ECS797– Machine Learning for Visual Analysis

Ioannis Patras *i.patras@qmul.ac.uk*
Tao Xiang *t.xiang@qmul.ac.uk*

Lecture 1, Wednesday 11/01/17

General Information

Course objectives

- How to apply the basic principles and techniques learned in ECS708 Machine learning and ECS709 Introduction to Computer Vision to solve real world computer vision problems
  - What are the computer vision problems one wants to solve
  - What are the main methodologies
  - What are the common models and algorithms
- How to build a computer vision system
  - How to design a computer vision system
  - How to implement the system
    - Matlab
    - Be resourceful
  - How to evaluate the system

Why?

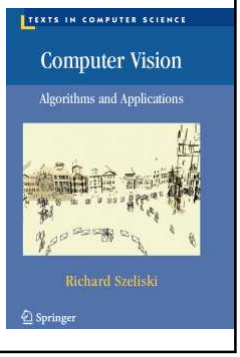
- Computer vision is at a critical stage with a great potential for applications
- It poses a great intellectual challenge
- A lot of useful models and tools will be learned, which can land you a job in many related fields

Course structure

- Week 1: - Overview of visual recognition + representation
- Week 2: - Classifier + Image categorisation
- Week 3: - Face recognition
- Week 4: - Face detection
- Week 5: - General object detection
- Week 6: - Further face analysis
- Week 7: - Reading week
- Week 8: - Action Recognition
- Week 9: - Tracking
- Week 10: - Paper reading (deep learning)
- Week 11: - Paper reading (pose estimation)
- Week 12: - Revision

Course resources

- **Lectures**: Wed 12:00-14:00 Queens EB1
- **Labs**: One two-hour session: 9-11 Thursdays, ITL 2F;
  - Weeks 2, 3,4 and then assessment in week 6 for the first two courseworks
  - Weeks 8, 9,10 and then assessment in week 12 for the last two courseworks
  - Lab sheets can be found at the course webpage
- **Textbook**: Computer Vision: Algorithms and Applications online free version, by Richard Szeliski, Springer, Amazon link.
- **Course website**: http://qmplus.qmul.ac.uk/course/view.php?id=9559
  Forum: follow the link on the course website



Course staff

- Lecturers
  - Dr. Ioannis Patras: i.patras@qmul.ac.uk
  - Dr. Tao Xiang: t.xiang@qmul.ac.uk, office hour 2-3, Fridays, office CS/324
- Lab Assistants (TAs):
  - TBC
- Methods of communication:
  - Forum
  - Email us/TAs directly
  - Talk to us during break, after lecture and in office hour

Assessment

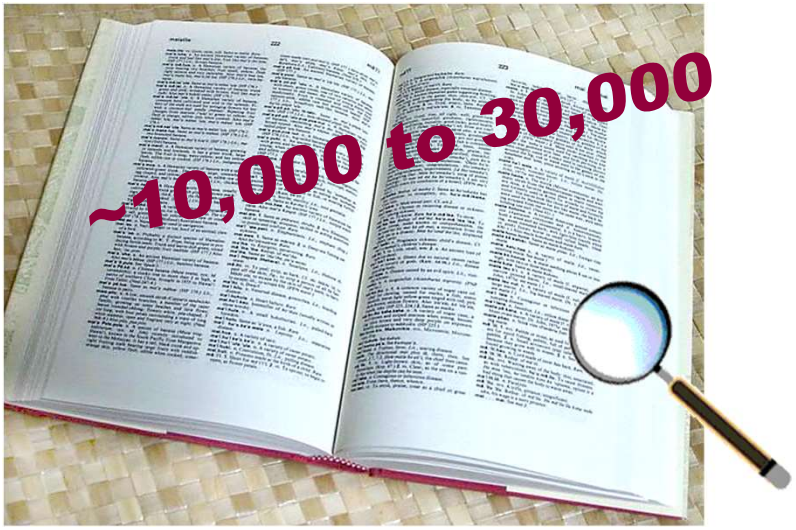
- Final exam in May: 60%
- Four courseworks (mini-projects): 40% (10% each)
  - CW1: Image categorisation
  - CW2: Face recognition
  - CW3: Human age estimation
  - CW4: Face detection and tracking
- CW1+2: report and in-lab assessment in Week 6
- CW3+4: report and in-lab assessment in Week 12

Important notes

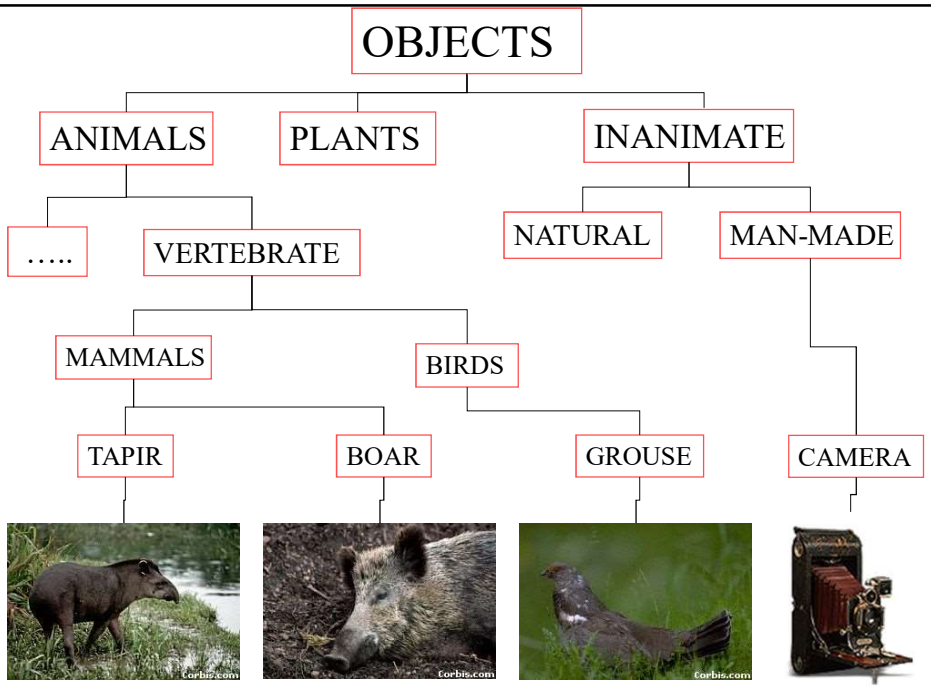
- Attend Lectures
- Attend Labs and also spend extra time on reading materials and practice
- Ask questions
- Visit the course website and discussion forum regularly
- Plagiarism cases will be handled seriously

VISUAL RECOGNITION: AN OVERVIEW

How many visual object categories are there?

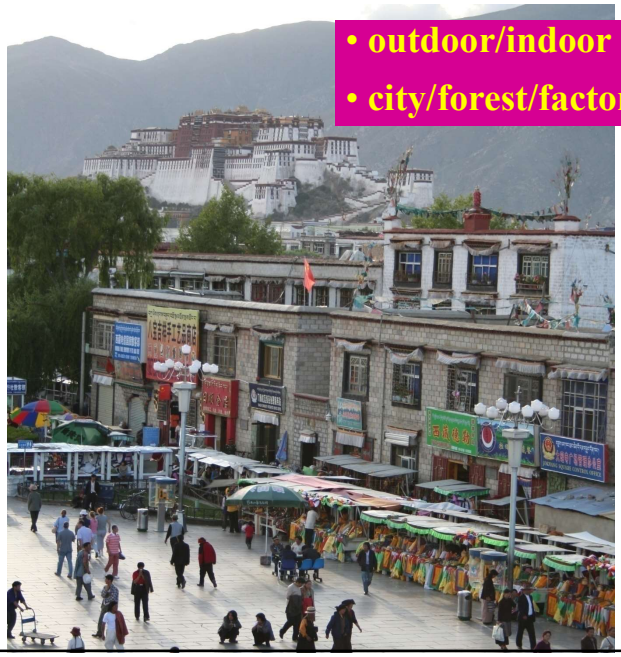


Biederman 1987



Specific recognition tasks

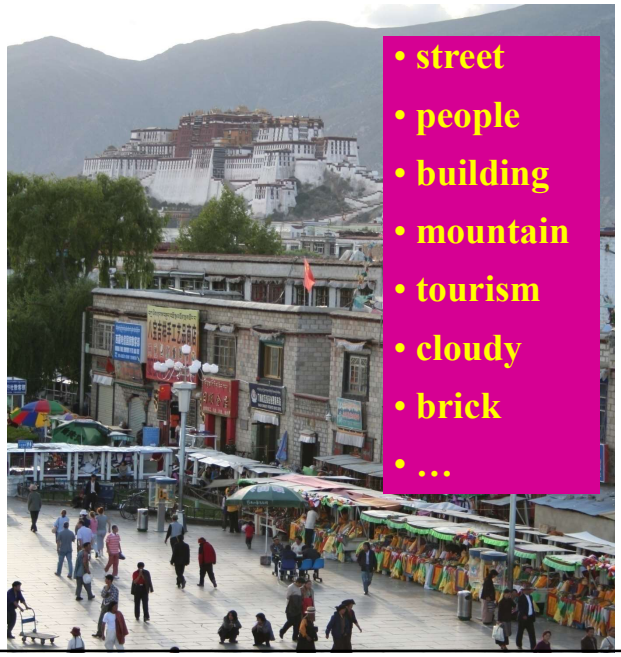
Scene categorization or classification



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

Svetlana Lazebnik

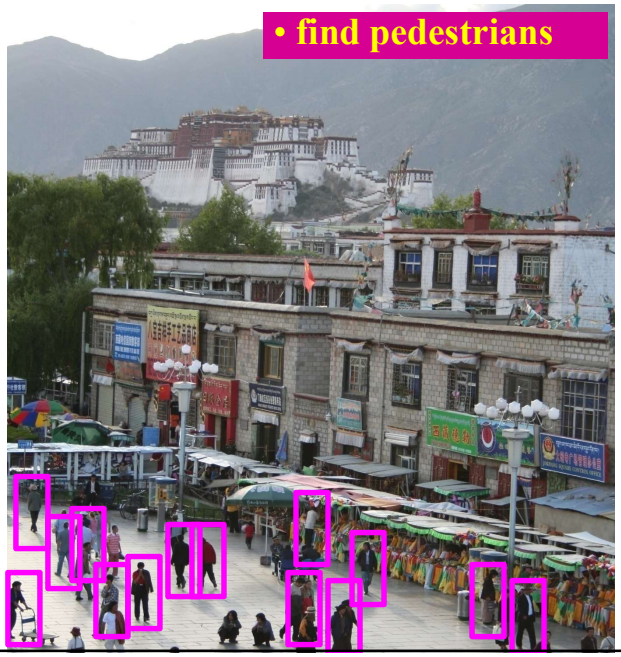
Image annotation / tagging / attributes



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

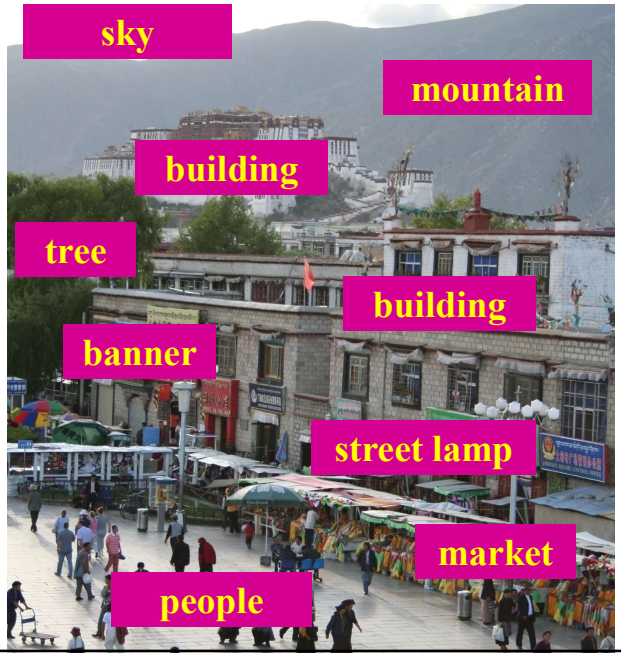
Svetlana Lazebnik

Object detection



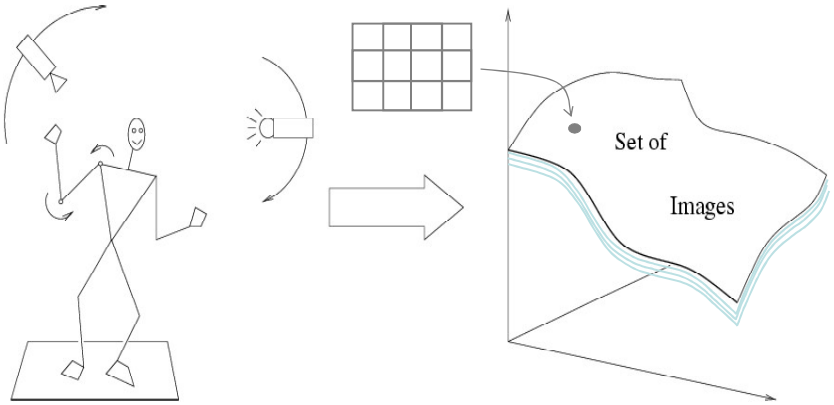
Svetlana Lazebnik

Image parsing



Svetlana Lazebnik

Recognition is all about modeling variability



Variability: Camera position
Illumination
Shape parameters

Within-class variations?

Svetlana Lazebnik

Within-class variations

Svetlana Lazebnik

History of ideas in recognition

- 1960s – early 1990s: the geometric era

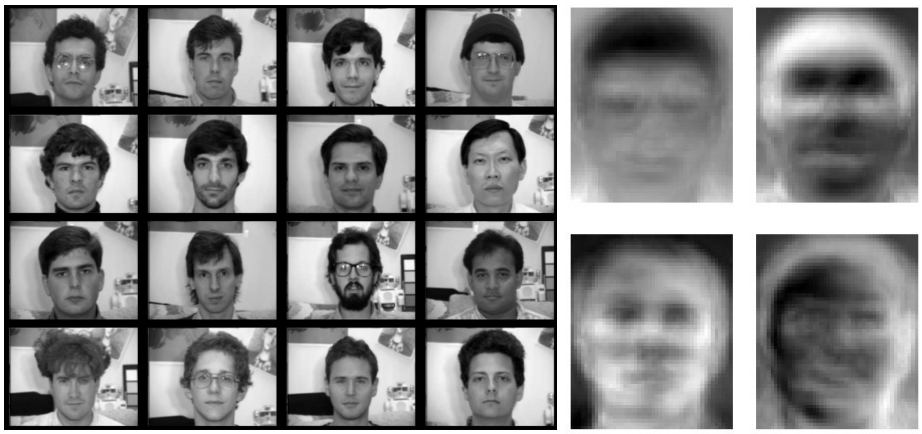
Svetlana Lazebnik

History of ideas in recognition

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

Svetlana Lazebnik

Eigenfaces (Turk & Pentland, 1991)



Experimental Condition	Correct/Unknown Recognition Percentage		
	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

Limitations of global appearance models

- Requires global registration of patterns
- Not robust to clutter, occlusion, geometric transformations

Svetlana Lazebnik

History of ideas in recognition

- 1960s – early 1990s: the geometric era
- 1990s: appearance-based models
- 1990s – present: sliding window approaches

Svetlana Lazebnik

Sliding window approaches

- Turk
- Bell
- 199
- Schn

- Schneide
- Agarwal
- Poggio e

History of ideas in recognition

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

Svetlana Lazebnik

Local features for object instance recognition

D. Lowe (1999, 2004)

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

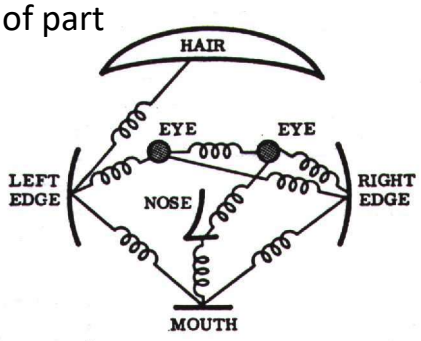


Figure from [Fischler & Elschlager 73]

Constellation models

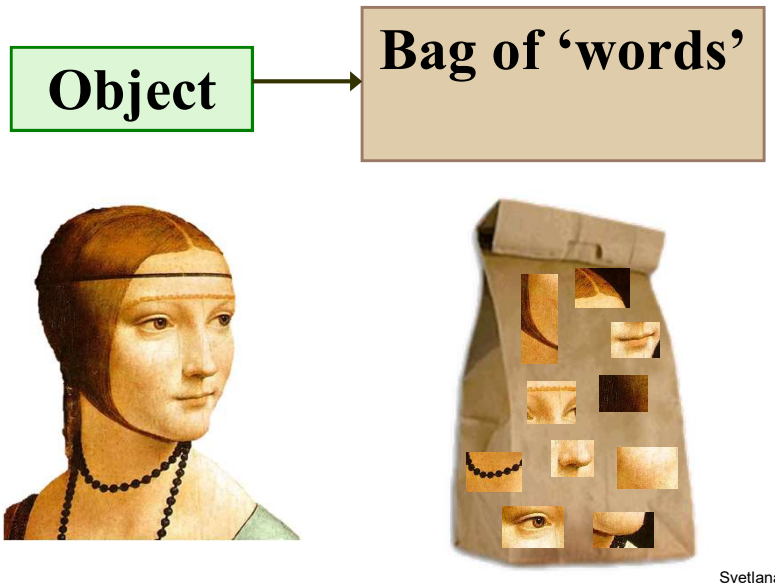
Weber, Welling & Perona (2000), Fergus, Perona & Zisserman (2003)

History of ideas in recognition

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

Svetlana Lazebnik

Bag-of-features models



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
- Mid-2000s: bags of features
- Present trends: deep learning rules

Svetlana Lazebnik

Deep learning is everywhere

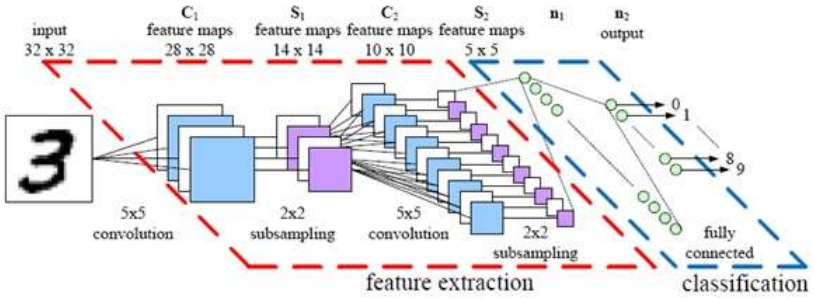


IMAGE CATEGORIZATION

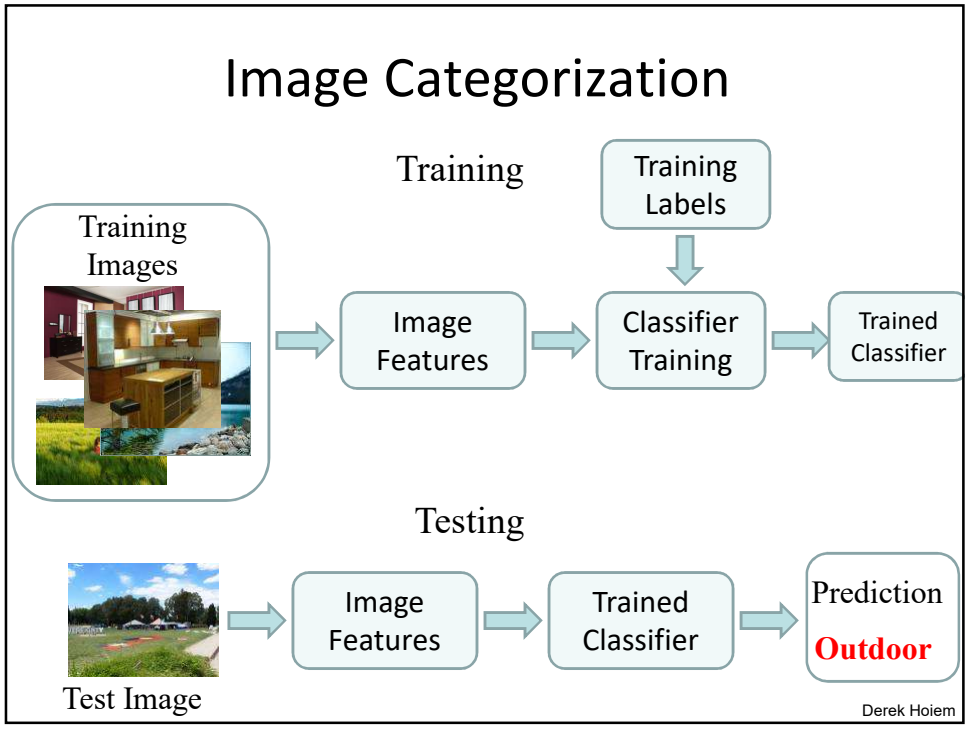


Image Categorization
Training
Training Labels
Training Images
Image Features
Classifier Training
Trained Classifier
Testing
Image Features
Trained Classifier
Prediction
Outdoor
Test Image
Derek Hoiem

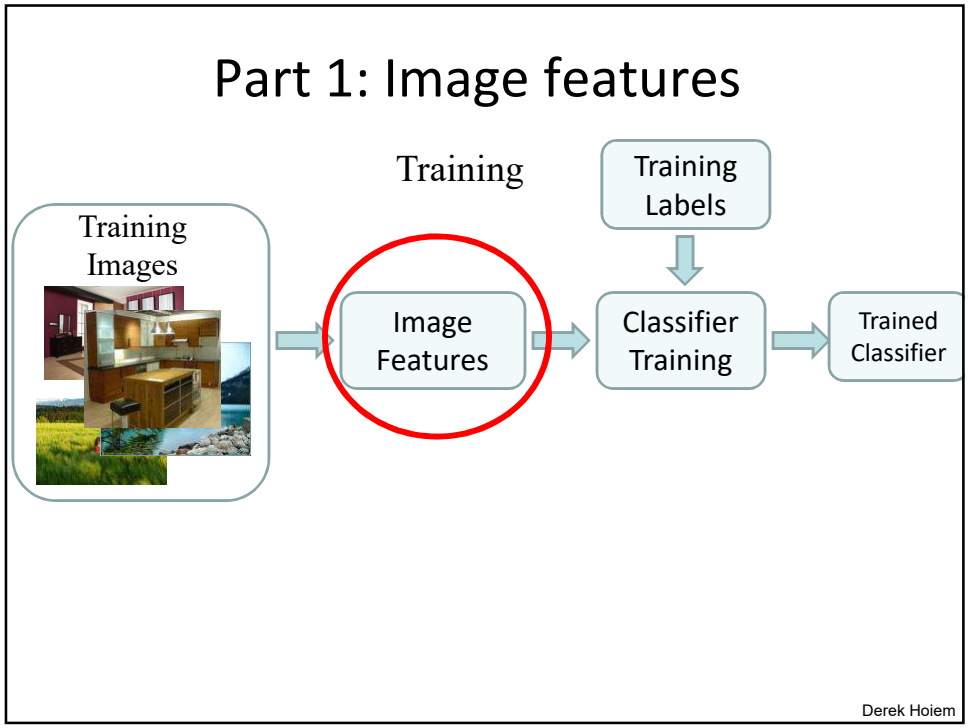
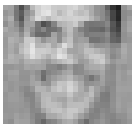

Part 1: Image features
Training
Training Labels
Training Images
Image Features
Classifier Training
Trained Classifier
Derek Hoiem

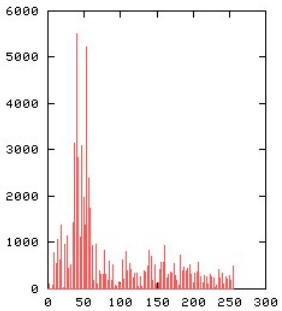
Image representations

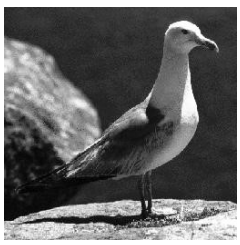
- Templates
  - Intensity, gradients, etc.



- Histograms
  - Color, texture, SIFT descriptors, etc.

Image Representations: Histograms





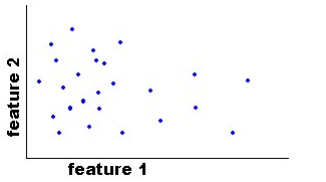
Global histogram

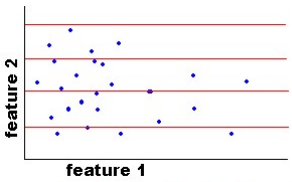
- Represent distribution of features
  - Color, texture, depth, …

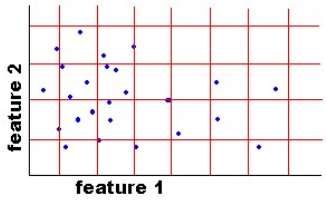
Images from Dave Kauchak

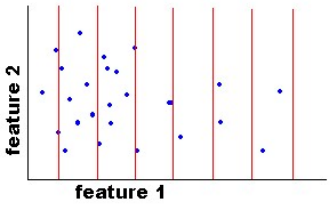
Image Representations: Histograms

Histogram: Probability or count of data in each bin









- Joint histogram
  - Requires lots of data
  - Loss of resolution to avoid empty bins

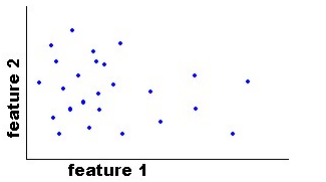
Marginal histogram

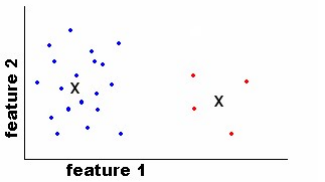
- Requires independent features
- More data/bin than joint histogram

Images from Dave Kauchak

Image Representations: Histograms

Clustering





Use the same cluster centers for all images

Images from Dave Kauchak

Computing histogram distance

$$\text{histint}(h_i, h_j) = 1 - \sum_{m=1}^{K} \min\left(h_i(m), h_j(m)\right)$$

Histogram intersection (assuming normalized histograms)

$$\chi^2(h_i, h_j) = \frac{1}{2} \sum_{m=1}^{K} \frac{[h_i(m) - h_j(m)]^2}{h_i(m) + h_j(m)}$$

Chi-squared Histogram matching distance

Cars found by color histogram matching using chi-squared

Histograms: Implementation issues

- Quantization
  - Grids: fast but applicable only with few dimensions
  - Clustering: slower but can quantize data in higher dimensions

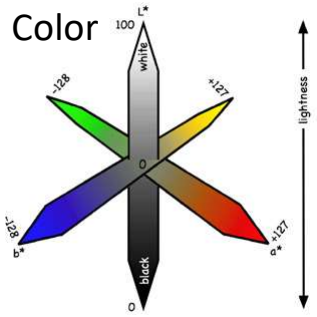
Few Bins
Need less data
Coarser representation

Many Bins
Need more data
Finer representation

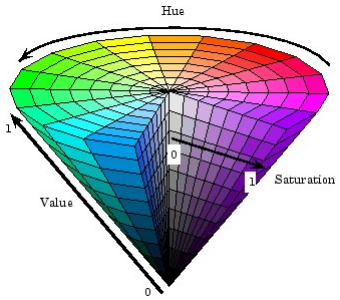
- Matching
  - Histogram intersection or Euclidean may be faster
  - Chi-squared often works better
  - Earth mover's distance is good for when nearby bins represent similar values

What kind of things do we compute histograms of?

- Color



L*a*b* color space

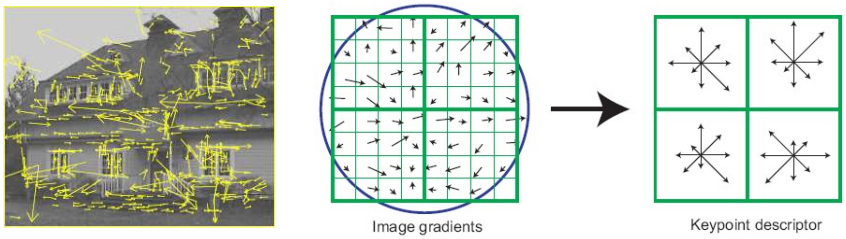


HSV color space

- Texture (filter banks or HOG over regions)

What kind of things do we compute histograms of?

- Histograms of oriented gradients



SIFT – Lowe IJCV 2004

SIFT vector formation

- Computed on rotated and scaled version of window according to computed orientation & scale
  - resample the window
- Based on gradients weighted by a Gaussian of variance half the window (for smooth falloff)

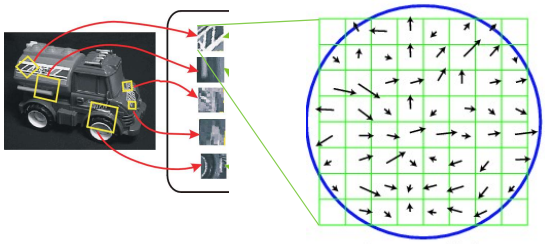


Image gradients

SIFT vector formation

- 4x4 array of gradient orientation histograms
  - not really histogram, weighted by magnitude
- 8 orientations x 4x4 array = 128 dimensions
- Motivation: some sensitivity to spatial layout, but not too much.

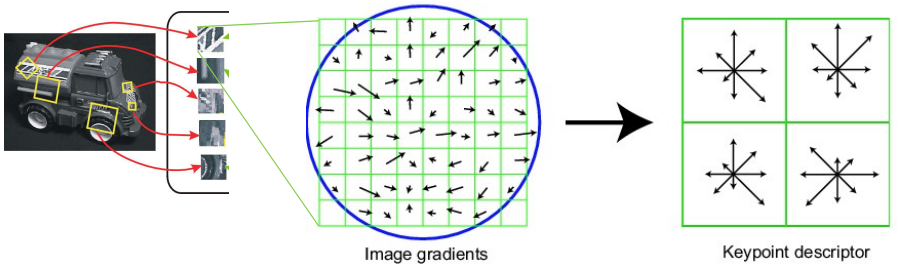


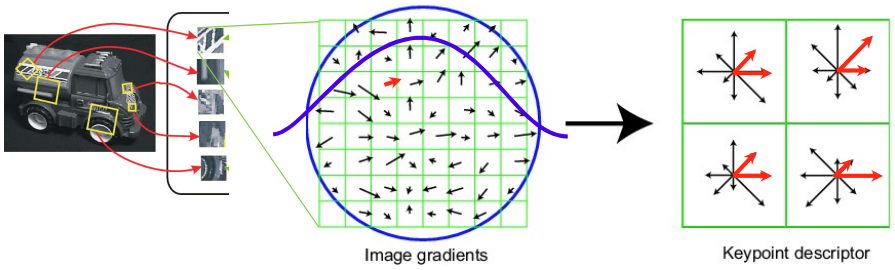
Image gradients

Keypoint descriptor

showing only 2x2 here but is 4x4

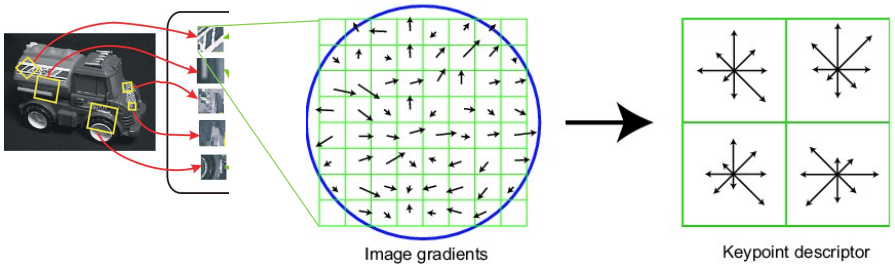
Ensure smoothness

- Gaussian weight
- Trilinear interpolation
  - a given gradient contributes to 8 bins: 4 in space times 2 in orientation



Reduce effect of illumination

- 128-dim vector normalized to 1
- Threshold gradient magnitudes to avoid excessive influence of high gradients
  - after normalization, clamp gradients >0.2
  - renormalize



Variants of SIFT

- HOG (Histogram of Gradients)
  - http://lear.inrialpes.fr/people/triggs/pubs/Dalal-cvpr05.pdf
  - http://www.vlfeat.org/overview/hog.html
- GIST
  - http://people.csail.mit.edu/torralba/code/spatialenvelope/
- SURF
  - http://www.vision.ee.ethz.ch/~surf/

Right features depend on what you want to know

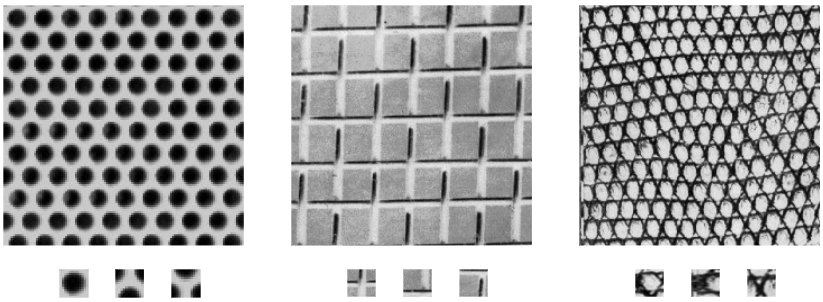
- Shape: scene-scale, object-scale, detail-scale
  - 2D form, shading, shadows, texture, linear perspective
- Material properties: albedo, feel, hardness, …
  - Color, texture
- Motion
  - Optical flow, tracked points

Bag-of-features models

Svetlana Lazebnik

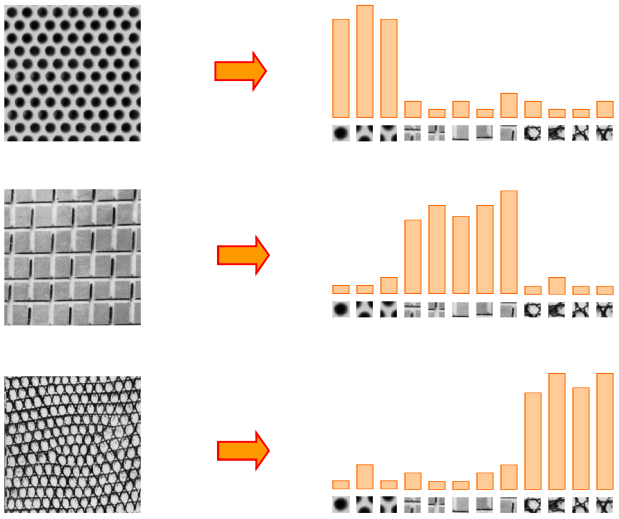
Origin 1: Texture recognition

- 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)

Origin 2: Bag-of-words models

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

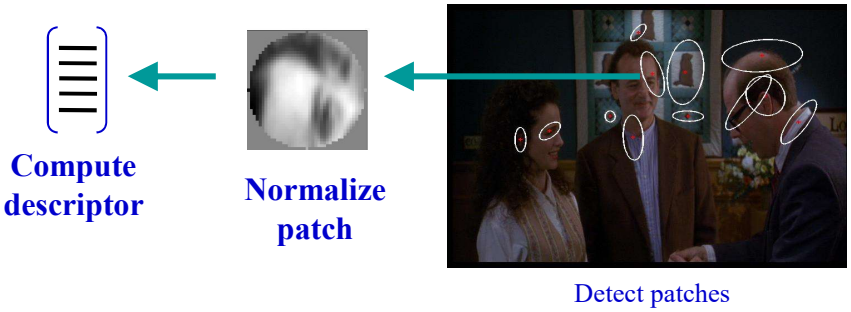
Origin 2: Bag-of-words models

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

1. Feature extraction

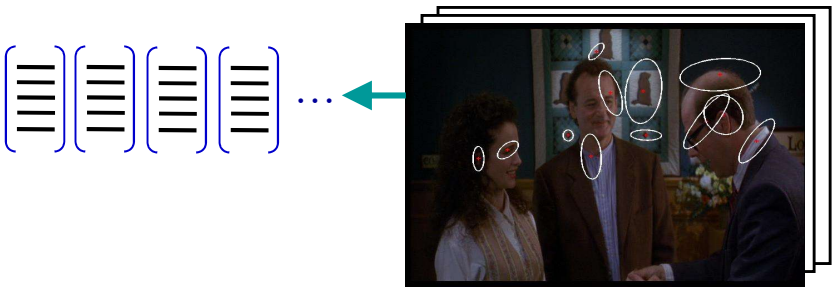
- Regular grid or interest regions

1. Feature extraction



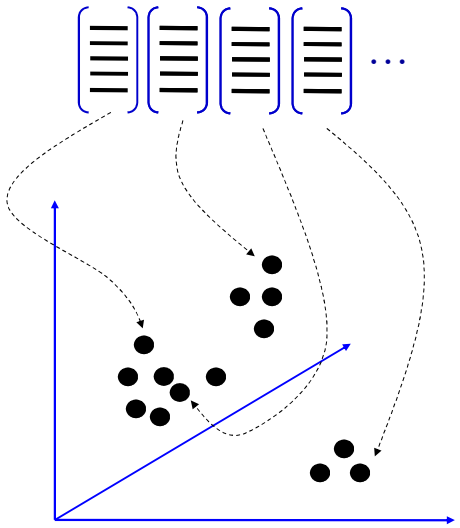
Slide credit: Josef Sivic

1. Feature extraction



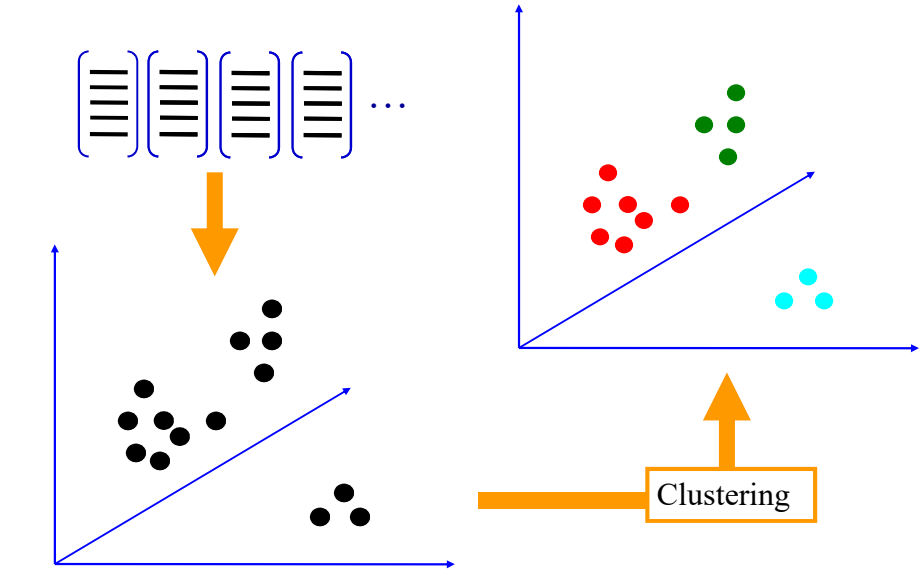
Slide credit: Josef Sivic

2. Learning the visual vocabulary



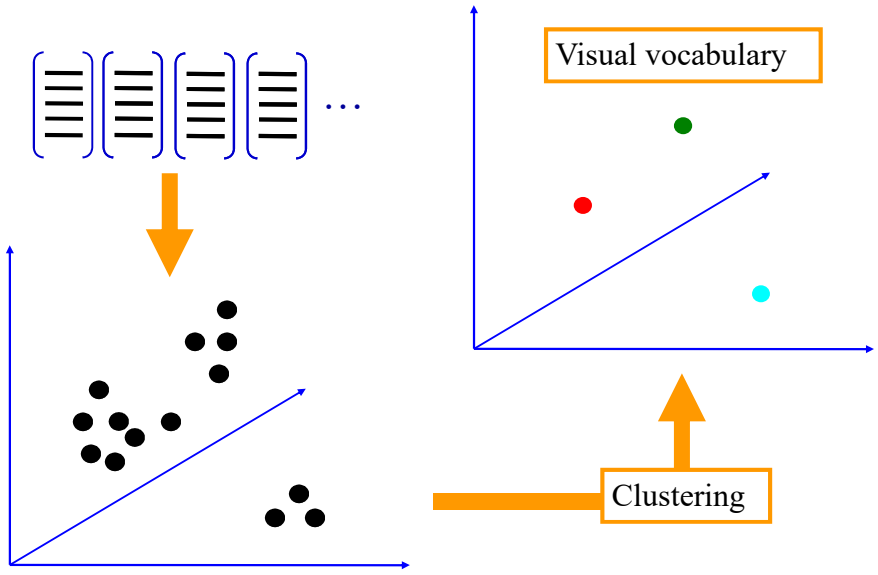
Slide credit: Josef Sivic

2. Learning the visual vocabulary



Slide credit: Josef Sivic

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

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

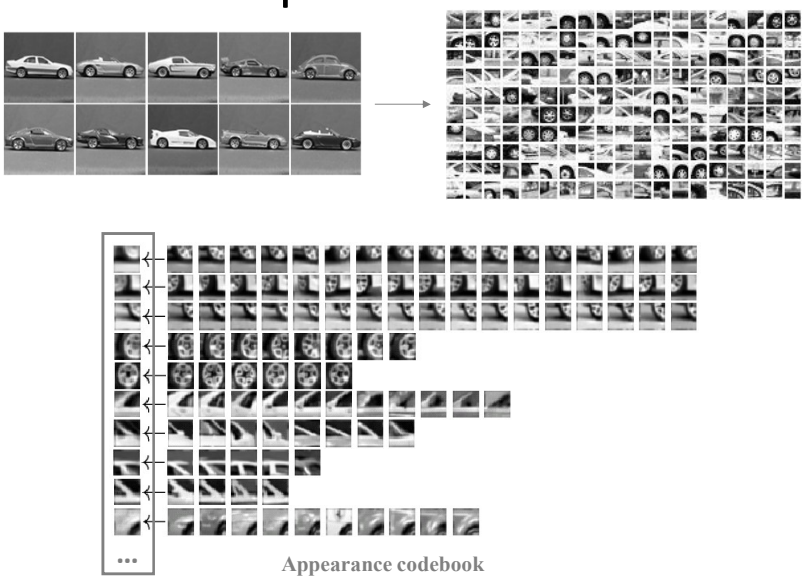
Algorithm:

- 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



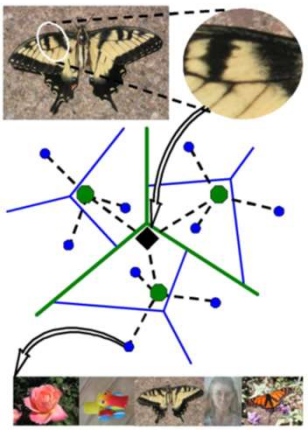
Source: B. Leibe

Another codebook

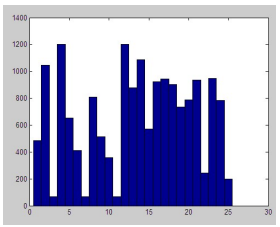
Source: B. Leibe

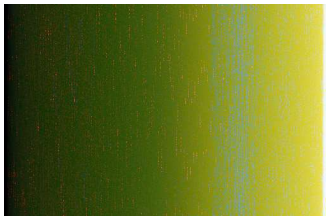
Visual vocabularies: Issues

- 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)



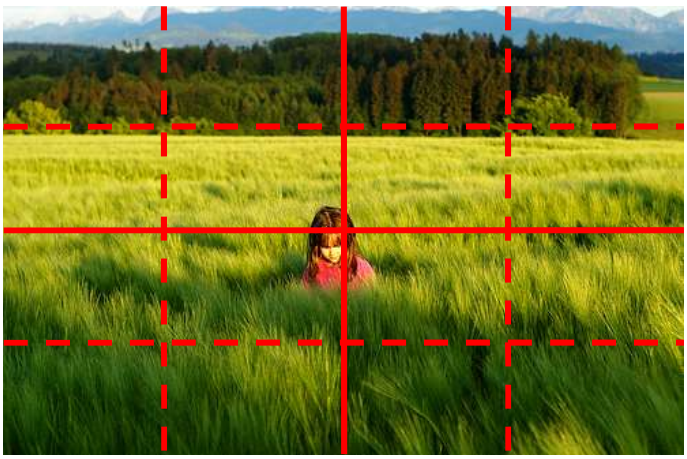
But what about layout?





All of these images have the same color histogram

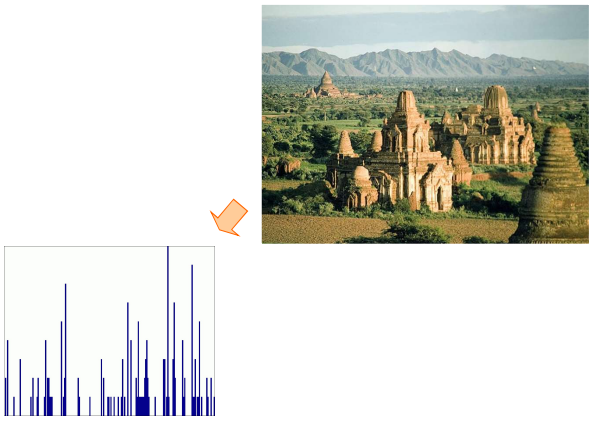
Spatial pyramid



Compute histogram in each spatial bin

Spatial pyramid representation

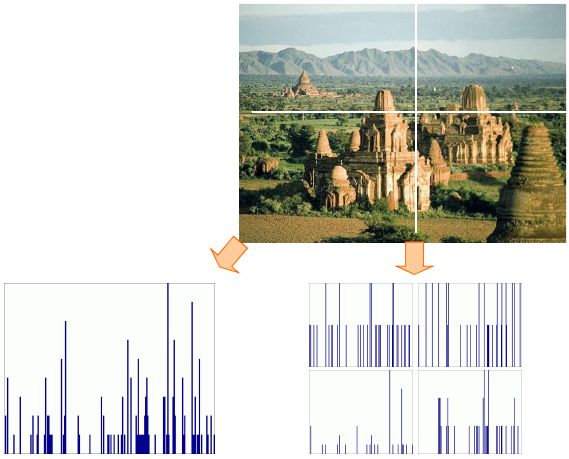
- Extension of a bag of features
- Locally orderless representation at several levels of resolution



Lazebnik, Schmid & Ponce (CVPR 2006)

Spatial pyramid representation

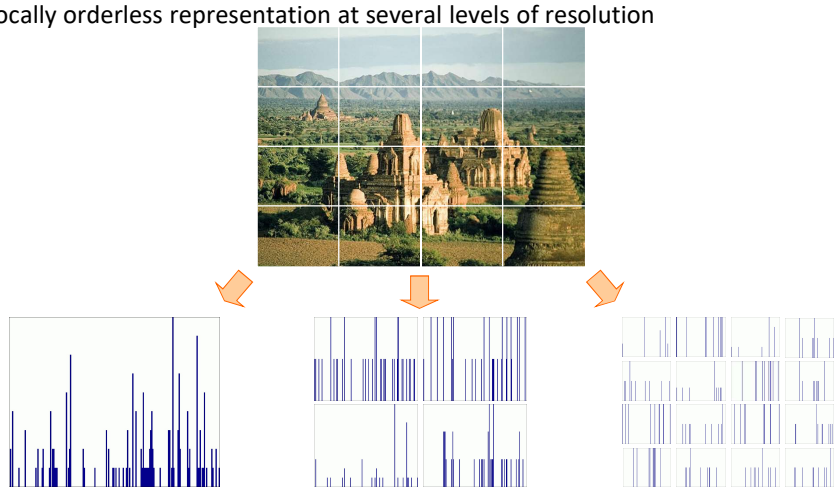
- Extension of a bag of features
- Locally orderless representation at several levels of resolution



Lazebnik, Schmid & Ponce (CVPR 2006)

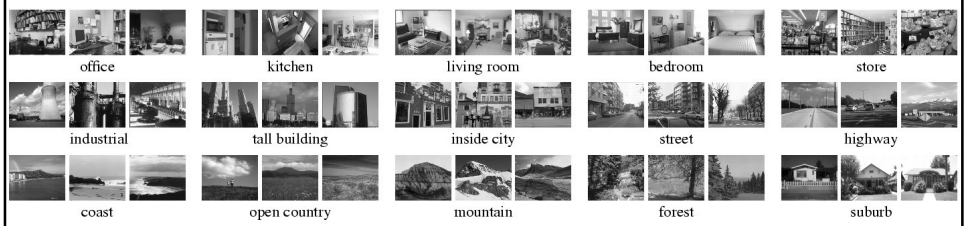
Spatial pyramid representation

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



Lazebnik, Schmid & Ponce (CVPR 2006)

Scene category dataset

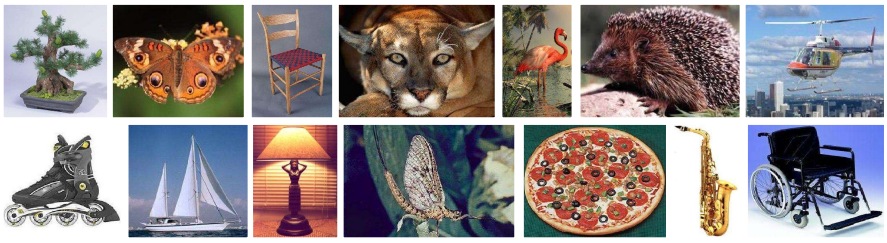


Multi-class classification results

	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

SUN database: http://groups.csail.mit.edu/vision/SUN/

Caltech101 dataset



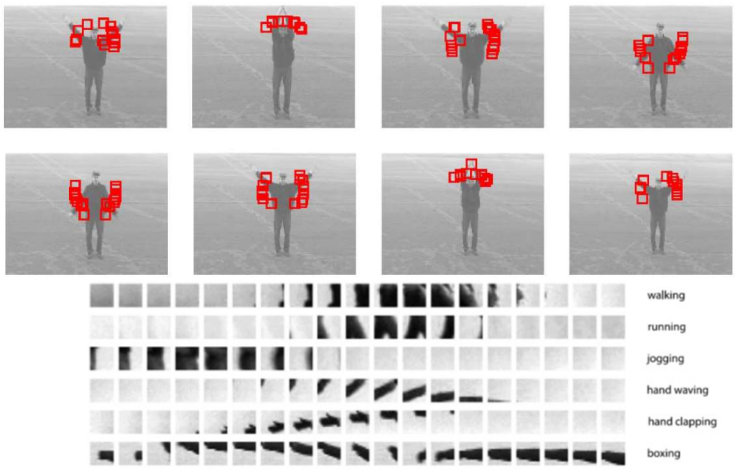
Multi-class classification results (30 training images per class)

	Weak features (16)		Strong features (200)	
Level	Single-level	Pyramid	Single-level	Pyramid
0	15.5 ±0.9		41.2 ±1.2	
1	31.4 ±1.2	32.8 ±1.3	55.9 ±0.9	57.0 ±0.8
2	47.2 ±1.1	49.3 ±1.4	63.6 ±0.9	**64.6** ±0.8
3	52.2 ±0.8	**54.0** ±1.1	60.3 ±0.9	64.6 ±0.7

http://www.vision.caltech.edu/Image_Datasets/Caltech101/Caltech101.html

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.

Bags of words: pros and cons

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

- - basic model ignores geometry – must verify afterwards, or encode via features
- - background and foreground mixed when bag covers whole image
- - interest points or sampling: no guarantee to capture object-level parts
- - optimal vocabulary formation remains unclear