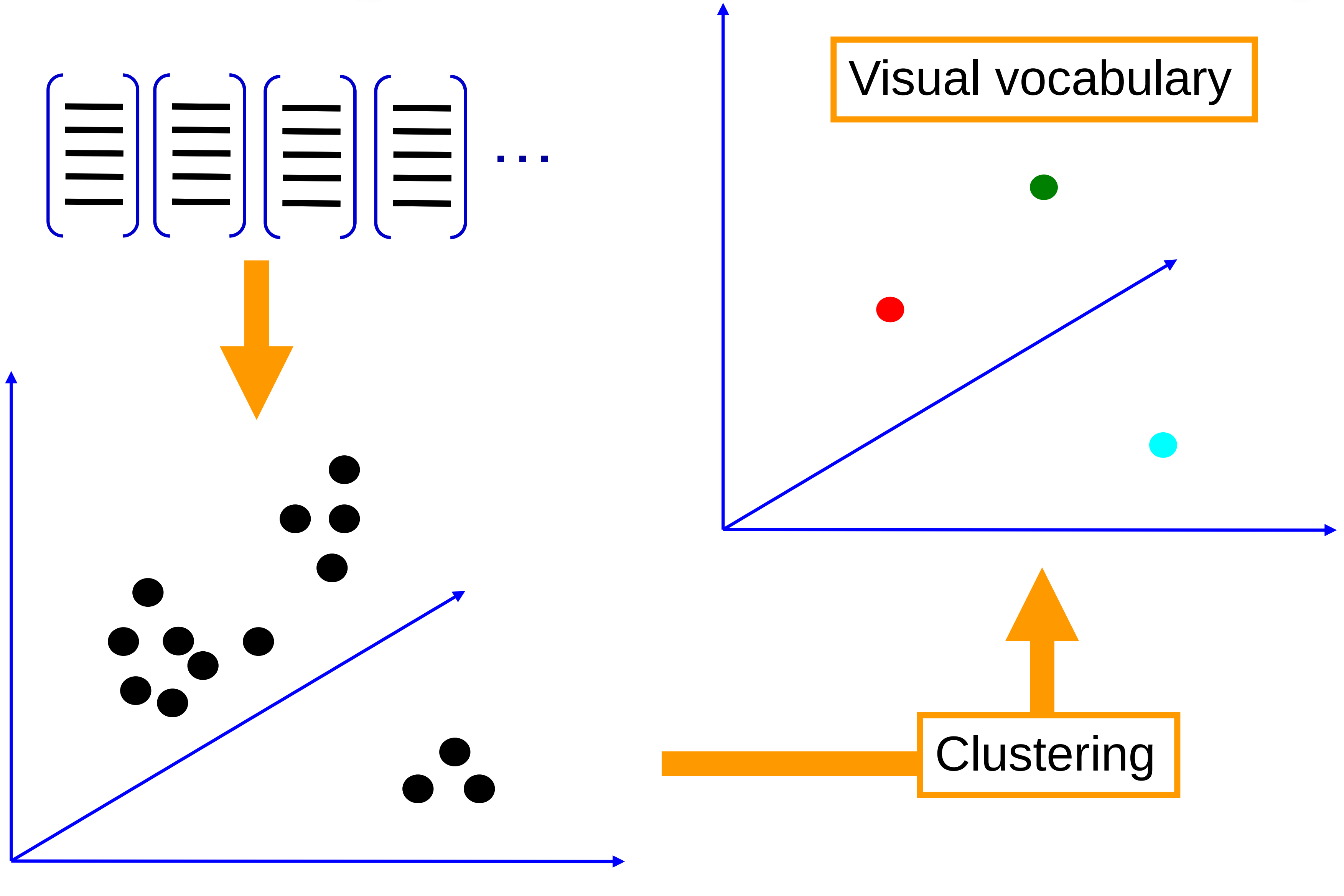
Slide credit: Josef Sivic

2. Learning the visual vocabulary



Slide credit: Josef Sivic

2. Learning the visual vocabulary



Slide credit: Josef Sivic

K-means clustering

Want to minimize sum of squared Euclidean distances between points x_i and their nearest cluster centers m_k :

$$D(X,M) = \sum_{\text{cluster } k} \sum_{\substack{\text{point } i \text{ in} \\ \text{cluster } k}} (x_i - m_k)^2$$

Algorithm:

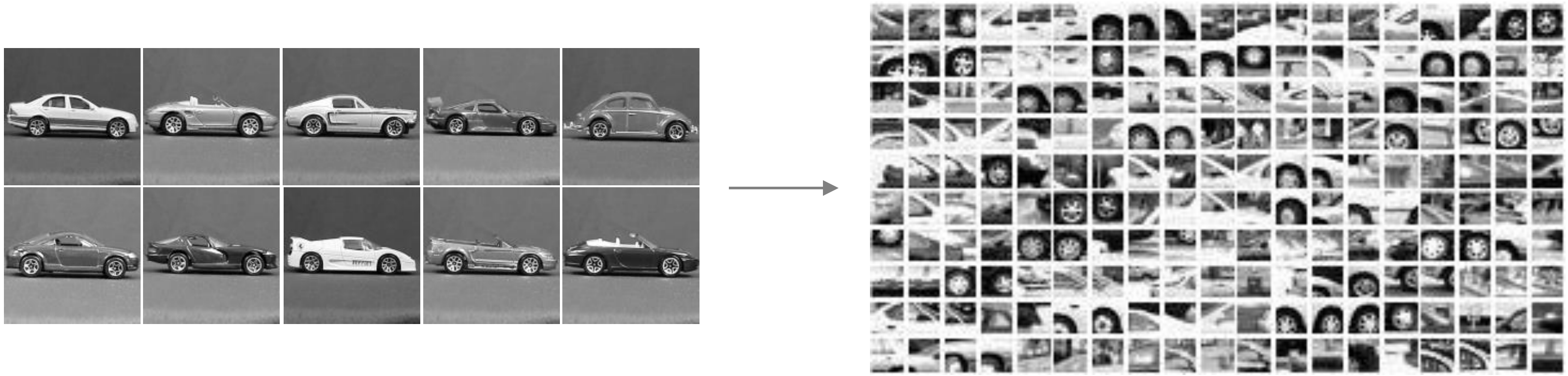
- Randomly initialize K cluster centers
- Iterate until convergence:
  - Assign each data point to the nearest center
  - Recompute each cluster center as the mean of all points assigned to it

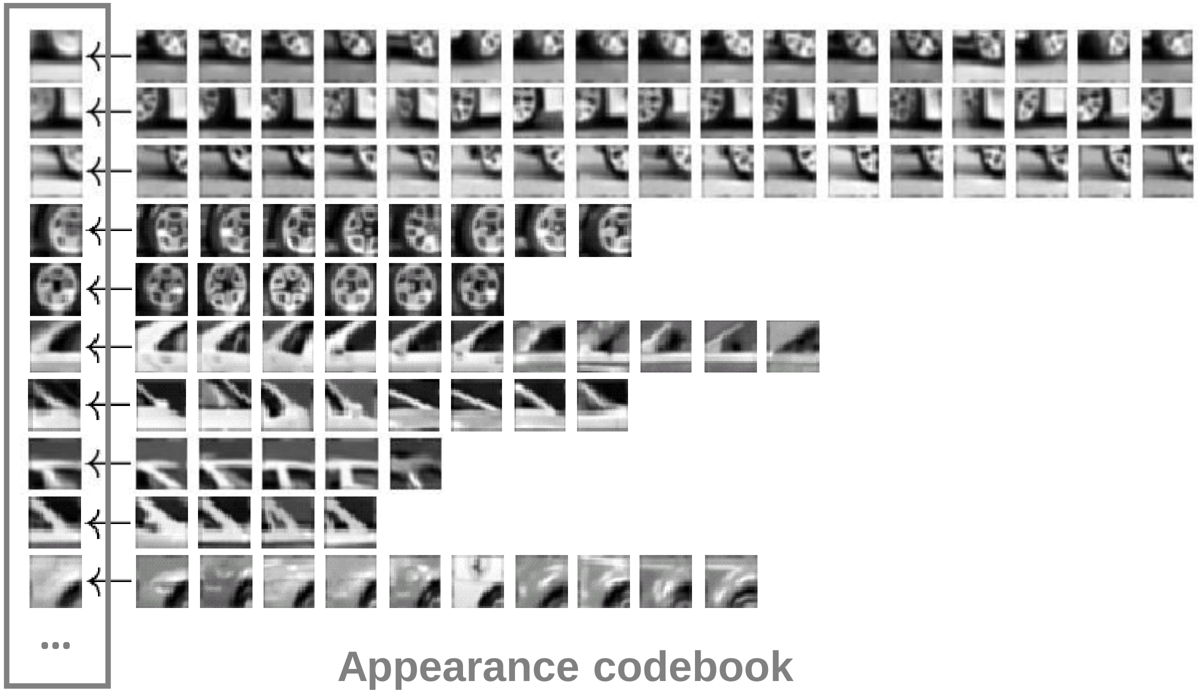
Clustering and vector quantization

- Clustering is a common method for learning a visual vocabulary or codebook
  - Unsupervised learning process
  - Each cluster center produced by k-means becomes a codevector
  - Codebook can be learned on separate training set
  - Provided the training set is sufficiently representative, the codebook will be “universal”

- The codebook is used for quantizing features
  - A *vector quantizer* takes a feature vector and maps it to the index of the nearest codevector in a codebook
  - Codebook = visual vocabulary
  - Codevector = visual word

Example codebook





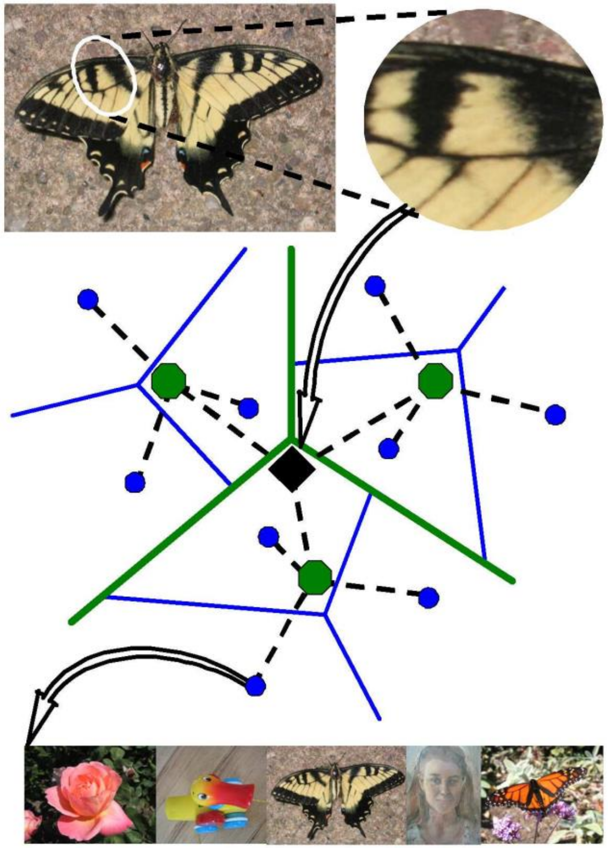
Appearance codebook

Source: B. Leibe

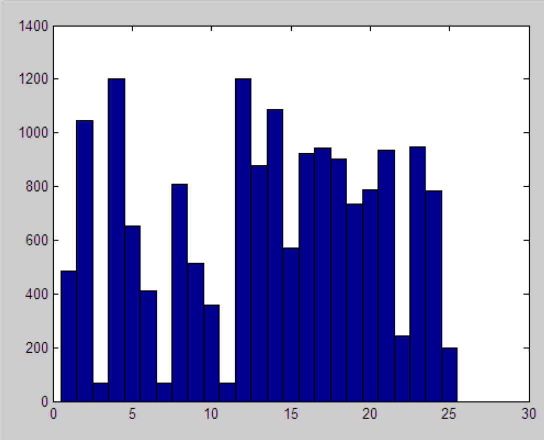
Visual vocabularies: Issues

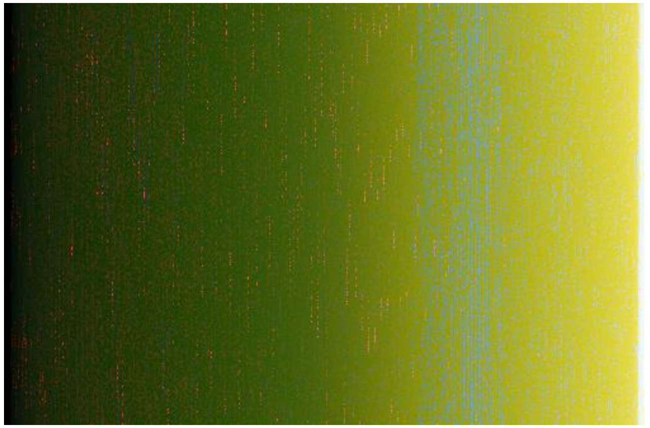
- How to choose vocabulary size?
  - Too small: visual words not representative of all patches
  - Too large: quantization artifacts, overfitting

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



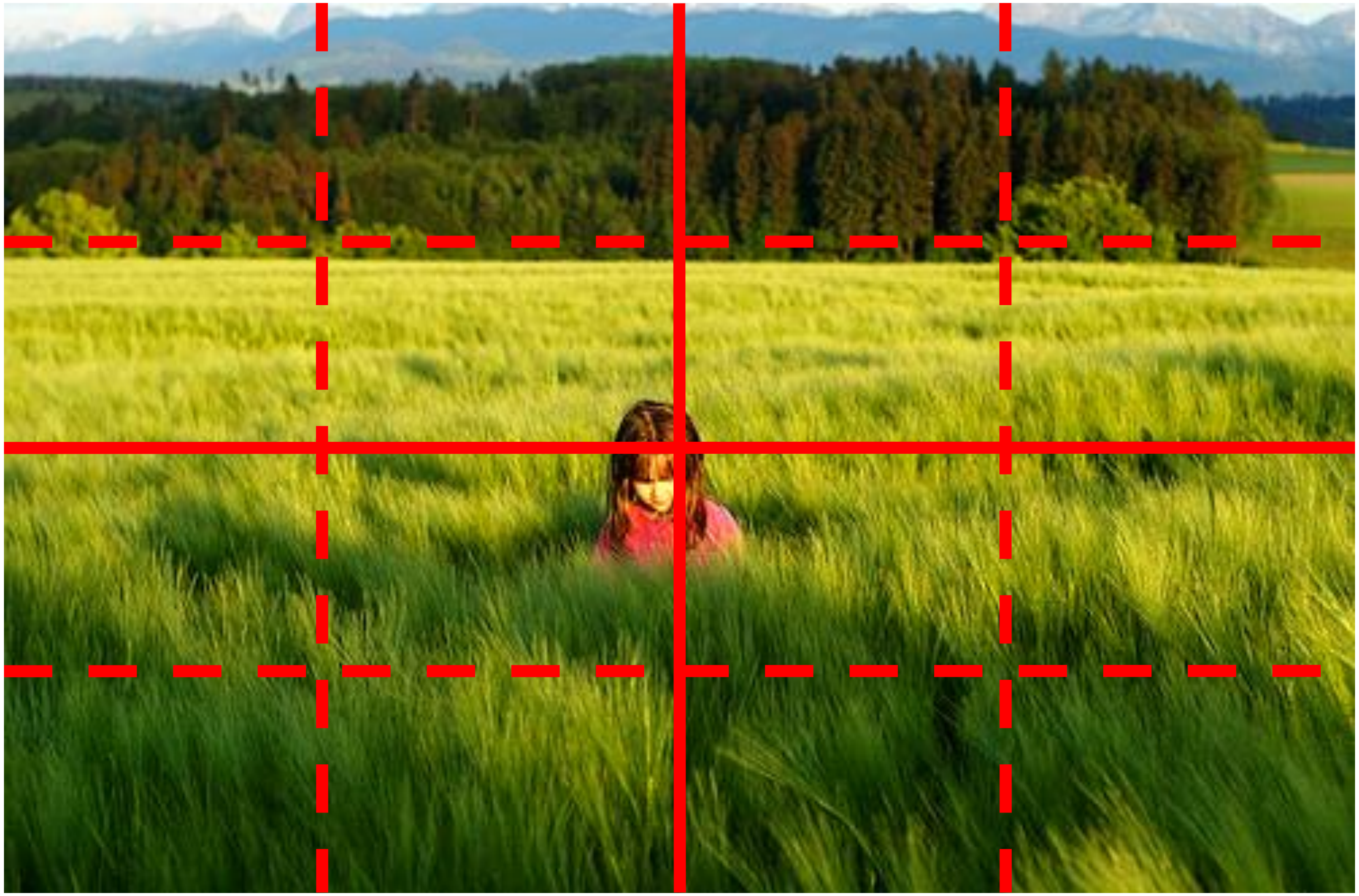
But what about layout?





All of these images have the same color histogram

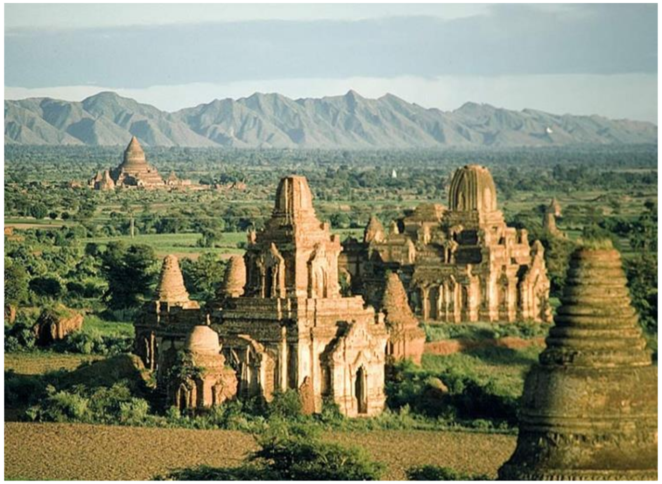
Spatial pyramid

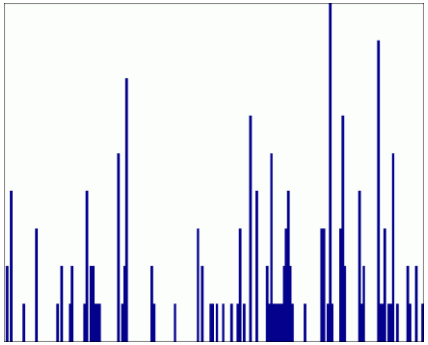


Compute histogram in each spatial bin

Spatial pyramid representation

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



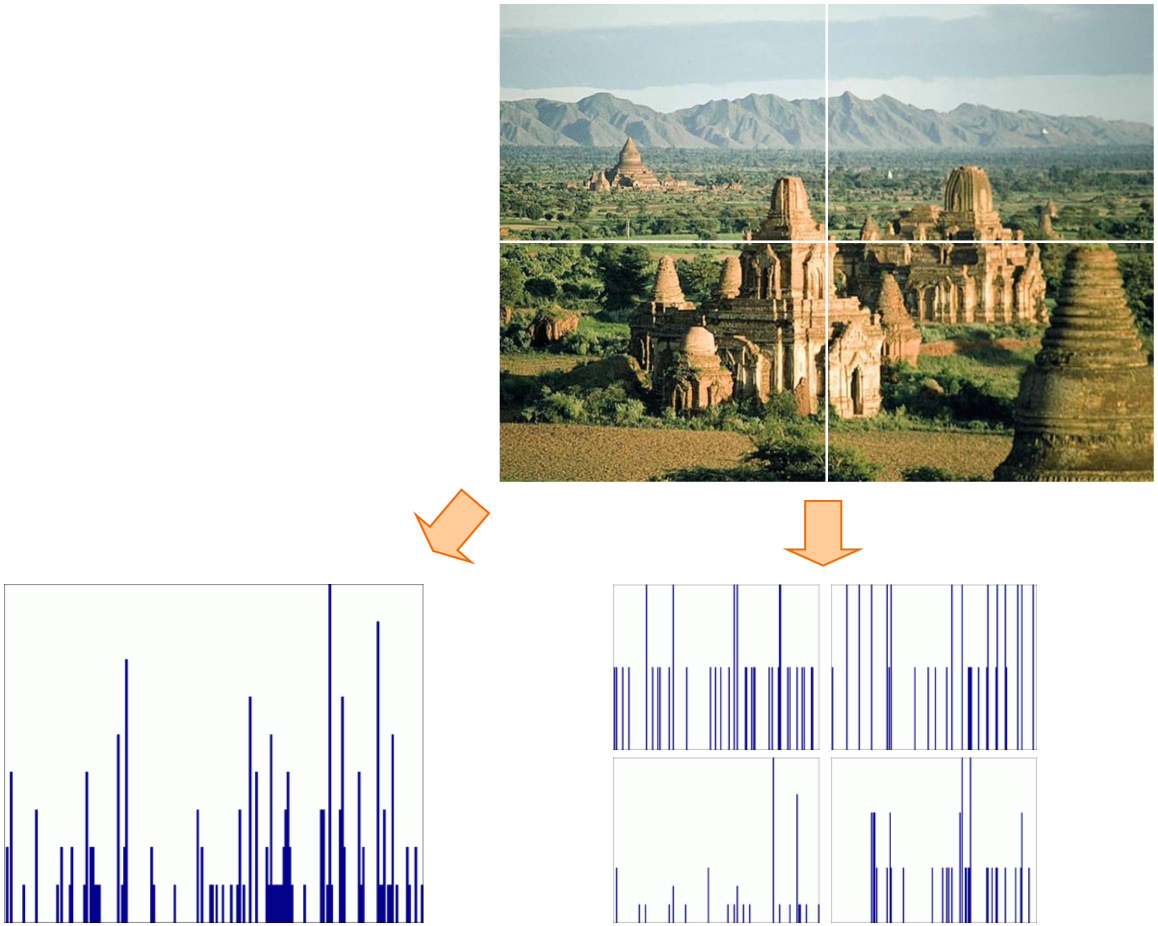


level 0

Lazebnik, Schmid & Ponce (CVPR 2006)

Spatial pyramid representation

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



level 0 level 1

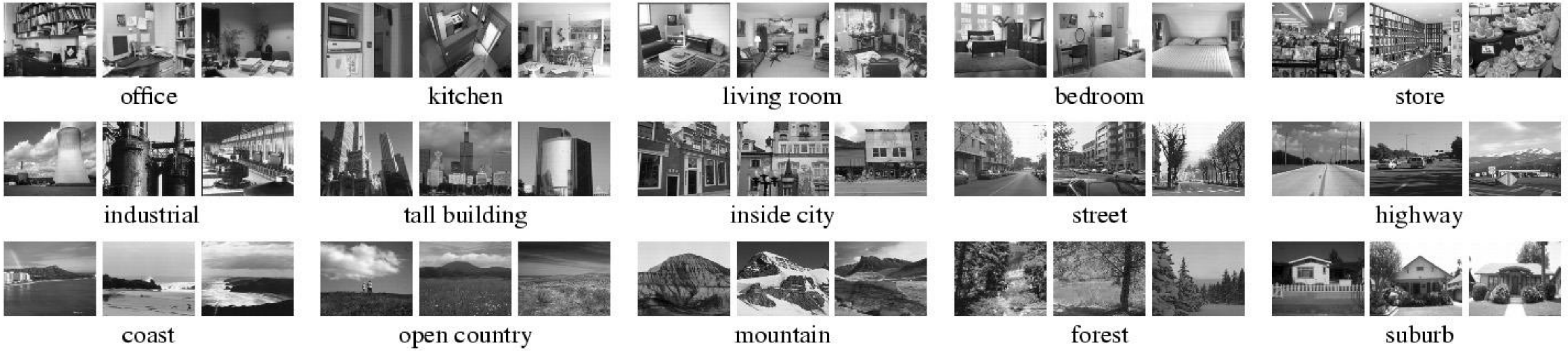
Lazebnik, Schmid & Ponce (CVPR 2006)

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

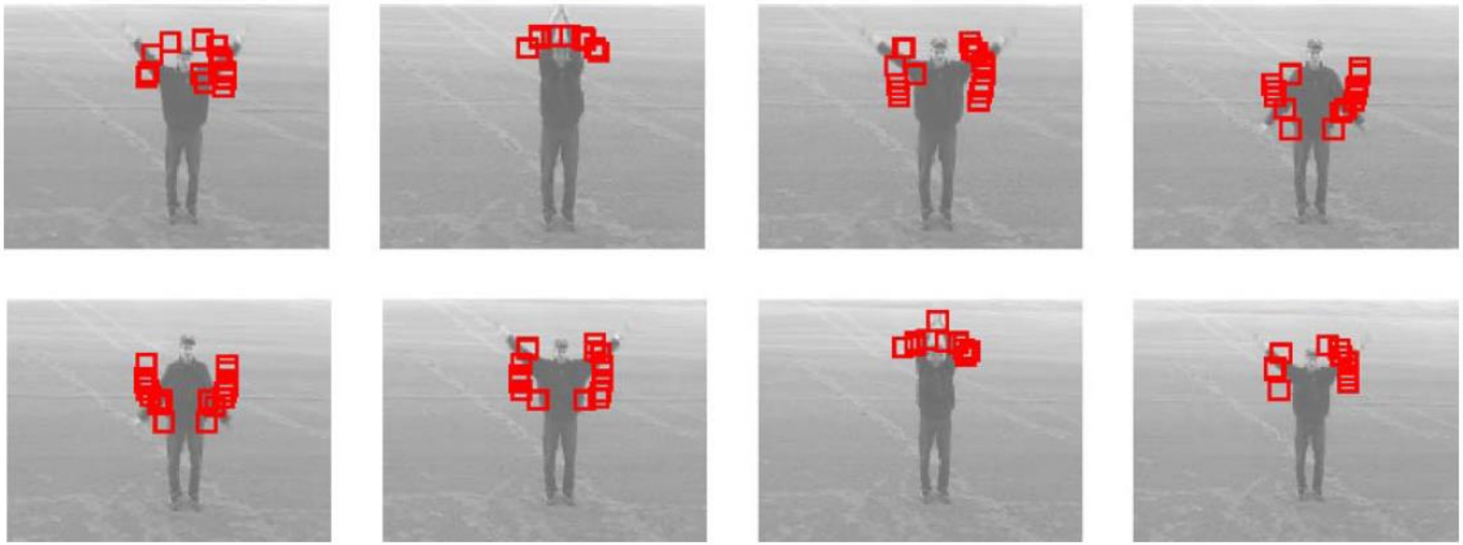


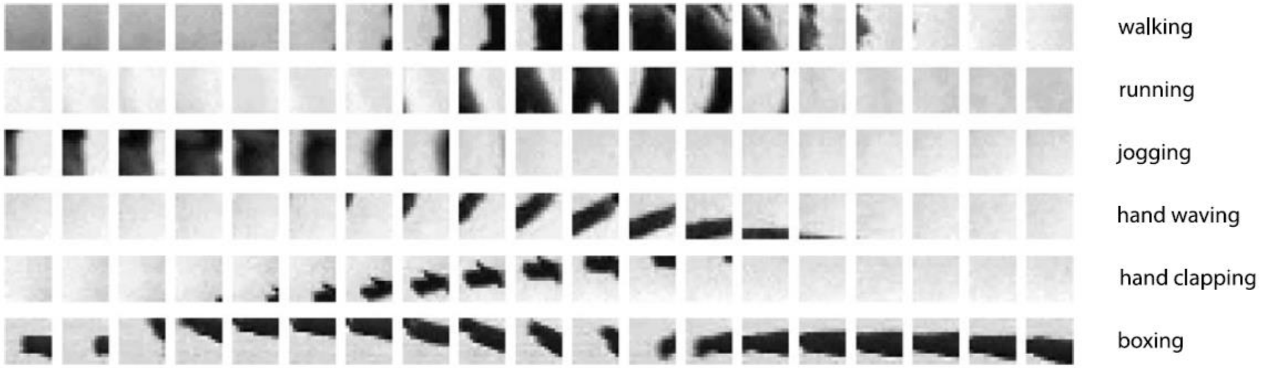
Multi-class classification results (100 training images per class)

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

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.

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 — Early GPU compute.
- Present trends: combination of local and global methods, data-driven methods, context, deep learning

GPU/cloud compute.

Svetlana Lazebnik