SHIATSU

Semantic-Based Hierarchical Automatic Tagging of Videos by Segmentation using Cuts

Ilaria Bartolini, Marco Patella, and Corrado Romani

MultiMedia DataBases Group

DEIS - University of Bologna

Italy

Florence, Italy, October 29, 2010

The Problem

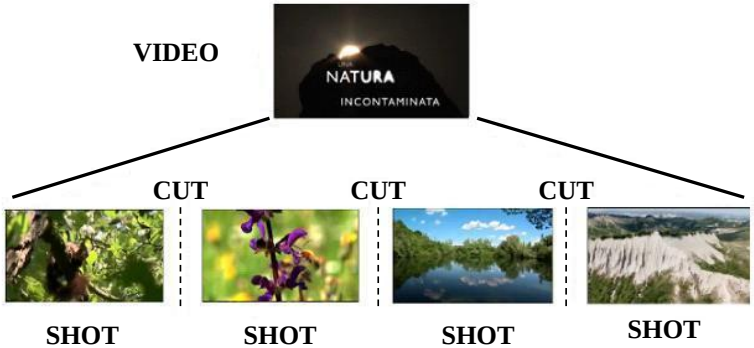
The dramatic widespread of multimedia content over the Internet and other media channels points the interest of media broadcasters to the topics of video retrieval and content indexing

- Video indexing by context analysis and/or manual processing are common solutions, but
  - context could be missing or simply unrelated with video content
  - manual processing is a highly time consuming task
- Recent studies focus on video indexing/retrieval techniques based on visual content
  - analysis of video keyframes features and correlation
  - extraction of (semantic) concepts
  - we are skipping audio (for the moment)
- Semantic concepts (tags) are assigned as meaningful labels to videos so as to enable a high-level, concept-based retrieval

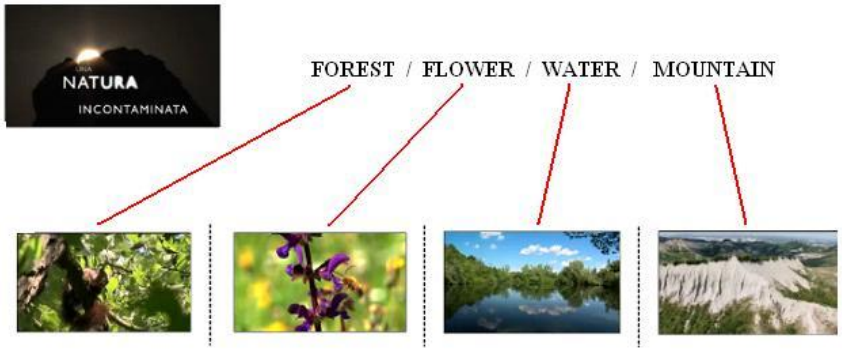
Video Annotation

The task is usually decomposed into two major phases

- Video segmentation detection aims to divide a video into scenes sharing the same visual features in order to help content analysis



- Video semantic indexing has the purpose of extracting semantic concepts from video data in order to provide a compact description



Video Segmentation

Since videos can hold a wide range of subjects, detection of hard cuts and gradual transitions is used in order to segment a video object into sequences sharing the same visual content

- The task involves the computation of visual differences between consecutive frames and the application of a criteria to detect a scene cut (transition)
- Shot detection algorithms use visual features (color histograms, objects edges, motion vectors, wavelet filters) to describe keyframes as feature vectors
- Differences between those vectors (thus keyframes) are evaluated against fixed/dynamic thresholds or using clustering techniques

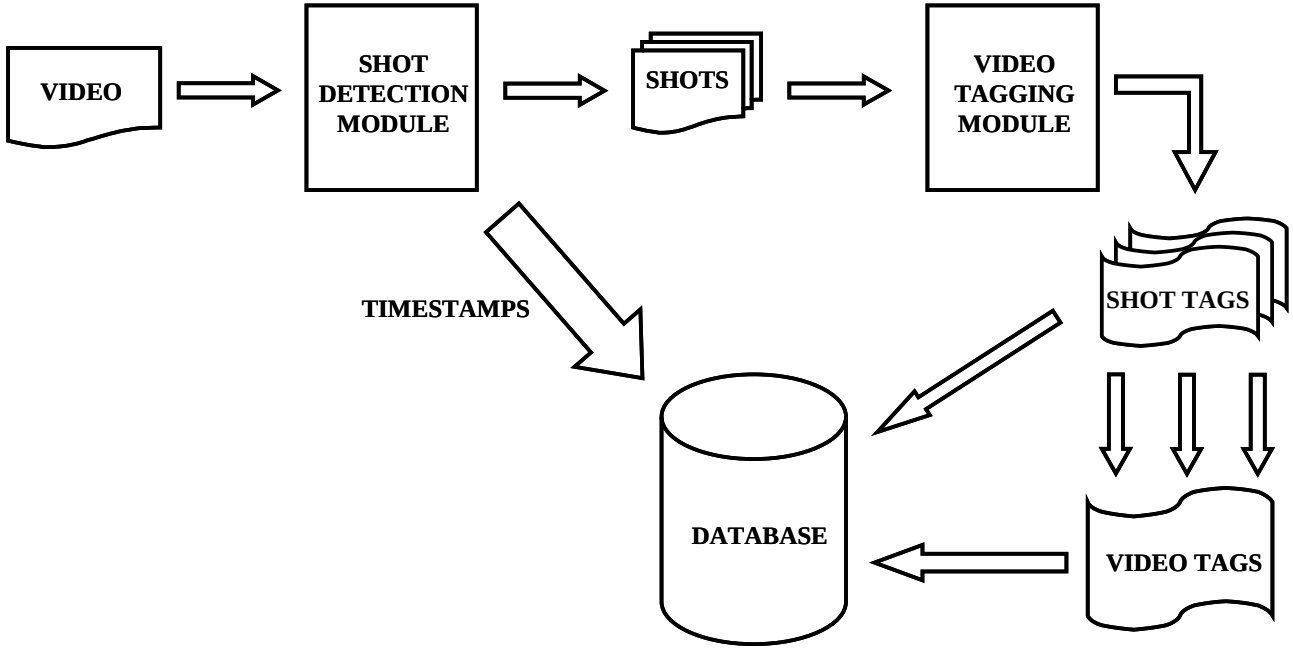
Video Tagging

- The task of (semi-)automatically tagging videos based on semantic concepts is usually carried out using annotation of key frames, so as to detect and categorize objects inside video scenes

- Key frames annotation usually involves system training by means of a set of images that are manually annotated
  - labels associated to the “best” images are proposed for labeling

- Image annotation aims to overcome the so-called “Semantic Gap”, i.e. the perceptual distance between visual features and the inherent meaning of the image/video

SHIATSU Overview



- SHIATSU consists of two main components: the Shot Detection Module and the Video Tagging Module
- They work sequentially to segment a video into coherent frame sequences and to attach semantic concepts to them
- Shot tags are then propagated to the whole video, so as to obtain semantic indices for both the video and its shots

Shot Detection Module

- Our shot detection algorithm computes color histograms and object edges of every frame
- Features are used to compare consecutive frames by applying two different distance metrics: usually a shot transition produces a change in both the color and the texture structure of the frames
- Shot selection process is done with a double dynamic threshold system which takes into account video content in order to adapt to different video types
- Frames are filtered on their color features and then on their edge features

HSV Color Histogram

- We characterize the color information of each frame as the distribution of HSV values of its pixels
  - each frame is represented using 3 histograms (12 bins for Hue, 5 for Saturation, and 5 for Value)

- The difference between histograms of two given frames is computed using a L_1 bin-to-bin metric

$$d_{b2b}(h_1, h_2) = \frac{1}{2N} \sum_i | h_1[i] - h_2[i] |$$

N denotes the number of frame pixels

- HSV distance between two consecutive frames *k* and *k+1* is defined as the sum of bin to bin difference for the three HSV histograms

$$d_{HSV}(k, k+1) = \frac{1}{6N} \sum_{i=1}^{22} | h_k[i] - h_{k+1}[i] |$$

Edge Change Ratio

- Edges of frame objects usually change across a shot boundary, we then compute the quantity of entering edges and the quantity of exiting edges between consecutive frames to detect a shot cut
  - Entering edges are those new edges which have appeared with respect to the previous frame
  - Exiting edges are those edges which are present in the actual frame but not in the next frame

- We compute edge pixels using a Canny filter and determine entering and exiting edges by analyzing edge pixels between frames

- The Edge Change Ratio (ECR) is calculated as follows

$$ECR(k,k+1) = \max\left(\frac{X_k^{OUT}}{\sigma_k}, \frac{X_{k+1}^{IN}}{\sigma_{k+1}} \right)$$

σ_k is the number of edge pixels in frame k, X_{IN} and X_{OUT} are the entering and exiting edge pixels in frames k and k+1 respectively

Cut Selection Process

- Once all the frames have been processed (*O(N)* complexity), the system calculates a double threshold based on frame features

- HSV threshold (θ_{HSV})

$$\theta_{HSV} = \frac{\beta_{HSV}}{M / fr} \sum_{i=1}^{M/fr} L_{HSV}(i)$$

β_{HSV} is a sensitivity parameter (default 1), M is the number of video frames, fr is the framerate and L_{HSV} is the (descending) ordered list of HSV distances among all shots

- ECR threshold (θ_{ECR})

$$\theta_{ECR} = \frac{\beta_{ECR}}{2M / fr} \sum_{i=1}^{2M/fr} L_{ECR}(i)$$

β_{ECR} is a sensitivity parameter (default 1) and L_{ECR} is the (descending) ordered list of ECR values

- Frames that exceed both thresholds are selected as shot cut frames

Video Tagging Module

- The tagging module of SHIATSU exploits the Imagination system and uses a set of pre-annotated images as a knowledge base

- Semantic concepts in the knowledge base can be organized into either (multiple) tree-shaped taxonomies (terms are linked with a parent/child relationship) or as a flat structure (all terms at the same level) and can be easily modified and expanded
  - Relationships in a taxonomy-shaped set of concepts (ontology?) can be exploited during retrieval

- Tagging of key frames is based on the intuition that similar images are usually tagged with the same labels
  - In its essence, the Imagination system exploits a Pagerank-like algorithm for tagging shots

Shot Tagging

- Using shot cuts timestamps, SHIATSU extracts a set of key frames of each shot sequence, computes their visual features and compares them with those contained in the knowledge base
  - For efficiency purposes, the M-Tree index is exploited to efficiently retrieve images having features similar to each keyframe

- The module suggests a set of concepts for each key frame: only terms recurring in the majority of key frames are selected as suitable concepts to describe the whole shot sequence

- To avoid producing an overwhelming number of tags for each shot, only the most frequent tags retrieved for each key frame in the sequence are maintained

- The proposed tags can then be reviewed by the user so as to verify them and are finally stored into the database

Hierarchical Tagging

- Shot tags are useful to browse sequences across different videos, but they could be too specific to index a whole video
- The system selects video tags from the set of shot tags, depending on their frequency and the length of the shot they are associated to

- Rank *R(t)* for every shot tag *t* is defined as

$$R(t) = \frac{1}{N_s} \sum_s W(s) A(t,s)$$

N_s is the total number of shots, W(s) is the shot length and A(t,s) is 1 if shot s contains tag t, 0 otherwise

- Tags are ordered by descending *R(t)* values and the first 10 tags (if available) become video tags
- Concepts extracted from long shots and/or that appear in several ones are more relevant to describe the content of a whole video

Experimental Setup

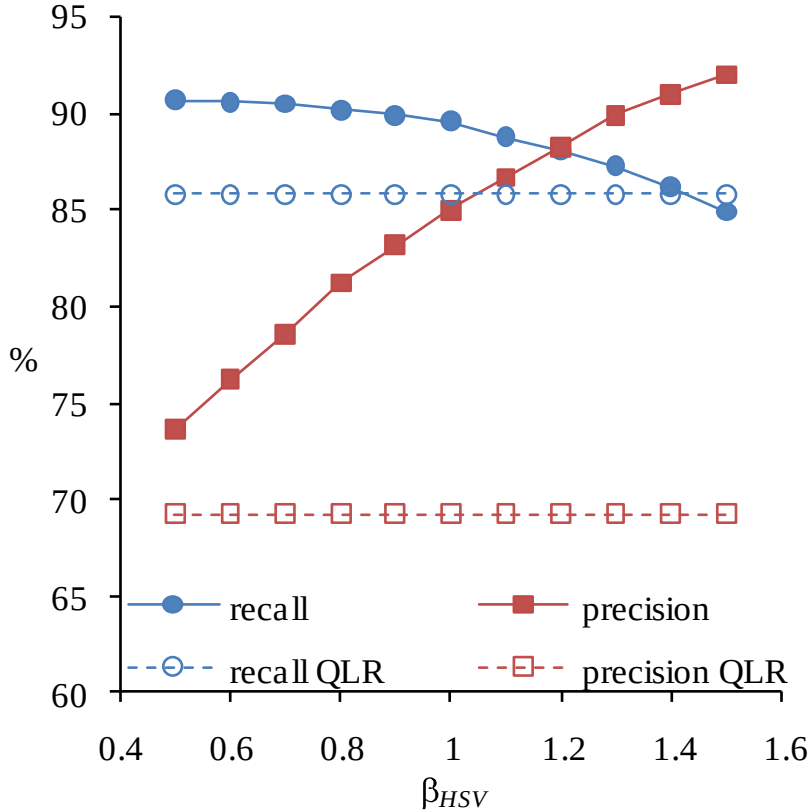
- The Segmentation Module was tested on the video set for the Item Segmentation task of the Mammie system
  - The benchmark consists of 43 videos containing a total of 1.935.000 video frames (4225 shot cuts) for a playback time of over 1075 minutes

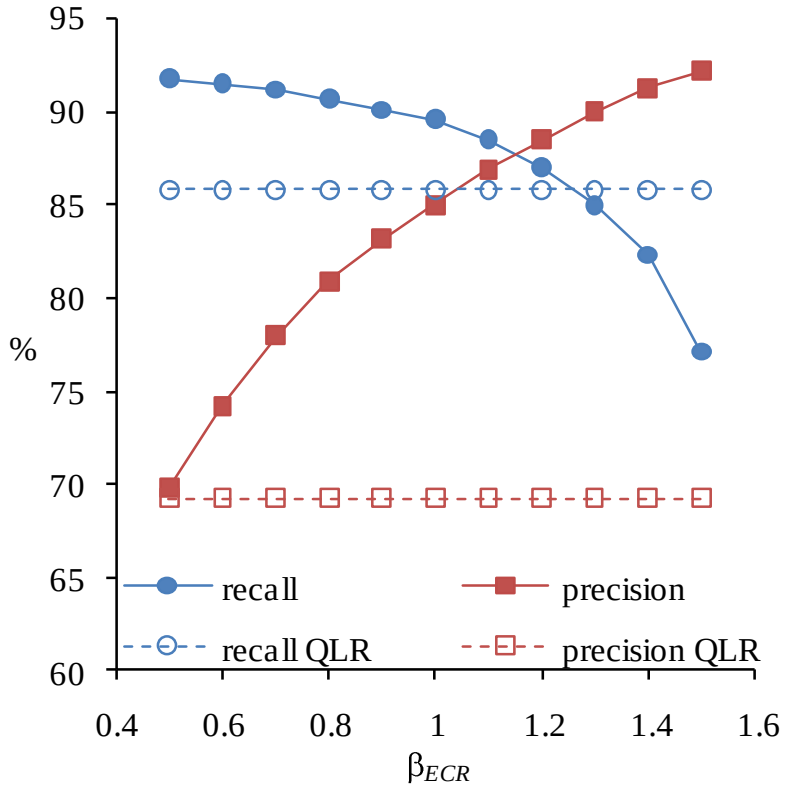
- The Tagging Module was tested on the TRECVID benchmarks
  - We used the knowledge base of the TRECVID-2007 High-Level Feature task (110 videos, 50 hours total length, 21500 key frames, 9000 shot cuts). Each shot is described by tags coming from 36 semantic concepts
  - We tested the module against 7 videos from the TRECVID-2007 High-Level Feature task test set

- Standard Precision *P* and Recall *R* metrics were used to evaluate system performances (*CC* is number of correct results, *MC* the false negatives and *FC* the false positives)

$$P = \frac{CC}{CC + FC} \qquad R = \frac{CC}{CC + MC}$$

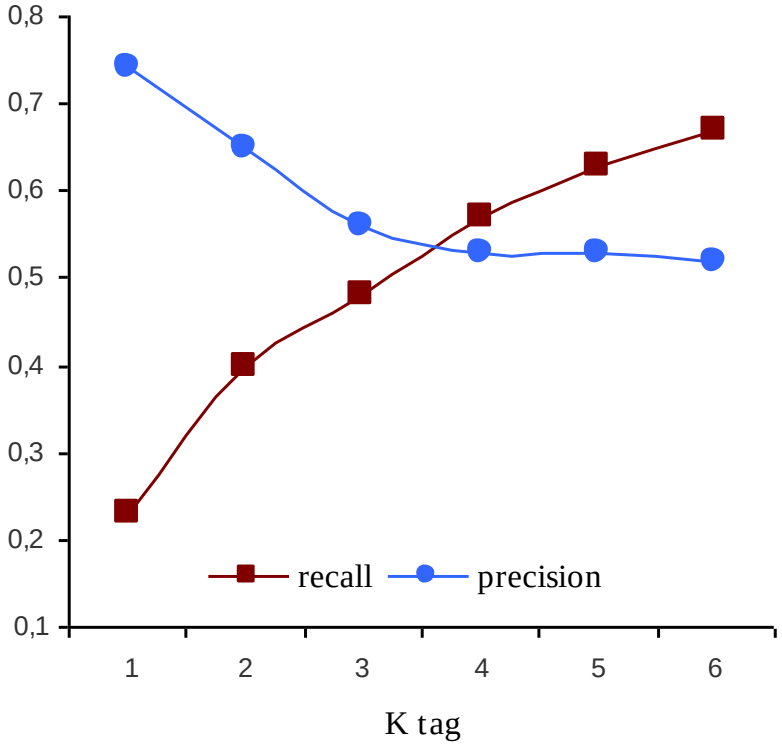
Segmentation Performance

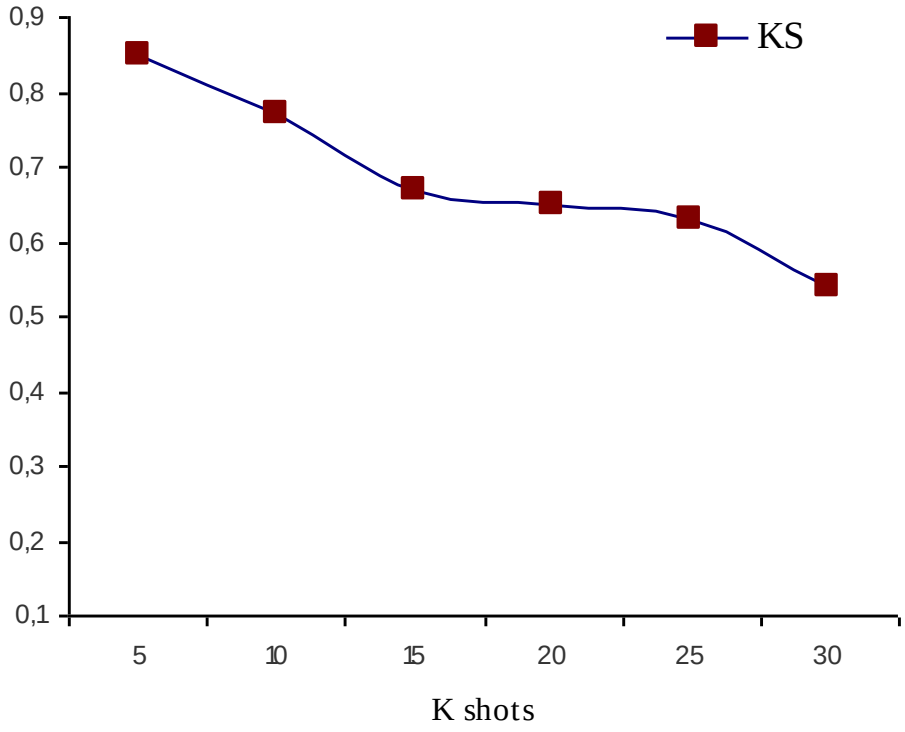




- The first graph shows P/R values when varying β_{HSV} (β_{ECR}=1)
- The second graph shows P/R values when varying β_{ECR} (β_{HSV} =1)
- QLR refers to the competitor segmentation algorithm

Tagging Performance





- The first graph shows *P*/*R* tagging values depending on the number *k* of retrieved tags
- The second graph shows *P* retrieval values depending on the number *k* of retrieved shots

Conclusions and Future Directions

- We have presented SHIATSU, a tool for the (semi-)automatic hierarchical tagging of videos
  - Video tags are derived from tags assigned to video sequences (shots)
  - Video shots are obtained by comparing consecutive frames on color and edges using a double dynamic thresholding system
  - Shots tags are obtained using a Pagerank-like algorithm that exploits a knowledge base of already tagged images
- Our experiments, conducted on two video benchmarks, show promising results for effectiveness
  - Both segmentation and tagging modules are accurate
- Results on effectiveness are particularly impressive when one considers efficiency issues
  - Results on a low-end system show that segmentation of a video requires a time almost equal to the video length
  - Tagging of a scene is usually performed in real-time
- We are considering clustering of keyframes in order to reduce the complexity of tagging
- Do you want to put audio in the loop? Definitely yes!