Multimedia Database Systems

Retrieval by Content

MIR Motivation

Large volumes of data world-wide are not only based on text:

- ➢ Satellite images (oil spill), deep space images (NASA)
- ➢ Medical images (X-rays, MRI scans)
- ➢ Music files (mp3, MIDI)
- ➢ Video archives (youtube)
- ➢ Time series (earthquake measurements)

Question: how can we organize this data to search for information?

E.g., *Give me music files that <u>sound like</u> the file "query.mp3"*

Give me images that <u>look like</u> the image "query.jpg"

MIR Motivation

One of the approaches used to handle multimedia objects is to exploit research performed in classic IR.

Each multimedia object is annotated by using free-text or controlled vocabulary.

Similarity between two objects is determined as the similarity between their textual description.

MIR Challenges

- Multimedia objects are usually large in size.
- Objects do not have a common representation (e.g., an image is totally different than a music file).
- Similarity between two objects is subjective and therefore objectivity emerges.
- Indexing schemes are required to speed up search, to avoid scanning the whole collection.
- The proposed techniques must be effective (achieve high recall and high precision if possible).

MIR Fundamentals

In MIR, the user information need is expressed by an object *Q* (in classic IR, *Q* is a set of keywords). *Q* may be an image, a video segment, an audio file. The MIR system should determine objects that are similar to *Q*.

Since the notion of similarity is rather subjective, we must have a function $S(Q,X)$, where *Q* is the query object and *X* is an object in the collection. The value of $S(Q,X)$ expresses the degree of similarity between *Q* and *X*.

MIR Fundamentals

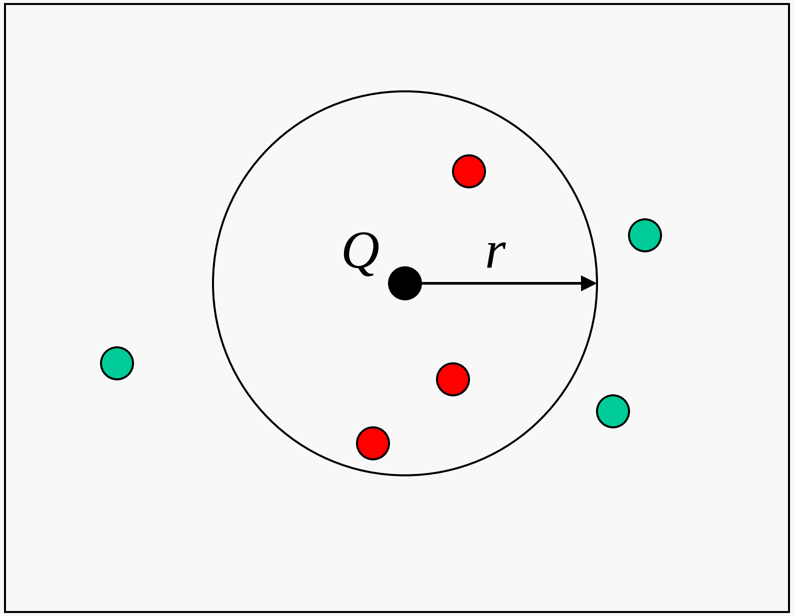
Queries posed to an MIR system are called similarity queries, because the aim is to detect similar objects with respect to a given query object. Exact match is not very common in multimedia data.

There are two basic types of similarity queries:

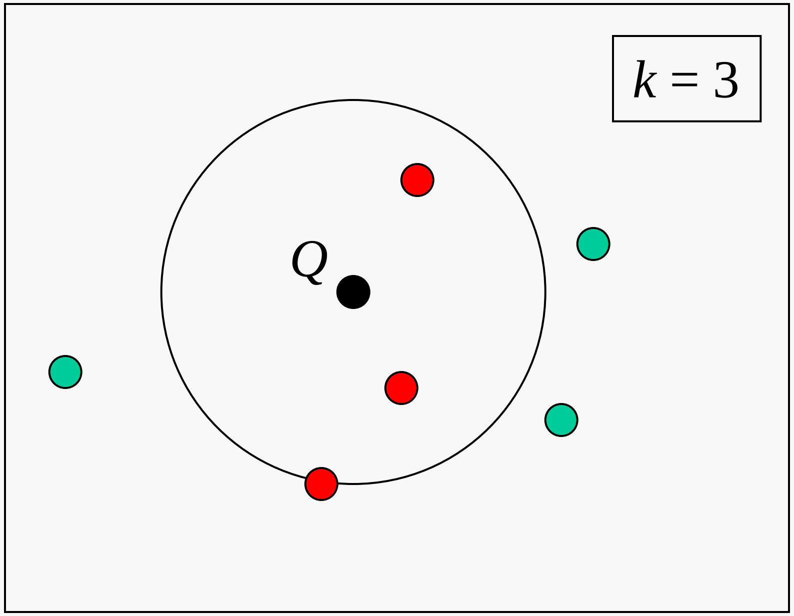
- A range query is defined by a query object *Q* and a distance *r* and the answer is composed of all objects *X* satisfying *S*(*Q*,*X*) <= *r*.
- A *k*-nearest-neighbor query is defined by an object *Q* and an integer *k* and the answer is composed of the *k* objects that are closer to *Q* than any other object.

MIR Fundamentals

Similarity queries in 2-D Euclidean space



range query



k-NN query

MIR Fundamentals

Given a collection of multimedia objects, the ranking function *S*( ), the type of query (range or *k*-NN) and the query object *Q*, the brute-force method to answer the query is:

Brute-Force Query Processing

[Step1] Select the next object *X* from the collection

[Step2] Test if *X* satisfies the query constraints

[Step 3] If YES then report *X* as part of the answer

[Step 4] GOTO Step 1

MIR Fundamentals

Problems with the brute-force method

- The whole collection is being accessed, increasing computational as well as I/O costs.
- The complexity of the processing algorithm is independent of the query (i.e., O(*n*) objects will be scanned).
- The calculation of the function *S*( ) is usually time consuming and *S*( ) is evaluated for ALL objects, the overall running time increases.
- Objects are being processed in their raw form without any intermediate representation. Since multimedia objects are usually large in size, memory problems arise.

MIR Fundamentals

Multimedia objects are rich in content. To enable efficient query processing, objects are usually transformed to another more convenient representation.

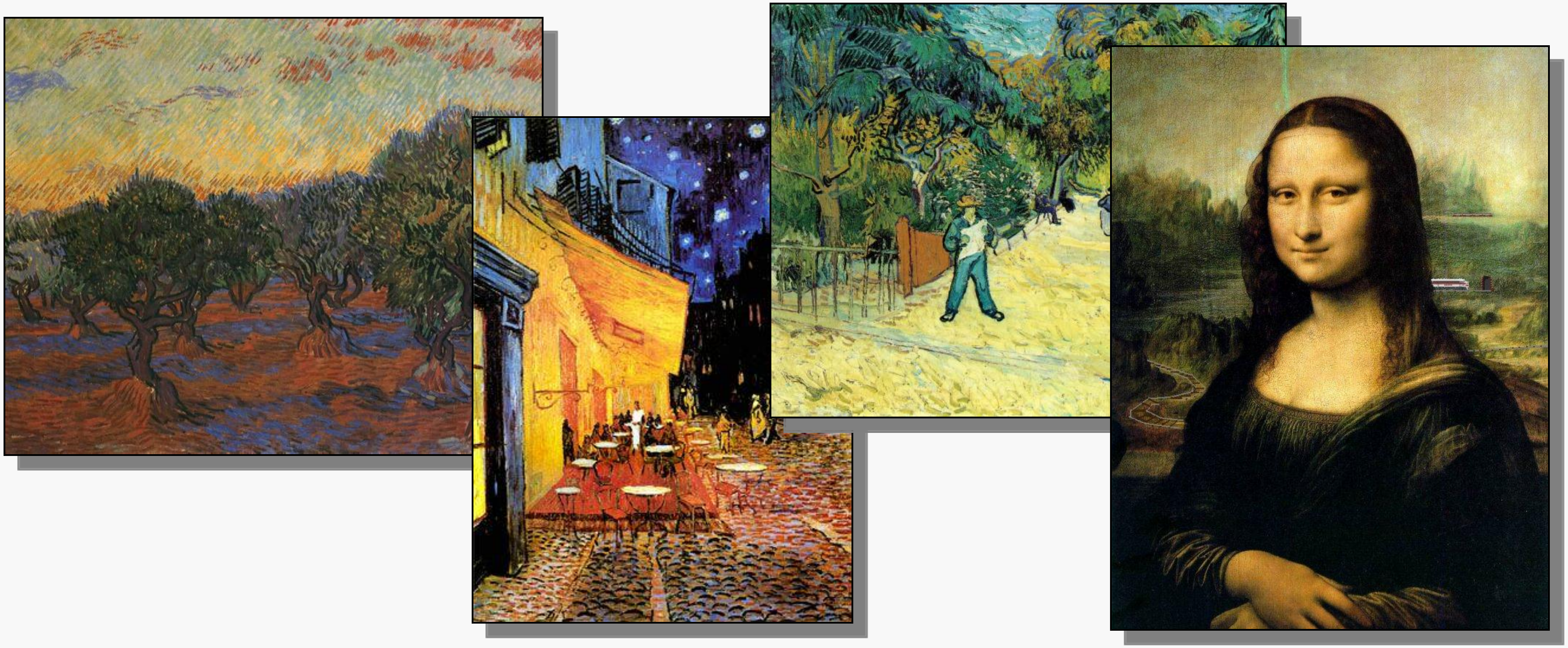
Each object *X* in the original collection is transformed to another object *T*(*X*) which has a simpler representation than *X*.

The transformation used depends on the type of multimedia objects. Therefore, different transformations are used for images, audio files and videos.

The transformation process is related to feature extraction. Features are important object characteristics that have large discriminating power (can differentiate one object from another).

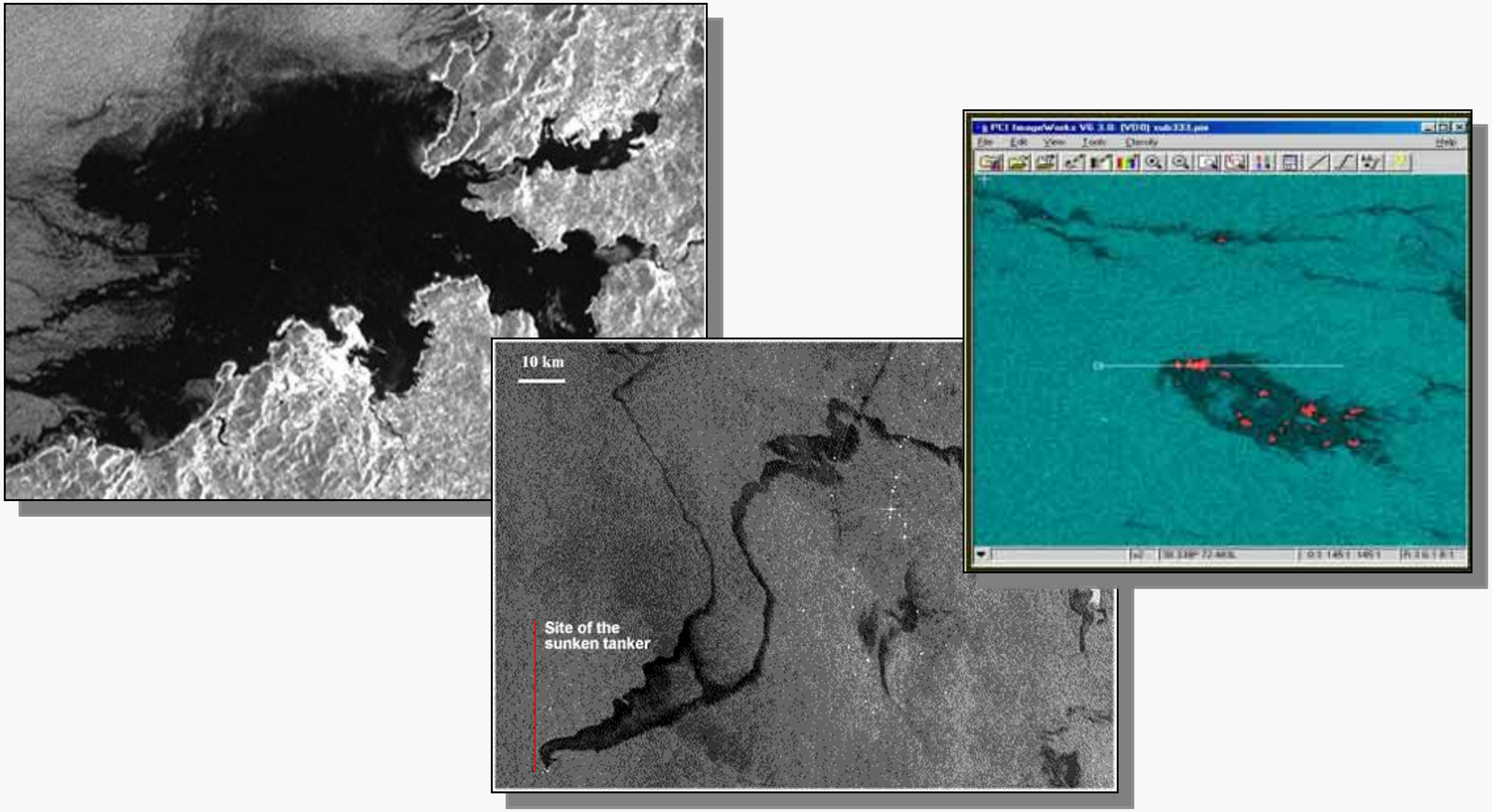
MIR Fundamentals

Image Retrieval: paintings could be searched by artists, genre, style, color etc.



MIR Fundamentals

Satellite images – for analysis/prediction



MIR Fundamentals

Audio Retrieval by content: e.g, music information retrieval.

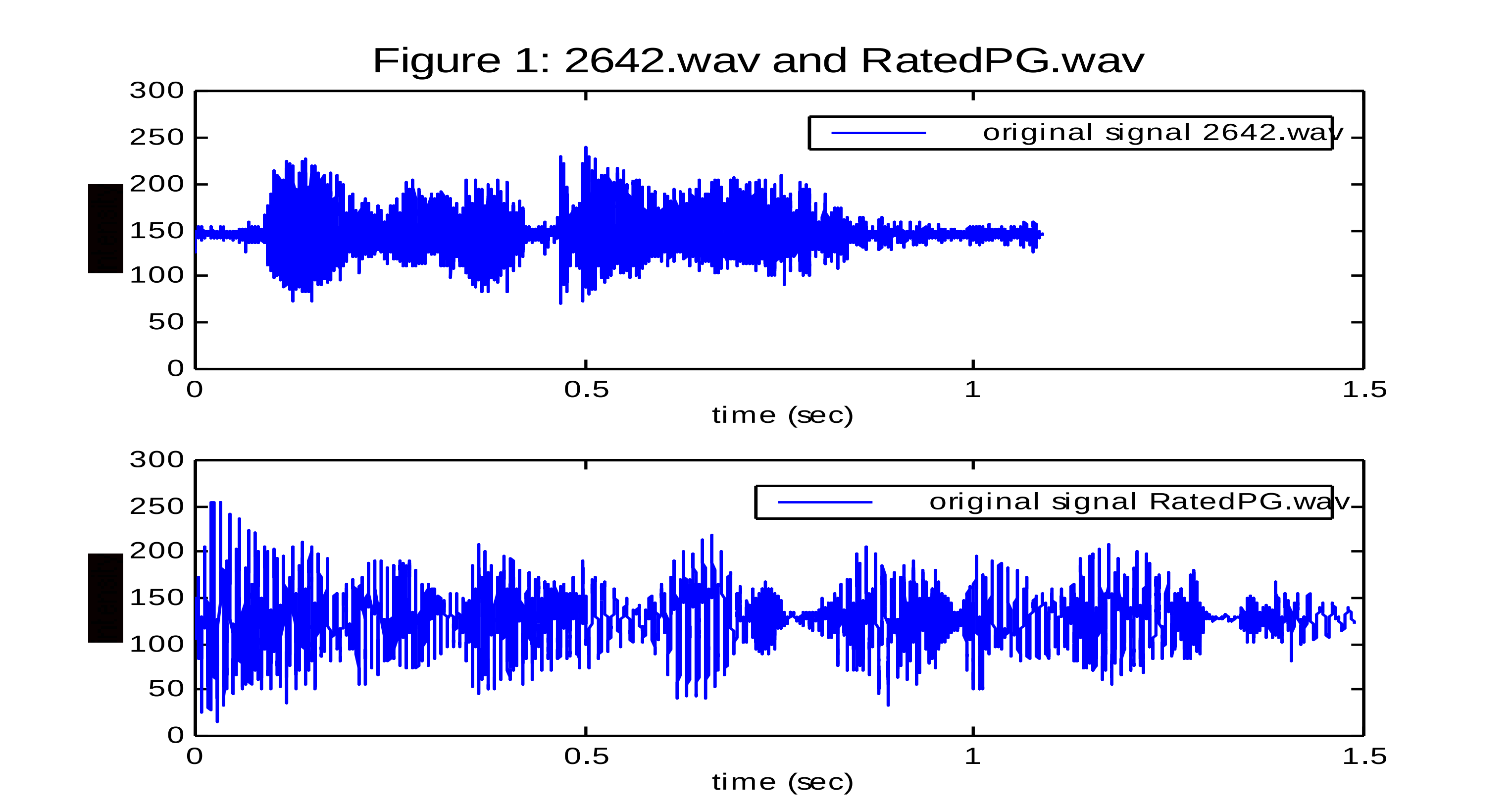


Figure 1: 2642.wav and RatedPG.wav

MIR Fundamentals

Each multimedia object (text,image,audio,video) is represented as a point (or set of points) in a multidimensional space.

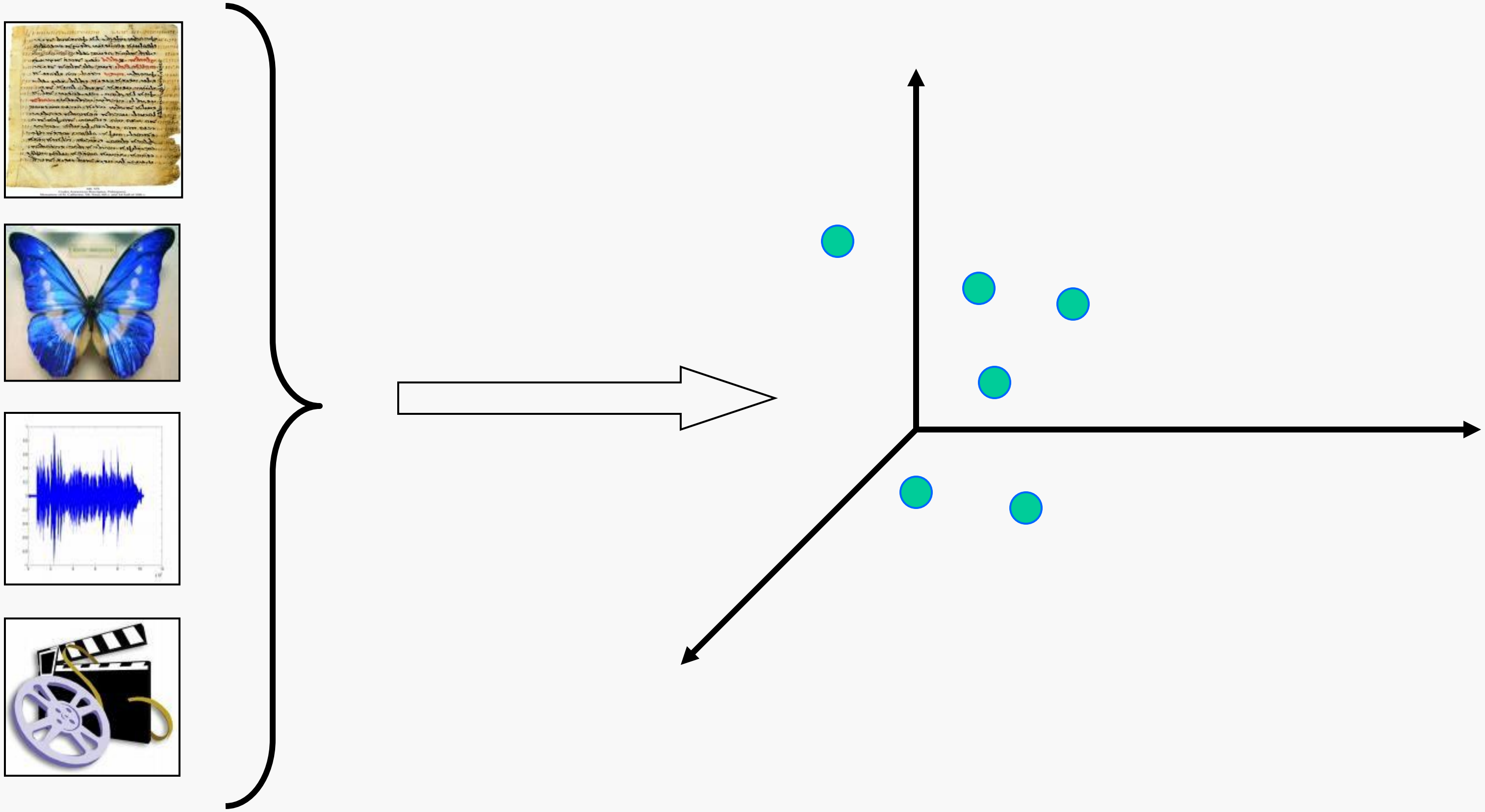
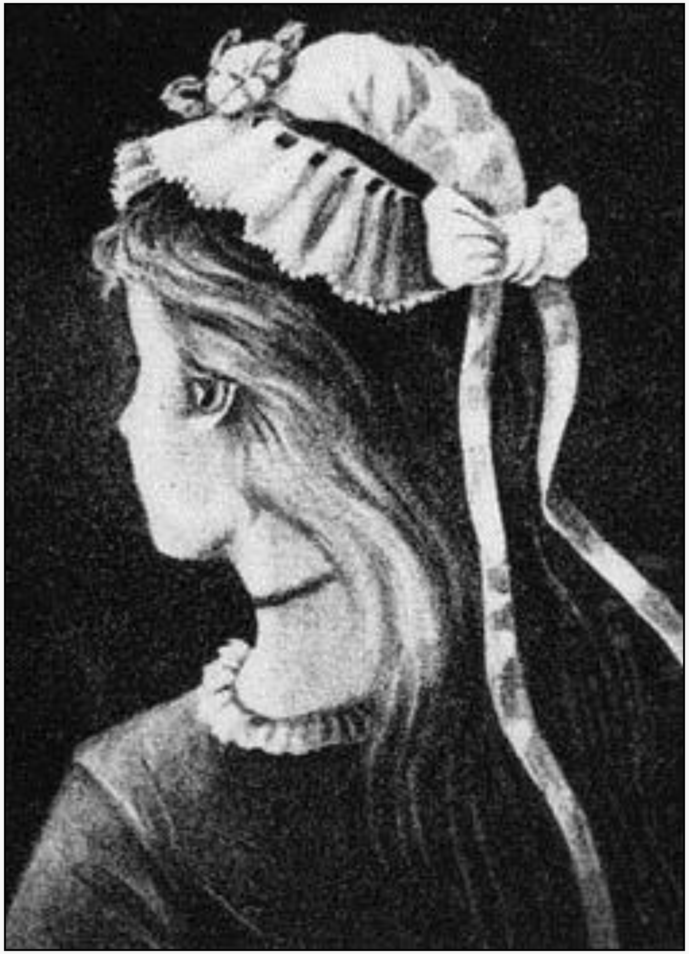


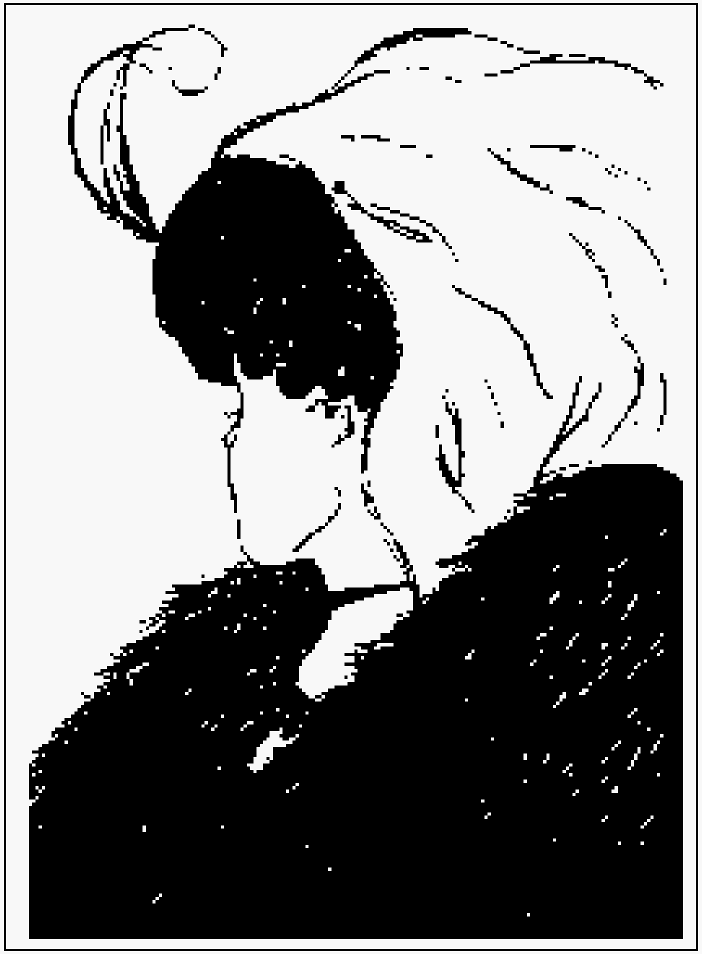
Image Retrieval Using Annotations

- Problem of image annotation
  - Large volumes of databases
  - Valid only for one language – with image retrieval this limitation should not exist
- Problem of human perception
  - Subjectivity of human perception
  - Too much responsibility on the end-user
- Problem of deeper (abstract) needs
  - Queries that cannot be described at all, but tap into the visual features of images.

Image Retrieval Using Annotations

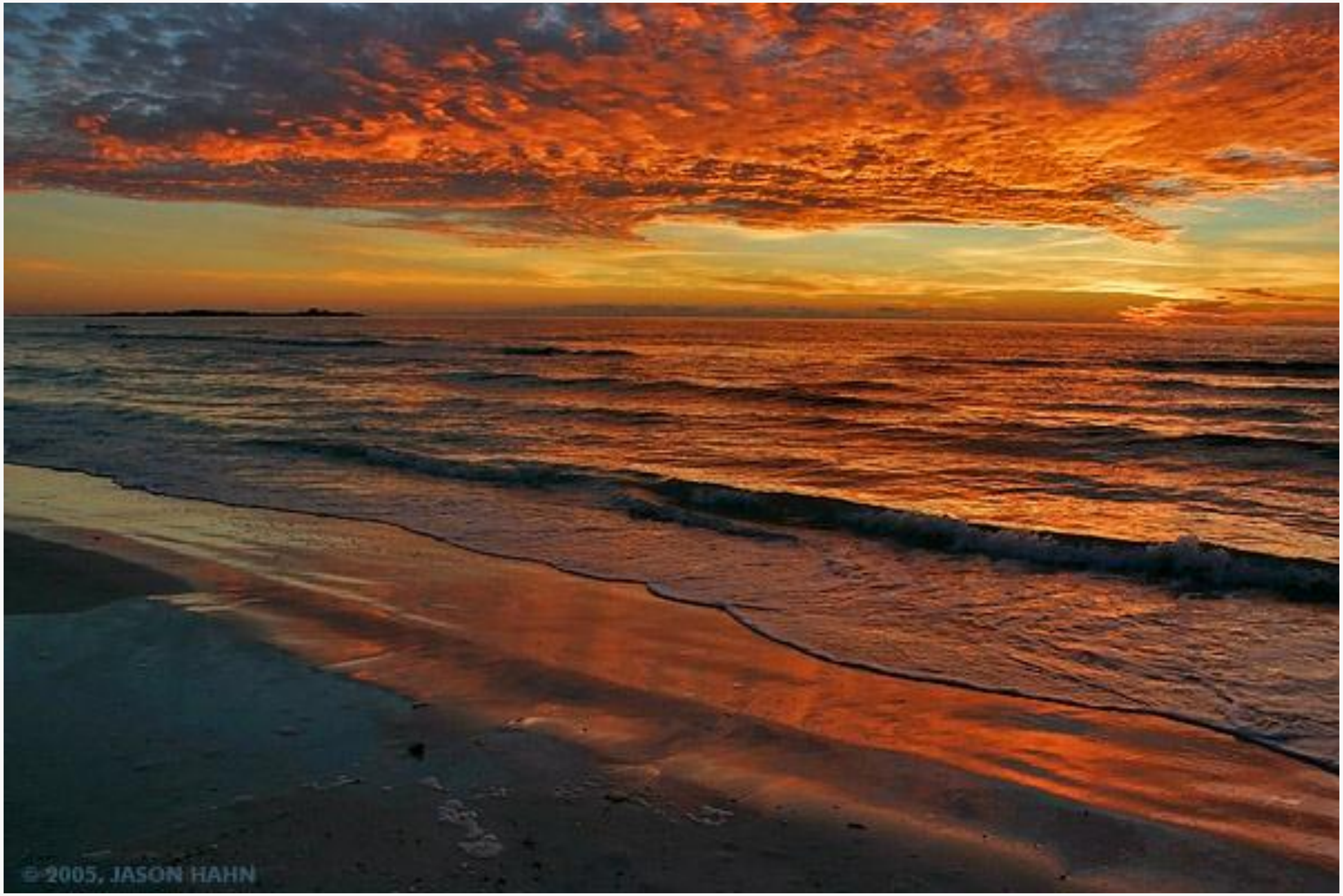


young lady
or
old lady
?



Can you annotate these images?

Image Retrieval Using Annotations



Can you annotate this image?

Image Retrieval By Content

There are three important image characteristics:

- Color
- Texture
- Shape

Although each can be used by itself, two images that have similar colors, similar textures and depict similar shapes, are considered similar.

The determination of similar images is called Image Retrieval By Content.

Image Retrieval By Content

Major modules

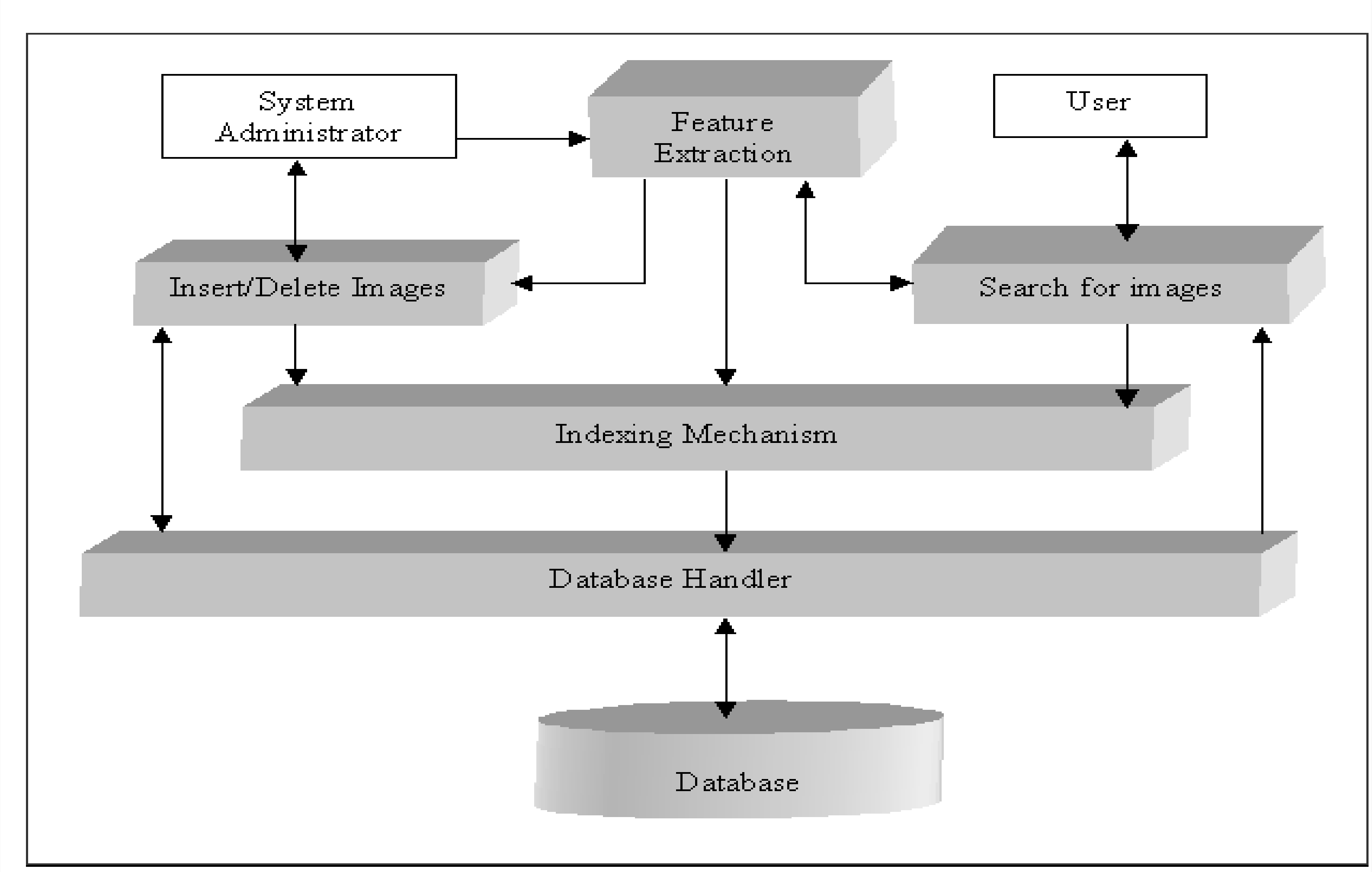


Image Retrieval By Content

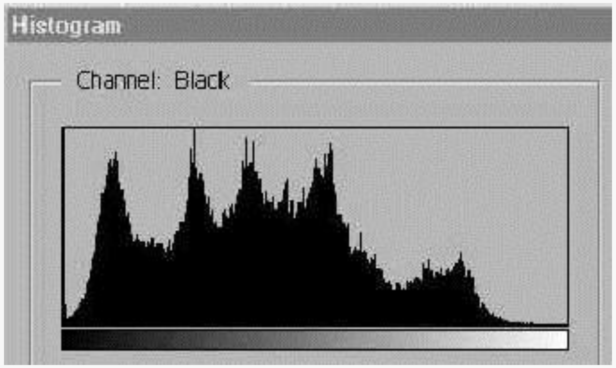
Some systems

- QBIC, IBM Almaden, 1993.
- CANDID, Los Alamos National Laboratory, 1995.
- Berkeley Digital Library Project, University of California @ Berkeley, 1996.
- FOCUS, University of Massachusetts @ Amherst, 1997.
- MARS, University of Illinois @ Urbana-Champaign, 1997.
- Blobworld, University of California @ Berkeley, 1999.
- C-Bird, Simor Fraser University, 1998.

Color Features

- Examining images based on the colors they contain is one of the most widely used techniques because it does not depend on image size or orientation.
- Color searches will usually involve comparing color histograms, though this is not the only technique in practice.

Image



Corresponding histogram

Color Features

- The histogram of an image counts the number of pixels containing a specific color. For true-color images the RGB space is mapped to 1-D space.
- To avoid a large number of colors (>16M) quantization is used.
- The dissimilarity between a query image *Q* and a DB image *X* corresponds to the distance between the histograms H_Q and H_X.

Color Features

The distance between two histograms can be calculated by various distance measures such as:

Histogram Intersection

$$\boxed{H(H_Q,H_X)=\frac{\sum_i \min\left(H_Q(i),H_X(i)\right)}{\min\left(\sum_i H_Q(i),\sum_i H_X(i)\right)}}$$

Euclidean Distance (L2)

$$\boxed{L_2(H_Q,H_X)=\sqrt{\sum_i \left(H_Q(i)-H_X(i)\right)^2}}$$

Bhattacharyya Distance

$$\boxed{B(H_Q,H_X)=-\ln\sum_i \sqrt{H_Q(i)\times H_X(i)}}$$

Matusita Distance

$$\boxed{M(H_Q,H_X)=\sqrt{\sum_i \left(\sqrt{H_Q(i)}-\sqrt{H_X(i)}\right)^2}}$$

Divergence

$$\boxed{D(H_Q,H_X)=\sum_i \left[\left(H_Q(i)-H_X(i)\right)\ln\frac{H_Q(i)}{H_X(i)}\right]}$$

Note: *H*Q(*i*) is the *i*-th histogram position.

Color Features

The previous measures assume that two colors are independent. Therefore, small color differences may increase the dissimilarity of two images. To tackle this problem the Quadratic-Form distance functions have been proposed.

$$D(Q,X) = (H_Q - H_X)^T \underset{\substack{\big\uparrow \\ \text{color similarity matrix}}}{A} (H_Q - H_X)$$

Texture Features

Texture measures look for visual patterns in images and how they are spatially defined. Textures are represented by texels which are then placed into a number of sets, depending on how many textures are detected in the image. These sets not only define the texture, but also the texture location in the image.

Examples of different textures

Texture Features

- Texture – innate property of all surfaces
  - Clouds, trees, bricks, hair etc…
- Refers to visual patterns of homogeneity.
- Does not result from presence of a single color.
- Most accepted classification of textures based on psychology studies – Tamura representation:

Coarseness
Contrast
Directionality
} most important

Line-likeness
Regularity
Roughness

Texture Features

The most commonly used texture representation is the texture histogram or texture description vector, which is a vector of numbers that summarizes the texture in a given image or image region.

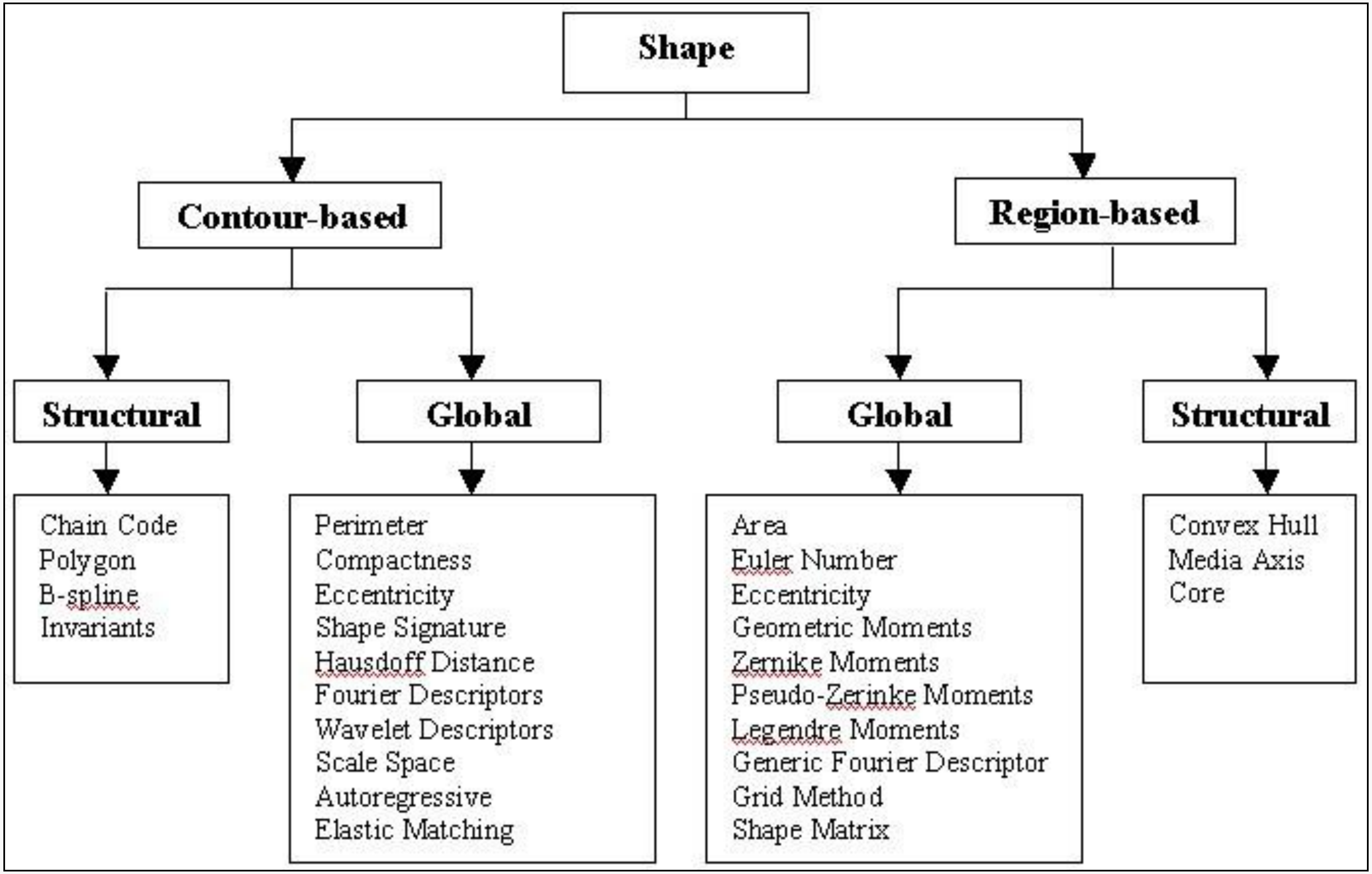
Usually, the first three texture features (i.e., Coarseness, Contrast, Directionality) are used to generate the texture histogram.

Texture dissimilarity is calculated by applying an appropriate distance function (e.g., divergence).

Shape Features

- Shape does not refer to the shape of an image but to the shape of a particular region.
- Shapes will often be determined by first applying segmentation or edge detection to an image.
- In some cases accurate shape detection will require human intervention, but this should be avoided.
- Popular methods: projection matching, boundary matching.

Shape Features



Shape

Contour-based

Region-based

Structural

- Chain Code
- Polygon
- B-spline
- Invariants

Global

- Perimeter
- Compactness
- Eccentricity
- Shape Signature
- Hausdoff Distance
- Fourier Descriptors
- Wavelet Descriptors
- Scale Space
- Autoregressive
- Elastic Matching

Global

- Area
- Euler Number
- Eccentricity
- Geometric Moments
- Zernike Moments
- Pseudo-Zerinke Moments
- Legendre Moments
- Generic Fourier Descriptor
- Grid Method
- Shape Matrix

Structural

- Convex Hull
- Media Axis
- Core

Shape Features

- Boundary Matching algorithms require the extraction and representation of the boundaries of the query shape and image shape.
- The boundary can be represented as a sequence of pixels or maybe approximated by a polygon.
- For a sequence of pixels, one classical matching technique uses Fourier descriptors to compare two shapes.

Shape Features

From this sequence of points a sequence of unit vectors and a sequence of cumulative differences can be computed

$$v_k = \frac{V_{k+1} - V_k}{\left|V_{k+1} - V_k\right|}$$

unit vectors

$$\begin{aligned} l_k &= \sum_{i=1}^{k} \left|V_i - V_{i-1}\right|; \quad k > 0 \\ l_0 &= 0 \end{aligned}$$

cumulative differences

Shape Features

- ➢The Fourier descriptors {a_{-M}, …, a_0, …,a_M} are then approximated by
- ➢These descriptors can be used to define a shape distance measure.

$$a_n = \frac{1}{L\left(\frac{n2\pi}{L}\right)^2} \sum_{k=1}^{m} (v_{k-1} - v_k) e^{-jn(2\pi/L)l_k}$$

Image Similarity Computation

- From color, texture and shape features a single vector V_X can be produced for image *X*, and a single function S(V_Q,V_X) is used to compute similarity.
- Alternatively, each image *X* is scored separately for color, texture and shape similarity with respect to the query *Q*. Partial scores are combined to compute the final score.

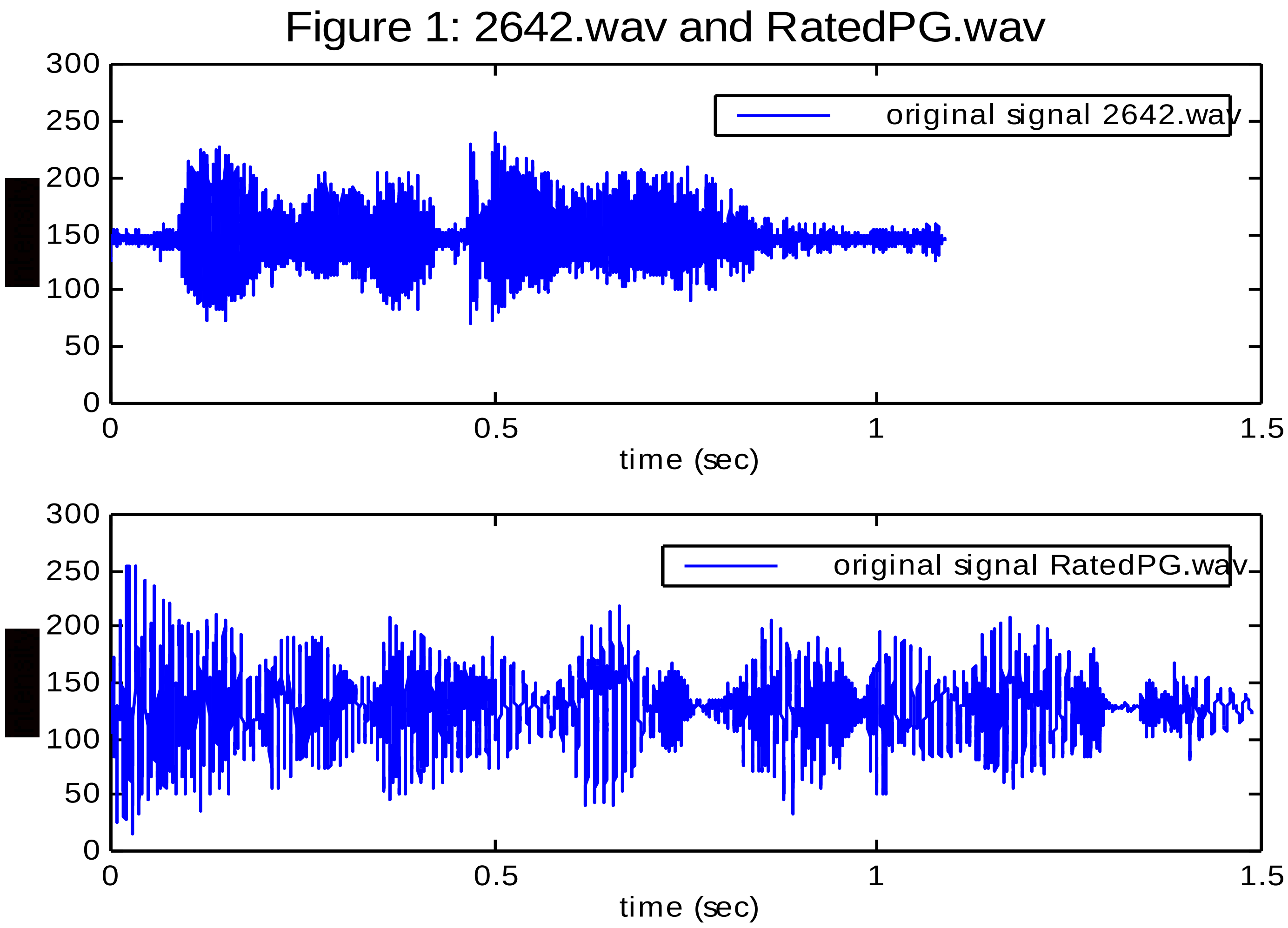
Potential Problem: feature vectors may be large (many dimensions) causing problems in indexing.

Audio Retrieval By Content

Deals with the retrieval of similar audio pieces (e.g., music)

- A feature vector is constructed by extracting acoustic and subjective features from the audio in the database.
- The same features are extracted from the queries.
- The relevant audio in the database is ranked according to the feature match between the query and the database.

Audio Retrieval By Content



Audio Retrieval By Content

As in the image case, characteristic features should be extracted from audio files. Audio features are separated in two categories:

- Acoustic features (objective)
- Subjective/Semantic features

Acoustic Features

Acoustic features describe an audio in terms of commonly understood acoustical characteristics, and can be computed directly from the audio file

- Loudness
- Spectrum Powers
- Brightness
- Bandwidth
- Pitch
- Cepstrum

Acoustic Features

- Loudness is approximated by the square root of the energy of the signal computed from the Short-Time Fourier Transform (STFT), in decibels.
- Spectrum Powers include total spectrum power and sub-band powers. They are all represented with logarithmic forms.
- Brightness is computed as the centroid of the Short-Time Fourier Transform (STFT) and is stored as a log frequency .
- Bandwidth is computed as the power-weighted average of the difference between the spectral components and the centroid.

Acoustic Features

- Pitch is the fundamental period of a human speech waveform, and is an important parameter in the analysis and synthesis of speech signals. However we can still use pitch as a low-level feature to characterize changes in the periodicity of waveforms in different audio signals
- The Cepstrum has proven to be highly effective in automatic speech recognition and in modeling the subjective pitch and frequency content of audio signals. It can be illustrated by use of the Mel-Frequency Cepstral Coefficients, which are computed from FFT power coefficients.

Subjective/Semantic Features

- Subjective features describe sounds using personal descriptive language. The system must be trained to understand the meaning of these descriptive terms.
- Semantic features are high-level features that are summarized from the low-level features. Compared with low-level features, they are more accurate to reflect the characteristics of audio content.

Subjective/Semantic Features

Major features:

- Timbre: it is determined by the harmonic profile of the sound source. It is also called tone color.
- Rhythm: represents changes in patterns of timbre and energy in a sound clip.
- Events: are typically characterized by a starting time, a duration, and a series of parameters such as pitch, loudness, articulation, vibrato, etc.
- Instruments: instrument identification can be accomplished using the histogram classification system. This requires that the system has been trained on all possible instruments.

Audio Similarity Computation

Given a query audio file *Q* the retrieval process is applied to the feature vectors generated by the audio signals. As in the image case, we need a similarity function S(*Q*,*X*) to compute the similarity between *Q* and any audio file *X*.

To enable sub-pattern similarity (e.g., find audio files containing a particular audio piece) additional processing is required (e.g., segmentation).

Video Retrieval By Content

Video is the most demanding and challenging multimedia type containing:

- Text (e.g., subtitles)
- Audio (e.g., music, speech)
- Images (frames)

All these change over time!

Video Retrieval By Content

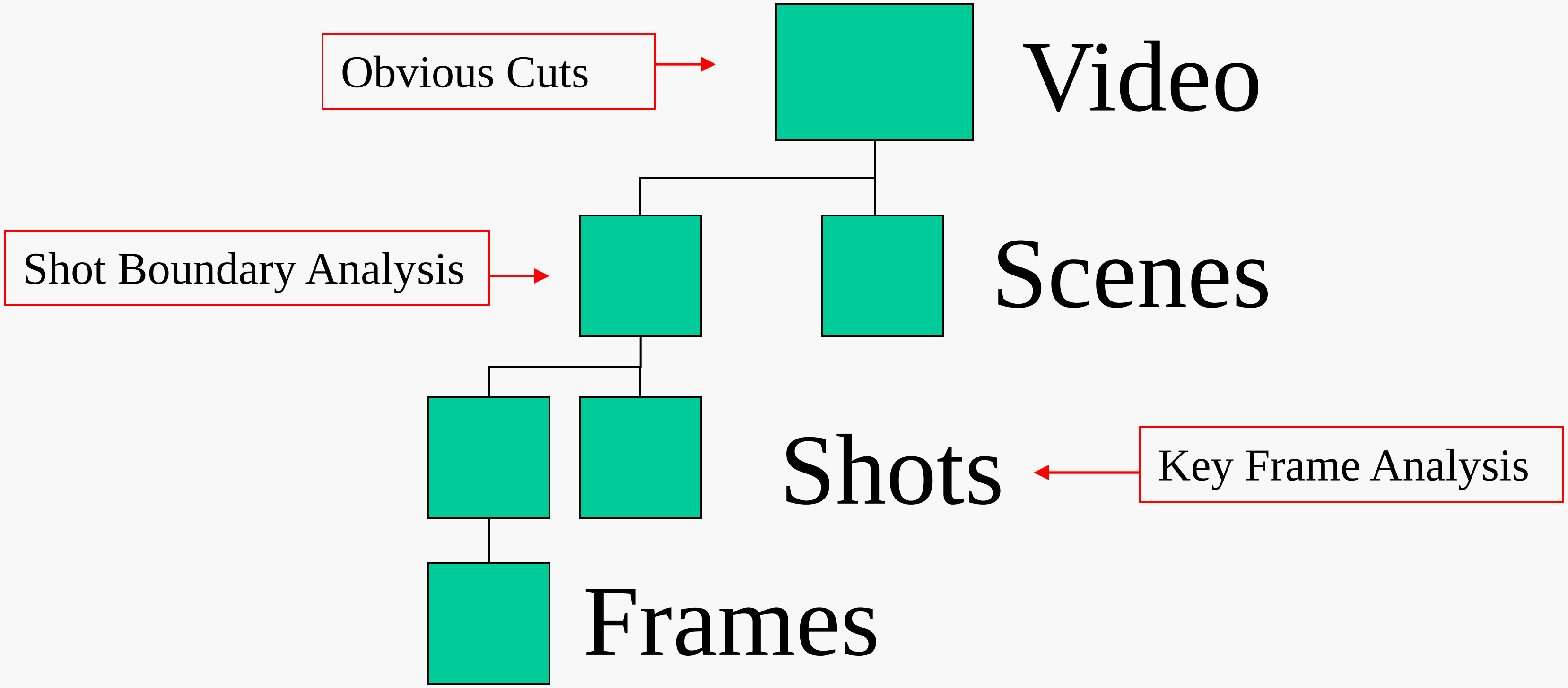
- There is an amazing growth in the amount of digital video data in recent years.
- Lack of tools for classify and retrieve video content.
- There exists a gap between low-level features and high-level semantic content.
- To let machine understand video is important and challenging.

Video Features

Decomposition

- Scene: single dramatic event taken by a small number of related cameras.
- Shot: A sequence taken by a single camera.
- Frame: A still image.

Video Features



Obvious Cuts

Video

Shot Boundary Analysis

Scenes

Shots

Key Frame Analysis

Frames

Shot Detection

- A shot is a contiguous recording of one or more video frames depicting a contiguous action in time and space.
- During a shot, the camera may remain fixed, or may exhibit such motions as panning, tilting, zooming, tracking, etc.
- Segmentation is a process for dividing a video sequence into shots.
- Consecutive frames on either side of a camera break generally display a significant quantitative change in content.
- We need a suitable quantitative measure that captures the difference between two frames.

Shot Detection

- Use of pixel differences: tend to be very sensitive to camera motion and minor illumination changes.
- Global histogram comparisons: produce relatively accurate results compared to others.
- Local histogram comparisons: produce the most accurate results compared to others.
- Use of motion vectors: produce more false positives than histogram-based methods.
- Use of the DCT coefficients from MPEG files: produce more false positives than histogram-based methods.

Shot Representation

Three major ways:

- Based on representative frames
- Based on motion information
- Based on objects

We focus on the first one.

Representative Frames

- The most common way of creating a shot index is to use a representative frame to represent each shot.
- Features of this frame are extracted and indexed based on color, shape, texture (as in image retrieval).
- If shots are quite static, any frame within the shot can be used as a representative.
- Otherwise, more effective methods should be used to select the representative frame.
- Two issues: (i) how many frames must be selected from each shot and (ii) how to select these frames.

Representative Frames

How many frames per shot? Three methods:

- One frame per shot. The method does not consider the length and content changes.
- The number of selected representatives depends on the length of the video shot. Content is not handled properly.
- Divide shots into subshots and select one representative frame from each subshot. Length and content are taken into account.

Representative Frames

How are the frames selected?

- Select the first frame from each segment (shot or subshot)
- An average frame is defined so that each pixel in this frame is the average of pixel values at the same grid point in all the frames of the segment. Then, the frame within the segment that is most similar to the average frame is selected as the representative.
- The histograms of all the frames in the segment are averaged. The frame whose histogram is closest to this average histogram is selected as the representative frame of the segment.

Representation of Multimedia Objects

Each multimedia object (text,image,audio,video) is represented as a point (or set of points) in a multidimensional space.