Multimed Tools Appl (2009) 41:337–373
DOI 10.1007/s11042-008-0237-9

Concept detection and keyframe extraction using a visual thesaurus

Evaggelos Spyrou · Giorgos Tolias · Phivos Mylonas · Yannis Avrithis

Published online: 8 November 2008

Abstract This paper presents a video analysis approach based on concept detection and keyframe extraction employing a visual thesaurus representation. Color and texture descriptors are extracted from coarse regions of each frame and a visual thesaurus is constructed after clustering regions. The clusters, called region types, are used as basis for representing local material information through the construction of a model vector for each frame, which reflects the composition of the image in terms of region types. Model vector representation is used for keyframe selection either in each video shot or across an entire sequence. The selection process ensures that all region types are represented. A number of high-level concept detectors is then trained using global annotation and Latent Semantic Analysis is applied. To enhance detection performance per shot, detection is employed on the selected keyframes of each shot, and a framework is proposed for working on very large data sets.

Keywords Concept detection · Keyframe extraction · Visual thesaurus · Region types

E. Spyrou (✉) · G. Tolias · P. Mylonas · Y. Avrithis
School of Electrical and Computer Engineering,
National Technical University of Athens,
Iroon Politechniou 9 Str., Zographou Campus,
157 73 Athens, Greece
e-mail: espyrou@image.ntua.gr

G. Tolias
e-mail: gtolias@image.ntua.gr

P. Mylonas
e-mail: fmylonas@image.ntua.gr

Y. Avrithis
e-mail: iavr@image.ntua.gr

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 (web, personal databases, professional databases and so on). This trend has also enabled propagation of content adaptation and semantics' aspects throughout the entire multimedia analysis value chain. In the process, the value of the richness and subjectivity of semantics in human interpretations of audiovisual media has been clearly identified. Research in this area is extremely important because of the overwhelming amount of multimedia information available and the very limited understanding of the semantics of such data sources. However, in spite of the multitude of those activities, there is a lack of appropriate outlets for presenting high-quality research in the prolific and prerequisite field of multimedia content analysis. More specifically, the traditional keyword-based annotation approaches have started to reveal severe disadvantages. Firstly, manual annotation of digital content appears a very tedious and time consuming task, due to the exponential increasing quantity of digital images and videos and also because "images are beyond words" [32], that is to say that their content can not be fully and efficiently described by a set of words. For these problems, certain content-based retrieval and detection algorithms have been proposed to support efficient image and video analysis and understanding.

However, the well-known "semantic gap" [32] often characterizes the differences between descriptions of a multimedia object by discrete and heterogeneous representations and the linking from the low- to the high-level features. Therefore, one of the most interesting problems in multimedia content analysis remains the detection of high-level concepts within multimedia documents. These high-level features may either characterize a multimedia document globally, e.g. an image depicting an *indoor* or an *outdoor* scene or locally e.g. concept *sand* detected in a *beach* scene. This problem is often referred to as *Scene/Global Classification*. On the other hand, local high-level features may be distinguished in two major categories. The first contains those that are often denoted as *materials*, since they cannot have a specific shape, but they are described solely by their color and texture properties. Some examples within this category are *sea*, *sky*, *vegetation*, *road*, etc. The latter contains concepts that may be characterized based mainly on their shape such as *person*, *car*, *airplane* etc. These concepts are often denoted as *objects* and the specific problem as *Object Detection*. The nature of the high-level concepts that have to be detected plays a crucial role in the selection of both appropriate low-level features and applicable machine learning techniques.

In this work, we extend our previous research efforts on high-level concept detection [35, 38] using a region thesaurus of visual words on keyframe extraction, based on a locally extracted (within a video/video shot) region thesaurus [36] and unify them in a framework capable for video analysis and summarization (Fig. 1). More specifically, the objective of this work is to provide a generic approach for the detection of certain high-level concepts within the scope of TRECVID 2007. Detection of a concept within a video shot is achieved by representing each shot with a single keyframe and while trying to enhance the performance of detection, a keyframe extraction algorithm is used on each shot. Representative frames are

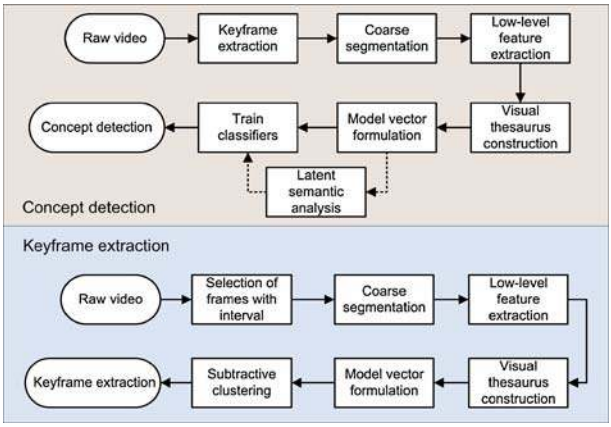

Fig. 1 Keyframe extraction and concept detection by utilizing a visual thesaurus

therefore selected from each shot and the high-level feature extraction algorithm is applied on them. The keyframe extraction algorithm presented herein uses a visual dictionary that is formed within the shot in order to provide a model vector that describes each frame based on the region types it contains. Moreover, a keyframe extraction algorithm is applied in order to provide summarization of a video by selecting a relatively small number of frames, able to catch the visual and semantic properties of a video or a video shot. Thus, our approach emphasizes both content coverage and perceptual quality and is capable of reducing content redundancy.

Using a visual dictionary that is formed based on coarsely segmented regions from keyframes extracted from all the available development data videos, our approach tackles 9 concepts within the TRECVID 2007 collection, using a generic detection approach for all and not specialized algorithms for each one. The concepts that have been selected are *vegetation*, *road*, *explosion_fire*, *sky*, *snow*, *office*, *desert*, *outdoor* and *mountain* and as obvious, they cannot be described as "objects", but rather as "materials" and "scenes". For the extraction of low-level features from these keyframe regions, MPEG-7 descriptors have been selected. Since neither of the aforementioned concepts falls within the category of "objects", color and texture features are the only applicable low-level features. For each concept an SVM-based detector is trained based on features extracted from keyframe regions, while keyframes are annotated globally. The next step of the presented high-level concept detection approach, is to apply the Latent Semantic Analysis (LSA) technique in an effort to exploit the latent relations among the set of keyframes and the region types they contain.

Our approach exploits the novel model vector representation, in order to extract a number of representative keyframes and perform video summarization. Video summarization is essential to enable the user to skim through the content and speed up the browsing of large video databases, instead of traditionally representing the video as a sequence of consecutive frames, each of which corresponds to a constant time interval. This linear representation, though adequate for playing a video in a movie, is not appropriate for the new emerging multimedia services that require new tools and mechanisms for interactive navigating video information over various networks and devices of limited bandwidth. With video summarization, a video table

of contents and video highlights are constructed to enable end users to sift through all this data and find what they want.

The rest of this article is organized as follows: Section 2 deals with current similar research efforts in the field, whereas Section 3 summarizes the extracted MPEG-7 color and texture descriptors and their properties. Therein, from each coarsely segmented region of a keyframe, a *feature vector* is formed containing all the necessary low-level features. Then, in Section 4, a visual thesaurus is formed. The most representative region types are selected. These region types allow a model vector representation of each keyframe's visual features, as described in Section 5. In Section 6 the LSA technique is described. This technique acts as a transformation on the model vector and provides an image representation based not only on the region types but also on their latent relations. The keyframe extraction algorithm using a local visual thesaurus is described in Section 7. In Section 8 our concept detection approach is presented for a single frame and for several keyframes extracted from each shot. Extensive experimental results are presented in Section 9, where the two techniques are compared in a large dataset consisting of 110 videos segmented into shots and derived from the TRECVID 2007 development data and the summarization results of the keyframe extraction are presented. Finally, conclusions and plans for future work are drawn in Section 10.

2 Related work

Due to the continuously growing volume of audiovisual content, the problem of high-level concept detection within multimedia documents has attracted a lot of interest within the multimedia research community during the last years. Many research efforts have set focus on the extraction of various low-level features, such as audio, color, texture and shape properties of audiovisual content, in order to extract meaningful and robust descriptions in a standardized way. Some of the aforementioned works have led to the MPEG-7 standard [5] that focuses on the description of multimedia content, providing among others, a set of low-level descriptors useful for tasks such as image classification, high-level concept detection, image/video retrieval and so on. The MPEG-7 visual descriptors and many similar works aim to extract features globally, or locally, i.e. from regions or image patches. On the other hand, many low-level description schemes are inspired from the SIFT features [19]. Descriptors that fall within this category are locally extracted, based on the appearance of the object at particular interest points.

Furthermore, utilization of machine learning approaches in multimedia processing/manipulation problems is a trend followed by a huge number of works during the last years. More specifically, techniques such as Neural Networks [13], Fuzzy Systems [16], Genetic Algorithms [23] and Support Vector Machines [40] have been successfully applied to the aforementioned problems, in order to link the low-level features to the high-level features. In almost all of these approaches, an extracted low-level description of (part of) a multimedia document is fed to an appropriately trained machine learning-based detector/classifier, which makes the decision of the presence or absence of the concept in question.

Another important aspect of any high-level concept detection problem is the availability of annotated training data and also their annotation. The number of

available and annotated globally data sets has been increased during the last few years. An example falling in this category is the *LSCOM workshop annotation* [26]. Therein, a very large number of shots of news bulletins is globally annotated for a large number of concepts. A common annotation effort [2] has been a result of cooperation among many TRECVID 2007 participants for a large number of shots of various cultural TV programmes. We should also note here that large region-based annotated sets have started to appear, such as *LabelMe* [29] and *PASCAL* [11], however these sets offer only a few thousands of annotated images, while the LSCOM and the collaborative TRECVID annotation offer tens of thousands of annotated images.

The idea of using a visual dictionary to describe a decomposed image that derived from a clustering or a segmentation or a keypoint extraction approach has also been exploited by many researchers. For instance, an image is divided into regions using a segmentation algorithm, a visual dictionary is formed and a region-based approach in content retrieval using LSA is presented in [34]. Therein, image regions are regarded as words and LSA aims in exploiting the latent (hidden) relations amongst them. A similar approach is presented in [30]. Therein, the pixels of a given image are clustered using a mean-shift algorithm. Then, color and texture features are extracted from the formed clusters and low-level features are assigned to the high-level concepts using again a visual dictionary. In [12], images are partitioned in regions, regions are clustered to obtain a codebook of region types, and a bag-of-regions approach is applied for scene representation. Moreover, in [9] visual categorization is achieved using a bag-of-keypoints approach.

One of the most well-known systems for multimedia analysis and retrieval, MARVEL, is presented in [14]. This prototype system uses multi-modal machine learning techniques in order to model semantic concepts in video, from automatically extracted multimedia content. Also, in [41], a region-based approach is presented, that uses knowledge encoded in the form of an ontology. MPEG-7 visual features are extracted and combined and high-level concepts are detected. Moreover, a hybrid thesaurus approach is presented in [3], where semantic object recognition and identification within video news archives is achieved, with emphasis to face detection and TV channel logos. Finally, in [28], separate shape detectors are trained using a shape alphabet, which is actually a dictionary of curve fragments.

Another lexicon design for semantic indexing in media databases is also presented in [27]. In the same context, [18] presents an approach for texture and object recognition that uses scale- or affine-invariant local image features in combination with a discriminative classifier. Support vector machines have been used for image classification based on their histogram as in [6] and for the detection of semantic concepts such as *goal*, *yellow card* and *substitution* in the soccer domain [33]. A Self-Organized Map (SOM) that uses MPEG-7 features is presented in [17]. Within this work, content-based image and information retrieval is achieved in large non-annotated image databases. Learning algorithms are compared and novel fusion algorithms are explored in [44], while detectors for 374 high-level concepts are presented in [43].

However, efficient implementation of content-based retrieval algorithms requires a more meaningful representation of visual contents. In this context many works exist in the area of keyframe extraction for video summarization. For example, in [45] keyframes are extracted in a sequential fashion via thresholding. A more

sophisticated scheme based on color clustering can be found in [46]. Avrithis et al. [1] presents a stochastic framework for keyframe extraction while in [22], a summarization scheme based on simulated users experiments is presented. A multimodal approach for video summarization is presented in [20]. Finally, in [8] keyframe selection is performed by capturing the similarity to the represented segment and preserving the differences from other segment keyframes.

Last but not least, evaluation and comparison to similar techniques has always been important for every research work. Among others, special note should be given to an effort to effectively evaluate and benchmark various approaches in the field of information retrieval, by the TREC conference series, which has become very popular during the last few years. Within this series, the TRECVID [31] evaluation attracts many organizations and researchers, interested in comparing their algorithms in tasks such as automatic segmentation, indexing, and content-based retrieval of digital video. For the high-level feature detection task of TRECVID, the aforementioned global annotations have been offered by different organizations and a huge database of video keyframes has been available to active participants [2].

3 Image decomposition and low-level feature extraction

We begin by initially presenting the descriptor extraction procedure followed within our approach. The fist step to consider is the extraction of low-level features from regions of still images selected from raw video. A given video is segmented into shots. For the keyframe extraction approach, a significantly large number of frames are selected from each shot with a manually selected and arbitrarily small time interval and then the representative keyframes are determined among them, while for the concept detection and scene classification approach one or more keyframes are selected from each shot and the high-level concepts are extracted within them. Let $k_i \in K$ denote the aforementioned selected video frames (still images) and K the set of all those images.

For the extraction of the low-level features of a still image, there exist 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 to be 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 [25], tuned to produce a coarse segmentation. This way, the produced segmentation can intuitively facilitate a briefly qualitative description of the input image.

To make this easier to understand, a given image, along with its coarse segmentation is depicted in Fig. 2. Therein, one can intuitively describe the visual content of this image either in a high-level manner (i.e. the image contains *sky*, *road* and

Fig. 2 Input frame and segmentation result

vegetation) or in a lower level, but higher than a low-level description (i.e. a *"light blue" region*, a *"grey" region*, and two *"green" regions*. To begin, let R denote the set of all regions, resulted after the aforementioned segmentation, and let $R(k_i) \subset R$ denote the set of all regions of the frame k_i.

For the representation of the low-level features of a given image, the well known MPEG-7 standard [5] has been selected. In particular, several MPEG-7 color and texture descriptors [21] have been used to capture the low-level features of each region $r_i \in R(k_i)$. More specifically, *Dominant Color Descriptor* (DCD), *Color Structure Descriptor* (CSD), *Color Layout Descriptor* (CLD) and *Scalable Color Descriptor* (SCD) are extracted to capture the color properties and *Homogeneous Texture Descriptor* (HTD) and *Edge Histogram Descriptor* (EHD) the texture properties.

Dominant Color Descriptor (DCD) is one of the most useful MPEG-7 descriptors and probably the most useful for applications such as similarity retrieval using color, as a set of dominant colors in a region of interest or in an image provide a compact, yet effective representation. The descriptor comprises of the dominant colors' values, their percentages and variances and the spatial coherency. Before the evaluation of the descriptor, the colors present in an image are clustered in order to have a small number in the remaining colors. This clustering is followed by the calculation of their percentages and optionally their variances. It is important to mention that these colors are not fixed in the color space but are computed each time based on the given image. The spatial coherency is a single number that represents the overall spatial homogeneity of the dominant colors in an image. The method of the dominant color extraction is described in detail in [42]. Each image could have up to a maximum of 8 dominant colors, however experimental results show that 3–4 colors are generally sufficient to provide a good characterization of the region colors.

Color Layout Descriptor (CLD) is a compact and resolution-invariant MPEG-7 visual descriptor designed to represent the spatial distribution of color in the YCbCr color space. It can be used globally in an image or in an arbitrary-shaped region of interest. The given picture or region of interest is divided into $8 \times 8 = 64$ blocks and the average color of each block is calculated as its representative color. However the representative color of each block is only implicitly recommended to be the average color. A discrete cosine transformation is performed into the series of the average colors and a few low-frequency coefficients are selected using zigzag scanning. The CLD is formed after quantization of the remaining coefficients, as described in [42]. In conclusion, the CLD is an effective descriptor in applications such as sketch-based image retrieval, content filtering using image indexing and visualization.

Color Structure Descriptor (CSD) captures both the global color features of an image and the local spatial structure of the color. The latter feature of the CSD

provides the descriptor the ability to discriminate between images that have the same global color features but different structure, thus a single global color histogram would fail. An 8×8 structuring element scans the image and the number of times a certain color is found within it is counted. This way, the local color structure of an image is expressed in the form of a "color structure histogram". This histogram is identical in form to a color histogram, but is semantically different. Let $c_0, c_1, c_2, \ldots, c_{M-1}$ denote the M quantized colors. Then the color structure histogram can be declared as: $h(m), m = 0, 1, \ldots, M-1$. The value in each bin represents the number of occurrences of structuring elements as they scan the image, that contain at least one pixel with color c_M. The color representation is given in the HMMD color space.

Scalable Color Descriptor (SCD) is a Haar-transform based transformation applied across values of a color histogram that measures color distribution over an entire image. The color space used here is the HSV, quantized uniformly to 256 bins. The histogram values are extracted, normalized and nonlinearly mapped into a four-bit integer representation, giving higher significance to small values. To sufficiently reduce the large size of this representation, the histograms are encoded using a Haar transform which provides the desired scalability when the full resolution is not required.

Homogeneous Texture Descriptor (HTD) provides a quintative characterization of texture and is an easy to compute and robust descriptor. This descriptor is computed by first filtering the image with a bank of orientation and scale sensitive filters, and computing the mean and standard deviation of the filtered outputs in the frequency domain. The frequency space is divided in 30 channels, as described in [42], and the energy e_i and the energy deviation d_i of each channel are computed. These two values are logarithmically scaled to obtain e_i and d_i respectively, where i is the i-th feature channel.

Edge Histogram Descriptor (EHD) captures the spatial distribution of edges. It represents local-edge distribution in the image. Specifically, dividing the image in 4×4 subimages, the local edge distribution for each subimage can be represented by a histogram. To generate the histogram, edges in the subimages are categorized into five types, namely vertical, horizontal, 45° diagonal, 135° diagonal and nondirectional edges. Since a given image is divided into 16 subimages, a total of $5 \times 16 = 80$ histogram bins are required. This descriptor is useful for image to image matching, even when the underlying texture is not homogeneous.

To obtain a single region description from all the extracted region descriptions, we choose to follow an "early fusion" approach, thus merging them after their extraction [37]. The vector formed will be referred to as "feature vector". The feature vector that corresponds to a region $r_i \in R$ is thus given by equation (1):

$$f_i = f(r_i) = \Big[DCD(r_i), CLD(r_i), SCD(r_i), CSD(r_i), HTD(r_i), EHD(r_i) \Big], \quad r_i \in R \tag{1}$$

where $DCD(r_i)$ is the Dominant Color Descriptor for region r_i, $CLD(r_i)$ is the Color Layout Descriptor for region r_i etc. Each feature vector is denoted by f_i and F is the set of all feature vectors. In other words: $f_i \in F, i = 1 \ldots N_F = N_R$. Each descriptor is comprised by a vector of a predefined dimension, while for the Dominant Color Descriptor we keep only the values and the percentage of the most dominant color.

Table 1 Dimension of the extracted MPEG-7 color and texture descriptors

Descriptor	DCD	SCD	CLD	CSD	EHD	HTD
Number of Coefficients	4	256	256	18	80	62

Since many of the MPEG-7 descriptors allow the user to select their level of detail, thus offering a large number of available extraction profiles, we follow a procedure similar to the one presented in [24], in order to select the one that best suits the needs of our approach. The dimensions of the extracted descriptors are depicted in Table 1, while the final dimension of the merged feature vector is 676.

The extraction of the low-level descriptors is performed using the Visual Descriptor Extraction (VDE) application. This application is used to extract the selected visual MPEG-7 descriptors from the input images and for all the corresponding regions maps created after the coarse segmentation. VDE is an application developed based on the eXperimentation Model of MPEG-7 [42], using its extraction algorithms and expanding its facilities. It is optimized in order to provide a faster performance than the XM, while it remains fully compatible in terms of the introduced descriptions. It is developed in C++ and tested for Windows. OpenCV computer vision library is used for faster image loading and Xerces is used for the generation of the MPEG-7 XML format output. Certain bugs detected in the XM implementation for arbitrary regions have been fixed in VDE and new features have been made available. To make it easier to understand the way descriptors can be extracted, an initial input image and all possible ways of specifying a region from which descriptors are extracted, are depicted in Fig. 3. Using the VDE application, descriptors may be extracted globally from the entire image, locally from a single image region of a binary mask, from all regions of a region map or finally by providing coordinates of rectangular regions.

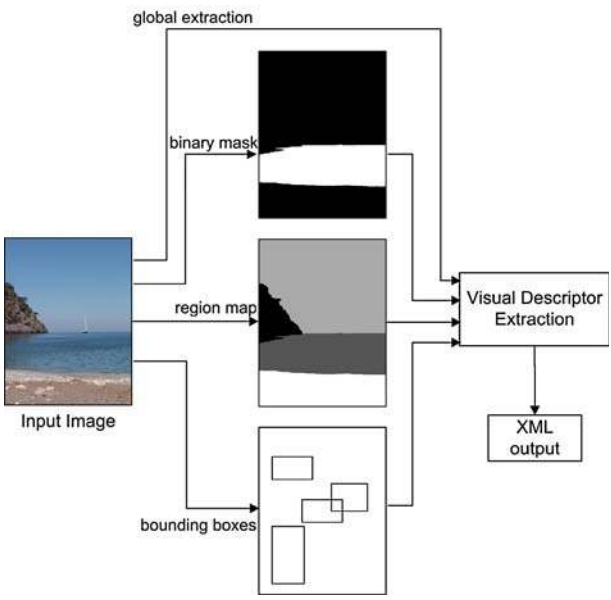

Fig. 3 Multiple ways of extracting descriptor from image regions with the VDE application

The current version of VDE is available for downloading at our research team's web site.[1]

4 Visual thesaurus construction

Given the entire set of video frames and their extracted low-level features as described in Section 3, one can easily observe that those regions that belong to similar semantic concepts, also have similar low-level descriptions and also those images that contain the same high-level concepts are consisted of similar regions. Thus certain similar regions often co-exist with some high-level concepts. In other words, region co-existences should be able to characterize the concepts that exist within a keyframe.

It is eligible that all regions r_i should be organized in order to construct a structured knowledge base. By using this, an image will be represented as a set of regions. Thus, it aims to bridge the low-level features to the high-level concepts. Based on the aforementioned observations, a *subtractive clustering* [7] algorithm is applied on all segmented regions. Let the number of clusters created be N_T. It can be easily observed that each cluster does not contain only regions from the same concept and also that regions from the same concept could end up belonging to different clusters. For example, regions from the concept *vegetation* can be found in more than one clusters, differing e.g. in the color of the *tree* leaves. Moreover regions of the concept *sea* could be mixed up with regions of the concept *sky*, considering that a typical *sea* region is often much similar to a typical *sky* region. The region lying closest to the centroid of the cluster is selected as the representative for each cluster. These regions will be referred to as *region types*. Finally our constructed "knowledge base" has the form of a *Visual Thesaurus*, which is actually a set of "visual words". The visual words will be denoted w_i and the definition of the visual thesaurus is depicted in Eq. 2.

$$T = \left\{ w_i, \quad i = 1, \ldots, N_T \right\}, \quad w_i \subset R \tag{2}$$

where N_T denotes the number of region types of the thesaurus (and, obviously, the number of clusters) and w_i is the i-th cluster, which is a set of regions that belong to R, as it is presented in Eq. 2. Additionally, according to Eqs. 3 and 4, the utilization of all clusters provides the entire R set, if and only if all regions are used for clustering and different clusters do not contain common regions.

$$\bigcup_i w = R, \quad i = 1, \ldots, N \tag{3}$$

$$\bigcap_{i,j} w = \emptyset, \quad i \neq j \tag{4}$$

A *thesaurus* is generally a list of terms (a list of region types in our approach) and a set of related regions to each region type the list. Each region type is selected as

[1] http://www.image.ntua.gr/smag/tools/vde.

the region whose feature vector has the smallest distance from the centroid of the cluster it belongs, as it has already been mentioned before. The calculation of the centroid is depicted in (5) where $|w_i|$ is the number of elements of the i-th cluster and the selection of the region type is depicted in Eq. 6. The related regions (synonyms) of each region type are all the remaining regions of the cluster.

$$z(w_i) = \frac{1}{|w_i|} \sum_{r \in w_i} f(r) \tag{5}$$

$$f(w_i) = f\Big(\arg\min_{r \in w_i} \Big\{ d\big(f(r), z(w_i)\big) \Big\} \Big) \tag{6}$$

Each region type is represented by a feature vector, which contains the fused low-level information extracted from the region. As it is obvious, a low-level description does not carry any semantic information. The region types lie in-between the low-level features and the high-level concepts. They carry the appropriate information to describe color and texture features with a higher semantic ability than low-level descriptions, but yet a step lower than the semantic concepts.

5 Model vector formulation

This section presents the algorithm used to create a model vector to represent the visual properties of a video frame based on the all the visual words (region types) of the visual thesaurus constructed as described in Section 4.

To compare the low-level properties of two image regions, the Euclidean distance has been applied on their feature vectors, as depicted in Eq. 7. Therein, $f_1, f_2 \in \mathcal{F}$ and $F \subset \mathcal{F}$. F denotes the set of feature vectors for the specific set of regions, whereas $\mathcal{F}$ is the entire feature vector space.

$$d\big(f_1, f_2\big) = \sqrt{\sum_{i=1}^{n} \big(f_1^i - f_2^i\big)^2} \tag{7}$$

A model vector has the same dimension as the number of region types that consist the visual thesaurus and is formulated by the following procedure: having calculated all distances between all regions of each image and all region types of the visual thesaurus, the minimum among them is kept for each region type. Thus, the i-th element of a model vector is the minimum distance among w_i and all feature vectors of all regions of the corresponding image.

In particular the model vector describing keyframe k_i, is depicted in equation (17). Each model vector is denoted by $m_i \in M, i = 1 \ldots N_K$, where is the set of all model vectors, m_i is the model vector of frame k_i and N_K is the cardinality of K. More formally:

$$m_i = \Big[m_i(1), m_i(2), \ldots, m_i(j), \ldots, m_i(N_T) \Big], \quad i = 1, \ldots, N_K \tag{8}$$

where:

$$m_i(j) = \min_{r \in R(k_i)} \left\{ d\Big(f(w_j),\, f(r) \Big) \right\}, \quad j = 1, \ldots, N_T \tag{9}$$

Figure 4 presents an example of an image, segmented into 4 regions and a visual thesaurus consisted of 10 region types. The model vector describing this image would be the one depicted in Eq. 10. In the left image, the distances of all segmented regions to region type 4 are depicted. In the right image, the distances between the marked region of *vegetation* and all 10 region types are depicted. Based on the aforementioned model vector calculation algorithm, the corresponding element of the model vector, $m(4)$ would be equal to 0.1. Finally, in Fig. 5 the model vectors for 5 images and the visual thesaurus of 10 region types, are depicted. The lower values of model vectors are highlighted so as to note which region types of the thesaurus are

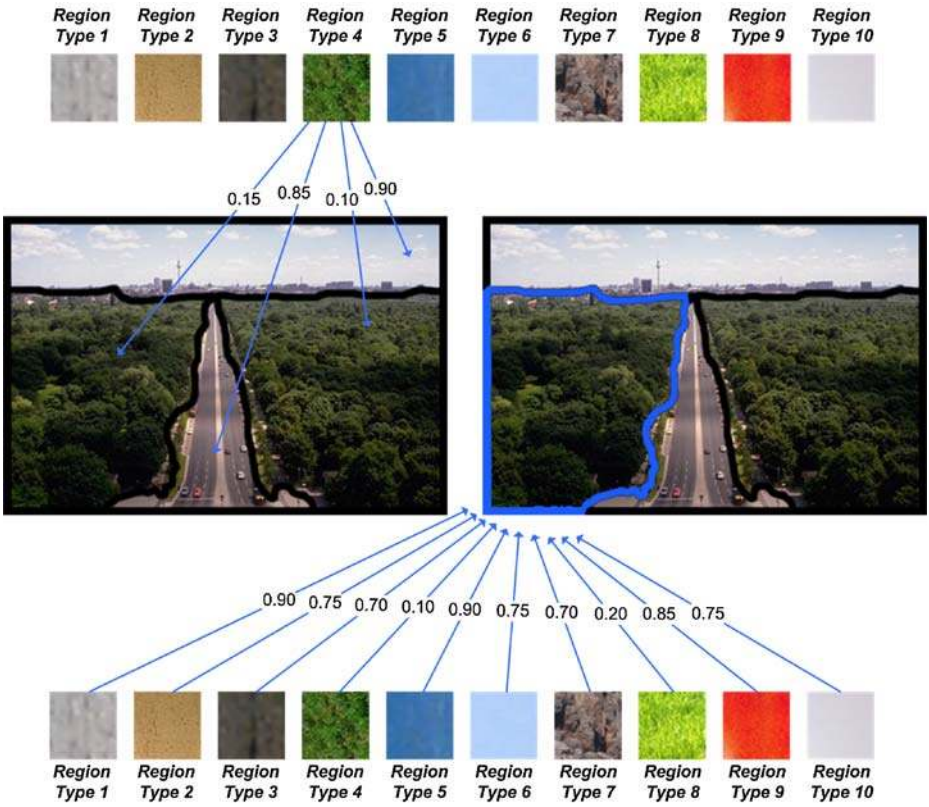

Fig. 4 Distances between regions and region types: on the *top of the figure* distances between an image region and all region types are depicted, whereas on the *bottom*, distances between all regions and a specific region type are depicted

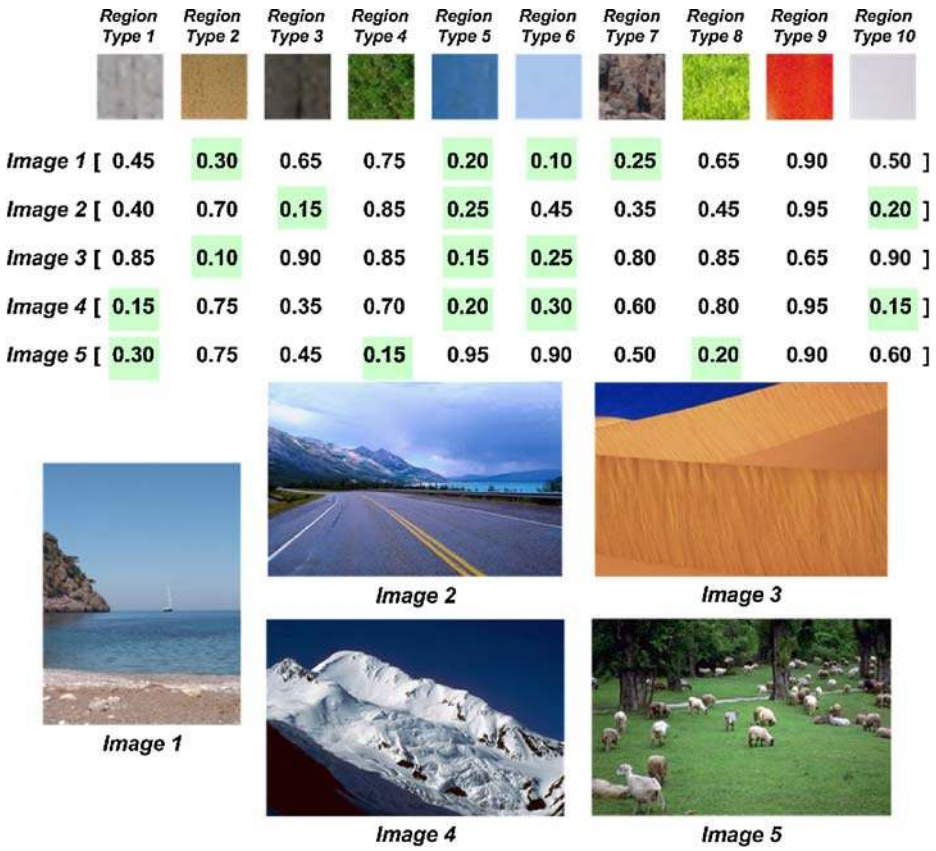

Fig. 5 Indicative selection of images and corresponding model vectors. *Green highlighted values* are the smallest distances

contained within each image. These low values correspond to low distances between the corresponding region types and a region of the image.

$$m = [m(1), m(2), \ldots, m(10)] \tag{10}$$

6 Latent semantic analysis

The next step of the presented high-level concept detection approach, is the use of the well-known LSA technique [10], initially introduced in the field of natural language processing. LSA aims to exploit the latent relations among a set of documents and the terms they contain. In this work, a frame corresponds to a document and its segmented regions correspond to the terms. The goal is to investigate how these hidden relations among region types may be exploited to improve the semantic analysis.

After the formulation of the model vectors m_i, all their values are normalized so that they fall within [0, 1], with 1 depicting the maximum confidence of a region type to a keyframe. The normalized model vectors will be denoted as m'_i.

This way, the co-occurrence matrix $\mathcal{M}$ is formed, as depicted in equation (11), describing the relations of region types to keyframes.

$$\mathcal{M} = \begin{pmatrix} m'_1(1) & \dots & m'_{N_K}(1) \\ \vdots & \ddots & \vdots \\ m'_1(N_T) & \dots & m'_{N_K}(N_T) \end{pmatrix} \tag{11}$$

More specifically, each line of $\mathcal{M}$, $q_i^T = \left(m'_1(i), \dots, m'_{N_K}(i)\right)$, describes the relationship of region type w_i, with each frame k (term vector). Also, each column of $\mathcal{M}$, $m'_j = \left(m'_j(1) \dots m'_j(N_T)\right)^T$ corresponds to a specific frame, describing its relation with every region type (document vector).

Thus, the co-occurrence matrix $\mathcal{M}$ may be described using the extracted (normalized) model vectors m'_i as:

$$\mathcal{M} = \left[m'^T_1, \dots, m'^T_{N_K}\right] \tag{12}$$

Let q_i and q_p denote two term vectors. Then, their inner product $q_i^T q_p$ denotes their correlation. Thus, it may easily observed that $\mathcal{M}\mathcal{M}^T$ actually consists of all those inner products. Moreover, $\mathcal{M}^T\mathcal{M}$ consists of all inner products between the document vectors $m'^T_i m'_p$, describing their correlation over the terms.

A decomposition of $\mathcal{M}$ is described by Eq. 13.

$$\mathcal{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \tag{13}$$

When $\mathbf{U}$ and $\mathbf{V}$ are orthonormal matrices and $\mathbf{\Sigma}$ is a diagonal matrix, This is the Singular Value Decomposition (SVD), depicted in Eq. 14.

$$\mathcal{M} = \left(\mathbf{u_1} \dots \mathbf{u_{N_T}}\right) \begin{pmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_{N_T} \end{pmatrix} \left(\mathbf{v_1} \dots \mathbf{v_{N_T}}\right)^T \tag{14}$$

In this equation σ_i are the singular values and $\mathbf{u}_i$,$\mathbf{v}_i$ are the singular vectors of U and V, respectively.

By keeping the N_L larger singular values $\boldsymbol{\sigma}_i$ along with the corresponding columns of $\mathbf{U}$ and rows of $\mathbf{V}$, an estimate of $\mathcal{M}$, described in Eq. 15 occurs.

$$\hat{\mathcal{M}} = \mathcal{M}_{N_L} = \mathbf{U}_{N_L}\mathbf{\Sigma}_{N_L}\mathbf{V}^T_{N_L} \tag{15}$$

A normalized model vector m'_i, is transformed to the concept space, using $\mathbf{\Sigma}$ and $\mathbf{U}$, as depicted in Eq. 16, where $\hat{m}_i$ is the transformed in the concept space vector.

$$\hat{m}_i = \mathbf{\Sigma}^{-1}_{N_L}\mathbf{U}^T_{N_L}m'_i \tag{16}$$

This way, all model vectors extracted from frames of the training set are transformed to the concept space and are then used to train several high-level concept detectors, as described in Section 8.

7 Keyframe extraction

We extend the notion of the visual thesaurus described approach towards efficient keyframe extraction from video sequences. In the majority of the summarization algorithms, certain low-level features are exploited and a frame description that relies on them is created. More specifically, a relatively small number of representative keyframes is selected, in order to capture the visual content of a video shot. Our research effort lies clearly within the *Image Storyboards* video summarization approach. In principle our approach may be decomposed in the following fundamental steps [39]:

- Determine an appropriate feature space to represent the images
- Cluster the image sequence in the feature space
- Compute a measure of importance to select the keyframes

First, we extract certain MPEG-7 low-level features, as described in Section 3. Then, we form a local region thesaurus (within the video/video shot), following the procedure described in Section 4. The selection of the representative frames is performed based on this region thesaurus and the additional two steps that will be presented within this section.

We should note here that our representation based on local region features is more close to a semantic description than to a visual one. It relies on all the region types that consist the local visual thesaurus and as we have already mentioned, the region types although they are not actual high-level concepts, carry significant semantic information.

As it has already been described in the previous sections, the semantic content of a given video frame is modeled by combining certain MPEG-7 features extracted locally from its segmented regions and with the aid of a locally extracted visual thesaurus mapped to region types, the model vectors are formed. The first thing we have to define is a distance function to compare the model vector of a frame with that from any given frame within its shot.

One of the most popular distance functions used for comparing such descriptions that have the form of a vector and carry semantic information is the cosine similarity function. More specifically, let m_1 and m_2 denote the model vectors of two video frames:

$$m_i = \Big[m_i(1), m_i(2), \ldots, m_i(j), \ldots, m_i(N_T)\Big], \quad i = 1,\ 2 \tag{17}$$

Then, we use the cosine distance function to calculate their distance $D_{cos}(m_1, m_2)$:

$$D_{cos}(m_1, m_2) = \arccos \frac{m_1 \cdot m_2}{||m_1|| \cdot ||m_2||} \tag{18}$$

where $m_i(j)$ has been defined in Section 5.

Within the first step of our algorithm we extract all the model vectors of the frames selected with an interval within the shot (10 frames per second for example). These are the frames from which the visual thesaurus was constructed. Then we apply the *subtractive clustering* [7] algorithm in order to cluster them into groups of semantically similar frames (Eq. 19 depicts these clusters), since this method estimates the number of clusters N_S and their corresponding centroids. This way,

we keep a subset of the frames within the shot. The representative keyframes will be selected among them.

$$S = \left\{ w_i', \quad i = 1 \ldots N_S \right\}, \quad w_i' \subset K \tag{19}$$

We should emphasize here that 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 within all the frames of the shot. When the video content to be analyzed comes in large quantities, the application of such algorithms can become very slow when performed on every frame individually. Thus the number of frames that will be extracted, should contain all the semantic concepts, but should also remain relatively small to allow the application of the current detection algorithms in a large amount of video data.

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. This way, the user is able to perceive the content of the full video document, rather than the one of the presented frame. For those aforementioned reasons, more than 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.

The presented approach is used for the selection of a relatively small number of representative keyframes within a video shot. The role of these keyframes is to provide a video summary and also permit the aforementioned high-level concept detection approach to be applied on them instead of the whole shot. The former is achieved through keeping all those frames whose corresponding model vectors lie closer to the aforementioned subset of cluster centroids. Equation 20 depicts the centroids in the model vector space and Eq. 21 the model vectors closest to them. Set M_Z defined in Eq. 22 is the set of all those model vectors corresponding to the keyframes which will be kept. Let K_z be this set of keyframes. The latter is achieved by keeping the appropriate number of frames, within the subset K_z of the preselected frames, which contain as much as possible information for all region types of the region thesaurus.

$$z(w_i') = \frac{1}{|w_i'|} \sum_{k \in w_i'} m_k \tag{20}$$

$$m(w_i') = m_k, \; k = \arg\min_{k \in w_i'} \left\{ D_{cos}\left(m_k, z(w_i')\right) \right\} \tag{21}$$

$$M_z = \{m(w_i')\}, \; i = 1 \ldots N_S \tag{22}$$

The selection of the initial (most representative) keyframe involves finding the region type with the larger number of synonyms, in other words, this region type whose cluster in the feature vector space has the highest cardinality. The next step is to find the model vector within M_z, which has the highest confidence to contain this region type, i.e. the smallest distance to this region type. Thus if the region type selected is the i-th of the Visual Thesaurus, then the selected model vector among the M_z set is the one for which the value of the i-th element is minimized. For the first selected representative frame and for every next one, it is checked which of the

region types are contained within it. A video frame is supposed to contain a region type if the distance for this particular region type is below a preselected threshold t_s. Let $R_{s,i}$ and $M_{s,i}$ denote the sets of the selected region types and model vectors, at the i-th iteration of the algorithm, respectively. $R_{s,i}$ actually contains the region types' indices. The selected region type and the model vector of the selected frame are added to the initially empty sets $R_{s,0}$ and $M_{s,0}$ respectively, so they cannot be selected again. We should also mention here that all other region types contained in each selected frame are also added in $R_{s,i}$ during the i-th iteration. The set of these region types is depicted in Eq. 24.

Every other frame is then selected using the aforementioned procedure. The model vector with the highest confidence for the region type with the larger number of synonyms, omitting those region types which are contained in the already selected frames. This process ends when all region types of the thesaurus are contained in the selected frames ($|R_{s,i}| = N_T$). It is obvious that the number of the selected keyframes to represent the shot with this approach cannot be more than N_T, which denotes the number of the region types. The set $R_{s,k}$ at the k-th iteration is depicted in Eq. 23, while the determination of the region type that is selected each time is depicted in Eq. 25.

$$R_{s,k} = \{r_i\} \cup \{r_{c,i-1}\},\ i = 0 \ldots k-1,\ R_{s,0} = \emptyset \tag{23}$$

$$r_{c,k} = \{j \in [1 \ldots N_T] :\ \grave{m}_k(j) < t_s\},\ r_{c,0} = \emptyset \tag{24}$$

$$r_k = \arg\max_i(|w_i|),\ i \notin R_{s,k} \tag{25}$$

Also, the set of the selected model vectors at the k-th iteration, $M_{s,k}$ is depicted in Eq. 26 while the calculation of the model vector $\grave{m}_k$ that is selected is depicted in Eq. 27.

$$M_{s,k} = \{\grave{m}_i\},\ i = 0 \ldots k-1,\ M_{s,0} = \emptyset \tag{26}$$

$$\grave{m}_k = \arg\min_m \left(m(r_k)\right),\ m \in M_z,\ m \notin M_{s,k} \tag{27}$$

When our approach for keyframe extraction is used for video summarization, it provides a large number of keyframes to represent the video features. This number is the number of clusters resulted after the subtractive clustering algorithm is applied on the set of model vectors (N_S). Moreover, when it is used for the selection of some representative keyframes within a shot it results to a number of keyframes less or equal to the one of the region types of the thesaurus. Thus, keyframes are selected to carry as much information as possible of the entire visual thesaurus. The latter can be useful in high-level concept detection in video, where a more meaningful representation of the visual content is sometimes required.

8 Visual concept detection

In this section, we present the proposed approach for high-level concept detection in video sequences. The concept detectors are applied on a small set of keyframes extracted from each video shot (as it was described in Section 7) or on a single keyframe representing the shot (following a more simplistic approach).

When working on a single keyframe per shot and after extracting model vectors from all keyframes of the (annotated) training set, an SVM-based detector is trained separately for each high-level concept. For the implementation of the SVM detectors, the well-known LIBSVM [4] library has been used and a polynomial kernel type has been selected. The input of the detectors is either a model vector m_i describing a frame of the input video in terms of the visual thesaurus, or a transformed in the concept space vector $\hat{m}_i$ describing a keyframe in terms of regions co-occurrence, in case that LSA was used. The output of the detector is the confidence that the given video frame contains the specific concept, in other words, a value between 0 and 1. Values of confidence close to 1 indicate a high certainty that the concept is depicted in the video frame, while values close to 0 indicate a low one. It is important to clarify that the detectors are trained based on annotation per image and not per region. The same stands for their output, thus they provide the confidence that the specific concept exists somewhere within the frame in question. Several experiments, presented in Section 9 indicate that the threshold above which it is decided that a concept exists, also varies depending on the classifier and should be determined experimentally, in a separate process for each concept. The confidence output for a single keyframe represented by a model vector m is depicted in Eq. 28 and the binary output after applying a threshold t to the confidence value is depicted in Eq. 29. Instead of m, $\hat{m}$ is used if LSA is applied.

$$c(m) \in [0, 1], \quad m \in M \tag{28}$$

$$d(m) = \begin{cases} 0, & c(m) \leq t \\ 1, & c(m) > t \end{cases} \tag{29}$$

Keyframe extraction (as described in Section 7) is used to select a number of frames able to represent the visual and the semantic content of a video shot. On each video frame, the set of the high-level concept detectors that have already been described are applied. Let N_s be the number of keyframes per shot, which varies between the various shots. If $m_j, \quad j = 1 \ldots N_s$, are the corresponding model vectors to the selected keyframes, then the binary output of the detection per shot by using all extracted keyframes is depicted in Eq. 30, where s is the shot in question.

$$d_s(s) = \begin{cases} 0, & \sum_{j=1}^{N_s} d(m_j) = 0 \\ 1, & \sum_{j=1}^{N_s} d(m_j) > 0 \end{cases} \tag{30}$$

This approach facilitates the high-level concept detection for those concepts that may be depicted within the shot but not in the single keyframe that has been selected to represents the shot content. Moreover, for some other cases, a concept may be depicted in a keyframe, but due to effects like *occlusion* or *blurring*, its detection is difficult, while in other frames of this shot those restraining effects may not exist. Thus, selecting more frames before and after the most representative keyframe increases the possibility of the concepts to be detected in one of them. It is expected that with this approach, the recall of the high-level concept detection will be significantly increased, while leaving precision to similar or slightly higher levels.

Fig. 6 Examples of keyframes containing representative positive examples for each of the detected high-level concepts

9 Experimental results

This section presents the results of the aforementioned techniques for concept detection and keyframe extraction, applied on the TRECVID 2007 Development Data. These data comprise a large dataset consisting of 110 videos, segmented into shots. The shot segmentation is provided to all active TRECVID participants and used herein. A keyframe has been selected from each shot, thus 18113 keyframes have been made available. The annotation used herein, has resulted from a joint effort among several TRECVID participants [2]. For the evaluation of our concept detection approach we selected 9 of the TRECVID concepts. Representative keyframes annotated as positive for these particular concepts are depicted in Fig. 6. Moreover the keyframe extraction algorithm is applied on each shot and several keyframes are selected. Then the high-level feature extraction algorithm is applied on them to enhance detection performance. We used different measures for evaluating the performance of the proposed detection approach, assuming a ground truth notion. Precision, recall and average precision, using the extracted keyframes, were determined and calculated using the ground truth data from the TRECVID 2007 dataset [2]. Our keyframe extraction approach is also used for a video summarization and the selected keyframes are presented as an output of the algorithm. Human evaluators/users were not involved in any evaluation procedure.

The TRECVID 2007 [31] development and test dataset consists of a number of MPEG-1 videos provided by the Netherlands Institute for Sound and Vision. This content is consisted of approximately 400 hours of news magazine, science news, news reports, documentaries, educational programming, and archival video in MPEG-1 format, available for use within the TRECVID benchmark. For the high-level feature extraction task, 50 hours of these data are available to be used for training.[2]

Last but not least and regarding our implementation of the aforementioned algorithms we should note that the segmentation algorithm mentioned in Section 3 is a multi-resolution variation of the RSST algorithm implemented as a prior work

[2]More details can be found in http://www-nlpir.nist.gov/projects/tv2007/tv2007.html#3.

Table 2 Number of positive examples within the development data and the constructed training/testing sets

Concept	Number of positives		
	Development data	Training	Testing
Desert	52	36	16
Road	923	646	277
Sky	2146	1502	644
Snow	112	78	34
Vegetation	1939	1357	582
Office	1419	993	426
Outdoor	5185	3000	1556
Explosion_fire	29	20	9
Mountain	97	68	29

of some of the authors. This research effort is presented in detail in [1], where a potential reader may seek further clarifications. The forthcoming extraction of visual descriptors is performed using the VDE application, which is also described in Section 3. Both the aforementioned implementations are command line tools, whereas the rest of the detection and keyframe extraction algorithms have been implemented in the MATLAB[3] computing environment and programming language.

9.1 Large scale high-level concept detection experiments

Table 2 summarizes the detected concepts and the number of positive examples within the development data and the constructed training/testing sets for each of them. Using the constructed training set, due to the large number of regions derived after segmentation, not all available regions are used. Instead, all regions derived from keyframes that contain at least one of the high-level concepts and an equal number of random regions derived from keyframes that do not contain any of the high-level concepts, are used to form the visual thesaurus. Subtractive clustering is applied, while N_T is the number of the region types created.

First of all, several experiments are performed by varying the ratio λ of negative to positive examples within the given training set. For a heterogeneous and loosely annotated dataset such as TRECVID, it is very difficult to model positive examples of each concept. More specifically, in Fig. 7 some representative examples of the high-level concept *vegetation* both from the TRECVID and the COREL datasets are depicted. It easy to observe that the low-level features of the regions that belong to the specific concept vary a lot in the case of TRECVID, while appear very similar in COREL. More examples are also shown in Fig. 6. Moreover, in TRECVID the annotation is global, i.e. for the entire keyframe, while the concepts cover only a small part of it. As it becomes obvious, a larger number of negative examples is needed, but should be selected appropriately, in order to avoid biasing the detectors towards the negative examples. To test the performance of the differently trained classifiers, a testing set with a ratio $\lambda = 1$, i.e. consisting of an equal number of positive and negative values is used. Results are summarized in Table 3, where the empasized values are the highest average precision values for each concept. It may be easily

[3] http://www.mathworks.com/.

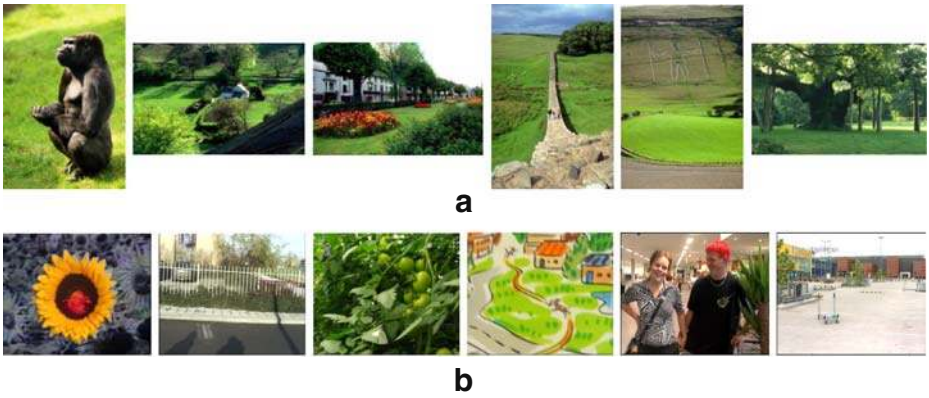

Fig. 7 Example images depicting the concept *vegetation* from COREL (**a**) and TRECVID (**b**)

observed that for almost every concept, a value of λ between 4–5 is appropriate to achieve the highest possible average precision (AP) [15] by the classifiers.

Having selected the training set, experiments on the threshold confidence value for each classifier are performed. As testing set for each concept, the set of all remaining keyframes is used. Precision and recall measures are calculated for each high-level concept and for a range of threshold values, starting from 0 and increasing with a step of 0.1 until they reach the value of 0.9. Then, the threshold value where precision is almost equal to recall is selected (Fig. 8). This way, both measures are kept in equally good values, as it is generally desirable. Curves for concepts *desert* and *explosion_fire* having a too high ratio λ cannot be seen clearly because of the low precision values. Table 4 summarizes the selected threshold values for all 9 concepts. As it may be observed, for those concepts that their positive examples do not vary a lot, in respect to their model vectors, such as *desert* and *mountain*, a high threshold value is selected.

In the last part of the experiments, the proposed approach is evaluated on the testing sets derived from the TRECVID 2007 development data. The testing set of each concept contains 30% of all positive examples and is complemented using part from negative examples, i.e. from all keyframes that do not contain the specific

Table 3 Average precision on a test set with $\lambda = 1$, for several values of the ratio λ within the training set

Concept	Average precision				
	$\lambda = 1$	$\lambda = 2$	$\lambda = 3$	$\lambda = 4$	$\lambda = 5$
Desert	0.659	**0.699**	0.365	0.477	0.663
Road	0.594	0.609	0.595	0.606	**0.695**
Sky	0.679	0.723	0.688	0.719	**0.736**
Snow	0.914	0.905	0.929	0.917	**0.950**
Vegetation	0.717	0.773	0.764	0.752	**0.780**
Office	0.633	0.707	**0.738**	0.707	0.723
Outdoor	0.683	0.684	**0.697**	–	–
Explosion_fire	0.387	0.367	0.348	**0.647**	0.382
Mountain	0.687	0.611	0.545	0.625	**0.766**

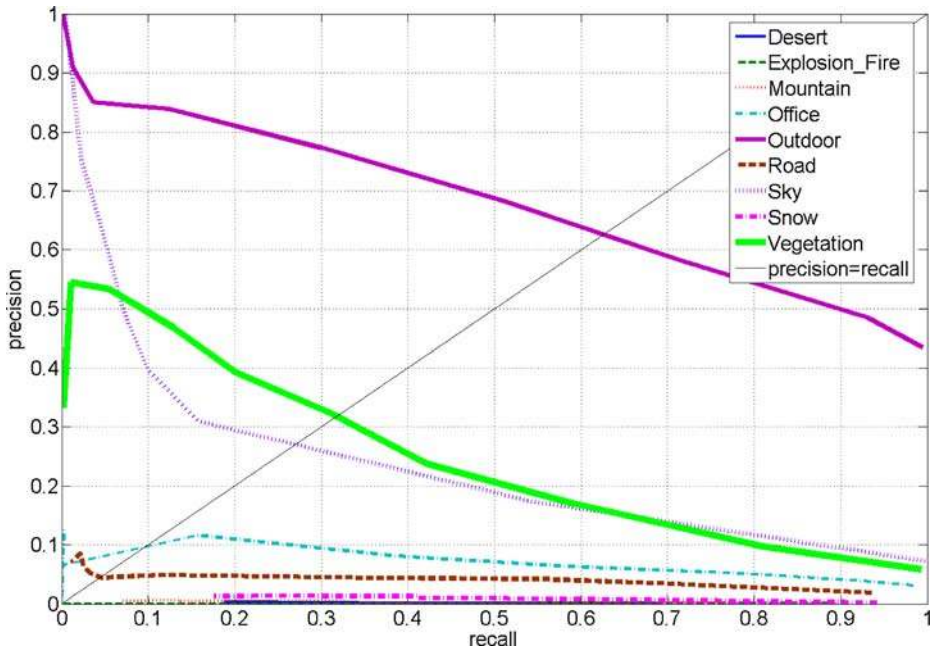

Fig. 8 Precision-recall for increasing threshold values

concept. The number of negative keyframes increases gradually, until it reaches certain values of λ. For each concept, the value of λ is increased until it reaches its maximum possible value. Each time the AP is calculated, with a window equal to all the testing set.

Figures 9 and 10 show how AP changes with respect to λ of the test set. The number of positive examples is kept fixed, while the number of negative increases. It may be observed that when the value of λ is relatively small, i.e. $\lambda = 4$, as in the case of typical data sets, the performances remain particularly high. When λ increases, then the performances fall as expected. Table 5 summarizes the concepts that are detected and the detection results.

In the TRECVID 2007 benchmark, 20 of the high-level concepts were selected for evaluation. *Office*, *desert*, *explosion_fire* and *mountain* were included in the evaluation process. Thus, in Table 6 we present one group for each concept, which

Table 4 Thresholds for the high-level concept detectors

Concept	Threshold
Desert	0.8
Road	0.5
Sky	0.3
Snow	0.6
Vegetation	0.4
Office	0.5
Outdoor	0.3
Explosion_fire	0.2
Mountain	0.8

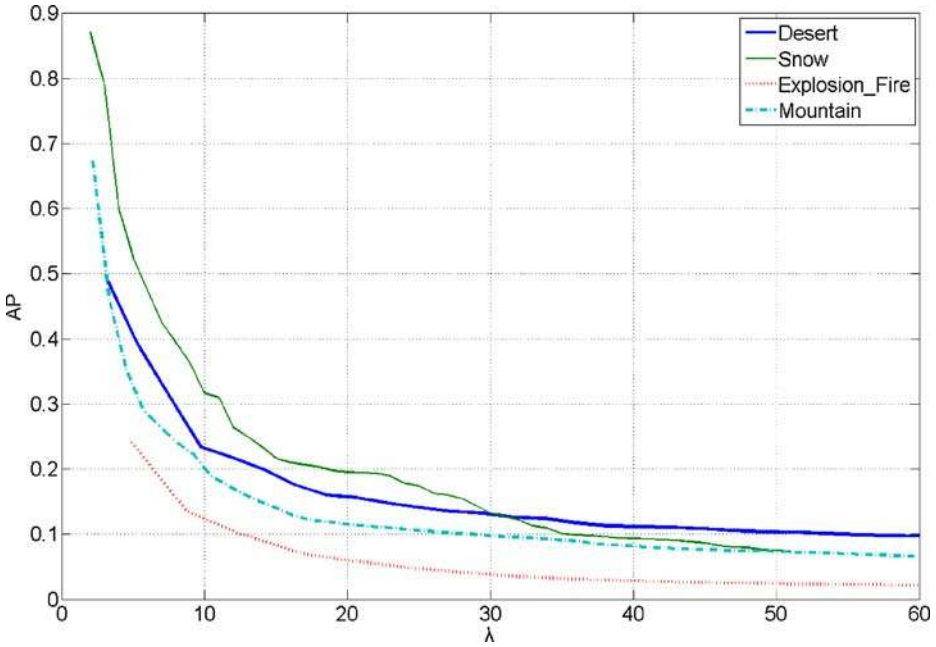

Fig. 9 AP vs. λ of the test set for *desert*, *snow*, *explosion_fire* and *mountain*

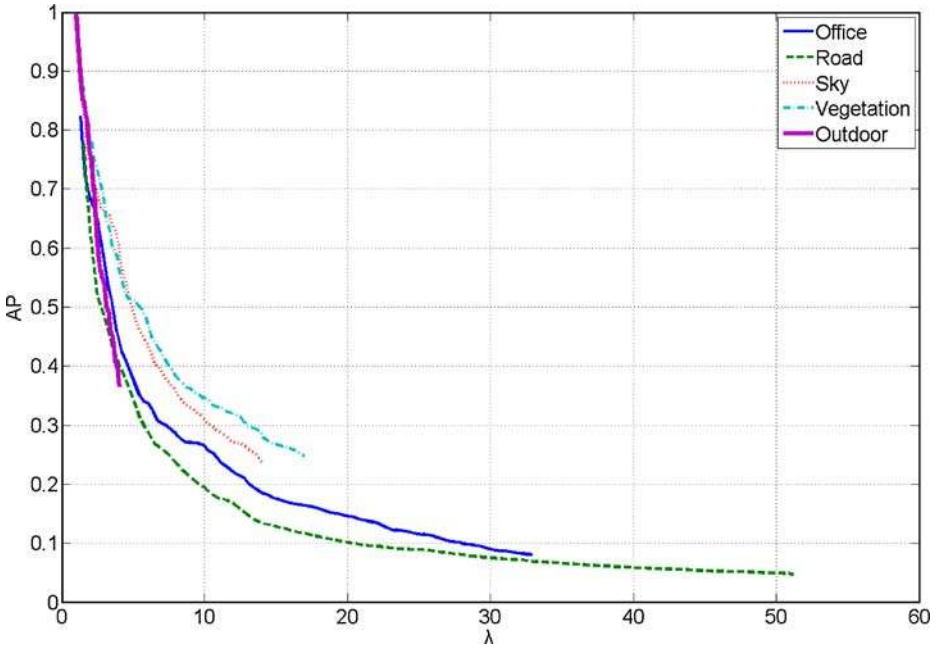

Fig. 10 AP vs. λ of the test set for *office*, *outdoor*, *road*, *sky* and *vegetation*

Table 5 Experiments for test sets with $\lambda = 4$ and with the maximum value of λ. P=precision, R=recall, AP=average precision

Concept	$\lambda = 4$			$\lambda = max$		
	P	R	AP	P	R	AP
Vegetation	0.643	0.312	0.460	0.322	0.313	0.232
Road	0.295	0.046	0.280	0.045	0.047	0.043
Explosion_fire	0.291	0.777	0.182	0.000	0.000	0.001
Sky	0.571	0.304	0.436	0.258	0.304	0.214
Snow	0.777	0.411	0.460	0.013	0.412	0.008
Office	0.446	0.157	0.318	0.117	0.157	0.072
Desert	0.333	0.312	0.287	0.003	0.313	0.064
Outdoor	0.425	0.514	0.361	0.425	0.514	0.361
Mountain	0.444	0.137	0.241	0.003	0.379	0.037

achieved one of the top rankings (between first and third) and the corresponding average precision, mean and median score within results of all groups. The measure calculated is the inferred average precision on a 50% random pool sample, while in our experiments the actual average precision has been calculated. It should also be noted that those results are on the TRECVID 2007 test data. The test data of our approach is a subset of the TRECVID 2007 development data because 70% of the positive examples and some negative examples (1 to 4 times the number of positives) have been used to train the detectors. Our test data has a higher value of ratio λ from the test data TRECVID 2007 thus leading to an uneven comparison since detection performance depends on λ like we have already mentioned. By observing results of Table 6 with the ones of Table 5 (AP of maximum λ) one can say that although our method achieves lower scores than the best ones, it performs better than the mean and median scores for 3 of the 4 concepts. For the concept *explosion_fire* the low AP is achieved with our approach probably because only 20 positive examples were used to train the detector.

Columbia university group employs a simple baseline method, composed of three types of features, individual SVMs trained independently over each feature space, and a simple late fusion of the SVMs and finally trains detectors for 374 high-level concepts [43]. In addition a new cross-domain SVM (CDSVM) algorithm for adapting previously learned support vectors from one domain to help classification in another domain has been developed. Tsinghua university group uses 26 types of various color, edge and texture features and compares Under-sampling SVM (USVM) and SVM with RankBoost, direct Boosting algorithms. For fusion, sequential forward floating feature selection (SFFS), simulated annealing fusion (SA) and Borda based rank fusion approaches are designed and compared respectively.

Table 6 Trecvid 2007 results on the evaluation data

Concept	Group	AP	Mean	Median
Office	CITY UHK	0.222	0.074	0.066
Desert	Tsinghua	0.155	0.022	0.009
Explosion_fire	COL	0.037	0.010	0.006
Mountain	OXVGG	0.120	0.037	0.032

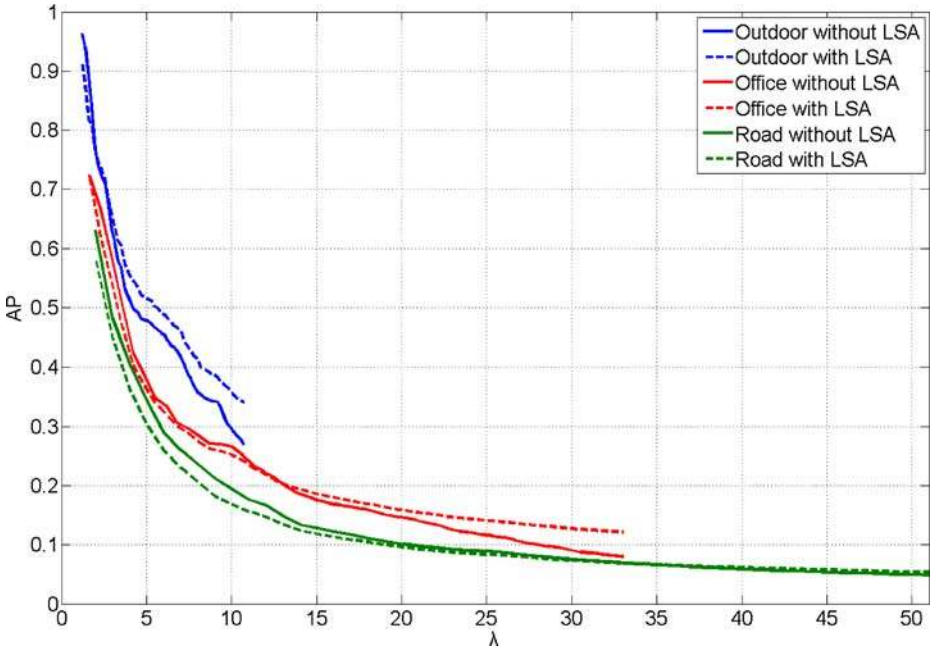

Fig. 11 AP vs λ for *outdoor*, *office* and *road*

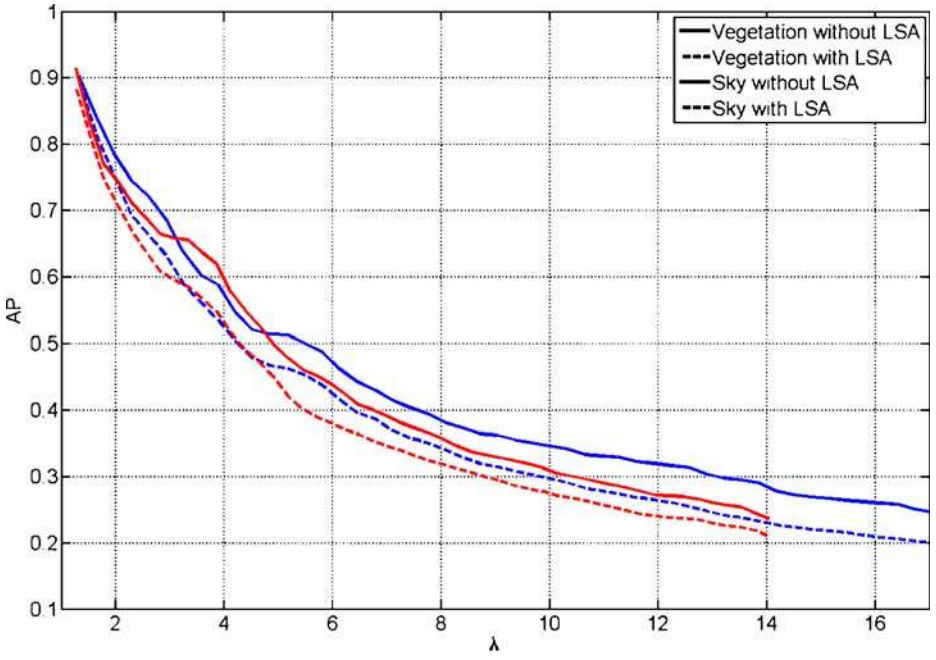

Fig. 12 AP vs λ for *vegetation* and *sky*

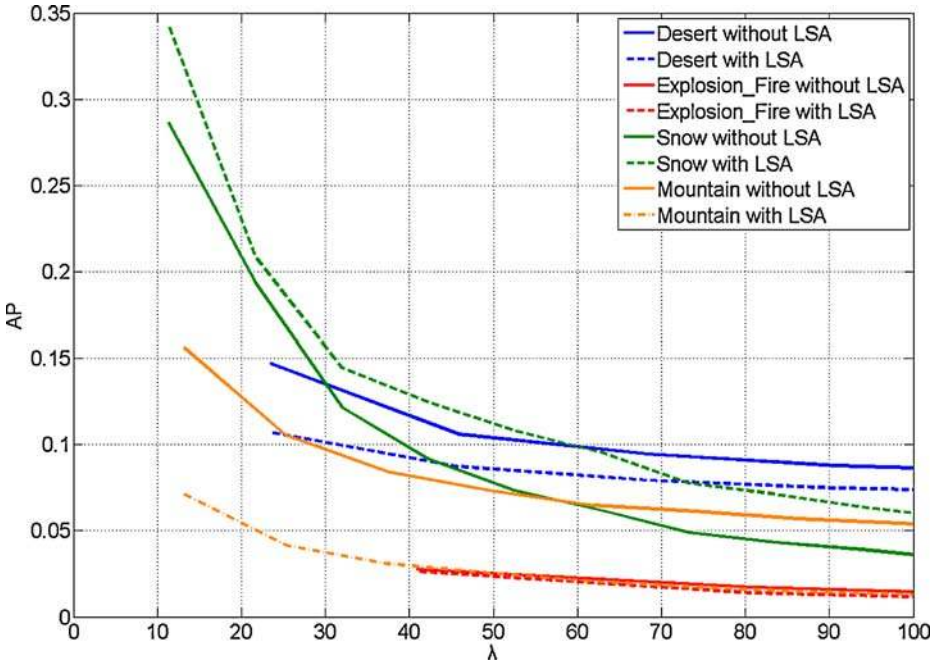

Fig. 13 AP vs λ for *desert*, *explosion_fire*, *snow* and *mountain*

9.2 Utilizing latent semantic analysis

A separate training and testing set has been generated for each concept. 70% of the positive examples was randomly selected for the training set of each concept and the remaining 30% for the testing set. Negative examples were selected randomly from the remaining keyframes.

After segmenting every keyframe of the training set to coarse regions a visual thesaurus of region types is constructed, by subtractive clustering. After extracting

Table 7 Experiments with the use of LSA for test sets with $\lambda = 4$ and with the maximum value of λ

Concept	$\lambda = 4$			$\lambda = max$		
	P	R	AP	P	R	AP
Vegetation	0.626	0.221	0.395	0.268	0.222	0.179
Road	0.400	0.050	0.210	0.036	0.051	0.044
Explosion_fire	0.200	0.111	0.148	0.001	0.111	0.000
Sky	0.559	0.271	0.372	0.288	0.207	0.184
Snow	0.818	0.264	0.529	0.023	0.265	0.012
Office	0.406	0.147	0.285	0.095	0.148	0.110
Desert	0.215	0.687	0.246	0.001	0.438	0.063
Outdoor	0.331	0.634	0.382	0.331	0.634	0.382
Mountain	0.110	0.035	0.072	0.003	0.172	0.001

a concept *sky*

b concept *vegetation*

Fig. 14 True positive examples (**a**, **b**)

model vectors from all images of the training set, LSA is applied. The number k of the largest singular values to keep is set to 70. Then an SVM-based detector is trained for each concept. Its input is the output of the LSA algorithm $\hat{m}_i$ and its output denotes the confidence that the specific concept exists.

Figures 11, 12 and 13 show the Average Precision (AP) [15] vs the ratio λ of negative to positive examples. The number of positive examples is kept fixed, while the number of negative increases. It may be observed that when λ has a relatively small value, i.e. $\lambda = 4$, as in the case of typical test sets, the performance of the

a concept *sky*

b concept *vegetation*

Fig. 15 False negative examples (**a**, **b**)

a concept *sky*

b concept *vegetation*

Fig. 16 False positive examples (**a**, **b**)

classifiers remains particularly high. When λ increases, then the performance falls. Moreover, we may observe that the use of LSA does not always improve the results. For certain concepts such as *outdoor*, *office* and *road*, LSA improves the results, while λ increases, as depicted in Fig. 11. This means that positive examples are detected in a lower and more correct rank. The common property of these concepts is that they cannot be described in a straightforward way, such as e.g. *vegetation* and *sky* (Fig. 12). That becomes obvious when examining the TRECVID data. Finally, for the concepts depicted in Fig. 13, where the available number of positive examples is particularly small, using LSA improves only the semantic concept *snow*. Table 7 summarizes the concepts that are detected and the detection results with the use of LSA.

9.3 Discussion on the detection results

In this section, some comments regarding the detection results are presented, focusing on examples of true and false detections of high-level concepts. Examples are

Table 8 Detection results while using more representative keyframes selected with our algorithm

Concept	P	R	AP
Vegetation	0.625	0.450	0.515
Road	0.300	0.062	0.324
Explosion_fire	0.220	0.780	0.180
Sky	0.525	0.522	0.585
Snow	0.635	0.405	0.452
Office	0.433	0.170	0.320
Desert	0.312	0.375	0.348
Outdoor	0.410	0.658	0.450
Mountain	0.422	0.135	0.230

Fig. 17 Frames within a shot

depicted for concepts *sky* and *vegetation* in Figs. 14, 15 and 16 for true positive, false negative and false positive examples, respectively.

In Fig. 14a, 4 keyframes containing the concept *sky* and in Fig. 14b 4 keyframes containing the concept *vegetation* are depicted. These examples correspond to correct detection of the aforementioned concepts to the presented keyframes. One can easily observe the characteristic regions of *sky* and *vegetation*.

Figure 15 presents false negative keyframes, which is keyframes depicting the corresponding concepts but detected from the trained classifiers as negatives. First and third images of Fig. 15a are examples of artificial regions of *sky* which have too different visual features from the normal ones. A classifier trained to detect blue, grey (cloudy) or even with an orange tone *sky* (sunset) faces difficulties in detecting artificial purple (first) or yellow (third) *sky*. Moreover in the first and the second image the segmentation actually merges region of *sky* with tree branches so visual features are degraded. In the fourth image there is a totally white region of *sky* due to the lightning conditions. As for the concept *vegetation*, in first, second and last image of Fig. 15b, the existing small regions of *vegetation* are also merged with other regions as a result of the under-segmentation of the image. Finally, third image contains

a one keyframe **b** more keyframes extracted

Fig. 18 Single keyframe selected manually by TRECVID to represent the shot (**a**) and more keyframes extracted with our algorithm (**b**)

a flower, but due to poorly trained annotators has been mistakenly annotated as *vegetation*.

Finally, there also exist some examples of images which do not depict the selected concepts, however they were detected as positive. These false positive examples are depicted in Fig. 16. All images falsely detected are positive for the concept *sky* (Fig. 16a) contain light blue regions which are similar to a typical region of *sky* in both color and texture. Moreover, for the rest of the images, two annotation errors are present where the concept *vegetation* is depicted but were not annotated as positive. These are the first and third image in Fig. 16b. Second image of this figure has actually a dark green tone and texture thus is similar with the ones that a typical *vegetation* region would have.

Fig. 19 Frames within the video

9.4 High-level concept detection using keyframe extraction on shots

The approach described in Section 8 of high-level concept detection while from each shot a relatively small number of representative keyframes is extracted, as described in Section 7, is evaluated in 6 of the TRECVID videos and compared with detection results using a single keyframe. Results are depicted in Table 8. Results show exactly what was expected, i.e. recall values are higher and precision values are more or less of similar values.

To better understand why the representation of a shot with more than one keyframes enhances detection in video shots, we present herein the example where from a shot, a single keyframe extracted, and a larger number of more extracted representative keyframes. A large number of frames within the shot are extracted to represent its visual and semantic content and are depicted in Fig. 17. The duration of this is almost equal to 5 seconds. The concepts it contains among those that have been selected for the high-level concept detection are obviously *outdoor*, *vegetation*

Fig. 20 Keyframes extracted

and *sky*. The single keyframe chosen to represent the shot is the one depicted in Fig. 18a. Our keyframe extraction algorithm extracted the 4 keyframes depicted in Fig. 18b. It is obvious that the 1st keyframe in Fig. 18b is better for detecting *sky* and the 2nd is better for detecting *vegetation* than the single keyframe randomly selected. Moreover the 4 keyframes will surely increase the possibility of detecting correctly the *outdoor* concept too.

9.5 Keyframe extraction for summarization

In this final section we present indicative frames (Figs. 19 and 20) containing some preliminary results of our proposed method for summarization using the aforementioned keyframe extraction algorithm. For the sake of a more clear presentation, we used a part of a TRECVID video from the CNN news, since in this case, the keyframes are more heterogeneous and the keyframe selection is more difficult.

The video contains the anchorman in the studio, some commercials, some indoor and outdoor interviews, charts, etc. Its duration is approximately 620 seconds. In Fig. 19 we present in brief the visual content of it. Then in Fig. 20 we present the extracted keyframes. We should emphasize again that those keyframes are all the frames closer to the centroid of the cluster created with subtractive clustering on the model vectors.

Our approach for video summarization exploits the novel model vector representation by utilizing a region thesaurus that is formed within the given video. This way, a number of representative keyframes is extracted. In Section 9.4 the keyframe extraction algorithm was evaluated in combination with high-level concept detection. Therein, significant improvement on the performance has been observed. In Fig. 20 we present the results of the algorithm when used to summarize a whole video. Keyframes extracted depict the most essential high-level concepts included in the video. The novelty of this approach relies on the fact that the visual thesaurus is constructed on each video. Thus the thesaurus is adapted on the specific content and the keyframes are dynamically represented by the region types contained in the particular video. However, we should note that it remains to further work to evaluate each video individually using appropriate ground truth or involving users in the evaluation procedure.

10 Conclusions and future work

Our research effort indicates clearly that high-level concepts can be efficiently detected when an image is represented by a mid-level model vector with the aid of a visual thesaurus. We presented a methodology on working with large data sets, such as the TRECVID collections. Extensive experiments have been presented and the effect of the ratio λ of negative to positive examples on training and testing data has been examined and the applied generic approach tackled successfully most of the selected high-level concepts. LSA was also exploited by transforming the image descriptions into the concept space. Moreover a keyframe extraction algorithm based also on image representation by a model vector was combined with the high-level

concept detection scheme to provide a more meaningful representation of the visual content and finally enhanced the detection performance.

To conclude, the main advantage of the presented high-level concept detection approach is that it provides a generic method of detecting material-like concepts or scenes, instead of concept oriented or heuristic methods. Moreover, since it is based on the co-occurrences of the region types, it is invariant to the size of the area that contains the concept in question. However, its main limitation is that it depends a lot in the segmentation algorithm that is used at the initial stage. If the area of a concept is significantly small and similar to its neighbor it may easily be merged, thus a misleading model vector may be formed. Moreover, it is limited to this certain type of concepts, that is materials and scenes while it cannot be extended to detect objects. The keyframe extraction algorithm provides a novel approach based on the local (within a shot) mid-level features and allows the concept detection approach to efficiently detect the concepts in question within a shot without having to analyze every video frame separately.

Future work aims to compare the presented algorithm with other approaches, within the same data sets derived from the TRECVID collections. Moreover, the contextual relations among image regions may also be exploited in order to assist the results of the high-level detection. Finally, the summarization algorithm will be separately evaluated for a large number of videos possibly with the involvement of users.

Acknowledgements This work was partially supported by the European Commission under contracts FP7-215453 WeKnowIt, FP6-027026 K-Space and FP6-027685 MESH.

References

1. Avrithis Y, Doulamis A, Doulamis N, Kollias S (1999) A stochastic framework for optimal key frame extraction from mpeg video databases. Comput Vis Image Underst 5(1):3–24
2. Ayache S, Quenot G (2007) TRECVID 2007 collaborative annotation using active learning. In: TRECVID 2007 workshop, Gaithersburg, 5–6 November 2007
3. Boujemaa N, Fleuret F, Gouet V, Sahbi H (2004) Visual content extraction for automatic semantic annotation of video news. In: IS&T/SPIE conference on storage and retrieval methods and applications for multimedia, part of electronic imaging symposium, San Jose, January 2004
4. Chang CC, Lin CJ (2001) LIBSVM: a library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm
5. Chang SF, Sikora T, Puri A (2001) Overview of the MPEG-7 standard. IEEE Trans Circuits Systems Video Technol 11(6):688–695
6. Chapelle O, Haffner P, Vapnik V (1999) Support vector machines for histogram-based image classification. IEEE Trans Neural Netw 10(5):1055–1064
7. Chiu S (1997) Extracting fuzzy rules from data for function approximation and pattern classification. In: Dubois D, Prade H, Yager R (eds) Fuzzy information engineering: a guided tour of applications. Wiley, New York
8. Cooper M, Foote J (2005) Discriminative techniques for keyframe selection. In: Proceedings of the IEEE international conference on multimedia & expo (ICME), Amsterdam, 6–9 July 2005
9. Dance C, Willamowski J, Fan L, Bray C, Csurka G (2004) Visual categorization with bags of keypoints. In: ECCV—international workshop on statistical learning in computer vision
10. Deerwester S, Dumais S, Furnas GW, Landauer TK, Harshman R (1990) Indexing by latent semantic analysis. J Soc Inf Sci 41(6):391–407

11. Everingham M, Van Gool L, Williams CKI, Winn J, Zisserman A (2007) The PASCAL visual object classes challenge 2007 (VOC2007) results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html
12. Gokalp D, Aksoy S (2007) Scene classification using bag-of-regions representations. In: IEEE conference on computer vision and pattern recognition (CVPR), Minneapolis, 18–23 June 2007
13. Haykin S (1998) Neural networks: a comprehensive foundation. Prentice Hall, Englewood Cliffs
14. IBM (2005) MARVEL multimedia analysis and retrieval system. IBM Research White Paper
15. Kishida K (2005) Property of average precision and its generalization: an examination of evaluation indicator for information retrieval. NII Technical Reports, NII-2005-014E
16. Klir GJ, Yuan B (1995) Fuzzy sets and fuzzy logic—theory and applications. Prentice Hall, Englewood Cliffs
17. Laaksonen J, Koskela M, Oja E (2002) Picsom, self-organizing image retrieval with MPEG-7 content descriptors. IEEE Trans Neural Netw 13:841–853
18. Lazebnik S, Schmid C, Ponce J (2006) A discriminative framework for texture and object recognition using local image features. In: Towards category-level object recognition. Springer, New York, pp 423–442
19. Lowe D (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
20. Ma YF, Lu L, Zhang HJ, Li M (2002) A user attention model for video summarization. In: MULTIMEDIA '02: Proceedings of the tenth ACM international conference on multimedia. ACM, New York, pp 533–542
21. Manjunath B, Ohm J, Vasudevan V, Yamada A (2001) Color and texture descriptors. IEEE Trans Circuits Syst Video Technol 11(6):703–715
22. Mérialdo B, Huet B, Yahiaoui I, Souvannavong F (2002) Automatic video summarization. In: International thyrrenian workshop on digital communications, advanced methods for multimedia signal processing, Palazzo dei Congressi, Capri, 8–11 September 2002
23. Mitchell M (1998) An introduction to genetic algorithms. MIT, Cambridge
24. Molina J, Spyrou E, Sofou N, Martinez JM (2007) On the selection of MPEG-7 visual descriptors and their level of detail for nature disaster video sequences classification. In: 2nd international conference on semantics and digital media technologies (SAMT), Genova, 5–7 December 2007
25. Morris OJ, Lee MJ, Constantinides AG (1986) Graph theory for image analysis: an approach based on the shortest spanning tree. IEE Proc 133:146–152
26. Naphade MR, Kennedy L, Kender JR, Chang SF, Smith JR, Over P, Hauptmann A (2005) A light scale concept ontology for multimedia understanding for TRECVID 2005. IBM Research Technical Report
27. Natsev A, Naphade M, Smith J (2003) Lexicon design for semantic indexing in media databases. In: International conference on communication technologies and programming, Varna, 23–26 June 2003
28. Opelt A, Pinz A, Zisserman A (2006) Incremental learning of object detectors using a visual shape alphabet. In: IEEE computer society conference on computer vision and pattern recognition, New York, 17–22 June 2006
29. Russell BC, Torralba A, Murphy KP, Freeman WT (2008) Labelme: a database and web-based tool for image annotation. Int J Comput Vis 77:157–173
30. Saux BL, Amato G (2004) Image classifiers for scene analysis. In: International conference on computer vision and graphics, Warsaw, 22–24 September 2004
31. Smeaton AF, Over P, Kraaij W (2006) Evaluation campaigns and TRECVid. In: MIR '06: proceedings of the 8th ACM international workshop on multimedia information retrieval. ACM, New York
32. Smeulders AWM, Worring M, Santini S, Gupta A, Jain R (2000) Content-based image retrieval at the end of the early years. IEEE Trans Pattern Anal Mach Intell 22(12):1349–1380
33. Snoek CGM, Worring M (2003) Time interval based modelling and classification of events in soccer video. In: Proceedings of the 9th annual conference of the advanced school for computing and imaging (ASCI), Heijen, June 2003
34. Souvannavong F, Mérialdo B, Huet B (2005) Region-based video content indexing and retrieval. In: CBMI 2005, fourth international workshop on content-based multimedia indexing, Riga, 21–23 June 2005
35. Spyrou E, Avrithis Y (2007) A region thesaurus approach for high-level concept detection in the natural disaster domain. In: 2nd international conference on semantics and digital media technologies (SAMT), Genova, December 2007

36. Spyrou E, Avrithis Y (2007) Keyframe extraction using local visual semantics in the form of a region thesaurus. In: 2nd international workshop on semantic media adaptation and personalization (SMAP), London, 17–18 December 2007
37. Spyrou E, LeBorgne H, Mailis T, Cooke E, Avrithis Y, O'Connor N (2005) Fusing MPEG-7 visual descriptors for image classification. In: International conference on artificial neural networks (ICANN), Warsaw, 11–15 September 2005
38. Spyrou E, Tolias G, Mylonas P, Avrithis Y (2008) A semantic multimedia analysis approach utilizing a region thesaurus and LSA. In: International workshop on image analysis for multimedia interactive services (WIAMIS), Klagenfurt, 7–9 May 2008
39. Sundaram H, Chang SF (2003) Video analysis and summarization at structural and semantic levels, multimedia information retrieval and management: technological fundamentals and applications. In: Feng D, Siu WC, Zhang H (Eds) Springer, New York
40. Vapnik V (1995) The nature of statistical learning theory. Springer, New York
41. Voisine N, Dasiopoulou S, Mezaris V, Spyrou E, Athanasiadis T, Kompatsiaris I, Avrithis Y, Strintzis MG (2005) Knowledge-assisted video analysis using a genetic algorithm. In: 6th international workshop on image analysis for multimedia interactive services (WIAMIS 2005), Montreux, 13–15 April 2005
42. Yamada A, Pickering M, Jeannin S, Cieplinski L, Ohm J, Kim M (2001) MPEG-7 Visual part of eXperimentation model version 9.0
43. Yanagawa A, Chang SF, Kennedy L, Hsu W (2007) Columbia universitys baseline detectors for 374 LSCOM semantic visual concepts. Columbia University ADVENT Technical Report
44. Yuan J, Guo Z et al (2007) THU and ICRC at TRECVID 2007. In: 5th TRECVID workshop, Gaithersburg, November 2007
45. Zhang H, Wu J, Zhong D, Smoliar S (1997) An integrated system for content-based retrieval and browsing. Pattern Recogn 30:643–658
46. Zhuang Y, Rui Y, Huang T, Mehrotra S (1998) Adaptive keyframe extraction using unsupervised clustering. In: Proc of international conference oh image processing (ICIP), Chicago, 4–7 October 1998

Evaggelos Spyrou was born in Athens, Greece in 1979. He received the Diploma in Electrical and Computer Engineering from the National Technical University of Athens (NTUA) in 2003. He is currently a PhD student at the Image, Video and Multimedia Laboratory of NTUA. His research interests include semantic analysis of multimedia, high-level concept detection, visual context and neural networks. He has published 2 book chapters and 20 papers in international conferences and workshops.

Giorgos Tolias was born in Athens, Greece, in 1984. He received the Diploma in electrical and computer engineering from the Department of Electrical Engineering, National Technical University of Athens (NTUA), Athens, Greece, in 2007. He is currently working as a research associate at the Semantic Multimedia Analysis Group of the Image Video and Multimedia Laboratory, NTUA. His research interests include high-level feature extraction, object recognition, as well as content based retrieval and image matching.

Phivos Mylonas obtained his Diploma in Electrical and Computer Engineering from the National Technical University of Athens (NTUA) in 2001, his Master of Science in Advanced Information Systems from the National & Kapodestrian University of Athens (UoA) in 2003 and his Ph.D. degree at the former University (NTUA) in 2008. He is currently a Researcher by the Image, Video and Multimedia Laboratory of NTUA. His research interests lie in the areas of content-based information retrieval, visual context representation and analysis, knowledge-assisted multimedia analysis, issues related to multimedia personalization, user adaptation, user modelling and profiling. He has published 19 articles in international journals and book chapters, he is the author of 34 papers in international conferences and workshops, he has published 3 books and is a guest editor of 2 international journals, he is a reviewer for 7 international journals and has been involved in the organization of 13 international conferences and workshops. He is an IEEE and ACM member.

Yannis Avrithis received the Diploma degree in Electrical and Computer Engineering from the National Technical University of Athens in 1993, the M.Sc. in Communications and Signal Processing (with Distinction) from the Department of Electrical and Electronic Engineering of the Imperial College of Science, Technology and Medicine, University of London, UK, in 1994, and the PhD from NTUA on digital image processing in 2001. He is currently a senior researcher at the Image, Video and Multimedia Systems Laboratory (IVML) of NTUA, coordinating R&D activities in Greek and EU projects, and lecturing in NTUA. His research interests include image/video segmentation and interpretation, knowledge-assisted multimedia analysis, annotation, content-based and semantic indexing and retrieval, video summarization and personalization. He has published 1 book, 13 articles in international journals, 15 book chapters and 62 in conferences and workshops. He is an IEEE member, and a member of ACM and EURASIP.