Indexing with local features, Bag of words models

Thursday, Oct 30

Kristen Grauman
UT-Austin

Today

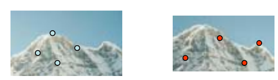
- Matching local features
- Indexing features
- Bag of words model

Main questions

- Where will the interest points come from?
  - What are salient features that we'll *detect* in multiple views?
- How to *describe* a local region?
- How to establish *correspondences*, i.e., compute matches?

Last time: Local invariant features

- Problem 1:
  - Detect the *same* point *independently* in both images

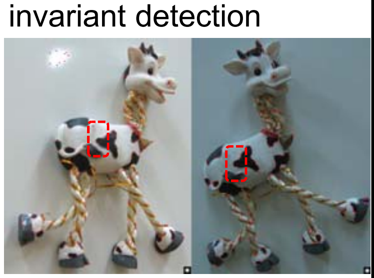


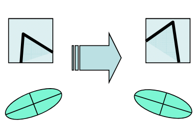
no chance to match!

We need a repeatable detector

Harris corner detector: rotation invariant detection

- Algorithm steps:
  - Compute M matrix within all image windows to get their R scores
  - Find points with large corner response $R >$ threshold)
  - Take the points of local maxima of R



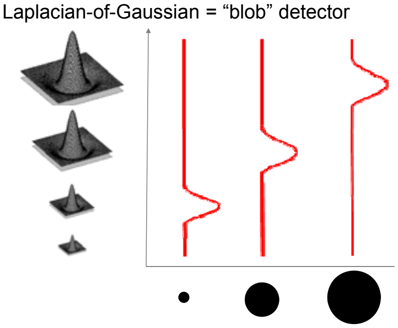


Corner response R is invariant to image rotation.

Ellipse rotates but its shape (i.e. eigenvalues) remains the same.

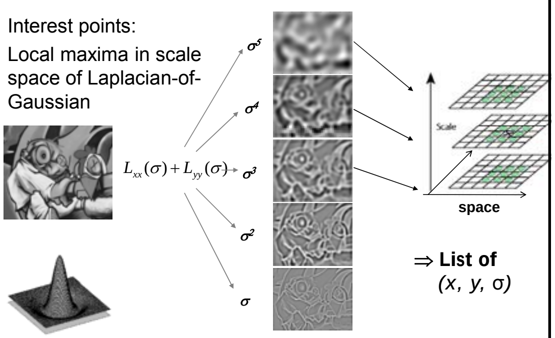
Laplacian of Gaussian: scale invariant detection

- Laplacian-of-Gaussian = "blob" detector



Laplacian of Gaussian: scale invariant detection

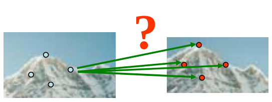
- Interest points:
  Local maxima in scale space of Laplacian-of-Gaussian



Laplacian of Gaussian: scale invariant detection

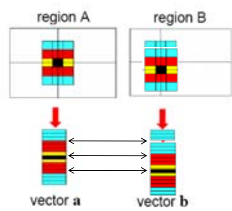
Last time: Local invariant features

- Problem 2:
  – For each point correctly recognize the corresponding one



We need a reliable and distinctive descriptor

Raw patches as local descriptors

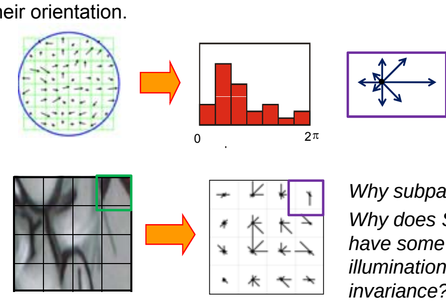


The simplest way to describe the neighborhood around an interest point is to write down the list of intensities to form a feature vector.

But this is very sensitive to even small shifts, rotations.

SIFT descriptors [Lowe 2004]

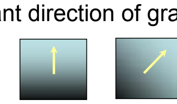
- More robust way to describe the neighborhood: use histograms to bin pixels within sub-patches according to their orientation.



Why subpatches?
Why does SIFT have some illumination invariance?

Rotation invariant descriptors

- Find local orientation
  Dominant direction of gradient for the image patch

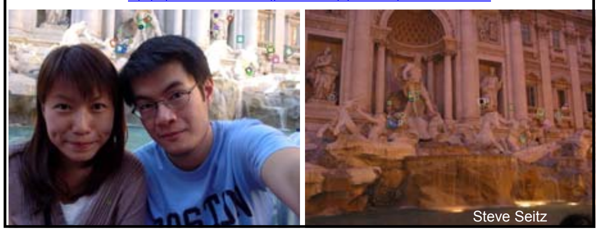


- Rotate patch according to this angle
  This puts the patches into a canonical orientation.

Feature descriptors: SIFT

Extraordinarily robust matching technique

- Can handle changes in viewpoint
  - Up to about 60 degree out of plane rotation
- Can handle significant changes in illumination
  - Sometimes even day vs. night (below)
- Fast and efficient—can run in real time
- Lots of code available
  - http://people.csail.mit.edu/albert/ladypack/wiki/index.php/Known_implementations_of_SIFT



Steve Seitz

Interest points + descriptors

- So far we have methods to find interest points and describe the surrounding image neighborhood.
- This will map each image to a list of local descriptors.

- *How many detections will an image have?*

Many Existing Detectors Available

• Hessian & Harris	[Beaudet '78], [Harris '88]
• Laplacian, DoG	[Lindeberg '98], [Lowe 1999]
• Harris-/Hessian-Laplace	[Mikolajczyk & Schmid '01]
• Harris-/Hessian-Affine	[Mikolajczyk & Schmid '04]
• EBR and IBR	[Tuytelaars & Van Gool '04]
• MSER	[Matas '02]
• Salient Regions	[Kadir & Brady '01]
• Others…	

Visual Object Recognition Tutorial

K. Grauman, B. Leibe

15

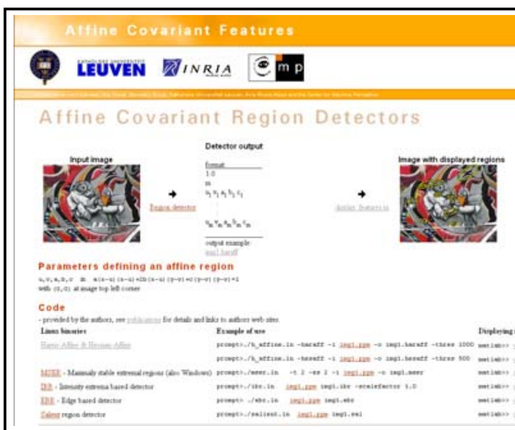
You Can Try It At Home…

- **For most local feature detectors, executables are available online:**
- **http://robots.ox.ac.uk/~vgg/research/affine**
- **http://www.cs.ubc.ca/~lowe/keypoints/**
- **http://www.vision.ee.ethz.ch/~surf**

Visual Object Recognition Tutorial

K. Grauman, B. Leibe

16



http://www.robots.ox.ac.uk/~vgg/research/affine/detectors.html#binaries

Main questions

- Where will the interest points come from?
  - What are salient features that we'll *detect* in multiple views?
- How to *describe* a local region?
- How to establish *correspondences*, i.e., compute matches?

Matching local features

Matching local features

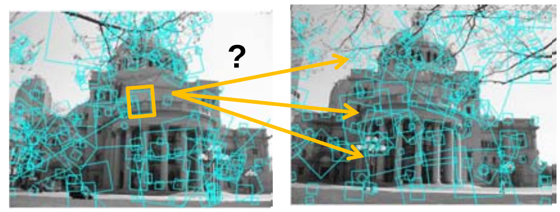


Image 1 Image 2

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

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

Matching local features

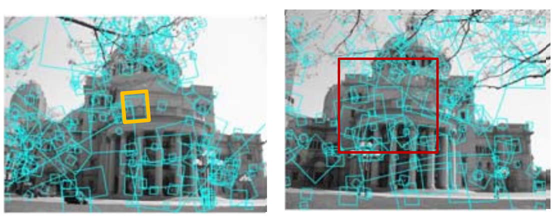


Image 1 Image 2

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

Ambiguous matches

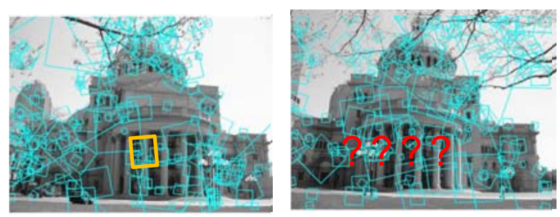


Image 1 Image 2

At what SSD value do we have a good match?

To add robustness to matching, can consider **ratio** : distance to best match / distance to second best match

If high, first match looks good.

Applications of local invariant features & matching

- Wide baseline stereo
- Motion tracking
- Panoramas
- Mobile robot navigation
- 3D reconstruction
- Recognition
  - Specific objects
  - Textures
  - Categories
- …

Wide baseline stereo

[Image from T. Tuytelaars ECCV 2006 tutorial]

Panorama stitching

Brown, Szeliski, and Winder, 2005

Automatic mosaicing

http://www.cs.ubc.ca/~mbrown/autostitch/autostitch.html

Recognition of specific objects, scenes

Schmid and Mohr 1997

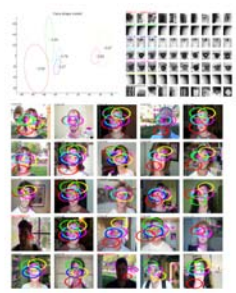
Sivic and Zisserman, 2003

Rothganger et al. 2003

Lowe 2002

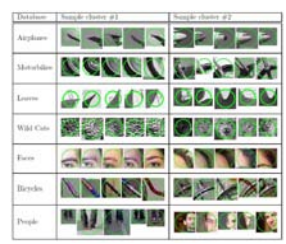
Recognition of categories

Constellation model



Weber et al. (2000)
Fergus et al. (2003)

Bags of words



Csurka et al. (2004)
Dorko & Schmid (2005)
Sivic et al. (2005)
Lazebnik et al. (2006), …

[Slide from Lana Lazebnik, Sicily 2006]

Value of local features

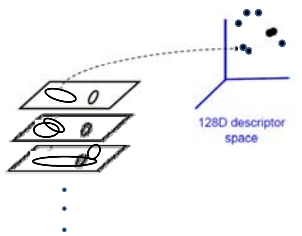
- Critical to find distinctive and repeatable local regions for multi-view matching
- Complexity reduction via selection of distinctive points
- Describe images, objects, parts without requiring segmentation; robustness to clutter & occlusion
- Robustness: similar descriptors in spite of moderate view changes, noise, blur, etc.

Today

- Matching local features
- **Indexing features**
- Bag of words model

Indexing local features

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

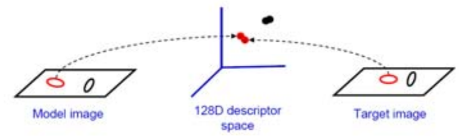


Visual Object Recognition Tutorial

K. Grauman, B. Leibe

Indexing local features

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



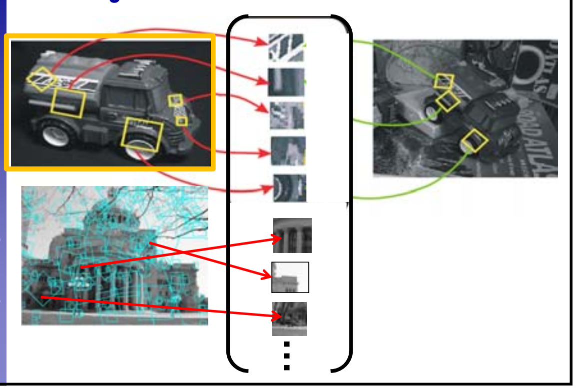
- This is of interest not only for 3d reconstruction, but also for retrieving images of similar objects.

Figure credit: A. Zisserman

Visual Object Recognition Tutorial

K. Grauman, B. Leibe

Indexing local features



Visual Object Recognition Tutorial

Indexing local features

- With potentially thousands of features per image, and hundreds to millions of images to search, how to efficiently find those that are relevant to a new image?
  - Low-dimensional descriptors : can use standard efficient data structures for nearest neighbor search
  - High-dimensional descriptors: approximate nearest neighbor search methods more practical
  - Inverted file indexing schemes

Visual Object Recognition Tutorial

K. Grauman, B. Leibe

Indexing local features: inverted file index

- For text documents, an efficient way to find all *pages* on which a *word* occurs is to use an index…
- We want to find all *images* in which a *feature* occurs.
- To use this idea, we'll need to map our features to "visual words".

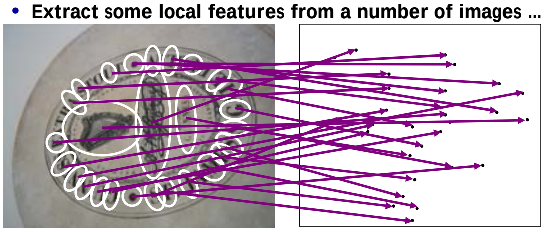
Visual Object Recognition Tutorial

Text retrieval vs. image search

- What makes the problems similar, different?

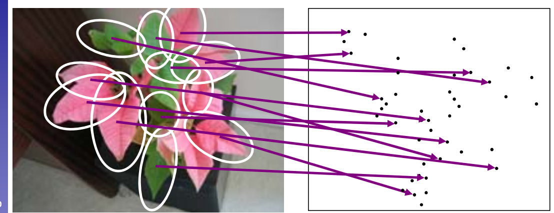
Visual words: main idea

- Extract some local features from a number of images …

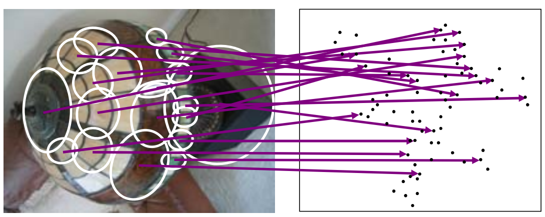


Slide credit: D. Nister

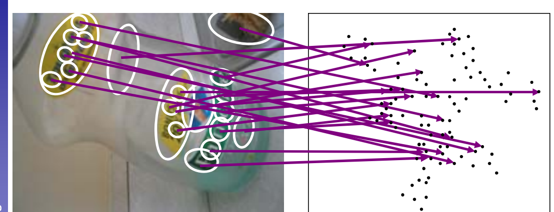
Visual words: main idea

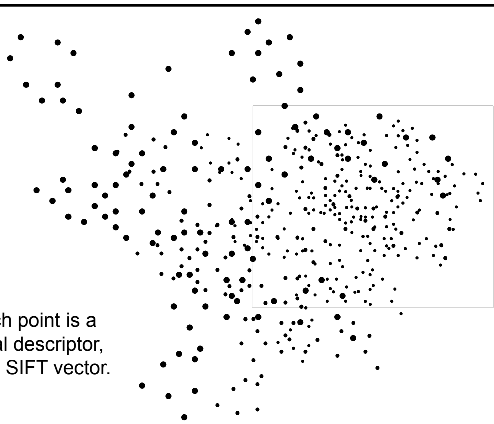


Visual words: main idea

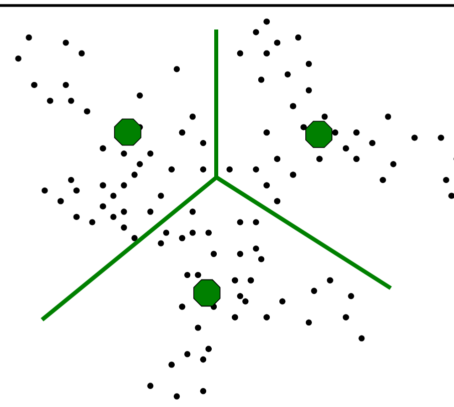


Visual words: main idea





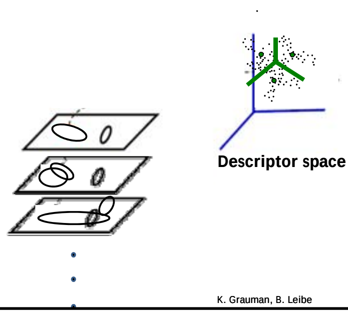
Each point is a local descriptor, e.g. SIFT vector.



Visual words: main idea

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

- Quantize via clustering, let cluster centers be the prototype "words"

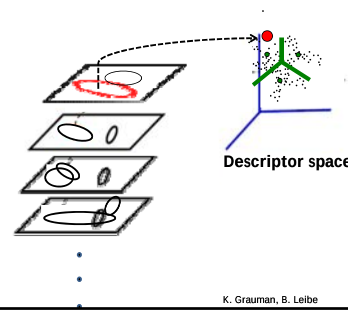


Visual Object Recognition Tutorial

Visual words: main idea

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

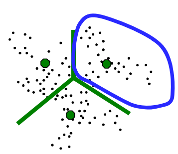
- Determine which word to assign to each new image region by finding the closest cluster center.



Visual Object Recognition Tutorial

Visual words

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



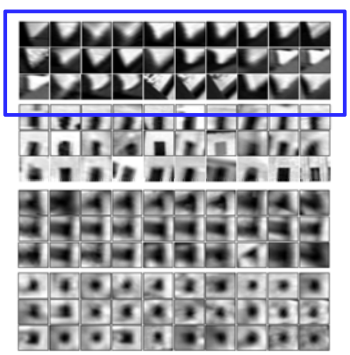


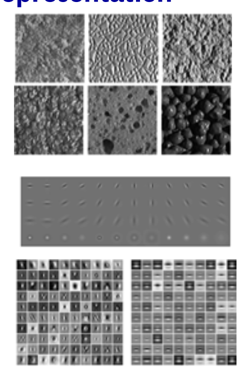
Figure from Sivic & Zisserman, ICCV 2003

Visual Object Recognition Tutorial

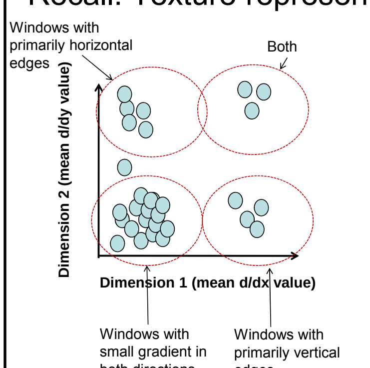
Visual words: texture representation

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

Leung & Malik 1999; Varma & Zisserman, 2002; Lazebnik, Schmid & Ponce, 2003;



Recall: Texture representation example

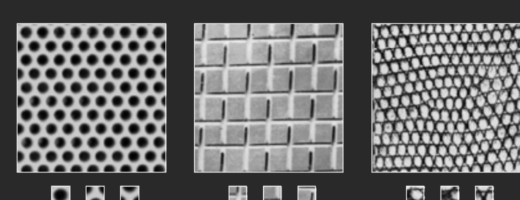


	mean d/dx value	mean d/dy value
Win. #1	4	10
Win.#2	18	7
⋮		
Win.#9	20	20
⋮		

statistics to summarize patterns in small windows

Visual words: texture representation

- 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



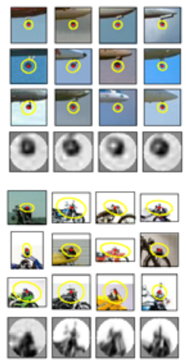
Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Source: Lana Lazebnik

Visual words

Visual Object Recognition Tutorial

- More recently used for describing scenes and objects for the sake of indexing or classification.



Sivic & Zisserman 2003; Csurka, Bray, Dance, & Fan 2004; many others.

K. Grauman, B. Leibe

49

Inverted file index for images comprised of visual words

Visual Object Recognition Tutorial

When will this give us a significant gain in efficiency?

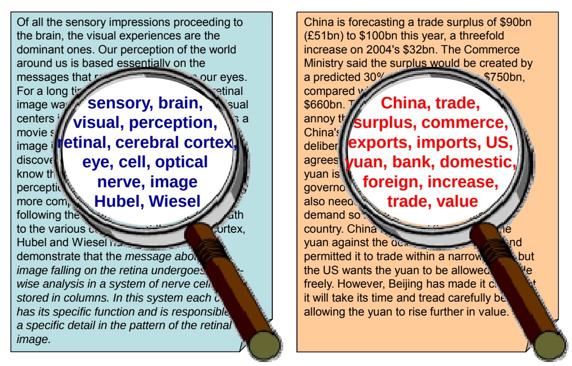
Image credit: A. Zisserman

K. Grauman, B. Leibe

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

K. Grauman, B. Leibe

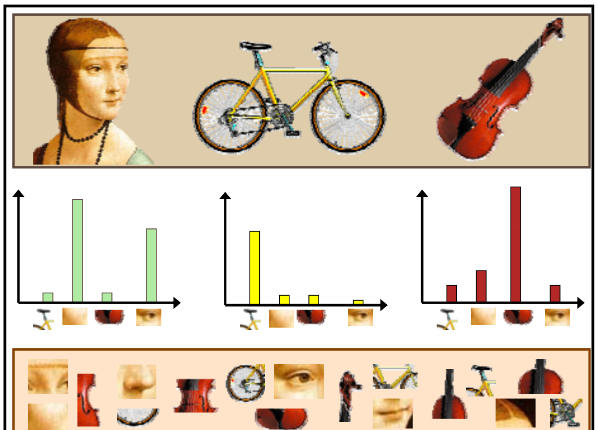
Analogy to documents



ICCV 2005 short course, L. Fei-Fei

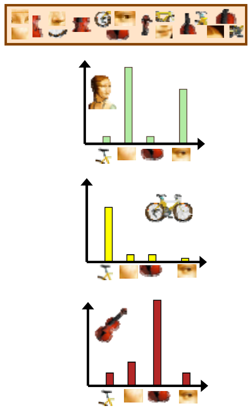
Object → **Bag of 'words'**

ICCV 2005 short course, L. Fei-Fei



Bags of visual words

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



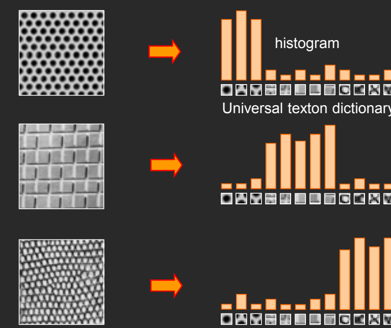
Visual Object Recognition Tutorial

Image credit: Fei-Fei Li

K. Grauman, B. Leibe

55

Similarly, bags of textons for texture representation



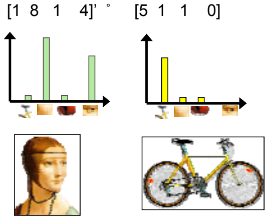
Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Source: Lana Lazebnik

Comparing bags of words

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

[1 8 1 4]' ° [5 1 1 0]



$$sim(d_j, q) = \frac{\vec{d_j} \bullet \vec{q}}{|\vec{d_j}| \times |\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{j=1}^{t} w_{i,q}^2}}$$

$\vec{d}_j$ $\vec{q}$

tf-idf weighting

- **T**erm **f**requency – **i**nverse **d**ocument **f**requency
- Describe frame by frequency of each word within it, downweight words that appear often in the database
- (Standard weighting for text retrieval)

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

Number of occurrences of word i in document d

Number of words in document d

Total number of documents in database

Number of occurrences of word i in whole database

Bags of words for content-based image retrieval

What if query of interest is a portion of a frame?

Visually defined query

"Groundhog Day" [Rammis, 1993]

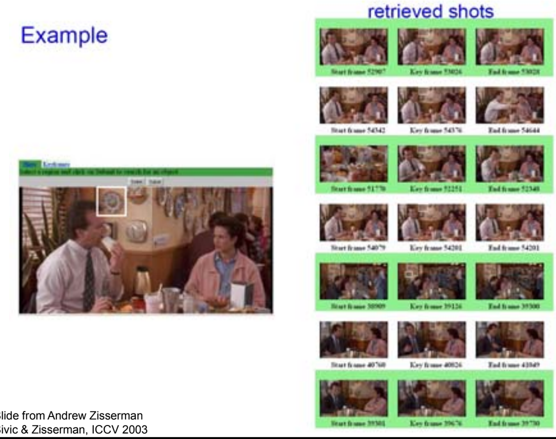
"Find this clock"

"Find this place"

Slide from Andrew Zisserman
Sivic & Zisserman, ICCV 2003

Example

retrieved shots



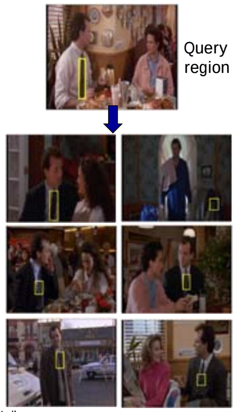
Slide from Andrew Zisserman
Sivic & Zisserman, ICCV 2003

Video Google System

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

Sivic & Zisserman, ICCV 2003

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

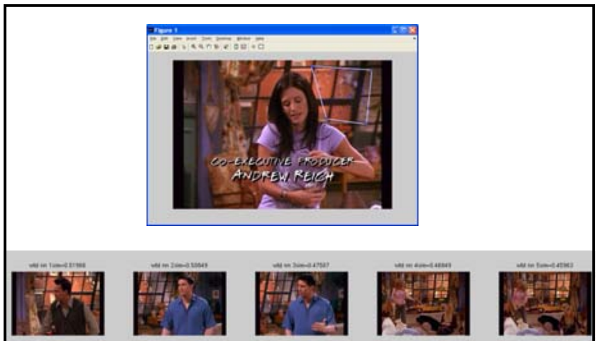


Query region

Retrieved frames

- Collecting words within a query region

Query region: pull out only the SIFT descriptors whose positions are within the polygon



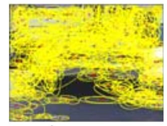
Bag of words representation: spatial info

- A bag of words is an orderless representation: throwing out spatial relationships between features
- Middle ground:
  - Visual “phrases” : frequently co-occurring words
  - Semi-local features : describe configuration, neighborhood
  - Let position be part of each feature
  - Count bags of words only within sub-grids of an image
  - After matching, verify spatial consistency (e.g., look at neighbors – are they the same too?)

Visual vocabulary formation

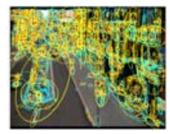
Issues:
- **Sampling strategy: where to extract features?**
- **Clustering / quantization algorithm**
- **Unsupervised vs. supervised**
- **What corpus provides features (universal vocabulary?)**
- **Vocabulary size, number of words**

Sampling strategies


Sparse, at interest points


Dense, uniformly


Randomly


Multiple interest operators

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

[See Nowak, Jurie & Triggs, ECCV 2006]

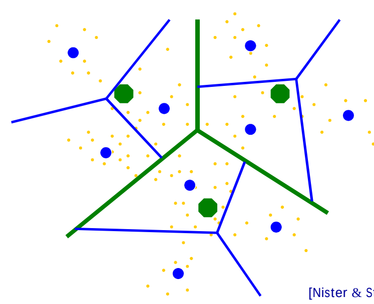
Image credits: F-F. Li, E. Nowak, J. Sivic

Clustering / quantization methods

- **k-means (typical choice), agglomerative clustering, mean-shift,…**
- **Hierarchical clustering: allows faster insertion / word assignment while still allowing large vocabularies**
  - **Vocabulary tree [Nister & Stewenius, CVPR 2006]**

Example: Recognition with Vocabulary Tree

- **Tree construction:**

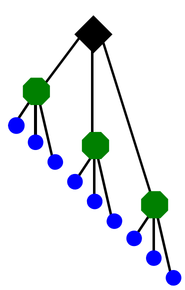


[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

Vocabulary Tree

- **Training: Filling the tree**

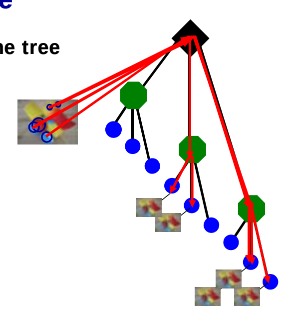


[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

Vocabulary Tree

- **Training: Filling the tree**

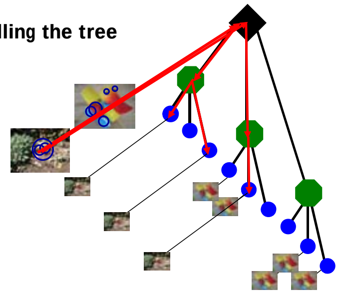


[Nister & Stewenius, CVPR'06]

Slide credit: David Nister

Vocabulary Tree

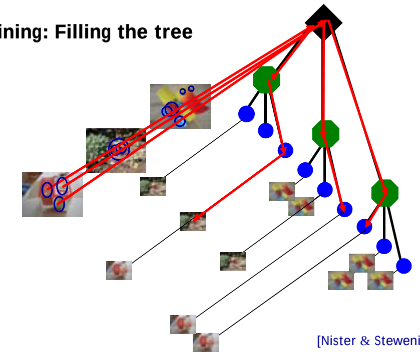
- **Training: Filling the tree**



[Nister & Stewenius, CVPR'06]

Vocabulary Tree

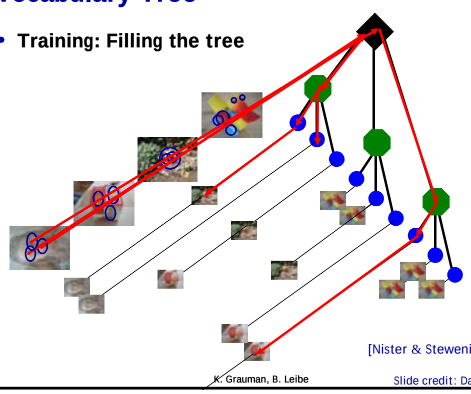
- **Training: Filling the tree**



[Nister & Stewenius, CVPR'06]

Vocabulary Tree

- **Training: Filling the tree**

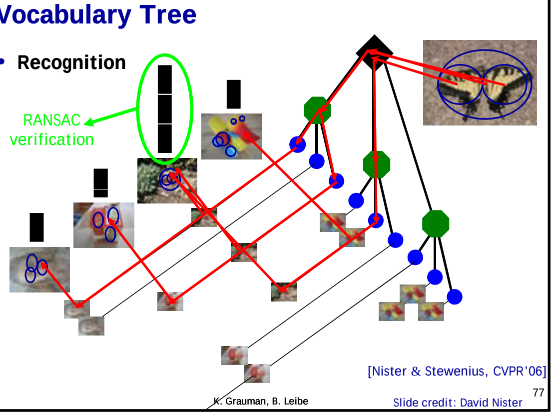


[Nister & Stewenius, CVPR'06]

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

Vocabulary Tree

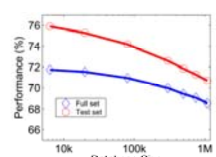
- **Recognition**



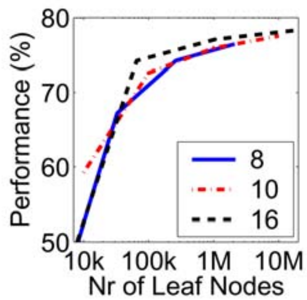
[Nister & Stewenius, CVPR'06]

Vocabulary Tree: Performance

- **Evaluated on large databases**
  - **Indexing with up to 1M images**
- **Online recognition for database of 50,000 CD covers**
  - **Retrieval in ~1s**
- **Find experimentally that large vocabularies can be beneficial for recognition**



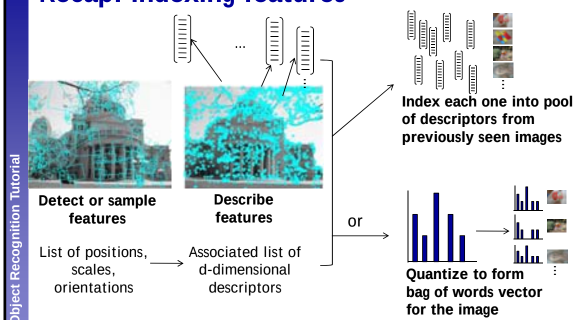
[Nister & Stewenius, CVPR'06]



Larger vocabularies can be advantageous…

But what happens if it is too large?

Recap: indexing features



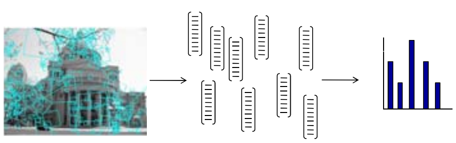
Visual Object Recognition Tutorial

K. Grauman, B. Leibe

80

Learning and recognition with bag of words histograms

- Bag of words representation makes it possible to describe the unordered point set with a single vector (of fixed dimension across image examples)



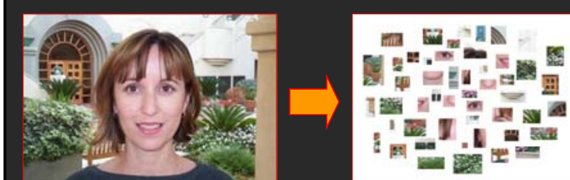
- Provides easy way to use distribution of feature types with various learning algorithms requiring vector input.

Visual Object Recognition Tutorial

K. Grauman, B. Leibe

81

Bags of features for object recognition



face, flowers, building

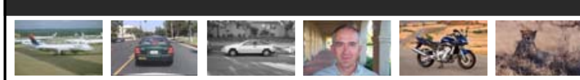
- Works pretty well for image-level classification

Csurka et al. (2004), Willamowski et al. (2005), Grauman & Darrell (2005), Sivic et al. (2003, 2005)

Source: Lana Lazebnik

Bags of features for object recognition

Caltech6 dataset



class	bag of features Zhang et al. (2005)	bag of features Willamowski et al. (2004)	Parts-and-shape model Fergus et al. (2003)
airplanes	**98.8**	97.1	90.2
cars (rear)	98.3	**98.6**	90.3
cars (side)	**95.0**	87.3	88.5
faces	**100**	99.3	96.4
motorbikes	**98.5**	98.0	92.5
spotted cats	**97.0**	—	90.0

Source: Lana Lazebnik

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

Visual Object Recognition Tutorial

K. Grauman, B. Leibe

84

Summary

- Local invariant features: distinctive matches possible in spite of significant view change, useful not only to provide matches for multi-view geometry, but also to find objects and scenes.
- To find correspondences among detected features, measure distance between descriptors, and look for most similar patches.
- Bag of words representation: quantize feature space to make discrete set of visual words
  - Summarize image by distribution of words
  - Index individual words
- Inverted index: pre-compute index to enable faster search at query time

Next

- Next week : Object recognition