

The slide features three images at the top: a stereo pair of a building facade with yellow bounding boxes, a 3D reconstruction of the same scene, and a group photo of four people. Below the images is the title "Instance recognition". The date and presenter information "Tues April 4 Kristen Grauman UT Austin" are centered. On the right is a diagram of a graph with nodes and edges. On the left is a small table with two columns of text.

| Index | |
|-------|---------------------------|
| 1 | Image 1/2: From Desktop |
| 2 | Image 2/2: From Desktop |
| 3 | Image 3/2: From Desktop |
| 4 | Image 4/2: From Desktop |
| 5 | Image 5/2: From Desktop |
| 6 | Image 6/2: From Desktop |
| 7 | Image 7/2: From Desktop |
| 8 | Image 8/2: From Desktop |
| 9 | Image 9/2: From Desktop |
| 10 | Image 10/2: From Desktop |
| 11 | Image 11/2: From Desktop |
| 12 | Image 12/2: From Desktop |
| 13 | Image 13/2: From Desktop |
| 14 | Image 14/2: From Desktop |
| 15 | Image 15/2: From Desktop |
| 16 | Image 16/2: From Desktop |
| 17 | Image 17/2: From Desktop |
| 18 | Image 18/2: From Desktop |
| 19 | Image 19/2: From Desktop |
| 20 | Image 20/2: From Desktop |
| 21 | Image 21/2: From Desktop |
| 22 | Image 22/2: From Desktop |
| 23 | Image 23/2: From Desktop |
| 24 | Image 24/2: From Desktop |
| 25 | Image 25/2: From Desktop |
| 26 | Image 26/2: From Desktop |
| 27 | Image 27/2: From Desktop |
| 28 | Image 28/2: From Desktop |
| 29 | Image 29/2: From Desktop |
| 30 | Image 30/2: From Desktop |
| 31 | Image 31/2: From Desktop |
| 32 | Image 32/2: From Desktop |
| 33 | Image 33/2: From Desktop |
| 34 | Image 34/2: From Desktop |
| 35 | Image 35/2: From Desktop |
| 36 | Image 36/2: From Desktop |
| 37 | Image 37/2: From Desktop |
| 38 | Image 38/2: From Desktop |
| 39 | Image 39/2: From Desktop |
| 40 | Image 40/2: From Desktop |
| 41 | Image 41/2: From Desktop |
| 42 | Image 42/2: From Desktop |
| 43 | Image 43/2: From Desktop |
| 44 | Image 44/2: From Desktop |
| 45 | Image 45/2: From Desktop |
| 46 | Image 46/2: From Desktop |
| 47 | Image 47/2: From Desktop |
| 48 | Image 48/2: From Desktop |
| 49 | Image 49/2: From Desktop |
| 50 | Image 50/2: From Desktop |
| 51 | Image 51/2: From Desktop |
| 52 | Image 52/2: From Desktop |
| 53 | Image 53/2: From Desktop |
| 54 | Image 54/2: From Desktop |
| 55 | Image 55/2: From Desktop |
| 56 | Image 56/2: From Desktop |
| 57 | Image 57/2: From Desktop |
| 58 | Image 58/2: From Desktop |
| 59 | Image 59/2: From Desktop |
| 60 | Image 60/2: From Desktop |
| 61 | Image 61/2: From Desktop |
| 62 | Image 62/2: From Desktop |
| 63 | Image 63/2: From Desktop |
| 64 | Image 64/2: From Desktop |
| 65 | Image 65/2: From Desktop |
| 66 | Image 66/2: From Desktop |
| 67 | Image 67/2: From Desktop |
| 68 | Image 68/2: From Desktop |
| 69 | Image 69/2: From Desktop |
| 70 | Image 70/2: From Desktop |
| 71 | Image 71/2: From Desktop |
| 72 | Image 72/2: From Desktop |
| 73 | Image 73/2: From Desktop |
| 74 | Image 74/2: From Desktop |
| 75 | Image 75/2: From Desktop |
| 76 | Image 76/2: From Desktop |
| 77 | Image 77/2: From Desktop |
| 78 | Image 78/2: From Desktop |
| 79 | Image 79/2: From Desktop |
| 80 | Image 80/2: From Desktop |
| 81 | Image 81/2: From Desktop |
| 82 | Image 82/2: From Desktop |
| 83 | Image 83/2: From Desktop |
| 84 | Image 84/2: From Desktop |
| 85 | Image 85/2: From Desktop |
| 86 | Image 86/2: From Desktop |
| 87 | Image 87/2: From Desktop |
| 88 | Image 88/2: From Desktop |
| 89 | Image 89/2: From Desktop |
| 90 | Image 90/2: From Desktop |
| 91 | Image 91/2: From Desktop |
| 92 | Image 92/2: From Desktop |
| 93 | Image 93/2: From Desktop |
| 94 | Image 94/2: From Desktop |
| 95 | Image 95/2: From Desktop |
| 96 | Image 96/2: From Desktop |
| 97 | Image 97/2: From Desktop |
| 98 | Image 98/2: From Desktop |
| 99 | Image 99/2: From Desktop |
| 100 | Image 100/2: From Desktop |

Last time

- Depth from stereo: main idea is to triangulate from corresponding image points.
- Epipolar geometry defined by two cameras
 - We've assumed known extrinsic parameters relating their poses
- Epipolar constraint limits where points from one view will be imaged in the other
 - Makes search for correspondences quicker
- To estimate depth
 - Limit search by epipolar constraint
 - Compute correspondences, incorporate matching preferences

Stereo error sources

- Low-contrast ; textureless image regions
- Occlusions
- Camera calibration errors
- Violations of *brightness constancy* (e.g., specular reflections)
- Large motions

Virtual viewpoint video

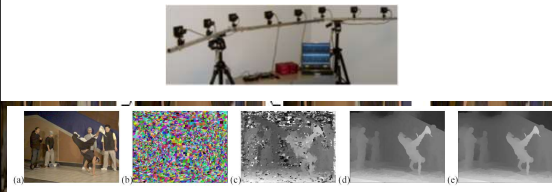


Figure 6: Sample results from stereo reconstruction stage: (a) input color image; (b) color-based segmentation; (c) initial disparity estimates d_{12} ; (d) refined disparity estimates; (e) smoothed disparity estimates $d_s(x)$.

C. Zitnick et al, High-quality video view interpolation using a layered representation, SIGGRAPH 2004.

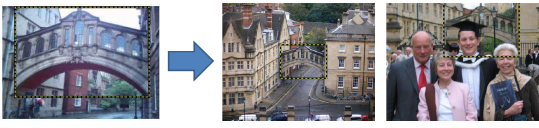
Review questions (on your own)

- When solving for stereo, when is it necessary to break the soft disparity gradient constraint?
- What can cause a disparity value to be undefined?
- Suppose we are given a disparity map indicating offset in the x direction for corresponding points. What does this imply about the layout of the epipolar lines in the two images?

Slide credit: Kristen Grauman

Today

- Instance recognition
 - Indexing local features efficiently
 - Spatial verification models



Recognizing or retrieving specific objects

Example I: Visual search in feature films

Visually defined query

"Groundhog Day" [Rammis, 1993]

"Find this clock"

"Find this place"

Slide credit: J. Sivic

Recognizing or retrieving specific objects

Example II: Search photos on the web for particular places

Find these landmarks ...in these images and 1M more

Slide credit: J. Sivic

Google Goggles

Use pictures to search the web

Get Google Goggles

Android (1.6+ required)

Send Goggles to Android phone

iPhone (iOS 4.0 required)

Send Goggles to iPhone


Text Landmarks Books Contact info Artists Wine Logos

Lammfleisch vom Bauern mit Schindeln, Tomatensauce und Basilikum-Grochli


Lamm chops from the farmer with the shallots, tomato sauce and basil grochli

Why is it difficult?


Want to find the object despite possibly large changes in scale, viewpoint, lighting and partial occlusion




Scale



Viewpoint



Lighting



Occlusion

Slide credit: J. Sivic

Recall: matching local features




?


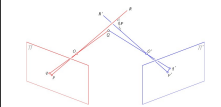
Image 1
Image 2

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)

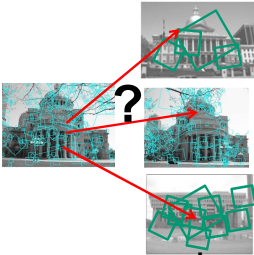
Slide credit: Kristen Grauman

Multi-view matching



Matching two given views for depth

vs



Search for a matching view for recognition

Slide credit: Kristen Grauman

Indexing local features

Slide credit: Kristen Graumar

Indexing local features

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

Slide credit: Kristen Graumar

Indexing local features

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

Slide credit: Kristen Graumar

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?
- Possible solutions:
 - Inverted file
 - Nearest neighbor data structures
 - Kd-trees
 - Hashing

Slide credit: Kristen Graumar



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".

Slide credit: Kristen Graumar

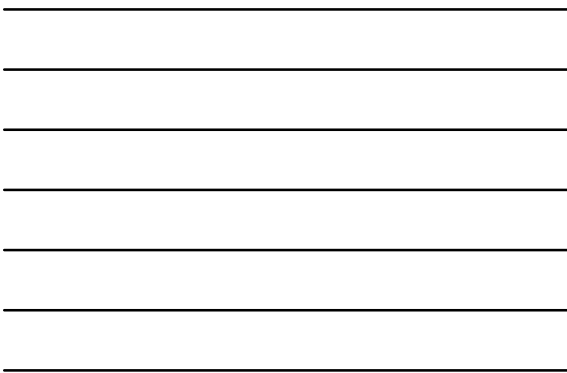


Visual words

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

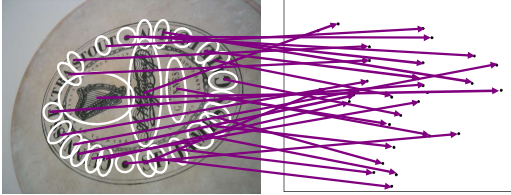
- Quantize via clustering, let cluster centers be the prototype "words"
- Determine which word to assign to each new image region by finding the closest cluster center.

Slide credit: Kristen Graumar



Visual words: main idea

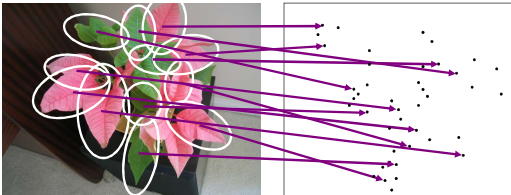
- Extract some local features from a number of images ...



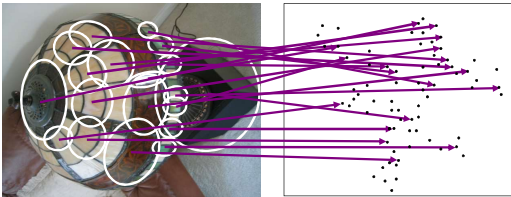
e.g., SIFT descriptor space: each point is 128-dimensional

Slide credit: D. Nister, CVPR 2006

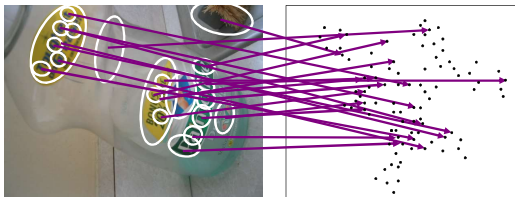
Visual words: main idea



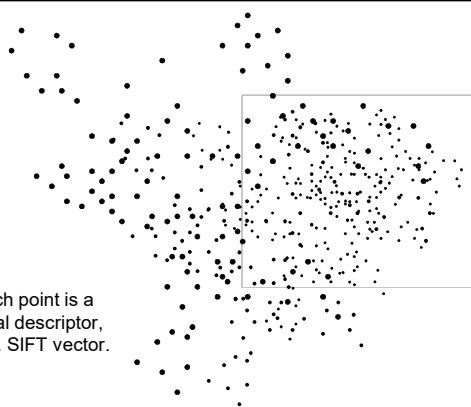
Visual words: main idea

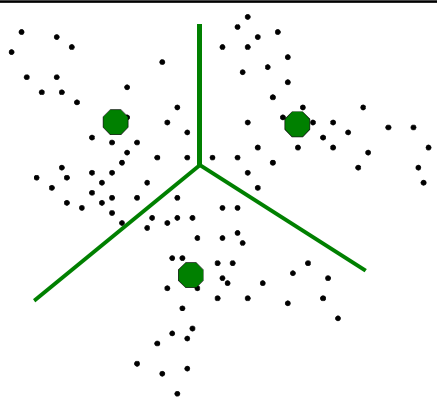


Visual words: main idea



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





Visual words

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

Figure from Sivic & Zisserman, ICCV 2003

Visual words and textons

- 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

Slide credit: Kristen Grauman

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

Slide credit: Kristen Grauman

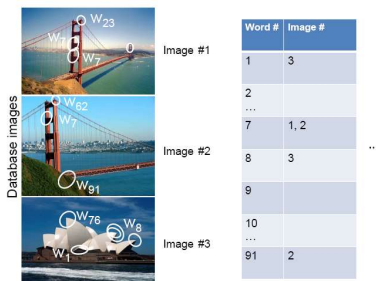
Visual vocabulary formation

Issues:

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

Slide credit: Kristen Graumar

Inverted file index

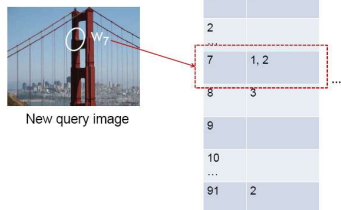


- Database images are loaded into the index mapping words to image numbers

Slide credit: Kristen Graumar

Inverted file index

When will this give us a significant gain in efficiency?



- New query image is mapped to indices of database images that share a word.

Slide credit: Kristen Graumar

Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

Slide credit: Kristen Grauman

Analogy to documents

Of all the sensory impressions proceeding to the brain, the visual experiences are the dominant ones. Our perception of the world around us is based essentially on the messages that come to our eyes. For a long time, the retinal image was considered as a movie strip. It was discovered that we know the perceptual more completely following the message with to the various parts of the cortex. Hubel and Wiesel have demonstrated that the message about image falling on the retina undergoes a fine-grained analysis in a system of nerve cells stored in columns. In this system each cell has its specific function and is responsible for a specific detail in the pattern of the retinal image.

sensory, brain, visual, perception, retinal, cerebral cortex, eye, cell, optical nerve, image Hubel, Wiesel

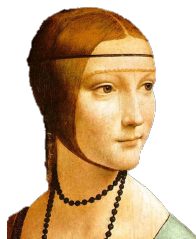
China is forecasting a trade surplus of \$90bn (£51bn) to \$100bn this year, a threefold increase on 2004's \$32bn. The Commerce Ministry said the surplus would be created by a predicted 30% increase in exports, compared with \$750bn, compared with \$660bn. The increase will annoy the US because of China's deliberate policy to keep the yuan undervalued. The government also needs to increase demand so that the country can absorb the yuan against the dollar. The US has permitted it to trade within a narrow range but the US wants the yuan to be allowed to rise freely. However, Beijing has made it clear that it will take its time and tread carefully before allowing the yuan to rise further in value.

China, trade, surplus, commerce, exports, imports, US, yuan, bank, domestic, foreign, increase, trade, value

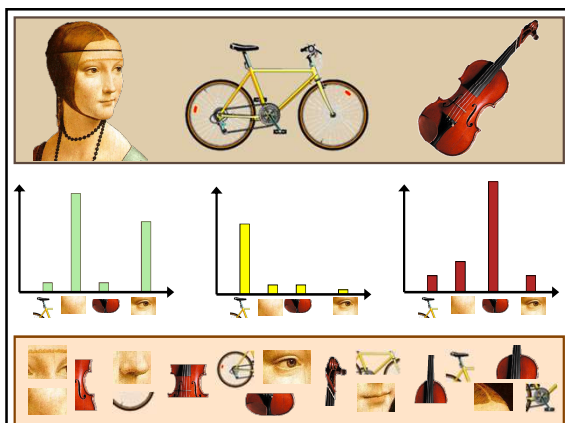
ICCV 2005 short course, L. Fei-Fei

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.

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]

\vec{d}_j

[5 1 1 0]

\vec{q}

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

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

for vocabulary of V words

Slide credit: Kristen Grauman

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)

Number of occurrences of word i in document d


Number of words in document d

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


Total number of documents in database

Number of documents word i occurs in, in whole database

Inverted file index and bags of words similarity




| Word # | Image # |
|--------|---------|
| 1 | 3 |
| 2 | |
| 7 | 1, 2 |
| 8 | 3 |
| 9 | |
| 10 | |
| ... | |
| 91 | 2 |




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


Slide credit: Kristen Graumar

Bags of words for content-based image retrieval




→





→

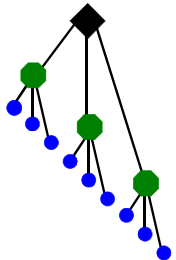


Slide from Andrew Zisserman
Sivic & Zisserman, ICCV 2003

Visual Object Recognition Tutorial

Vocabulary Tree

- Training: Filling the tree

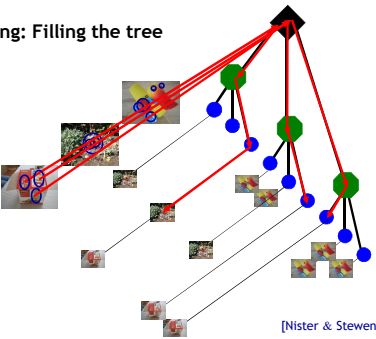


[Nister & Stewenius, CVPR'06]
K. Grauman, B. Leibe Slide credit: David Nister

Visual Object Recognition Tutorial

Vocabulary Tree

- Training: Filling the tree

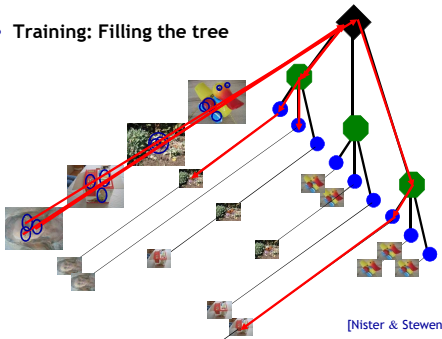


[Nister & Stewenius, CVPR'06]
K. Grauman, B. Leibe Slide credit: David Nister

Visual Object Recognition Tutorial

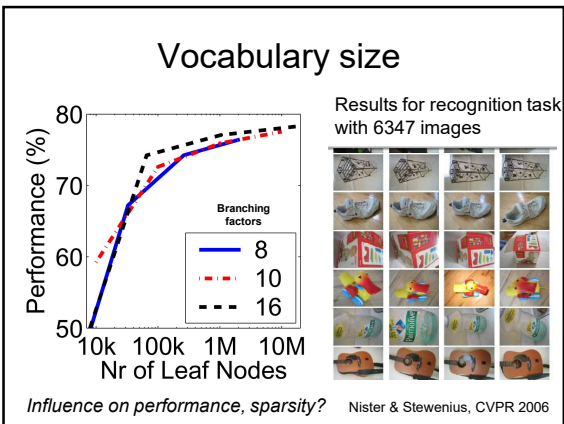
Vocabulary Tree

- Training: Filling the tree



[Nister & Stewenius, CVPR'06]
K. Grauman, B. Leibe Slide credit: David Nister 45

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



Bags of words: pros and cons

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

- basic model ignores geometry – must verify afterwards, or encode via features
- background and foreground mixed when bag covers whole image
- optimal vocabulary formation remains unclear

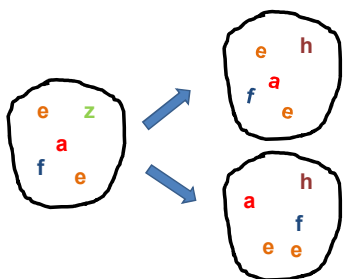
Slide credit: Kristen Grauman

Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- How to score the retrieval results?

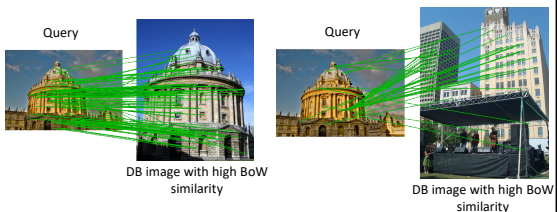
Slide credit: Kristen Grauman

Which matches better?



Derek Hoiem

Spatial Verification



Both image pairs have many visual words in common.

Slide credit: Ondrej Chum

Spatial Verification

Query Query

DB image with high BoW similarity DB image with high BoW similarity

Only some of the matches are mutually consistent

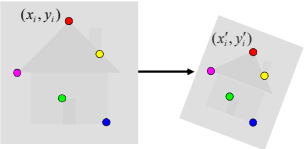
Slide credit: Ondrej Chum

Spatial Verification: two basic strategies

- RANSAC
 - Typically sort by BoW similarity as initial filter
 - Verify by checking support (inliers) for possible transformations
 - e.g., "success" if find a transformation with $> N$ inlier correspondences
- Generalized Hough Transform
 - Let each matched feature cast a vote on location, scale, orientation of the model object
 - Verify parameters with enough votes

RANSAC verification

Recall: Fitting an affine transformation

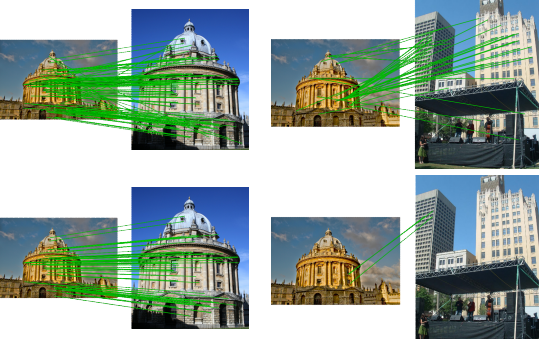


Approximates viewpoint changes for roughly planar objects and roughly orthographic cameras.

$$\begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$$

$$\begin{bmatrix} \dots & \dots & \dots & \dots & \dots & \dots \\ x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} \dots \\ x'_i \\ y'_i \\ \dots \end{bmatrix}$$

RANSAC verification



Spatial Verification: two basic strategies

- **RANSAC**
 - Typically sort by BoW similarity as initial filter
 - Verify by checking support (inliers) for possible transformations
 - e.g., "success" if find a transformation with > N inlier correspondences
- **Generalized Hough Transform**
 - Let each matched feature cast a vote on location, scale, orientation of the model object
 - Verify parameters with enough votes

Voting: Generalized Hough Transform

- If we use scale, rotation, and translation invariant local features, then each feature match gives an alignment hypothesis (for scale, translation, and orientation of model in image).



Model

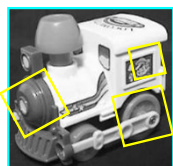


Novel image

Adapted from Lana Lazebnik

Voting: Generalized Hough Transform

- A hypothesis generated by a single match may be unreliable,
- So let each match **vote** for a hypothesis in Hough space



Model



Novel image

Gen Hough Transform details (Lowe's system)

- **Training phase:** For each model feature, record 2D location, scale, and orientation of model (relative to normalized feature frame)
- **Test phase:** Let each match btwn a test SIFT feature and a model feature vote in a 4D Hough space
 - Use broad bin sizes of 30 degrees for orientation, a factor of 2 for scale, and 0.25 times image size for location
 - Vote for two closest bins in each dimension
- Find all bins with at least three votes and perform geometric verification
 - Estimate least squares *affine* transformation
 - Search for additional features that agree with the alignment

David G. Lowe. ["Distinctive image features from scale-invariant keypoints."](#) *IJCV* 60 (2), pp. 91-110, 2004.

Slide credit: Lana Lazebnik

Example result

Background subtract for model boundaries Objects recognized, Recognition in spite of occlusion

[Lowe]

Recall: difficulties of voting

- Noise/clutter can lead to as many votes as true target
- Bin size for the accumulator array must be chosen carefully
- In practice, good idea to make broad bins and spread votes to nearby bins, since verification stage can prune bad vote peaks.

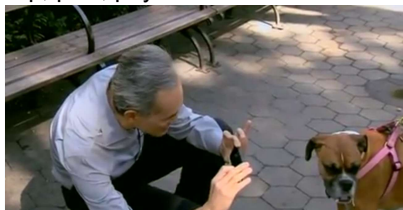
Gen Hough vs RANSAC

| | |
|--|---|
| <p><u>GHT</u></p> <ul style="list-style-type: none"> • Single correspondence -> vote for all consistent parameters • Represents uncertainty in the model parameter space • Linear complexity in number of correspondences and number of voting cells; beyond 4D vote space impractical • Can handle high outlier ratio | <p><u>RANSAC</u></p> <ul style="list-style-type: none"> • Minimal subset of correspondences to estimate model -> count inliers • Represents uncertainty in image space • Must search all data points to check for inliers each iteration • Scales better to high-d parameter spaces |
|--|---|

Slide credit: Kristen Grauman

Example applications

- Snap, pick, pay



- <https://www.usatoday.com/videos/tech/2014/10/31/18261641/>

Slide credit: Kristen Graumar

Example Applications



Mobile tourist guide

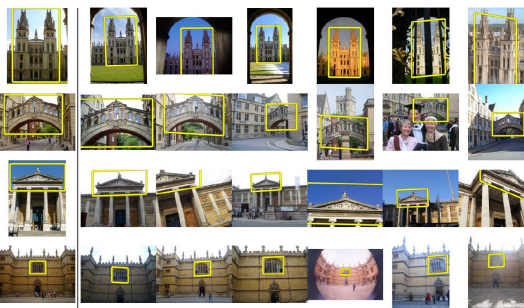
- Self-localization
- Object/building recognition
- Photo/video augmentation

B. Leibe

[Quack, Leibe, Van Gool, CIVR'08]

Visual Object Recognition Tutorial

Application: Large-Scale Retrieval



Query Results from 5k Flickr images (demo available for 100k set)

[Philbin CVPR'07]

Visual Object Recognition Tutorial

Web Demo: Movie Poster Recognition

50'000 movie posters indexed

Query-by-image from mobile phone available in Switzerland

1. Take a picture with your mobile phone camera
 2. Send it:
 - in Switzerland to 3333 (Charge Customers 079 334 3700),
 - in Germany to 86000,
 - everywhere else to info@kooaba.ch
 3. Search result is sent straight to your phone.

http://www.kooaba.com/en/products_engine.html#

Instance recognition: remaining issues

- How to summarize the content of an entire image? And gauge overall similarity?
- How large should the vocabulary be? How to perform quantization efficiently?
- Is having the same set of visual words enough to identify the object/scene? How to verify spatial agreement?
- [How to score the retrieval results?](#)

Kristen Grauman

Scoring retrieval quality

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

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

Results (ordered):

Slide credit: Ondrej Chum

Recognition via alignment

Pros:

- Effective when we are able to find reliable features within clutter
- Great results for matching specific instances

Cons:

- Scaling with number of models
- Spatial verification as post-processing – not seamless, expensive for large-scale problems
- Not suited for category recognition.

Summary

- **Matching local invariant features**
 - Useful not only to provide matches for multi-view geometry, but also to find objects and scenes.
- **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
- **Recognition of instances via alignment**: matching local features followed by spatial verification
 - Robust fitting : RANSAC, GHT

Kristen Grauman
