Hidden Markov Model Based Structuring of Cricket Video Sequences Using Motion and Color Features

M. H. Kolekar and S. Sengupta
Electronics and Electrical Communication Engineering Department,
Indian Institute of Technology, Kharagpur-721302, West Bengal, INDIA
{mhkolekar, ssg}@ece.iitkgp.ernet.in

Abstract

In this paper, we propose our techniques and results on automatic analysis of cricket video to facilitate highlight generation and content-based retrieval. We use Dynamic Programming based on Hidden Markov Model (HMM-DP) approach for structure analysis of cricket video sequences. We introduce a scaling factor α with the likelihood values to achieve classification more robust and accurate. We obtained 87.5 % average overall classification for motion and 75 % for color, which clearly indicate that motion is more effective likelihood function than color for 'Cricket' video sequences. We classified real 'Cricket' video sequence into four semantically meaningful classes and demonstrated the validity of our approach.

1. Introduction

Cricket is a very popular game in some nations like India, Sri Lanka, Pakistan, Australia, New Zealand, South Africa, England, Zimbabwe etc. One-day matches are highly action-packed and large amount of data are generated, making highlight/summary generation very tedious, when carried out manually. Automatic generation of highlights is highly demanding topic for cricket video analysis due to its tremendous commercial potentials. In this paper, we attempt to provide an automatic detection and classification of video shots, which can be further used for cricket video highlight generation or cricket video summarization.

We exploit the available domain knowledge about cricket video and demonstrate that it is possible to generate meaningful semantic annotations applicable to the domain. Our goal is to automate the generation of useful and high-level annotations like 'Ball Movement', 'Fielding', 'Pitch Action' (Balling and Batting), 'Wicket Keeping' (behind the wicket), to pertinent segments of a video. We have chosen these annotations because they form some of the most common play events in a cricket video. Annotating raw, unstructured cricket video with this high-level content can help professional players and coaches to retrieve video segments in a more meaningful manner from a digital library of cricket video. For example, a player who wants to improve his balling or batting would only be interested in retrieving and studying a variety of 'Pitch Action' video segments. If the above-mentioned high-level annotations are available for indexing a digital library of cricket video, then players and coach can get content-based access to relevant video segments. Further, the above-mentioned annotation is highly useful to generate cricket highlights and cricket video summarization.

Automatic video indexing requires new techniques, which support at least two tasks. First, a sequence of images needs to be temporally partitioned into a set of video shots, each of which consists of frames with similar image contents. Second, various image properties need to be extracted from individual video shot to build up semantic descriptions, from which we can construct indices. To date, variety of algorithms has been developed to fully or partially support the first task [5]. On the other hand, automatic generation of semantic descriptions from general video scene is still a challenging task for the current computer vision applications. We are carrying out both automatic video shot detection and classification simultaneously on cricket video using HMM-DP approach.

Despite numerous efforts in sports video analysis, it is hard to develop a generic approach to sports video analysis. Currently most work focus on specific games especially tennis [13], baseball [2], TV Soccer [4], football [1]. The cricket game structuring is altogether different from the above games. The current research has not yet paid any attention to cricket game may be because of the challenges in cricket video processing such as long duration of the game, high speed of actions, occlusion problems, higher complexity of rules, wide coverage area, etc.

Video-shot detection and classification is a fundamental step for efficient accessing, retrieving [7], [13], browsing, highlight generation [11], and summarization of large amount of video data and is one of the popular fields of video research in very recent times. Shot detection and classification is usually based on the likelihood values, computed through feature space, such as color histogram, pixel difference, statistical pixel difference, motion vectors, etc. These techniques are mostly based on heuristic

decisions and are not robust. Also, temporary plunge in likelihood values of the correct class may result in an incorrect class taking over and hence, wrong classifications would result.

Hidden Markov Model [12] based classifiers do not suffer from these limitations, since they exploit the temporal behaviors of the shots and make the classification and shot transition decisions based on accumulated likelihood values, rather than only the likelihood values of the frames. Dynamic Programming [10] is one of the popular approaches to HMM implementation. HMM based video classifications are more robust and find increasing use in video shot detection and classification [3], [6], [8], [9].

In this paper, we propose a modified Hidden Markov Model based Dynamic Programming algorithm for video shot detection and classification and determine the effect of introduction of scaling factor α to likelihood values.

The problem of video classification is generally delinked from the shot detection. Classification necessarily assumes that the shot detection is successfully solved. Any error in shot detection therefore results in a misclassification and in this context; we argue that the two problems should be solved in an integrated way.

It is seen that most of the cricket video sequences are highly structured in the sense that the number of classes are generally limited in number and there are repeated transitions, often back and forth between those classes. For cricket video sequences, it is possible to derive the structure in terms of transition probabilities from one class to another and train a HMM [12] for classification of a test video sequence. For structured videos of sports or movies, HMM has emerged as a very popular technique for classification and a number of works have been reported in the recent past [3], [6], [8], [9]. However, in these techniques, HMM based classifiers are applied over the detected shots. In our approach, shot detection is not done *a priori*. HMM is applied on a frame-by-frame basis and achieves classification through a process of backtracking using dynamic programming. The process of backtracking can be utilized for shot boundary detection as well and this motivated us to treat the problem of shot detection and classification in an integrated manner using HMM, as proposed herein.

Our solution differs from classical HMM-based methods in the following main points: (1) It performs video shot detection and classification simultaneously. (2) We introduced the scaling factor α to the contribution of current likelihood values in the accumulated value computations, which results in more robust shot boundary detection. (3) We suggested use of HMM-DP approach for structuring of cricket video sequence with motion as a likelihood function. (4) The classification contents are semantically meaningful to users.

The remainder of the paper is organized as follows. In section-II we introduce the problem of integrated shot detection and classification. Section-III presents an overview of the system and briefly describes the motion and color likelihood functions. Experimental results are presented in section-IV and section-V concludes the paper with directions for future works.

2. An Integrated Approach to Video-Shot Detection and Classification

This approach starts with the computation of the likelihood that a video frame in a given sequence belongs to a shot of particular scene class and an optimum scene transition path is constructed for the entire input video sequence to determine the highest accumulative likelihood. Our approach can be summarized as follows:

Step-1: ***Likelihood computations:*** Given a sequence of frames *(t =1,2,……,T),* each of which can be classified into one of *N* classes (*k=1,2,……,N),* we first compute the likelihood $l_t(k)$ that the frame–*t* belongs to the class-*k*, based on the similarity $S_t(k)$ of the features (such as color, motion etc.) of frame-*t* with those of class-*k* and are given by

$$l_t(k) = \frac{S_t(k)}{\sum_{k=1}^{N} S_t(k)} \tag{1}$$

Step-2: ***Accumulated Likelihood Initialization:***
Corresponding to the starting frame (t =1), the accumulated likelihood $L_t(k)$ for every class is initialized to the current likelihood for the corresponding class, times a scaling factor α. Also, the backtracking indices $A_t(k)$ are initialized to zero, as given below:

$$L_1(k) = \alpha\, l_1(k) \quad \text{and } A_1(k) = 0 \text{ for } k = 1, 2, 3, \ldots, N \tag{2}$$

Step-3: ***Accumulated Likelihood recursions****:* For all subsequent frames (*t=2,3…T*), the accumulated likelihood of every class is computed through a dynamic programming based optimum path search. The backtracking indices of the classes have to point to the class that would contribute to the maximum accumulated likelihood. These are given by:

$$L_t(k) = \max_{1 \le i \le N} \{L_{t-1}(i) + C(i,k)\} + \alpha\, l_t(k) \tag{3}$$

$$A_t(k) = \arg\max_{1 \le i \le N} \{L_{t-1}(i) + C(i,k)\}$$

for all $k = 1, 2, 3, \ldots, N$ (4)

In the above equations, *C (i, k)* indicates the transitional probability from class-*i* to class-*k*, determined through training sequences of known class. It is obvious from equation (3) that higher value of scaling factor α contributes to predominance of current likelihood over the accumulated ones.

Step-4: ***Frame-by-frame classification****:* Following the accumulated likelihood computations for all the classes in all the frames of the sequence, the frames are classified individually, starting with the last frame of the sequence through a process of backtracking, as given below

$$C_T^* = \arg\max_{1 \le i \le N} L_T(i) \quad (5)$$

$$C_t^* = A_{t+1}\left(C_{t+1}^*\right), \quad t = T-1, T-2, \ldots\ldots\ldots\ldots, 1. \quad (6)$$

Step-5: ***Shot transition detection****:* A sequence of frames *p, p+1,,…,p+s* belongs to a shot of class-*i* if

$$C_p^* = C_{p+1}^* = \cdots\cdots = C_{p+s}^* = i \quad (7)$$

and the frames *(p+s)* to *(p+s+1)* correspond to a shot transition if

$$C_{p+s+1}^* \neq i$$

3. System Overview

In this section, we briefly describe the system of video shot detection and classification. We have used 11 different cricket video sequences to train HMM. For unknown cricket video sequences, the likelihood values for each frame are computed. HMM-DP algorithm is applied to find the accumulated likelihood computations for all the classes and for all the frames of the sequence and the frames are classified individually, starting with the last frame of the sequence through a process of backtracking. The complete video sequence is classified into four classes i.e. 'Ball Movement' (class 0), 'Pitch Action' (class1), 'Fielding' (class 2), 'Wicket Keeping' (class 3).

The efficiency of HMM based dynamic programming shot detection and classification is highly dependent on the selection of likelihood function. The most popular likelihood functions are based on color histogram, motion estimation, texture, transforms, etc. We experimented with color and motion as likelihood function.

3.1. Motion as a Likelihood Function

Here, we briefly describe the method of estimating the motion. First, the frame *t* is divided into *L* blocks. Then a block matching procedure is applied to find best match for each block $b_i(t)$ in frame t a corresponding block $b_{i,m}(t-1)$ in frame $t-1$, such that it is most similar to the block $b_i(t)$ according to a chosen criterion (D), that is

$$D_{t,t-1}(i) = D\left(b_i(t), b_{i,m}(t-1)\right) \quad (8)$$

$$MAD[i] = \min_{j=1,2,\ldots\ldots L} D\left(b_i(t), b_{i,j}(t-1)\right) \quad (9)$$

Note that in the above equation *D* is computed as a net absolute difference between the blocks. The algorithm used above is basically a full-search, minimum absolute difference algorithm *(MAD).* For all the blocks of the frame, these *MAD* values are computed and $MAD_t[i]$ is generated. Similarly, in HMM training process, we have computed *MAD* values for different videos of known classes to generate vector $MAD_k[i]$.

$$m(t,k) = \sum_{i=1}^{L} \left| MAD_t[i] - MAD_k[i] \right| \quad (10)$$

$$m'(t,k) = \frac{m(t,k)}{\sum_{k=1}^{N} m(t,k)} \quad (11)$$

where, N = total number of classes.

The likelihood function for frame *t* for each class is computed as

$$l(t,k) = \frac{1 - m'(t,k)}{\sum_{k=1}^{N} 1 - m'(t,k)} \quad (12)$$

3.2. Color Histogram as a Likelihood Function

In color histogram based approach, we create three-dimensional histograms in RGB color space with 5 bins for R, G, and B respectively, resulting in a total of 125 bins. To incorporate spatial information of the color distribution, we divide each frame into 2x2 blocks, and create 3D-histogram for each of the blocks. These four histograms are then concatenated together to form a *L*-dimensional-feature vector $histo_t[i]$ for the frame *t.* Similarly, in HMM training process; we have computed values for different videos of known classes to generate $histo_k[i]$.

$$color(t,k) = \sum_{i=1}^{L} \left| histo_t[i] - histo_k[i] \right| \quad (13)$$

This color function is assigned to likelihood function for frame *t* for each class *k* after normalization by using equations similar to equation (11) and (12).

3.3. State Transition Probabilities

The state transition probability matrix expresses the probability of moving from one hidden state to another. There are at most N^2 transitions among the hidden states (with N as the number of states) since it is possible for any one state to follow another or itself. Generally, the states are interconnected to each other in such a way that any state can be reached from any state (i.e. an ergodic model) at regularly spaced discrete times. In our case we are using sequences of four classes and thus each state corresponds to a class. As shown in Fig. 1, we generated the state transition probability matrix after training through 11 cricket video sequences.

4. Experimental Results and Discussion

We evaluate our HMM-DP based classification method by classifying four types of cricket events. They are 'Ball Movement', 'Fielding', 'Pitch Action' (Balling and Bating), and 'Wicket Keeping' (behind the wicket). Fig. 2 shows the detected shot boundaries for 'Cricket' video sequence. Classification results are shown in Table 1. The class discrimination observed is 100% for $\alpha = 6$ for motion likelihood function while for color likelihood function require $\alpha = 11$. Hence, it is clear that the motion likelihood function is more effective than color. It is also clear that the classification results get improved with increase in the value of α. In our experiments, we use two metrics to gauge the performance of our model. Their definitions are given as follows:

Recall: For a video class of interest, *Recall* is the ratio of correct classification for this class over all the classification correctly belonging to this class.

From Table 1, *Recall* for class 0 for motion for $\alpha = 2$ is computed as follows:

$$\text{Re}\,call\,(0) = \frac{|\{correct\} \cap \{classified\}|}{|\{correct\}|} = \frac{|\{2\} \cap \{3\}|}{|\{2\}|} = \frac{2}{2} = 100\%$$

Precision: For a video class of interest, *Precision* is the ratio of correct classification for test sequence made into this class over all the classifications made into this class.

From Table 1, *Precision* for class 0 for motion for $\alpha = 2$ is computed as follows:

$$\text{Pr}\,ecision\,(0) = \frac{|\{correct\} \cap \{classified\}|}{|\{classified\}|} = \frac{|\{2\} \cap \{3\}|}{|\{3\}|} = 66.66\,\%$$

In the above definitions, $\{correct\}$ denotes the set of shots belonging to the class 0; $\{classified\}$ denotes the set of shots classified into the class 0 by our model.

The overall HMM-DP classification performance for four classes is reported in Table 2 for $\alpha = 2$. Overall results yield an average 87.5 % for motion likelihood function and 75 % for color likelihood function. Hence, it is clear that for cricket video sequence motion is more effective likelihood function than color. Since class to class transition is more, 100% overall classification is obtained for 'Ball Movement' to 'Ball Movement', 'Fielding' to 'Fielding' and 'Pitch Action' to 'Pitch Action' for motion likelihood function, whereas for color likelihood function, we obtained 100 % overall classification for 'Fielding' to 'Fielding' and 'Pitch Action' to 'Pitch Action'.

Table 3 shows *Recall* values for the four classes that occur diagonally in Table 2. For motion, we obtained 100 % classification results on three classes i.e. 'Ball Movement', 'Fielding', 'Pitch Action' and somewhat less successful results i.e. 50 % on 'Wicket Keeping' class. For color, we obtained less successful results up to 50 % on two classes; 'Ball Movement' and 'Wicket Keeping'. Results are poor for color likelihood function because color features are not discriminable since most of frames in cricket videos contain green ground. In contrast, motion is right candidate and our classes are action dependent, hence results are good for motion.

Table 4 shows *Precision* value for each of the four classes. In our particular framework, the precision measure indicates the interaction between classes, since it encapsulates the number of right and wrongly classified shots. In this table, we note that for motion, 'Wicket Keeping' class has relatively low precision value up to 50 %. Obviously, these are confused with 'Ball Movement'. In color as likelihood, we observed poor precision for 'Pitch Action' and 'Wicket-keeping' up to 33.33 % and 50 % respectively. Here, 'Pitch Action' class is confused with 'Ball Movement' and 'Wicket Keeping'. Of the 'Ball Movement' shots, 50 % are misclassified as a 'Pitch Action' and of the 'Wicket Keeping' shots, 50 % are misclassified as a 'Pitch Action'. Hence, for 'Pitch Action' class although recall is 100 %, precision is come down to 33.33 %.

As shown in Table 1, our results can be improved by increasing the value of α. Also there is a room to improve the results by taking more number of videos for training of HMM.

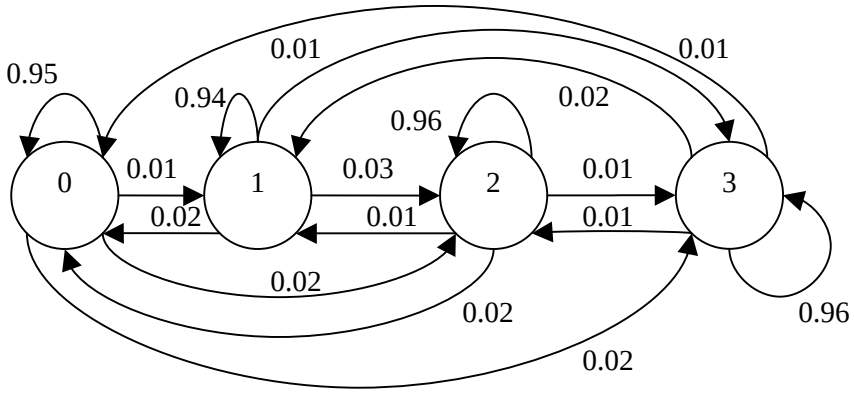

Figure 1: State Transition Diagram for 'Cricket Video' Sequence

5. Conclusion

In this paper, we have presented HMM based model for cricket video structure parsing. We have proposed an integrated approach to video shot detection and classification using HMM. A scaling factor is introduced in the dynamic programming recursions to improve robustness in shot detection. The selection of likelihood function is sequence dependant. For cricket sequence, for $\alpha = 2$, we observed 87.5% overall classification for motion likelihood and 75 % for color likelihood. Since our classes are action based and motion is frequently changing from shot to shot, we obtained better classification for motion likelihood function than color. It is possible to extend this work further by introducing more interesting and semantically relevant classes.

References

[1] N. Ancona, G. Cicirelli, A. Branca, and A. Distante, "Goal detection in football by using support vector machine for classification", *in Proc of IEEE Int. Joint Conf. on Neural Network*, vol.1, pp. 611-616, 2001

[2] P.Chang, M. Han, and Y. Gong, "Extract Highlights from Baseball Game Video with Hidden Markov Models," *in Proc. of IEEE Int. Conf. on Image Processing*, vol.1, pp. 609-612, 2002

[3] S. Eickeler and S. Müller, "Content-Based Video Indexing of TV Broadcast News using Hidden Markov Models," *in Proc. of IEEE ICASSP*, vol. 6, pp. 2997-3000, 1999.

[4] Y. Gong; L. T. Sin; C. H. Chuan; H. Zhang; M. Sakauchi, "Automatic parsing of TV soccer programs," *in Proc. of Int. Conf. on Multimedia Computing and Systems*, pp.167-174, 1995

[5] A. Hanjalic, "Shot-Boundary Detection: Unraveled and Resolved," *IEEE Transactions on Circuits and Systems for Video Technology* 12(2): 90-105, 2002.

[6] J. Huang, Z. Liu, and Y. Wang, "Joint Video Scene Segmentation and Classification based on Hidden Markov Model," *in Proc. of IEEE Int. Conf. on Multimedia and Expo*, pp. 1551-1554, 2000

[7] A. Jaimes and S.F. Chang, "Model-based Classification of Visual Information for Content-based Retrieval," *Storage and Retrieval for Image and Video database VIII, IS & T/SPIE, San Jose*, 1999

[8] E. Kijak, L. Oisel and P.Gros, "Hierarchical structure analysis of sports videos using HMMs", *in Proc. of IEEE Int. Conf. on Image Processing*, vol.3, pp. 1025-1028, 2003

[9] T. Liu, J. R. Kender, "A Hidden Markov Model Approach to the Structure of Documentaries," *in Proc. of IEEE Workshop on Content-based Access of Image and Video Libraries*, pp.111-115, 2000.

[10] H. Ney, S. Orthmanns, "Dynamic Programming Search for Continuous Speech Recognition," IEEE *Signal Processing Magazine*, pp. 64-83, 1999.

[11] H. Pan, P. Van Beek, and M. I. Sezan, "Detection of Slow-Motion Replay Segments in Sports Video for Highlights Generation," *in Proc. IEEE ICASSP*, vol. 3, pp. 1649–1652, 2001.

[12] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *in Proc. of IEEE*, vol. 77, no. 2, pp. 257-286, 1989.

[13] G. Sudhir, J.C.M. Lee, A.K. Jain, "Automatic Classification of Tennis Video for high-level Content based Retrieval," *in Proc. IEEE Workshop on Content-Based Access of Image and Video Database*, pp. 81-90, 1998.

Figure 2: Shot boundary of 'Cricket' sequence at frame numbers 59, 79, 98, 125, 145,180

Table 1: Classification Results for 'Cricket' sequence for various values of α

Shot Number (Beginning-End Frame) length	Motion as a Likelihood Function			Color as a Likelihood Function		
	$\alpha = 1$	$\alpha = 2$ *to* 5	$\alpha = 6$	$\alpha = 1$	$\alpha = 2$ *to* 10	$\alpha = 11$
	Class Observed/ Actual	Class Observed/ Actual	Class Observed/ Actual	Class Observed/ Actual	Class Observed/ Actual	Class Observed/ Actual
0 (0-59) 60	0/0	0/0	0/0	0/0	0/0	0/0
1 (60-79) 20	3/3	3/3	3/3	3/3	3/3	3/3
2(80-98) 19	**2/1**	1/1	1/1	**2/1**	1/1	1/1
3 (99-125) 27	2/2	2/2	2/2	2/2	2/2	2/2
4 (126-145) 20	**0/3**	**0/3**	3/3	**2/3**	**2/3**	3/3
5 (146-180) 35	0/0	0/0	0/0	**2/0**	**2/0**	0/0

Table 2: Overall Classification Performance (in %) for $\alpha = 2$

Result		Motion as a Likelihood Function				Color as a Likelihood Function			
		Observed Class				Observed Class			
		Ball Movement	Fielding	Pitch Action	Wicket Keeping	Ball Movement	Fielding	Pitch Action	Wicket Keeping
Actual Class	Ball Movement	100	0	0	0	50	0	50	0
	Fielding	0	100	0	0	0	100	0	0
	Pitch Action	0	0	100	0	0	0	100	0
	Wicket Keeping	50	0	0	50	0	0	50	50

Table 3: Recall Performance (in %) for $\alpha = 2$

Motion as a Likelihood Function				Color as a Likelihood Function			
Ball Movement (0)	Fielding (1)	Pitch Action (2)	Wicket Keeping (3)	Ball Movement (0)	Fielding (1)	Pitch Action (2)	Wicket Keeping (3)
100	100	100	50	50	100	100	50

Table 4: Precision Performance (in %) for $\alpha = 2$

Motion as a Likelihood Function				Color as a Likelihood Function			
Ball Movement (0)	Fielding (1)	Pitch Action (2)	Wicket Keeping (3)	Ball Movement (0)	Fielding (1)	Pitch Action (2)	Wicket Keeping (3)
66.66	100	100	100	100	100	33.33	50