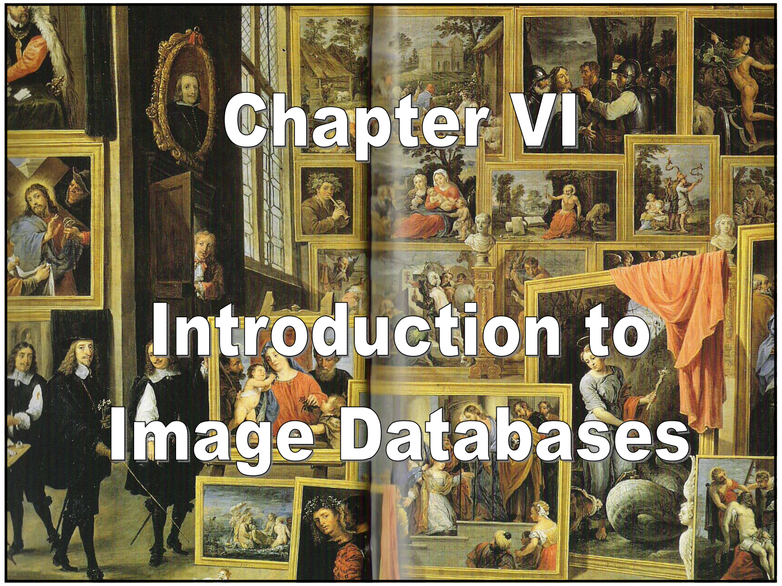
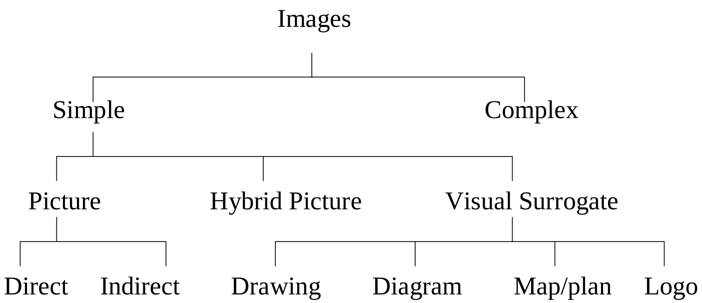


Chapter VI

Introduction to Image Databases

Different types of images

Taxonomy of fixed images



Introduction to image databases

- 6.1 – Generalities
- 6.2 – Retrieval based on key-words
- 6.3 – Retrieval based on content
- 6.4 – Physical structuring of image databases
- 6.5 – Example: Image Rover
- 6.6 – Conclusions

6.1 – Generalities

- Several types
  - Collections of fixed images
  - Very big rasters
    - Raster maps
    - Aerial photos
    - Satellite images
  - Video Sequences, films

Retrieval

- Based on key-words (typically a user gives a list of key-words)
- Based on content (user gives an image-example (query by example)
  - colors
  - shapes
  - textures
  - spatial relations

High and low level characteristics

- High Level Characteristics
  - nature and semantics of objects present on image, and their relations
  - example : a boat on the sea at sunset
- Low Level Characteristics
  - pixels, colors, textures, etc.
  - example : images with 40 % of light blue.

Interrogation of image bases

- Retrieval based on key-words
- Retrieval based on content
- Retrieval based on colors
- Retrieval based on shapes

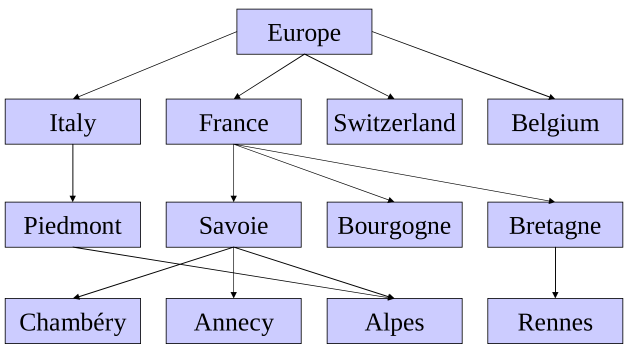
6.2 – Retrieval based on key-words

- Each image is described by a list of key-words (operation named indexing)
- Generally from 3 to 10 key-words given by the author or by an expert
- Those key-words are regrouped in a thesaurus with 3 relations
  - synonymy
  - genericity / specificity

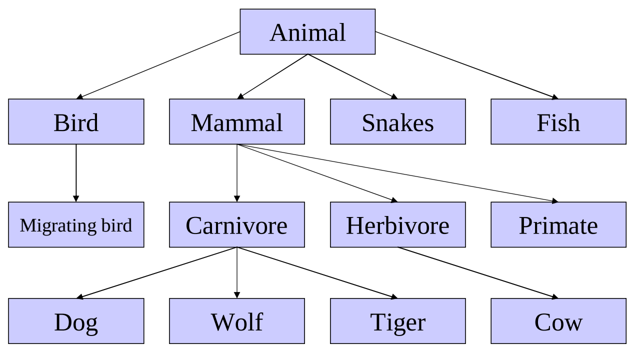
Annotations

- Additional Information
- Retrieval of key-words characterizing MM documents
- Lists of objects, persons, etc.
- Origin of the document (author, device, date, etc.)
- Generally made visually
- Seldom named “manual indexing"

Example of a geographic thesaurus



Other example of thesaurus

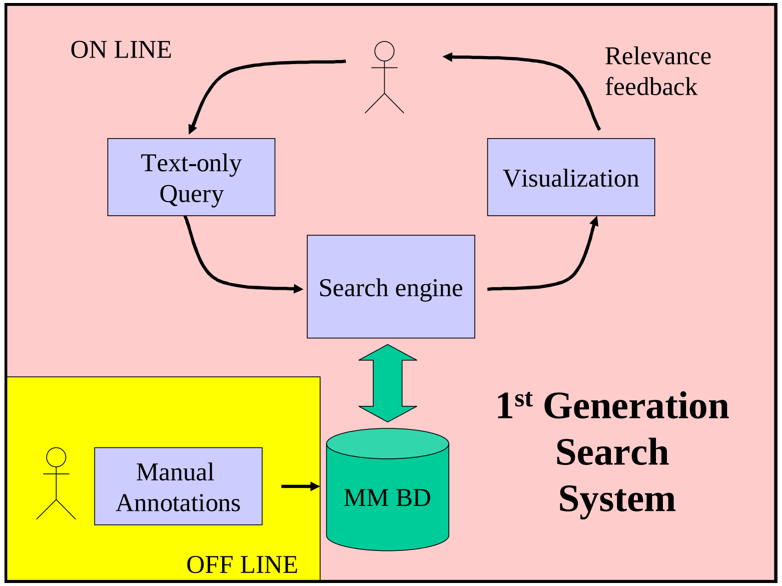


Queries

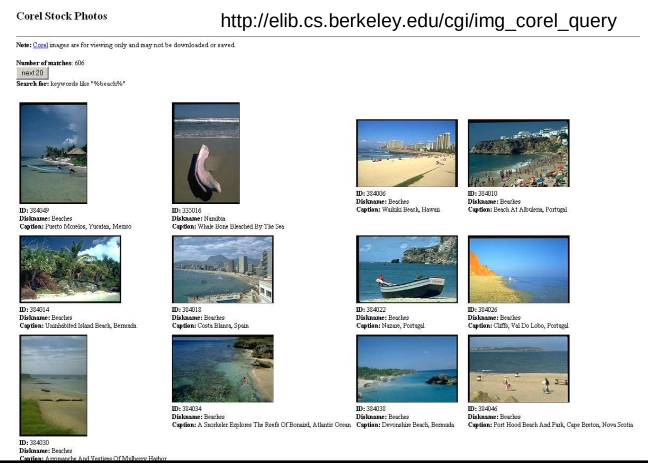
- Boolean list of key-words (AND, OR, EXCEPT)
- Noise = too much non relevant documents
- Silence = too less or absence results
- Example : “*I looking for photos dealing with the culture of cauliflowers in Australia*”

Reformulation

- R : `culture and cauliflower and Australia`
- Transformation of the query based on thesaurus
- R’: `(culture except civilization)and cauliflower or vegetable)and(Australia or Oceania)`



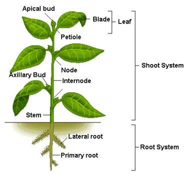
Keyword: Beach



Plant database

Linnaean Taxonomic Hierarchy

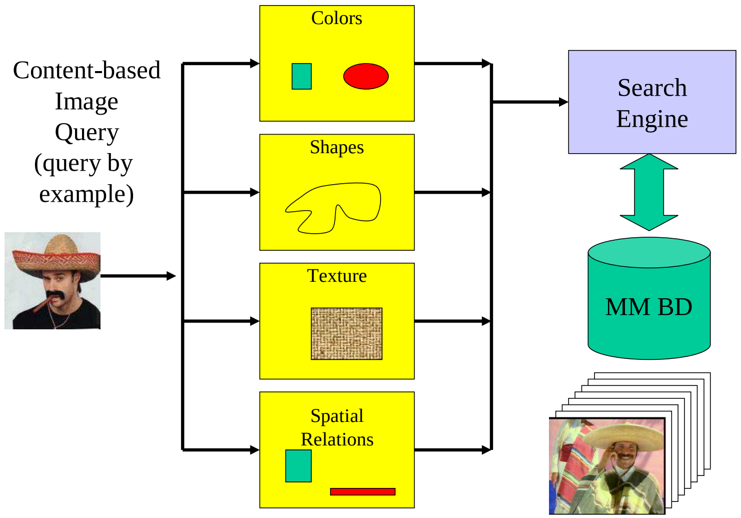
CATEGORY	
1. Kingdom	*Plantae*
2. Division	*Angiosperms*
3. Class	*Dicotyledons*
4. Order	*Asterales*
5. Family	*Asteraceae*
6. Genus	*Helianthus*
7. Species	*Helianthus annus*

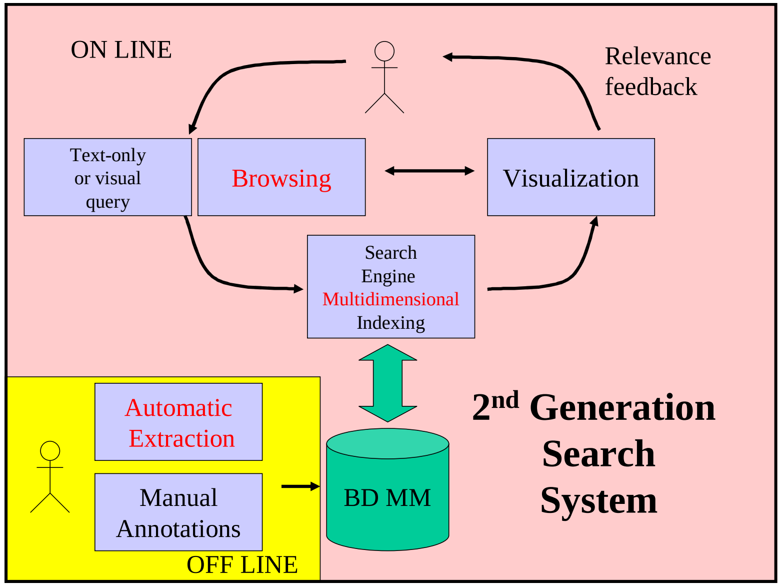


http://www.ualr.edu/~botany/images.html

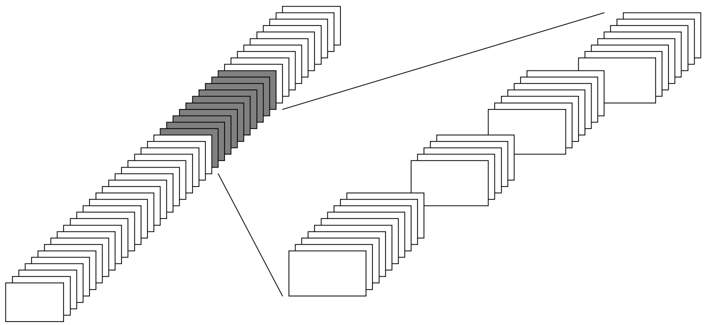
6.3 – Retrieval based on content

- Generally based on an example "*Query-by-example*"



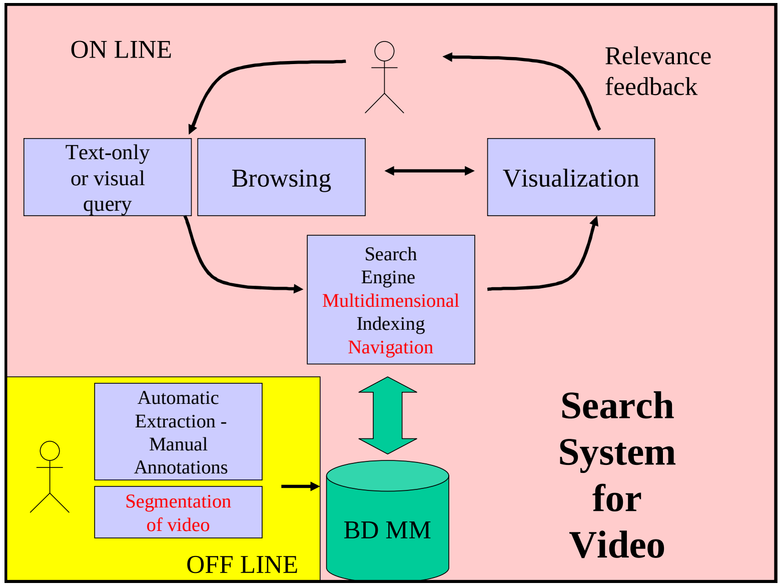


Structuring a video flow



Segmentation

- Manual vs. automatic Segmentation
- Splitting in
  - episodes
  - scenes
  - shots
- Semantics (same actor, same place, etc.)



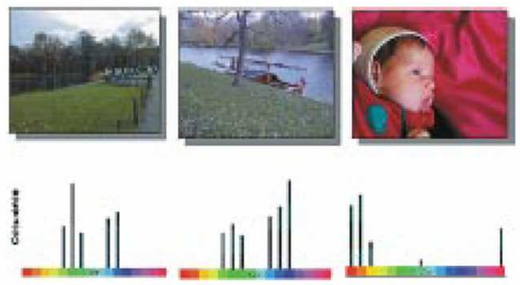
Content-based Retrieval

- 6.3.1 – Retrieval based on colors
- 6.3.2 – Retrieval based on shape
- 6.3.3 – Retrieval based on parameters
- 6.3.4 – Retrieval based on spatial relations

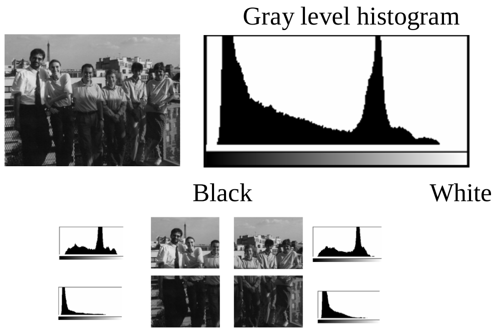
6.3.1 – Retrieval based on colors

- Containing a color in a certain proportion
- Similarity of colors in the whole image
- Similarity of only a part
- Based on an object having some particular color
- etc.

Example of histograms



Histograms of regions



Solution

- Histogram of colors for the image-query $H(I_Q)$ and for other images $H(I_D)$
- Attention: same resolution, same color-encoding system

Comparison of histograms

$$D_H(I_Q, I_D) = \sum_{j=1}^{n} \left| H(I_Q, j) - H(I_D, j) \right|$$

$$D_H(I_Q, I_D) = \sqrt{\sum_{j=1}^{n} (H(I_Q, j) - H(I_D, j))^2}$$

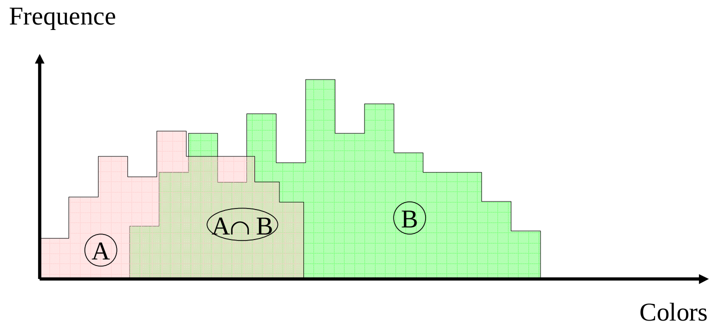
- Comparison by test (χ^2 type)
- But weak performances

Method of Swain and Ballard

- Intersection of histograms

$$D_H(I_Q, I_D) = \frac{\sum_{j=1}^{n} \min(H(I_Q, j), H(I_D, j))}{\sum_{j=1}^{n} H(I_Q, j)}$$

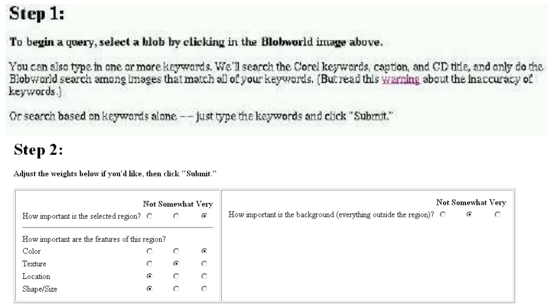
Intersection of histograms

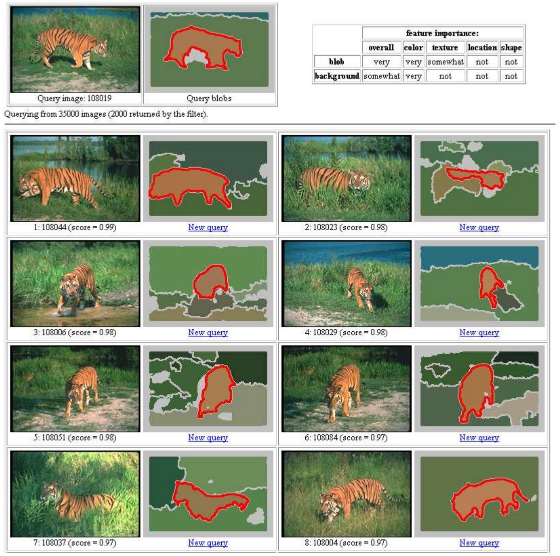


Blobworld

http://elib.cs.berkeley.edu/photos/blobworld/

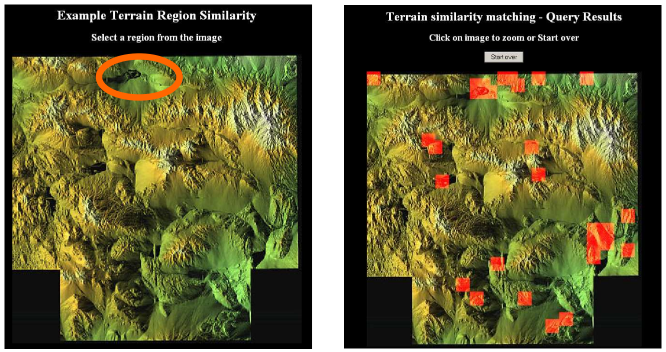
- University of California, Berkeley
- 35 000 images





MARS http://www-db.ics.uci.edu/pages/research/mars.shtml

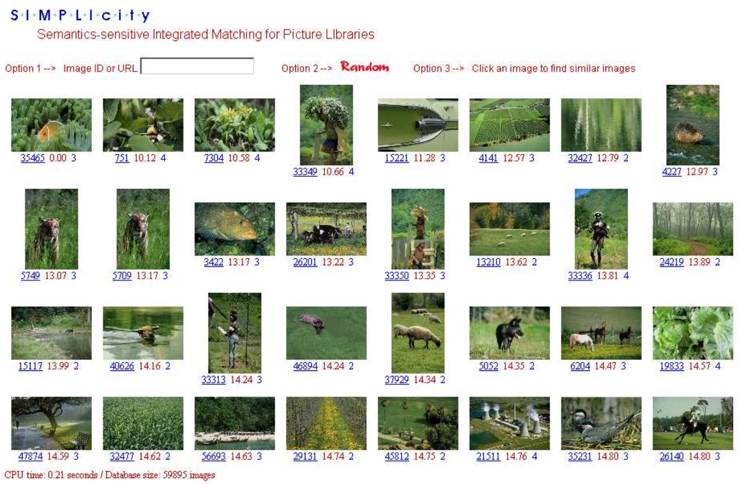
- MARS (Multimedia Analysis and Retrieval System)
- University of Illinois at Urbana-Champaign, University of California at Irvine,
- Demos http://www-db.ics.uci.edu/pages/demos/index.shtml.



Simplicity

http://wang16.ist.psu.edu/cgi-bin/zwang/regionsearch_show.cgi

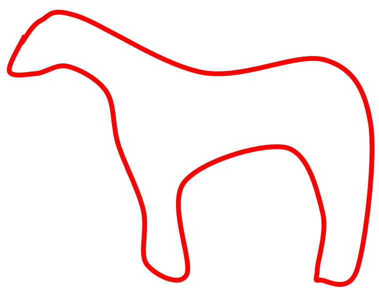
- Stanford university
- 200 000 images
- Similarity



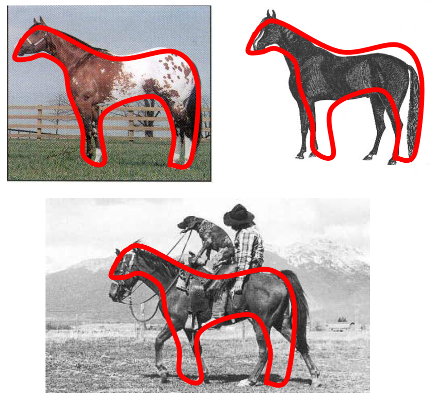
6.3.2 – Retrieval based on shape

- The user gives, for instance gives a shape manually draught (template)
- Comparison with other images
- Linear deformations
  - translations
  - rotations
  - scaling
- Other transformations (warping)

Example “horse”



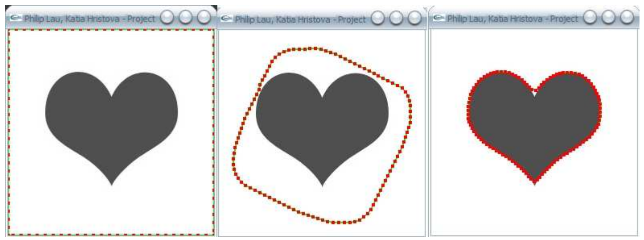
Example of results



Snakes – Active Contours

- Original shape is deformed to reach gradually some other shape
- Objectives
  - To follow edges the best possible
  - To minimize deformation energy

Example



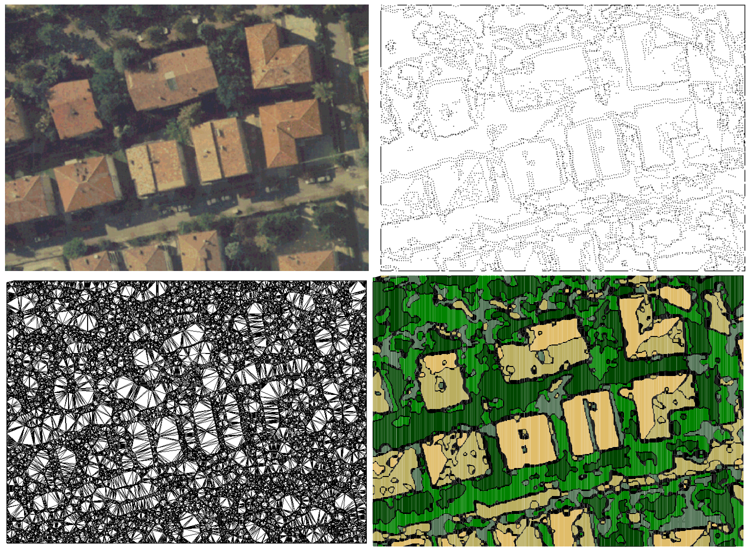
Shape deformation

- Intrinsic equation of the curve

$$\vec{\phi}(s) = \vec{\tau}(s)\vec{\theta}(s)$$

- Measure of deformation

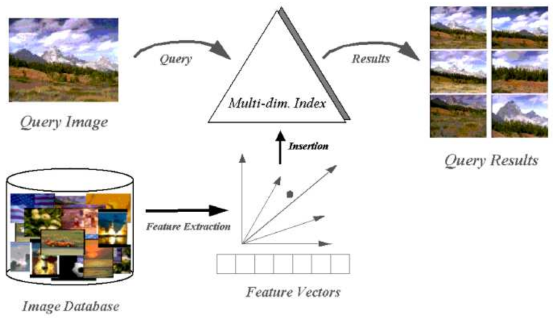
$$M = \int_0^1 \left[\nabla I\left(\vec{\phi}(s)\right)\right]^2 ds$$



6.3.3 – Retrieval based on parameters

- Principle :
  - each image is described by a set of parameters issued from image processing
  - until several dozens of parameters
    - ➔index (feature extraction)
  - the image-query is analyzed to evaluate those parameters
  - comparison (distance)

Structure



Usual features

$$f_1 = \text{Energy} = \sum_{i,j} \hat{f}(i,j)^2 \;,$$

$$f_2 = \text{Entropy} = \sum_{i,j} \hat{f}(i,j) \log \hat{f}(i,j) \;,$$

$$f_3 = \text{Correlation} = \sum_{i,j} \frac{(i-\mu_i)(j-\mu_j)\hat{f}(i,j)}{\sigma_i \sigma_j} \;,$$

$$f_4 = \text{Inverse Difference Moment} = \sum_{i,j} \frac{1}{1+(i-j)^2} \hat{f}(i,j) \;,$$

$$f_5 = \text{Inertia} = \sum_{i,j} (i-j)^2 \hat{f}(i,j) \;,$$

$$f_6 = \text{Cluster Shade} = \sum_{i,j} ((i-\mu_i)-(j-\mu_j))^3 \hat{f}(i,j) \;,$$

$$f_7 = \text{Cluster Prominence} = \sum_{i,j} ((i-\mu_i)-(j-\mu_j))^4 \hat{f}(i,j) \;,$$

$$\text{where } \mu_i = \sum_i i \sum_j \hat{f}(i,j) \;,$$

$$\mu_j = \sum_j j \sum_i \hat{f}(i,j) \;,$$

$$\sigma_i = \sum_i (i-\mu_i)^2 \sum_j \hat{f}(i,j) \;,$$

$$\sigma_j = \sum_j (j-\mu_j)^2 \sum_i \hat{f}(i,j) \;,$$

$$f_8 = \text{Haralick's Correlation} = \frac{\sum_{i,j}(ij)\hat{f}(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y}$$

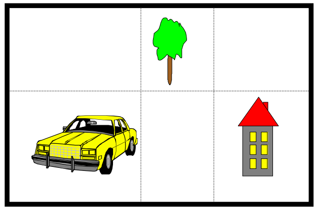
6.3.4 – Retrieval based on spatial relations

- Symbolic Projections (2D strings)
- Using Egenhofer relations
- Using Jungert operators
- Signature

Chang 2D strings

- Principle :

x-projection : car < tree < house
y-projection : car and house < tree



Description of an image

d		
	b	*c*
a	*a*	

- *I = (u, v)*
- *(a=d<a=b<c , a=a<b=c<d)*
- = same place
- < left-right
- normal 2D strings : *(ad<ab<c , aa<bc<d)*

Pattern matching

Initial Image

d		
	b	*c*
a		

Example of queries :

	b
a	

	c
a	

b	*c*
a	

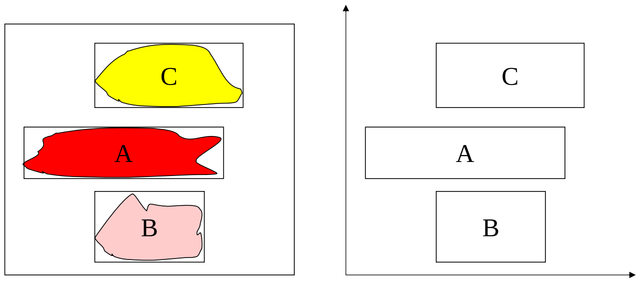
Conclusion about 2 D strings

- Interesting tool
- Difficulties of description when overlaps
- Problems of zones with holes (non connected objects)
- ➔ other operators

Encoding with 2D B strings

$$u : A_{bx} < B_{bx} = C_{bx} < Be_x < Ae_x < Ce_x$$

$$v : B_{by} < Be_y = A_{by} < Ae_y = C_{by} < Ce_y$$



Jungert operators (1/2)

A<B	center(A) < center(B)	A B
A=B	center(A) = center(B)	AB
A|B	Side by side	A B
A%B	Min(A) > Min (B) Max(A) < Max (B) Length(A) < Length(B)	A B

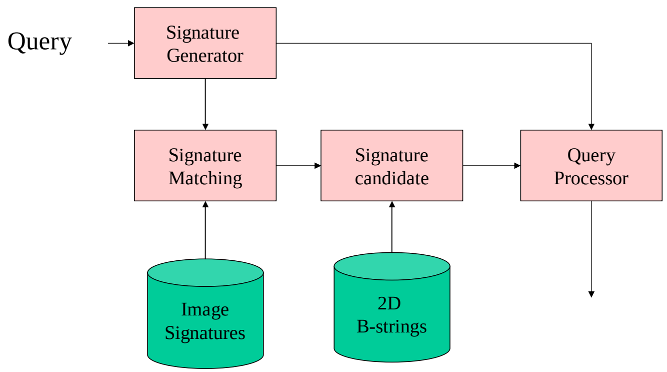
Jungert operators (2/2)

A[B	Min(A) = Min (B) Length(A) < Length(B)	A B
A]B	Max(A) = Max (B) Length(A) < Length(B)	A B
A\B	Min(A) < Min (B) Length(A) ≤ Length(B)	A B
A/B	Max(A) > Max (B) Length(A) ≤ Length(B)	A B

Signature

- 1 – Encoding an image with 2D strings or more exactly with 2D B strings (Lee, Yang, Chen) (= part encodage)
- 2 – Hashing Function

Retrieval by signature



6.4 – Physical Structuring of Image Databases

- Image
  - A set of encoded pixels
  - A set of descriptors (= parameters)
  - Possibly some recognized pictorial objects
- Image Databases
  - Zillions of images (usually same format)
  - An access system based on descriptors
- Big raster
  - A unique very big image (several billions of pixels)
  - Stored position of some pictorial objects

Descriptors

- Descriptors: any characteristics of an image
  - Low level (before interpretation)
  - High level (after interpretation); classifiers
- Several dozens, possibly 100+
- ➔1 point in a *n*-dimension space
- Examples
  - Colors: encoding, histogram, etc.
  - Texture: granularity, direction, repetitiveness, etc.

Query

- Let be a set of descriptors
- Let be a query-image
  - Analysis to evaluate descriptors
- Relevance feedback
  - Several query-images

Relevance feedback

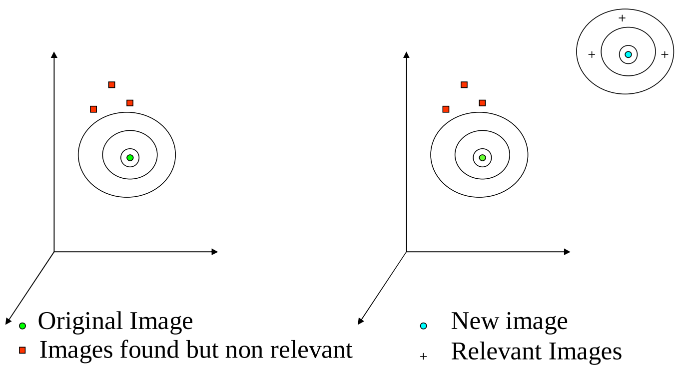


Image query ➔ spatial query

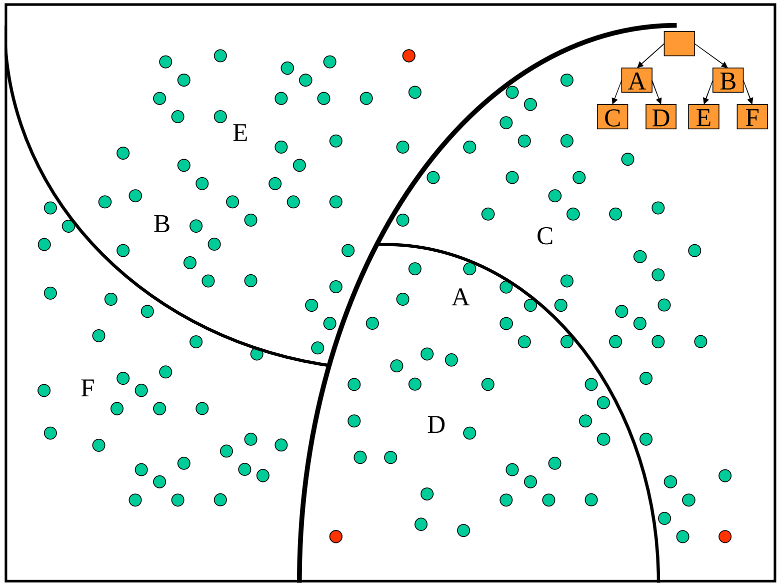
- Point query (*n*-dimension space)
- Research of *k*-nearest neighbors
- ➔ Multipoint query

Physical Access Method

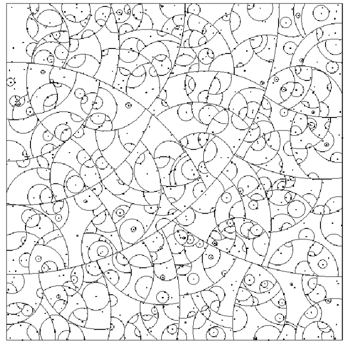
- Descriptors ➔ *n*-dimension space
- Distances
  - Euclidian
  - etc.
- Algorithms: Research of *k*-nearest neighbors
- Solutions :
  - R-tree or R^+-tree
  - VP-tree (Vantage Point Tree)

VP-tree

- Retrieval of neighbors
- ➔ distance between points (*n*-dimension)
- We start from a point (named Vantage point), which will be the center of a "circle" so that 50 % of the points will belong to this "circle"
- Then we continue in a tree-like manner
- Vantage point :
  - barycenter
  - The farthest point



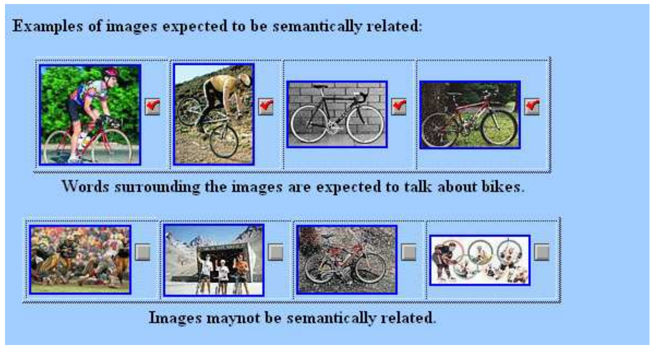
Example



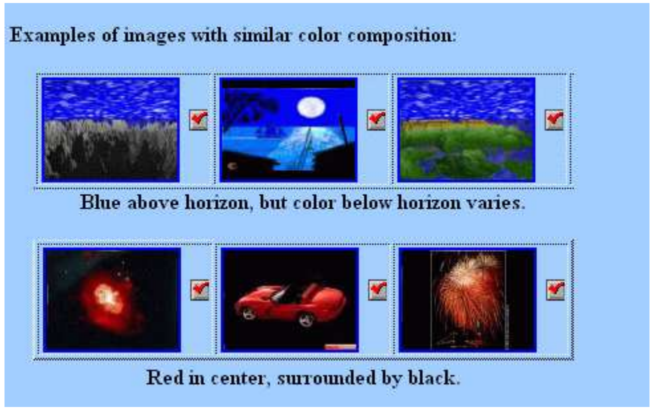
6.5 – Example: Image Rover

- Boston University, MA.
- Demo http://www.cs.bu.edu/groups/ivc/ImageRover/demo.html.
- Semantic Associations
- Color Associations
- Orientation Associations

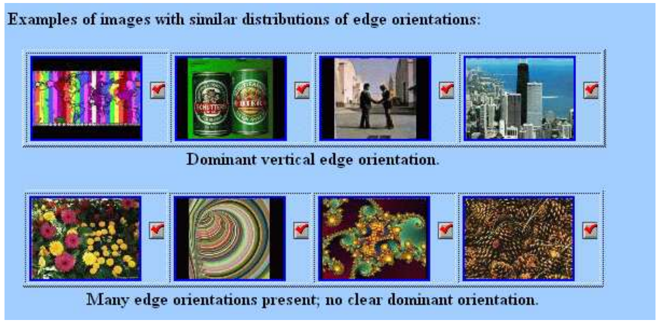
Semantic Associations



Color Associations



Orientation Associations



6.6 – Conclusion

- Difficulty of retrieving images based on contents
  - Passing from pixels to semantic objects
- Low and high level descriptors
- Dedicated search engines
- Often disappointing results
- Using semantic networks and ontologies

Demos

- http://wang.ist.psu.edu/IMAGE/
- http://www.terraserver.microsoft.com/default.aspx
- http://earth.google.com