Lecture 14:
Indexing with local features

Thursday, Nov 1

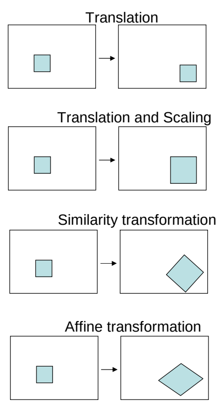
Prof. Kristen Grauman

Outline

- Last time: local invariant features, scale invariant detection
- Applications, including stereo
- Indexing with invariant features
- Bag-of-words representation for images

Classes of transformations

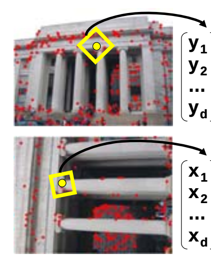
- **Euclidean/rigid**: Translation + rotation
  – Lengths and angles preserved
- **Similarity**: Translation + rotation + uniform scale
- **Affine**: Similarity + shear
  – Valid for orthographic camera, locally planar object
  – Lengths and angles **not** preserved



Invariant local features

Subset of local feature types designed to be *invariant* to

– Scale
– Translation
– Rotation
– Affine transformations
– Illumination

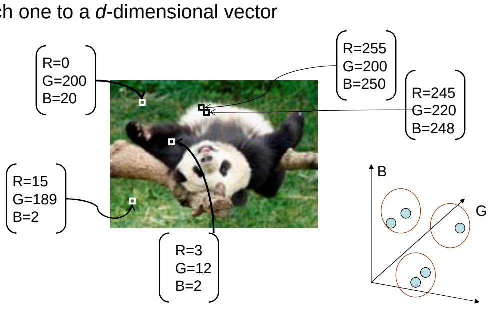


1) Detect distinctive interest points
2) Extract invariant descriptors

[Mikolajczyk & Schmid, Matas et al., Tuytelaars & Van Gool, Lowe, Kadir et al.,…]

Recall: segmentation as clustering

- Previously we represented *pixels* with features, mapping each one to a *d*-dimensional vector



Recall: segmentation as clustering

- Previously we represented *pixels* with features, mapping each one to a *d*-dimensional vector

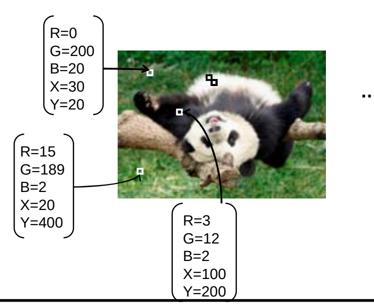
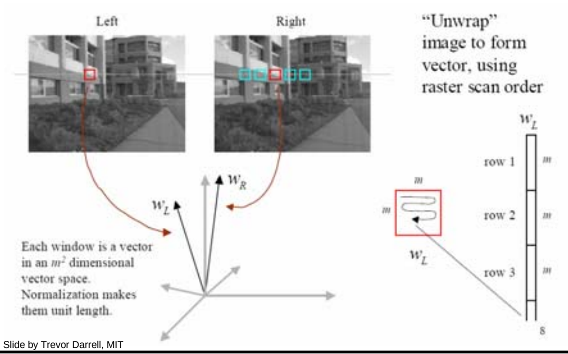
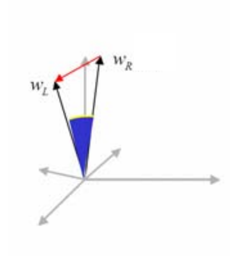


Image patches as vectors



Slide by Trevor Darrell, MIT

Image metrics

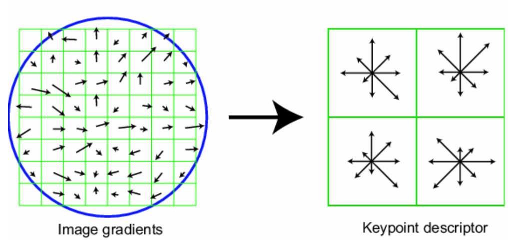


Can compare those vector descriptions

- SSD
- Dot product
- …

SIFT descriptors: vector formation

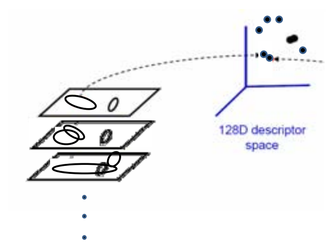
- Thresholded image gradients are sampled over 16x16 array of locations in scale space
- Create **array of orientation histograms**
- 8 orientations x 4x4 histogram array = 128 dimensions



David Lowe, UBC

Indexing with local features

- Now we have patches or regions, still mapping each one to a *d*-dimensional vector (e.g., *d*=128 for SIFT)



Indexing with local features

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

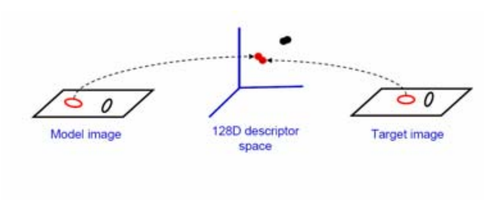


Figure from Andrew Zisserman, University of Oxford

What are the limitations of describing image patches with a stack of pixel intensities?

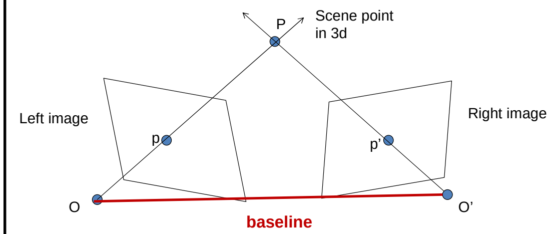
Why should something like a SIFT descriptor be more robust?

What role does the interest point detection play?

Many applications of local features

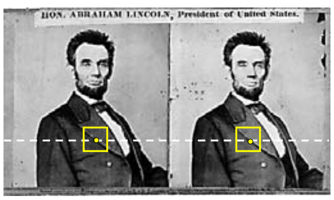
- Wide baseline stereo
- Motion tracking
- Panoramas
- Mobile robot navigation
- 3D reconstruction
- Recognition
  - Specific objects
  - Textures
  - Categories
- …

Recall: Triangulation



Estimate scene point based on camera relationships and correspondence.

Dense correspondence search



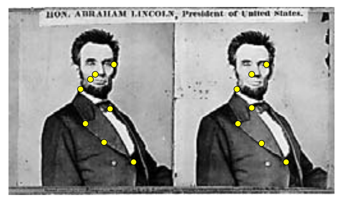
For each epipolar line

For each pixel / window in the left image

- compare with every pixel / window on same epipolar line in right image
- pick position with minimum match cost (e.g., SSD, correlation)

Adapted from Li Zhang

Sparse correspondence search



- Restrict search to sparse set of detected features
- Rather than pixel values (or lists of pixel values) use *feature descriptor* and an associated *feature distance*
- Still narrow search further by epipolar geometry

Wide baseline stereo

- 3d reconstruction depends on finding good correspondences
- Especially with wide-baseline views, local image deformations not well-approximated with rigid transformations
- Cannot simply compare regions of fixed shape (circles, rectangles) – shape is not preserved under affine transformations

Wide baseline stereo

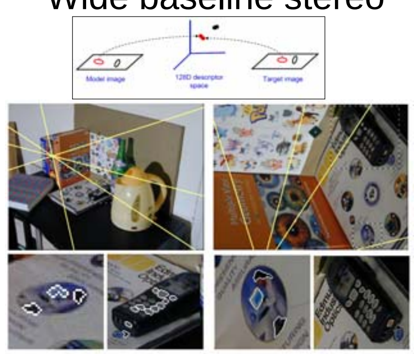


Figure 1: BOOKSHELF: Estimated epipolar geometry on indoor scene with significant scale change. In the cutouts the change in the resolution of detected DRs is clearly visible.

J. Matas, O. Chum, M. Urban, T. Pajdla. Robust Wide Baseline Stereo From Maximally Stable Extremal Regions, BMVC 2002.

Wide baseline stereo

Figure 2: VALBONNE: Estimated epipolar geometry and points associated to the matched regions are shown in the first row. Cutouts in the second row show matched bricks.

J. Matas, O. Chum, M. Urban, T. Pajdla. Robust Wide Baseline Stereo From Maximally Stable Extremal Regions, BMVC 2002.

Wide baseline stereo

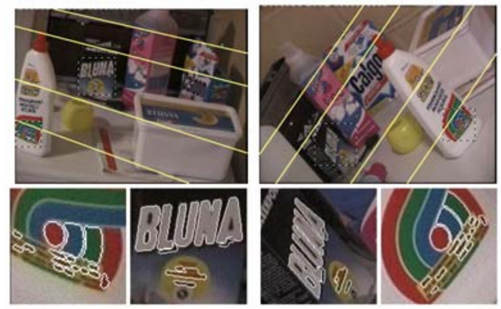
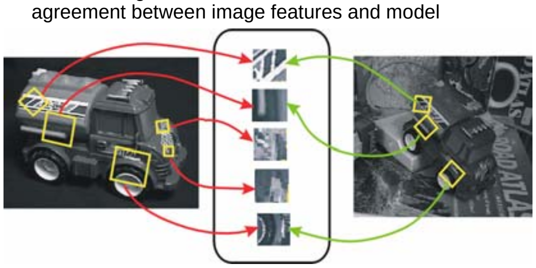


Figure 3: WASH: Epipolar geometry and dense matched regions with fully affine distortion.

J. Matas, O. Chum, M. Urban, T. Pajdla. Robust Wide Baseline Stereo From Maximally Stable Extremal Regions, BMVC 2002.

SIFT matching and recognition

- Index descriptors
- Generalized Hough transform: vote for object poses
- Refine with geometric verification: affine fit, check for agreement between image features and model



[Lowe 1999] **SIFT Features**

Recognition of specific objects, scenes

Schmid and Mohr 1997

Sivic and Zisserman, 2003

Rothganger et al. 2003

Lowe 2002

Panorama stitching

Brown, Szeliski, and Winder, 2005

Value of local (invariant) features

- Complexity reduction via selection of distinctive points
- Describe images, objects, parts without requiring segmentation
- Local character means robustness to clutter, occlusion
- Robustness: similar descriptors in spite of noise, blur, etc.

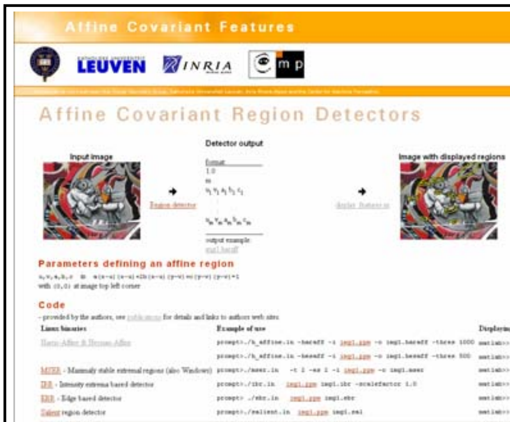
Comparative evaluations

Testing various detector and descriptor options for relative *repeatability* and *distinctiveness*

Planar objects / flat scenes:
Mikolajczyk & Schmid (2004)

3D objects:
Moreels & Perona (2005)

[Images from Lazebnik, Sicily 2006]



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

Outline

- Last time: local invariant features, scale invariant detection
- Applications, including stereo
- Indexing with invariant features
- Bag-of-words representation for images

Success of text retrieval

- efficient
- scalable
- high precision

Can we use retrieval mechanisms from text retrieval?

Need a visual analogy of a textual word.

Slide from Andrew Zisserman, University of Oxford

Visual problem

- Retrieve key frames containing the same object

query

?

Slide from Andrew Zisserman

Problem specification: particular object retrieval

Example: visual search in feature films

Visually defined query

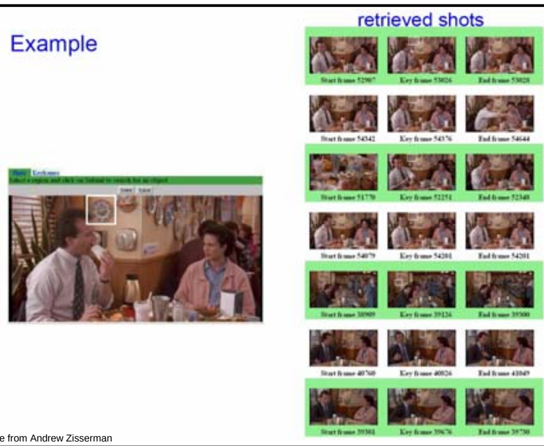
"Groundhog Day" [Rammis, 1993

"Find this clock"

"Find this place"

Slide from Andrew Zisserman

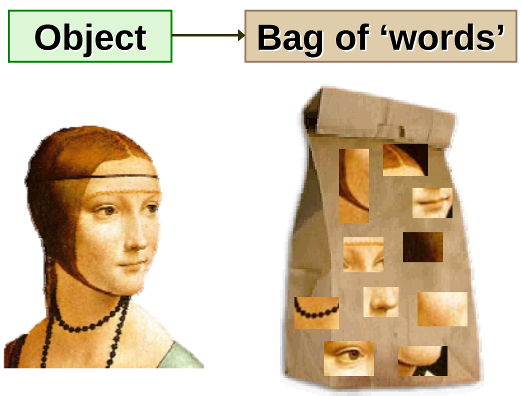
Example



Slide from Andrew Zisserman

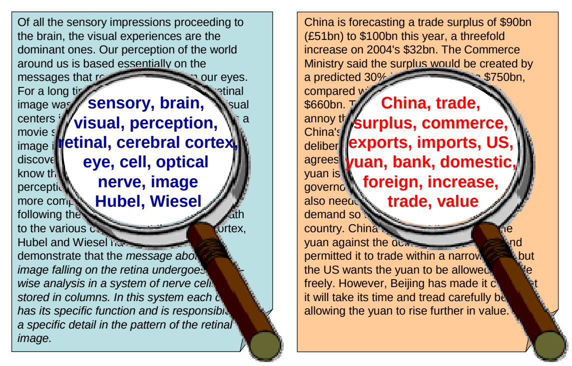
Text retrieval vs. image search

- What makes the problems similar, different?

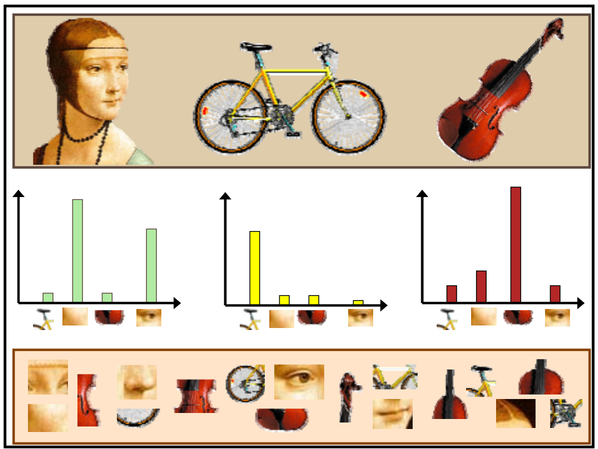


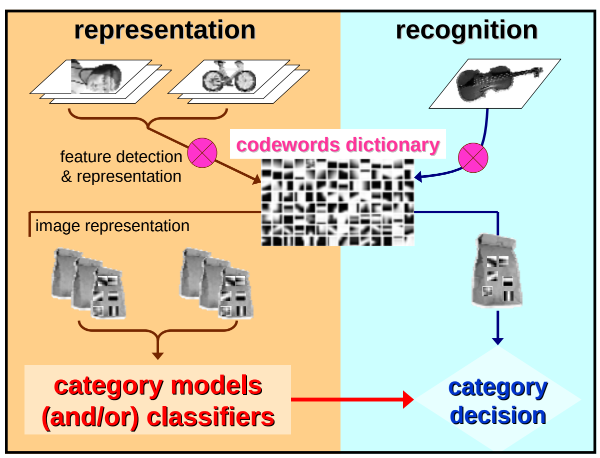
ICCV 2005 short course, L. Fei-Fei

Analogy to documents



ICCV 2005 short course, L. Fei-Fei



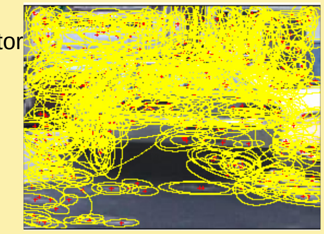


1.Feature detection and representation

- Regular grid

1.Feature detection and representation

- Regular grid
- Interest point detector



1.Feature detection and representation

- Regular grid
- Interest point detector
- Other methods
  – Random sampling
  – Segmentation based patches

1.Feature detection and representation

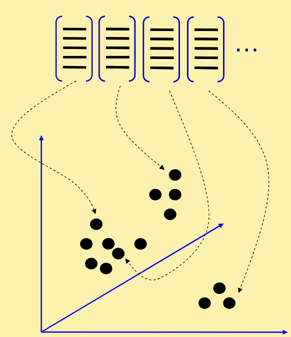
Detect patches

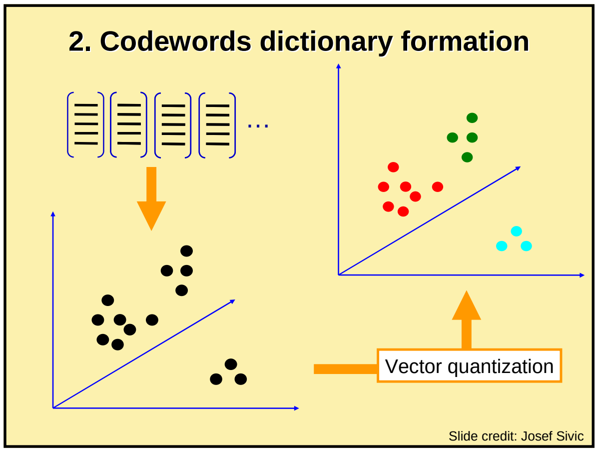
[Mikojaczyk and Schmid '02]
[Matas et al. '02]
[Sivic et al. '03]

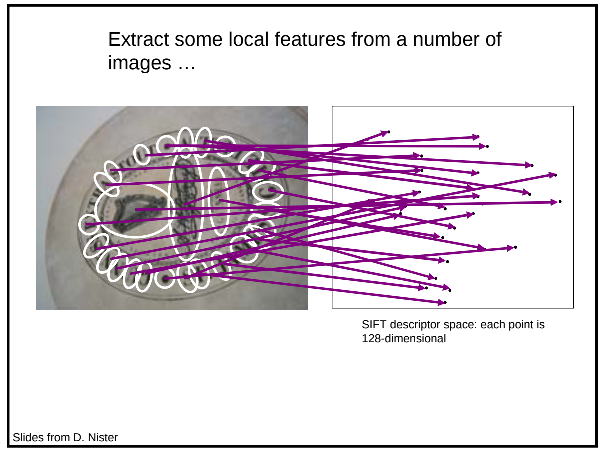
Slide credit: Josef Sivic

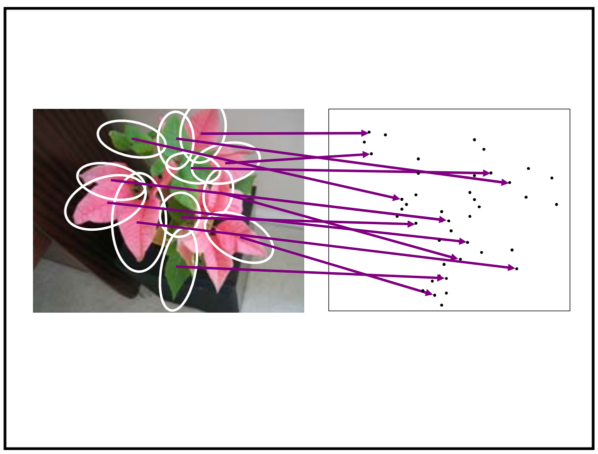
1.Feature detection and representation

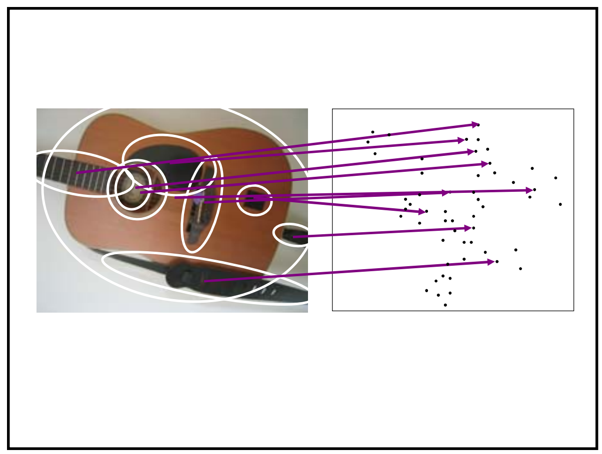
2. Codewords dictionary formation

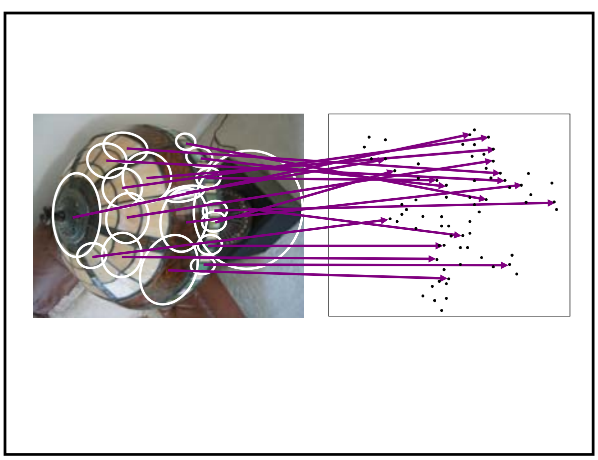


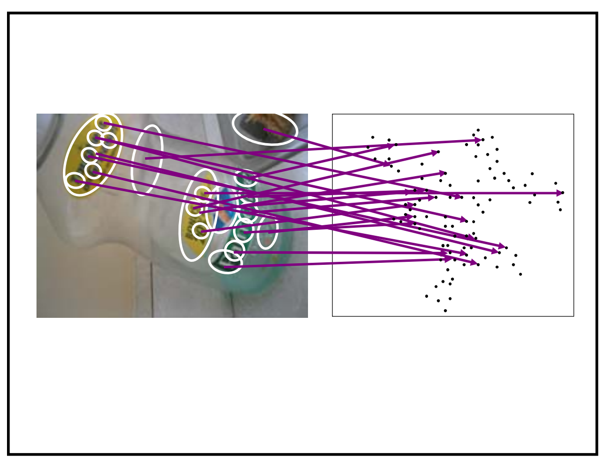

2. Codewords dictionary formation
…
Vector quantization
Slide credit: Josef Sivic

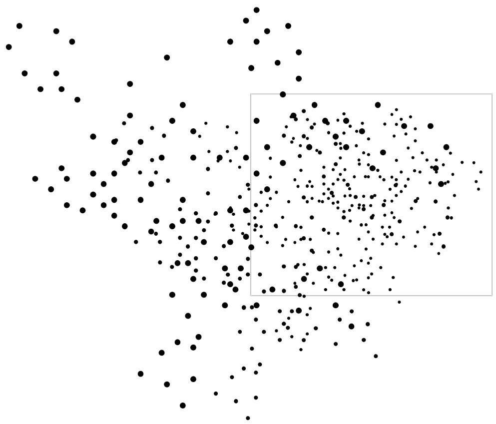

Extract some local features from a number of images …
SIFT descriptor space: each point is 128-dimensional
Slides from D. Nister











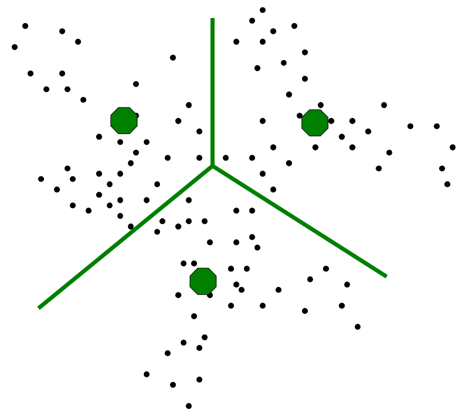
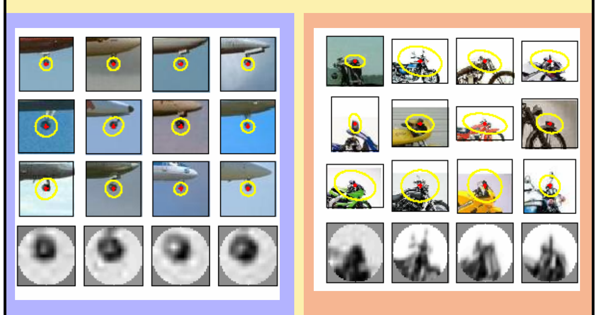
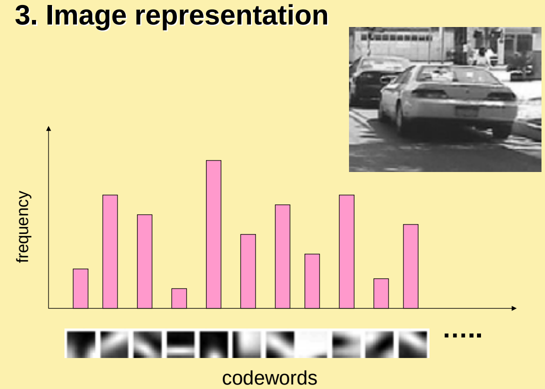


Image patch examples of codewords



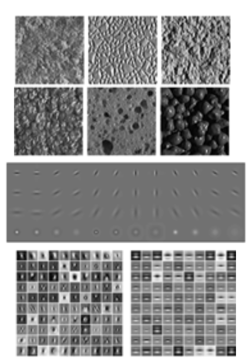
Sivic et al. 2005

3. Image representation



Visual words = textons

- *Texton* = cluster center of filter responses over collection of images [Leung and Malik, 1999]
- Represent texture or material with histogram of texton occurrences (or prototypes of whatever feature type employed)



Visual words and bags of words

- Have a way to represent
  - Individual local image regions as “tokens” / discrete set of visual words
  - Entire image in terms of its distribution of words
- How to use this for indexing task?
- Again, can look to text retrieval for inspiration

Inverted file index

- For each word, store list of documents (pages) where that word occurs

Inverted file index for images

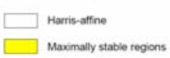
When would using an inverted file reduce the amount of images we need to search/compare?

Figure from Andrew Zisserman, University of Oxford

Video Google [Sivic & Zisserman, 2003]

In each frame independently
determine elliptical regions (segmentation covariant with camera viewpoint)
compute SIFT descriptor for each region [Lowe '99]

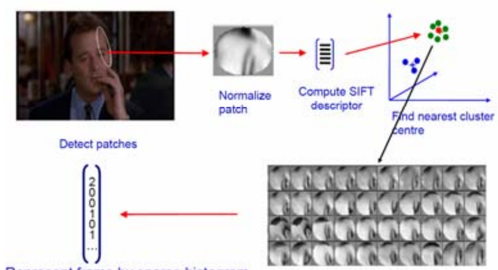
1000+ descriptors per frame



Slide from Andrew Zisserman, University of Oxford

Video Google [Sivic & Zisserman, 2003]

Assign visual words and compute histograms for each key frame in the video



Slide from Andrew Zisserman

Video Google [Sivic & Zisserman, 2003]

- Stage 1: generate a short list of possible frames using bag of visual word representation:

1. Accumulate all visual words within the query region
2. Use "book index" to find other frames with these words
3. Compute similarity for frames which share at least one word

- Generates a tf-idf ranked list of all the frames in dataset

Slide from Andrew Zisserman, University of Oxford

tf-idf weighting

- Term frequency – inverse document frequency
- 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

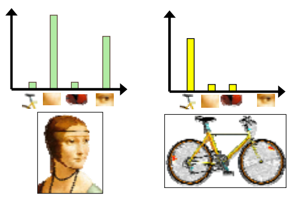
Total number of documents in database

Number of occurrences of word i in whole database

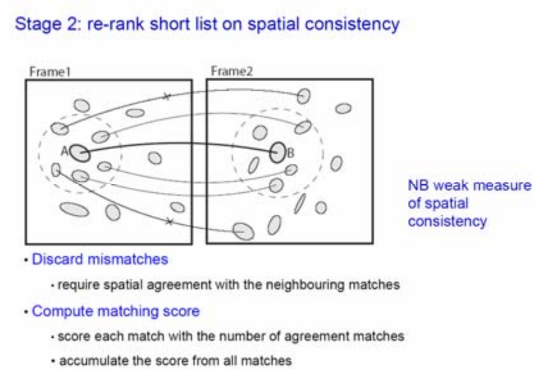
Comparing bags of words

- Rank frames by dot product between their (tf-idf weighted) occurrence counts

[1 8 1 4]' ° [5 1 1 0]



Video Google [Sivic & Zisserman, 2003]

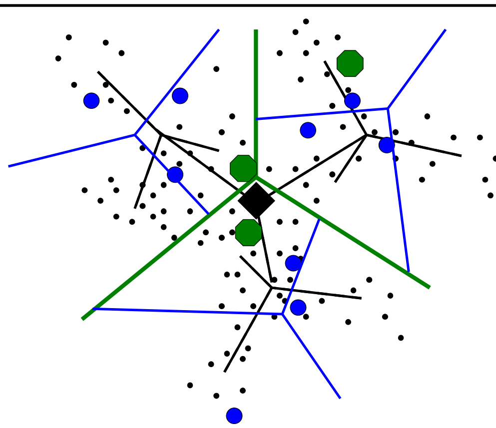


Video Google demo

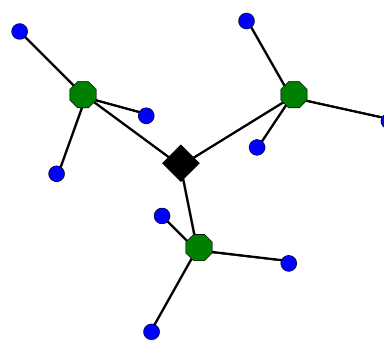
http://www.robots.ox.ac.uk/~vgg/research/vgoogle/index.html

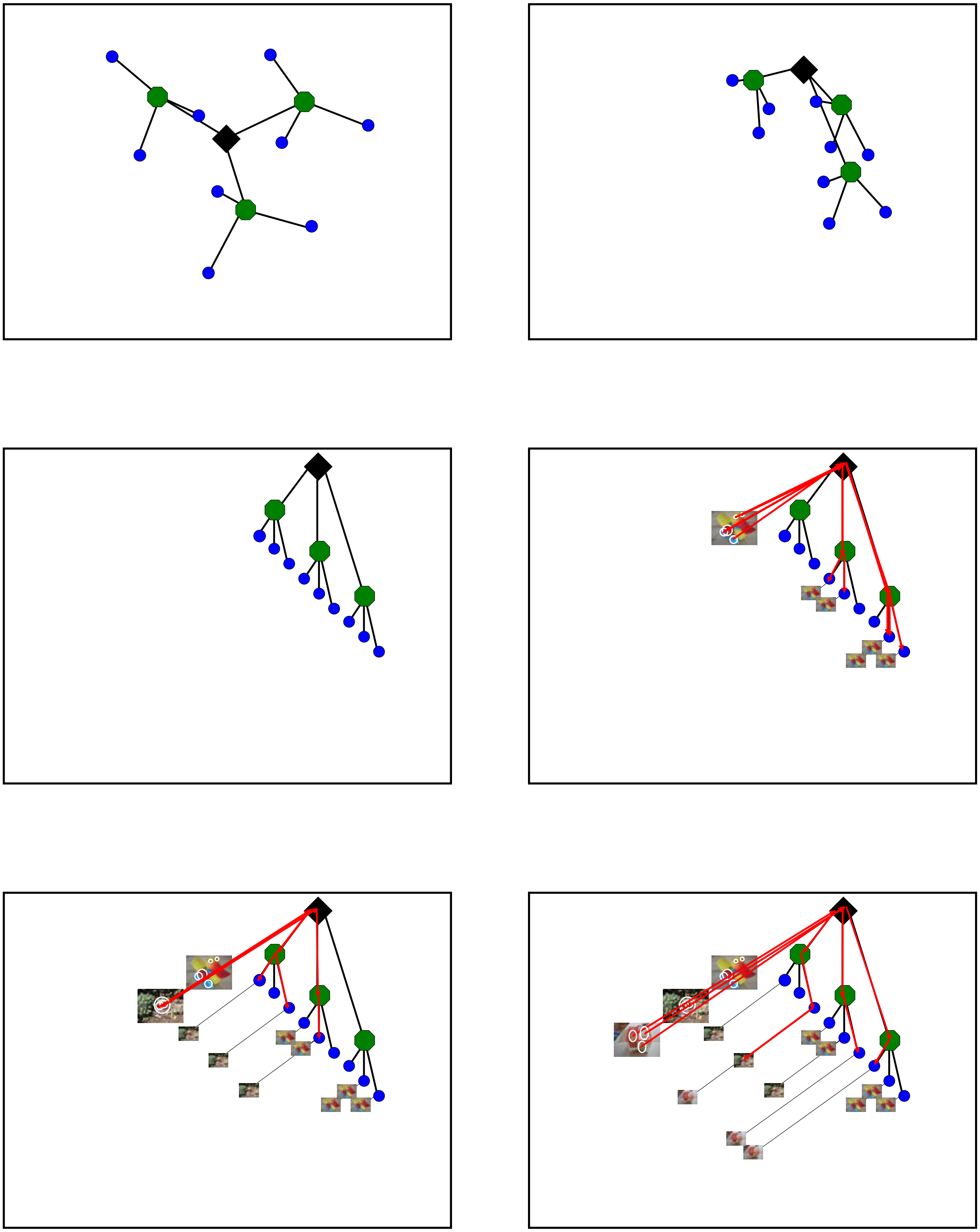
Hierarchical vocabulary

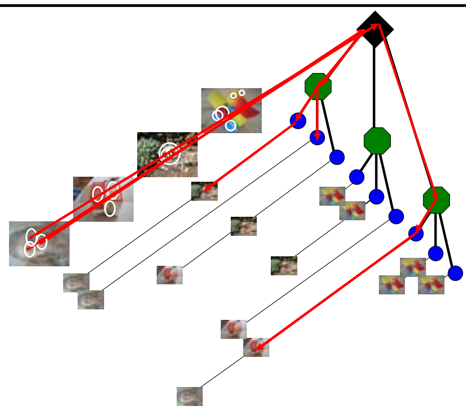
- To manage a large vocabulary efficiently, we can form the quantization of feature space in a hierarchical way
- David Nister & Henrik Stewenius, Scalable Recognition with a Vocabulary Tree, CVPR 2006

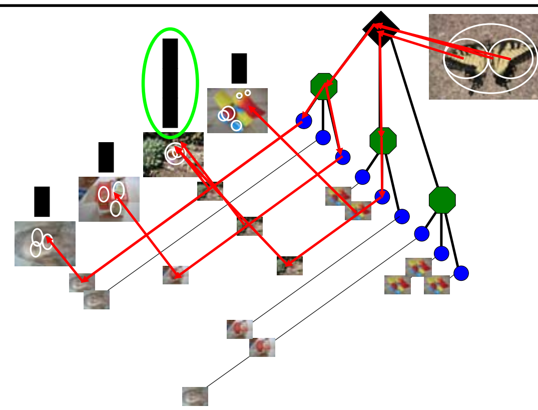


Slides from D. Nister

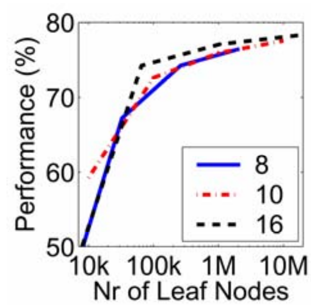








What is the computational advantage of the hierarchical representation bag of words, vs. a flat vocabulary?



Larger vocabularies can be advantageous…

But what happens if it is too large?

Bag of words representation: advantages

- Flexibility comes with ignoring geometry (?)
- Compact description, yet rich
- Local features → vector
  - Usable representation
  - Relatively efficient learning
- Yields good results in practice

Bag of words representation: Issues

- Flexibility comes with ignoring geometry (!)
- Background/foreground treated at once
- Vocabulary formation
  - Number of words/clusters?
  - Universal, or dataset specific?
  - May be expensive
- How to localize/segment object?

Making the Sky Searchable: Fast Geometric Hashing for Automated Astrometry

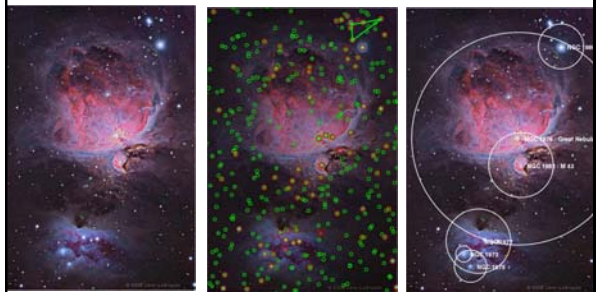
Sam Roweis, Dustin Lang & Keir Mierle
University of Toronto

David Hogg & Michael Blanton
New York University

Check out the slides at:
cosmo.nyu.edu/hogg/research/2006/09/28/astrometry_google.ppt

Example

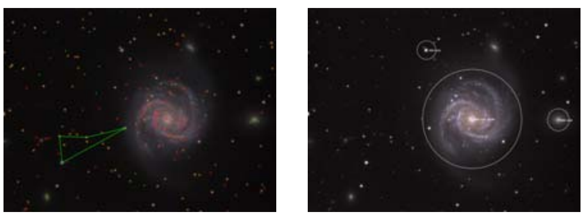
Roweis, Lang, Mierle, Hogg & Blanton



A shot of the Great Nebula, by Jerry Lodriguss (c.2006), from astropix.com
http://astrometry.net/gallery.html

Example

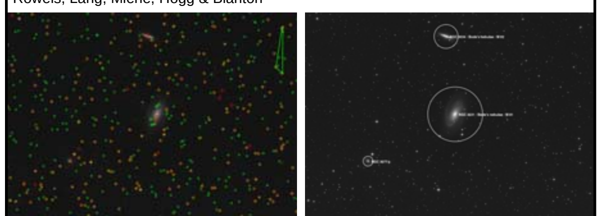
Roweis, Lang, Mierle, Hogg & Blanton



An amateur shot of M100, by Filippo Ciferri (c.2007) from flickr.com
http://astrometry.net/gallery.html

Example

Roweis, Lang, Mierle, Hogg & Blanton



A beautiful image of Bode's nebula (c.2007) by Peter Bresseler, from starlightfriend.de
http://astrometry.net/gallery.html

Today: key ideas

- Invariant features: distinctive matches possible in spite of significant view change, useful for wide baseline stereo
- 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

Coming up

- Next week:
  - Model-based object recognition
  - Face recognition, detection
- Read FP 18.1-18.5, FP 22.1-22.3