36-350: Data Mining

Handout 3
August 29, 2003

Similarity searching and image retrieval.

Images are represented in the computer as a matrix of **pixels**. The color of each pixel can be represented in different ways, the most typical is RGB.

RGB representation of color—A color is represented by three numbers: a red value, a green value, and a blue value from 0 to 1, representing how the three primary lights should be mixed to produce that color. Black is $(0, 0, 0)$ and white is $(1, 1, 1)$. By varying these numbers, there is an infinite amount of colors that you can display on a computer monitor. (However, not every color can be represented in this way.)

RGB cube—The set of possible colors in the RGB representation.

HSV representation of color—A color is represented by three numbers: a hue, a saturation, and a value (a.k.a. lightness). This representation is not based on color mixing, but tends to be more intuitive.

HSV cube—The set of possible colors in the HSV representation.

Synonym problem—Just as there are different words that mean the same thing, there are different colors that look essentially the same. Thus, for our purposes, we will want to merge similar colors into a small number of "prototypical" colors. For example, merging all shades of red into one "red".

Color quantization—Reducing the number of different colors by merging similar ones. Geometrically, it means cutting the RGB or HSV cube into cells, and assigning one name to all colors in that cell.

Since images are already matrices of numbers, it is possible to compute Euclidean distance directly between images. However, this doesn't work well, because e.g. if an object moves slightly, the Euclidean distance is large.

Bag of colors representation—For each quantized color, the number of times it appears in the image (including zeros). Every image has the same size representation.

Today's data is photographs of flowers, photographs of tigers, and photographs of the ocean.

	violetred4	goldenrod	lightskyblue4	gray58.2
flower1	59	0	0	0
flower2	128	0	0	0
flower3	166	75	0	0
tiger1	0	4	457	85
tiger2	0	0	0	2
tiger3	0	0	0	115
ocean1	0	0	4326	433
ocean2	0	0	2761	142
ocean3	0	0	1179	8596

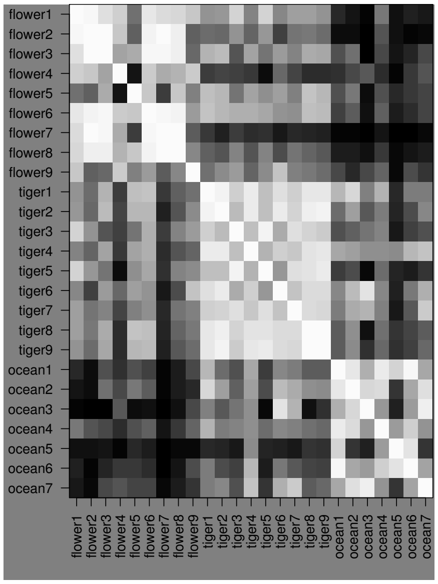
Procedure:

1. The user provides an image Q. Convert it into color counts.
2. For each image in the collection, measure the distance to Q.
3. Return the images with smallest distance.

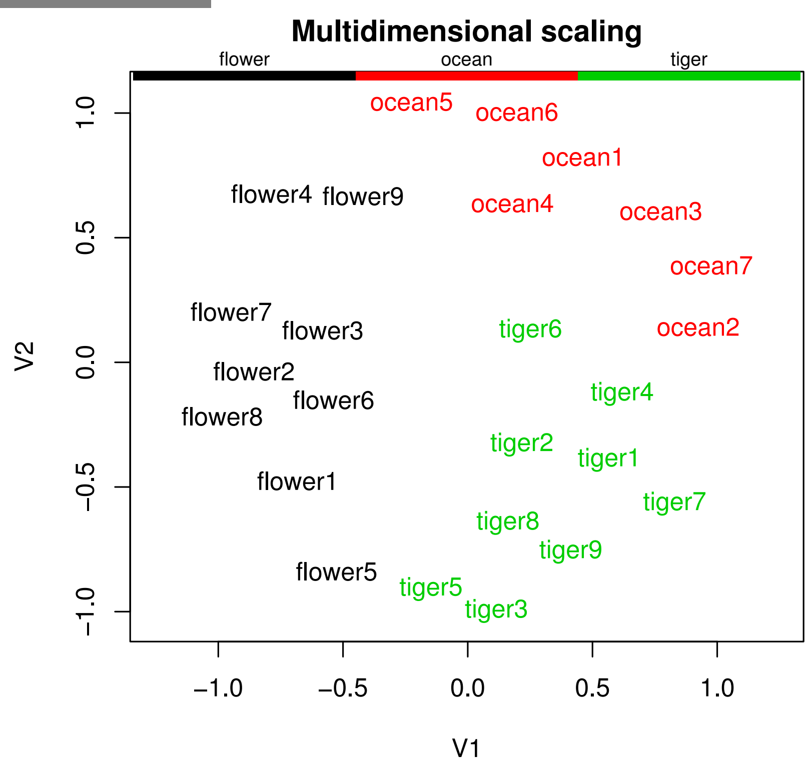
Inverse Picture Frequency—A method for emphasizing "important" colors by deemphasizing common colors. The count for color c in each image is multiplied by

$$IPF(c) = \log\left(\frac{1}{PF(c)}\right)$$

Colors which occur frequently, like a black border, will have a small $IPF(c)$.



Distance matrix
Normalized by Euclidean length
Lighter = Closer
No retrieval errors



Retrieval errors:

Normalization	Equal weight	IDF weight
None	8	4
Picture size	8	0
Euclidean length	7	0

Invariance is a useful principle for judging and designing distance measures. What is the set of changes to the object which do not affect the distance?

The first example of invariance was document length when matching articles. Some other invariances of the normalized bag-of-words representation:

- Punctuation, grammar, and word order
- Usage of common English words

Invariance to synonyms is also possible if you merge their counts, as we do for colors. All of these invariances make the representation good at matching articles on the same topic, even if they are written by different authors.

What invariances do we get for images?

- Position of objects, pose, small changes in camera angle
- Small amounts of zooming
- Small amounts of unusual colors
- Differences in texture (bad)

What invariances do these image representations have? What invariances do they lack?

1. Represent an image by something much smaller: its average pixel color.
2. Represent an image by something four times bigger: arbitrarily divide it into four parts, compute the color counts for each part, and stack them together.

References

[1] David Hand, Heikki Mannila, and Padhraic Smyth. *Principles of Data Mining*, Section 14.5. MIT Press, 2001.

[2] Online demo of Convera's Visual RetrievalWare. `http://vrw.convera.com:8015/cst`

[3] Content-based image retrieval demos. `http://www.cs.washington.edu/research/imagedatabase/demo/cbir.html`