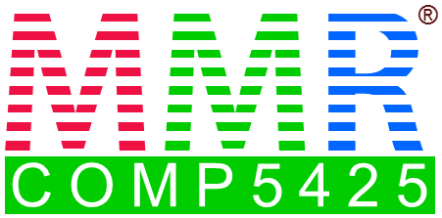


MULTIMEDIA RETRIEVAL

THE UNIVERSITY OF SYDNEY

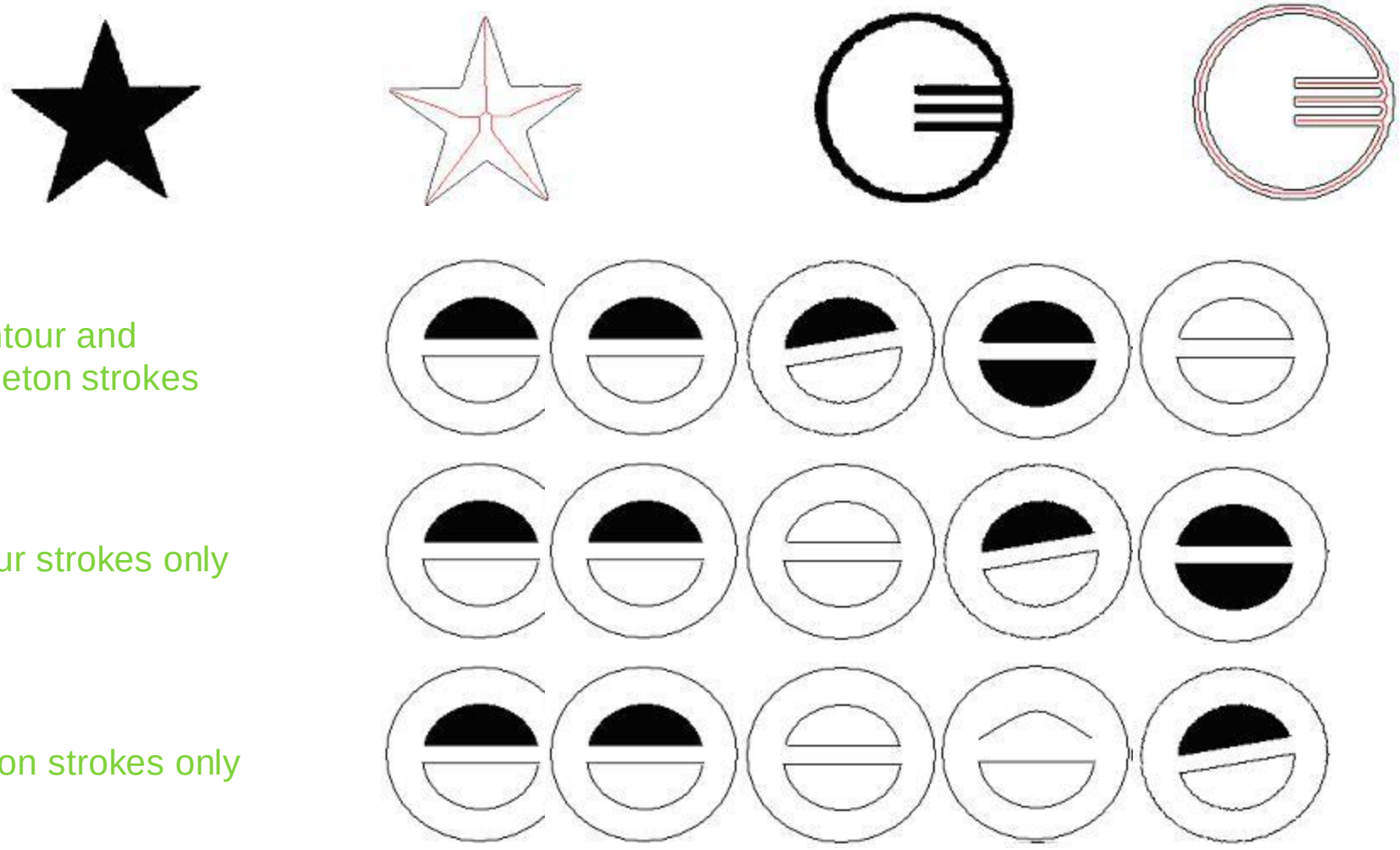
Week06

Semester 1, 2025

Content Based Retrieval II

- Visual feature extraction
  - Shape
- Indexing
- Presentation & Evaluation

Trademark Retrieval System

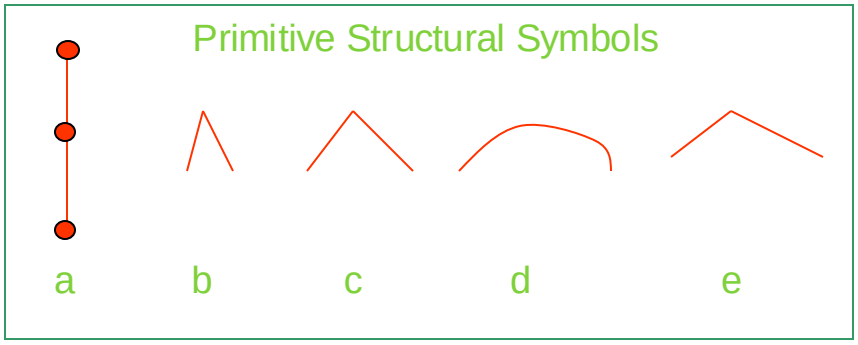


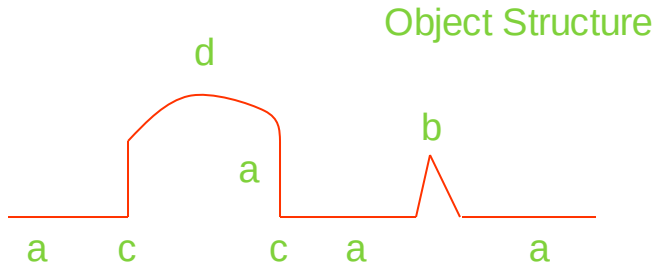
Contour and skeleton strokes

Contour strokes only

Skeleton strokes only

Syntactic Representation

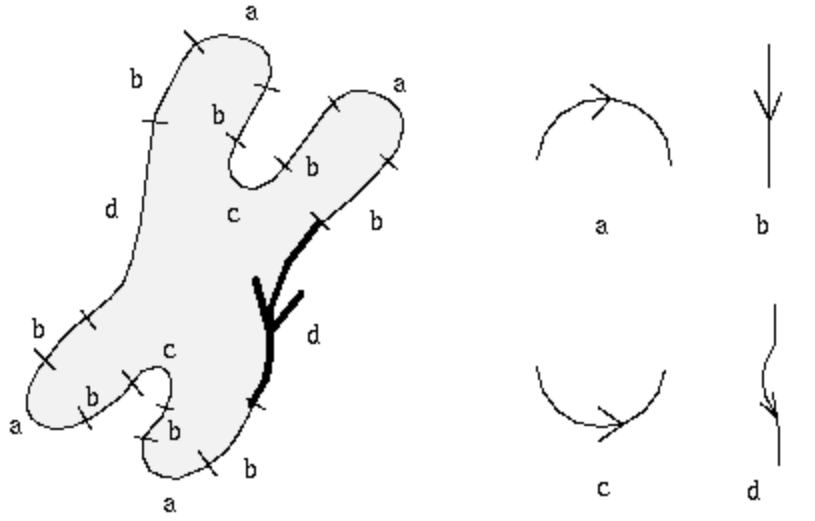




Syntactic Representation: a c d e a c a b a

- We can reduce an object to a set of structural elements, or primitives.
- By adding a syntax, such as connectivity rules, it is possible to obtain a syntactic representation, which is simply a string of symbols, each representing a primitive. The syntax allows a unique representation and interpretation of the string.
- The design of a syntax that transforms the symbolic and the syntactic representations back and forth is a difficult task. It requires specification of a complete and unambiguous set of rules, which have to be derived from the understanding of the scene under study.

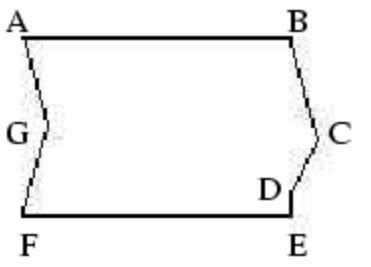
Syntactic Representation



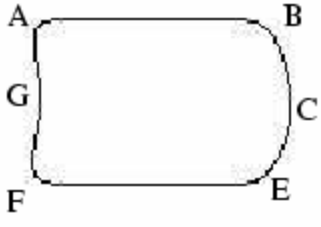
Structural description of chromosomes by a chain of boundary segments, code word: d, b, a, b, c, b, a, b, d, b, a, b, c, b, a, b (adapted from [Fu 74]).

Need making use of domain knowledge

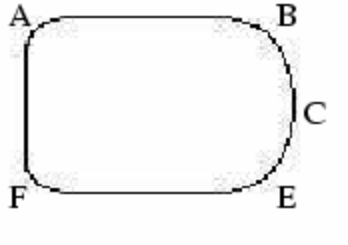
Curvature Scale Space



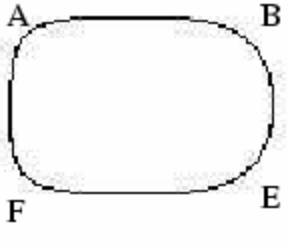
(a)



(b)δ=10



(c)δ=15



(d)δ=20

$$X(t,\sigma) = x(t) * g(t,\sigma) = \int_{-\infty}^{t} x(s)g(t-s,\sigma)ds$$

$$Y(t,\sigma) = y(t) * g(t,\sigma) = \int_{-\infty}^{t} y(s)g(t-s,\sigma)ds$$

$$g(t,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{t^2}{2\sigma^2}}$$

Curvature Scale Space (CSS)

Representations Based on CSS

- CSS image
- Interval tree
- Segment tree

$$K(t,\sigma) = \frac{X_t(t,\sigma)Y_{tt}(t,\sigma) - X_{tt}(t,\sigma)Y_t(t,\sigma)}{(X_t^2(t,\sigma) + Y_t^2(t,\sigma))^{\frac{3}{2}}}$$

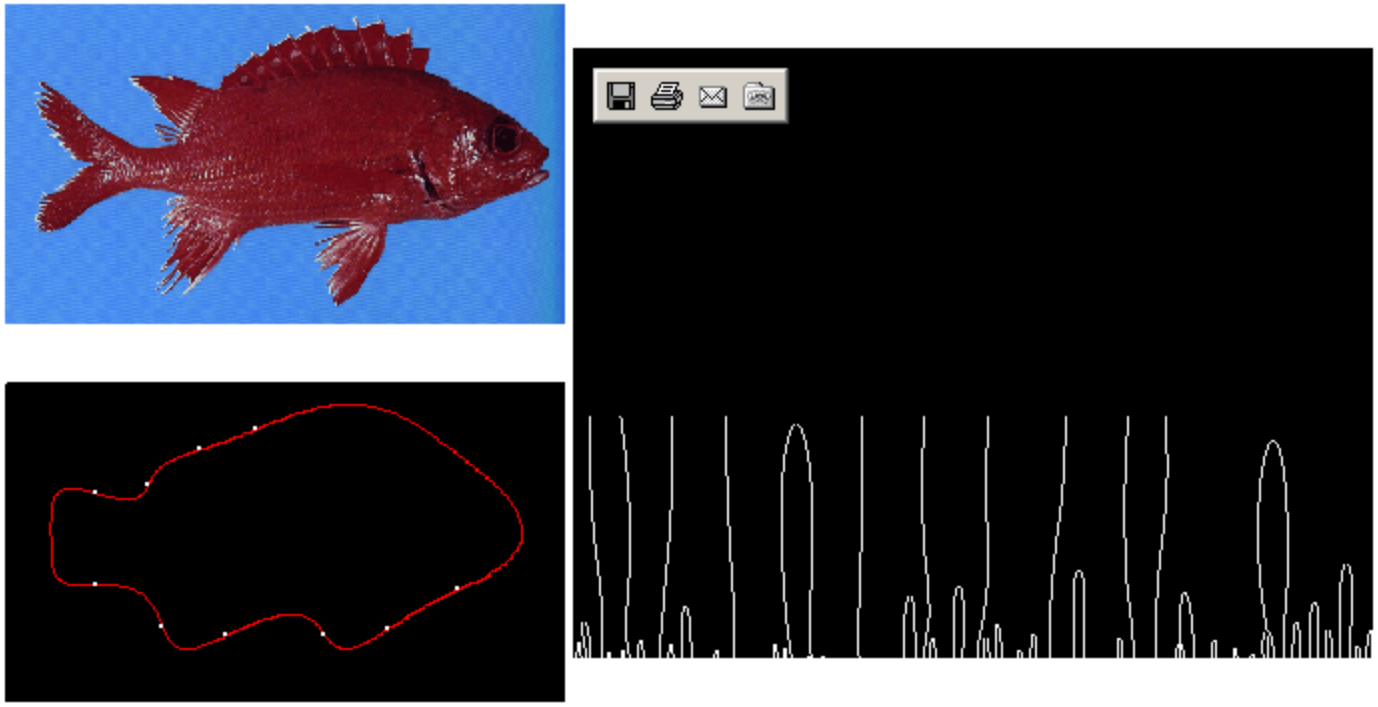
$$X_t(t,\sigma) = \frac{\partial}{\partial t}(x(t) * g(t,\sigma)) = x(t) * g_t(t,\sigma)$$

$$X_{tt}(t,\sigma) = \frac{\partial^2}{\partial t^2}(x(t) * g(t,\sigma) = x(t) * g_{tt}(t,\sigma)$$

$$Y_t(t,\sigma) = \frac{\partial}{\partial t}(y(t) * g(t,\sigma)) = y(t) * g_t(t,\sigma)$$

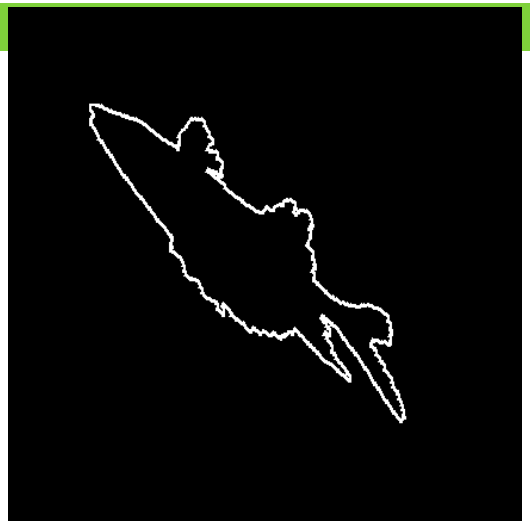
$$Y_{tt}(t,\sigma) = \frac{\partial^2}{\partial t^2}(y(t) * g(t,\sigma) = y(t) * g_{tt}(t,\sigma)$$

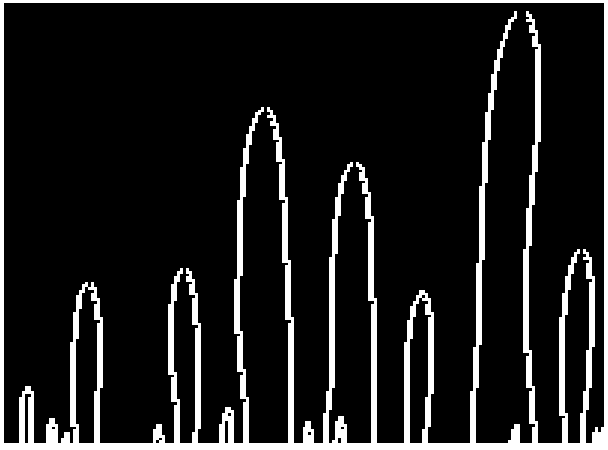
CSS Image

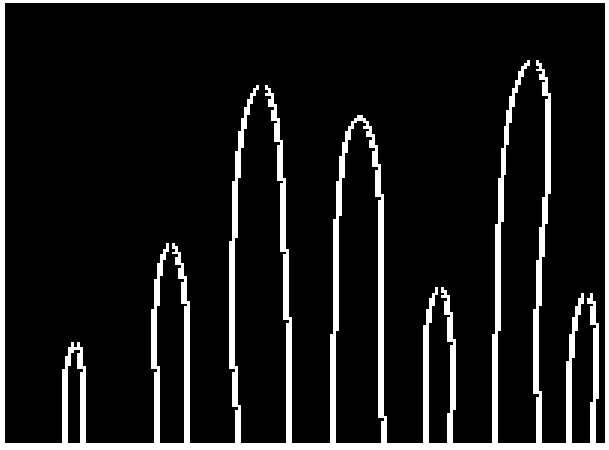


http://www.ee.surrey.ac.uk/Research/VSSP/imagedb/demo.html

CSS Image

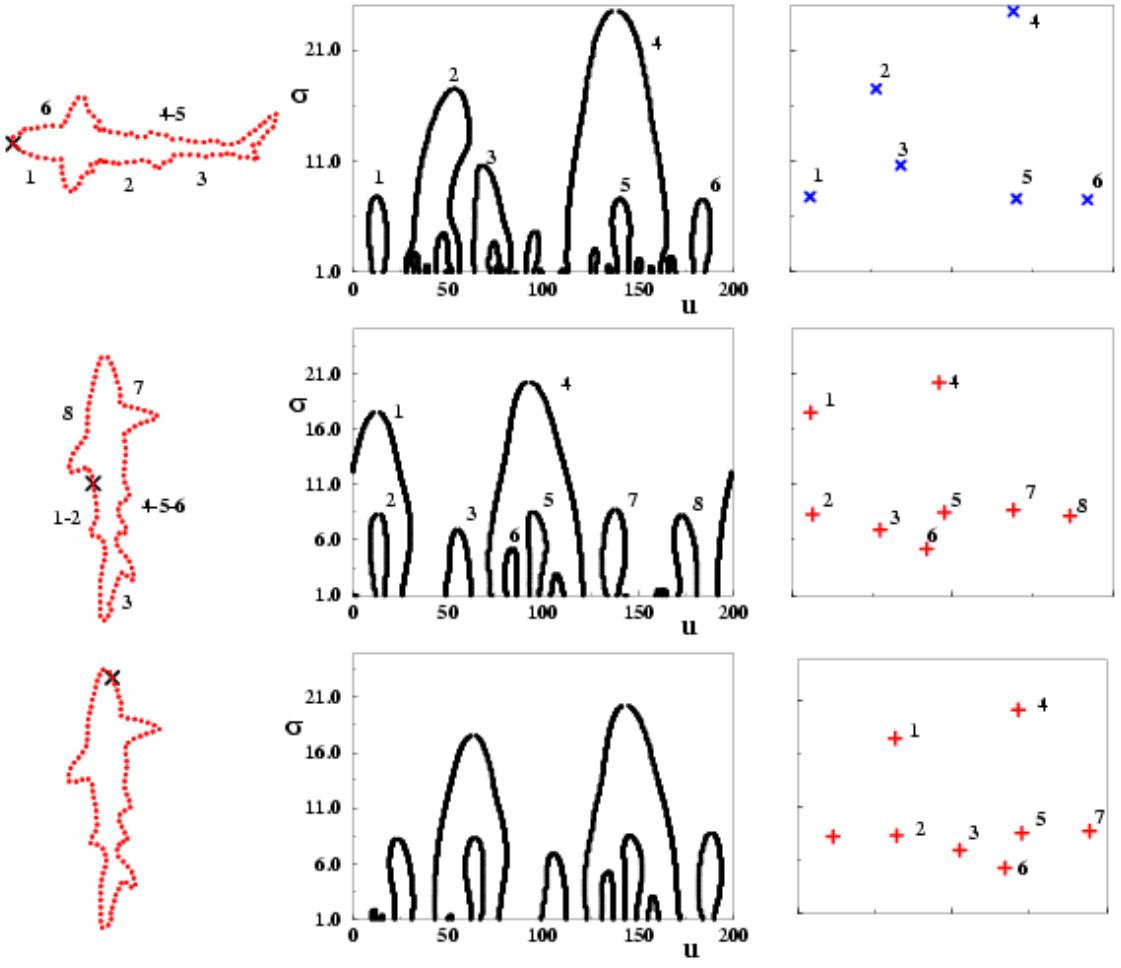




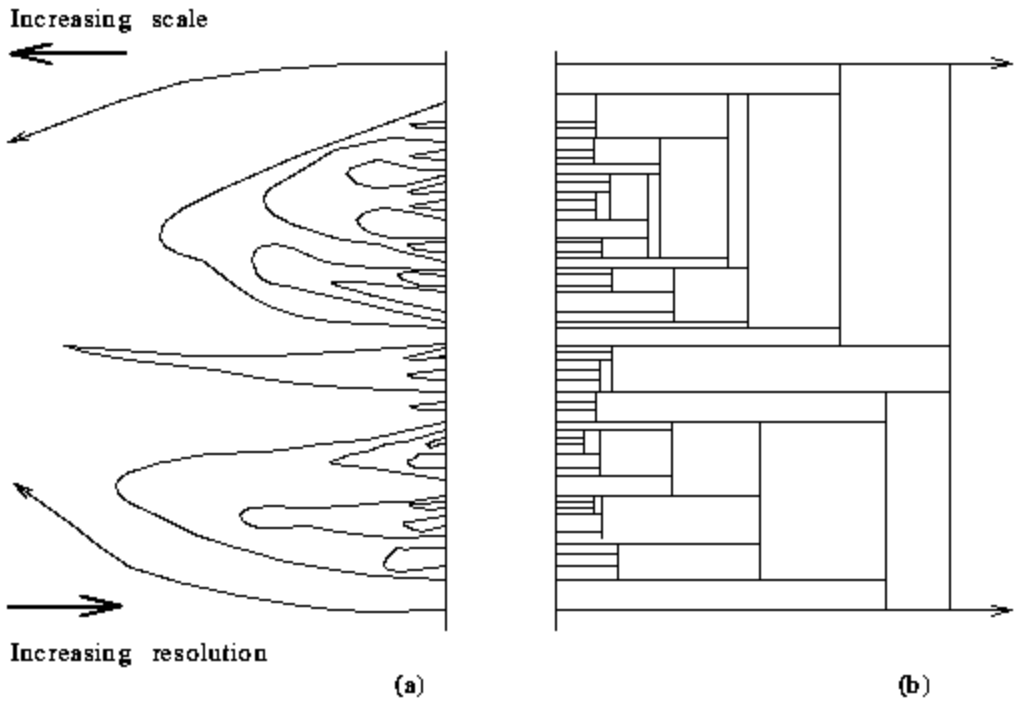


Invariant to affine transformation

CSS Image Matching

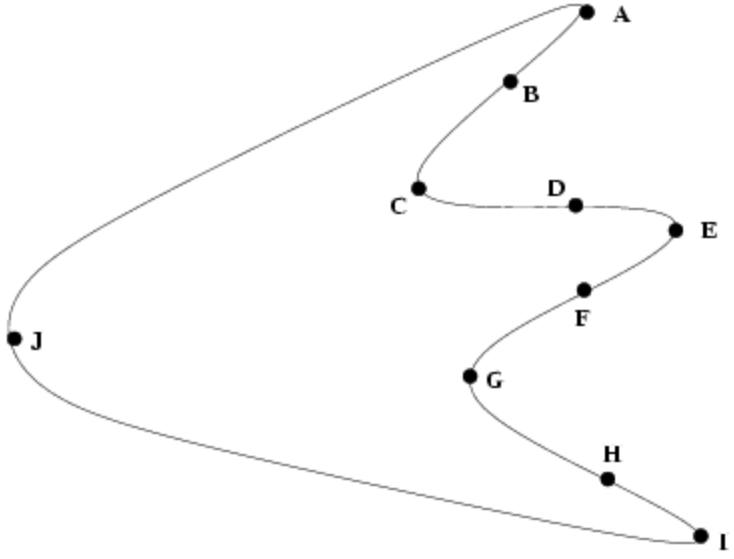


Interval Tree



Scale-space image: (a) Varying number and locations of curve segmentation points as a function of scale, (b) curve representation by an interval tree.

Segment Tree



$$K(t,\sigma) = \frac{X_t(t,\sigma)Y_{tt}(t,\sigma) - X_{tt}(t,\sigma)Y_t(t,\sigma)}{(X_t^2(t,\sigma) + Y_t^2(t,\sigma))^{\frac{3}{2}}}$$

$$X_t(t,\sigma) = \frac{\partial}{\partial t}(x(t) * g(t,\sigma)) = x(t) * g_t(t,\sigma)$$

$$X_{tt}(t,\sigma) = \frac{\partial^2}{\partial t^2}(x(t) * g(t,\sigma) = x(t) * g_{tt}(t,\sigma)$$

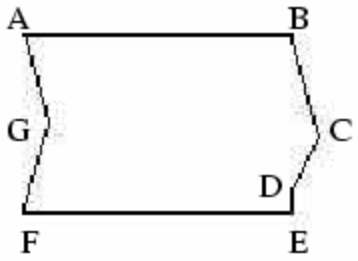
$$Y_t(t,\sigma) = \frac{\partial}{\partial t}(y(t) * g(t,\sigma)) = y(t) * g_t(t,\sigma)$$

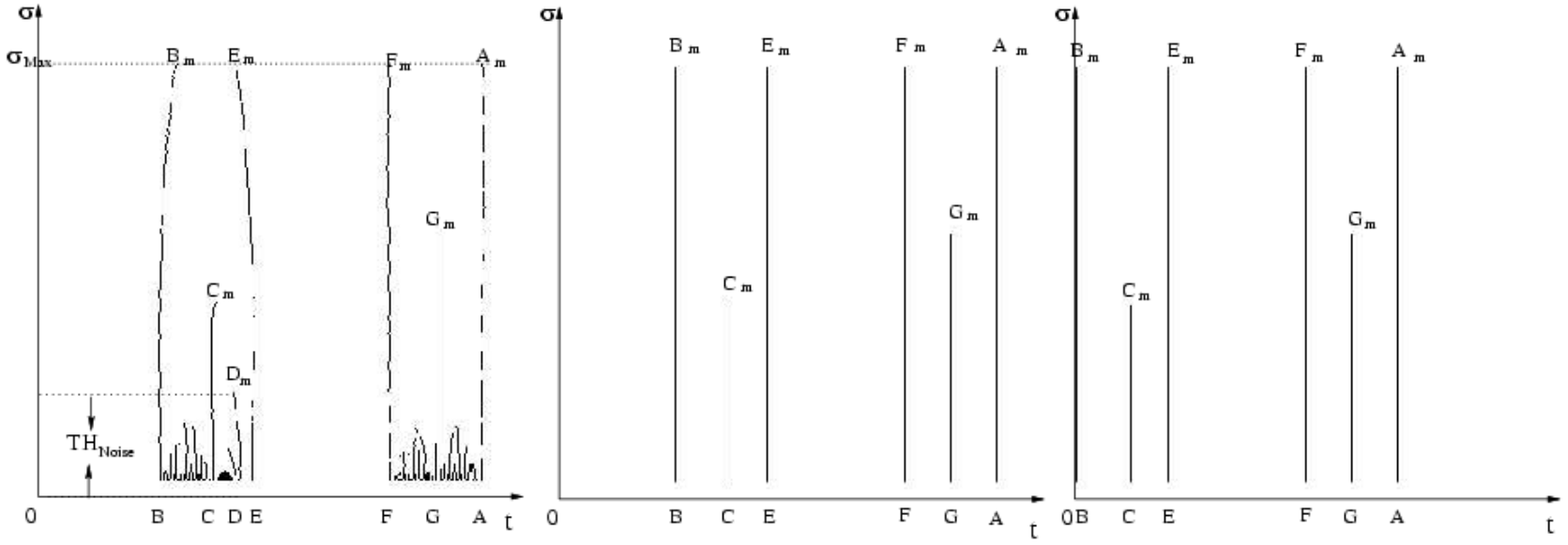
$$Y_{tt}(t,\sigma) = \frac{\partial^2}{\partial t^2}(y(t) * g(t,\sigma) = y(t) * g_{tt}(t,\sigma)$$

$$|K(t_0,\sigma)| = \underset{t_i \in T}{Max}\{|K(t_i,\sigma)|\} \qquad \frac{\partial K(t,\sigma)}{\partial t}\Big|_{t=t_0} = 0$$

Segment Tree

$$|K(t_0,\sigma)| = \underset{t_i \in T}{Max}\{|K(t_i,\sigma)|\} \qquad \left.\frac{\partial K(t,\sigma)}{\partial t}\right|_{t=t_0} = 0$$



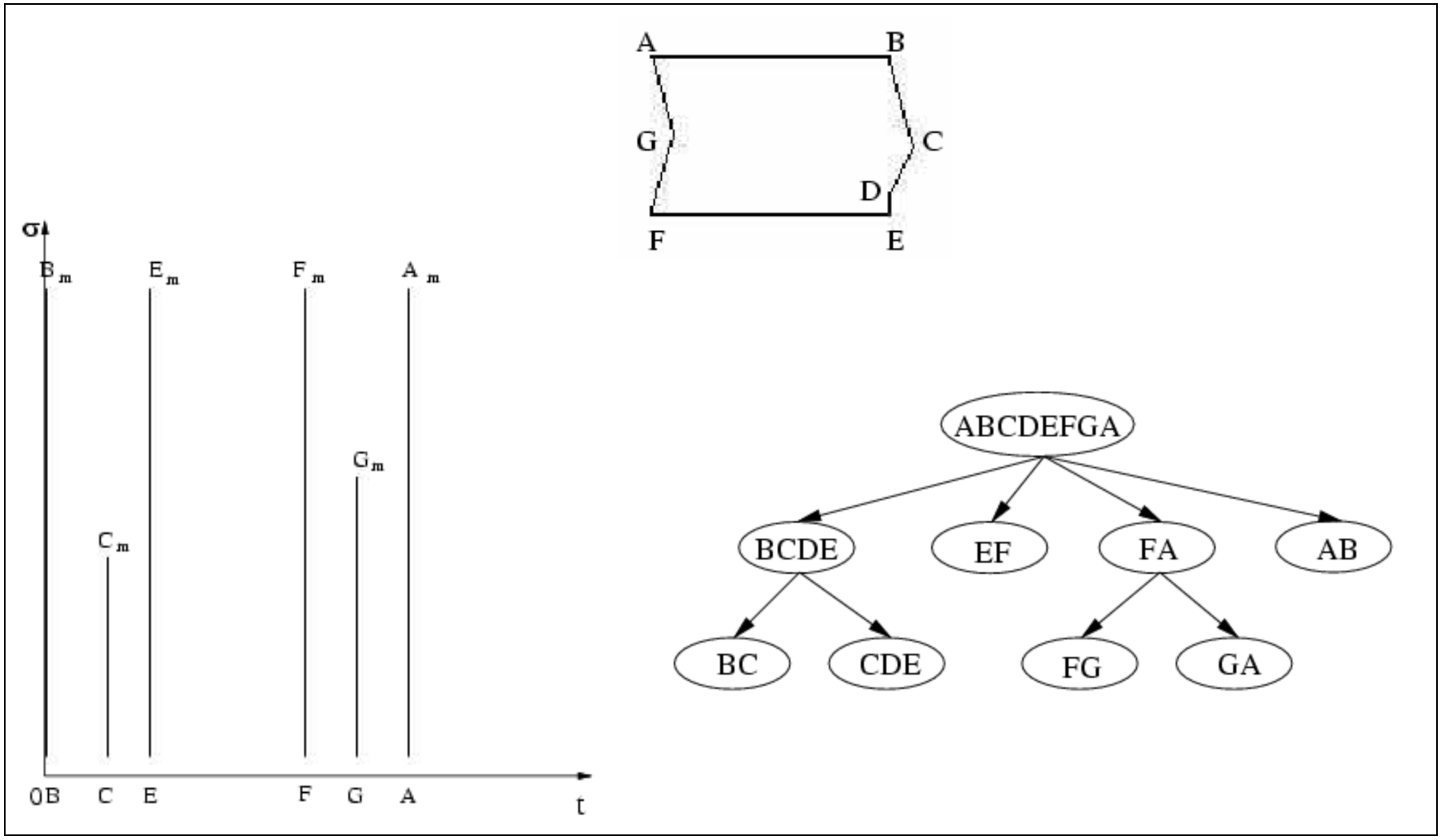


Original CSS Image

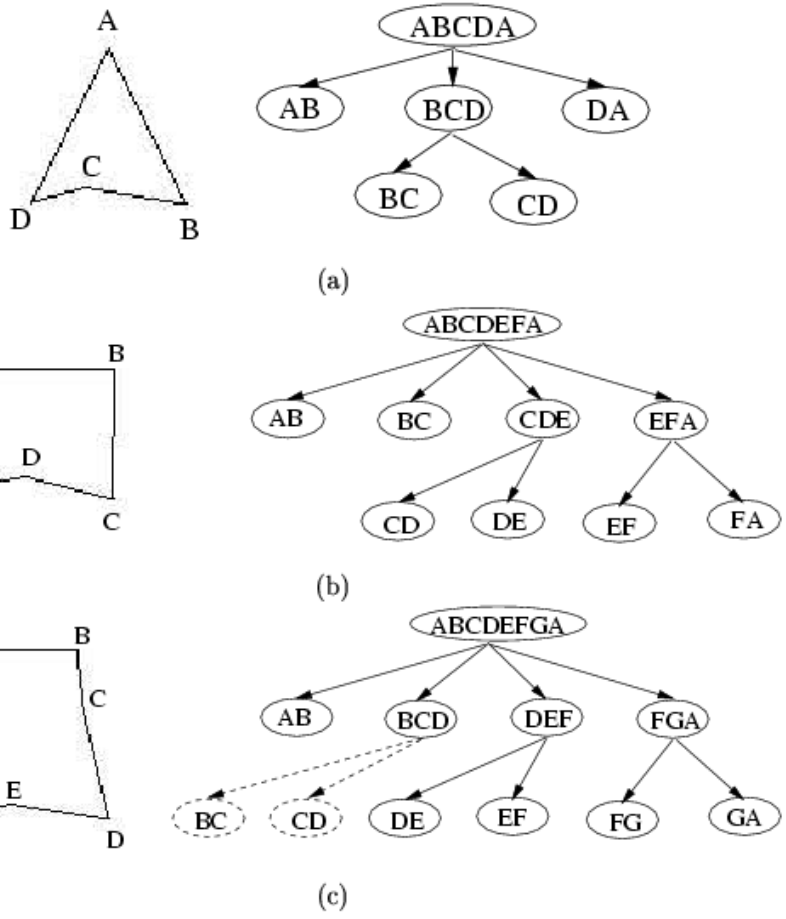
Traced CSS Image

Aligned CSS Image

Segment Tree



Segment Tree Samples



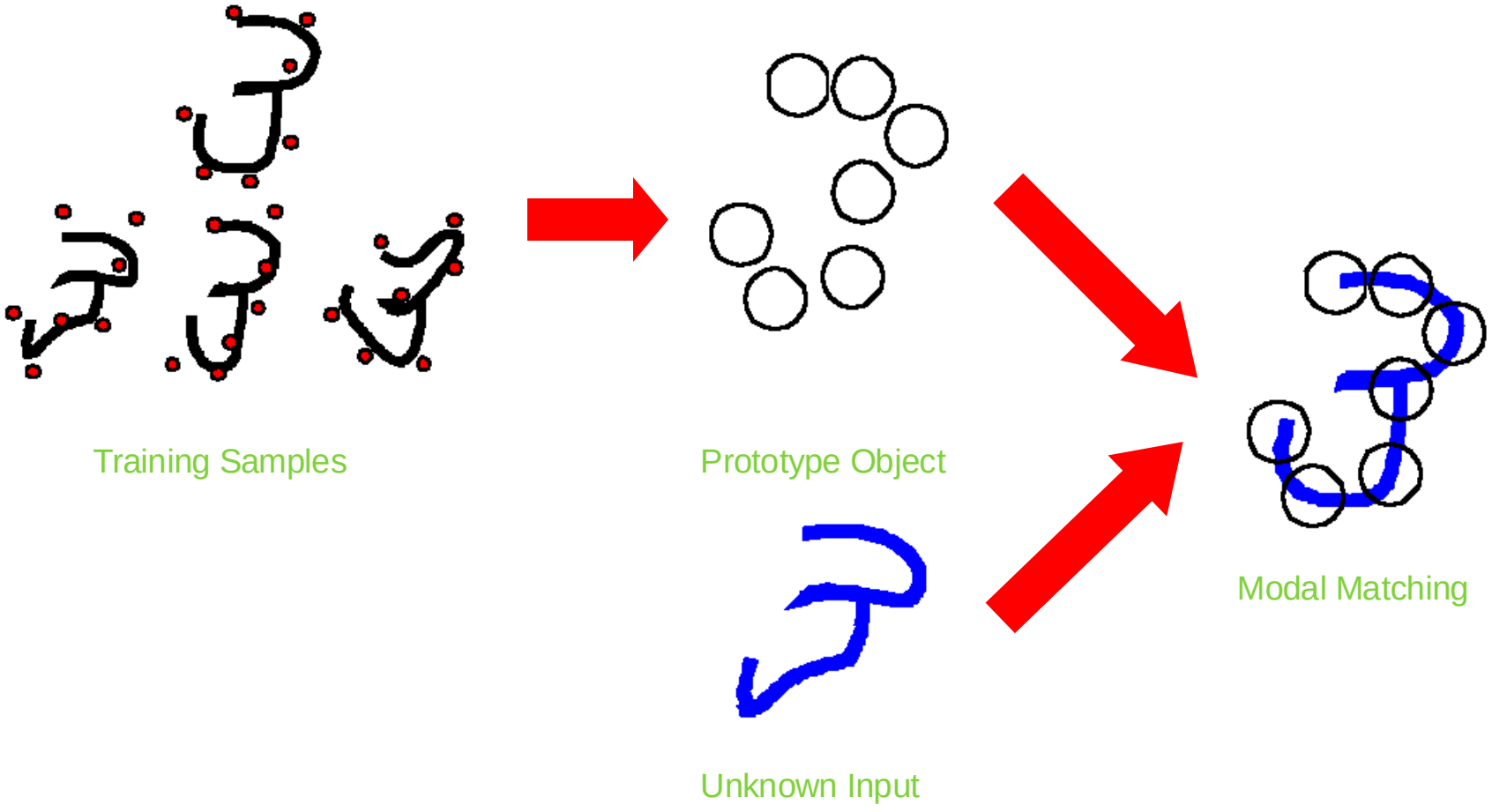
Segment Tree

- Hierarchical representation
- Present scale information
- Comply with human perception
- Invariant to translation, rotation, and scaling
- Robust to noise
- Potentials handling shape occlusion

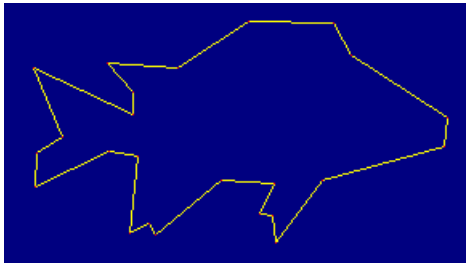
Model based Shape Matching

- Modal matching
  - Build a modal template for each prototype object and analyze similarity between the source object and the template
- Elastic template matching
  - A shape can be regarded as a template and will be deformed in order to match with a target image.
  - The deformation is modeled by a discrete set of parameters.
  - The adaptation of these parameters is controlled through a cost function.
  - The original shape is deformed until it finds a reasonable fitting with the target image.

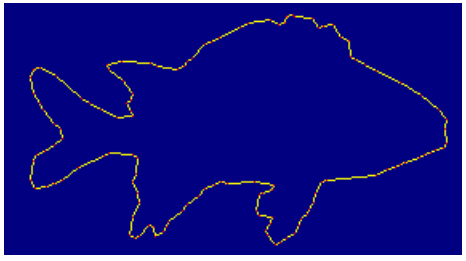
Modal matching



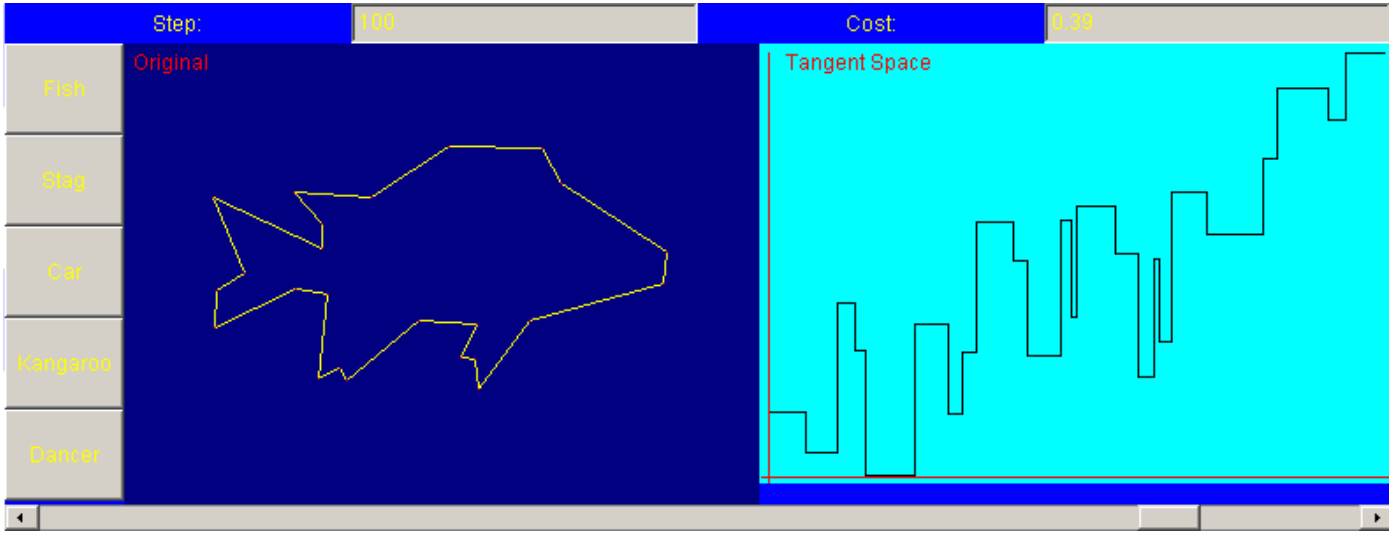
Elastic Template Matching



User Sketch



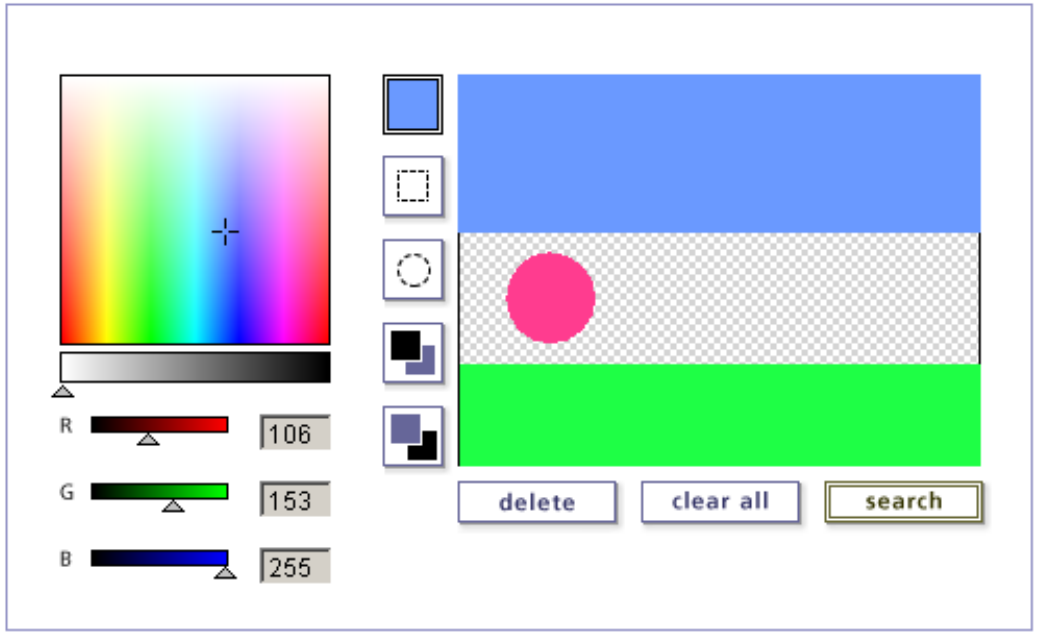
Database Image



Model based Shape Matching

- Finding correspondence between shapes
- Shape recognition/classification and retrieval, usually for non-grid objects
- To make use of domain knowledge
- Computationally expensive
- Refine initial retrieval results

QBIC LAYOUT SEARCH



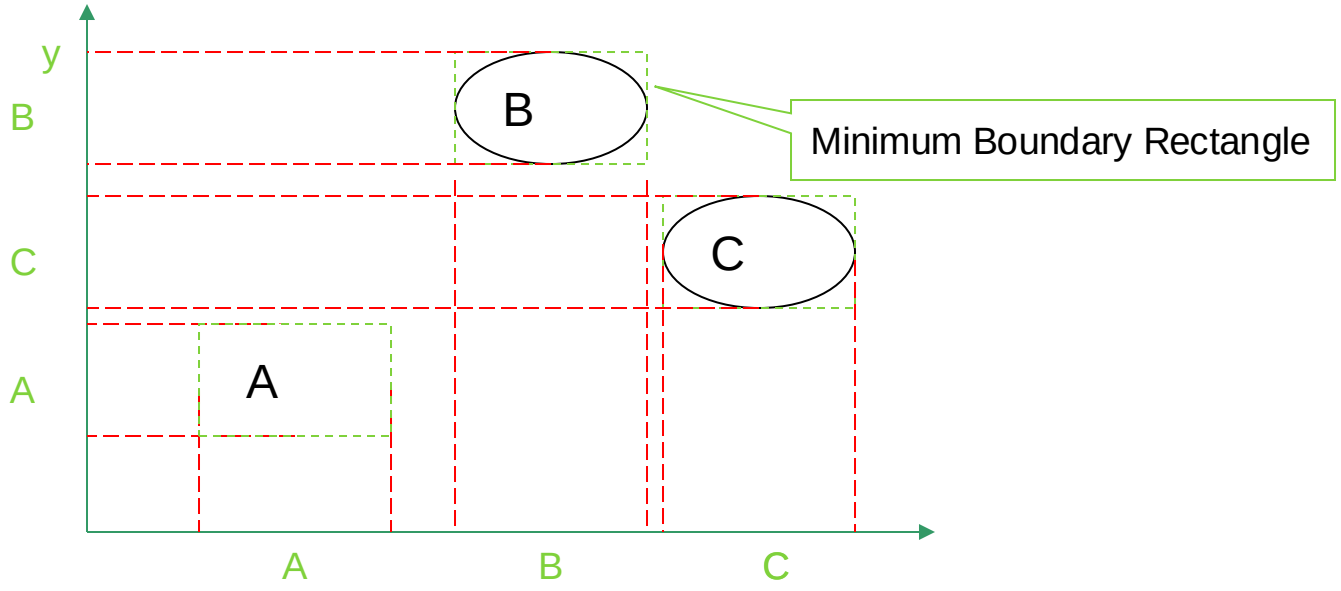
http://www.hermitagemuseum.org/fcgi-bin/db2www/qbicLayout.mac/qbic?selLang=English

Spatial Relationship

- Representation of spatial relationship
  - Directional relationship: right, left, above, …
  - Topological relationship: disjunction, adjacency, overlapping, …
- 2D String
- 2D G-String
- 2D C-String

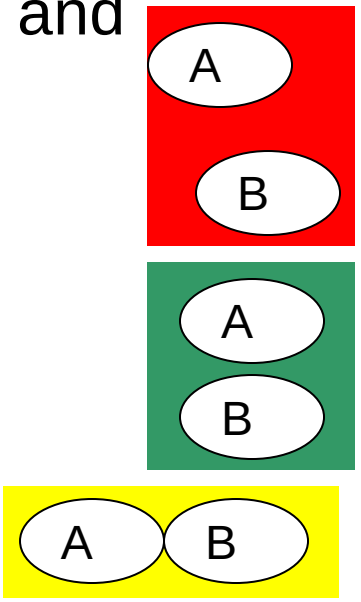
2D String

- Find centroid of each object
- Project each centroid to the *x*- and *y*-axis
- For (*u*, *v*) string along *x*- and *y*-axis
  - (ABC, ACB), which is 2D string.



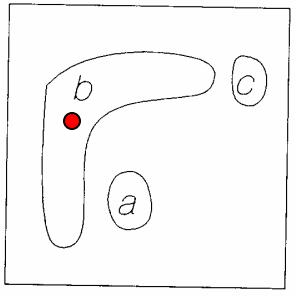
2D String

- Usually, there are three spatial operators appearing in the 2D string.
  - < :denote the left-right relationship in *u* string and below-above relationship in *v* string.
  - =:denote the spatial relationship “at approximately the same spatial location as”
  - |:denote the edge-to-edge relationship

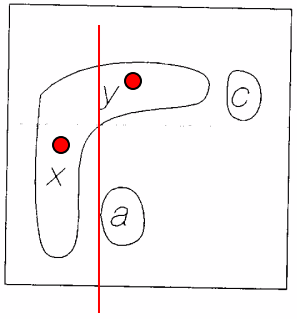


2D String

Complicated example



(a)



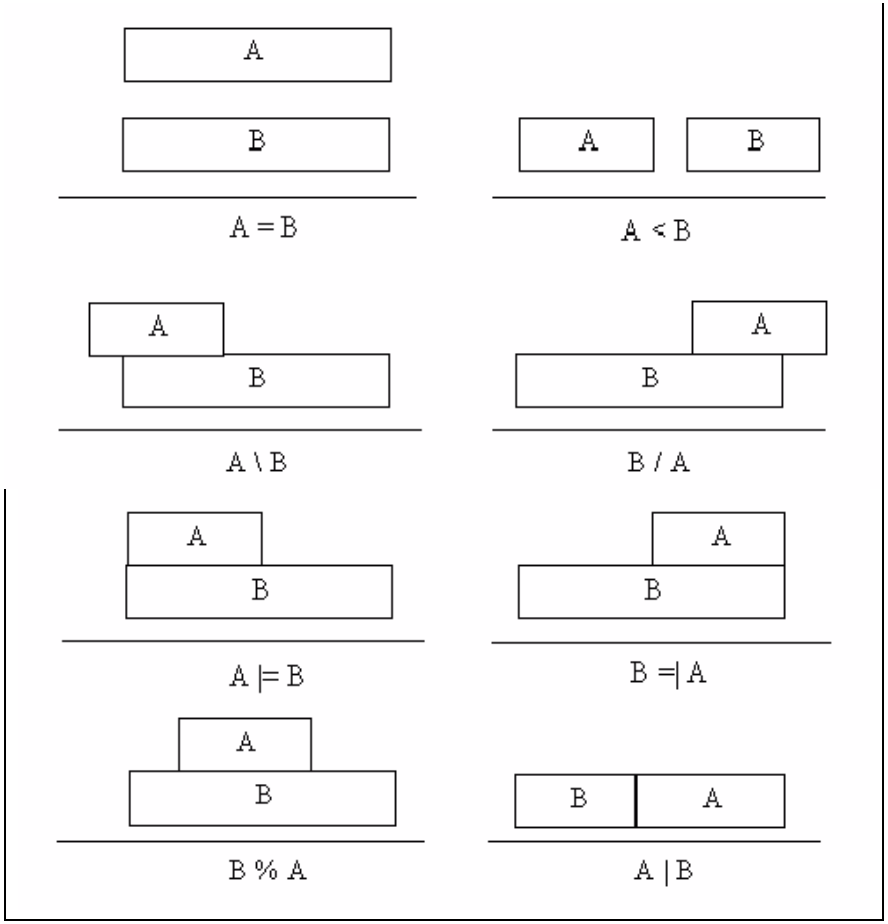
(b)

(b < a < c, a < b < c) (b < a = b < c, a < b < c < b)

```
begin
  /*object recognition*/
  recognize objects in the picture;
  find minimum boundary rectangle (MBR) of each
    object;
  /*segmentation*/
  while the MBRs overlap
    begin
      segment overlapping objects into constituent
    objects;
      find MBR of each segmented object;
    end
  /*now all objects are disjoint*/
  find centroid of each object;
  find 2D string representation using Procedure
    2Dstring
end
```

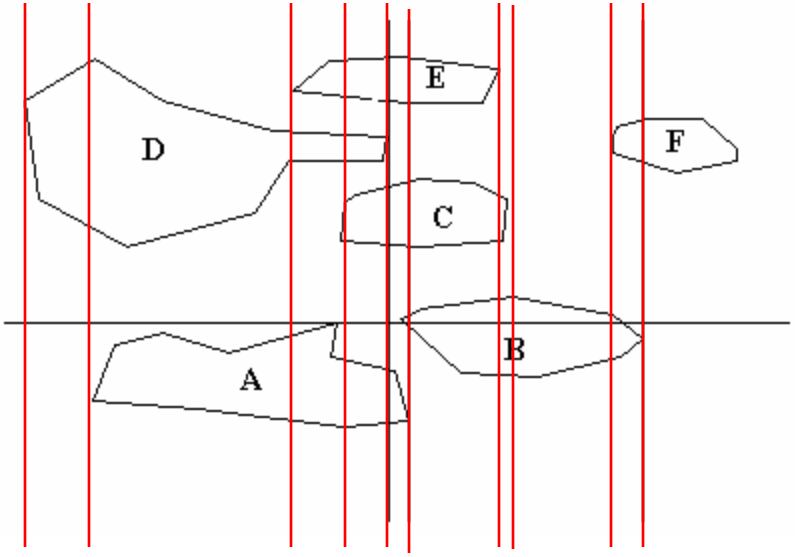
2D G-String

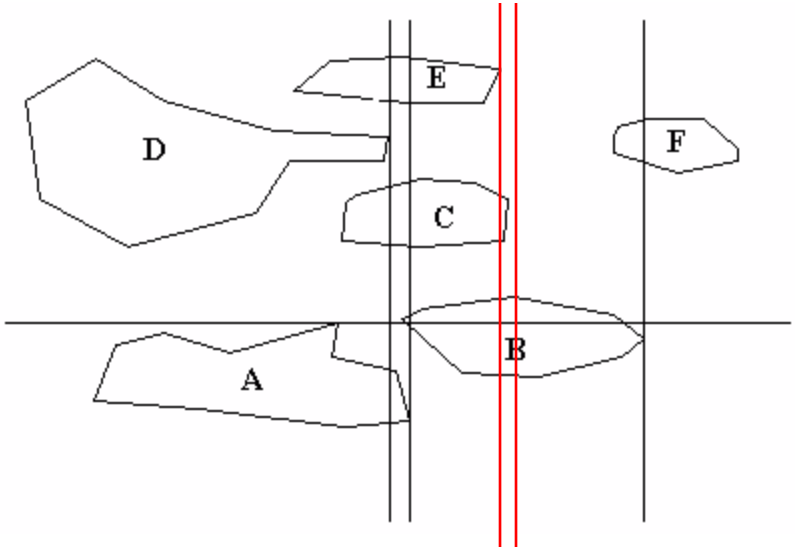
- Generalized 2D String
  - Extend spatial operator
  - Cut all the objects along their MBR
  - Extend spatial relationship to two sets of spatial operators, R_l and R_g
  - R_l defines the local spatial relationships that are partially overlapping projections between two objects.
  - R_g defines global spatial relationships such as disjoining, adjoining, and in the same position.



2D C-String

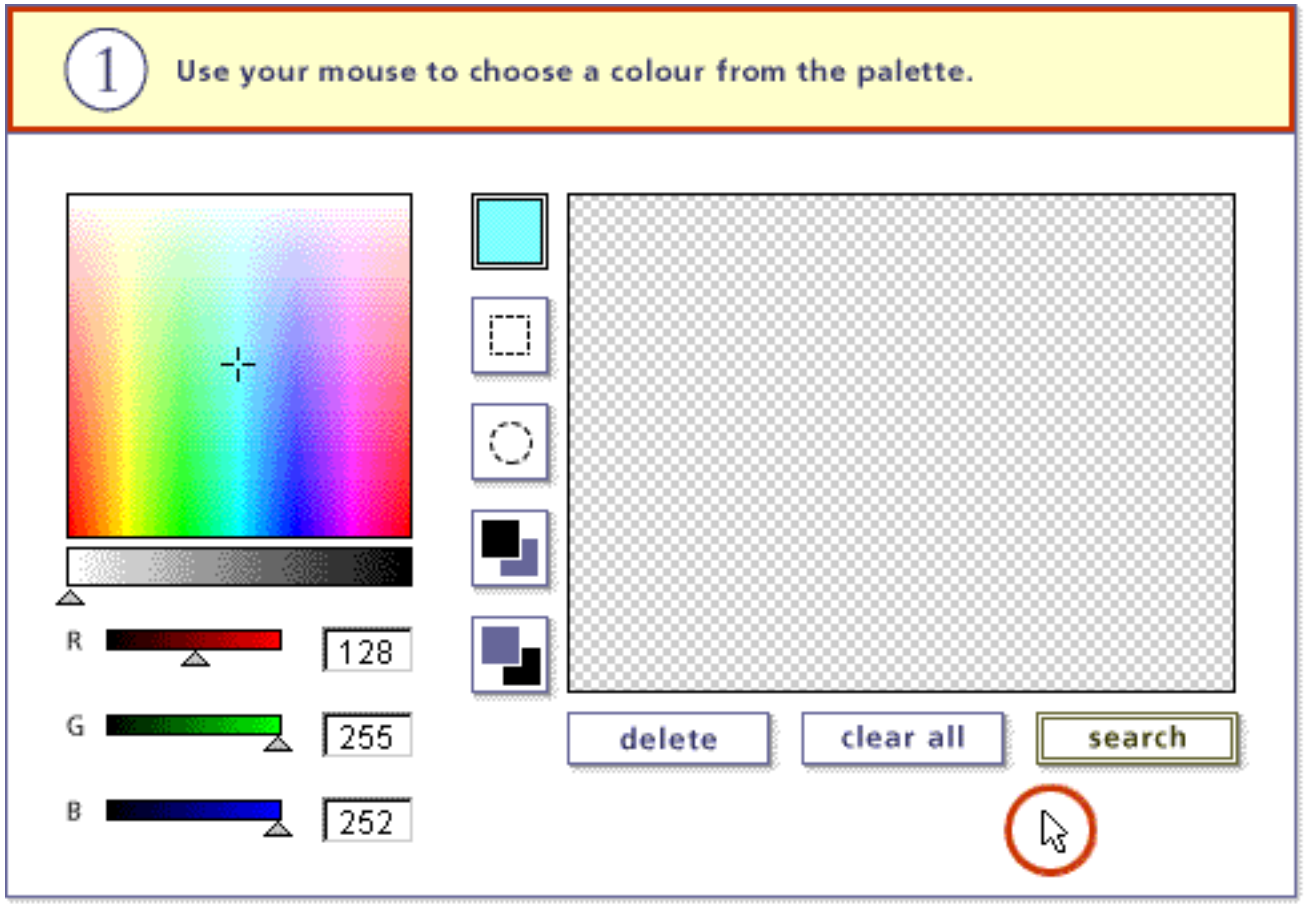
- 2D C-string tries to minimize the number of cutting objects by keeping the leading object uncut.





More About 2D String

- Other variants of 2D string
  - 2D string based on Minimum Boundary Ellipse (MBE) and Minimum Boundary Circle (MBC)
  - 2D string invariant to rotation
  - … …
- Similarity measure is based on string matching.


1
Use your mouse to choose a colour from the palette.
R
128
G
255
B
252
delete
clear all
search

Query Clear Reset

B

A

Grid Paint Help

A B C D E F G
H I J K L M N
O P Q R S T U
V W X Y Z 1 2

Spatial Query:

☑ Absolute ☐ Relative

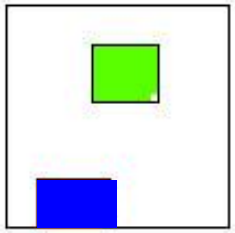
Query Weights:

Spatial 10
Feature 10
Size 10
Region 10

Photographs

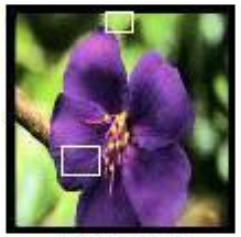
SaFe Spatial and Feature query system

http://www.ctr.columbia.edu/VisualSEEk/



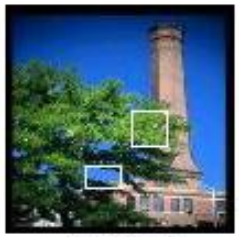
VisualSEEk Photographs Database: Spatial and Feature Query

1049 matches for REGION 1
1398 matches for REGION 2

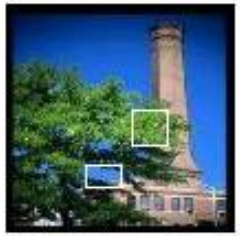


0 [2227] (556.232)

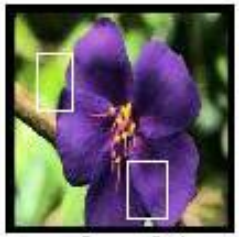
1 [2099] (570.828)



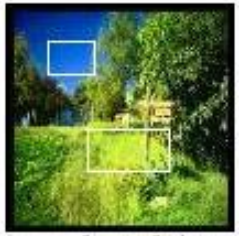
2 [361] (603.384)



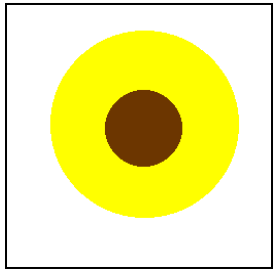
3 [96] (603.384)



4 [1228] (620.328)

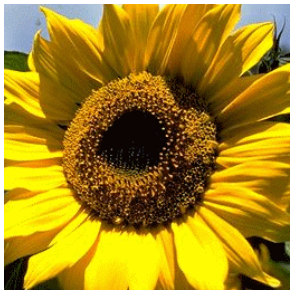


5 [2841] (633.984)







C.-C. Chang, Dept. of Computer Science,
National Tsinghua University, Taiwan

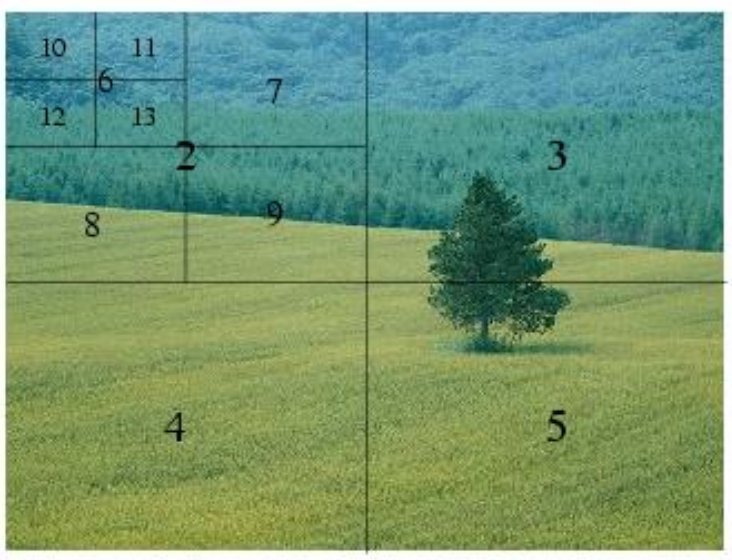
Other Image Representations

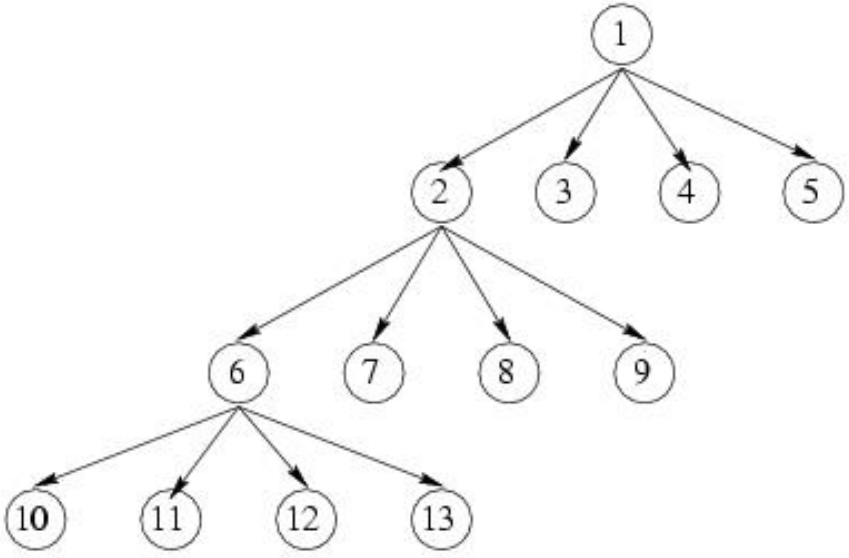
- Compression domain: fractal coding
  - Multimedia data are compressed
  - Both compression and retrieval seek less information characterizing contents.
- Graph representation

Graph Representation

- Can be considered as a special spatial information representation.
- Segmentation free
- Segmentation based

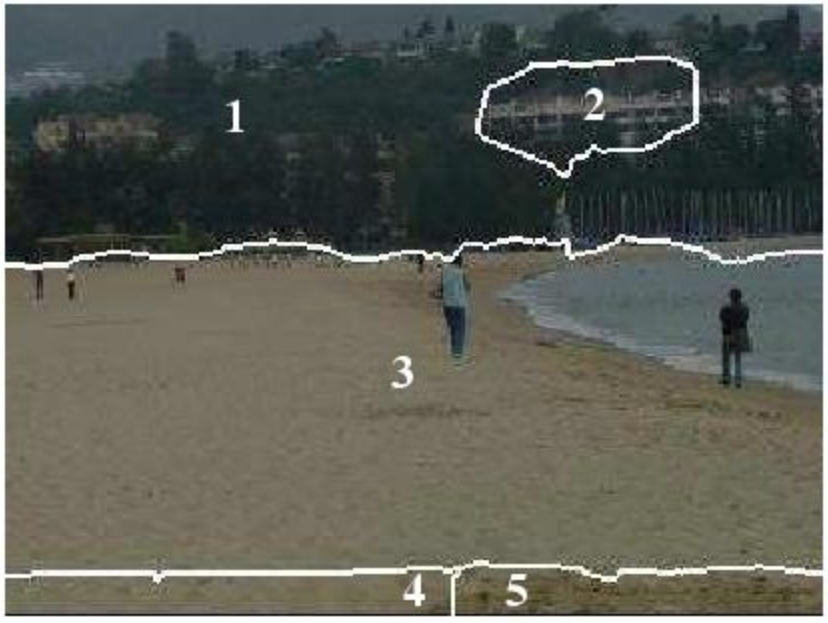
Quad-tree

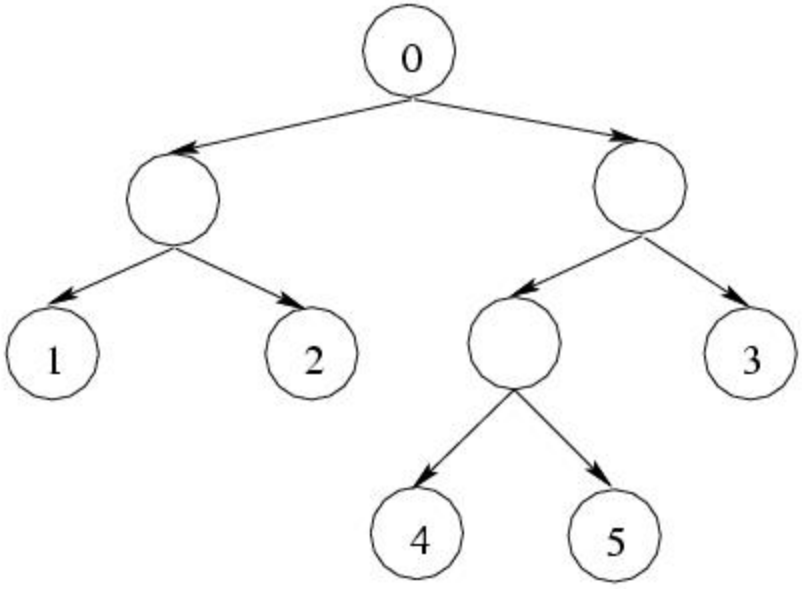




- Segmentation free
- Tree nodes are lack of semantic meaning.

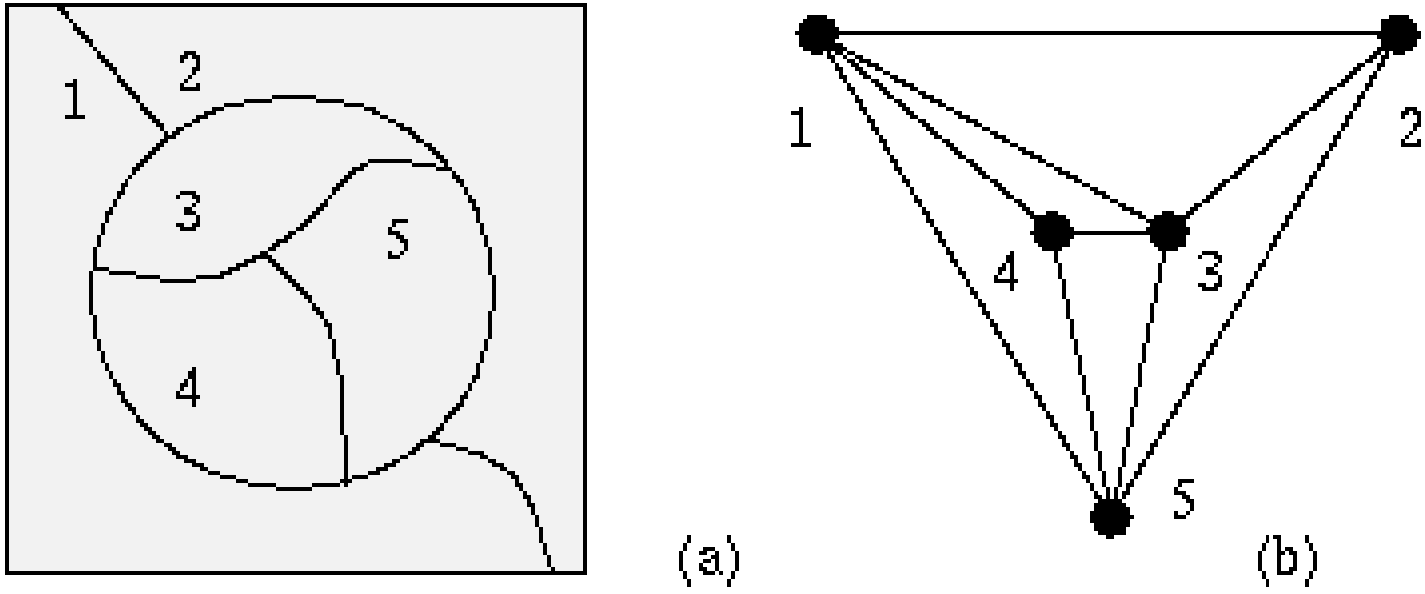
Region-based Tree





- Tree nodes are more meaningful.
- Building tree is more complicated.
- Tree is very sensitive to segmentation results.

Adjacent Graph

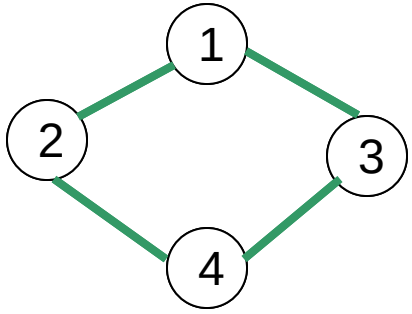


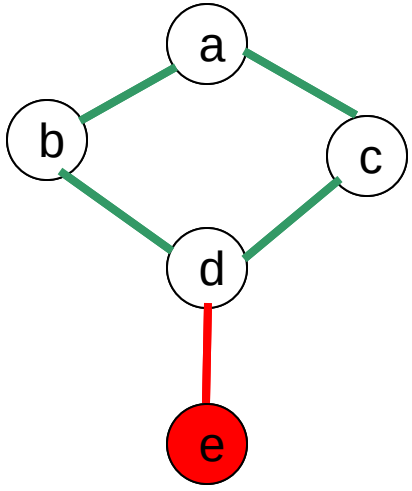
Attributed graph -

- Node attributes.
- Edge attributes.

Processing of Graph Representation

- Graph matching
  - Editing distance
- Neural network
  - Adaptive processing of data structure
  - Structured self-organizing map (SOM)
  - …



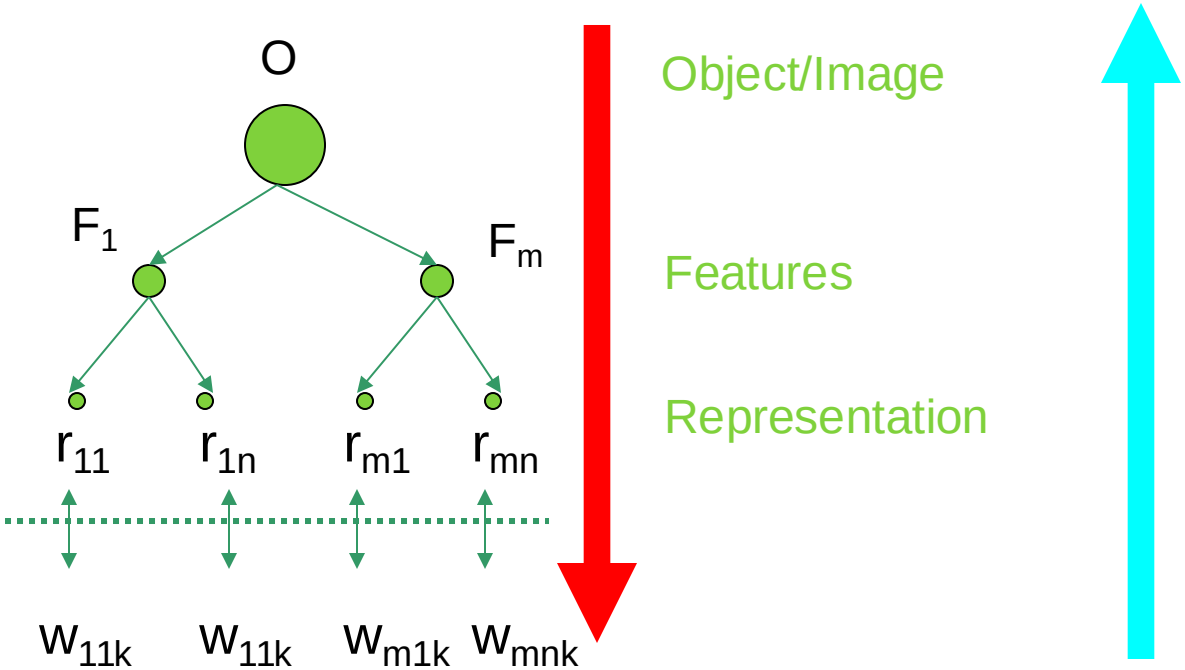


Concerns on Feature Extraction

- Effective: good at characterizing contents
  - Discrimination
  - Tolerance
- Storage efficient
- Time efficient

Multiple Features

- Different feature types
- Different feature representations for a feature type
- No generally best methods are found.
- Combined features lead to better performance



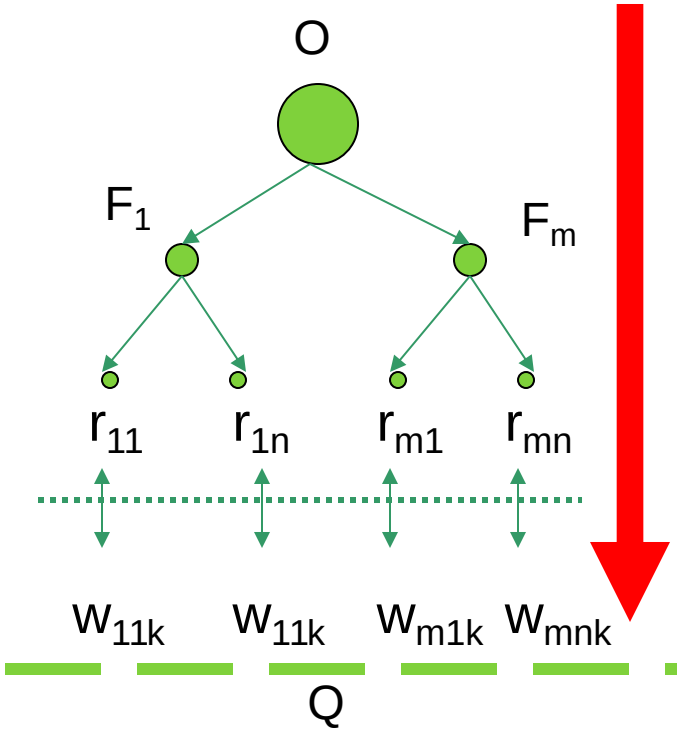
Feature Combination

Combination methods

- Weighted combination
- Fuzzy techniques
- Neural network etc.

BUT

- Different features have different dynamic range value.
- Very high dimension vector



Object/Image

Features

Representation

$$S = \sum_{i=1}^{m} W_i S(F_i) \quad \text{Eq 5}$$

$$S(F_i) = \sum_{j=1}^{n} W_{ij} S(r_{ij}) \quad \text{Eq 6}$$

$$S(r_{ij}) = m_{ij}(r_{ij}, W_{ijk}) \quad \text{Eq 7}$$

Feature Normalization

- Intra-normalization
  - To perform normalization within a feature vector r_{ij} in the whole database.
- Inter-normalization
  - To perform normalization so that equal emphasis will be put on each similarity value $S(r_{ij})$ in Eq 6 within the overall similarity value S (Eq 5).

Intra-Normalization

$$\begin{bmatrix} F_{00} & F_{01} & \cdots & F_{0n} & \cdots & F_{0N} \\ F_{10} & F_{11} & \cdots & F_{1n} & \cdots & F_{1N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ F_{m0} & F_{m1} & \cdots & F_{mn} & \cdots & F_{mN} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ F_{M0} & F_{M1} & \cdots & F_{Mn} & \cdots & F_{MN} \end{bmatrix}$$

N-dimensional features of M images

For column n, min_n, max_n, μ_n, and δ_n, respectively, are the minimum, maximum, mean, standard deviation values of the column.

1. $$F'_{mn} = \frac{F_{mn} - \min_n}{\max_n - \min_n}$$ Eq 8

2. $$F'_{mn} = \frac{F_{mn} - \mu_n}{\alpha\delta_n}$$ Eq 9

- α=1, the probability of each value being in [-1,1] is 68%.
- α=3, the probability of each value being in [-1,1] is 99%.

Inter-normalization

Steps:

- For any pair of images I_m and I_n ($m \neq n$), compute $S_{mn}(r_{ij})$
- For the obtained $M \times (M-1)/2$ similarity $S_{mn}(r_{ij})$ values, we have min_{ij}, max_{ij}, μ_{ij}, and δ_{ij}.
- Perform normalization to each $S_{mn}(r_{ij})$ value by using either Eq 8 or Eq 9.

Feature selection and dimension reduction

- Principal component analysis
- Neural network training
- … …

Textbook: pp.194-199,Chapter 8

Similarity Measure

- Distance function is employed to measure similarity.
- Some commonly used similarity measures
  - Euclidean distance
  - Minkowski-Form (MF) distance
  - Quadratic-Form (QF) distance
  - Mahalanobis distance
  - Kullback-Leibler (KL) divergence and
  - Jeffrey-Divergence (JD)

Euclidean Distance

- One of the most popularly used distance.

$$D(I_Q, I_D) = \left(\sum_i | F_i(I_Q) - F_i(I_D) |^2\right)^{1/2}$$

where $D(I_Q, I_D)$ is the distance measure between the query (example) image I_Q and the image I_D in the database; and $F_i(I)$ is the value of ith dimension feature of the image I .

Minkowski-Form (MF) Distance

- MF distance is a generalization of Euclidean distance by assuming that each dimension of image feature vector is independent of each other and is of equal importance.

$$D_p(I_Q, I_D) = \left(\sum_i | F_i(I_Q) - F_i(I_D) |^p\right)^{1/p}$$

- While p = 1, it becomes City-block distance, p = 2, it becomes Euclidean distance.

Quadratic-Form Distance

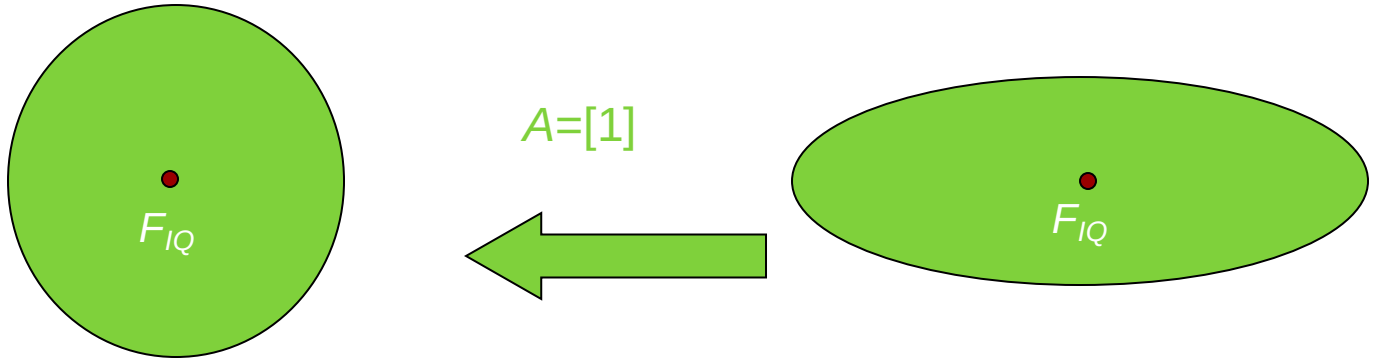
- The MF distance treats all dimensions of image feature entirely independently and does not account for the fact that certain pairs of feature dimensions which are perceptually more similar than other pairs.
- To solve this problem, the QF distance is introduced:

$$D(I_Q, I_D) = \sqrt{(\mathbf{F}_{I_Q} - \mathbf{F}_{I_D})^T \mathbf{A} (\mathbf{F}_{I_Q} - \mathbf{F}_{I_D})}$$

where where $A=[a_{ij}]$ is a similarity matrix, and a_{ij} denotes the similarity between the i-th and j-th dimension feature. $\mathbf{F}_{I_Q}$ *and* $\mathbf{F}_{I_D}$ are vectors that list all the entries in $F_i(I_Q)$ and $F_i(I_D)$.

- The QF distance has been used in many retrieval systems for color histogram-based image retrieval, such as IBM's QBIC. It has been shown that QF distance can lead to perceptually more desirable results than *Euclidean* distance and histogram intersection method as it considers the cross similarity between colors.

Euclidean vs. Quadratic



Euclidean distance

Quadratic distance

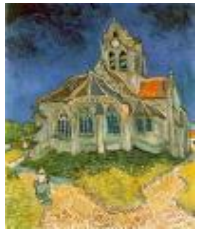
Relevance Feedback (RF)

- Visual features are most based on low level features and lack semantic meanings.
- End users are an essential part of the system and they should be allowed to comment the retrieval results.
- Human perception is subjective and also task-dependent.
  - Different users expect different results even from the same query example.
  - The same user expect different results at different times even from the same query example.
- To make the retrieval process more flexible and more adaptive to the end users.




Sunflower in the context of flower






Sunflower in the context of Van Gogh’s paintings

Relevance Feedback

- RF techniques are to provide users with the opportunity to refine the retrieval results and incorporate human perception subjectivity into the retrieval process.
- RF techniques is first developed for text information retrieval to improve the effectiveness of information systems.
- RF uses positive and/or negative examples marked by the users among the retrieved results.
  - The system returns a result based on a predefined similarity measure.
  - The users marked the result items as relevant or irrelevant.
  - The system models the users' subjectivity and updates its similarity measure by making use of the feedback information.
  - The system returns a new result with the most updated similarity measure
  - Repeat the above 4 steps till a satisfied result is obtained.

Relevance Feedback

- Query space modeling methods
- Similarity metrics updating methods
- Hybrid methods
  - Combine different methods
  - Integrate semantic information (textbook pp.65-70)

Query Space Modeling

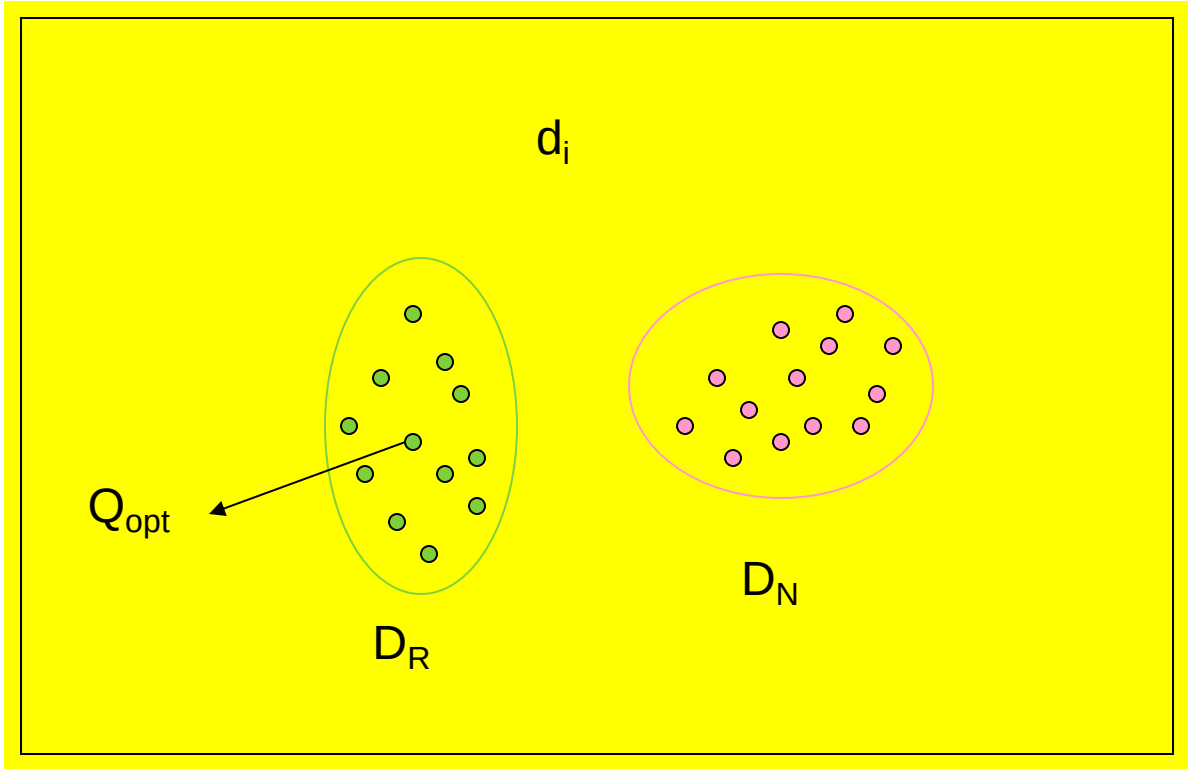
Rocchio's formula for text information retrieval.

$$Q_{opt} = \frac{1}{N_R} \sum_{i \in D_R} d_i - \frac{1}{N_T - N_R} \sum_{i \in D_N} d_i$$

where N_T and N_R **are the number of the total returned documents and the number of the relevant documents among the returned documents.**

$$Q' = \alpha Q + \beta \left(\frac{1}{N_{R'}} \sum_{i \in D'_R} d_i \right) - \gamma \left(\frac{1}{N_{T'} - N_{R'}} \sum_{i \in D'_N} d_i \right)$$

Query Space Modeling



Similarity Metrics Updating

$$\begin{bmatrix} F_{00} & F_{01} & \dots & F_{0n} & \dots & F_{0N} \\ F_{10} & F_{11} & \dots & F_{1n} & \dots & F_{1N} \\ \dots & \dots & \dots & \dots & \dots\dots & \\ F_{m0} & F_{m1} & \dots & F_{mn} & \dots & F_{mN} \\ \dots & \dots & \dots & \dots & \dots\dots & \\ F_{M0} & F_{M1} & \dots & F_{Mn} & \dots & F_{MN} \end{bmatrix}$$

Feature matrix of relevant images

For column n, μ_n and δ_n, respectively, are the minimum, maximum, mean, standard deviation values of the column.

$$D(I_Q, I_D) = \sqrt{(\mathbf{F}_{I_Q} - \mathbf{F}_{I_D})^T \mathbf{A} (\mathbf{F}_{I_Q} - \mathbf{F}_{I_D})}$$

Objectives -

- The feature components that contribute more similarity to the match are considered more important.
- The feature components with smaller contribution are considered to be less important..

	Color	Size	Shape	Texture
	Red	Small	Round	Uniform
	Red	Small	Round	Uniform
	Orange	Big	Round	Non-Uniform
	Yellow	Small	Round	Non-Uniform

Similarity Metrics Updating

$$\begin{bmatrix} F_{00} & F_{01} & \dots & F_{0n} & \dots & F_{0N} \\ F_{10} & F_{11} & \dots & F_{1n} & \dots & F_{1N} \\ \dots & \dots & \dots & \dots & \dots\dots & \\ F_{m0} & F_{m1} & \dots & F_{mn} & \dots & F_{mN} \\ \dots & \dots & \dots & \dots & \dots\dots & \\ F_{M0} & F_{M1} & \dots & F_{Mn} & \dots & F_{MN} \end{bmatrix}$$

Feature matrix of relevant images

$$D(I_Q, I_D) = \sqrt{(\mathbf{F}_{I_Q} - \mathbf{F}_{I_D})^T \mathbf{A} (\mathbf{F}_{I_Q} - \mathbf{F}_{I_D})}$$

$$a_{ii} = \frac{1}{\delta_i}$$

$$a_{ii}^{+} = a_{ii}(1 + \bar{\delta} - \delta_i)$$

$$a_{ii}^{-} = a_{ii}(1 - \bar{\delta} + \delta_i)$$

Similarity Metrics Updating

- Instead of updating the individual components of a distance metric, the updating can also happen between different metrics.
- RF can also be used to dynamically determine the feature models to best characterize image contents for the following retrieval.

Relevance Feedback

- RF can be considered as a supervised learning problem in the view point of pattern recognition.
- Most machine learning methods are suitable for RF???
  - Enough training examples.
  - Real-time response.
  - Support vector machine (SVM)

Image Space Visualization

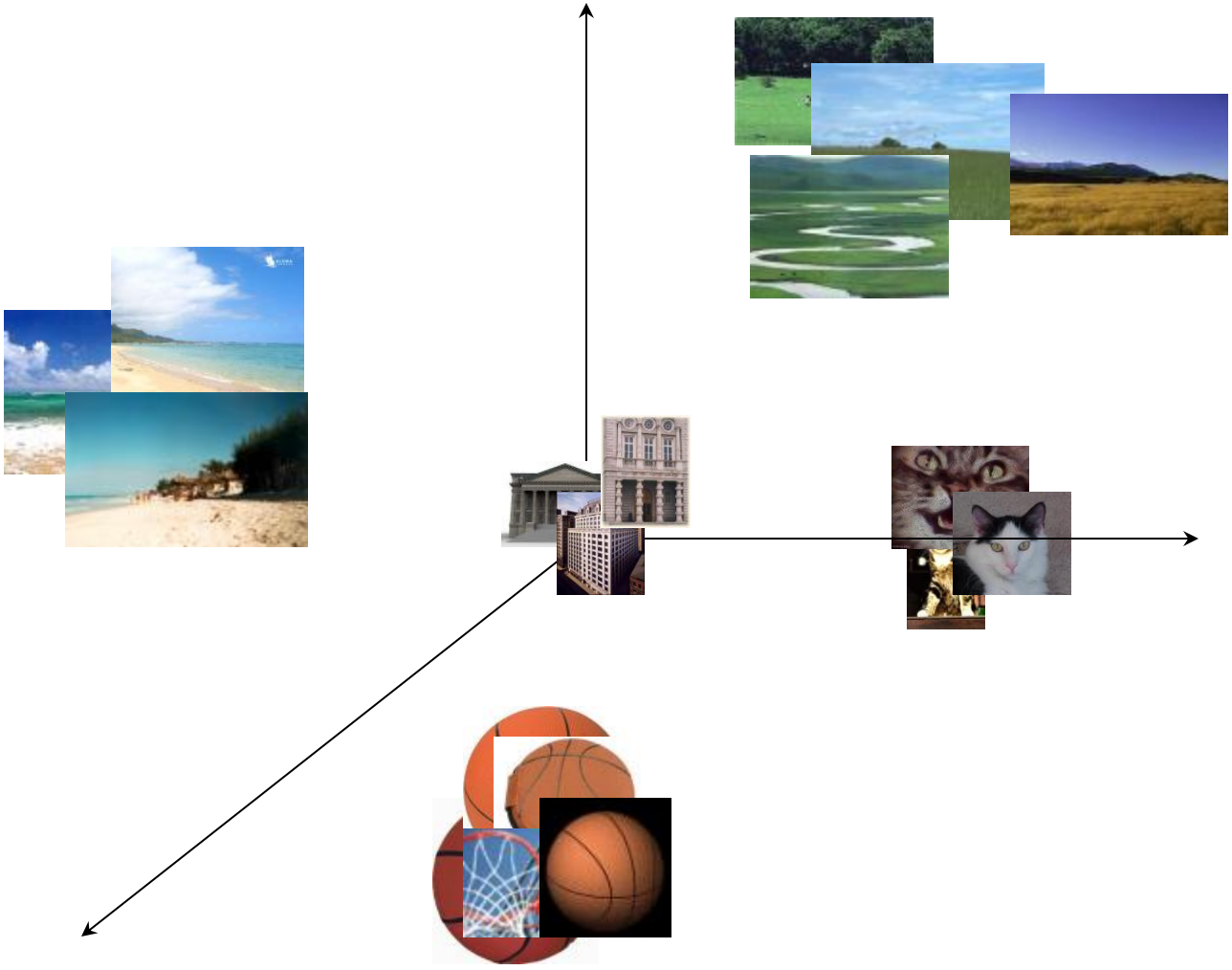
- Flat 2-D visualization






- High dimensional visualization

Image Space Visualization



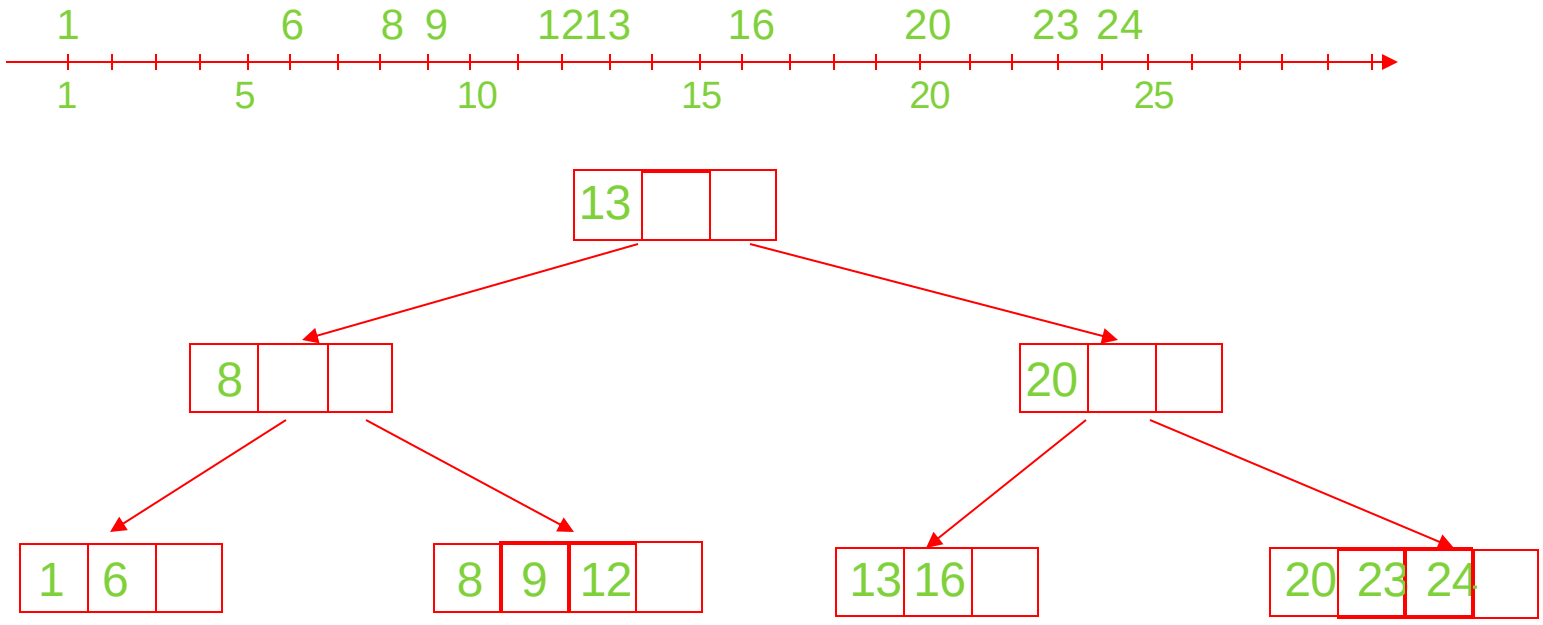
Indexing Techniques

- Consider a feature vector [f_1, f_2, …, f_d] as a point p in a d-dimensional space.
- Retrieving images similar to a given query image is transformed to a *k-nearest neighbor searching* problem.
- Solutions
  - Exhaustive/Linear search
  - Indexing techniques: pre-organize data as to achieve efficient search.

Indexing Techniques

- One dimensional indexing
  - Linear
  - Binary tree
  - Distance matrix
- Grid files
- K-d trees, k-d B trees
- R-tree family: R-trees, R^+ Trees, R^* Trees
- SS-trees, CSS^+-Trees: see pp.183-189, Chapter 8, textbook.
- Other trees: TV-trees, X-trees,
- Hash-based methods
- Projection-based methods

One-dimension Indexing



- Binary tree search
- Widely used in relational database system

For example, q = 14

Distance Matrix

$$\begin{pmatrix} dist_{0,0} & dist_{0,1} & \cdots & dist_{0,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ dist_{n-1,0} & dist_{n-1,1} & \cdots & dist_{n-1,n-1} \end{pmatrix}$$

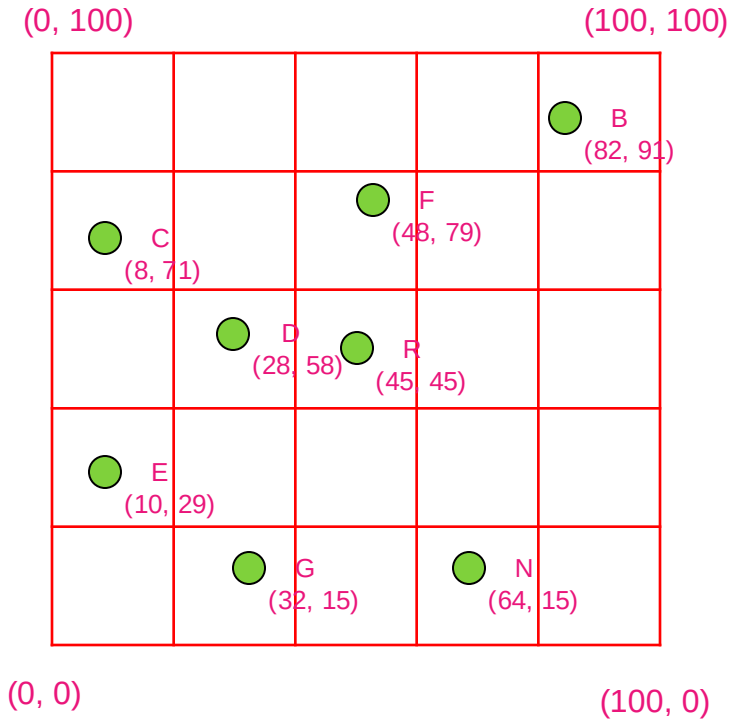
Where $dist_{i,j}$ is the distance between p_i and p_j

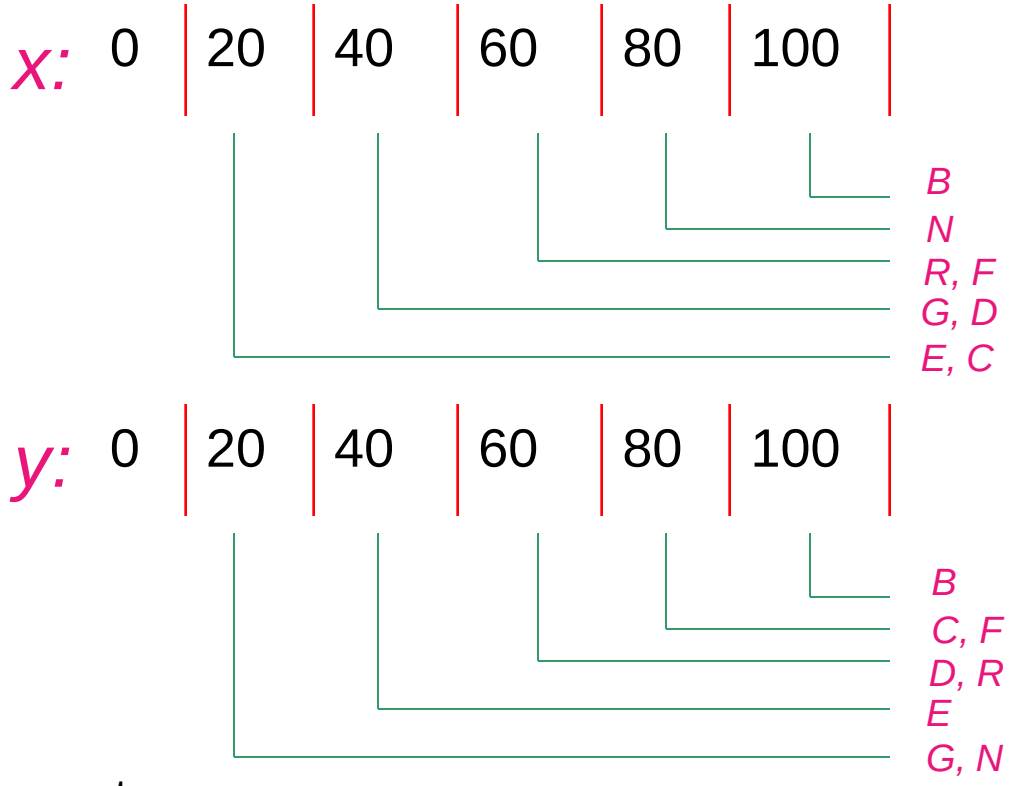
- This matrix is symmetric.
- It can offer constant access time.
- However, it is not flexible and scalable
  - Query item should be from the database.
  - Adding new item is an expensive operation.
  - It will be impossible to retrieve more than *m* best matches of a query image.

Search procedure for p_i

- Sort the distances in row *I*
- Return the first *m* best matches

Two-dimension Indexing





- Records are retrieved with, at the most, two access.
- As the buckets overflow, a splitting process will take place.

- The bucket size need not to be equal.
- Grid files are suitable for range search.

Grid files

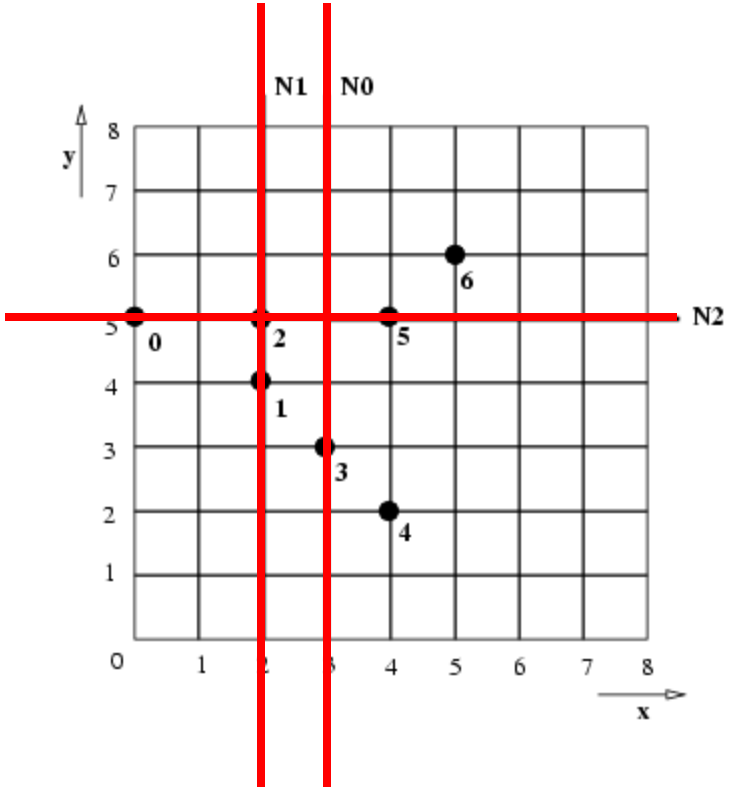
K-d Trees

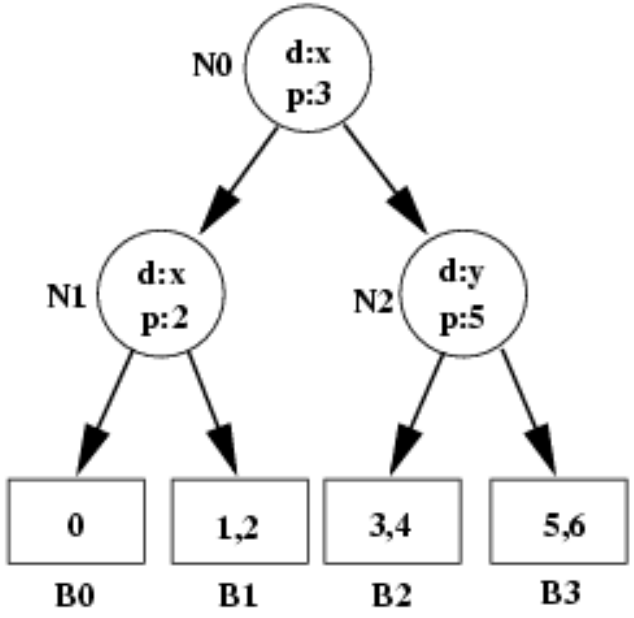
- A *k*-d tree is a binary tree in which each node represents a hyper-rectangle and a hyper-plane orthogonal to one of the coordinate axis, which splits the hyper-rectangle into two parts.
- All data points with values less than the partition value belong to the left child, while those with a larger or equal value belong to the right child.
- The search space is partitioned until the number of data points in the hyper-rectangle falls below some given threshold.
- The leaf nodes of the tree are called *buckets*, and the threshold that limits the maximum number of data points in a bucket is called the *bucket size* of the tree.

*Note that data points are only stored in leaf nodes, not in the internal nodes.

- During searching, the value of one of the k features is checked at each node, to determine the appropriate subtree.

K-d Trees





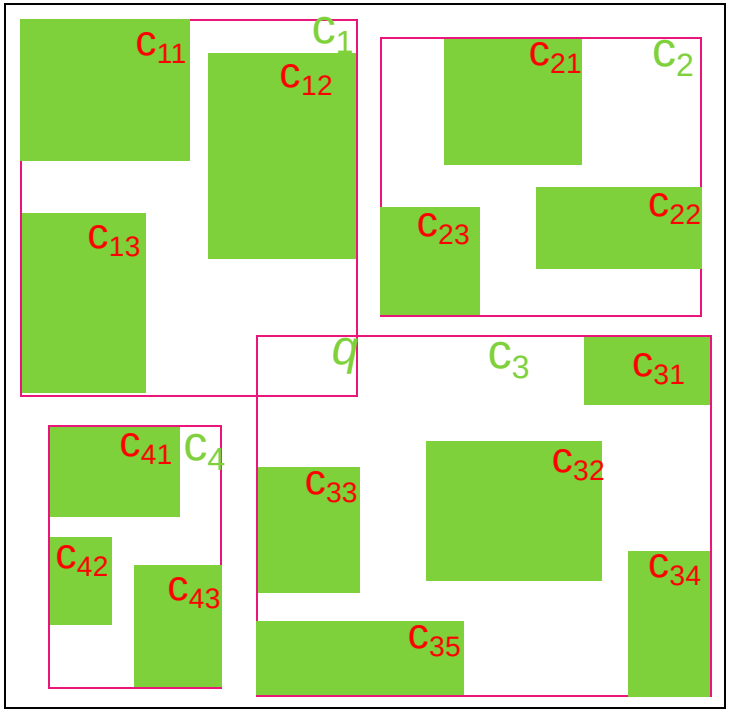
R. Egas et al., Visual99

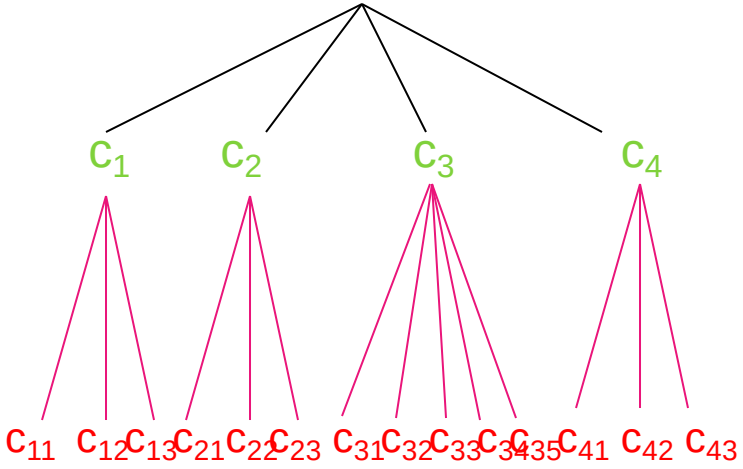
- Choose feature dimension: the dimension with the largest variance.
- Choose partition value: median of the data points

R-Trees

- R-Trees are very popular dynamic data structures for feature vector indexing. They are suitable for feature vectors of higher dimension than *k*-d trees.
- Properties
  - Each node is bounded by a spatially minimum bounding rectangle that contains all points within the node.
  - All leaf nodes are at the same level

R-Trees





- Major problems with R-tree
  - large overlapping: c1 and c3
  - inconsistent in distance and clusters: c23
  - expensive in building the tree
- other problems with R-tree
  - adaptive clustering
  - variant metrics

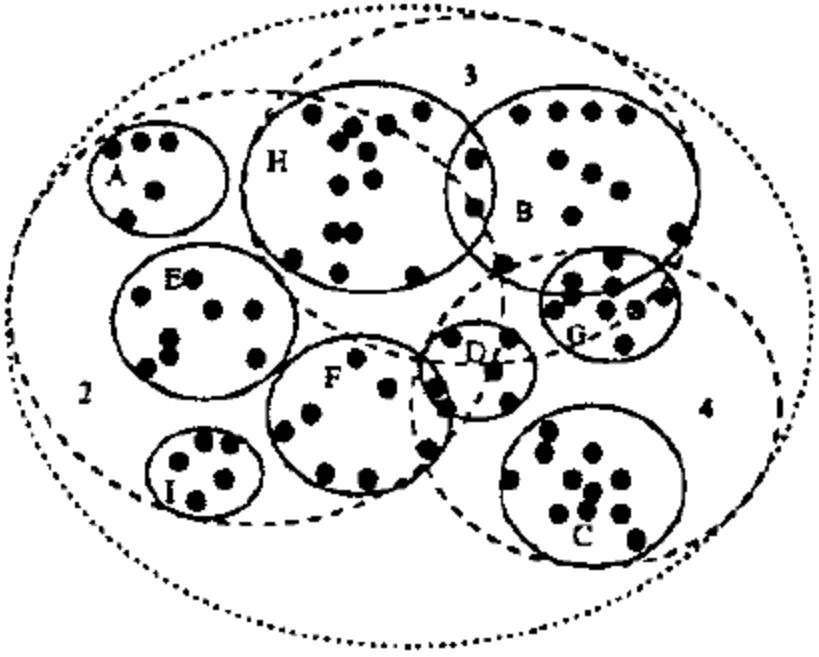
J. Jin, University of Newcastle

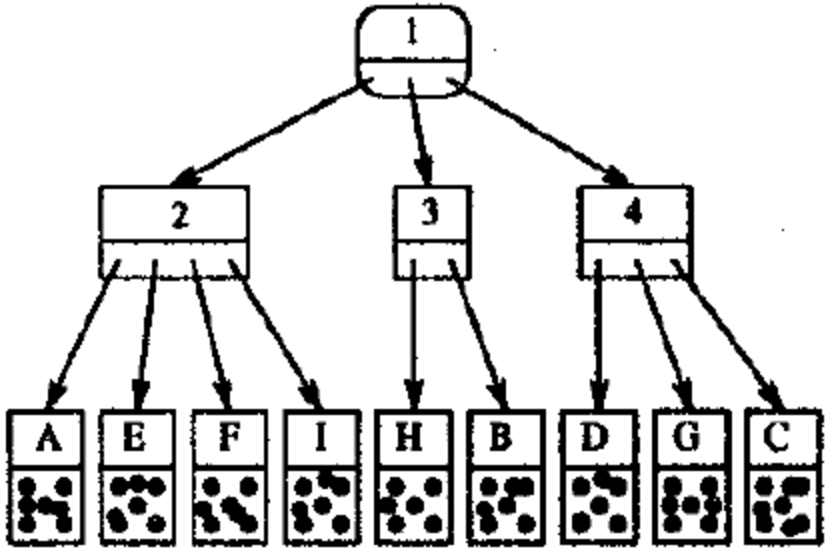
SS-Trees

- SS-Trees are dynamic data structures that have been specifically designed for similarity indexing of feature vectors in visual databases.
- Instead of using rectangle bounding box, SS-trees use bounding sphere, so that the distance in query is consistent with the bounding envelope distance, but the overlapping could be bigger than bounding rectangles.
- SS-Trees use a Quadratic distance/weighted Euclidean distance measure

$$D(x, y) = \sqrt{(x - y)^T W (x - y)}$$

SS-Trees



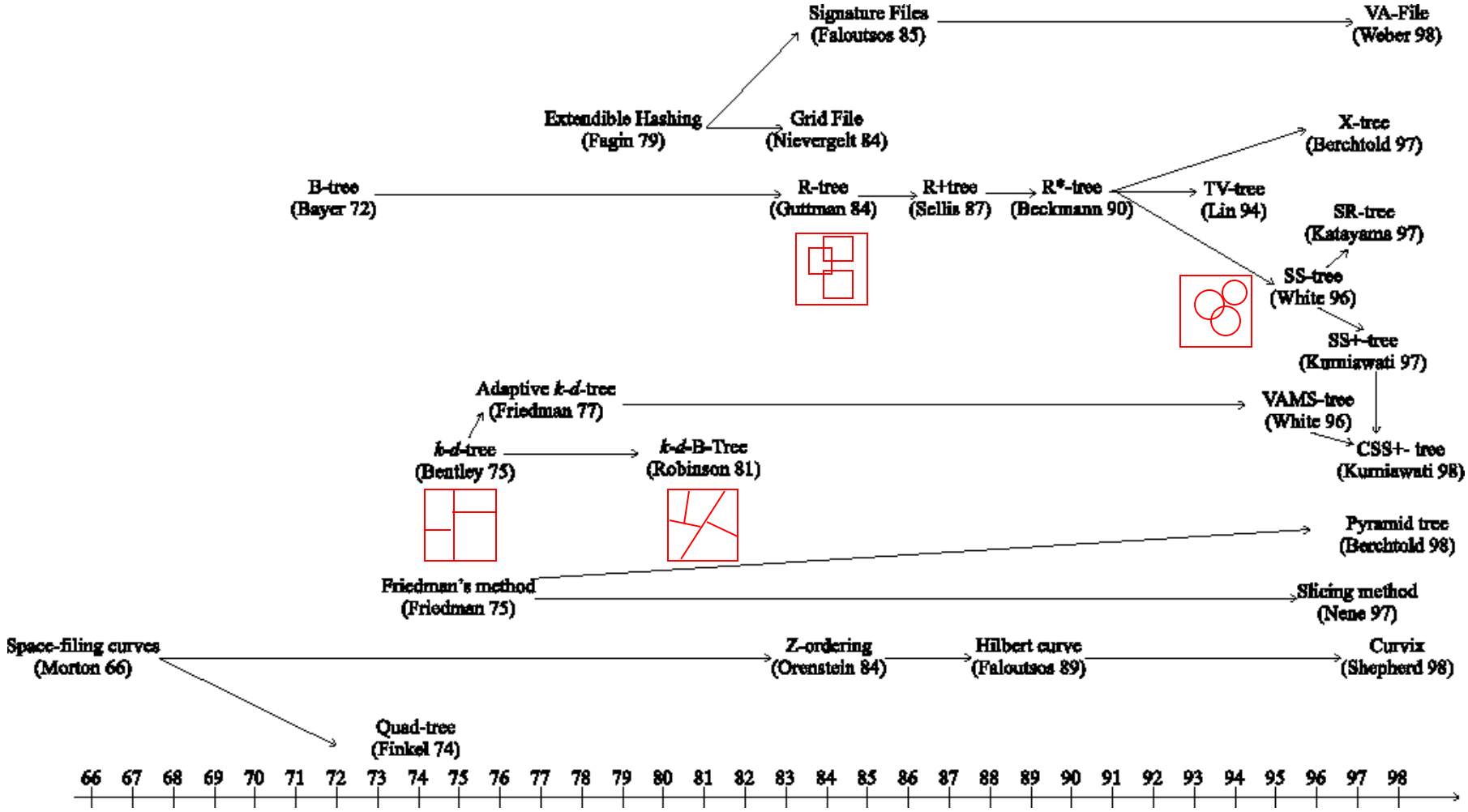


D. White and R. Jain, IEEE 1996

Representation of high dimensional data in 2D

SS-Tree Representation

Review of Indexing Techniques



Improving Indexing Performance

- Selecting the shape of bounding envelopes
- Splitting heuristics
- Heuristics to reduce overlapping
- Updating criteria
- Efficient search strategy

Issues of Indexing Techniques

- Searching efficiency
- Building cost
- Flexibility and scalability
  - Remove a node: deletion
  - Add a node: insertion

Performance Evaluation

- Traditionally, recall rate and precision are used to evaluate the retrieval performance.
  - Recall rate: to measure the ability of the system to retrieve all the documents that are relevant.
  - Precision: to measure the ability of the system to retrieve only documents that are relevant.
  - Goal: high recall rate and high precision
- Effectiveness is also used to measure the agreement between human evaluators and the system in ranking retrieved images according to the similarity to the query.

Performance Evaluation

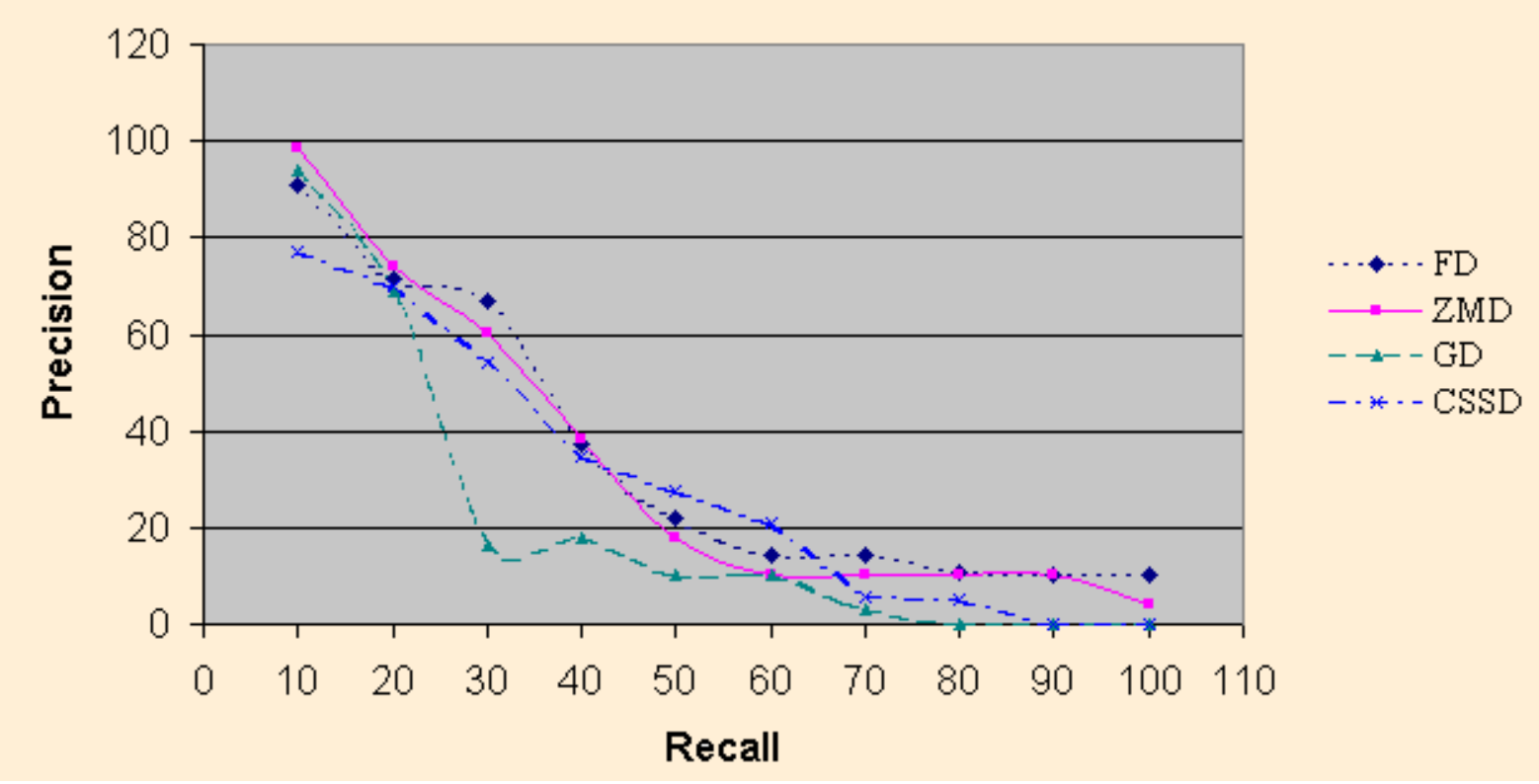
	Judgement by Evaluator	
	Relevant	Irrelevant
Retrieved	A (correctly retrieved)	B (falsely retrieved)
Not Retrieved	C (missed)	D (correctly retrieved)

$$Recall = \frac{\text{Relevant Correctly Retrieved}}{\text{All Relevant}} = \frac{A}{A+C}$$

$$Precision = \frac{\text{Relevant Correctly Retrieved}}{\text{All Retrieved}} = \frac{A}{A+B}$$

Implicitly, the number of returned documents (answer set) is referred to!

Precision-Recall (PR) Graph



Retrieval results from four shape descriptors

ANMRR

- Average normalized modified retrieval rank (ANMRR) is recommended by MPEG-7.
- It combines the precision and recall to obtain a single objective measure.

Denote the number of ground truth images for a given query *q* as *N(q)*, the maximum number of ground truth images for all *Q* queries, i.e., *max(N(q1), N(q2), …, N(qQ))* as *M*.

For a given query *q*, each ground truth image *k* is assigned a rank value *rank(k)* if it is in the first *K* results, or a rank value *K+1* if it is not. *K = min(4(N(q), 2M)*

- The average rank *AVR(q)* for query *q* is computed as

$$AVR(q) = \sum_{k=1}^{N(q)} \frac{rank(k)}{N(q)}$$

- The modified retrieval rank *MRR(q)* is computed as

$$MRR(q) = AVR(q) - 0.5 - 0.5 \times N(q)$$

- The normalized modified retrieval rank *NMRR(q)*, which ranges

$$NMRR(q) = MRR(q)/(K + 0.5 - 0.5 \times N(q))$$

- Finally, ANMRR is computed as

$$AVR(q) = \sum_{q=1}^{Q} NMRR(q)/Q$$

Other Evaluations

- *P(10)*, *P(30)*, *P(N_R)* – the *precision* after the first *10*, *30*, N_R documents are retrieved, where N_R is the number of relevant documents.
- Recall rate at 0.5 precision – recall at the rank where precision drops below 0.5

Retrieval Benchmark

- Corel image dataset (http://www.corel.com/)
- Brodatz and VisTex texture datasets
  - http://sipi.usc.edu/services/database/database.cgi?volume=textures
  - http://www-white.media.mit.edu/vismod/imagery/VisionTexture/vistex.html
- Individual collected dataset
- Viper (Visual Information Processing for Enhanced Retrieval)
  - http://viper.unige.ch/
- The Benchathlon (http://www.benchathlon.net/)
- University of Wasington
  - (http://www.cs.washington.edu/research/imagedatabase/groundtruth/

Reference Readings

- Multimedia information retrieval and management: technological fundamentals and applications, Springer, 2003
  - Chapter 1, pp.16-22
  - Chapter 2, pp.57-65
  - Chapter 8, pp.178-183

Need To Know

- Shape features: representation methods
  - Syntactic representation
  - Structure representation
  - Model representation
- Spatial feature representation methods
- Fractal and other representations
- Similarity measures, normalization
- Relevance feedback: motivations and methods
- Indexing: motivations and methods
- Performance evaluation: benchmark and parameters