Scene Detection using Visual and Audio Attention

Angelo Chianese
University of Naples
via Claudio, 21
80125, Naples, Italy
angchian@unina.it

Vincenzo Moscato
University of Naples
via Claudio, 21
80125, Naples, Italy
vmoscato@unina.it

Antonio Penta
University of Naples
via Claudio, 21
80125, Naples, Italy
a.penta@unina.it

Antonio Picariello
University of Naples
via Claudio, 21
80125, Naples, Italy
picus@unina.it

ABSTRACT

Shot and scene segmentation are basic steps for a variety of applications in video analysis and processing. In this paper, we propose a new method for automatic scene detection which takes visual patterns of movies and audio features into account. In particular, we also show that the use of audio analysis, to detect transitions in an audio stream, is suitable in order to capture the scene boundaries as well.

1. INTRODUCTION

While the amount of video data is rapidly increasing, multimedia applications are still very limited in content management capabilities: there is a growing demand for new techniques that can enable efficient processing, modeling and management of video contents, but it is generally recognized that no standard description still exists.

The major bottleneck that limits a wider use of digital video, is the ability of quickly finding desired information from a huge database. A reliable way to enable fast access to video clips, is to properly index video sequences using suitable descriptors.

A typical indexing and retrieval scenario of video content is shown in Figure 1. First, input videos and images are segmented into temporal consistent units. Visual and audio features are then extracted from these segments to build indices and summaries. Eventually, videos or images are browsed and retrieved, based on these features and structures.

To these purposes, shots and scenes detection are considered a base for a lot of semantic video abstraction applications. A *shot* is usually conceived in the literature as a series of interrelated consecutive frames taken contiguously by a single camera and representing a continuous action in time and space. In other terms, a shot is a video subsequence generated by the camera from the time it "starts" recording images, to the time it "stops" recording. Video shots must be opportunely merged together based on content, in order to partition a video into useful semantic entities: a *scene* is usually composed of a small number of interrelated shots that are unified by location or dramatic incident.

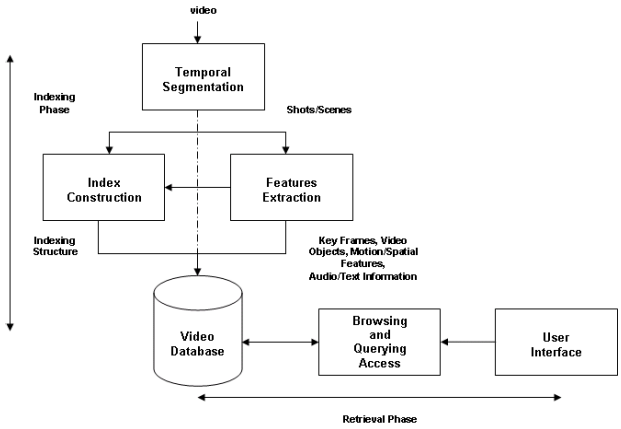

Figure 1: Block diagram of a video database management system for content-based video indexing and retrieval

The main goal of this process is to automatically determine shot clusters which a human would judge as "scenes": this is an important problem for a variety of reasons, especially due to the fact that video segmentation is the first step towards semantic understanding of the entire video, thus allowing for non-linear navigation of video data. Fortunately, although scene segmentation depends on subjective judgements, some basic units may be clustered by all viewers, such as contiguous shots that are usually combined by *continuing music* or, more generally, by *audio background*.

In the literature, several automatic techniques for video scene detection have been proposed. The majority of such methods uses jointly audio and visual information for accomplishing the above task. In [17], the main audio and visual features that can effectively characterize scene content

AMDIT 2008, February 14, Quebec, Canada

DOI 10.4108/ICST.AMBISYS2008.2828

and some algorithms for video segmentation and classification are reported. In [4] some visual useful metrics for scene change detection based on scene lighting and intensity distribution are presented; in opposite, [8] focuses the attention on the associated audio information for video scene analysis. In [18] a framework to group shots based on the analysis of video content continuity (in terms of visual, position, camera, motion and audio features) is performed. In [15] video scenes are detected on the base of chromacity, lighting conditions and ambient sound video properties and in [1] a Markov model approach for scene detection based on audio and visual video analysis is proposed. Eventually, in [11] a scheme for identifying scenes on the base of video genre has been developed, while in [6] a system capable of discovering, on the base of visual and audio features, the video blocks belonging to the same semantic class is discussed.

In this paper, we present a system for automatic scene detection. In particular, we have developed a joint audio-visual framework for video scene segmentation, starting from the consideration that a scene is a sequence of audio-visual frames that possesses consistent and contiguous audio and visual properties.

At the heart of our ability to detect changes from one view of a scene to the next is the mechanism of "attention", where the attention captures the cognitive functions that are responsible for filtering out unwanted information and bringing to consciousness what is relevant for the observer [10]. In a recent paper [2] some of the authors have proposed a novel technique based on attentive paradigm, *Animate Vision*, to reliably segment a video sequence in shots, detecting both *abrupt* and *gradual* transitions.

Here, we use both visual and audio attention to detect scene changes. Visual attention is related to how we view scenes in the real world and each scene change corresponds to a human attention shifting. Similarly the audio attention is connected to how some features inside one or more consecutive video shots can determine attention swings during information acquisition.

The paper is organized as follows. Section 2 briefly reports the systems architecture. In section 3, we describe the animate video segmentation process. In section 4 the chosen audio features for scene analysis are reported. Section 5 describes the scene detection system. In section 6 the experimental protocol and related results are provided. Eventually some concluding remarks are given in section 7.

2. SYSTEM OVERVIEW

Figure 2 describes at a glance the main characteristics of the proposed segmentation system. We basically distinguish three components: (i) a *video segmentation module* for the detection of shots on the base of animate vision descriptors; (ii) an *audio features extractor* module determining, for each shot, a set of characteristics related to the audio frames; and (iii) a *shot aggregation* module, determining video scenes on the base of audio features of consecutive shots.

For audio data, we focus on audio background. Usually, audio background purpose is that of emphasizing the emotion or of furnishing certain impressions to the audience. Other features, such as silence between spoken dialogues, may be used to enforce shot boundaries. In general, it is a common experience for movies that background audio, especially music and silence, are particularly suitable to merge video shots into contiguous scene boundaries.

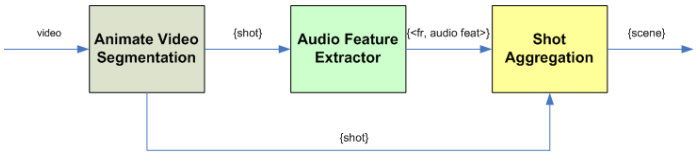

Figure 2: System Architecture Overview

In the next section we will describe the proposed methods and the related algorithms.

3. THE *ANIMATE* VIDEO SHOT SEGMENTATION PROCESS

As already discussed, the first step in an automatic video indexing process is the so called *video shot segmentation*, able to detect video shots. In a given video, different kinds of transitions may occur and the basic distinction is between *abrupt* and *gradual* ones.

Abrupt transitions are very easy to detect because the two successive frames involved in the transition are totally uncorrelated. On the contrary, gradual transitions are harder to detect from a data-analysis point of view because the difference between consecutive frames is substantially reduced and apposite algorithms have been proposed.

One of the main issue in segmenting a video sequence into shots is the ability to distinguish between scene breaks and normal changes into the scene. Moreover, camera movements such as panning, tilting and zooming, present similar features to transition effects such as dissolves. A reliable video segmentation algorithm must be able to recognize dissolve effects without misinterpreting camera movements as gradual transitions.

3.1 The *Animate* Image Similarity

As previously described, a fundamental component of a video segmentation system is an *image (frame) similarity detector*. In our approach we have used an image similarity algorithm based on the *Animate Vision* theory (see [2, 3] for more details) as briefly discussed in the following.

Visual attention is likely to be captured by *salient points* of an image. Each eye fixation attracted by such points, defines a *focus of attention* (`FOA`) on the foveated region of the scene, and the FOA sequence is denoted a saccadic *scanpath* [10]. According to scanpath theory, patterns that are visually similar, give rise to similar scanpaths when inspected by the same observer under the same viewing conditions. In other terms a scanpath respects the properties of distinctiveness and invariance, that are requested to a salient points based technique [14].

In general, the generation of a scanpath under free viewing conditions, can be accomplished in three steps: (i) selection of interesting regions; (ii) features extraction from the detected regions; (iii) search of the next interesting region.

To this aim, a *pre-attentive* image representation, undergoes specialized processing through a *Where System* devoted to localize a sequence of regions of interest, and a *What*

System tailored for analyzing them. Attentive mechanisms provide tight integration of these two information pathways, since in the What pathway, feature extraction is performed, while being subjected to the action of the Where pathway and the related attention shifting mechanism, so that uninteresting responses are suppressed. In this way, the Where pathway allows to collect saliency points simulating human attentive inspection of an image.

In our framework, the Where pathway is implemented according to the image pyramidal decomposition proposed by Itti [7]. It linearly computes and combines three preattentive contrast maps (color, brightness, orientation) into a *master or saliency map*, which is then used to direct attention to the spatial location with the highest saliency through a *winner take-all* (*WTA*) network (*attention shifting* stage). The region surrounding such location represents the current FOA, say F_s. By traversing spatial locations of decreasing saliency, it is then possible to observe a motor trace (scanpath) representing the stream of foveation points for an image I_i, namely:

$$\varsigma = \langle F_s^i(p_s;\tau_s)\rangle_{s=1,2,\ldots,N_f} \tag{1}$$

being $p_s = (x_s, y_s)$ the center of *FOA* s, N_f the number of explored *FOA*s (such parameter is set before the scanpath generation), and being the delay parameter τ_s the observation time spent on the *FOA* before a saccade shifts to F_{s+1}, provided by the *WTA* net.

An inhibition mechanism avoids that a winner point is thoroughly reconsidered in the next steps. Note that from the Where pathway two dynamical features are derived: the spatial position p_s of each *FOA* and the the fixation time τ_s.

In the What pathway, information is extracted from each *FOA*, related to color, texture and shape. In particular, for each *FOA* F_s^i, the What pathway extracts two specific features: the color histogram $h_b(F_s^i)$ in the *HSV* representation space and the edge covariance signature $\Xi_{F_s^i}$ of the image wavelet transform considering only a first level decomposition ($|\Xi| = 18$).

Eventually, for each considered image I_i the "flow" of such features, namely the *Information Path* IP^i is generated:

$$IP^i = IP_s^i = \{(F_s^i(p_s;\tau_s), h_b(F_s^i), \Xi_{F_s^i})\} \tag{2}$$

where $s = 1,\ldots,N_f$; an IP is thus a map, a visuomotor trace, of the image in the WW space.

For defining the similarity function A, we rely upon our original assumption, the IP generation process performed on a pair of similar images under the same viewing conditions will generate similar IPs, a property that we denote *attention consistency*. In Fig. 3 two similar images with respective IPs are shown.

Hence, the image-matching problem can be reduced to an IP matching; in fact, experiments performed by Walker and Smith [16], provide evidence that when observers are asked to make a direct comparison between two simultaneously presented pictures, a repeated scanning, in the shape of a *FOA* by *FOA* comparison, occurs [16]. Thus, in our system, two images are similar if *homologous FOA*s have similar color, texture and shape features, are in the same spatial regions of the image, and are detected with similar times. The procedure, is a sort of inexact matching, which we have denoted *Animate Matching*.

Given a fixation point $F_r^t(p_r;\tau_r)$ in the test image I_t, the

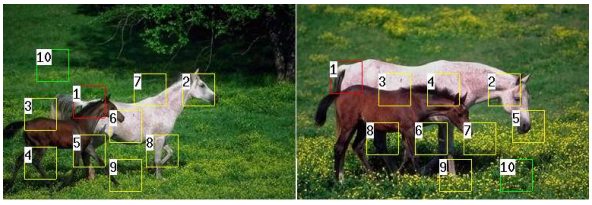

Figure 3: Similar images with similar *IP*s

procedure selects the homologous point $F_s^q(p_s;\tau_s)$ in the query image I_q among those belonging to a local temporal window, that is $\tau_s \in [s-H, s+H]$. The choice is performed by computing a local similarity $A^{r,s}$ for the pair F_r^t and F_s^q:

$$A^{r,s} = \alpha A_{spatial}^{r,s} + \beta A_{temporal}^{r,s} + \gamma A_{visual}^{r,s} \tag{3}$$

where $\alpha, \beta, \gamma \in [0,1]$, and by choosing the *FOA* s as $s = \arg\max\{A^{r,s}\}$. Such "best fit" is retained and eventually used to compute the consistency $A(IP^t, IP^q)$ as the average consistency of the first N_f' consistencies:

$$A = \frac{1}{N_f'} \sum_{f=1}^{N_f'} A_f^{r,s}, \tag{4}$$

where $N_f' <= N_f$, is the subset of image FOAs used for performing the matching procedure.

Right-hand terms of Eq. 3, namely $A_{spatial}^{r,s}$, $A_{temporal}^{r,s}$, $A_{visual}^{r,s}$, account for local measurements of spatial temporal and visual consistency, respectively. The former two are easily computed as $1 - d_{r,s}$ where $d_{r,s}$, generically represents the ℓ^1 distances either between (p_r, p_s) or (τ_r, τ_s) pairs, respectively.

Visual content consistency is given from the weighted mean $A_{visual}^{r,s} = \mu A_{col}^{r,s} + (1-\mu) A_{tex}^{r,s}$, where, similarly, color and texture consistencies $A_{col}^{r,s}, A_{tex}^{r,s}$ are obtained as 1 minus the ℓ^1 distance between color histograms and between texture covariances.

3.2 Video Segmentation using Attention Similarity

A video segmentation process can be modeled as follows.

Assume as input to a segmentation system a video sequence, that is a finite sequence of time parameterized images:

$$v = \langle f(t_0), f(t_1), \ldots f(t_N)\rangle \tag{5}$$

where each image $f(t_i)$ is called a *frame*. Each frame is a color image, namely a mapping from the discrete image support $\Omega \subseteq Z^2$ to an m-dimensional range:

$$f : \Omega \to Q \subseteq Z^m \tag{6}$$

In other terms, it is a set of single-valued images, or channels, sharing the same domain, i.e., $f(x,y) = (f_i(x,y))^T$, where the index $i = 1,..,m$, defines the i-th color channel and (x,y) denotes a point in the Ω lattice. $Q = \{q_1,...,q_N\}$ is the set of colors used in the image.

Each frame displays a view, a snapshot, of a certain visual configuration representing an original world scene.

A time segmentation of a video v defined on the time interval $[t_0, t_N]$ is a partition of the video sequence into N_b subsequences or blocks. One such partition can be obtained in two steps.

First, a mapping:

$$\mathcal{T} : Z^m \rightarrow F \tag{7}$$

of the frame $f(t_n) \in Z^m$ to a representation $\mathcal{T}(f(t_n)) \in F$, F being a suitable feature space, is performed.

Then, given two consecutive frames $f(t_n)$ and $f(t_{n+l})$, where $l \geq 1$ is the skip or inter-frame distance, a discriminant function:

$$\mathcal{D} : F \times F \rightarrow R^+ \tag{8}$$

is defined to quantify the visual content variation between $\mathcal{T}(f(t_n))$ and $\mathcal{T}(f(t_{n+l}))$, such that a boundary occurs at frame $f(t_n)$ if:

$$\mathcal{D}(\mathcal{T}(f(t_n)), \mathcal{T}(f(t_{n+l}))) > T \tag{9}$$

where T is a suitable threshold.

Thus, in principle, three steps must be undertaken to solve the shot detection problem: choose an appropriate mapping $\mathcal{T}$; define a robust discriminant function $\mathcal{D}$; devise a (universal) threshold T.

In our approach we have used as mapping $\mathcal{T}$ and discriminant function $\mathcal{D}$, those based on the discussed *Animate Image Similarity function*, while the threshold T has been set dynamically modeling the shot detection problem by means of a Bayesian statistical approach, as reported in [2].

In particular a a cut is detected in according to the equation:

$$p(A(t)|B) \cdot p(B) > p(A(t)|\bar{B})p(\bar{B}) \tag{10}$$

where $p(B)$ and $p(\bar{B})$ are the prior probabilities that model the Poisson process of shot boundary arrivals, and $p(A(t)|B)$ and $p(A(t)|\bar{B})$ are the probability of having or less a shot boundary on the base of the animate function $A(t)$ behavior.

In other terms our work proposes a novel approach in which video segmentation is drive by human attention. Shout boundaries corresponds to an attention shifting (i.e. decreasing of animate similarity) and are detected exploiting a prior knowledge on the context related to the video gradual and abrupt transitions.

4. AUDIO FEATURE EXTRACTION

4.1 Theoretical background

There are many features that can be used to characterize audio signals, usually separated in two categories, *time* and *frequency domain* [9].

In this work, we have pre-processed the audio signal by means of two classic techniques: *frame blocking* and *windowing*. The frame-blocking technique divides the audio signal in overlapped frames, using a certain sample frequency: in this way, it is possible to take into account the correlation between consecutive audio frames.

The windowing operation is made using the following *Hamming* function:

$$\mathcal{H}(n) = \alpha - \beta * \frac{cos(2\pi n)}{N-1} \tag{11}$$

being $\alpha = 0.54$ and $\beta = 0.46$, and N the frame length. The windowing operation ensures the elimination of discontinuity in the extreme parts of an audio frame, and allows to highlight the central samples.

We then consider the following features in *time domain*:

- *Volume (V)*: detects the variation of signal amplitude. In order to estimate the volume of frame n of length N, we used the *root mean square* (*RMS*) of the signal amplitude defined as follows:

$$V(n) = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1} x_n^2(i)} \tag{12}$$

  $x_n^2(i)$ being the i-th amplitude sample value of n-th frame.

  The volume is related to the acquisition device: only its variation is thus considered a discriminant value within a scene.

- *Low Short Time Energy Ratio (LSTER)* is defined as the ratio between the frames number whose *Short Time Energy* (*STE*) is less than of 0.5 respect to the average short time energy and the average short time energy itself. The short time energy for a given window ϖ of length K is defined as:

$$STE_{\varpi} = \sum_{i=0}^{K-1} x(i)^2 \tag{13}$$

  $x(i)$ being the i-th sample in the considered interval. $LSTER$ is thus defined as:

$$LSTER = \frac{1}{2I}\sum_{\varpi=0}^{I-1}[sign(0.5 \cdot \mu_{STE} - STE_{\varpi}) + 1] \tag{14}$$

  being I the total number of considered intervals, STE_k the short time energy at k-th interval and μ_{STE} the average STE.

  $LSTER$ is an effective feature, especially for discriminating speech and music signals.

In order to obtain the features in *frequency domain*, we compute the DFT (Discrete Fourier Transform) of the audio signal. Using X to denote the DFT of signal in time domain and M to denote the index of sample having highest frequency, we compute the following features:

- *Signal Energy*:

$$E = \sum_{n=0}^{M} |X(n)|^2 \tag{15}$$

- *Sub Band Energy*:

$$E^B = \sum_{n=0}^{M_B} |X(n)|^2 \tag{16}$$

  being M_B the highest frequency index in sub-band B.

- *Frequency Centroid (FC)*:

$$FC = \frac{\sum_{n=0}^{M} n|X(n)|^2}{\sum_{n=0}^{M} |X(n)|^2} \quad (17)$$

FC represents the average point of the spectral power distribution; this feature is used to differentiate the noisy frame from the silent one.

- *Frequency Bandwidth (FB)* is the size of frequency interval assigned to a signal, it is defined in this way:

$$FB = \sqrt{\frac{\sum_{n=0}^{M}(n - FC)^2|X(n)|^2}{\sum_{n=0}^{M} |X(n)|^2}} \quad (18)$$

- *Spectral Flux (SF)* is a measure of how quickly the power spectrum of a signal is changing, calculated by comparing the power spectrum for one frame against the power spectrum from the previous one:

$$SF = \frac{1}{K} \sum_{n=0}^{K-1} f(X, n)^2 \quad (19)$$

$f(X, n) = (log(|X(n)|+\epsilon) - log(|X(n-1)|+\epsilon)$, ϵ being an appropriate positive value and K being the number of samples of DFT; this feature is used to discriminate music from spoken signals.

- *Cepstral Flux (CF)* is defined as:

$$CF = \sum_{n=0}^{M} \parallel |C(n)| - |C(n-1)| \parallel \quad (20)$$

where C is the cepstral coefficient, computed in this way:

$$C(n) = IFFT(log(FFT(x(n)))) \quad (21)$$

Note that, V describes a volume characteristic; E, ESB, $LSTER$ represent signal energy characteristics; eventually, FB, FC, SF and CF are the spectral characteristics.

4.2 Shot Feature Vectors

Audio traces or *audio shots*, related to the previously calculated video shots, are then used in order to compute a set of features vectors determining attention swings during information acquisition. In fact, audio background in terms of speech, silence, music, and noise etc... can determine attention swings, while a user is observing a video [13]. In particular, for each audio shot s, we use:

- mean value μ_s;
- standard deviation σ_s;
- standard deviation of the differences of the audio frames ζ in which the audio shots s are divided, σ_ζ

We compute such statistics for volume, energy and spectrum characteristics. In addition, for volume we also use *dynamic volume interval* (vdr_s) defined as:

$$vdr_s = \frac{max_s(v) - min_s(v)}{max_s(v)} \quad (22)$$

being $min_s(v)$ and $max_s(x)$ the minimum and maximum volume value inside the audio shot respectively.

In other words, we describe each shot s_i by means of three feature vectors: ($\theta^v_{s_i}$, $\theta^e_{s_i}$, $\theta^s_{s_i}$).
In particular:

$$\theta^v_{s_i} = \langle \mu_s(V_i), \sigma_\zeta(V_i), vdr(V_i), \sigma_s(V_i) \rangle \quad (23)$$

$$\theta^e_{s_i} = \langle \sigma_s(E_i), \sigma_s(E^B_i), LSTER_i, \sigma_\zeta(E_i), \sigma_\zeta(E^B_i) \rangle \quad (24)$$

$$\theta^s_{s_i} = \langle \sigma_s(FB_i), \sigma_s(FC_i), \sigma_\zeta(SF_i), \sigma_\zeta(CF_i) \rangle \quad (25)$$

The vector values are then normalized for comparison reasons.

5. COMBINING VISUAL AND AUDIO FEATURES FOR VIDEO SCENE SEGMENTATION

The detection of scene changes is performed computing the euclidean distance among audio features on two contiguous video shots, and grouping such shots on the base of an euclidean distance and a dynamic threshold: shots whose audio distance is lower than the threshold are joined in a scene. In the following, we report the proposed scene change detection $\mathcal{SCD}$ algorithm.

Algorithm 1: Scene Change Detection ($\mathcal{SCD}$)

Input:
$\mathcal{S} = \{s_1.....s_n\}$
being s_i a video shot,
$\vartheta^v = \{\theta^v_{s_1}....\theta^v_{s_n}\}$,
$\vartheta^e = \{\theta^e_{s_1}....\theta^e_{s_n}\}$,
$\vartheta^s = \{\theta^s_{s_1}....\theta^s_{s_n}\}$
are volume, energy and spectrum feature sets
Th_v, Th_e, Th_s
are three dynamic thresholds.
Output: $\mathcal{SC} = \{sc_1.....sc_m\}$
being sc_j an element of set of scenes video.
begin
  $\mathcal{D}_v = \{\emptyset\}$,$\mathcal{D}_s = \{\emptyset\}$,$\mathcal{D}_e = \{\emptyset\}$
  $\mathcal{SC} = \{\emptyset\}$
  foreach $s_i, s_{i+1} \in \mathcal{S}$ **do**
    $\theta^v_{s_i}, \theta^v_{s_{i+1}} \in \theta^v$
    $\theta^e_{s_i}, \theta^e_{s_{i+1}} \in \vartheta^e$
    $\theta^s_{s_i}, \theta^s_{s_{i+1}} \in \vartheta^s$
    $d^{i,i+1}_v = euclidean(\theta^v_{s_i}, \theta^v_{s_{i+1}})$
    $d^{i,i+1}_e = euclidean(\theta^e_{s_i}, \theta^e_{s_{i+1}})$
    $d^{i,i+1}_s = euclidean(\theta^s_{s_i}, \theta^s_{s_{i+1}})$
    $\mathcal{D}_v \leftarrow d^{i,i+1}_v$, $\mathcal{D}_e \leftarrow d^{i,i+1}_e$, $\mathcal{D}_s \leftarrow d^{i,i+1}_s$
  Th_v, Th_s, Th_e=$builtThreshold(\theta_v, \theta_e, \theta_s)$
  foreach $d^{i,i+1}_v \in \mathcal{D}_v$, $d^{i,i+1}_e \in \mathcal{D}_e$, $d^{i,i+1}_s \in \mathcal{D}_s$
  $s_i, s_{i+1} \in \mathcal{S}$ **do**
    if $d^{i,i+1}_v > Th_v$ *or* $d^{i,i+1}_e > Th_e$ *or* $d^{i,i+1}_s > Th_s$
    then
      $sc_j = s_i$
      $\mathcal{SC} \leftarrow sc_j$
    else
      $s_i = s_i \cup s_{i+1}$
end

In the algorithm we use two functions: *euclidean*, returning the euclidean distance between two homologous feature vectors; *builtThreshold*, which calculates the dynamic threshold as in the following:

$$Th_r = \mu(\mathcal{D}_r^{'}) + \sigma(\mathcal{D}_r^{'}) \quad (26)$$

being $r \in \{v, e, s\}$ and $\mathcal{D}_r^{'}$ the set of euclidean distances computed between $s_{i-\Delta}$ and $s_{i+\Delta}$, $i-\Delta$ and $i+\Delta$ being an appropriate sliding window.

6. PRELIMINARY EXPERIMENTS AND RESULTS

To evaluate the performance of the proposed scene detection algorithm, a database of video/audio sequences has been obtained from documentaries and news belonging to TREC01 video repository and from famous movies.

The selected sequences are complex with extensive graphical effects. Videos were captured at a rate of 30 frames/sec, 640×480 pixel resolution, and stored in AVI format. For each sequence a ground-truth was obtained by three experienced humans using visual and audio inspection [5].

We want to notice that to obtain an estimate of parameters for scene detection, a different training set has been used [2].

Experiments for performance evaluation were carried out on a test set of 6309 sec. of video, which is summarized in Table 1.

Table 1: Description of the video sequences in the test set

ID	Video Sequence	Dur.(sec.)
1	ANNI005 (Doc.TREC01)	245
2	BOR02 (Doc.TREC01)	328
3	BOR07 (Doc.TREC01)	420
4	BOR08 (Doc.TREC01)	350
5	NAD31 (Doc.TREC01)	516
6	NAD33 (Doc.TREC01)	310
7	NAD53 (Doc.TREC01)	692
8	NAD55 (Doc.TREC01)	485
9	SENSES111 (Doc.TREC01)	388
10	Life is beautiful (Movie)	301
11	Harry Potter (Movie)	264
12	Final Destination 2 (Movie)	274
13	Flubber (Movie)	194
14	Gangs of New York (Movie)	237
15	The Lord of the Rings (Movie)	360
16	Signs (Movie)	261
17	Bruce Almighty (Movie)	390
18	Once Upon a Time in America (Movie)	304
-	Total	6309

The comparison between the proposed algorithm's output and the ground truth relies on the well known *recall* and *precision* figures of merit [5]:

$$recall = detects/(detects + MD) \quad (27)$$

$$precision = detects/(detects + FA) \quad (28)$$

detects denoting the correctly detected scene changes, MD and FA the missed detections and false alarms, respectively.

Table 2: Scene Detection performances of the proposed method

Video	Scene-Changes	Det	MD	FA	prec	rec
1	10	12	2	4	75%	86%
2	10	11	3	4	79%	85%
3	11	13	1	3	81%	93%
4	12	11	3	2	85%	79%
5	16	17	4	5	77%	81%
6	9	11	1	3	79%	92%
7	18	21	3	6	78%	87%
8	14	14	2	2	87%	87%
9	12	13	3	4	76%	81%
10	12	16	1	5	76%	94%
11	10	12	2	4	75%	86%
12	12	12	3	3	80%	80%
13	9	13	1	5	72%	93%
14	5	5	1	1	83%	83%
15	11	14	1	4	78%	93%
16	7	8	2	3	73%	80%
17	8	9	3	4	69%	75%
18	9	11	2	4	73%	85%
avg	-	-	-	-	77%	86%

In other terms, at fixed parameters, *recall* measures the ratio between right detected scene changes and total scene changes in a video, while *precision* measures the ratio between right detected scene changes and the total scene changes detected by the algorithm.

The obtained results are provided in Table 2. The proposed method achieves a 86% recall rate with a 77% precision rate on scene changes with respect to the human annotation.

7. CONCLUSIONS

In this paper a novel framework to group shots into scenes based on the analysis of attentive (audio and visual) video content features, has been described and discussed. The reported preliminary results show the effectiveness of the system performances. Future works will be devoted to compare our algorithm with other ones present in the literature an to introduce a particular module able to classify the detected scenes into predefined categories that are function of the video genre.

8. REFERENCES

[1] A. A. Altan, A. Akansu, and W. Wolf, Multi-Modal dialog Scene Detection using Hidden Markov Models for Content-Based Multimedia Indexing, *Multimedia Tools and Application Journal*, 14(2):1380-7501, 2001.

[2] G. Boccignone, A. Chianese, V. Moscato and A. Picariello, Foveated Shot Detection for Video Segmentation, *IEEE Trans. on Circuits and Systems for Video Technology*, 15(3):365-377, 2005.

[3] G. Boccignone, A. Chianese, V. Moscato, A. Picariello, Context-sensitive queries for image retrieval in digital libraries, *Journal of Intelligent Information Systems*, Online first, 2007.

[4] R. M. Ford, C. Robson, Daniel Temple, and M. Gerlach, Metrics for Scene Change Detection in digital Video Sequences, IEEE *Int. Conf. on Multimedia Computing and Systems*, 1997.

[5] U.Gargi, R. Kasturi and S.H. Strayer, Performance characterization of video-shot change detection methods, *IEEE Trans. on Circ. Sys. for Video Tech.*, 10(1):1-13, 2000.

[6] N. Haering, R. J. Qian, and M. Sezan, A Semantic Event-Detection Approach and Its Application to Detecting Hunts in Wildlife Video, *IEEE Trans. on Circuits and Systems for Video Tech.*, 10(6), 2000.

[7] L. Itti, C. Koch, and E. Niebur, A model of saliency based visual attention for rapid scene analysis, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20:1254-1259, 1998.

[8] Z. Liu, Y. Wang, znd T. Chen, Audio Feature Extraction and Analysis for Scene Segmentation and Classification, *Journal of VLSI Signal Processing*, 20(1-2):61-79, 1998.

[9] L. Lu, HJ Zhang, and H. Jiang, Content Analysis for Audio Classification and Segmentation, *IEEE Trans. on speech and audio processing*, 10(7):504-516, 2002.

[10] D. Noton, and L.Stark, Scanpaths in the saccdice eye movements during pattern perception, *Visual Research*, 11:929-942, 1990.

[11] S. Pfeiffer, R. Lienhart, and W. Effelsberg, Scene Determination based on Video and Audio Features, *Multimedia Tools and Application Journal*, 15:59-81, 2001.

[12] Y. Qi, A. Hauptmann, and T. Liu, Supervised Classification for Video Shot Segmentation, *IEEE Conference on Multimedia & Expo (ICME03)*, 2003.

[13] C. Saraceno, and R. Leonardi, Audio as a Support to Scene Change Detection and Characterization of Video Sequences, *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, pages 2597-2600, 1997.

[14] N. Sebe, Q. Tian, E. Loupias, M. Lew, and T. Huang, Evaluation of salient point techniques, *Image and Vision Computing*, 21:1087-1095, 2003.

[15] H. Sundaram, and S. Chang, Determing Computable scenes in Films and their Structures using Audio-Visual Memory, *ACM Iternational Multimedia Conference*, pages 95 - 104, 2000.

[16] G.J. Walker-Smith, A.G. Gale, and J.M. Findlay, Eye movement strategies involved in face perception, *Perception*, 6:313-326, 1997.

[17] Y. Wang, Z. Liu, and J. Huang, Multimedia Content Analysis, *IEEE Signal Processing Magazine*, 12-36, November 2002.

[18] J. Wang, and T. Chua, A Framework for Video Scene Boundary Detection, *ACM Iternational Multimedia Conference*, 2002.