

Indexing local features

Wed March 30
Prof. Kristen Grauman
UT-Austin

Index	Region
1	Region 1
2	Region 2
3	Region 3
4	Region 4
5	Region 5
6	Region 6
7	Region 7
8	Region 8
9	Region 9
10	Region 10
11	Region 11
12	Region 12
13	Region 13
14	Region 14
15	Region 15
16	Region 16
17	Region 17
18	Region 18
19	Region 19
20	Region 20
21	Region 21
22	Region 22
23	Region 23
24	Region 24
25	Region 25
26	Region 26
27	Region 27
28	Region 28
29	Region 29
30	Region 30
31	Region 31
32	Region 32
33	Region 33
34	Region 34
35	Region 35
36	Region 36
37	Region 37
38	Region 38
39	Region 39
40	Region 40
41	Region 41
42	Region 42
43	Region 43
44	Region 44
45	Region 45
46	Region 46
47	Region 47
48	Region 48
49	Region 49
50	Region 50

Matching local features

Kristen Grauman

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)

Kristen Grauman

Matching local features

Image 1 Image 2

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

Kristen Grauman

Indexing local features

Kristen Grauman

Indexing local features

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

Descriptor's feature space

Kristen Grauman

Indexing local features

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

The diagram illustrates the process of indexing local features. On the left, several 'Database images' are shown with red circles highlighting specific local features. These features are mapped into a 'Descriptor's feature space', represented as a 2D scatter plot where points that are close together indicate similar local content. On the right, a 'Query image' is shown with its own local features being compared against the indexed feature space.

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?

Kristen Graumar

Indexing local features: inverted file index

The screenshot shows an inverted file index with a search bar containing the word 'INDEX'. Below the search bar, a list of words is displayed, each followed by a list of page numbers where that word appears. For example, 'Alabama' is listed with page numbers 128, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000.

- 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".

Kristen Graumar

Text retrieval vs. image search

- What makes the problems similar, different?

Kristen Graumar

Visual words: main idea

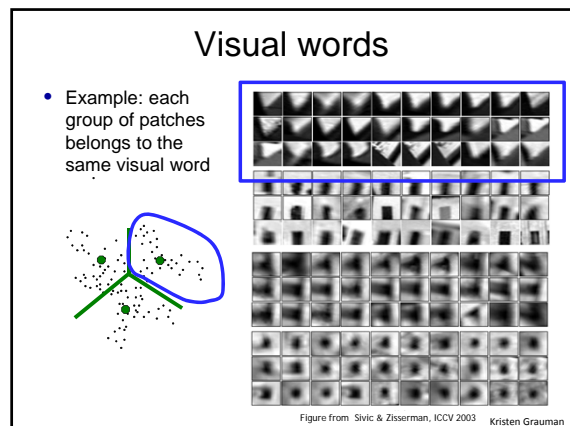
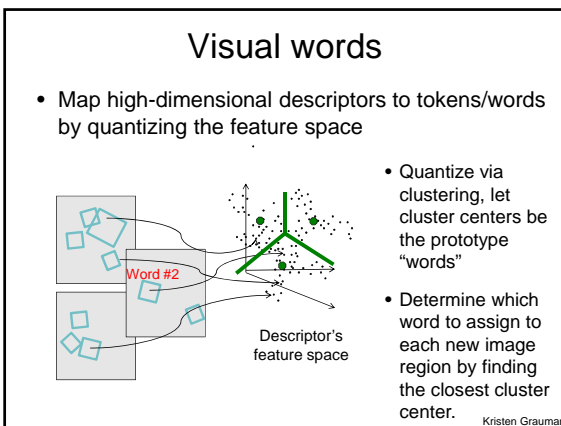
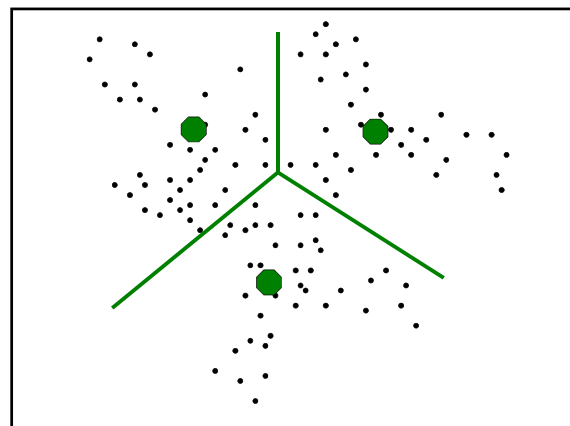
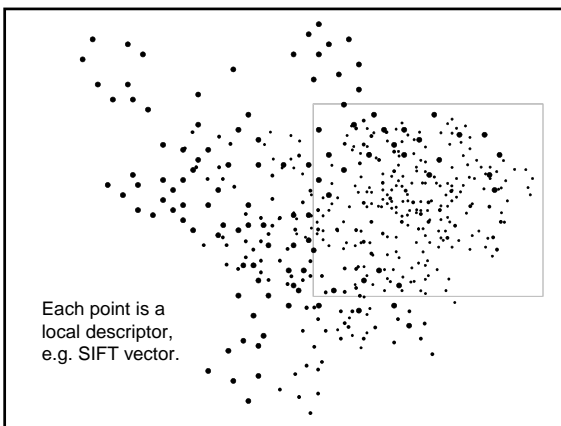
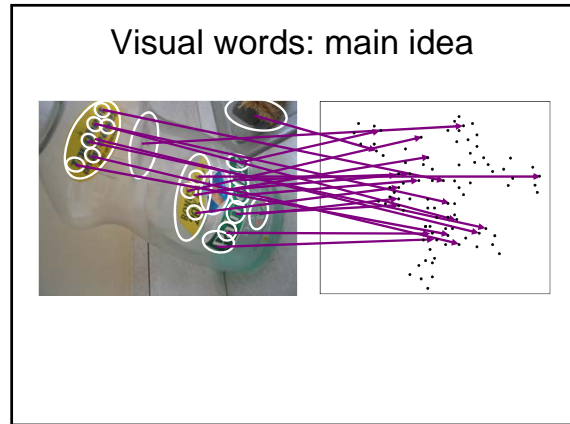
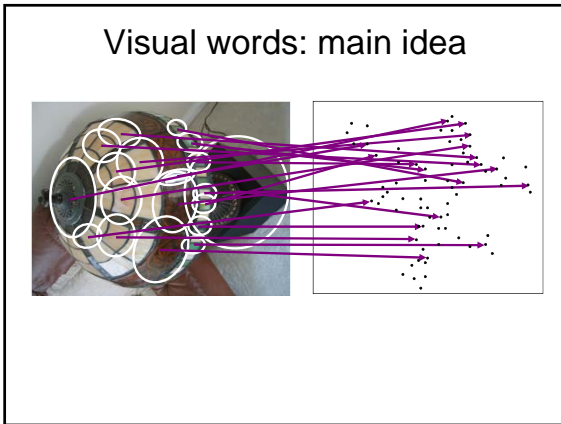
- Extract some local features from a number of images ...

The diagram shows a photograph of a pink flower with green leaves. Purple arrows point from various local features on the flower to a scatter plot of points in a descriptor space. The text below the plot states: "e.g., SIFT descriptor space: each point is 128-dimensional".

Slide credit: D. Nister, CVPR 2006

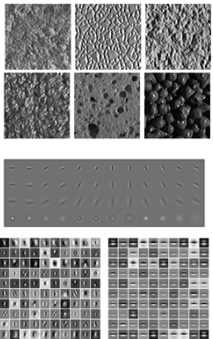
Visual words: main idea

This diagram is identical to the one in the previous slide, showing a flower image with arrows pointing to a scatter plot of points in a descriptor space.



Visual words and textons

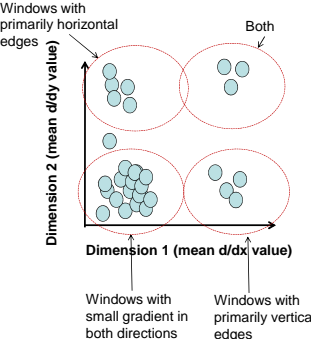
- First explored for texture and material representations
- *Texton* = cluster center of filter responses over collection of images
- Describe textures and materials based on distribution of prototypical texture elements.



Leung & Malik 1999; Varma & Zisserman, 2002

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

Kristen Grauman

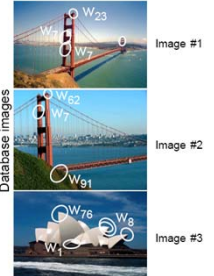
Visual vocabulary formation

Issues:

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

Kristen Grauman

Inverted file index



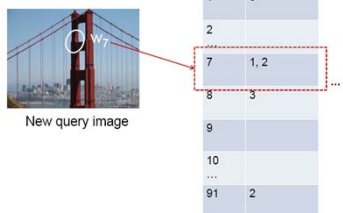
Word #	Image #
1	3
2	
...	
7	1, 2
8	3
9	
10	
...	
91	2

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

Kristen Grauman

Inverted file index

When will this give us a significant gain in efficiency?



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

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

Kristen Grauman

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

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 reach our eyes. For a long time, the retinal image was considered as a movie of the world. As a movie image, we do not know the perspective, we know the more complete image following the various paths to the various parts of the cortex. Hubel and Wiesel have demonstrated that the message about the image falling on the retina undergoes a *coarse 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.*

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 \$66bn in 2004. "It's a bit annoying!" said a senior official. "China's government has agreed to let the yuan rise against the dollar, but the US wants the yuan to be allowed to rise freely. However, Beijing has made it clear it will take its time and tread carefully before allowing the yuan to rise further in value."

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

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

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

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

 n_{id}

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

Total number of documents in database

 N

Number of documents word i occurs in, in whole database

 n_i

Kristen Grauman

Bags of words for content-based image retrieval

Visually defined query

"Find this clock"

"Find this place"

"Groundhog Day" [Rammis, 1993]

Slide from Andrew Zisserman
Sivic & Zisserman, ICCV 2003

Example

retrieved shots

Start Frame 52987 Key Frame 53025 End Frame 53025

Start Frame 54342 Key Frame 54376 End Frame 54644

Start Frame 51176 Key Frame 52251 End Frame 52544

Start Frame 54879 Key Frame 54934 End Frame 54934

Start Frame 39097 Key Frame 39126 End Frame 39300

Start Frame 48769 Key Frame 49026 End Frame 49091

Start Frame 39381 Key Frame 39576 End Frame 39750

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

Visual Object Recognition Tutorial

K. Grauman, B. Leibe

32

Scoring retrieval quality

Query:

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

Results (ordered):

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

Slide credit: Andrei Chum

Vocabulary Trees: hierarchical clustering for large vocabularies

- Tree construction:

[Nister & Stewenius, CVPR'06]

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

Visual Object Recognition Tutorial

Vocabulary Tree

- Training: Filling the tree

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

Vocabulary Tree

- Training: Filling the tree

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

Vocabulary Tree

- Training: Filling the tree

[Nister & Stewenius, CVPR'06]
K. Grauman, B. Leibe Slide credit: David Nister 39

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

Vocabulary Tree

- Recognition

RANSAC verification

[Nister & Stewenius, CVPR'06]
Slide credit: David Nister

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

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