Multimedia Systems

- Indexing and retrieval of still images

Dr. Yi-Zhe Song

EBU706U

Today's agenda

- Introduction
  - Motivations
  - Challenge
  - Current systems and problems
- Colour-based retrieval ✔
- Shape-based retrieval ✔
- Texture-based retrieval ✔
- Examples

EBU706U

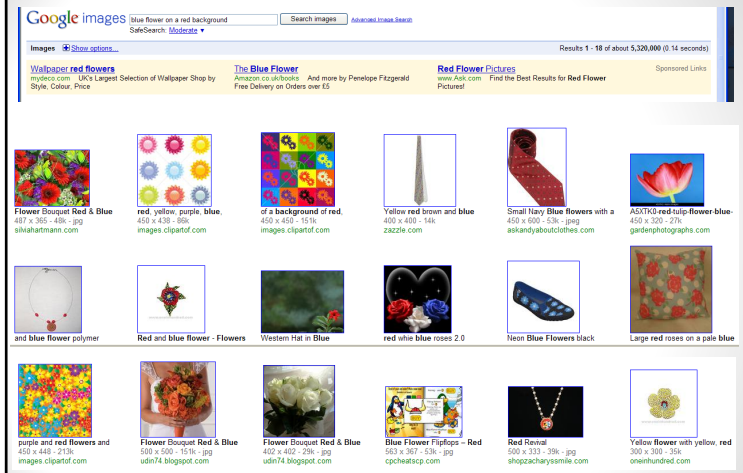
Motivations

- Increasing availability of multimedia information
- More and more people wanting to find specific multimedia information

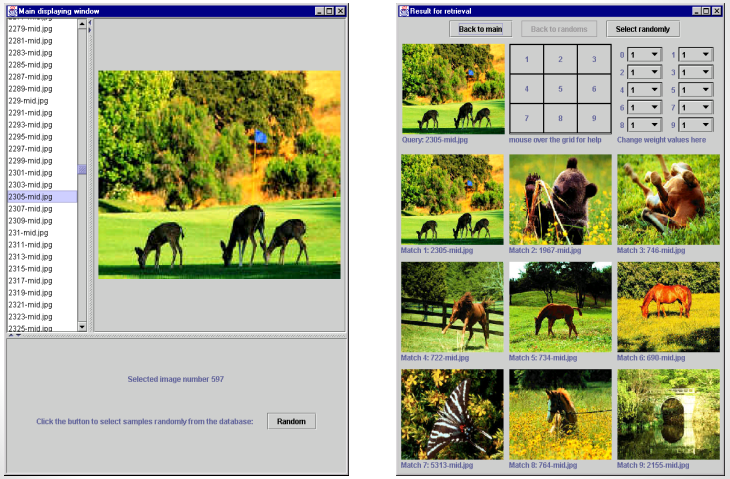
- Difficult to get what you need:
  - searching on the Web
  - selecting one of the many TV programmes
  - browsing in your own home database

- Systems for searching exist on the web, but:
  - mostly for textual information
  - difficult to use for 'non-engineers'

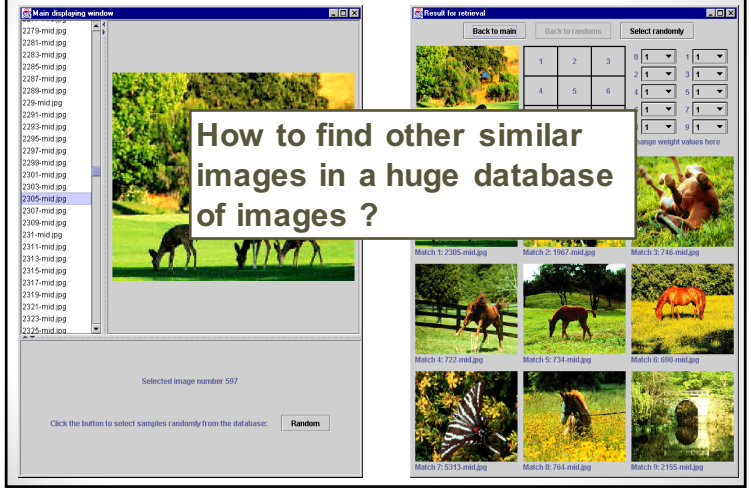
EBU706U



Content-based Image Retrieval



The challenge



Requirements

- User requirements
  - Response times
  - Multimedia support
  - Scope of search
  - Types of queries
  - Semantic-level descriptors
  - Human-computer interaction
  - Relevance feedback

EBU706U

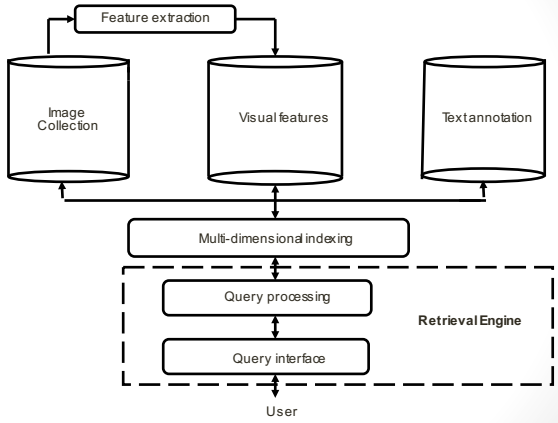
Problems 多看往年案例题

- Problems in Content-Based Image Retrieval (CBIR)
  - computational complexity increases with the number of features used
  - scalability: retrieval time depend on the database size
  - access latency
  - inefficiency in indexing and searching
  - unrealistic assumptions needed in some systems
  - gap between low-level and semantic descriptors

EBU706U

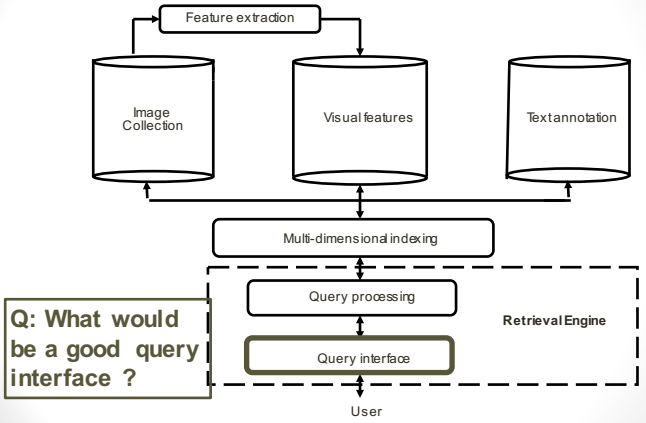
examable...

Architecture of a CBIR system



EBU706U

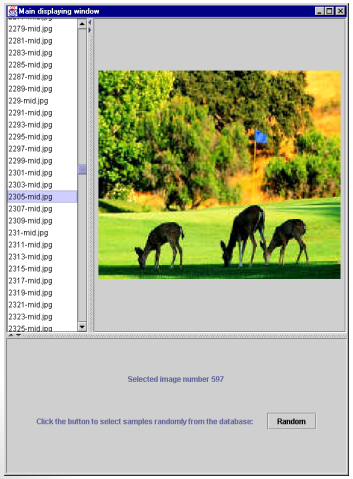
Architecture of a CBIR system

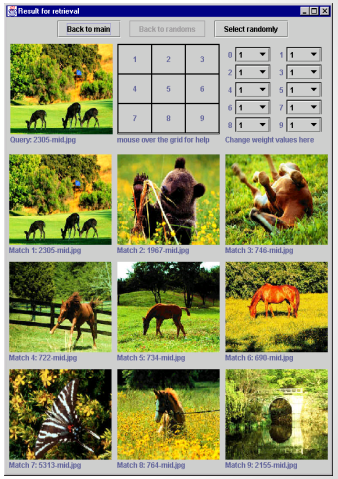


Q: What would be a good query interface ?

EBU706U

Example ..





Today's agenda

- Introduction
  - Motivations
  - Challenge
  - Current systems and problems
- Colour-based retrieval
- Shape-based retrieval
- Texture-based retrieval
- Examples

EBU706U

Today's agenda

- Introduction
  - Motivations
  - Challenge
  - Current systems and problems
- Colour-based retrieval
- Shape-based retrieval
- Texture-based retrieval
- Examples

EBU706U

scenario:

Q: How to find red cars in a database of car images?

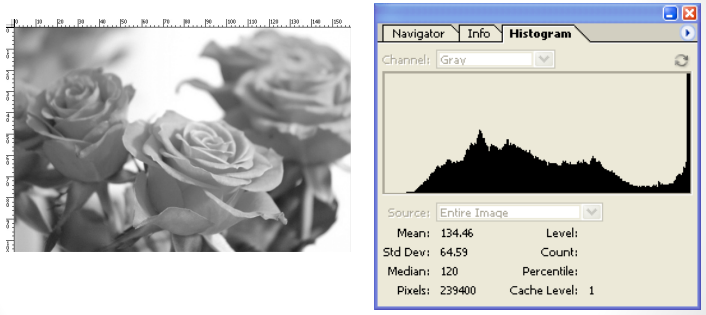
Histograms

one way to encode colours

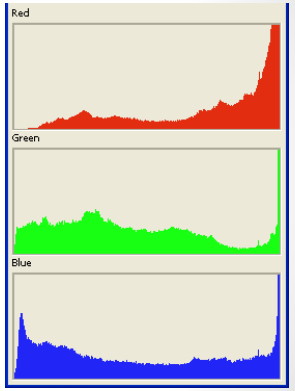
- Definition
  - function that maps the quantization levels into the frequency of each quantization level in the image
  - "counts" how many times each "colour" appears in the whole image
- Grey-scale histogram
  - number of pixels at each grey-level or in a range of grey levels
  - plot of frequencies of grey levels as function of pixel value
  - statistical equivalent to the Probability Density Function (pdf)

EBU706U

Histogram: example



Colour histograms



biggest problem: global

solution: divide the image into local region and histogram in each block

problem again: expensive to construct a feature vector

Histogram: properties

EBU706U

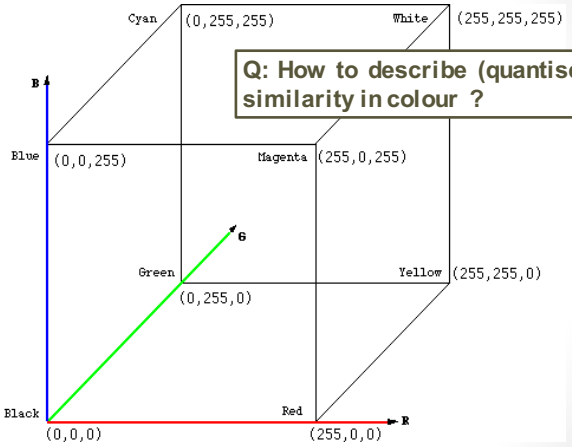
Q: How will the histograms of these two images compare?

identical – histograms do not contain spatial information

Histogram: properties

Q: How about these ones?

EBU706U



Q: How to describe (quantise) similarity in colour ?

EBU706U

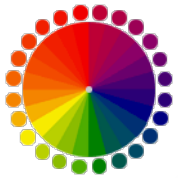
Additive vs. Subtractive

- When painting, an artist has a variety of paints to choose from, and mixed colors are achieved through the **subtractive color** method.
- When a designer is utilizing the computer to generate digital media, colors are achieved with the **additive color** method.

EBU706U

Additive vs. Subtractive

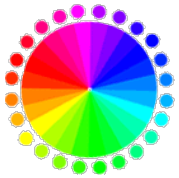
- When we mix colors using paint, or through the printing process, we are using the subtractive color method.
- Subtractive color mixing means that one begins with white and ends with black; as one adds color, the result gets darker and tends to black



EBU706U

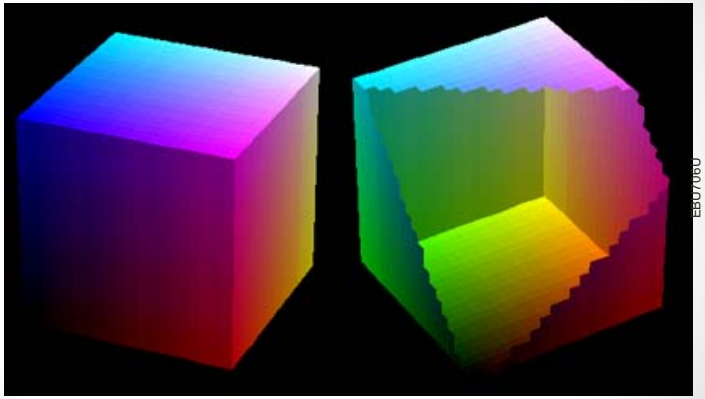
Additive vs. Subtractive

- If we are working on a computer, the colors we see on the screen are created with light using the additive color method.
- Additive color mixing begins with black and ends with white; as more color is added, the result is lighter and tends to white.



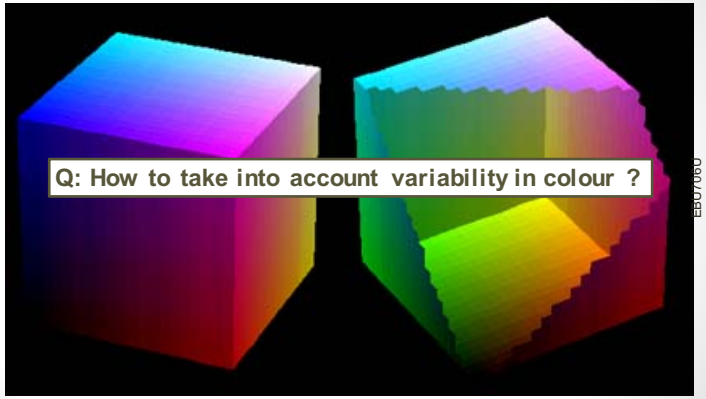
EBU706U

More than 16.7 million different colours !



EBU706U

More than 16.7 million different colours !



EBU706U

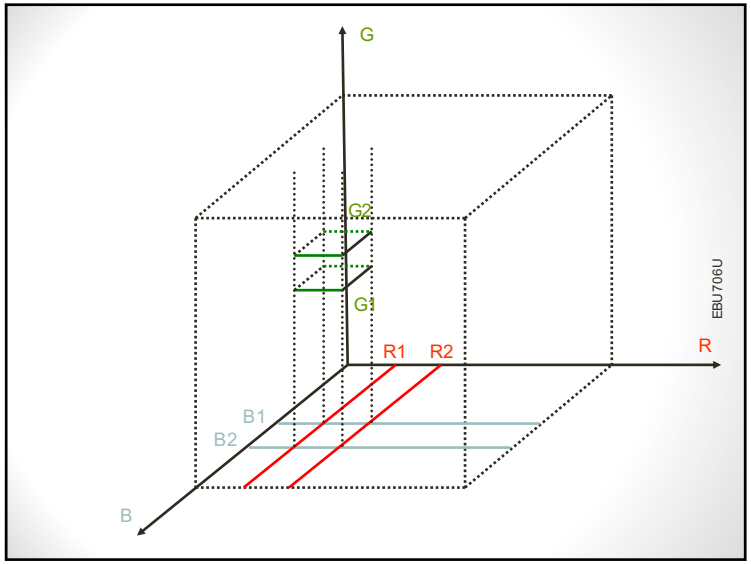
Simple colour clustering

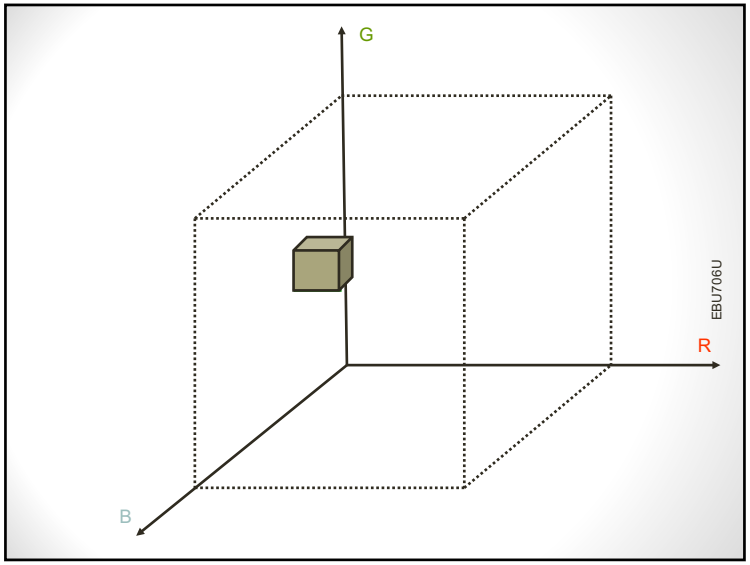
- One object with known colour (R,G,B)
  - Model of the colour

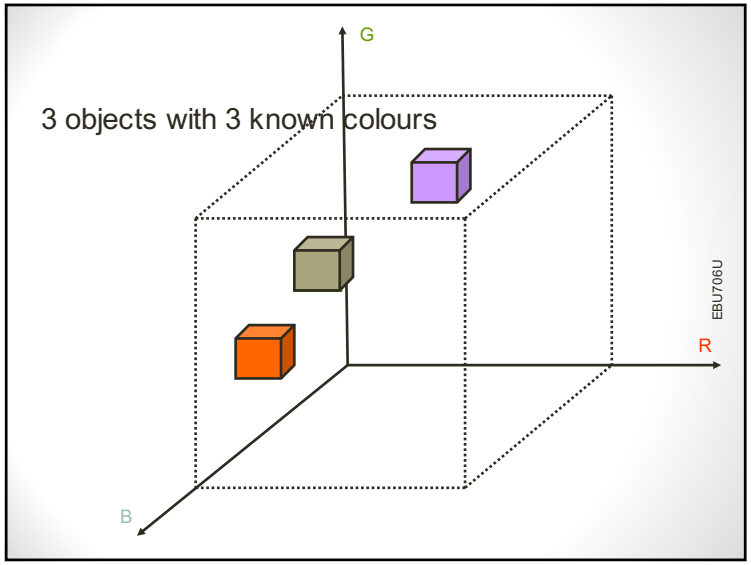
(R,G,B) : R1<R<R2
G1<G<G2
B1<B<B2

Q: How to represent these inequalities in the RGB space?

EBU706U

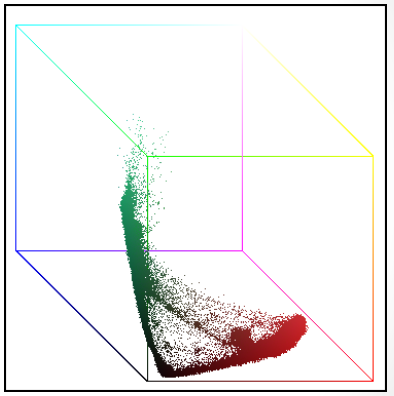




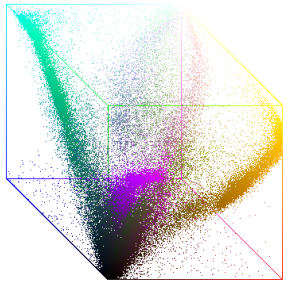


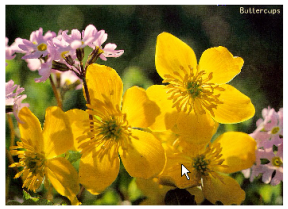
Q: How many colours are there in this picture?

EBU706U



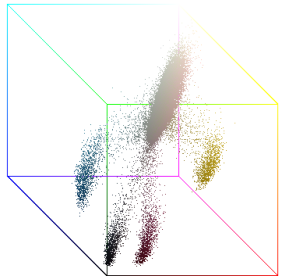
EBU706U

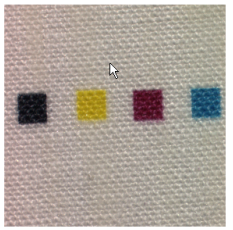




EBU706U

Q: How do you explain this colour distribution?

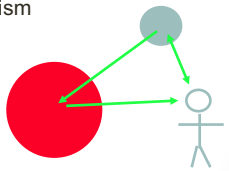




EBU706U

Why does simple colour clustering fail?

- Little colour cube
  - unrealistic colour distribution
- Appearance of a surface: interaction between
  - Illumination
  - Surface reflectance properties
  - Response of a chromatic mechanism (color receptors or color filters)



EBU706U

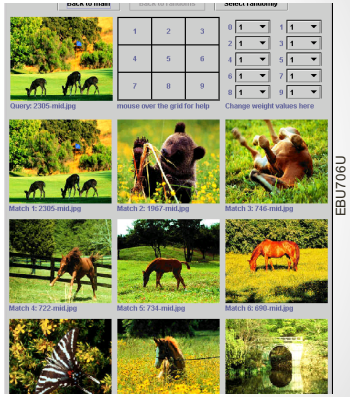
Colour-based approaches

- Conventional
  - colour quantization:
    - requires allocating cells to colours in the selected colour space
    - by dividing the colour space into several clusters containing specific colours
- Content-based
  - Colour bins are defined according to the distribution of dominant colours
  - Histogram modes and relevant local extremes are detected and used to define the quantization intervals

EBU706U

Improving colour-based retrieval

- Use of local histograms
  - local refers to the portion of the image we are interested in
  - alpha value: assign a weighting factor to each "local" histogram
- Use of relevance feedback
  - user can modify weights according to the retrieval result
  - positive and negative



EBU706U

Colour-based retrieval - Summary

- Colour-based retrieval using histogram
  - compute and store colour histogram (feature, *metadata*)
  - compare colour feature between query image and all images in the database
  - include user feedback on retrieval result for better performance
- To use colour feature close to human visual perception
  - e.g., transform from RGB to HSV colour space
- Colour clustering
  - less computation workload
  - human eyes can only discriminate a limited number of colours
  - suitable representation of statistical colour distribution

EBU706U

Today's agenda

- Introduction
  - Motivations
  - Challenge
  - Current systems and problems
- Colour-based retrieval
- Shape-based retrieval
- Texture-based retrieval
- Examples

EBU706U

Today's agenda

- Introduction
  - Motivations
  - Challenge
  - Current systems and problems
- Colour-based retrieval
- Shape-based retrieval
- Texture-based retrieval
- Examples

scenario

E.g. How to find all images that contain a car ?

EBU706U

考试题型：给一组图片 问用哪种retrieval最好

Today's agenda

- Introduction
  - Motivations
  - Challenge
  - Current systems and problems
- Colour-based retrieval
- Shape-based retrieval
- Texture-based retrieval
- Examples

E.g. How to find all images that contain a car ?

Q: How to encode the information about a shape ?

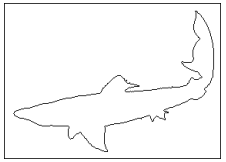
Q: How to calculate the similarity between shapes ?

EBU706U

Shape-based retrieval



contour

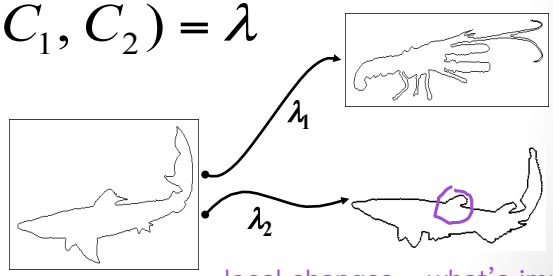


EBU706U

Shape-based retrieval

Definition of a similarity measure (metric) between two given contours C_1 and C_2.

$$D(C_1, C_2) = \lambda$$



local changes – what's important

EBU706U

Metric for contour matching

- The distance between two contours C_1 and C_2 is defined as the distance between their polygonal approximations π_1 and π_2

EBU706U

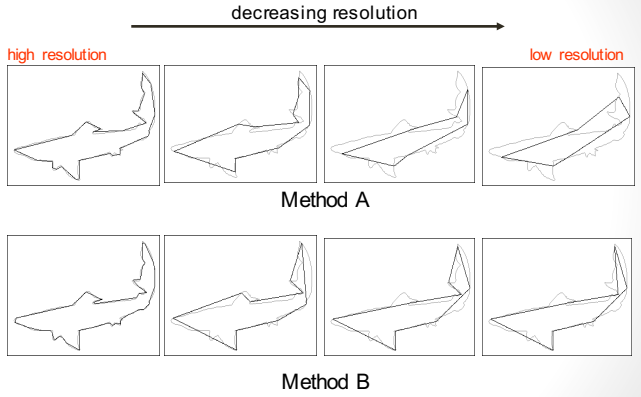
Polygonal approximation

why? reduce complexity

- Feature-driven polygonal approximation
  - The curvature extremes are estimated at several scales
  - An initial polygonal approximation is generated by linking the curvature extremes at lowest resolution with straight lines
  - New polygon vertices are inserted at locations where the approximation error exceeds a given value
  - The final polygon is normalized so that its perimeter is 1

EBU706U

Polygonal approximation



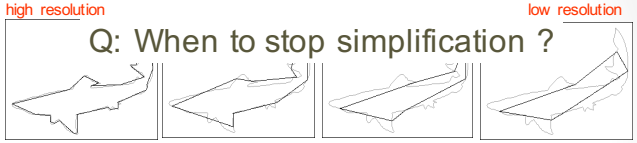
EBU706U

Polygonal approximation

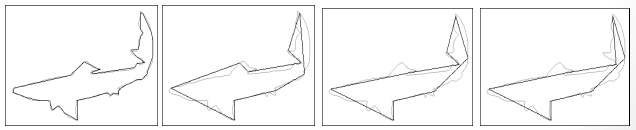
high resolution

low resolution

Q: When to stop simplification ?



Method A



Method B

EBU706U

Polygonal approximation

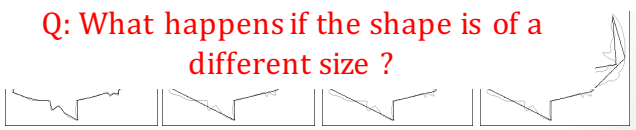
high resolution

low resolution

Q: When to stop simplification ?

Q: What happens if the shape is rotated ?

Q: What happens if the shape is of a different size ?



Method B

Metric for contour matching

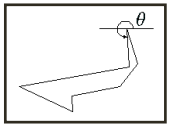
- The distance between two contours C_1 and C_2 is defined as the distance between their polygonal approximations π_1 and π_2
- Each polygonal approximation π_i is represented by its Turning Function
- The distance between π_1 and π_2 is defined as the L_2 norm of the difference between their corresponding turning functions

rotation problem solved

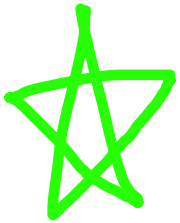
EBU706U

The turning function

- Turning function
  - measures the angle of the counterclockwise tangent as a function of the arclength
  - piecewise constant
  - invariant under translation and scaling
  - a rotation of the polygon corresponds to a vertical shift



EBU706U

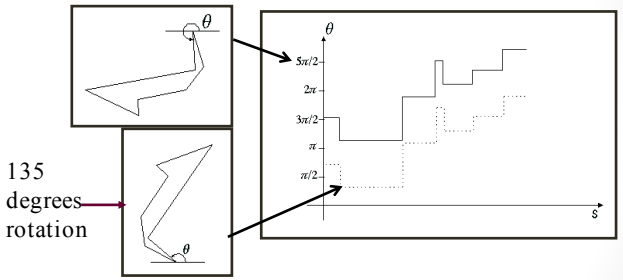


translation–invariant.
because the (x,y) values are based on the object–coordinate, instead of image–coordinate

rotation–invariant

scale–invariant
due to normalisation

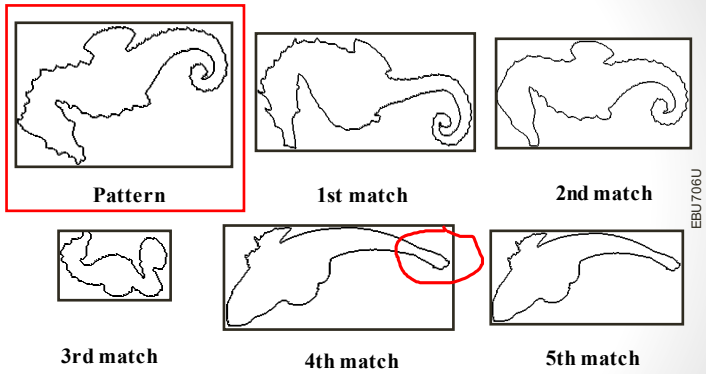
The turning function



starting point: usually the extreme point, then walk counter–clockwise

EBU706U

Example of retrieval



What this reflects:
Differentiation on tails does not work well

EBU706U

Other applications …

- Applications
  - Shape-Based Image Retrieval
  - Pattern Recognition
  - Shape or Contour Compression and Coding
  - Content-Based Coding (MPEG4)
  - Shape/Segmentation Assessment

EBU706U

Today's agenda

- Introduction
  - Motivations
  - Challenge
  - Current systems and problems
- Colour-based retrieval
- Shape-based retrieval
- Texture-based retrieval
- Examples

EBU706U

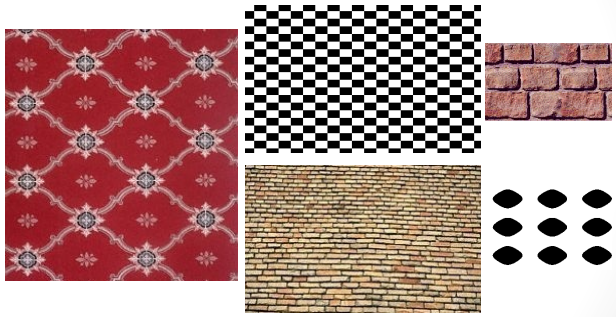
texture – most discriminative

Texture descriptors

- Texture
  - repeating patterns of local variations (textels) in image intensity which are too fine to be distinguished as separate objects
  - provides our visual systems with a huge amount of information
  - textels are made of more than one pixel
- Texture feature
  - powerful discriminating feature
  - refers to the visual patterns that have properties of homogeneity, not resulting from the presence of a single colour or intensity

EBU706U

Example: deterministic textures



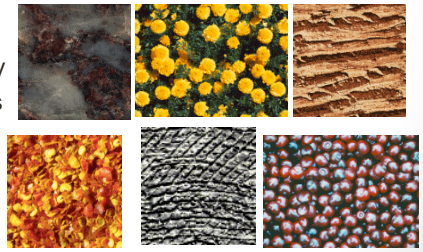
EBU706U

Example: statistical textures

EBU706U

Texture features

- Model proposed by Tamura
  - six visual texture properties
    - coarseness
    - contrast
    - directionality
    - line likeness
    - Regularity
    - roughness



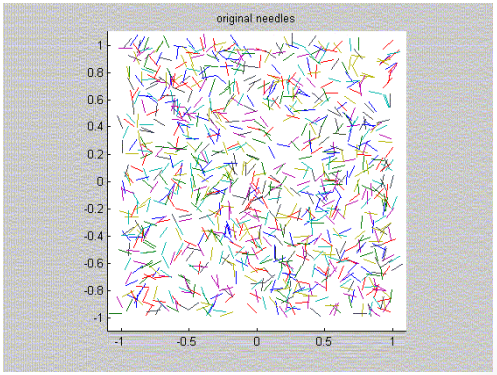
EBU706U

Example: statistical textures

Q: How would you generate grass synthetically?

EBU706U

Original texture/needles



EBU706U

Other descriptors …

- Combination of
  - Colour
  - Shape
  - Texture
  - Motion (video…)

EBU706U

Today's agenda

- Introduction
  - Motivations
  - Challenge
  - Current systems and problems
- Colour-based retrieval
- Shape-based retrieval
- Texture-based retrieval
- Examples

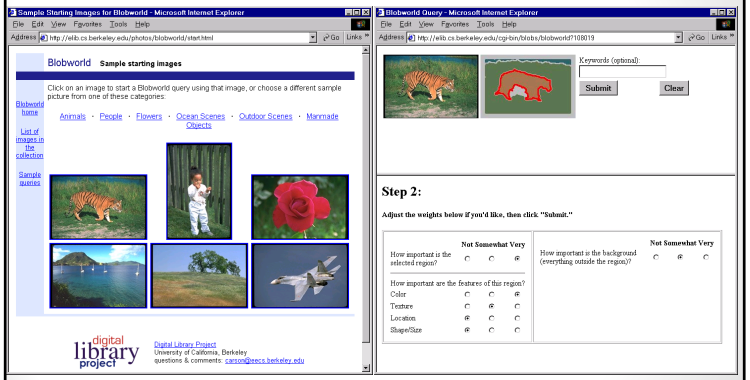
EBU706U

CBIR systems

- QBIC
- Photobook
- Virage
- VisualSEEk
- Netra & Netra-V
- …

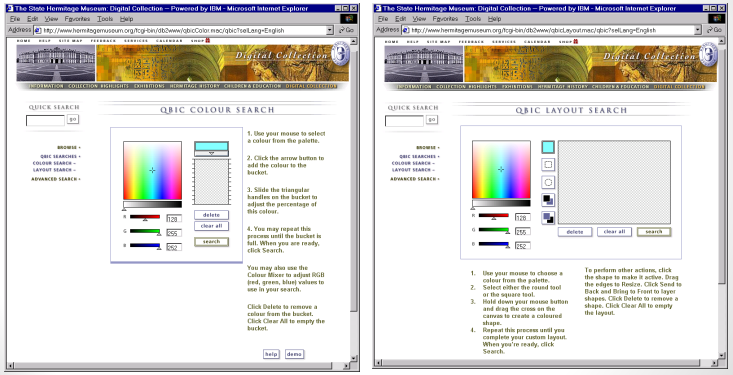
EBU706U

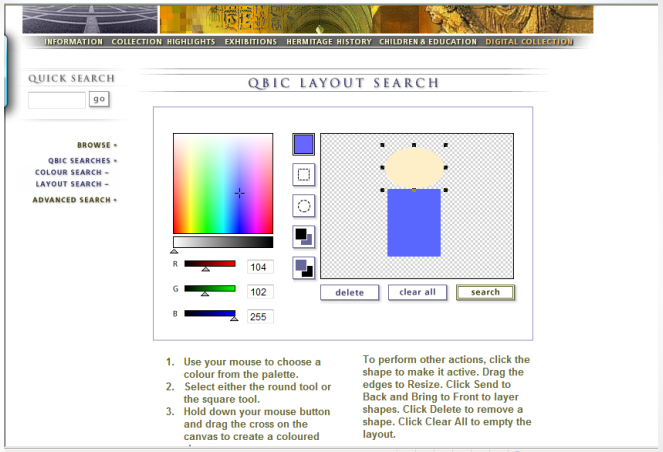
Query by example



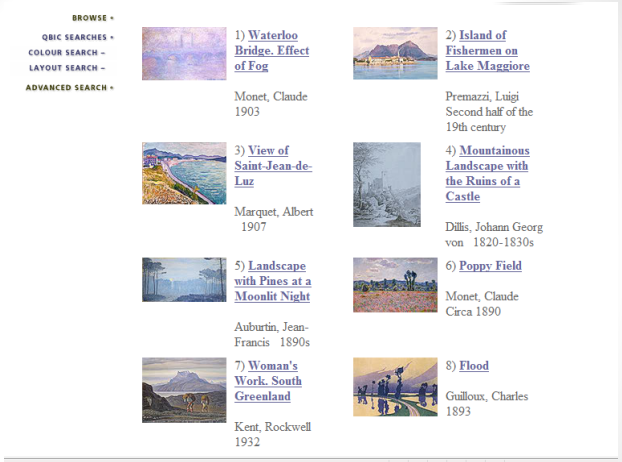
Colour search: QBIC

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





EBU706U



EBU706U

What did we learn today?

- Introduction
  - Motivations
  - Challenge
  - Current systems and problems
- Colour-based retrieval
- Shape-based retrieval
- Texture-based retrieval
- Examples

EBU706U

Past exam question

Describe a shape representation scheme appropriate for content based image retrieval.

EBU706U

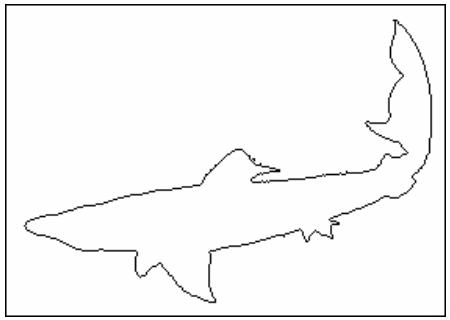
Shape-based retrieval

- The shape of the object can be represented by a polygonal curve.
- To achieve rotation, translation and scale invariance, the polygonal curve is transformed into the tangent space using the turning function. This is the turning angle of the polygonal line.
- Shapes are compared by comparing the turning functions of their polygonal approximations

EBU706U

Past exam question

Plot the turning function of a simplified shape of the shark depicted in Figure 1, starting from the top-right point. Explain how you can obtain a simplified shape and describe the properties of the turning function.



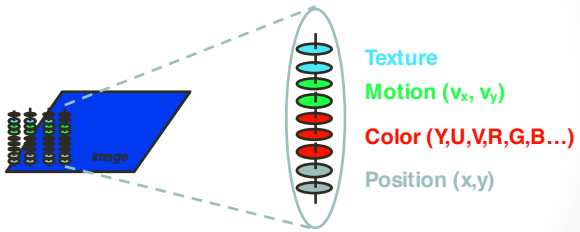
EBU706U

Past exam question

In the context of segmentation and retrieval, explain what a feature vector is. Describe how feature vectors may be used in image retrieval.

EBU706U

Feature vector



EBU706U

Example …

How would you represent a banana?

How many parameters do you need and what are they?

EBU706U

Multimedia Systems

- Indexing and retrieval of video

Dr. Yi-Zhe Song

EBU706U

Indexing and retrieval of video

- Q: How to index a video ?

EBU706U

Indexing and retrieval of video

- Q: How to index a video ?
- Q: How to select frames ?

EBU706U

Indexing and retrieval of video

- Q: How to index a video ?
- Q: How to select frames ?
- Q: How to summarise a video ?

EBU706U

Today's agenda

- Indexing and retrieval of video
  - Video content analysis
  - Shot detection
  - Video summarization (key-frames)
  - Uncompressed domain
  - Compressed domain
  - Indexing

EBU706U

Video summarization

- Temporal segmentation
  - Segment the video into basic units (shots)
  - Most used shot boundaries
    - Cut
    - Fade-in
    - Fade-out
    - Dissolve

    } Gradual transitions
- Key frame extraction
  - Salient content of each shot is represented by a small number of frames (key-frames)

EBU706U

Shot transition types

- Common shot transitions types
  - Cuts
    - Abrupt shot change between two consecutive frames
  - Dissolves
    - First shot images get dimmer, while second ones get brighter, with frames superimposed
  - Wipes
    - Image of the second shot replaces the first one in a regular pattern, such as vertical line

EBU706U

Video summarization

- Existing techniques
  - Temporal segmentation
    - Color-based algorithms
    - Edge-based algorithms
    - Motion-based algorithms
  - Key frame extraction
    - Shot-based criteria
    - Color-based criteria
    - Motion-based criteria

EBU706U

Compressed vs uncompressed domain

- Uncompressed (spatial/pixel) domain
  - ☺ more reliable
  - ☺ more information available
  - ☹ extremely inefficient → not suitable for
    - real time applications
    - large databases
- Compressed domain
  - ☹ unstable
  - ☹ less information available
  - ☺ extremely efficient → very suitable for
    - real time applications
    - large databases

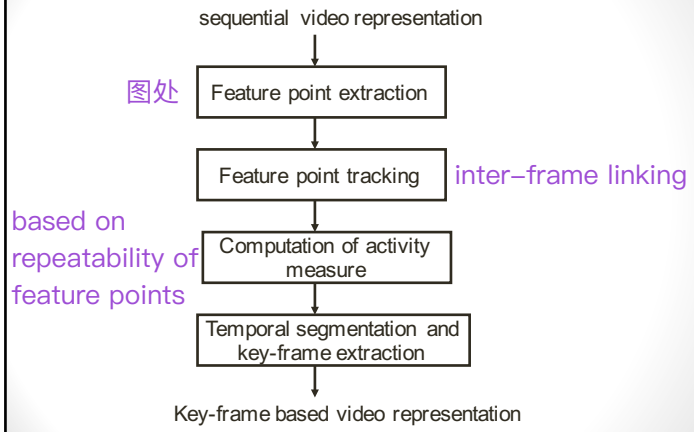
EBU706U

Today's agenda

- Indexing and retrieval of video
  - Video content analysis
  - Shot detection
  - Video summarization (key-frames)
  - Uncompressed domain
  - Compressed domain
  - Indexing

EBU706U

Video summarization algorithm



EBU706U

Video summarization

- Computation of activity measure:

cut a boundary here

Frame # 620 Frame # 621 Frame # 625

EBU706U

Definition of activity measure

- Use of tracks formed and maintained by a tracking algorithm
- Definition of the activity measure

EBU706U

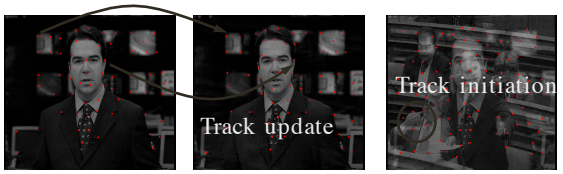
tracks lost in k–1 new tracks in k

$$a(k) = \max\left(\frac{\text{Number of teminated tracks at k}}{\text{Total number of tracks at k-1}}, \frac{\text{Number of initiated tracks at k}}{\text{Total number of tracks at k}}\right)$$

当前帧丢掉的点 / 上一帧总点数　　当前帧新增的点 / 当前帧总点数

Video summarization

- Computation of activity measure:



Frame # 620　　Frame # 621　　Frame # 625

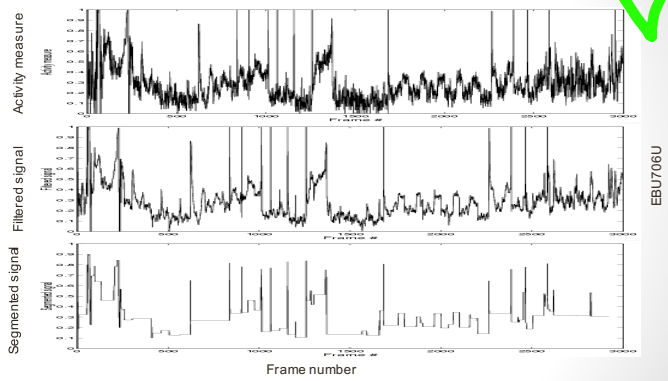
EBU706U

Temporal segmentation algorithm

- Requirements:
  - segments the activity measure into stationary segments (key-frames)
  - detects non-stationarities (shot boundaries)
- Exponential Weighted Moving Average (EWMA) algorithm is selected:
  - Filtering: $w(k) = \lambda a(k) + (1-\lambda) w(k-1)$
  - Segmentation: a change is detected if $|w(k) - \mu| \geq 3\sigma_w$

EBU706U

Temporal segmentation algorithm

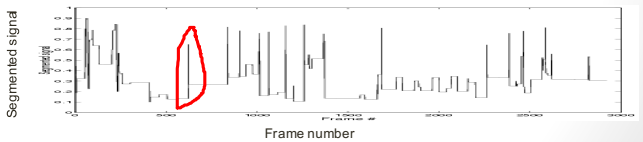


EBU706U

Temporal segmentation algorithm

Q: Where are the shot boundaries?

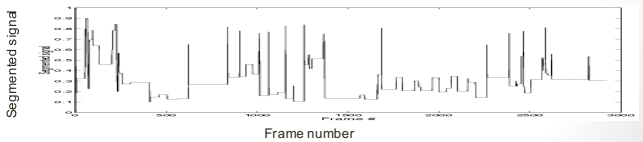
boundaries – non–stationarities



EBU706U

Temporal segmentation algorithm

Q: Where would you select the key frames then?



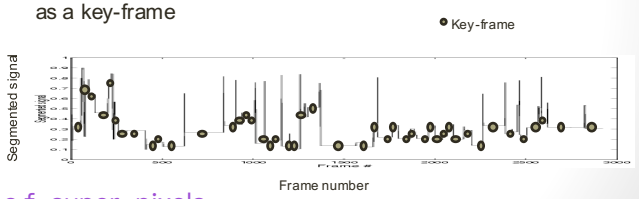
EBU706U

Key-frame extraction

- Key-frame extraction:

  Stationary segments ≡ important actions within a shot

  - Select the frame in the middle of each stationary segment as a key-frame

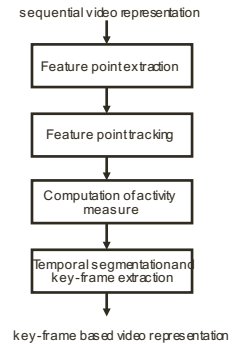


c.f. super–pixels

EBU706U

Video summarization

- Summary



EBU706U

Spatial domain techniques

- Feature tracking
- Histogram based
- Entropy based
- Motion estimation and analysis
- Edge extraction and tracking

EBU706U

Today's agenda

- Indexing and retrieval of video
  - Video content analysis
  - Shot detection
  - Video summarization (key-frames)
  - Uncompressed domain
  - Compressed domain
  - Indexing

EBU706U

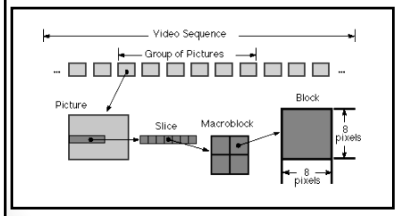
Shot detection in compressed domain

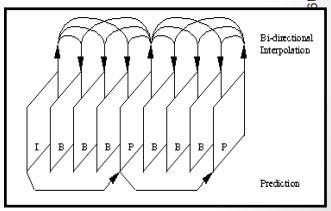
- Uncompressed domain
  - need for additional decompression of video sequence
  - ~40% of CPU time is spent in inverse DCT
- If time is an issue
  - Shot detection should happen in the compressed domain
- Used features
  - Frame and Macroblock (MB) types
  - Motion vectors
  - DC and AC coefficients in DCT domain

EBU706U

Shot detection using macroblock (MB) types

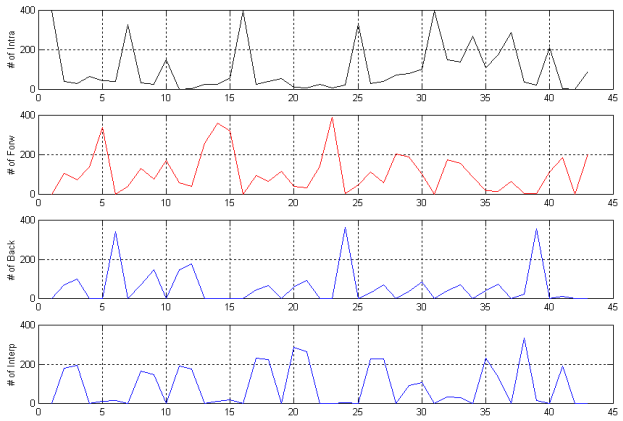
- Monitor the distribution of MB reference types based on
  - Position in Group Of Pictures (GOP)
  - Type of the current frame (I,P,B)



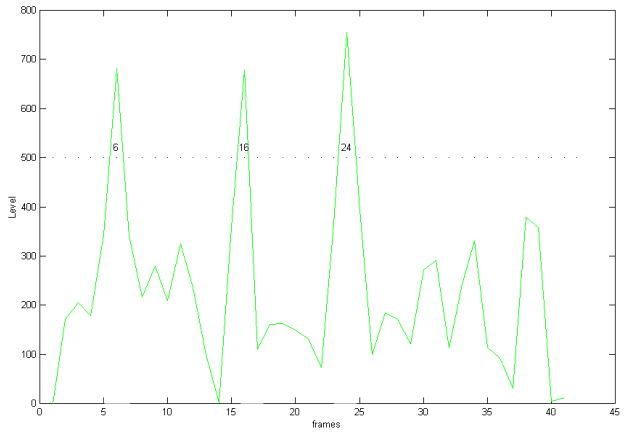


EBU706U

Statistical analysis



Shot detection



Other compressed domain techniques

- Motion analysis
- DC sequence analysis
- Using features extracted from AC coefficients (edge map, edge orientation, etc.)

EBU706U

Key-frame extraction (in the compressed domain)

- Key-frame
  - image that best represents the shot content
- Structure of the algorithm:
  - Detection of shot boundaries
  - Shot analysis
  - Key frame extraction
- Problems
  - More than one key-frame per shot
    - more information need to be stored and analysed
    - define a metric to measure "most representative" frame

EBU706U

Today's agenda

- Indexing and retrieval of video
  - Video content analysis
  - Shot detection
  - Video summarization (key-frames)
  - Uncompressed domain
  - Compressed domain
  - Indexing

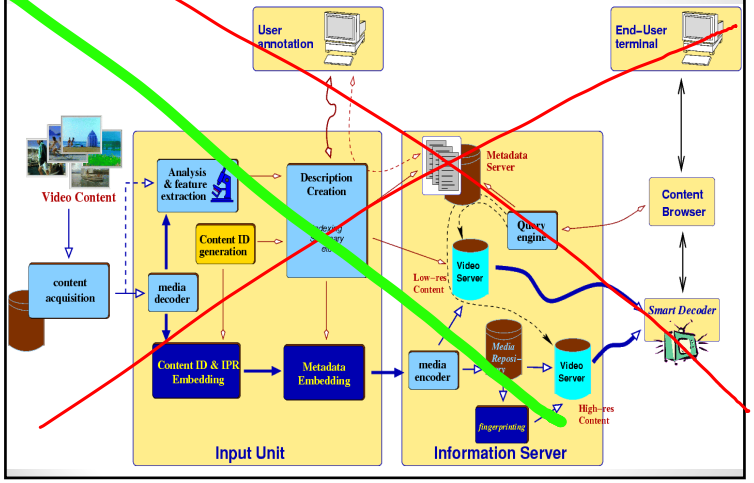
EBU706U

Indexing

- Key-frames are indexed
  - the metadata of key-frames is also metadata of video
  - retrieval is done by comparing descriptors in the descriptor space
- The final steps
  - Indexing video data
  - Linking and organizing descriptors in a database
  - Implementing GUIs for search and retrieval

EBU706U

Video content analysis and retrieval



What did we learn today?

- Indexing and retrieval of video
  - Video content analysis
  - Shot detection
  - Video summarization (key-frames)
  - Uncompressed domain
  - Compressed domain
  - Indexing

EBU706U

Past exam question

Outline the main differences between shot detection techniques in the pixel and in the compressed domain

EBU706U

Spatial/pixel Domain

Shot detection in the uncompressed domain supply better results, since

- more information is available: colour, texture, activity statistics, etc.
- These techniques are inefficient in terms of computational complexity,
- not suitable for real-time applications or systems where low access latency is demanded.
  - More reliable
  - More information available
  - Extremely inefficient
  - Not suitable for real time applications
  - Not suitable for large databases

EBU706U

Compressed Domain

- In the compressed domain only few video features are available.
- Most techniques exploit the statistics of the DCT coefficients in MPEGx compressed streams or available motion vectors extracted directly from the video stream.
- The accuracy of results is poor but these algorithms are efficient and suitable for real-time applications.
  - Unstable
  - Less information available
  - Extremely efficient
  - Suitable for real time applications
  - Very suitable for large databases

EBU706U

Past exam question

The following questions refer to video summarisation.

- List the two basic techniques for video summarisation and briefly explain their purpose.
- Explain what is an activity measure. How can it be computed?
- How can an activity measure be used for video summarisation?

EBU706U

Multimedia Systems

- Indexing and retrieval of audio

Dr. Yi-Zhe Song

audio – 1D signal

EBU706U

Indexing and retrieval of audio

- Q: What kind of information can be found in the audio track of a video?

EBU706U

Today's agenda

- Indexing and retrieval of audio: motivation
- Audio features
- Content-based audio retrieval
  - Feature extraction
  - Segmentation
  - Classification
  - Queries
  - Retrieval
- Audio classification
  - Feature extraction
  - Classification

EBU706U

Applications

- Audio-based multimedia applications
  - Music retrieval
  - Audio google
  - Child monitoring
  - Surveillance
    - complement video surveillance
- Microphones for surveillance
  - cheap sensors
  - ubiquitous: e.g., mobile phones
  - (possibly) larger areas covered by one microphone than by one camera
  - the 'people google' issue (privacy)

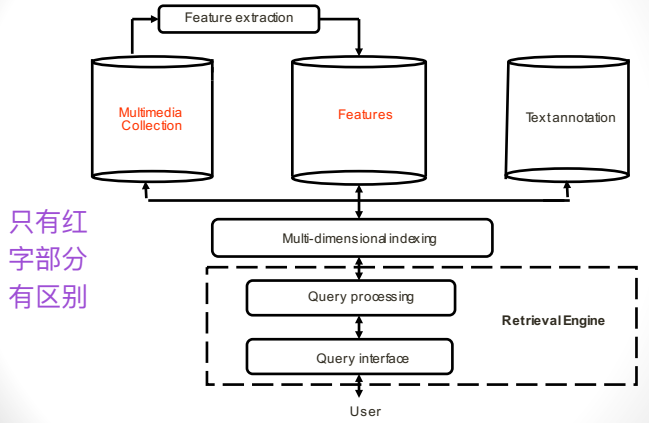
EBU706U

Annotation and retrieval

- Rapid increase of audio data
- Poor representation of audio data
  - file name
  - sampling rate
- Searching for a particular sound
  - can be a difficult task
- Solution
  - Content-based audio processing
  - Remember? same reasons for images and video …

EBU706U

Architecture of a CBIR system



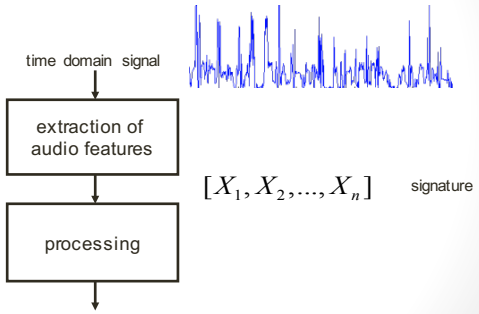
只有红字部分有区别

EBU706U

nature of audio – change of air pressure

Audio processing

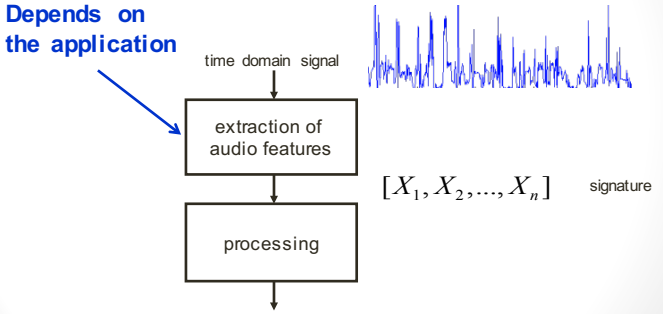
- Standard content-based audio analysis algorithm



EBU706U

Audio processing

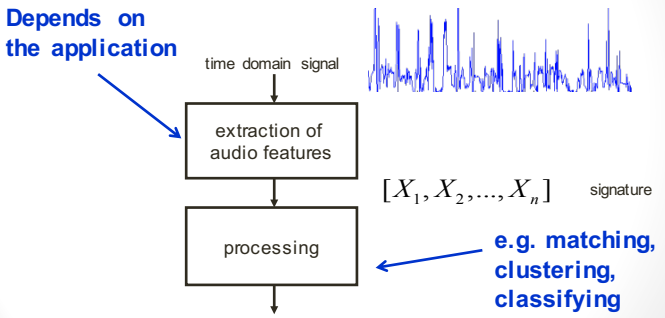
- Standard content-based audio analysis algorithm



EBU706U

Audio processing

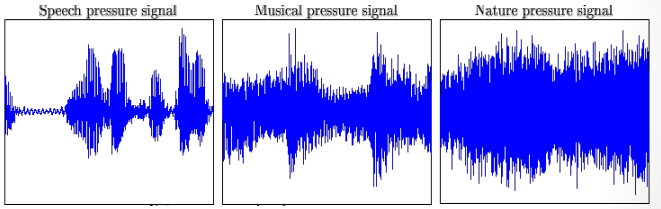
- Standard content-based audio analysis algorithm



EBU706U

Audio signals: examples

- Signals in the time domain
  - air pressure variation versus time



EBU706U

Today's agenda

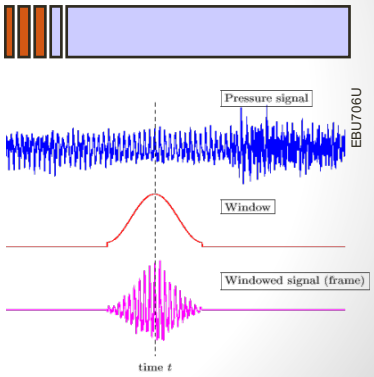
- Indexing and retrieval of audio: motivation
- Audio features
- Content-based audio retrieval
  - Feature extraction
  - Segmentation
  - Classification
  - Queries
  - Retrieval
- Audio classification
  - Feature extraction
  - Classification

Q: What kind of features can be extracted from audio signals?

EBU706U

Audio features

- Computation of frames
- Frame
  - group of neighboring samples
    - e.g., 10 to 40 ms
  - obtained by multiplying the pressure signal by a window located at time t



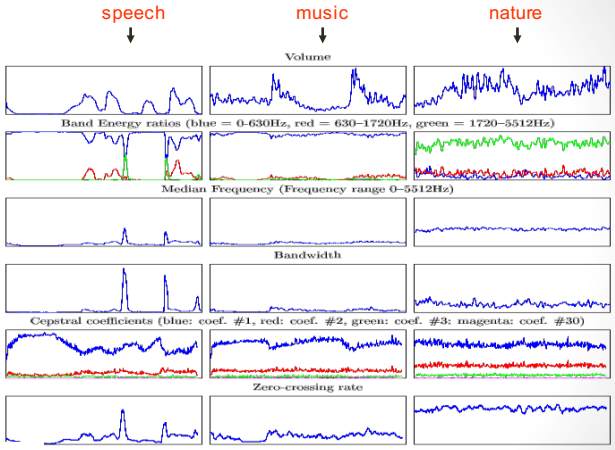
EBU706U

Audio features

- Computation of audio features
  - Volume (loudness)
  - Band energy
  - Median frequency (brightness)
  - Bandwidth
  - Cepstral coefficients
  - Zero-crossing rate

  - pitch [mean, variance, autocorrelation]
  - amplitude [mean, variance, autocorrelation]
  - brightness [mean, variance, autocorrelation]
  - bandwidth [mean, variance, autocorrelation]

EBU706U



EBU706U

Today's agenda

- Indexing and retrieval of audio: motivation
- Audio features
- Content-based audio retrieval
  - Feature extraction
  - Segmentation
  - Classification
  - Queries
  - Retrieval
- Audio classification
  - Feature extraction
  - Classification

EBU706U

Content-based audio retrieval

- Muscle fish
  - commercial product for audio signals classification and retrieval
  - sounds can be classified or queried by their audio content
  - users can retrieve sounds by
    - any one or a combination of the acoustical features
    - giving an example to find similar or dissimilar sounds

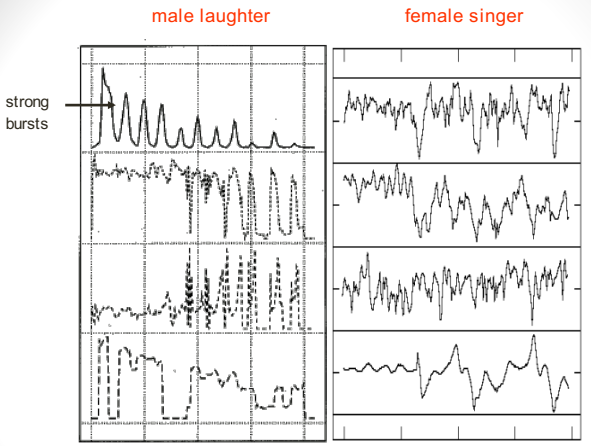
- Architecture
  - Feature extraction
  - **Segmentation** (remember for images and video?)
  - Classification
  - Indexing → index database ← search engine ← user

EBU706U

Feature extraction

- Frame level acoustic features
  - Loudness
  - Brightness
  - Bandwidth
  - Pitch

EBU706U



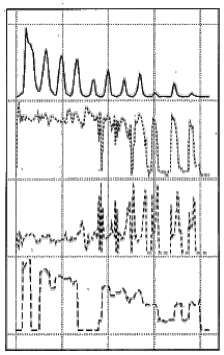
EBU706U

Feature extraction

- Problem with frame-level features
  - the amount of information is too large
- High-level acoustic features → summary
  - Average
  - Variance
  - Autocorrelation

EBU706U

Statistical analysis



Property	Mean	Variance	Autocorrelation
Loudness	-54.4112	221.451	0.938929
Pitch	4.21221	0.151228	0.524042
Brightness	5.78007	0.0817046	0.690073
Bandwidth	0.272099	0.0169697	0.519198

Male laughter. Duration: 2.12571

EBU706U

Audio database (indexing)

- Audio database
  - Sound file attributes
    - File name
    - Sample rate, sample size
    - Sound file format, number of channels
    - Creation date, analysis date
  - User attributes
    - Keywords
    - Comments
  - Feature vector
    - duration
    - pitch [mean, variance, autocorrelation]
    - amplitude [mean, variance, autocorrelation]
    - brightness [mean, variance, autocorrelation]
    - bandwidth [mean, variance, autocorrelation]

EBU706U

How to submit queries?

- By physical attributes
  - By specifying some acoustic characteristics
    - i.e., brightness, pitch, and loudness
    - Example: find the sound whose loudness is closest to -25dB
- By example
  - find similar or dissimilar sounds
- By subjective features
  - describing the sounds using personal descriptive language
    - Example: find the scratchy sounds
- By semantic content
  - text content for speech recordings and the score for musical recordings
    - Example: find the speech which contains the word "*coursework*"

EBU706U

How to retrieve sounds?

- In small databases
  - Compute the distance measures for all the sounds
  - Choose the sounds that match the desired result
- In large databases
  - Index the sounds in the database by all the acoustic features
  - Hyper-rectangle based technique

EBU706U

Retrieval

- Problem: Retrieve the top M sounds in a class with the mean μ, and the covariance matrix R

more than 3 dimensional

  - search all the sounds in a hyper-rectangle centered in μ with volume V such that $V / V_0 = M / M_0$
    - V_0 → volume of the hyper-rectangle surrounding the entire database
    - M_0 → total number of sounds in the database
  - Compute the distance measure for all the sounds returned
  - Return the closest M sounds

EBU706U

Today's agenda

- Indexing and retrieval of audio: motivation
- Audio features
- Content-based audio retrieval
  - Feature extraction
  - Segmentation
  - Classification
  - Queries
  - Retrieval
- Audio classification
  - Feature extraction
  - Classification

EBU706U

Audio classification

- Example: explosion filter
  - how content-based audio analysis help automatic understanding of the semantic meanings of a multimedia document
  - An audio-visual hierarchical model to detect explosion scenes from MPEG stream
    - *Input*: audiovisual stream
    - *Components*: audio analysis and video analysis (separated)
    - *Output*: explosion? Yes/No

binary classifier: gives 0 or 1

EBU706U

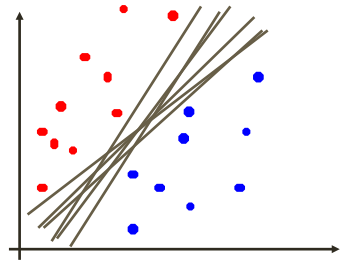
Audio classifier

- Support Vector Machine (SVM)
  - statistical learning algorithm [Vapnik 1992]
  - among the best performer for a number of classification tasks
    - from text to multimedia data
  - used to solve many practical problems
    - such as face detection and speech recognition

EBU706U

SVM

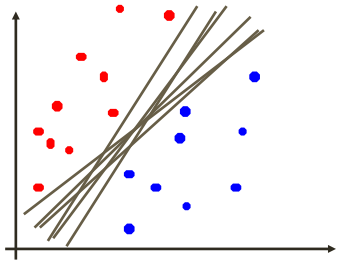
- Binary classification
  - can be viewed as the task of separating classes in the feature space
  - which separator (hyper-plane) is optimal?



EBU706U

SVM

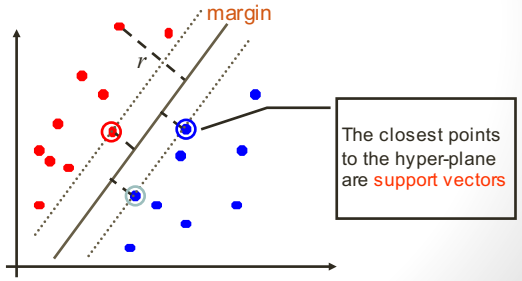
- Binary classification
  - can be viewed as the task of separating classes in the feature space
  - which separator (hyper-plane) is optimal?



EBU706U

SVM

- Main idea
  - given a set of training vectors belonging to separate classes
  - optimal separating hyper-plane → the one maximizing the margin
    - i.e., the distance between the hyper-plane and the nearest data point of each class



EBU706U

Audio classifier

- SVM
  - Training samples
  - Test samples → classification
- Hierarchical coarse-grained SVM
  - Coarse SVM
    - discriminate explosion and explosion-like sounds from others
  - Fine-grained SVM
    - discriminate explosion sound from explosion-like sounds

EBU706U

Multi-task Learning for Sketch-based Image Retrieval

Yi-Zhe Song

SketchX Lab
Queen Mary University of London

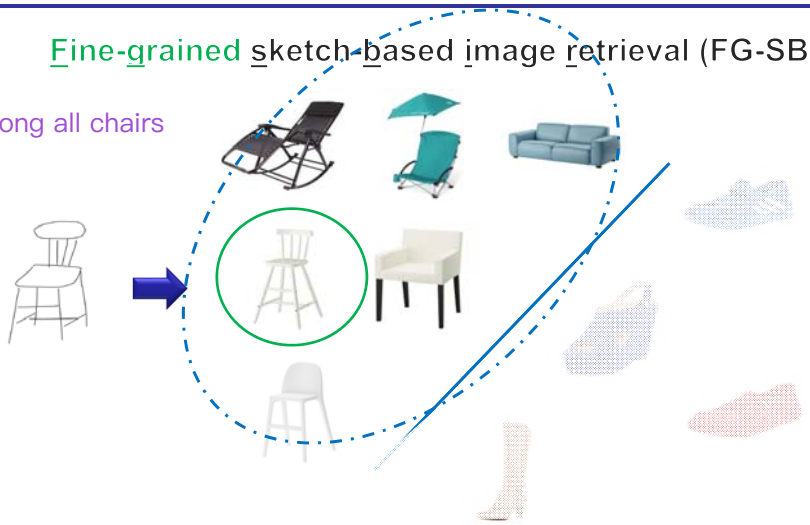
SketchX

Queen Mary University of London

The Problem

Fine-grained sketch-based image retrieval (FG-SBIR)

find THE chair among all chairs



Conventional sketch-based image retrieval (SBIR)

SketchX

Queen Mary University of London

Challenges

- Shared by SBIR and FG-SBIR:
  - Sketches are abstract.
  - Cross domain gap.

- Unique to fine-grained SBIR:
  - Fine-grained details detection.
  - Fine-grained rank.

1

2

fine-grained ranking

SketchX

Queen Mary University of London

Current Solution

- Dataset (CVPR2016):
  - **1,432** sketches and photos, **32,000** triplets pairs
  - **21** attributes (15 for chairs)

- Triplet samples: – human annotated 人为评分

Anchor Positive Negative

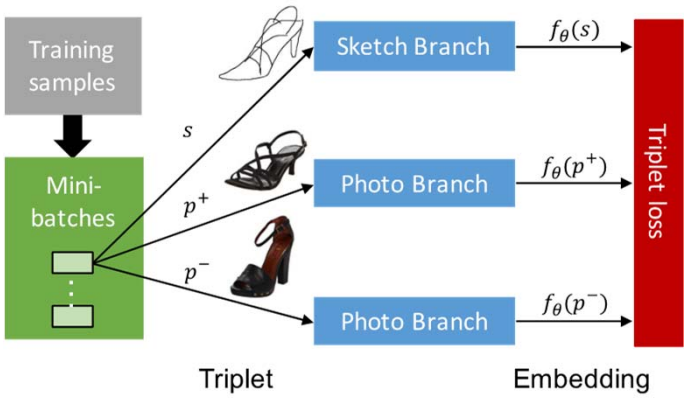
SketchX

[1] Yu, Qian and Liu, Feng and Song, Yi-Zhe and Xiang, Tao and Hospedales, Timothy M and Loy, Chen Change, "Sketch me that shoe." CVPR, 2016, Oral

Queen Mary University of London

Current Solution

- State-of-the-art [1] method (Triplet model, CVPR2016):



[1] Yu, Qian and Liu, Feng and Song, Yi-Zhe and Xiang, Tao and Hospedales, Timothy M and Loy, Chen Change, "Sketch me that shoe." CVPR, 2016, Oral

Current Solution

- Triplet model, CVPR 2016:

✓ **Pros**:

1. More **aligned** with Learned feature
2. Learn **ranking** with triplet annotation

× **Cons**:

1. Candidates with similar attributes **ranked behind**
2. Triplet samples need **expensive** human annotation

[1] Yu, Qian and Liu, Feng and Song, Yi-Zhe and Xiang, Tao and Hospedales, Timothy M and Loy, Chen Change, "Sketch me that shoe." CVPR, 2016, Oral

The Role of Visual Attributes

Attribute-driven side task with attribute **ranking** layer

Deep Feature Rank

Deep Attribute Rank

Automatically generated **chain-like** triplets using **attributes**

SketchX

Queen Mary University of London

Why Attribute?

- A semantic bridge between sketch and photo

With shoelace

With buckle

High boot

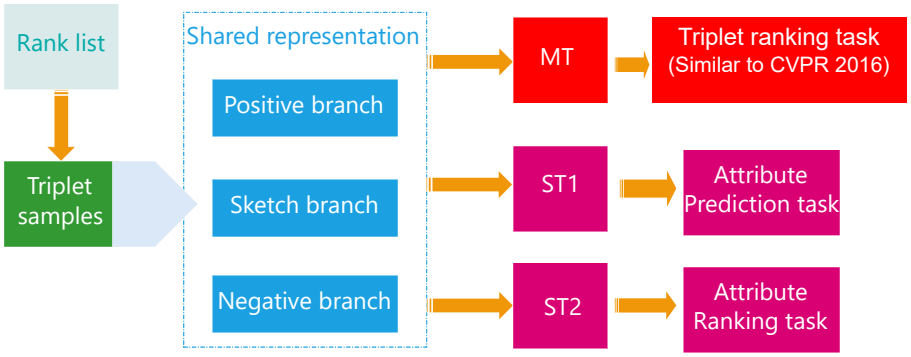
Semantic level √

Narrow cross-domain gap √

SketchX

Queen Mary University of London

➢How we use attributes



SketchX

Queen Mary University of London

Multi-task Learning

➢ Multi-task Learning:

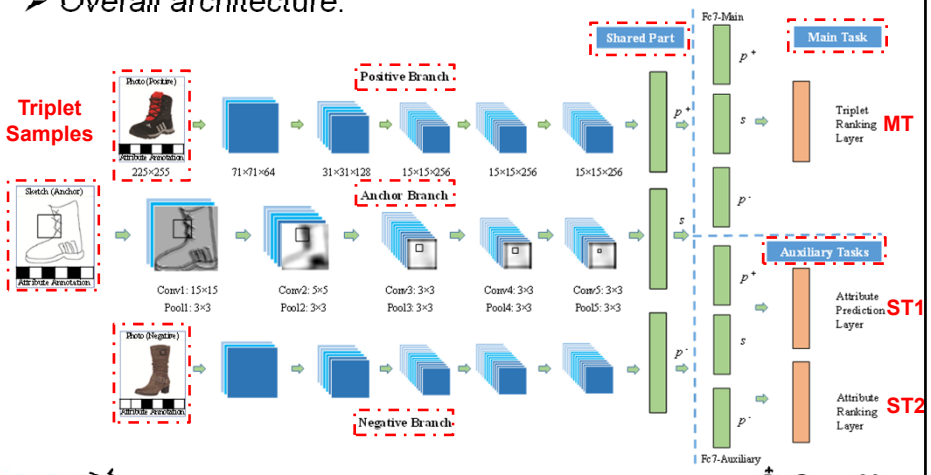
- Multi-task Learning
  - Main task
    - Face landmark detection (Zhang, Zhanpeng et. al, PAMI 2016)
    - Pedestrian detection (Tian, Yonglong, et. al., CVPR 2015)
    - Video thumbnail selection (Liu, Wu, et. al., CVPR 2015)
  - Side task
    - Attribute prediction

SketchX

Queen Mary University of London

Methodology

➢ *Overall architecture:*



Methodology

- Main triplet ranking task: Distance between **sketch** and **positive** photo feature Distance between **sketch** and **negative** photo feature

$$L_\theta(s,i^+,i^-)=\max\left(0,\Delta+D\left(f_\theta(s),f_\theta(i^+)\right)-D\left(f_\theta(s),f_\theta(i^-)\right)\right)$$

- Attribute prediction task:

$$L_p(s,t^s)=-\frac{1}{N}\sum_{n=1}^{N}\left[t_n^s\log f_{\theta,n}^{ap}(s)+\left(1-t_n^s\right)\log\left(1-f_{\theta,n}^{ap}(s)\right)\right]$$

- Attribute ranking task:

$$L_a(s,i^+,i^-)=\max\left(0,\Delta+H\left(f_\theta^{ap}(s),f_\theta^{ap}(i^+)\right)-H\left(f_\theta^{ap}(s),f_\theta^{ap}(i^+)\right)\right)$$

$$L_a(s,i^+)=H\left(f_\theta^{ap}(s),f_\theta^{ap}(i^+)\right)$$

Cross-entropy distance between **sketch** and **positive** photo attributes

Cross-entropy distance between **sketch** and **negative** photo attributes

Multi-task Ranking Loss

- Multi-task training loss:

MT ST1 ST2

$$L(s,i^+,i^-)=L_\theta(s,i^+,i^-)+\lambda_s L_p(s,t^s)+\lambda_{i^+}L_p(s,t^{i^+})+\lambda_{i^-}L_p(s,t^{i^-})+\lambda_a L_a(s,i^+)+\lambda_\theta\|\theta\|_2^2$$

- Multi-task ranking score:

Deep **Feature Ranking** Score

$$R_s(s,i)=D(f_\theta(s),f_\theta(i))+\lambda_a H(f_\theta^{ap}(s),f_\theta^{ap}(i))$$

Deep **Attribute Ranking** Score

SketchX

Queen Mary University of London

Contribution

Attribute-driven side task with attribute **ranking** layer

Deep Feature Rank

Deep Attribute Rank

Automatically generated **chain-like triplets** using **attributes**

SketchX

Queen Mary University of London

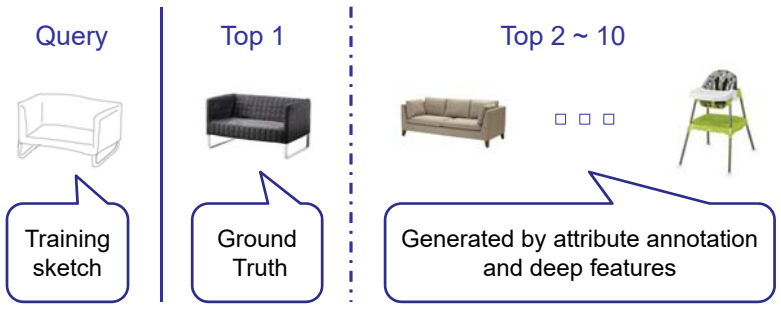
Automatic Triplet Annotation using Attributes

- Comparison between triplet annotation cost:
  1) CVPR 2016: 32,000 triplets, 100h
  2) Ours: 7,160 triplets, automatically generated
- Advantage of proposed annotation strategy:
  1) Save ***90%*** human annotations
  2) Improve ***5%*** top 1 retrieval accuracy

SketchX

Queen Mary University of London

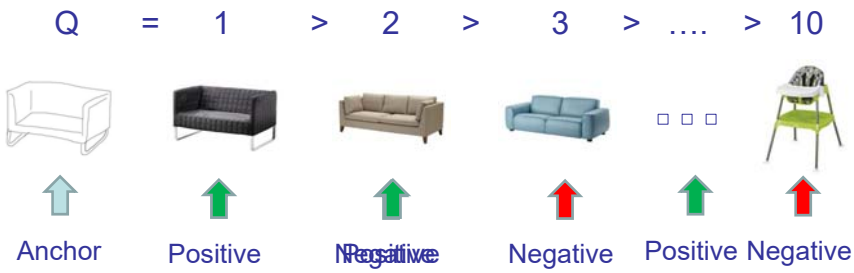
Generated Reference Rank List



SketchX

Queen Mary University of London

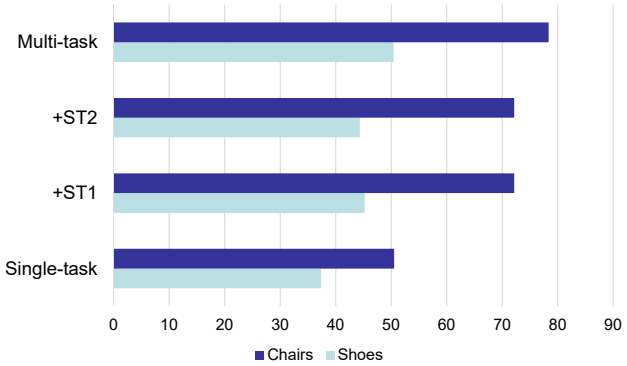
Chain-like Hard Triplets



SketchX

Queen Mary University of London

Contribution of Different Components

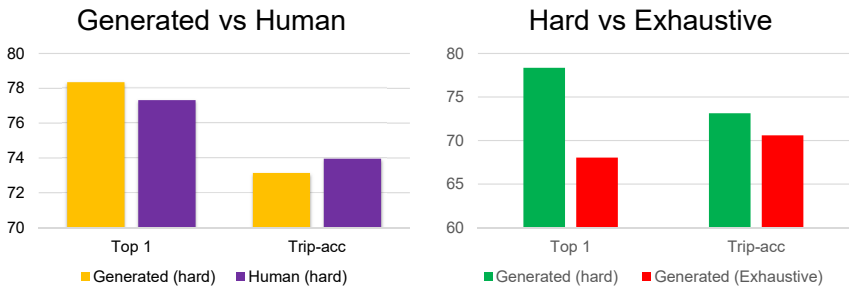


- **ST1**: attribute prediction task. **ST2**: attribute ranking task.

SketchX

Queen Mary University of London

Effect of Hard Triplet Annotations

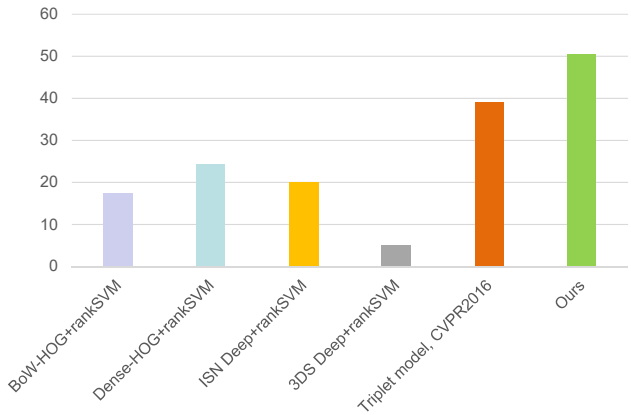


- Performance: Generated triplets ≈ human triplets.
- Hard triplets > exhaustive triplets.

SketchX

Queen Mary University of London

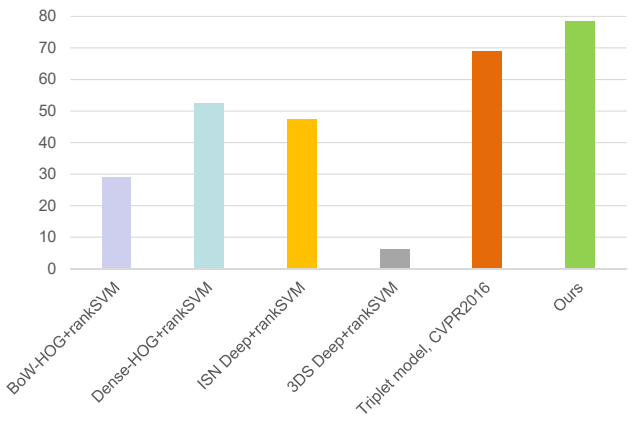
Top 1 Accuracy (Shoe Dataset)



SketchX

Queen Mary University of London

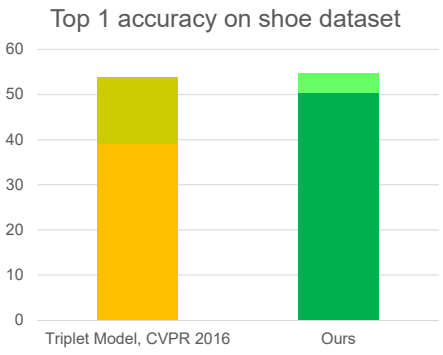
Top 1 Accuracy (Chair Dataset)

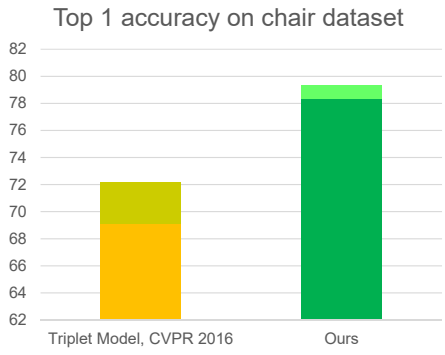


SketchX

Queen Mary University of London

Result after Refined Data

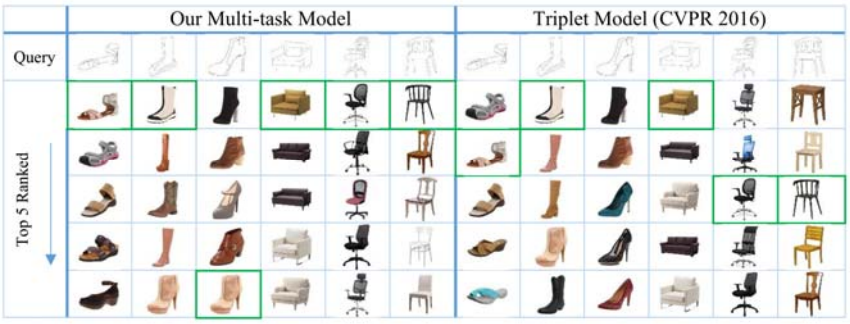




SketchX

Queen Mary University of London

Qualitative Results against the State-of-the-art



SketchX

Queen Mary University of London