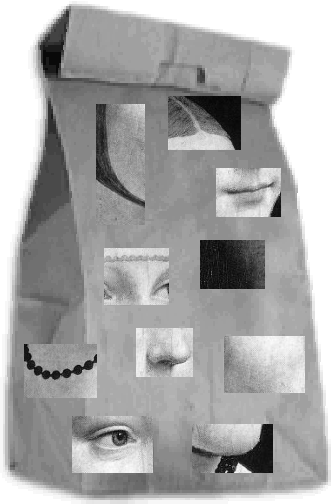
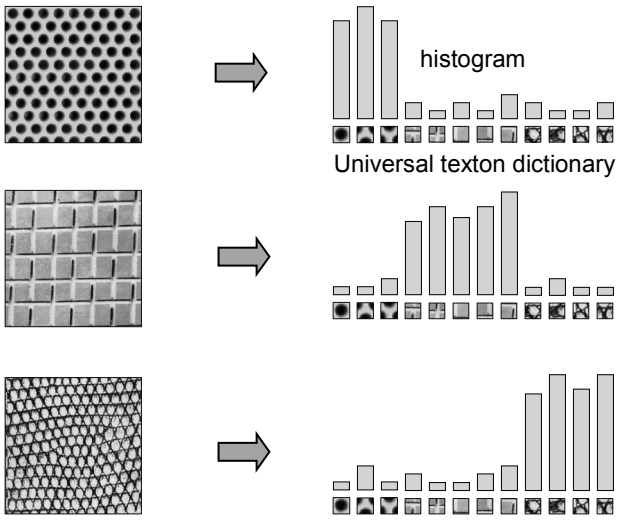
# Bag-of-features models



Slide from Lazebnik: "Many slides adapted from Fei-Fei Li, Rob Fergus, and Antonio Torralba"

# Origin 1: Texture recognition




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

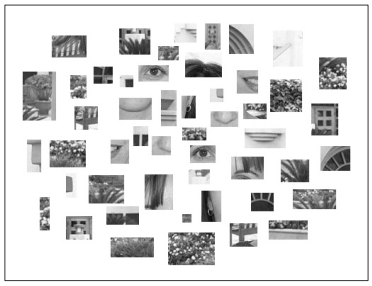
## Image Classification and Document Retrieval

- Similarities:
  - Image and text contain a lot of data.
  - Structure of data hard to understand automatically.
  - Locality is important.
  - Similarity and Classification important
  - Lots of data available.
- Differences
  - Words are powerful features; features less obvious in images.
  - Documents are (more) discrete.

## IR & Bag of words

- “Bag of words” is histogram of words.
- Classification can often be made just on this.
  - “save”, “viagra”, “discount”, “enhance”
  - “meeting”, “deadline”, “papers”, “accepted”
- Issues: how to compare histograms
  - Weight some words (unusual ones) more than others (“multilateral”)
  - Discount very common words (“the”)

# Bags of features for object recognition



face, flowers, building

- Works pretty well for image-level classification

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

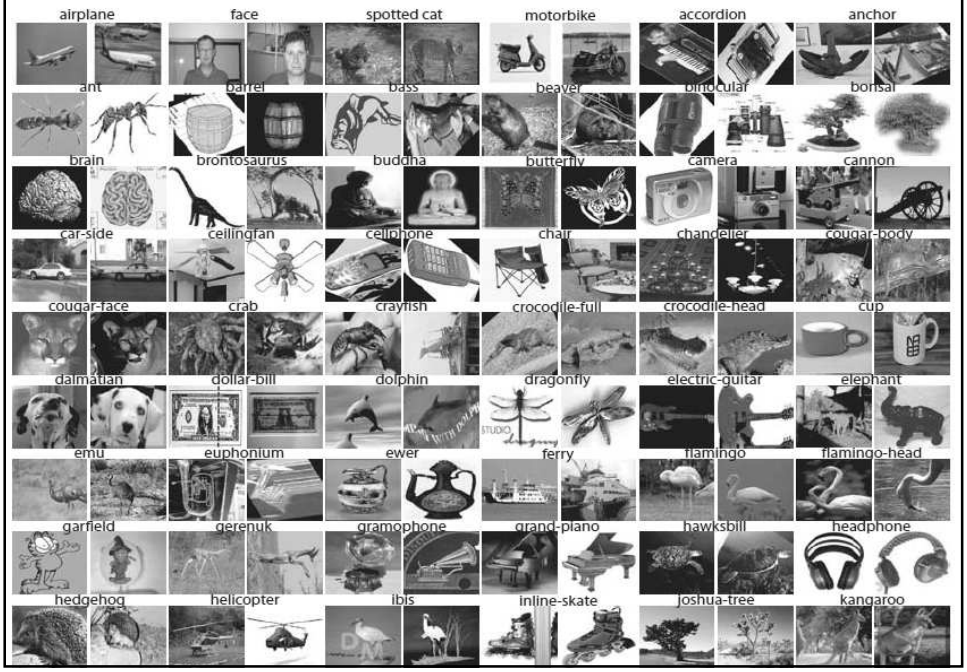
# Bags of features for object recognition

Caltech6 dataset

| class | bag of features | bag of features | Parts-and-shape model |
|---|---|---|---|
| | Zhang et al. (2005) | Willamowski et al. (2004) | 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 |

(Slide Lazebnik)

## Visual Classification: Caltech 101

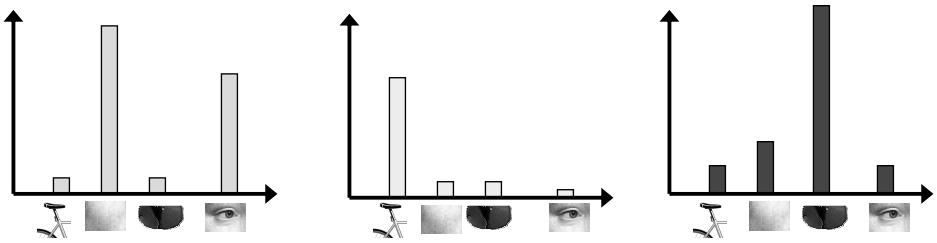



## Visual Words

- SIFT
  - Blob locations and scale give 100s-1000s of “words” per image.
  - Other feature detectors also used
  - Or even random locations
  - Descriptor acts like a word (unique semantics)

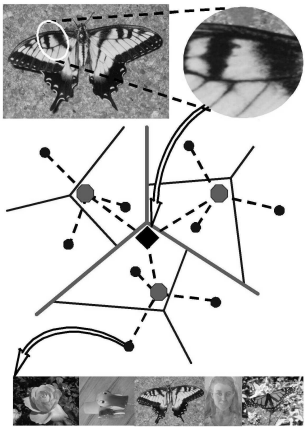
# Discretizing: codebooks

- We need histogram of words – histogram of histograms.
- Turn each SIFT descriptor into a word
  - Run Kmeans on a large set of descriptors.
    - Each cluster center is a “word” in codebook.
  - Assign a new descriptor to nearest center.
  - Compare histograms of these words.



# Visual vocabularies: Issues

- How to choose vocabulary size?
  - Too small: visual words not representative of all patches
  - Too large: quantization artifacts, overfitting
- Generative or discriminative learning?
- Computational efficiency
  - Vocabulary trees (Nister & Stewenius, 2006)



(Slide Lazebnik)