

Multimedia Systems

- Indexing and retrieval of still images

Dr. Yi-Zhe Song

EBU706U

Today's agenda

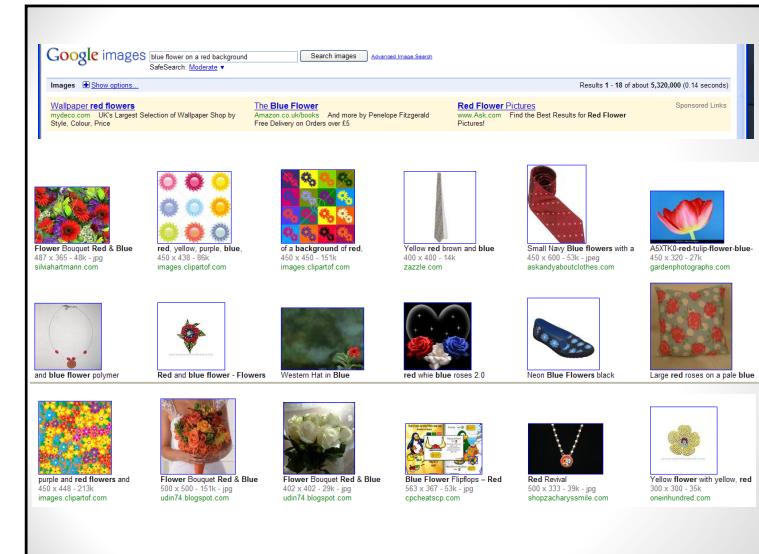
- Introduction
- Motivations
- Challenge
- Current systems and problems
- Colour-based retrieval ✓
- Shape-based retrieval ✓
- Texture-based retrieval ✓
- Examples

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'

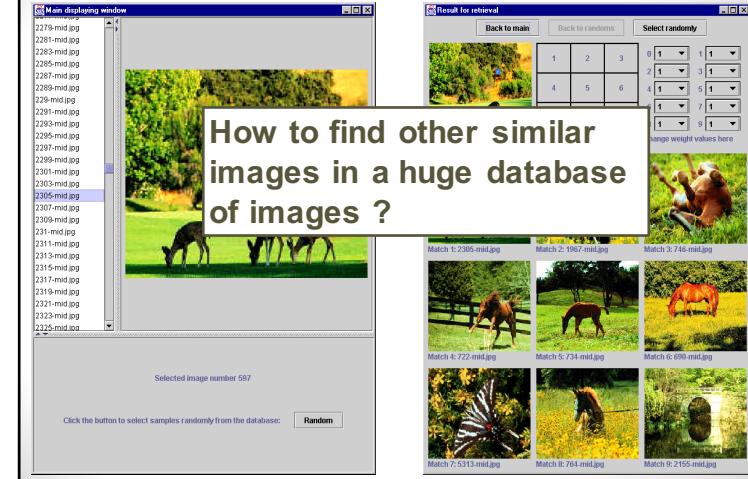
EBU706U



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 多看往年案例題

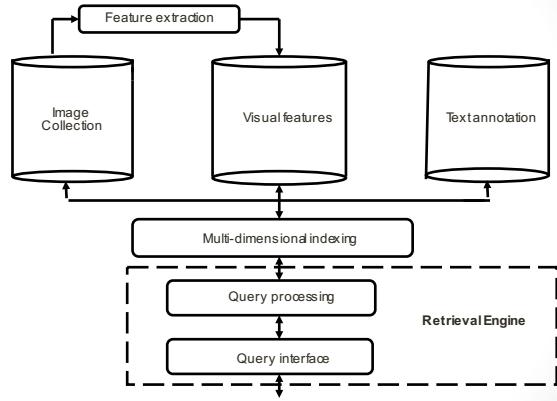
- Problems in Content-Based Image Retrieval (CBIR)
 - computational complexity increases with the number of features used
 - scalability: retrieval time depend on the database size
 - access latency
 - inefficiency in indexing and searching
 - unrealistic assumptions needed in some systems
 - gap between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

EBU706U



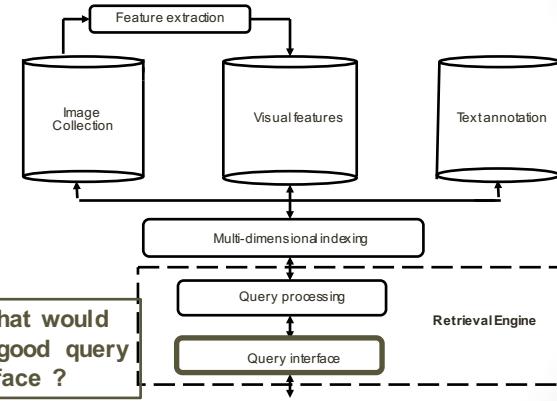
examable...

Architecture of a CBIR system



EBU706U

Architecture of a CBIR system



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

scenario:

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

EBU706U

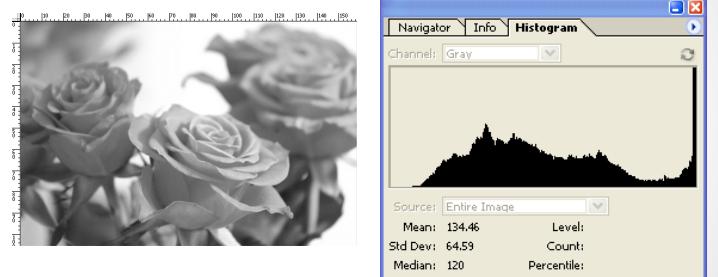
Histograms

one way to encode colours

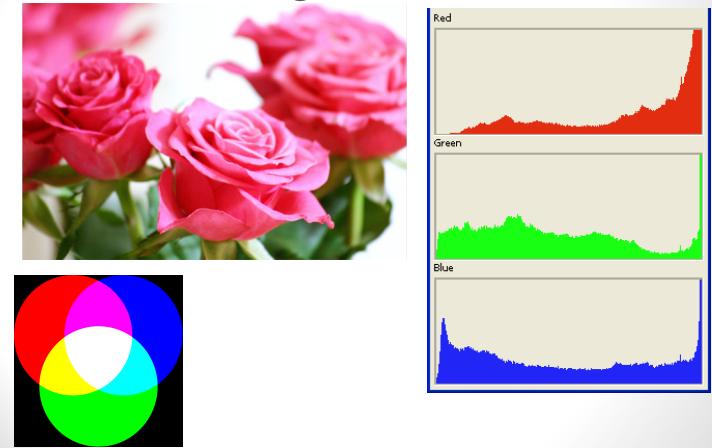
- Definition
 - function that maps the quantization levels into the frequency of each quantization level in the image
 - "counts" how many times each "colour" appears in the whole image
- Grey-scale histogram
 - number of pixels at each grey-level or in a range of grey levels
 - plot of frequencies of 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



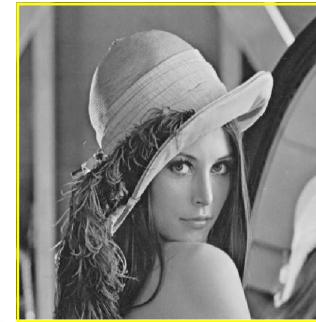
EBU706U

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

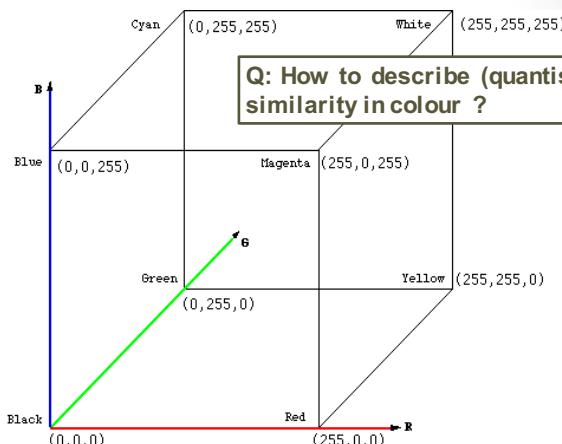
identical – histograms do not contain spatial
information

Histogram: properties

Q: How about these ones?



EBU706U



EBU706U

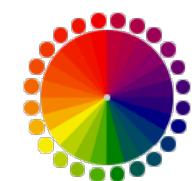
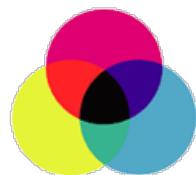
Additive vs. Subtractive

- 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

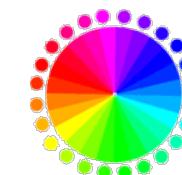
- When we mix colors using paint, or through the printing process, we are using the subtractive color method.
- Subtractive color mixing means that one begins with white and ends with black; as one adds color, the result gets darker and tends to black



EBU706U

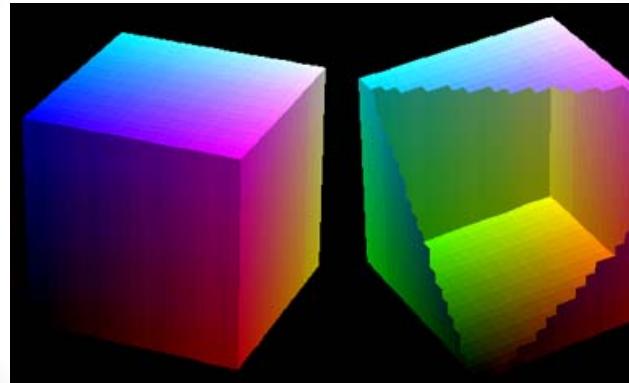
Additive vs. Subtractive

- 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



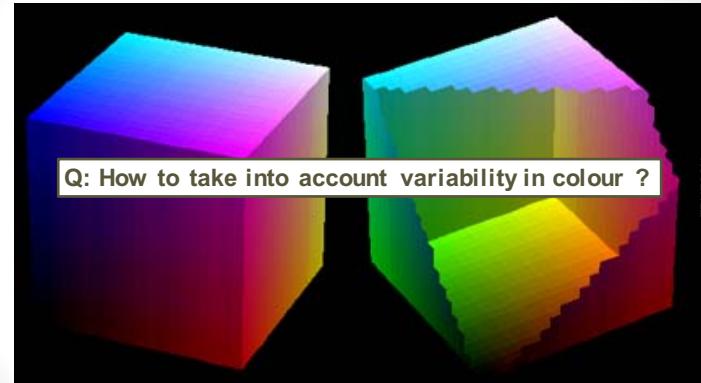
EBU706U

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) :

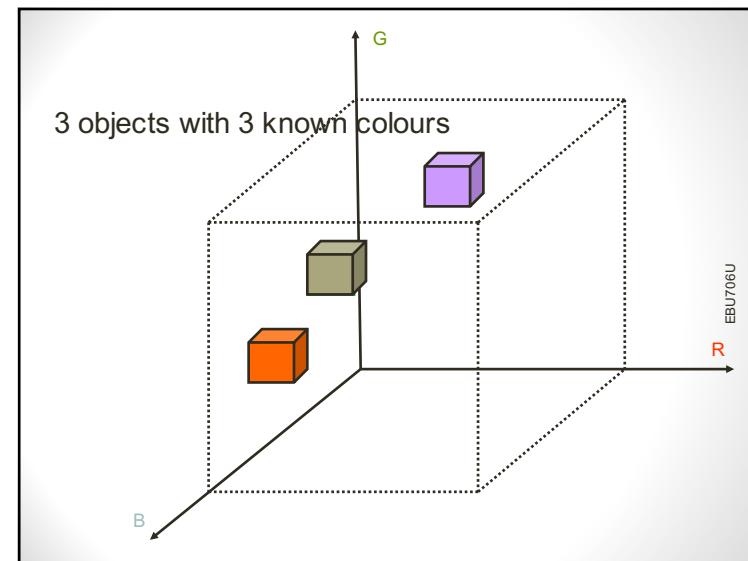
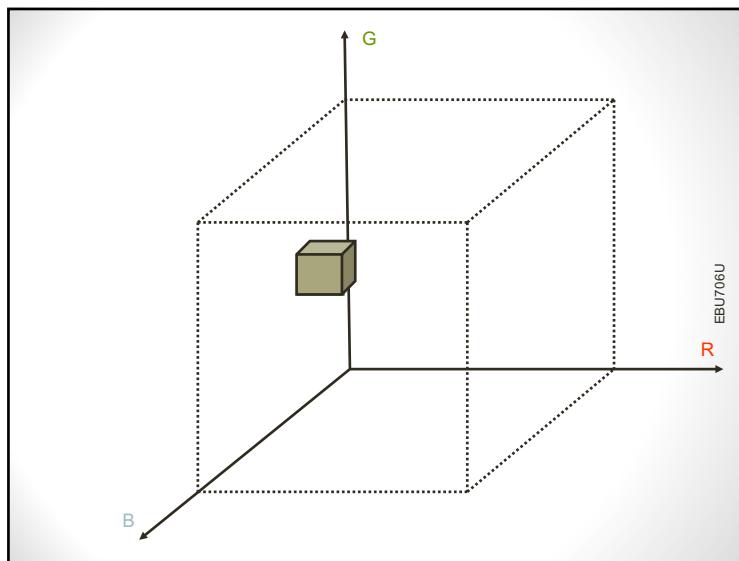
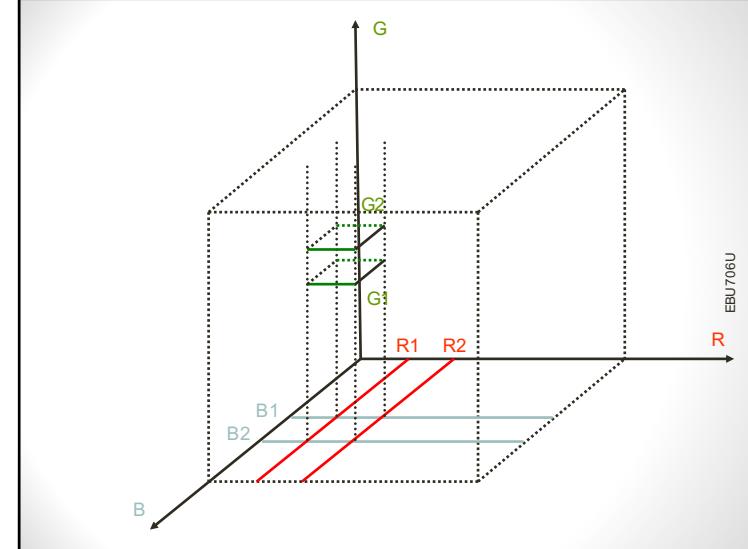
$$R_1 < R < R_2$$

$$G_1 < G < G_2$$

$$B_1 < B < B_2$$

EBU706U

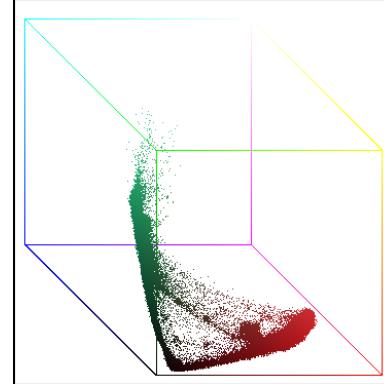
Q: How to represent these inequalities in the RGB space?



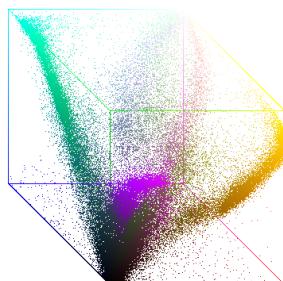


Q: How many colours are there in this picture?

EBU706U

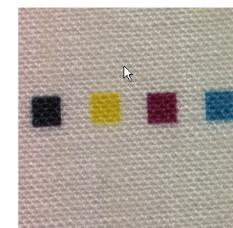
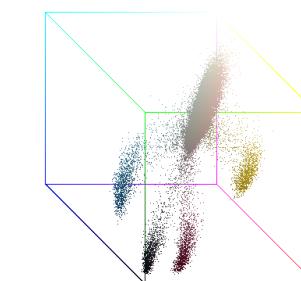


EBU706U



EBU706U

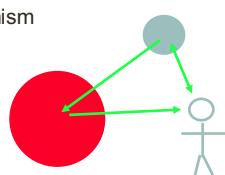
Q: How do you explain this colour distribution?



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

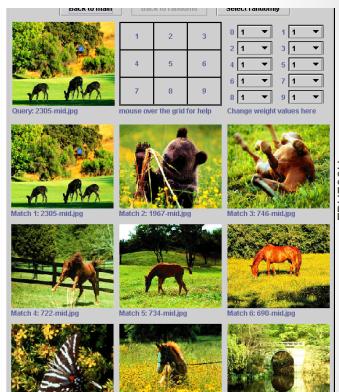
Colour-based approaches

- 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

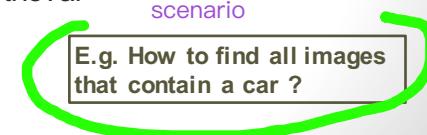
Today's agenda

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

EBU706U

scenario

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



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

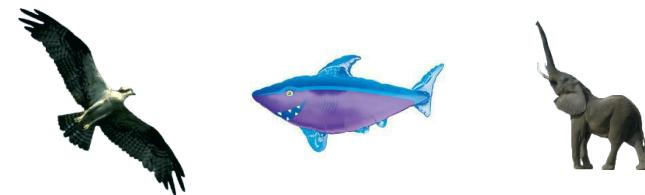
E.g. How to find all images that
contain a car ?

EBU706U

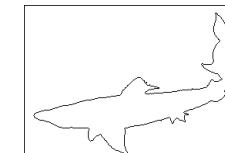
Q: How to encode the information
about a shape ?

Q: How to calculate the similarity
between shapes ?

Shape-based retrieval



contour

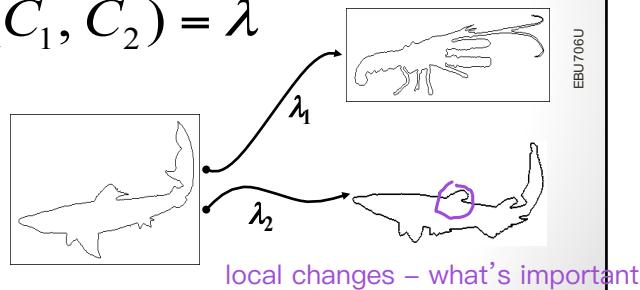


EBU706U

Shape-based retrieval

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

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



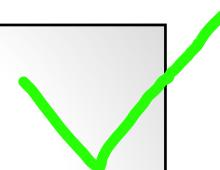
Metric for contour matching

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

Polygonal approximation

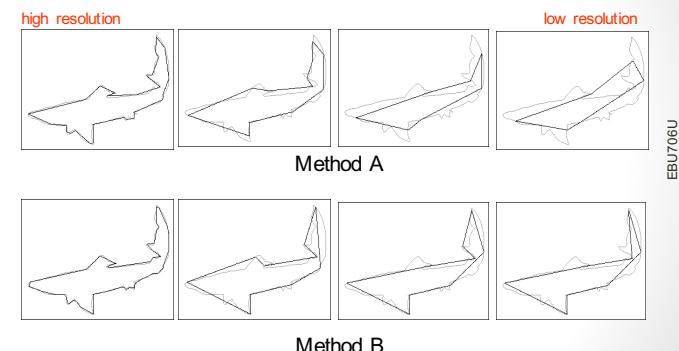
why? reduce complexity

- 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



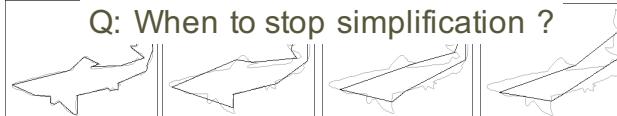
Polygonal approximation

decreasing resolution

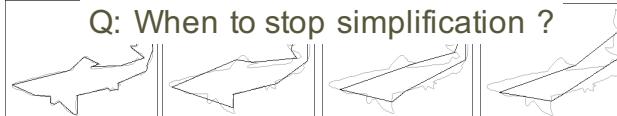


Polygonal approximation

high resolution

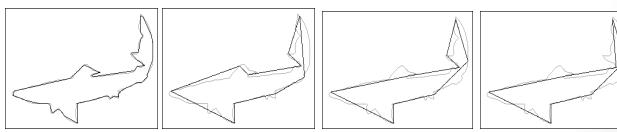


low resolution



Method A

EBU706U



Method B

Polygonal approximation

high resolution



low resolution

Q: What happens if the shape is rotated ?

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



Method B

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
- Each polygonal approximation π_i is represented by its **Turning Function** rotation problem solved
- The distance between π_1 and π_2 is defined as the L_2 norm of the difference between their corresponding turning functions

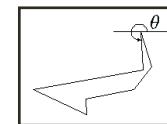
EBU706U

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

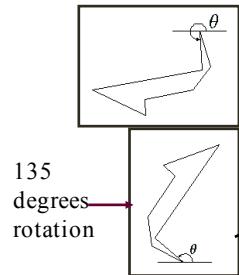
rotation-invariant

scale-invariant

due to normalisation



The turning function

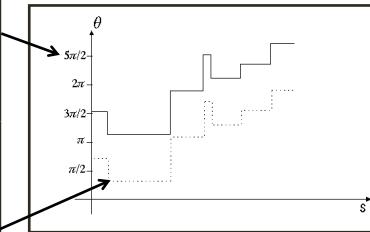


135
degrees
rotation

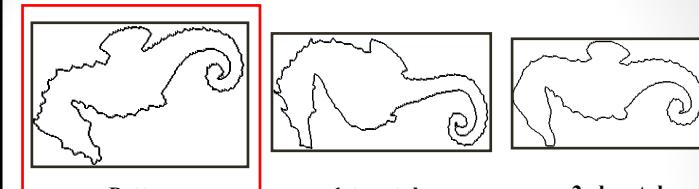
starting point: usually the extreme point,
then walk counter-clockwise



EBU706U



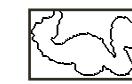
Example of retrieval



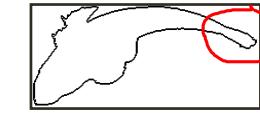
Pattern

1st match

2nd match



3rd match



4th match



5th match

EBU706U

What this reflects:
Differentiation on tails does not work well

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

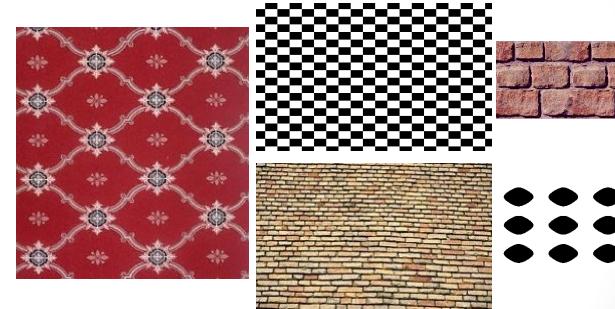
Texture descriptors

- 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

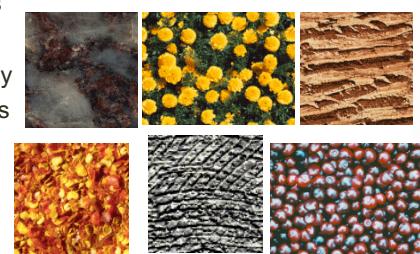
Example: statistical textures



EBU706U

Texture features

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



EBU706U



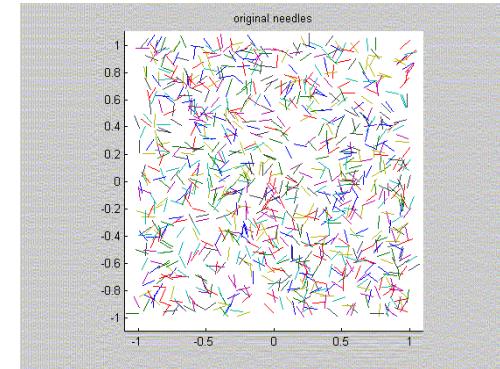
Example: statistical textures

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

Step 2:
Adjust the weights below if you'd like, then click "Submit."

Not Important	Not Somewhat	Very	Not Important	Not Somewhat	Very																																				
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	How important is the selected region?			How important is the background (everything outside the region)?			How important are the features of this region?						Color	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Color	<input type="radio"/>	Texture	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Texture	<input type="radio"/>	Location	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Location	<input type="radio"/>	Shape/Size	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Shape/Size	<input type="radio"/>
<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>																																							
How important is the selected region?			How important is the background (everything outside the region)?																																						
How important are the features of this region?																																									
Color	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Color	<input type="radio"/>																																				
Texture	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Texture	<input type="radio"/>																																				
Location	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Location	<input type="radio"/>																																				
Shape/Size	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Shape/Size	<input type="radio"/>																																				

Colour search: QBIC

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

QBIC COLOUR SEARCH

1. Use your mouse to select a colour from the palette.
2. Click the arrow button to add the colour to the bucket.
3. Slide the triangular handles on the bucket to set the percentage of this colour.
4. You may repeat this process until the bucket is full of the percentage of this colour.

You may also use the Colours palette to add red, green, blue values to use in your search.

QBIC LAYOUT SEARCH

1. Use your mouse to choose a colour from the palette.
2. Select the round tool or the square tool.
3. Hold down the mouse button and drag the cross on the mouse to draw a new shape.
4. Repeat this process until you complete your custom layout. When finished, click Search.

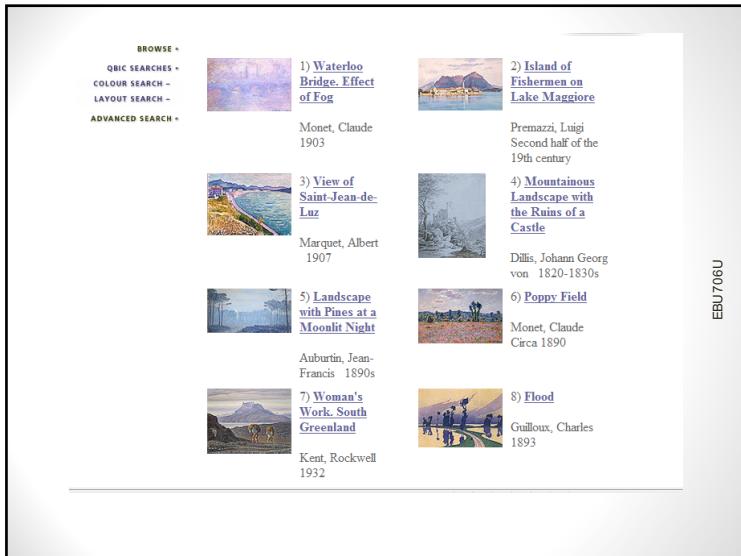
EBU706U

QBIC LAYOUT SEARCH

BROWSE +
QBIC SEARCHES +
COLOUR SEARCH +
LAYOUT SEARCH +
ADVANCED SEARCH +

1. Use your mouse to choose a colour from the palette.
2. Select either the round tool or the square tool.
3. Hold down your mouse button and drag the cross on the mouse to draw a new shape.
4. Repeat this process until you complete your custom layout. When finished, click Search.

To perform other actions, click the shape to make it active. Drag the edges to Resize. Click Send to Back and Bring to Front to layer shapes. Click Delete to remove a shape. Click Clear All to empty the layout.



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

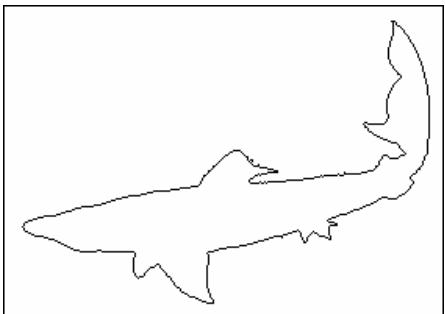
Shape-based retrieval

- 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.



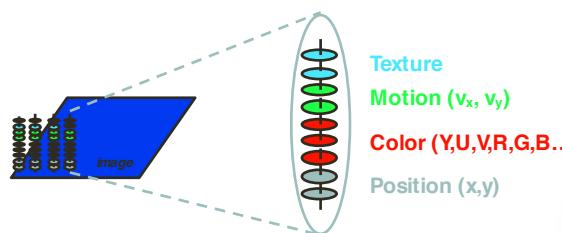
EBU706U

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
- 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

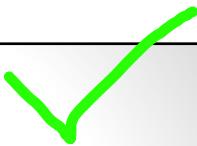
- Existing techniques
- Temporal segmentation
 - Color-based algorithms
 - Edge-based algorithms
 - Motion-based algorithms
- Key frame extraction
 - Shot-based criteria
 - Color-based criteria
 - Motion-based criteria

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

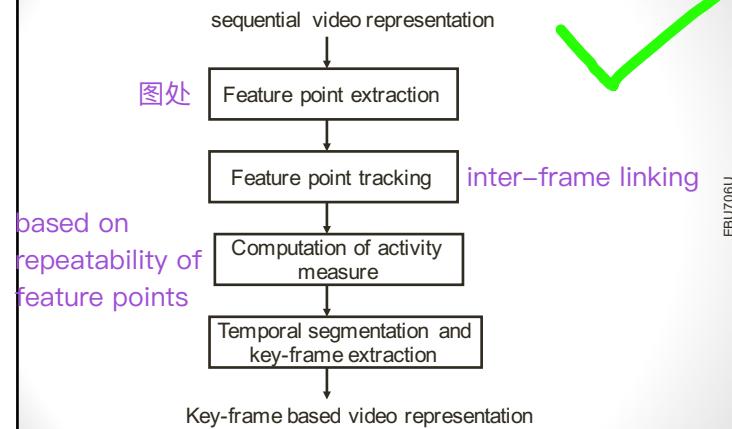


Today's agenda

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

EBU706U

Video summarization algorithm



Video summarization

- Computation of activity measure:



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

tracks lost in k-1 new tracks in k

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

EBU706U



Video summarization

- Computation of activity measure:



Frame # 620

Frame # 621

Frame # 625

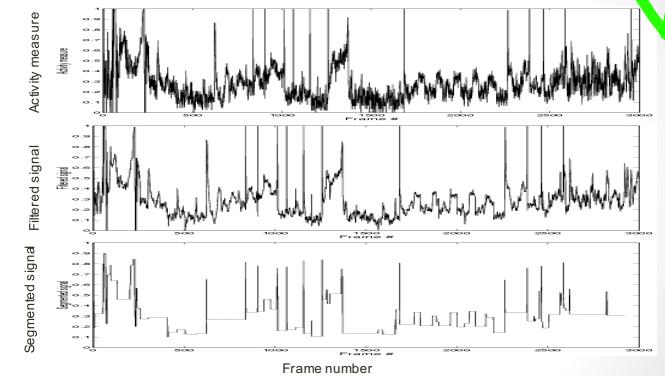
EBU706U

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

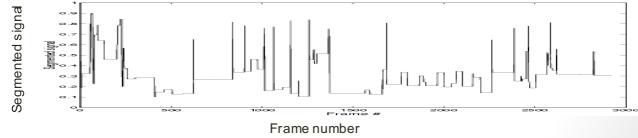
Temporal segmentation algorithm



EBU706U

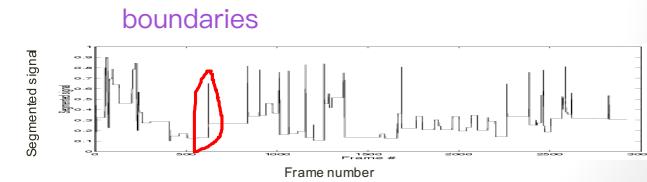
Temporal segmentation algorithm

Q: Where are the shot boundaries?



Temporal segmentation algorithm

Q: Where would you select the key frames then?



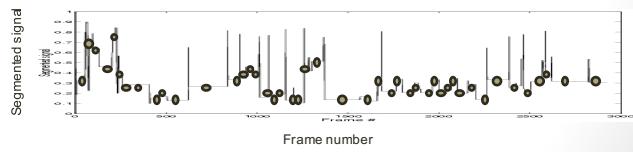
Key-frame extraction

- Key-frame extraction:

Stationary segments = important actions within a shot

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

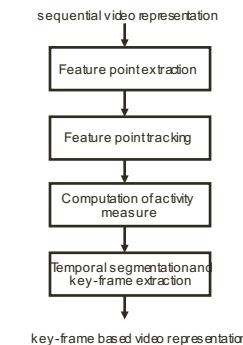
• Key-frame



c.f. super-pixels

Video summarization

- Summary



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

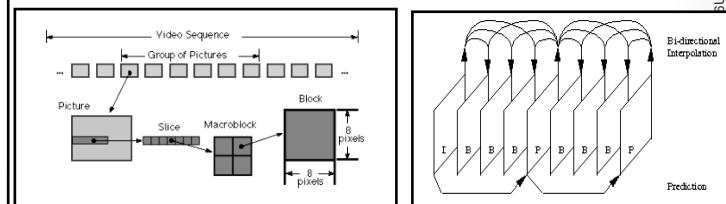
Shot detection in compressed domain

- 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

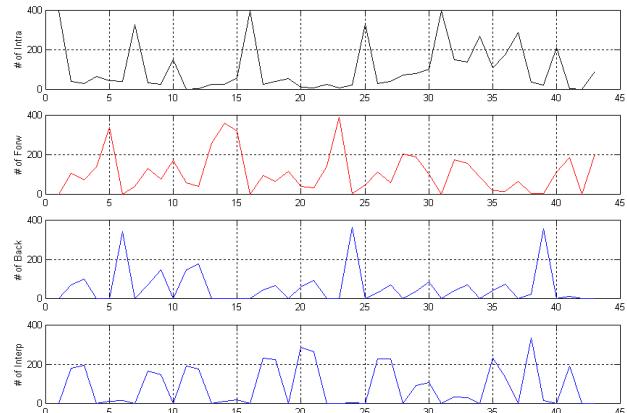
EBU706U

Shot detection using macroblock (MB) types

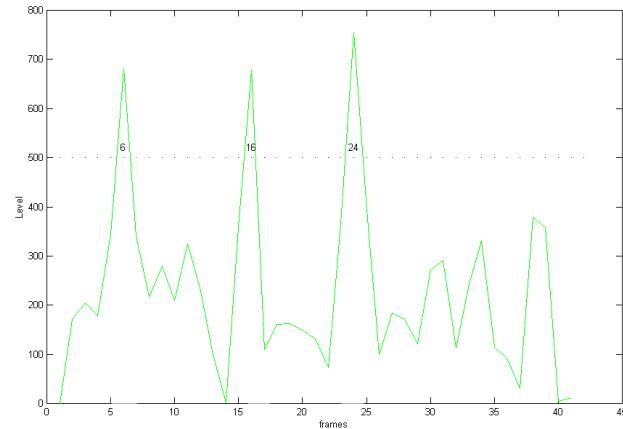
- Monitor the distribution of MB reference types based on
 - Position in Group Of Pictures (GOP)
 - Type of the current frame (I,P,B)



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

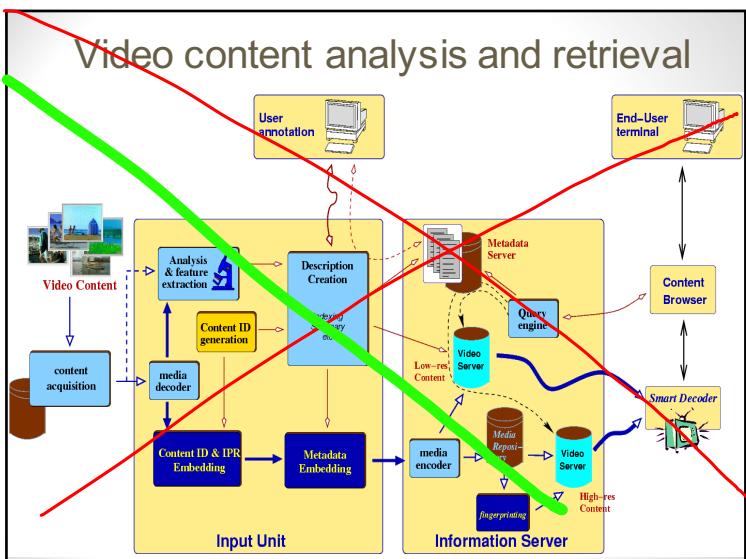
- Indexing and retrieval of video
 - Video content analysis
 - Shot detection
 - Video summarization (key-frames)
 - Uncompressed domain
 - Compressed domain
 - **Indexing**

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



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)

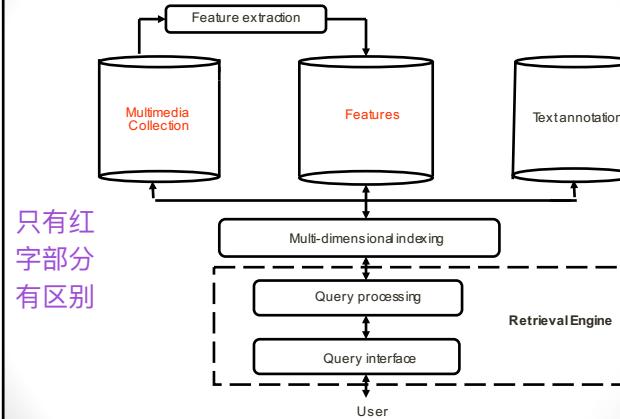
EBU706U

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



EBU706U

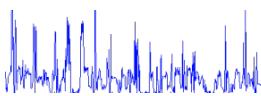
nature of audio – change of air pressure

Audio processing

- Standard content-based audio analysis algorithm

time domain signal

extraction of
audio features



$[X_1, X_2, \dots, X_n]$ signature

EBU706U

Audio processing

- Standard content-based audio analysis algorithm

Depends on
the application

extraction of
audio features



$[X_1, X_2, \dots, X_n]$ signature

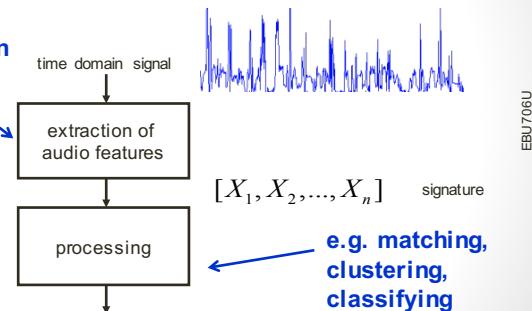
processing

EBU706U

Audio processing

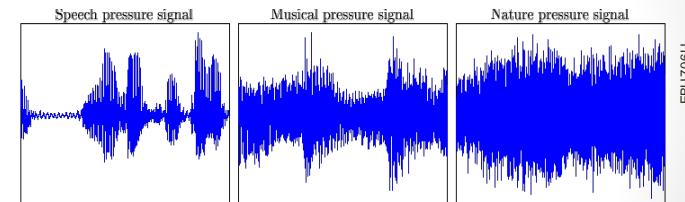
- Standard content-based audio analysis algorithm

Depends on
the application



Audio signals: examples

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



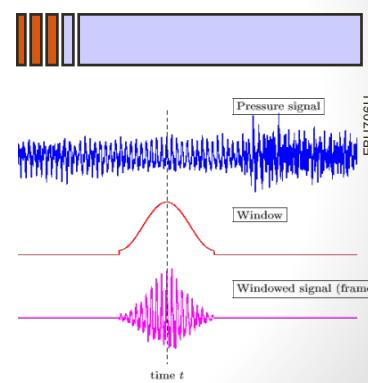
Today's agenda

- 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?

Audio features

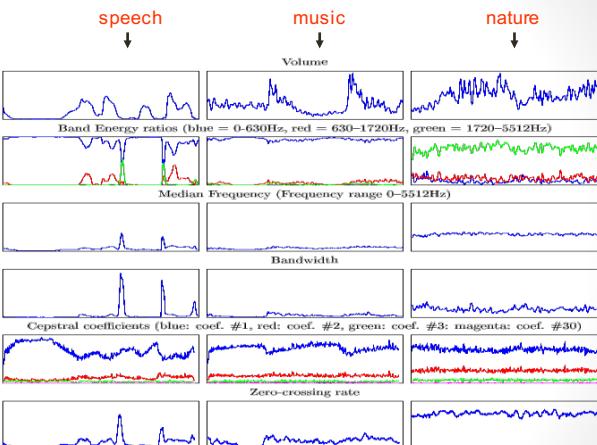
- Computation of **frames**
- Frame
 - group of neighboring samples
 - e.g., 10 to 40 ms
 - obtained by multiplying the pressure signal by a window located at time t



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

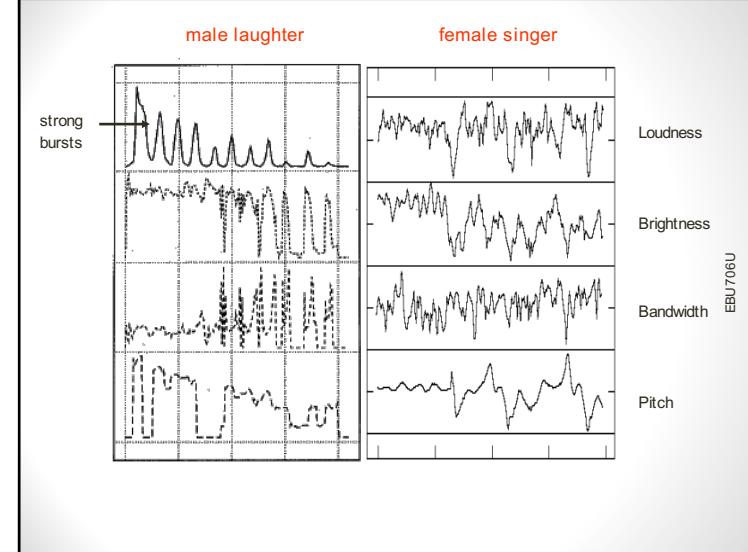
- Muscle fish
 - commercial product for audio signals classification and retrieval
 - sounds can be classified or queried by their audio content
 - users can retrieve sounds by
 - any one or a combination of the acoustical features
 - giving an example to find similar or dissimilar sounds
- Architecture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

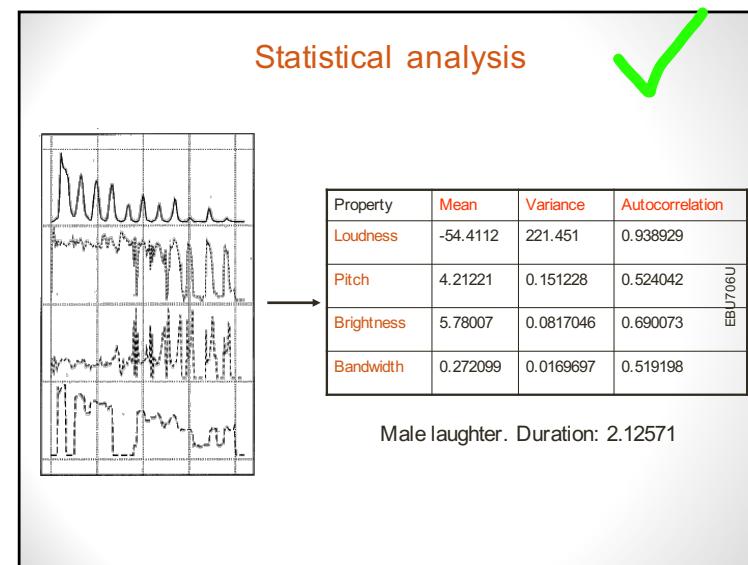
EBU706U



Feature extraction

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

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 $V / V_0 = M / M_0$
 - $V_0 \rightarrow$ volume of the hyper-rectangle surrounding the entire database
 - $M_0 \rightarrow$ 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

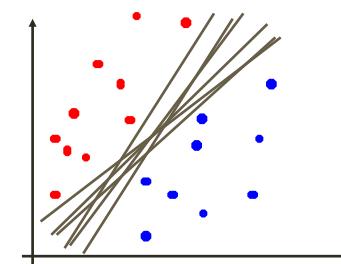
Audio classification

- 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

SVM

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



EBU706U

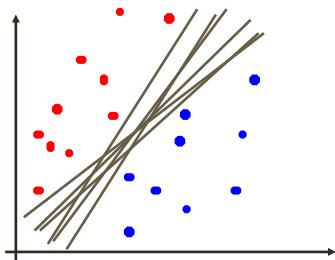
Audio classifier

- 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

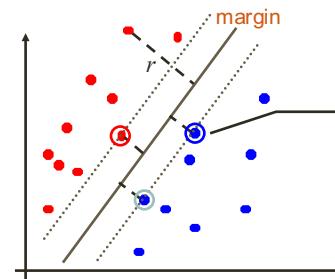
- Binary classification
 - can be viewed as the task of separating classes in the feature space
 - which separator (**hyper-plane**) is optimal?



EBU706U

SVM

- 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

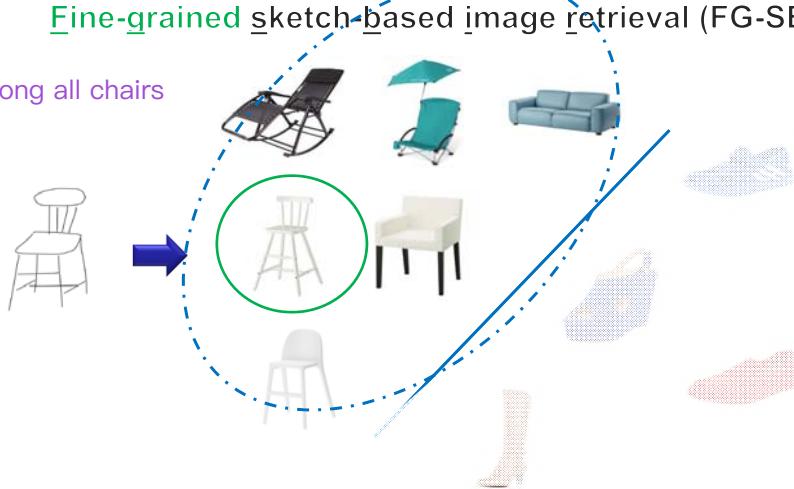
SketchX



The Problem

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

find THE chair among all chairs



Conventional sketch-based image retrieval (SBIR)

SketchX



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.



SketchX

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



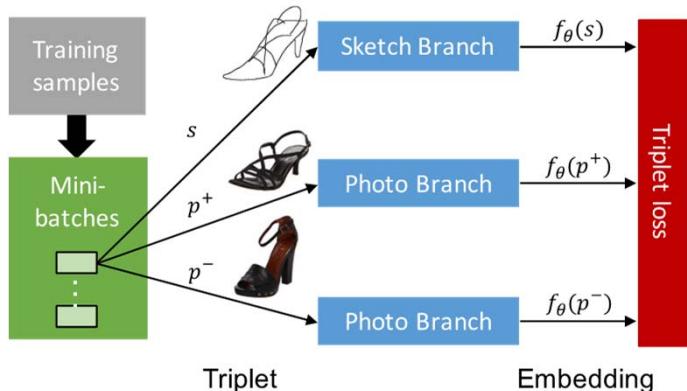
SketchX

[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):



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



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

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



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**

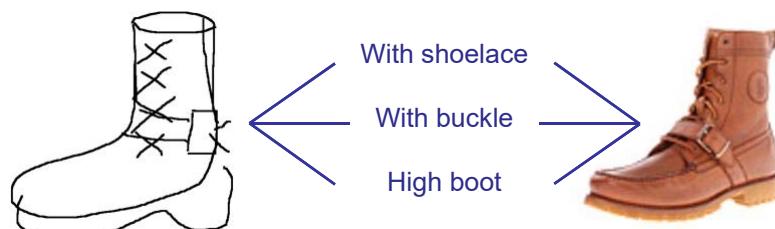


SketchX

Queen Mary
University of London

Why Attribute?

- A semantic bridge between sketch and photo



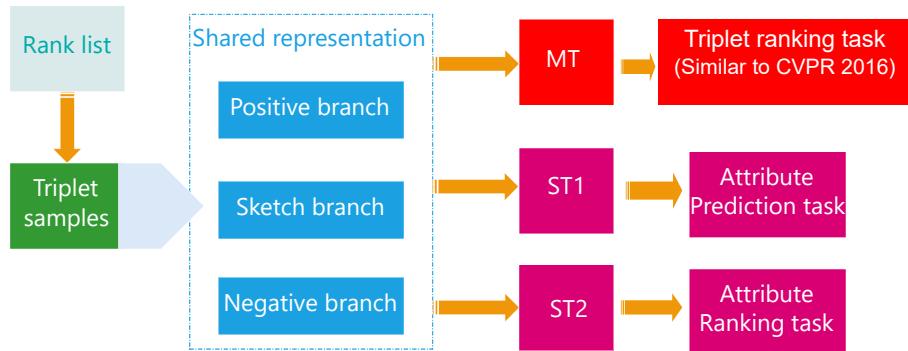
Semantic level ✓

Narrow cross-domain gap ✓

SketchX

Queen Mary
University of London

➤ How we use attributes

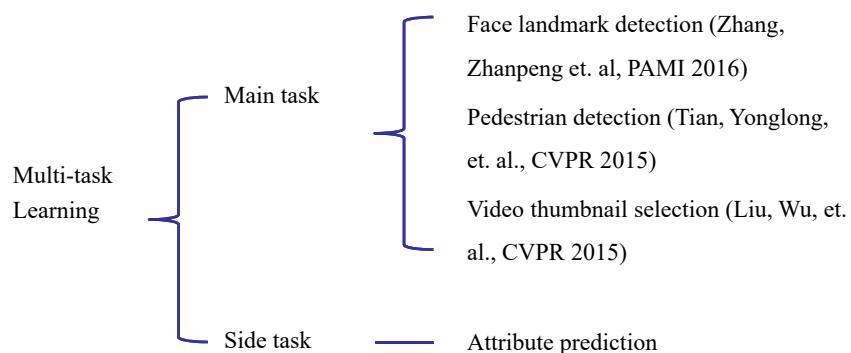


SketchX



Multi-task Learning

➤ Multi-task Learning:

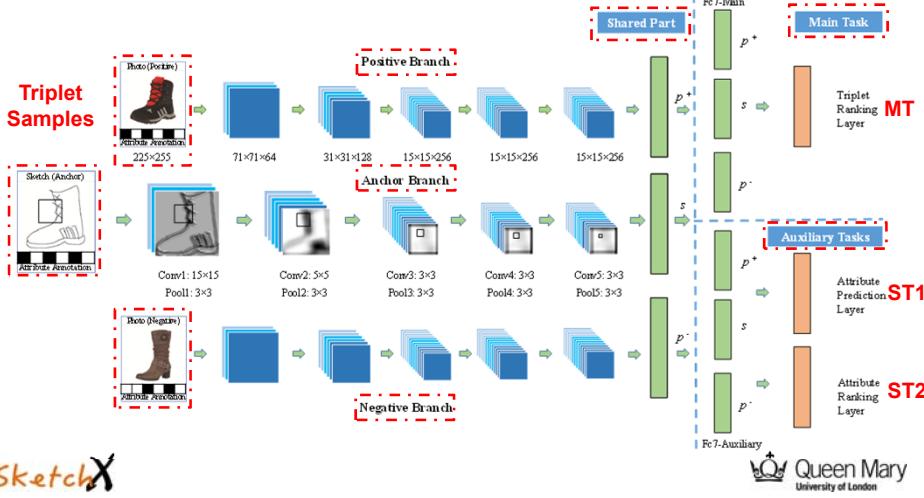


SketchX



Methodology

➤ Overall architecture:



Methodology

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

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

- Attribute prediction task:

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

- Attribute ranking task:

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

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

Cross-entropy distance between sketch and positive photo attributes

Cross-entropy distance between sketch and negative photo attributes

SketchX

Queen Mary
University of London

Multi-task Ranking Loss

- Multi-task training loss:

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

- Multi-task ranking score:

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

Deep Feature Ranking Score
Deep Attribute Ranking Score

SketchX



Contribution

Attribute-driven side task
with attribute ranking layer

Automatically generated chain-like triplets using attributes

Deep Feature Rank

Deep Attribute Rank



SketchX



Automatic Triplet Annotation using Attributes

➤ Comparison between triplet annotation cost:

- 1) CVPR 2016: 32,000 triplets, 100h
- 2) Ours: 7,160 triplets, automatically generated

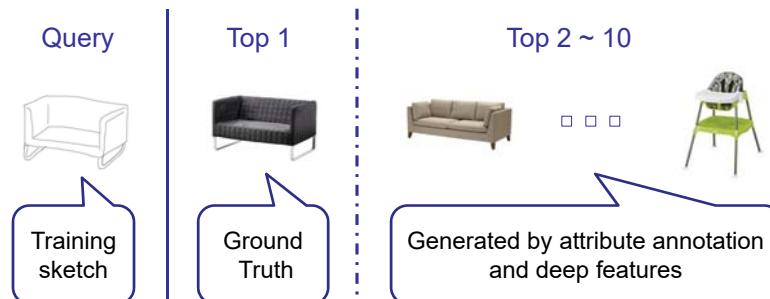
➤ Advantage of proposed annotation strategy:

- 1) Save **90%** human annotations
- 2) Improve **5%** top 1 retrieval accuracy

SketchX

Queen Mary
University of London

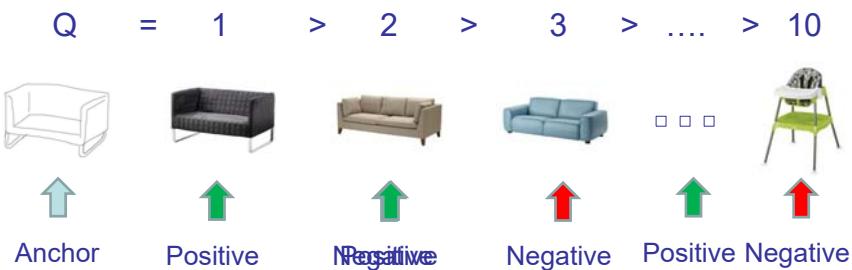
Generated Reference Rank List



SketchX

Queen Mary
University of London

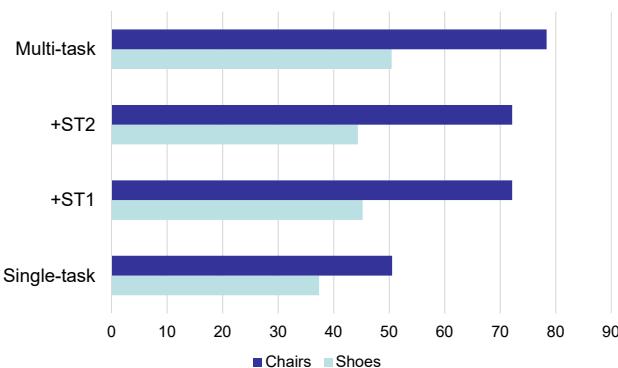
Chain-like Hard Triplets



SketchX



Contribution of Different Components

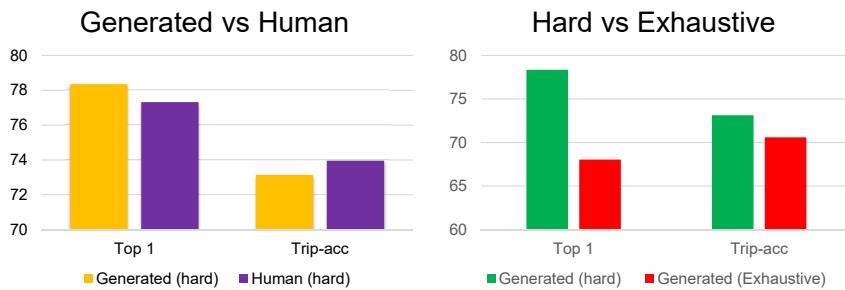


- ST1: attribute prediction task. ST2: attribute ranking task.

SketchX



Effect of Hard Triplet Annotations

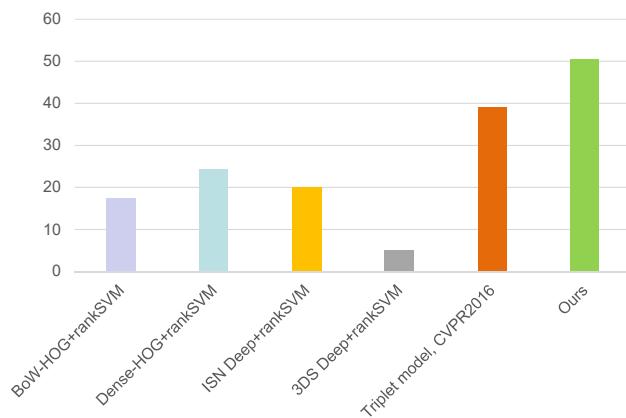


- Performance: Generated triplets \approx human triplets.
- Hard triplets > exhaustive triplets.

SketchX

Queen Mary
University of London

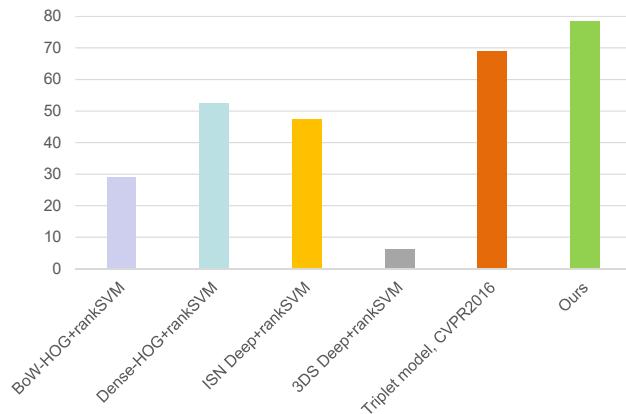
Top 1 Accuracy (Shoe Dataset)



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

	Our Multi-task Model						Triplet Model (CVPR 2016)					
Query												
Top 5 Ranked ↓												

SketchX

