AUDIOVISUAL FUSION WITH SEGMENT MODELS FOR VIDEO STRUCTURE ANALYSIS

M. Delakis[1], **G. Gravier**[2], **P. Gros**[2]

[1]IRISA/University of Rennes 1, [2]IRISA/CNRS
Campus de Beaulieu, 35042 Rennes Cedex, France
{Manolis.Delakis, Guillaume.Gravier, Patrick.Gros}@irisa.fr

Keywords: Video Structure Analysis, Multimodal Fusion, Hidden Markov Models, Segment Models

Abstract

Hidden Markov Models provide a powerful framework for bridging the semantic gap between low-level video features and high-level user needs by taking full advantage of our prior knowledge on the video structure. A serious flaw of HMMs is that they require all the modalities of a video document to be strictly synchronous before their fusion. Taking as a case study tennis broadcasts analysis, we introduce video indexing using Segment Models, a generalization of Hidden Markov Models, where the fusion of different modalities can be performed in a more flexible way. Operating essentially as a layered topology they allow the fusion of asynchronous modalities but do not rely on synchronization points fixed a priori. They also facilitate the fusion of audio models of high-level semantics, like the content of a complete scene, on top of the raw low-level audio frames. Segment Models provide encouraging experimental results.

1 Introduction

Automatic annotation of video documents is a powerful tool for managing large video databases or, more recently, for the development of sophisticated consumer products that meet high-level user needs like highlight extraction. One can accomplish this task by using explicit handcrafted and thus domain-dependent models which can perform reasonably well in some cases [2]. But soon it was realized that we need more effective ways to bridge the required high-level user needs and the low-level video features we have at hand, such as image histograms or speaker excitation. This problem is usually referred to as the *semantic gap* in the relative literature. In the last few years, pattern recognition techniques have been extensively used to extract semantic indexes. Such techniques estimate the parameters of a model given some training data and are therefore better candidates for generalization. Hidden Markov Models [9] (HMMs) is a powerful statistical approach that can model temporal patterns and can be used as a statistical parser of a video sequence [15], sharing notions from the field of speech recognition. Our prior knowledge on the video structure is incorporated implicitly in the model topology thus providing solutions to the semantic gap problem.

As video documents are inherently multimodal, an efficient indexing technique should take into consideration all the possible modalities (like images, audio, text, etc.). There are numerous approaches to multimodal fusion in the relative literature, reviewed in two surveys [14, 12]. The integration of audiovisual features with HMMs has been widely studied in the field of audiovisual speech recognition [8], but audio and visual features are considered as synchronous or nearly synchronous. In video indexing, however, the multiple information sources of a video document such as audio, video, text, etc. are generally asynchronous [11]. A first solution is to treat each modality separately and then to combine outputs of unimodal classifiers (e.g., [4]). A more advanced approach, referred to in the literature as *early integration*, is to merge information from all the modalities into a super-vector of observations and to use a single HMM to model the content (e.g., [4, 5]). However, with this approach, we explicitly assume synchronization between the modalities and, in addition, we force different modalities to follow the same topology. Layered HMMs [6, 16] use the outputs of HMMs operating at low levels to feed HMMs of the next level in a cascade fashion. They provide a number of advantages like fusion at different frame rates and with independent models but they require synchronization at fixed time intervals.

In this study, we introduce video indexing with Segment Models [7] (SMs). They provide a generalization of HMMs where each hidden state can emit several observation symbols instead of a single one. SMs were used in speech recognition in order to account for a more precise modeling of the speech generation process than HMMs can offer. We study their use in video indexing to perform a more efficient multimodal fusion by relaxing the synchrony constraint. As in Layered HMMs, each modality can be modeled independently. But in SMs we are able to extend the synchronization points at the scene boundaries, instead of having fixed ones, facilitating in this way the integration of non-synchronous high-level features. Furthermore, with SMs we are able to fuse models of high-level semantics, like the content of a complete scene, on top of the raw low-level audio frames.

We focus on tennis broadcasts structure analysis, extending previous work [5] based on HMMs. In this type of video, game rules as well as production rules result in a structured document. This prior structural knowledge we have on the video is easily incorporated in Segment Models, as in Hidden Markov Models. Our aim is to recover this structure in order to construct the table of contents of the video by segmenting it in human meaningful scenes. The semantic indexes thus obtained can meet the high-level user needs or can be used for managing large video databases.

This paper is organized as follows. The extraction of audio and visual features is discussed in section 2. In section 3 we see how the visual content is modeled by HMMs and SMs, providing in parallel a review of them and pointing out differences. Multimodal integration with these models is discussed in section 4. Parameter estimation details are given in section 5 and experimental results in section 6. Finally, section 7 concludes this study.

2 Visual and Audio Features

One can easily notice that large homogeneous segments characterize both the video and audio tracks. For the video track, where this effect is stronger and more visible, these segments are obviously the video shots. For the audio track, we consider segments whose content is homogeneous with respect to sound classes such as ball hits or applause. Having performed the sound and video track segmentations, we extract a unique visual or audio descriptor from these segments. These descriptors (or *observations* in the HMM terminology) are used as input features to the modeling stages of the following sections. We used a corpus of 6 tennis games, amounting to a total duration of 15 hours. Half of the games were reserved for testing purposes.

2.1 Visual Features

In order to detect the hard cuts of the video track we implemented the adaptive threshold selection method of [13]. Starts and ends of replays are usually signaled by smoothed progressive transitions between two shots, known as dissolve transitions, which are much harder to detect as they extend through time. Having detected the hard cuts, we proceed in a further investigation to see if a shot contains dissolve transitions. A first condition that is usually met during a dissolve transition is that for all the video frames f_i that belong to the dissolve, the bin-wise histogram distance of the consecutives frames f_i and f_{i+1} should be greater than a predefined threshold. As this condition is easily met due to motion fields in the video, we also used a second metric that will fully exploit the spatiotemporal smoothness appearing during a dissolve. Let denote by H_{t_0} the histogram of the absolute pixel-wise difference between the frames f_{t_o} and f_{t_o-1}. The second metric we used is:

$$D_s(t_o) = \sum_{k=\tau+1}^{M} \left(H_{t_o}(k) - \tau\right)^2 ,$$

where M is the total number of bins of the histograms and τ is a threshold for rejecting small differences produced mainly by image noise. With these two metrics we can detect a dissolve and also define its temporal extension. Some heuristics are then applied like rejection of false alarms after lighting compensation and fusion of very small dissolves. Finally, for every dissolve detected and given its temporal extension we formed a new type of shot labeled as ‘dissolve shot’. Overall, the corpus contains 11,971 shots, among which 1,196 correspond to dissolve transitions. The hard cut detection was almost perfect, while the dissolve transition gave 10 false alarms and 3 misses on the training data and 18/22 respectively on the test data.

Regarding the actual feature extraction, we are primarily interested in shots of exchanges between the two players (i.e., where the game actually takes place), referred to as ‘global views’ in this paper. In [3] global views were detected via edge information and the Hough transform, concerning only the video content. A simpler solution is to use color-based features as global views are dominated by the court color. The drawback of using color is that the court color changes from video to video and may also undergo slight changes in the same video due to lighting conditions. In [17], a semi-adaptive procedure was followed for detecting global views, provided that initial models of global views will lie close enough to the target global view. We propose a fully automatic procedure for detecting the global views in a tennis video without requiring any prior approximation of the color of the court. The idea consists in representing each shot by a key frame[1] and finding out among all the key frames a typical global view of the tennis court. Shots are then labeled according to their visual similarity to the prototype global view. To do so, the dominant colors of each key frame is determined by applying the k-means algorithm in the LUV color space. Most of the frames where the dominant color is present in more than 70% of the pixels correspond to global views. Such frames provide a list of candidates of the prototype global view K_{ref} which is then determined using the least median of squares method [10]. A prototype key frame is determined for each video. Finally, the visual similarity

[1] The key frame of a shot is taken as the frame in the middle of the shot.

of a key frame to K_{ref} is computed using a metric combining the simple bin wise distances of the LUV and edge histograms of the two frames. The use of edge information compensates the sensitivity of color histograms to lighting variations.

As a final result, a visual descriptor $O_t = [O_t^{vs}\ O_t^l\ O_t^{diss}]^T$ is defined for every key frame t, where O_t^{vs} is the visual similarity, O_t^l the shot length and O_t^{diss} indicates a dissolve shot or not. T denotes matrix transposition. We quantized homogeneously the values of O_t^{vs} and O_t^l into 10 bins each.

2.2 Audio Features

To caracterize the content of the soundtrack, we track the presence of the following key sound classes: music, applause, and ball hits. Tracking such events is carried out in a two step process as described in [1]. First, the soundtrack is segmented into homogeneous segments using a Bayesian information criterion. It is important to note that this segmentation is carried out independently of the shot segmentation. The detection of the key sound classes is carried out independently in each segment, the presence or absence of sound classes being detected using statistical hypothesis testing with Gaussian mixture models (GMMs). For each key sound i, we build a GMM to model the presence of the key sound ($X_i = 1$) and a GMM to model the absence of the key sound ($X_i = 0$). Given such models, the sound events presents are determined by maximimizing $\prod_{i=1}^{3} p(y|x_i)p(x_i)$ over x, where y is serie of cepstral coefficients representing the segment. The GMMs parameters and the prior probabilities $p(x_i)$ are estimated on the training corpus annotated by human listeners.

3 Visual Content Modeling

Our aim is to decode the tennis game according to some pre-identified scenes, namely first missed serve and exchange, exchange, replay and break, defined on top of the video shot segmentation. The first two scenes correspond to the actual points of the game while the last two are useful for highlight extraction (replays) or dead time removal (break) from the video. Each scene is defined as a collection of shots with well-defined starting and ending points. For example, an exchange scene is defined as a collection of shots with the first of them being the shot depicting the exchange itself and followed by a number of shots until a new scene begins. In this work, the succession of the scenes is modeled by an ergodic HMM though structural information on the rules of a tennis game could be introduced both for HMMs and segment models. In the first part of this section, we discuss how to model a scene also using an HMM (the resulting model also being an HMM), while in the second one we extend this approach to use Segment Models, where a segment corresponds to a scene. The decoding process involves the labeling of each shot as belonging to one of the scenes and the detection of the starting and ending points (i.e., the boundaries) of each scene.

3.1 Hidden Markov Models

The content of each scene individually can be modeled by an HMM. In Fig. 1.a we see an illustration of the HMM that models the missed serve and exchange scene. It is modeled as follows: a first missed serve corresponding to a shot of global view (state 1), then follow some shots of non-global views (state 2), a shot of global view of a normal exchange (state 3), and finally, some shots of non-global views after the exchange (state 4). There is also the possibility to transit from state 2 back to state 1 in cases of repeated missed serves. In a similar manner, we define the HMMs for the remaining three scenes, depicted in Fig. 1.b. The HMM of the exchange scene is straightforward to explain from the discussion above. The replay scene is a succession of dissolve transitions and non global views. Break scenes may contain dissolve transitions detected in commercials, too. These last two scenes also contain some global view shots (states 9 and 12) but they do not correspond to an exchange or missed serve as they do not contain game action. They appear as false detections, unfortunately inevitable as the visual content is exactly the same with the true global views of game action. Finally, to describe the content of the entire video, we simply interconnect the individual HMMs (Fig. 1.c) to form a large HMM with 12 hidden states that models the video as a succession of shots.

Assuming the parameters of the model are known, we can use it to decode the video, perceived as an observation sequence of shot descriptors, to the corresponding most likely hidden state sequence. This decoding translates to the optimization problem:

$$S^* = \arg\max_{s_1^T} p(O_1^T|s_1^T)p(s_1^T)$$

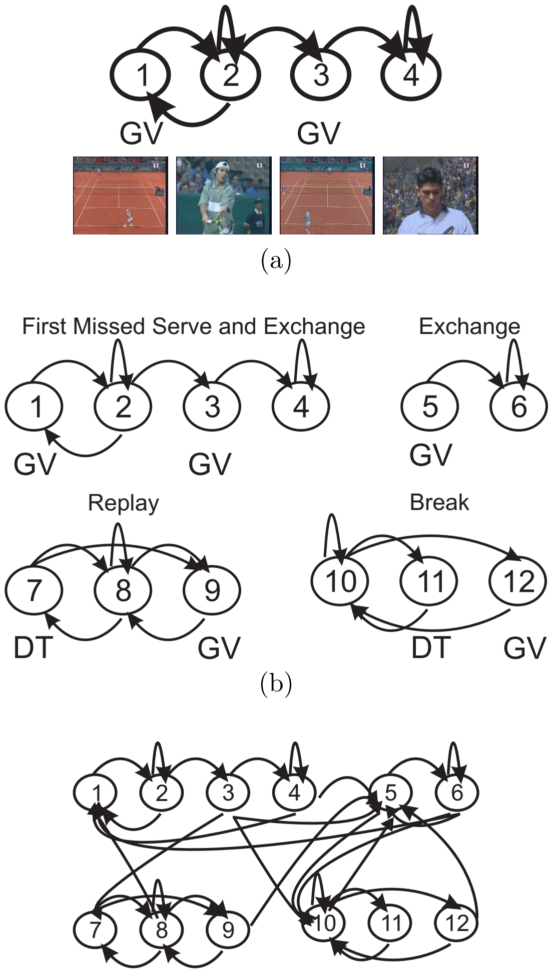
where $s_1^T = (s_1, s_2, \ldots s_T)$ is the hidden state sequence, $O_1^T = (O_1, O_2, \ldots O_T)$ is the observation sequence and T is the sequence length, that is the total number of shots of the video. The probabilities terms are defined as follows:

$$p(O_1^T|s_1^T) = \prod_{t=1}^{T} b_{s_t}(O_t)$$

is the likelihood of the observations O_t given the hidden states s_t and

$$p(s_1^T) = \prod_{t=2}^{T} P(s_t|s_{t-1})$$

is the likelihood of the hidden state sequence s_1^T. This optimization problem is solved efficiently and fast via the Viterbi algorithm. The state sequence S^* gives us the wanted human meaningful class labels of each video shot. Finally, the actual segmentation and labeling into scenes



(a)

(b)

(c)

Figure 1: (a) The HMM that models the visual content of the firt missed serve and exchange scene. (b) The 4 HMMs used to describe the content of each scene. 'GV' stands for 'global view' and 'DT' for 'dissolve transition'. (c) The states of each scene are interconnected to form a large HMM that models the video as a succession of shots.

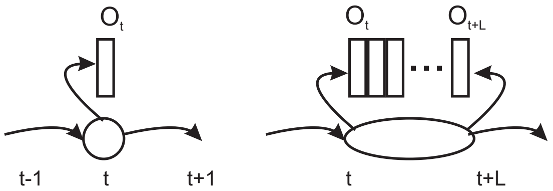


Figure 2: The generation of the observation sequence according to Hidden Markov Models (left) and to Segment Models (right).

(as opposed to shots) is straightforward as we know to which scene belongs each hidden state.

3.2 Segment Models

The key idea of SMs is that the observation at the state level is a sequence of observations, called *segment*, rather than a single feature vector. Therefore, each state in a SM defines a duration model that accounts for the segment length and an emission probability distribution of a sequence. From a generative point of view, this can be seen as a Markovian process that emits a sequence of observations whose length is governed by a duration model before transiting to another state. The difference between HMMs and SMs is illustrated in Fig. 2. On the left, we see what happens conceptually in the case of HMMs: at a given time instant the process is in a given state and generates one observation symbol and then transits to another state. On the right, we see how a sequence is generated according to Segment Models. At a given time instant, the stochastic process enters into a state and remains there according to a probability given by the segment duration model. A sequence of observations is generated, instead of a single one, according to a distribution conditioned on the segment label. Then the process transits to a new state with a transition probability, as in HMMs, and so on until the complete sequence of observations is generated.

In our tennis video case, we can think of a scene as a segment. Indeed, we can observe that the complete sets of observations of the scenes of Fig. 1 share a lot of common elements. For example, a scene of a break is an ensemble of shots of very short (commercials) or long (statistics) duration. In addition, we expect that all the break scenes will be of long absolute duration while the scenes of replays should be of short absolute duration. Under this understanding of segments, the video content is finally described as a succession of scenes. The situation is depicted in Fig. 3. The stochastic process enters into the hidden state of the first scene, emits a segment, which is a number of observations corresponding to the shots of the scene, and finally transits to a new hidden state. The model is ergodic except the non-allowed loop-back transitions of the last two scenes (e.g., a replay followed by another replay is considered as a unique replay scene).

The Segment Model is defined by the transition proba-

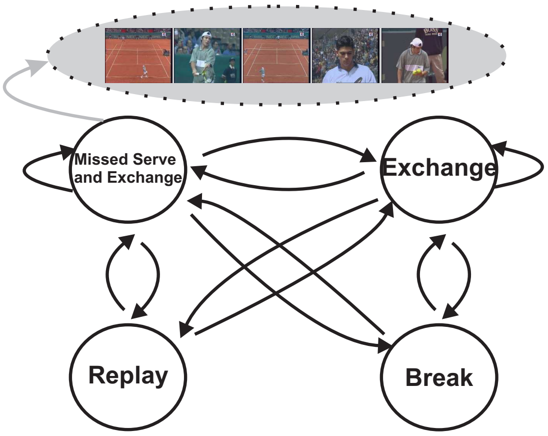


Figure 3: The Segment Model we used to model the video content.

bilities $P(i|j)$ from state j to state i, the duration models $p(l|a)$ and the segment-level observation probabilities $b_a(O_1, O_2, \ldots, O_l)$, conditioned on the segment label a (in their general formalism of [7], the observation probabilities were also conditioned on the segment duration l). While it is straightforward to express the law of the duration model (e.g., by using Gaussian or Poisson models or, more simply, by histograms), we have a lot of possibilities regarding the observation probabilities, as they are in charge of modeling sequences of data. Details on their approximation are given later in section 4.

During the decoding with the Segment Model, we have now to find not only the most likely segment labels, but also the most likely *segmentation* or, in other words, the most likely duration of each segment. This new enhanced maximization problem is formulated as:

$$(L, A)^* = \arg \max_{l_1^N, a_1^N} p(o_1^T | a_1^N, l_1^N) p(l_1^N | a_1^N) p(a_1^N)$$

where T is again the observation sequence length, N is the (unknown) number of segments (scenes), a_1^N the segment labels and l_1^N the segment durations (number of shots contained in each scene). The first term is the segment sequence given the hidden state sequence and the segmentation:

$$p(O_1^T | a_1^N, l_1^N) = \prod_{n=1}^{N} b_{a_n}(O_t O_{t+1} \ldots O_{t+l_n}).$$

The second term expresses the duration of each segment:

$$p(l_1^N | a_1^N) = \prod_{n=1}^{N} p(l_n | a_n).$$

Finally, the expression of the hidden state sequence is very similar to the one for HMMs:

$$p(a_1^N) = \prod_{n=2}^{N} P(a_n | a_{n-1}).$$

This problem is solved via a straightforward extension of the Viterbi algorithm for HMMs, described in [7]. Intuitively speaking, during the Viterbi for HMMs with N hidden states we apply the following maximization at each time step t and for each state i:

$$\delta_i(t) = \max_{1 \leq j \leq N} \delta_j(t-1) P(i|j) b_i(O_t)$$

where $\delta_i(t)$ is the score of the best path ending at time t in state i. For SMs, we extend the search also to previous time steps k to account for multiple segmentation possibilities:

$$\delta_i(t) = \max_{1 \leq j \leq N, 1 \leq k \leq M} \delta_j(t-k) P(i|j) p(l_k|i) b_i(O_{t-k+1} \ldots O_t) \tag{1}$$

To avoid unecessary computation we restricted our search for possible segmentations into a window of M time steps (or shots). We set this value to 70 as it is difficult to have scenes containing more than 70 shots. Supposing that the computation of the $b_i(O_{t-k+1} \ldots O_t)$ can be performed in $O(1)$ time, the Viterbi for SMs will give a computation cost of roughly M times higher than that of the HMM-based Viterbi algorithm, that is $O(MNT)$. If the computation cost of the observation probabilities scales linearly (or higher) with time, then the cost grows exponentially with M. But there are still some caching solution to speed-up the process as discussed in section 4.

4 Multimodal integration

In section 3 the observation vector was limited to a single vector of visual shot-based descriptors as we were concerned only with the visual content. The audio content however is an important source of information that should be taken into consideration in our modeling. For example, states 1, 3, and 5 of Fig. 1 are visually very similar as they correspond to the same global view type of shot. The length of these shots can give some hints as exchanges are generally longer than missed serves, but this is not always the case due to aces, idleness, etc. What can essentially differentiate the first state from the other two is the absence (state 1) or the presence (states 3 and 5) of applause after the exchange has finished. In addition, the states 9 and 12 can be detected reliably only with the absence of ball hits during these shots.

4.1 Fusion with Hidden Markov Models

In the HMM framework each state is strictly related to a unique observation symbol O_t. As a consequence, HMMs allow very little flexibility regarding the fusion of multiple modalities: they should be artificially aligned and synchronized. A common approach is to choose a reference modality (the video track, in our case). Using its segmentation (segmentation into shots, in our case), we segment the other modalities accordingly. We then extract descriptors from the other modalities and concatenate them to the observation vector of the reference modality. In this manner, we collect information from the other sources not based on their native segmentation but in an indirect way via the segmentation of the reference modality. The enhanced observation vector for the HMM is

$$O_t = [O_t^{vs}\ O_t^{l}\ O_t^{diss}\ O_t^{bh}\ O_t^{appl}\ O_t^{m}]^T \tag{2}$$

where O_t^{vs}, O_t^{l}, O_t^{diss} were defined in section 2, O_t^{bh} is a binary descriptor that denotes the presence or absence of ball hits, O_t^{appl} of applause, and O_t^{m} of music in the shot. We suppose independency between all the components of the observation vector which result in a simple product of discrete probabilities for the expression of $b_i(O_t)$.

4.2 Fusion with Segment Models

There are various ways to approach feature modeling in Segment Models. We can firstly model each scene using HMMs that will act as observation *scorers*, i.e., they will provide the likelihood that a given observation sequence belongs to one of the four segment classes. This will result into the following expression for the emission probabilities:

$$b_a(O_1 O_2 \ldots O_t) \equiv P(O|\lambda_a) = \sum_Q P(O, Q|\lambda_a), \tag{3}$$

where λ_a represents the HMM charged to model the observations of the segment a and Q is a hidden state sequence of it. The last term is merely the sum over all the possible hidden states of the HMM which is provided easily and efficiently by the forward pass of the forward-backward procedure [9].

When fusing these models with audio information in the form of shot-based audio descriptors, as in HMMs, we will refer to the 'VhmmAdescr' approach. In this case, the symbol O_t is as defined in eq. (2). When not using audio observations, we will call this approach 'Vhmm' in the remainder of this paper. Note that when choosing to model the segments via HMMs, then SMs resemble to Layered HMM topologies. Indeed, we have modeled the segment as a Markovian process that gives its output to a new Markovian process (succession of the hidden states of the SM) in a cascade fashion. The difference between the two models is that the synchronization points in SMs are not a-priori fixed and their detection is part of the optimization problem.

As we have essentially extended the synchronization points between audio and video to the scene boundaries in SMs, we can describe the audio content using its native audio-based segmentation. So, instead of collecting a number of descriptors for each shot, we can use features like 'presence of applause in the scene', etc to characterize the audio content of a complete scene. The visual features are still modeled via HMMs as in eq. (3). We will call this approach 'VhmmA1gram'. Another possibility is to use as features the succession of audio events in the segment, which can be done simply by a bigram model:

$$b_a(O_1^{\mathrm{a}} O_2^{\mathrm{a}} \ldots O_t^{\mathrm{a}}) = \prod_{k=2}^{t} P(O_k^{\mathrm{a}}|O_{k-1}^{\mathrm{a}}, a),$$

where O_t^{a} is a symbol indicating the detection of applause, ball hits or music in the segment a. We will call this approach 'VhmmA2gram'. Note that the sample rates of the symbols O_t^{a} (one per audio event) and O_t^{v} (one per video shot) are different.

Finally, we can model the audio content of a scene directly on top of the audio cepstral coefficients instead of using manually extracted audio descriptors. The reason for doing this is twofold: we avoid firstly the erroneous pre-segmentation of the audio track into homogeneous segments. The bounds of these segments are hard to detect and generally more vague that the hard cuts of the video track. Secondly, the content of these segments can contain more than one audio classes like ball hits superimposed by speech. This may make the pre-classification to sound classes and the extraction of audio descriptors from these segments erroneous. We will now use HMMs as observation scorers as in eq. (3). They will operate at the sampling rate of the audio track, fixed at 100 frames per second, and will have their own suitably adapted topologies. We will call this approach 'VhmmAcep'.

As a final note regarding computational cost, we can safely assume that the cost of calculating the observation scores is nearly independent of the length of the segment as long as the sampling rate is low. This is the case of the HMMs for the video content and audio event products. But as the audio sampling rate is generally of the order of 100 fps, it is clear that we need a strategy to speed up the computation of the observation scores during the Viterbi decoding. This can be naturally achieved by eliminating redundant computation. Indeed, having a look at the maximization step of eq. (1), we see that we need to evaluate successivelly the quantities $b_i(O_t)$, $b_i(O_{t-1}, O_t)$, etc (time t refers to video shot and not to audio frames in this context). These quantities start at different points but their end is common. The use of the backward pass to evaluate the likelihood of eq. (3), instead of the forward pass, will permit to reuse the computation of $b_i(O_{t-k+1} \ldots O_t)$ when we want to compute $b_i(O_{t-k} \ldots O_t)$ and so on. Using such an optimization, the argument that segment models are M times slower than HMMs still holds, where M is the length of the window in eq. (1).

5 Parameter Estimation

The entire corpus was automatically segmented into shots as described in section 2 and the resulting shots were manually aligned into scenes and states according to the HMM model defined in section 3. We identified a number of 1,792 scenes. Given this reference state alignment, it is straightforward to compute most of the parameters of the models. The transition probabilities are estimated according to the relative frequency of occurrence of the respective transition between hidden states. As observations are discretized, observation probabilities can also be estimated by the relative frequency of occurrence of the symbol.

Especially for the Segment Model, the segment duration law $p(l|a)$ is approximated using a 30-bin histogram of the absolute scene duration expressed in seconds. The visual and audio-visual HMMs used to model the sequence of shots within a segment were initialized according to the topology depicted in Fig. 1.b (i.e., same number of states and the allowed transitions were identical to the dominant transitions of the figure). The parameters were then estimated using the standard Baum-Welch algorithm. A simple back-off scheme was used for the estimation of the audio bigram probabilities in order to avoid null probabilities for unseen sequences. The estimation of the parameters of the HMMs of the 'VhmmAcep' approach is much harder as they have continuous-density observation distributions. We used the HTK toolkit [2] to this end. After a little experimentation, we concluded to a left-right model with transitions $a_{ij} = 0$ when $j > i+1$ and with 20 hidden states. The audio frames consist of 12 cepstral coefficients and the energy, plus the first order derivatives.

6 Experimental Results

The first three games of our corpus were used as training set to estimate model parameters while the last three as a test set. Performance is measured firstly in terms of the percentage C of shots assigned with the correct scene label. We also need a measurement for the quality of the segmentation which is simply given in terms of recall R and precision P rates on the detection of the scene boundaries. To make the comparison easier than comparing all the above three quantities we also used a combined measurement, defined as $\frac{3CPR}{C+P+R}$. As the ground truth of the games was collected on top of the video track segmentation, errors of the hard cut and dissolve detection are not taken into account in this analysis. Results on the test set are reported in table 1.

We first see the performance of the HMM of section 3.1 without (HMMs-V) or with (HMMs-AV) audio observations. As expected, the performance is improved when adding audio information in the observations. We see in the next five rows of table 1 the performance of Segment Models under various observation modeling alternatives. Firstly, in a direct comparison between HMMs and SMs without any audio integration, it is clear that SMs provide better results (SMs-Vhmm compared to HMMs-V). This suggests that the SM of Fig. 3 is a more realistic model of the stochastic process at hand. This advantage of SMs over HMMs and around with the same difference in performance remains when fusing audio information with the form of shot based descriptors to both models (approaches HMMs-AV and SMs-VhmmAdescr).

Table 1: Experimental results for various approaches on the test set regarding the average percentage of correct classification (C), recall (R), precision (P) rates and the combination $\frac{3CPR}{C+P+R}$.

	C	R	P	Comb.
HMMs-V	77.52	82.17	73.66	60.32
HMMs-AV	80.66	85.30	79.45	66.83
SMs-Vhmm	80.99	84.24	75.45	64.17
SMs-VhmmAdescr	84.50	86.53	79.35	69.52
SMs-VhmmA1gram	81.05	85.49	76.43	65.39
SMs-VhmmA2gram	82.12	84.06	79.48	67.00
SMs-VhmmAcep	80.99	85.21	75.39	64.60

In the remaining three rows of table 1 we see the performance of SMs when fusing native audio information. The approach VhmmA1gram gives performance slightly better compared to SMs-Vhmm. With this model, audio events that are asynchronous to the visual observations are used, but the succession of these events in the scene cannot be captured. This is important as the audio events of the two upper scenes of Fig. 1 occur with a strict temporal order. The succession of audio events can be captured effectively under the VhmmA2gram approach, where we clearly notice a performance improvement. Comparing VhmmA2gram with SMs-VhmmAdescr we see that the audio shot-based descriptors yield a slightly better performance. This may suggest mainly that the video and audio modalities do not suffer extensive asynchrony.

Finally, we see that the performance of SMs-VhmmAcep is marginally better compared to SMs-Vhmm. This is very encouraging and suggests that the direct fusion of low-level audio information can indeed yield the same level of performance than the integration of mid-level features. With this approach we avoid the pre-segmentation and pre-classification stages for the detection of sound classes. Despite the fact that the parameter estimation of these models is performed from-the-scratch in a high-dimensional space, it seems that there is indeed the possibility to bridge low-level audio information with high-level semantics like the content of a complete scene.

7 Conclusions

We proposed an alternative modeling of a video sequence based on Segment Models. The synchronization points between the different modalities in these models are not fixed a priori but are part of the Viterbi optimization. SMs thus offer a great deal of flexibility regarding the fusion of multiple modalities. We saw the fusion of asynchronous audio event descriptors to the visual ones. Furthermore, SMs facilitate the direct fusion of raw audio features like cepstral coefficients. Despite the fact the SMs operate in

[2] HTK toolkit Homepage: http://htk.eng.cam.ac.uk

an enhanced search space, they can achieve the same level of performance or better. In addition, the computational cost of the Viterbi decoding scales linearly with time as in HMMs.

Future work includes the incorporation of high-level game semantics into the Segment Model. We also plan to extend this framework to other domains of sport video, as an alternative to HMMs.

Acknowledgements

This work was partially supported by funding from EC Network of Excellence MUSCLE (FP6-507752).

References

[1] M. Betser and G. Gravier. Multiple events tracking in sound tracks. In *Intl. Conf. on Multimedia and Exhibition*, 2004.

[2] R. Brunelli, O. Mich, and C. M. Modena. A survey on the automatic indexing of video data. *Journal of Visual Communication and Image Representation*, 10(2):78–112, 1999.

[3] R. Dahyot, A. Kokaram, N. Rea, and H. Denman. Joint audio visual retrieval for tennis broadcasts. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 561–564, 2003.

[4] J. Huang, Z. Liu, Y. Wang, Y. Chen, and E.K. Wong. Integration of multimodal features for video classification based on HMM. In *Proceedings of IEEE Signal Processing Society Workshop on Multimedia Signal Processing*, pages 53–58, 1999.

[5] E. Kijak, G. Gravier, P. Gros, L. Oisel, and F. Bimbot. HMM based structuring of tennis videos using visual and audio cues. In *Proc. of the ICME*, volume 3, pages 309–312, 2003.

[6] N. Oliver, A. Garg, and E. Horvitz. Layered representations for learning and inferring office activity from multiple sensory channels. *Computer Vision and Image Understanding*, 96(2):163–180, 2004.

[7] M. Ostendorf, V. Digalakis, and O. Kimball. From HMMs to segment models: A unified view of stochastic modeling for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 4(5):360–378, 1996.

[8] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A.W.Senior. Recent advances in the automatic recognition of audiovisual speech. *Proceedings of the IEEE*, 91(9):1306–1326, 2003.

[9] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, 1989.

[10] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, 1987.

[11] C. Snoek and M. Worring. Time interval maximum entropy based event indexing in soccer video. In *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME)*, pages 481–484, July 2003.

[12] C. Snoek and M. Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools and Applications*, 25(1):5–35, 2005.

[13] B.T. Truong, C. Dorai, and S. Venkatesh. New enhancements to cut, fade, and dissolve detection processes in video segmentation. In *Proc. ACM on Multimedia*, pages 219–227, 2000.

[14] Y. Wang, Z. Liu, and J. Huang. Multimedia content analysis. *IEEE Signal Processing Magazine*, 17(6):12–36, 2000.

[15] W. Wolf. Hidden markov model parsing of video programs. In *Proceedings of ICASSP*, pages 2609–2611, 1997.

[16] D. Zhang, D. Gatica-Perez, S. Bengio, I. McCowan, and G. Lathoud. Modeling individual and group actions in meetings: a two-layer HMM framework. In *IEEE Workshop on Event Mining at the Conference on Computer Vision and Pattern Recognition, CVPR*, volume 7, pages 117–124, 2004.

[17] D. Zhong and S.-F. Chang. Structure analysis of sports video using domain models. In *IEEE International Conference on Multimedia and Expo*, 2001.