Keyframe Extraction using Local Visual Semantics in the form of a Region Thesaurus

Evaggelos Spyrou and Yannis Avrithis
National Technical University of Athens
Image, Video and Multimedia Laboratory
Zographou, 15773 Athens, Greece
espyrou@image.ece.ntua.gr

Abstract

This paper presents an approach for efficient keyframe extraction, using local semantics in form of a region thesaurus. More specifically, certain MPEG-7 color and texture features are locally extracted from keyframe regions. Then, using a hierarchical clustering approach a local region thesaurus is constructed to facilitate the description of each frame in terms of higher semantic features. The feature is consisted by the most common region types that are encountered within the video shot, along with their synonyms. These region types carry semantic information. Each keyframe is represented by a vector consisting of the degrees of confidence of the existence of all region types within this shot. Using this keyframe representation, the most representative keyframe is then selected for each shot. Where a single keyframe is not adequate, using the same algorithm and exploiting the coverage of the visual thesaurus, more keyframes are extracted.

1 Introduction

During the last few years, rapid advances in hardware and telecommunication technologies in combination with the world wide web proliferation have boosted wide scale creation and dissemination of digital visual content and stimulated new technologies for efficient searching, indexing and retrieval in multimedia databases. The traditional keyword-based annotation approaches have started to reveal severe disadvantages. Firstly, this manual annotation of digital content appears a very tedious and time consuming task due to the exponential increasing quantity of digital images and videos in all sort of databases (web, personal databases, professional databases and so on) and also because "images are beyond words" [15], that is to say their content can not be fully described by a list of words. For this problems certain content-based retrieval algorithms have been proposed to support efficient image and video retrieval.

Many content-based retrieval and indexing systems have been presented, such as the QBIC [8], Virage [5], VisualSeek [9], MARVEL [7], MediaMill [16] and so on. In the same context, the MPEG-7 standard [2] has become the first standard to allow interoperable searching, indexing, filtering and browsing of audio-visual (AV) content and unlike its predecessors, focuses on non-textual description of multimedia content aiming to provide interoperability among applications that use audio-visual content descriptions.

However, efficient implementation of content-based retrieval algorithms require more meaningful representation of visual contents. Many works exist in the area of keyframe extraction for video summarization. For example, in [6] keyframes are extracted in a sequential fashion via thresholding. A more sophisticated scheme based on color clustering can be found in [18]. In [1], a stochastic framework for keyframe extraction is presented. In [12] a summarization scheme that performs based on simulated users experiments is presented.Finally, in [3] keyframe selection is performed by capturing the similarity to the represented segment and preserving the differences from other segment keyframes.

This paper is structured as follows: Section 2 presents the approach we follow in order to efficiently describe the visual frame properties, using a locally extracted visual thesaurus. More specifically, in subsection 2.1 we present the segmentation algorithm that divides the image in regions, in subsections 2.2 and 2.3 the MPEG-7 visual descriptors that are extracted from image regions, in section 3 we present our approach on the extraction of a local region thesaurus and the representation of each frame using it. Then, in section 4.1 we present our approach in keyframe extraction for representing the semantic content of a video shot. Finally, experimental results are presented in section 5 and conclusions and plans for future work are drawn in section 6.

2 Low-Level Feature Extraction

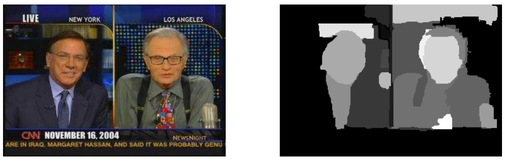
For the representation of the low-level features of a given image, descriptors from the ISO/IEC MPEG-7 standard [2] have been used. This section presents the descriptor extraction procedure followed within our approach.

2.1 Video Frame Segmentation

For the extraction of the low-level features of a still image and more specifically in our case of a given video frame, there are generally two categories of approaches:

- Extract the desired descriptors *globally* (from the entire video frame)
- Extract the desired descriptors *locally* (from regions of interest within the video frame)

While global descriptor extraction appears a trivial task, extracting descriptors locally may turn out a more complex task, since there does not exist neither a standardized way of dividing a given image to regions, from which the features are to be extracted, nor a predefined method to combine and use those features. In the presented approach, a color segmentation algorithm is first applied on a given image as a pre-processing step. The algorithm is a multiresolution implementation [1] of the well-known RSST method [13] tuned to produce a coarse segmentation. This way, the produced segmentation can intuitively provide a brief qualitative description of the image. To make this easier to understand, an input video frame along with its coarse segmentation is depicted in figure 1



(a) Input frame

(b) Segmentation

Figure 1. A video frame and its segmentation.

After segmenting each frame in a small number of regions, low-level descriptors of color and texture features are extracted from each region separately, as presented in subsections 2.2 and 2.3.

2.2 Color Features

For the representation of the color features of the image regions, three MPEG-7 color descriptors are used: The *Color Layout Descriptor*, the *Scalable Color Descriptor* and the *Color Structure Descriptor*. For the extraction of the aforementioned descriptors, the eXperimentation Model (XM)[14] of the MPEG-7 is used. More specifically:

Color Layout Descriptor (CLD) represents the spatial distribution of color in the YCbCr color space by dividing the input region of interest into $8 \times 8 = 64$ blocks and extracting the average color of each block. Then, few low-frequency DCT coefficients of its transformation are selected, forming the CLD after quantization.

Scalable Color Descriptor (SCD) is a Haar-transform based encoding scheme that measures color distribution over an entire image or region of interest in the HSV color space.

Color Structure Descriptor (CSD) captures both global color features of an images and the local spatial color structure. An 8×8 structuring element scans the image counting the number of times a certain color is found within it.

2.3 Texture Features

To efficiently capture the texture features of an image, the MPEG-7 Homogeneous Texture Descriptor (HTD) [11] is applied, since it provides a quintative characterization of texture. The image is first filtered with orientation and scale sensitive filters and the mean and standard deviations of the outputs are computed in the frequency domain. The frequency space is divided in 30 channels and the energy and energy deviation of each channel are computed and logarithmically scaled.

3 Region Thesaurus Construction

Given the complete set of regions of all video frames within a shot and their extracted low-level features as described in section 2, one can easily observe that semantically similar frames consist of visually similar regions and semantically similar regions have similar low-level descriptions.

As it becomes obvious, this region similarity can be exploited as region co-existences often characterize semantically a still image or video frame. In the following section we try to exploit this observation and build a local region thesaurus to facilitate the association of low- with high-level features.

3.1 Hierarchical Clustering

As it appears rather obvious, one cannot have a priori knowledge for the exact number of the required classes to capture the underlying semantic structure of a shot. In our approach, we adopt *Hierarchical clustering* [4] and apply

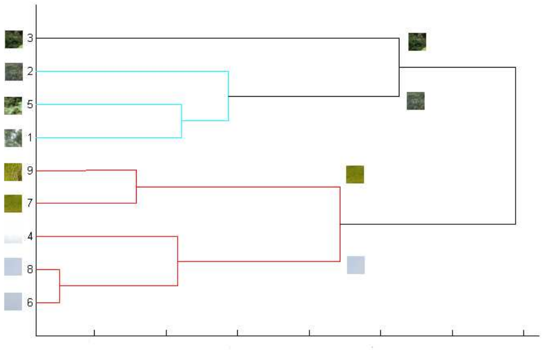


Figure 2. A dendrogram showing region type selection using hierarchical clustering

it on the low-level description set, since after the clustering, we can easily select the number of clusters to keep and easily modify it.

In hierarchical clustering the data are not partitioned into a particular cluster in a single step. Instead, a series of partitions takes place, which may run from a single cluster containing all objects to N clusters each containing a single object. We select an agglomerative approach which proceeds by series of fusions of the N objects into groups.

A simplistic, yet explanatory example from the application of hierarchical clustering on a small number of images and for different numbers of clusters is depicted on figure 3.1. The clustering process starts with 9 different regions and groups them into pairs. The binary tree that results facilitates the determination of the number of the clusters and allows easy modification of this choice without re-application of the method on the data. In the presented case, the desired number of clusters is set to 4 and the final occurring region types are those that constitute the region thesaurus.

3.2 Region Thesaurus

Generally, a thesaurus combines a list of every term in a given domain of knowledge and a set of related terms for each term in the list which are the synonyms of the current term. In our approach, the constructed Region Thesaurus contains all the Region Types that are frequently encountered within the training set. These region types are the centroids of the clusters and all the other feature vectors of a cluster are their synonyms. By using a significantly large training set of keyframes, our region thesaurus is constructed, providing a formalization of the conceptualization between the low and the high-level features, thus facilitating their association.

Each region type is represented as a feature vector that contains all the extracted low-level information for it. As it is obvious, a low-level descriptor does not carry any semantic information. It only constitutes a formal representation of the extracted visual features of the region. On the other hand, a high-level concept carries only semantic information. A region type lies in-between those features. It contains the necessary information to formally describe the color and texture features, but can also be described with a “lower” description than the high-level concepts and a “higher” that the low-level features. I.e., one can easily describe a region type as “a green region with a coarse texture”.

3.3 Model Vector Formulation

Having calculated the distance of each region (cluster) of the image to all the words of the constructed thesaurus, we describe the semantic content of a given keyframe by forming the model vector that semantically describes the visual content of the image in terms of the region types that consist the thesaurus.

The model vector has the size of the region thesaurus and is formed by keeping the smaller distance for each high-level concept. More specifically, let: $d_i^1, d_i^2, ..., d_i^j, i = 1, 2, \ldots, N_S$ and $j = N_C$, where N_C denotes the number of the region types, N_S the number of the frame segments and d_i^j is the distance of the i-th region of the clustered image to the j-th region type. Then, the model vector D_m is the one depicted in equation 1.

$$D_m = [min\{d_i^1\}, min\{d_i^2\}, ..., min\{d_i^{N_C}\}], i = 1, 2, \ldots, N_S \qquad (1)$$

4 Keyframe Extraction

A shot (sometimes depicted as “scene”) is a continuous segment of visual data within a video that shows consistency with respect to certain low-level feature properties. These properties can be either audio, visual or their combination. It is important to notice that this definition of shots does not use semantic features. The majority of the summarization algorithms exploits certain low-level features and creates a frame description that relies on them.

Generally, video summarization algorithms may be divided in two major categories[10]:

- **Image Storyboards**: Within this approach the general framework consists of determining/applying appropriate audiovisual features to represent the images, clustering based on those representations and keyframe selection based on some criteria.

- **Visual Skims**: The difference of skims to the storyboards is that they consist of small video clips instead of a number of selected frames (keyframes).

In our approach we follow the first category and aim to select one or more representative keyframes for each shot. The visual description we adopt is the one presented in section **??**, which uses a locally (within the shot) extracted visual thesaurus. Then we define some measures for deciding which frame(s) will be selected for representing the entire shot.

We should note here that our representation based on local region features is more close to a semantic description, since it relies on all the region types that consist the local region thesaurus and as we have already mentioned, the region types carry semantic information.

4.1 Representative Keyframe Selection

As it has been extensively described in the previous subsection, the visual content of a given video frame is modeled using the MPEG-7 low level features extracted from each of its segments and the aid of the region thesaurus that results from the entire video shot. The vector that captures the description of a frame is referred to as "model vector". The first thing we have to define is a distance function to compare the model vector a frame with that from any given frame within its shot.

One of the most popular distance functions that are used for comparing such descriptions that have the form of a vector is the well-known Euclidean distance. Let f_1 and f_2 denote two video frames:

$$f_1 = \left[d^1_{rt_1}, d^1_{rt_1}, \ldots, d^1_{rt_N}\right] \tag{2}$$

$$f_2 = \left[d^2_{rt_1}, d^2_{rt_1}, \ldots, d^2_{rt_N}\right] \tag{3}$$

Then, their distance $D(f_1, f_2)$ is calculated by:

$$D(f_1, f_2) = \sum_{i=1}^{N} \left(f_1\left(i\right) - f_2\left(i\right)\right)^2 \tag{4}$$

Where d^i_j denotes the distance of the j-th region type to the i-th frame of the shot, f_i the model vector that describes the i-th frame.

Within the first step of our algorithm we consider all the model vectors belonging to the same cluster and we find the centroid of the cluster. Then, the model vector that is closest to the centroid is selected. This way, we extract the "most representative keyframe", that is the one that describes uniquely the semantic properties of a shot in terms of its local region thesaurus as it carries semantic information based on the majority of the region types that are encountered within the shot.

This representation, as it is obvious, is not always able to capture efficiently all the visual and semantic content of a shot. However, certain applications require this simplistic single-keyframe shot representation.

4.2 Extraction of more Representative Keyframes

Some applications such as high-level concept detection in video sequences sometimes require more than one keyframes from each shot in order to be applied efficiently. That is because most of the times a high-level concept is not present to all the frames of the shot. When the available video content becomes in large quantities, the application of a such algorithms can become very slow and not efficient when performed on every frame individually. Moreover, video summarization and retrieval applications are more efficient when a shot is represented by a small set of frames rather than a single keyframe. For those aforementioned reasons, most of one keyframes should be extracted from a given shot, trying both to capture all possible semantic entities and keep their number as small as necessary to facilitate such tasks.

In the presented case, our algorithm is enriched with a further step to overcome this problem. After extracting the most representative keyframe of a shot which will be referred to as "RKF", we compare the "coverage" of its semantic content to the entire region thesaurus. We define this coverage by using the following equation:

$$Cov(RKF, RT) = \sum_{i=1}^{card(RT)} \left(1 - d_i^{RKF}\right) \tag{5}$$

Where RT denotes the local region thesaurus, $card(RT)$ the cardinality of the region thesaurus (the number of the selected region types) and $1 - d_i^{RKF}$ is the confidence that the i-th region type is contained within the selected RKF, where obviously d_i^{RKF} is the min distance among all the segments of the RKF and the i-th region type.

If this coverage is below a user-defined threshold, then our algorithm extracts more keyframes using the following procedure: We define the "size" of a region type as the cardinality of its synonyms:

$$Size(rt) = card(rt) \tag{6}$$

First we select all the frames that contain the largest cluster of the region thesaurus that is not encountered within the representative keyframe. The criterion with which we decide if a region type is contained within the keyframe is its distance to be lower than a predefined threshold. Then we follow the same procedure for all the frames that contain the smallest cluster of the region thesaurus that is also not encountered. This can be explained since within a summary it is of equal importance to present both common and rare region types, since the first give a broader perception of the visual content of a shot, while the latter may present important semantics that are not encountered within the whole shot duration but only in a small part.

In these extracted subsets of the initial shot, we follow the same procedure as the one applied in the whole shot for the extraction of the additional keyframe. This way, we extract one ore more representative keyframes which aim to capture the visual semantics that the RKF failed and also give emphasis to the most rare among them. Then, we repeat again the coverage of the region thesaurus this time by combining the region types within the RKF and the additional keyframes depicted as $NRKF_i$:

$$Cov(\{RKF, NRKF\}, RT) = \sum_{i=1}^{card(RT)} \left(1 - max\{d_i^{RKF}, max\{d_i^{NRKF}\}\}\right) \tag{7}$$

Where $max\{d_i^{NRKF}\}$ is the maximum distance of all selected NRKFs until this step of the algorithm.

Then, either we repeat the aforementioned procedure to include more keyframes, or we finish the keyframe extraction process for this shot.

5 Experimental Results

In this section we present some preliminary results of our proposed method. For the sake of a more clear presentation, we used a small video clip rather than an actual shot, since in this case, the keyframes are more heterogeneous and the keyframe selection more difficult.

We have used a small part of a video clip taken from TRECVID 2006 [17] Development Data. This clip is approximately 167 seconds long. In figure 3 we present in brief the visual content of this clip. Then, in figure 4 we present the most representative keyframe according to our algorithm. Finally, in figure 5 the NRKFs extracted are depicted in order of importance.

6 Conclusions - Future Work

In this work we have presented a method for extracting keyframes from video shots based on their semantic content. We modeled this semantic content using a visual thesaurus that was extracted locally from each shot. Initial results appear promising.

Future work will emphasize on expanding the algorithm to include intershot relations between keyframes. And also shot modeling based on local semantics. Moreover, the keyframe extraction algorithm will be combined with a still image classification scheme and allow it to expand its functionality efficiently in video documents.

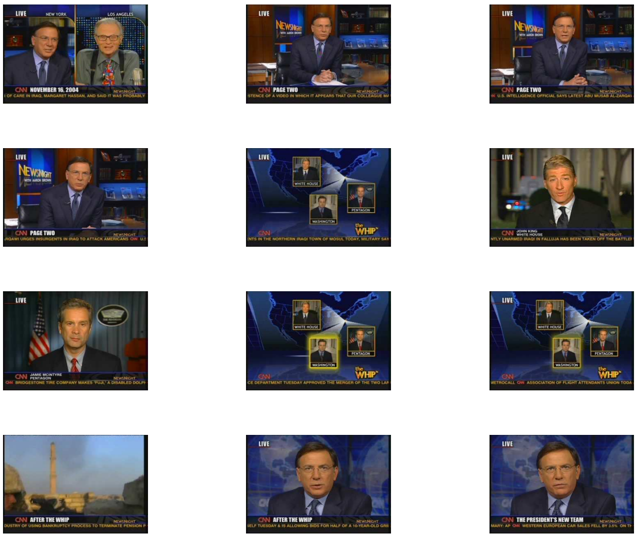


Figure 3. Characteristic Frames of a small Video Clip

7 Acknowledgements

Evaggelos Spyrou is partially funded by PENED 2003 Project Ontomedia 03ED475.

References

[1] Y. Avrithis, A. Doulamis, N. Doulamis, and S. Kollias. A stochastic framework for optimal key frame extraction from mpeg video databases. 1999.

[2] S.-F. Chang, T. Sikora, and A. Puri. Overview of the mpeg-7 standard. *IEEE trans. on Circuits and Systems for Video Technology*, 11(6):688–695, 2001.

[3] M. Cooper and J. Foote. Discriminative techniques for keyframe selection. In *IEEE International Conference on Multimedia and Expo, July 6, 2005*.

[4] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley Interscience, 2 edition, 2000.

[5] A. Hamrapur, A. Gupta, B. Horowitz, C. Shu, C. Fuller, J.Bach, M.Gorkani, and R.Jain. Virage video engine. In *SPIE Proceedings of Storage and Retrieval for Video and Image Databases V, San Jose, CA, February 1997, pp. 188-197*.

[6] H.J.Zhang, J.Wu, D.Zhong, and S.W.Smoliar. An integrated system for content-based retrieval and browsing. 1997.

[7] IBM. Marvel: Multimedia analysis and retrieval system.

[8] IBM. Qbic(tm) - ibm's query by image content.

[9] J.R.Smith and S.F.Chang. Visualseek: a fully automated content-based query system. In *Proc. of ACM Multimedia Conference, Boston, MA, November 1996, pp. 87-98*.

[10] Y. Li, S.-H. Lee, C.-H. Yeh, and C.-C. Kuo. Techniques for movie content analysis and skimming: tutorial and overview on video abstraction techniques.

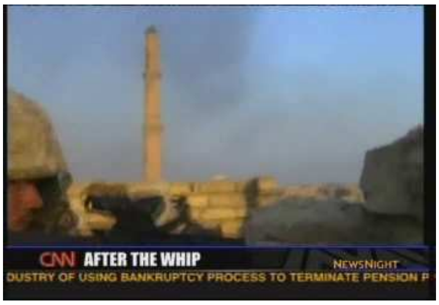


Figure 4. Extracted RKF

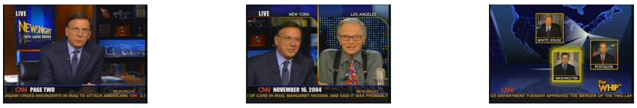


Figure 5. Extracted NRKFs

[11] B. Manjunath, J. Ohm, V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE trans. on Circuits and Systems for Video Technology*, 11(6):703–715, 2001.

[12] B. Mérialdo, B. Huet, I. Yahiaoui, and F. Souvannavong. Automatic video summarization. In *International Thyrrenian Workshop on Digital Communications, Advanced Methods for Multimedia Signal Processing September 8th - 11th, 2002, Palazzo dei Congressi, Capri, Italy*, Sep 2002.

[13] O. J. Morris, M. J. Lee, and A. G. Constantinides. Graph theory for image analysis: An approach based on the shortest spanning tree. 1986.

[14] MPEG-7. Visual experimentation model (xm) version 10.0. ISO/IEC/ JTC1/SC29/WG11, Doc. N4062, 2001.

[15] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE t. PAMI*, 22(12):1349–1380, 2000.

[16] C. G. M. Snoek, D. Koelma, J. van Rest, N. Schipper, F. J. Seinstra, A. Thean, and M. Worring. MediaMill: Searching multimedia archives based on learned semantics. In *Proceedings of the IEEE International Conference on Multimedia & Expo*, Amsterdam, The Netherlands, July 2005.

[17] TREC. - video retrieval evaluation. http://www-nlpir.nist.gov/projects/t01v/.

[18] Y.T.Zhuang, Y.Rui, T.S.Huang, and S.Mehrotra. Adaptive keyframe extraction using unsupervised clustering. 1998.