Score oriented Viterbi search in sport video structuring using HMM and segment models

M. Delakis, G. Gravier, P. Gros
IRISA
Campus de Beaulieu, 35042 Rennes Cedex, France
Email: {Manolis.Delakis, Guillaume.Gravier, Patrick.Gros}@irisa.fr

***Abstract*— A key question in video indexing is the effective use of all the possible sources of information. Hidden Markov models (HMM) and segment models (SM) provide powerful frameworks for audiovisual integration and structure knowledge encoding. However, incorporating punctual symbolic information, such as inlaid score labels, with these models is not straightforward. We demonstrate that using the score labels as an additional feature is not efficient and propose a novel algorithm to efficiently incorporate the score information in the Viterbi decoding process. This score oriented Viterbi search guarantees an optimal solution consistent with the available score information. Experimental results demonstrate the effectiveness of the method and its robustness to inlaid score detection errors.**

I. Introduction

Since video documents are inherently multimodal, it is clear that all the modalities should be taken into account in order to achieve efficient indexing algorithm. Numerous approaches to multimodal fusion, reviewed in [5], have been proposed over the past few years. In particular, Hidden Markov models (HMM) provide a powerful framework for the joint modeling of low-level video features such as image histograms or audio descriptors. One of the advantages of HMMs is that prior knowledge on the video structure can be easily incorporated in this framework using an appropriate topology. In a previous study [1], we experimented segment models (SM), a generalization of the HMM framework for video structuring. While multimodal HMMs require synchronized feature extraction for each modality, SMs loosen the synchrony constraints to scene boundaries. However, incorporating punctual semantic information, such as the inlaid score labels in sport broadcasts, into the HMM or SM framework is not straightforward. The score labels occasionally displayed on screen provide important high-level information on the game events and evolution, but are not efficiently exploited if used as an extra feature as shown in this paper. We propose instead a novel search strategy that uses score labels to guide the Viterbi search algorithm, in order to provide a solution consistent with the available information. Since score labels are not strongly synchronized with the image and audio streams, we believe that this information can be more efficiently handled with SMs than with HMMs.

The paper is organized as follows. In the first section, we review the HMM and SM formalism for tennis video structuring. We then discuss how score labels can be used in section III before presenting some experimental results in section IV.

II. Audiovisual Content Modeling

In the targeted application, our aim is to decode a tennis game according to some preidentified scenes, namely first missed serve plus exchange (FMS+E), exchange (E), replay (R) and break (B). Each scene is a sequence of shots with well-defined start and end times. For example, an exchange scene is a sequence of shots where the first shot contains the exchange itself and the remaining shots are fillers up to the beginning of a new scene.

In this section, we briefly recall the results previously published in [1] and show how scenes can be modeled with HMMs and SMs respectively. We also show how these models can be used with a hierarchical topology to take into account structural knowledge about a tennis game. For all the models discussed here, the same visual and audio features were used. First, hard cuts and dissolve transitions were detected on the video track. Each shot is then characterized by three visual descriptors indicating the shot length, the presence/absence of dissolve transition and a color-based distance to a reference tennis court view. The soundtrack is characterized by its content in terms of presence of music, applause and ball hits.

A. Hidden Markov Models

In the HMM framework, each scene is modeled with a HMM where the states correspond to shots. A study of broadcasting rules for tennis show that all the scenes can be modeled with a grand total of 12 states, as depicted in figure 1. Typically, some states represent court views corresponding to exchanges while other are used to catch dissolve transitions due to rediffusions or close-ups between two exchanges [2]. The feature vectors associated with shots include the three visual descriptors and three binary audio features characterizing the audio content of the shot in terms of music, applause and ball hits. All these features are assumed to be independent. Finally, scenes are connected assuming an ergodic scene structure. Given models whose parameters have been estimated, an unlabeled sequence of shots is decoded using the Viterbi algorithm [4] to find the most likely scene segmentation.

0-7803-9752-5/06/$20.00 ©2006 IEEE

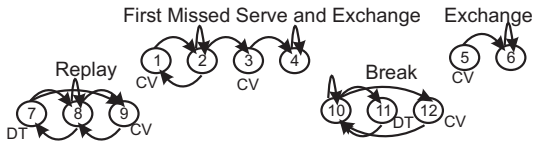

Fig. 1. The 12 states of the HMM we used, grouped into four scenes. 'CV' stands for 'court view' and 'DT' for 'dissolve transition'.

B. Segment Models

Segment models [3] provide a generalization of HMMs where a state models a whole sequence of shots, called segment, rather than a single shot. Each state therefore defines a duration model to account for the segment length and a model of the observation sequence to compute the state conditional probability for a sequence of feature vectors. As for HMMs, scenes are connected assuming an ergodic structure. In the multimodal framework, distinct state conditional probability models are provided for each stream of information (video and audio). The use of segmental features, as opposed to shot-based ones, can be beneficial for multimodal integration. Indeed, the audiovisual synchrony is now extended to the scene boundaries, allowing the use of separate and suitably adapted models for the audio content. In this study, the conditional probability of the audio features of a segment is given by a bigram model. The video features conditional probability is modeled by a HMM whose purpose is to give the probability of a sequence. Decoding in the SM framework aims at finding out 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 problem is solved via a straightforward extension of the Viterbi algorithm for HMMs with explicit state duration [4].

C. Hierarchical Scene Transitions

Tennis games exhibit a highly hierarchical structure with transitions between points, games and sets. Such a structure can easily be encoded in the HMM and SM frameworks using an adequate topology as depicted in figure 2, where the hierarchical structure encodes the game structure and serves as a source of constraints. Internal hidden states are thus introduced to guide the hierarchical structure of the Markovian process. At the top level, the root state of the model represents the complete tennis match. It is in turn modeled as a sequence of sets, occasionally interrupted by breaks. Each set is further analyzed as a succession of games and breaks, with specific interlacing rules. Each game is then analyzed as a succession of points with a minimum of 4 points. Finally, each point is either due to a first miss serve plus exchange scene or an exchange scene, optionally followed by a replay. In the SM framework, the states in white in Fig. 2 are the emitting states for which a segment is produced. In the HMM framework, these states are further expanded into the corresponding HMM.

III. Integration of score information

In tennis games, as in other sports, the score is usually inlaid on the screen after some game event has occurred,

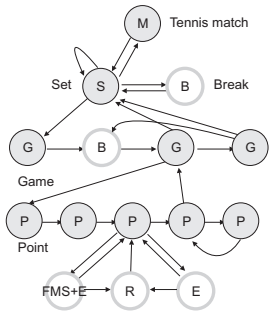

Fig. 2. The internal states of the hierarchical topology.

e.g. when a point is scored. In this section, we present two strategies to exploit this information. In this study, score labels were manually detected and extracted, although automatic techniques could have been used (see *e.g.*, [6]). We discuss the problem of detection errors in the last part of this section and experimentally show in the next section that our method is robust to detection errors.

A. Score Label as a Feature

A straightforward approach to integrate score information is to perform a fusion at the feature level by adding a descriptor indicating whether a score label is present or not. In this case, the labels are used essentially to *spot* the relative game events. In the HMM approach, a score presence/absence descriptor is associated to each shot, thus implicitly assuming a synchrony between the display of the score label and the corresponding scoring event. Limited asynchrony can be easily handled with SMs, where we use a descriptor at the scene level.

However, there are two main drawbacks to the feature fusion approach. Firstly, we use the information that a score has been displayed but we do not use the information of what the score is. Secondly, this approach is efficient under the unrealistic assumption that the score is displayed after each scoring event. In practice, this is not the case and we observed up to four consecutive points not acknowledged by a score label.

B. Score-Oriented Viterbi Search

An effective use of the score labels should not make any assumptions on the producer's style and should tolerate extensive asynchrony and undisplayed labels. To that extent, we propose a search strategy that will choose among all the possible paths, the most likely one that is consistent with the score indications available. An obvious solution for this problem would be an N-best search where we choose a posteriori the most consistent path among the N best candidates. However, to ensure an optimal solution, the number of best candidates to compute must be prohibitive, thus making this approach inefficient in practice. We therefore propose a score oriented search algorithm guaranteed to find out the best path consistent with the score labels.

Before proceeding to the description of the algorithm, let us note that a score label is displayed after the corresponding game event occurred. Hence, the scene to which a label refers generally starts before, and might end after, the actual display

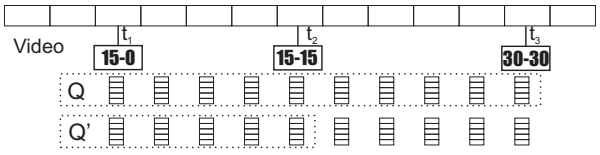

Fig. 3. Illustration of the local search.

of the score label. This means that the scene boundaries lie in time somewhere between the previous and next score labels instance time as depicted in Fig. 3, where the scene related to the score label displayed at t_2 should lie between t_1 and t_3. The idea of our approach is to perform a *local search* through all the paths between t_1 and t_3 and penalize those paths inconsistent with the score indication, before recombining the local search results. In the example of Fig. 3, only paths that results in exactly one point (*i.e.*, containing only one of the scenes 'first missed serve and exchange' or 'exchange') are allowed. The local searches operate in a pipeline where remaining paths are further developed until the end of the video.

To this end, we use two queues $Q_{t,i}$ and $Q'_{t,i}$, each of size S, per state i and time instant t, where S is the maximum number of points allowed between two score labels. These queues hold the best paths obtained for each number of scored points, *i.e.* $Q_{t,i}(s)$ is the best local path ending at t in state i that contains s points. The first queue holds the paths of the current local search, while the second one stores results from the previous local search. The queues store the path likelihood $\delta_{t,s}(i)$, for each time instant t, state i and number of points scored s, as well as backtracking information. Let us denote by n_{ij} a 0/1 indicator if the state transition i to j gives a new point (for example $n_{45} = 1$ in Fig. 1). In the case of HMMs, the local search for the current label is defined as:

$$\delta_{t,s}(i) = \max_{j,\sigma,s=\sigma(Q)+n_{ji}} \delta_{t-1,\sigma}(j) a_{ji} b_i(O_t)$$

for the time steps $t \in [t_1, t_3]$, where $\sigma \in Q_{t,j} \cup Q'_{t,j}$ for $t \in [t_1, t_2]$ and $\sigma \in Q_{t,j}$ for $t \in [t_2, t_3]$, and $\sigma(Q) = \sigma$ iff $\sigma \in Q$ and 0 otherwise. Finally, a_{ji} and $b_i(O_t)$ denote the transition and observation probabilities respectively. After the local search is performed, we penalize all the paths in all queues $Q_{t,i}$ not consistent with the required number of scored points s^*, *i.e.* $\delta_{t,s}(i) = \delta_{t,s}(i) + P(s, s^*)$, where $P(s, s^*)$ is a large negative value if $s \neq s^*$ and zero otherwise. Finally, for time $t \in [t_2, t_3]$, we transfer the contents of $Q_{t,i}$ to $Q'_{t,i}$ and then proceed to the next local search in the pipeline. Inductively, the best complete path thus obtained will agree with all the score indications and is guaranteed to be the most likely given the score constraints.

C. Taking errors into account

Due to undisplayed score labels, the number of points actually scored between two labels is not always known exactly. For example, between two labels 'advantage', the number of points can be 2, 4, etc. In addition, automatic score label detection and recognition might introduce some errors, even though we believe that this task could be carried out reliably as score labels usually appear framed at fixed position and clearly distinct from the background. To tackle these two problems, we use, instead of $P(s, s^*)$, a smooth penalty function $P(s, l_1, l_2)$ that captures the certainty we have on the number of points scored between two labels. The penalty $P(s, l_1, l_2)$ that s points are scored between the two labels l_1 and l_2 is defined as

$$P(s, l_1, l_2) = A\Big(1 - \frac{N(s, l_1, l_2)}{\sum_s N(s, l_1, l_2)}\Big) \qquad (1)$$

where A is a large negative value and $N(s, l_1, l_2)$ is the number of times s points were scored between the two labels l_1 and l_2 on the training corpus. Therefore, the penalty depends on an estimation of the probability that s points are scored between two specific labels. When using automatically detected score labels, the counts $N(s, l_1, l_2)$ must be determined using the automatic detector in order to take detection errors into account in the penalty function. In this way, the penalties will tend to be more uniform between the various values of s when many errors occur and the algorithm will tend toward the standard Viterbi algorithm.

D. Score oriented search with a hierarchical topology

The score-oriented search can provide further benefits when used with a hierarchical topology. Indeed, the hierarchical topology ensures a solution consistent with the tennis rules, but not necessarily with the number of sets or games per set actually scored. The score oriented search can be easily extended to the hierarchical topology by adding two more variables in the queues Q and Q' to keep track of the sets and games traversed. According to the transitions of the hierarchical topology we update these two variables at each time instant. When information on the game evolution is provided by a score label (they usually appear at the end of each game), the inconsistent paths can safely be pruned. In this way we can obtain a solution that not only agrees on the number of points scored, but also on the actual game structure.

IV. Experimental Results

Experiments were carried out on a corpus of 6 tennis videos with a total duration of 15 hours. The first three videos were used as a training set to estimate the model parameters. The last three videos were used as the test set. All the video were manually annotated based on the automatic video track segmentation and therefore, errors of the hard cut and dissolve detection are not taken into account in this analysis. The estimation of the parameters for the ergodic HMMs and SMs is straightforward [1]. In the hierarchical approach, estimating the transition probabilities between the hidden states is not possible given the limited size of our training corpus. Hence, these probabilities were arbitrarily set to 1 in the experiments reported here. Performances are measured in terms of the average percentage C of shots assigned with the correct scene label as well as in terms of recall R and precision P rates on the scene boundaries. To make the comparison easier

TABLE I

RESULTS SUMMARY

	HMM				SM			
	C	P	R	F	C	P	R	F
	Standard Viterbi search							
Ergodic	80.2	84.7	79.7	66.4	81.8	84.1	79.4	66.8
Hierarchic	79.3	84.9	77.9	65.0	81.2	85.6	77.2	66.0
w. scores	80.8	85.7	80.4	67.6	82.0	84.2	79.7	67.1
	Score-oriented search							
Ergodic	82.2	83.4	82.4	68.3	86.0	84.9	83.4	71.8
Hierarchic	82.7	84.3	80.5	68.0	85.8	85.1	82.9	71.6
	With simulated detection errors							
90%	81.6	83.9	82.2	68.2	85.6	85.5	82.7	71.6
50%	81.1	84.4	80.7	67.3	84.1	87.0	80.2	70.1

than comparing all the above three quantities we devised a combined measurement, defined as F=3CPR/(C+P+R).

Results are summarized in table I for the HMM and SM approaches. The first three rows of results give the performance for the baseline systems, with an ergodic structure (row 1), with the hierarchical structure (row 2) and with score labels used as features in the ergodic structure (row 3). The comparison of the HMM and SM systems show a marginal improvement using segment models when the score labels are not used. Using the hierarchical topology, we notice a light degradation of the overall performance measure F, mostly due to a decrease of the recall rate (*i.e.* an increased number of scene boundaries detected). This degradation is most probably due to the manual setting of the transition probabilities to 1 in the hierarchical approach, while the scene transition probabilities were estimated on the training data for the ergodic model. Finally, the introduction of the score label indicator as a new feature yields a marginal improvement in the HMM case. This improvement is mainly due to the spotting of some hidden states where a score label is likely to appear. We observe more or less the same performance in the SM case where score features are integrated at the scene level rather than at the shot level. The score feature is distributed more uniformly between the four scenes and thus cannot contribute as much if used that way.

The next two rows of results detail the performance obtained with the proposed score-oriented Viterbi search. Clearly, in all the cases, performance improve significantly when constraints on the score are used. This improvement is mainly due to a higher recall rate, and therefore classification rate. The score-oriented search seems to perform a better segmentation than the standard Viterbi algorithm, with less false alarms in boundary detection. As a consequence, the classification rate is also improved. As previously, the overall performance with the hierarchical topology is still lower than with the ergodic topology, due to an decreased recall rate. However, the performance gap between hierarchical and ergodic models is clearly reduced thus suggesting that the hierarchical models benefit more from the score-oriented search than the ergodic ones. Finally, one interesting result is that SMs perform significantly better than HMMs with a increase of the combined performance measure F of 5.03% for the former and 1.89% for the latter. This may be explained by the fact that the positions themselves of the score labels provide some rough approximations of the scene boundaries, giving some extra valuable information for the Viterbi decoding in SMs.

We observed that the structure recovered with the score-oriented search agree almost completely with the score indications while the game structure is perfectly recovered. Most errors are due to confusions between the two scenes 'first missed serve and exchange' and 'exchange' which cannot be disambiguated by score information since both results in a score change.

Finally, the last two rows of Tab. I report results for the score-oriented search with ergodic models when errors on the score labels are artificially introduced. Plausible errors were simulated both on the training and test corpora. The counts used in the computation of the penalty (see Eq. (1)) were then reestimated on the training corpus with artificial label recognition errors and applied on the corresponding test corpus. This experiment was repeated with two different label recognition error rates, 10% and 50% . The results clearly demonstrate the robustness of the score-oriented search to inlaid label recognition errors, with only a marginal decrease of the overall performance, even when half of the labels are misrecognized. High label recognition error rates seems to induce more boundary detection thus resulting in a better precision rate and a lower recall rate, only marginally lowering the scene classification rate.

V. CONCLUSIONS

We proposed a score-oriented Viterbi search to efficiently integrate into the HMM and SM frameworks punctual prior semantic information. We experimented this method on tennis video structuring using inlaid score labels to guide the search algorithm. This approach resulted in a significant improvement of the segmentation and classification performance of both the HMM and SM systems, the latter benefiting more from knowledge of the score label. We also demonstrated the robustness of our approach to recognition errors. Future work includes the use of an OCR system for score label recognition and the integration of other sources of semantic information, such as the player position.

ACKNOWLEDGMENT

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

REFERENCES

[1] M. Delakis, G. Gravier, and P. Gros. Multimodal segmental-based modeling of tennis video broadcasts. In *Proceedings of IEEE international Conference on Multimedia and Expo*, 2005.

[2] E. Kijak, G. Gravier, L. Oisel, and P. Gros. Audiovisual integration for tennis broadcast structuring. *Multimedia Tools and Application*, 2005.

[3] 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.

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

[5] 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.

[6] J. Xi, X.-S. Hua, X.-R. Chen, L. Wenyin, and H.-J. Zhang. A video text detection and recognition system. In *Proceedings of ICME*, pages 1080–1083, 2001.