SEMANTICS-BASED VIDEO INDEXING USING A STOCHASTIC MODELING APPROACH

Yong Wei, Suchendra M. Bhandarkar and *Kang Li*

Department of Computer Science, The University of Georgia, Athens, Georgia 30602-7404, USA
E-mails: {yong, suchi, kangli}@cs.uga.edu

ABSTRACT

Semantic video indexing is the first step towards automatic video retrieval and personalization. We propose a data-driven stochastic modeling approach to perform both video segmentation and video indexing in a single pass. Compared with the existing Hidden Markov Model (HMM)-based video segmentation and indexing techniques, the advantages of the proposed approach are as follows: **(1)** the probabilistic grammar defining the video program is generated entirely from the training data allowing the proposed approach to handle various kinds of videos without having to *manually* redefine the program model; **(2)** the proposed use of the Tamura features improves the accuracy of temporal segmentation and indexing; **(3)** the need to use an HMM to model the video edit effects is obviated thus simplifying the processing and collection of training data and ensuring that all video segments in the database are labeled with concepts that have clear semantic meanings in order to facilitate semantics-based video retrieval. Experimental results on broadcast news video are presented.

Index Terms— Video segmentation, Video semantic indexing, Hidden Markov Models

1. INTRODUCTION

Semantic video indexing is the first step towards automatic video browsing, retrieval and personalization. Semantic video indexing enables user to access videos according to their interests and preferences regarding video content.

Definition 1: *Semantic video indexing* is the process of attaching concept terms to segments of a video.

Semantic video indexing consists of two sub-processes, temporal segmentation of the video stream and semantic labeling of the resulting video segments. These two sub-processes are usually performed as two separate steps. In IBM's VideoAnnEx video annotation system [1], the video stream is first segmented into shots. An annotator then manually associates shots with terms selected from a predefined lexicon. For large amounts of video data, the manual annotation process involves intense human interaction and is extremely time consuming. Automatic temporal video segmentation methods usually involve computing pixel-level and/or histogram-based difference measures for each pair of successive frames in the video stream and then using shot boundary detection techniques to locate the positions of shot boundaries [2],[3]. More advanced temporal segmentation techniques use sophisticated image features such as edges [4], focus of expansion points [5] and image motion [6]. However, just as a phoneme can appear in many different words, visually similar video shots can appear in different video segments with different semantic meanings. Thus video shot segmentation by itself cannot support content-based video retrieval at a semantic level.

Definition 2: A *video semantic unit* is a video segment unit that can be associated with clear semantic meanings.

Instead of detecting video shots, it's more useful to recognize semantic units in the video stream. A semantic unit consists of a concatenation of semantically and temporally related video shots. In well organized videos, such as TV news broadcasts and sports programs, videos can be viewed as sequences of semantic units that are concatenated based on a video program syntax.

In continuous speech recognition, the continuous speech resulting from a spoken sentence is modeled at both the acoustic-phonetic (sub-word) level and the language level. In most modern speech recognition systems, these sub-word units are modeled by Hidden Markov Models (HMMs). In recent years, various applications of HMMs to video segmentation and annotation have been studied. Huang et al. [7] use both audio and visual features in an HMM-based scheme to perform video scene recognition. Li et al. [8] propose an HMM framework to detect *play* events in sports videos. Eickeler et al. [9] use an HMM-based predefined program model to index news programs. In the aforementioned works, however, the system performance could be compromised due to audio-visual mismatch [7] and inaccurate domain-dependent knowledge about the video program structure [8],[9].

In this paper, we propose a data-driven stochastic modeling approach to perform both video segmentation and indexing in a single pass. Inspired by the success of modern continuous speech recognition [10], a video stream is modeled at both the *semantic unit level* and the *program level*. For each semantic unit, an HMM is generated to model the stochastic behavior of the sequence of image feature emissions [9]. At the program level, a probabilistic grammar is generated by training on video data using maximum likelihood estimation. The grammar thus generated regulates the transitions amongst the semantic units. A concatenation of the HMMs based on the above probabilistic grammar constitutes the final search space for the video segmentation and indexing algorithm. To segment and classify semantic units in a video stream, the Viterbi algorithm (based on dynamic programming) is used to determine the optimal path, through the concatenation of the HMMs, which maximizes the likelihood of the observed sequence of image features emissions. The advantages of the proposed approach are as follows.

- The probabilistic grammar defining the video program is generated entirely from the training data. In contrast to existing HMM-based video segmentation and indexing techniques [9], no domain-dependent knowledge about the structure of video programs is used. This allows the proposed approach to handle a wide variety of video types without having to *manually* redefine the program model.

- The use of the Tamura features improves the accuracy of temporal segmentation and indexing of the video.
- The proposed data-driven approach does not need to use an HMM to model the video edit effects. Semantic unit level HMMs are used to model only video segments with clear semantic meanings. This not only simplifies the processing and collection of training data, but also ensures that all video segments in the database are labeled with concepts with clear semantic meanings thus facilitating semantics-based retrieval.

The rest of the paper is organized as follows. Section 2 describes the image features used in the construction of the video semantic unit level HMMs. In Section 3 we describe the construction of HMMs for the semantic units, and the organization (via concatenation) of the individual HMMs based on a video program model. The data-driven approach for learning the video program model is detailed. Section 4 explains the performance measures used and the experimental results. Section 5 concludes our work with an outline for future research.

2. IMAGE FEATURES FOR SEMANTIC UNIT HMMs

Applications of HMMs to video segmentation and indexing have been reported in recent literature [7],[8],[9]. A successful HMM-based video segmentation and indexing scheme depends greatly on the selection of a suitable multi-dimensional feature vector to represent each image frame in the video stream. In most existing HMM-based video segmentation and indexing techniques, the dynamic characteristics of the image frames comprising the video stream, are captured using differences of successive image frames at both, the pixel level and the histogram level. Various motion-based measures describing the movement of the objects in the image frames are used, such as the location of the image centroid and intensity of motion. Measures of illumination changes at both the pixel level and the histogram level are also included in the multi-dimensional feature vector. A detailed description of these features is provided in [9].

In the proposed approach, in addition to the aforementioned category of image features, Tamura features [11] are used to capture the textural characteristics of the image frames at the level of human perception. Tamura contrast, Tamura coarseness and Tamura directionality have been used successfully in content-based image retrieval [12]. In our work, these features improve significantly the accuracy of temporal segmentation and indexing.

3. HMMS FOR VIDEO SEGMENTATION AND INDEXING

3.1. HMMs for Video Semantic Units

In our semantics-based video segmentation and indexing system, we define a semantic unit for each of the following six semantic concepts, i.e., *News Anchor*, *News*, *Sports News*, *Commercial*, *Weather Forecast* and *Program Header*. Representative images for each of these semantic concepts are shown in Fig. 1.

An HMM is established for each individual semantic unit. The HMM parameters for each semantic unit are optimally learned using feature vector sequences obtained from the training video sets. In our approach, the HMMs for individual semantic units are trained separately using the training feature vector sequences described in Section 2. This allows great flexibility in being able to accommodate various types of video data. When new video data for a semantic unit are presented, we only need to retrain the corresponding HMM for the relevant semantic unit without having to retrain any other HMM in the overall system. We choose a universal left-to-right HMM topology with continuous observations of the emissions. The number of Gaussian mixture components in these HMMs is chosen to be three in our implementation. The reason for these choices is that in actual video data, little is known about the underlying physical processes which generate the observable visual features in the video stream. Using the above choices as default makes it easy to build semantic unit-level HMMs for unknown data [9].

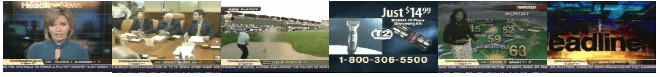

Fig. 1 Representative Image Frames of Semantic Units From Left to Right: *News Anchor*, *News*, *Sports News*, *Commercial*, *Weather Forecast* and *Program Header*

In real video programs, there are many different kinds of transitional scenes used to connect the major scenes. Examples of transitional scenes include cuts, fades, zooms and pans. A large number of transitional states corresponding to these transitional scenes are typically used to connect the simple HMMs corresponding to the major scenes [9]. However, these transitional scenes are highly domain dependent and difficult to classify in terms of their semantic content. In our approach, semantic concepts are not attached to these transitional scenes; only semantic concepts with clear definitions are used. This not only improves the robustness and generality of our system in terms of its ability to handle a large variety of videos, but also simplifies the training data collection process.

3.2. Concatenation of HMMs for Video Program Model

The proposed single pass video segmentation and indexing procedure is formulated in terms of the following Bayesian decision rule: Given a sequence of image feature vectors, determine a video semantic unit sequence such that

$$\max(\Pr(U_1...U_N \mid f_1...f_T)) \sim \max(\Pr(U_1...U_N)\bullet\Pr(f_1...f_T \mid U_1...U_N)) \quad (3.1)$$

where $f_1...f_T$ are image feature vectors extracted from the image frames in the video stream to be segmented and indexed and $\Pr(U_1...U_N)$ is the video program model.

The video program model regulates the transition probability from a predecessor semantic unit to a successor semantic unit. The individual HMM for each semantic unit models the stochastic behavior of the sequence of image features within the scope of the semantic unit. At the boundary of the semantic unit, the transition probability follows the video program model given by $\Pr(U_1...U_N)$. The search space for the single-pass video segmentation and indexing procedure is the concatenation of the individual HMMs for the semantic units. A Viterbi algorithm is used to determine the optimal path in the HMM concatenation. Fig. 3 depicts an example of single-pass video segmentation and indexing with semantic concepts *A*, *B* and *C*. The y-axis represents the hidden states within the HMM for an individual semantic concept. The bold curves in Fig. 3 depict the state transitions within the video semantic units *A*, *B* and *C*. The video stream in the example is segmented into a semantic unit sequence *BACBA*. Bold curves within a semantic unit are monotonically non-decreasing since the HMMs corresponding to the individual semantic units have a left-to-right topology, i.e., no backward state transitions are permitted.

In this work, a pure data-driven approach is taken to estimate the video program model directly from the training data using sequential maximum likelihood estimation, i.e., no domain-dependent knowledge about the structure of video programs is used. Most researchers typically use domain-specific knowledge about the video program in order to determine the video program model [7],[8],[9]. This knowledge-driven approach becomes untenable as the size of the semantic unit vocabulary and the complexity of video program increase. The inaccuracy in the estimation of the video program model directly affects the segmentation and indexing results. The video program model in this work is represented as a 2-gram model [13] for the purpose of efficient training. The training data for learning the video program model is assumed to be manually pre-labeled.

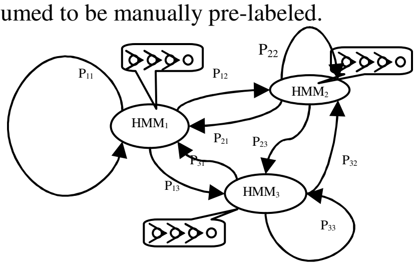

Fig. 2 Concatenation of HMMs of 3 Video Semantic Units

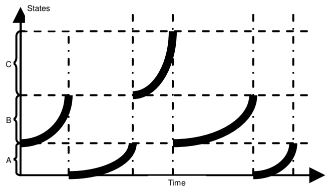

Fig. 3 An Example of Video Segmentation and Indexing with 3 Semantic Concepts *A*, *B* and *C*

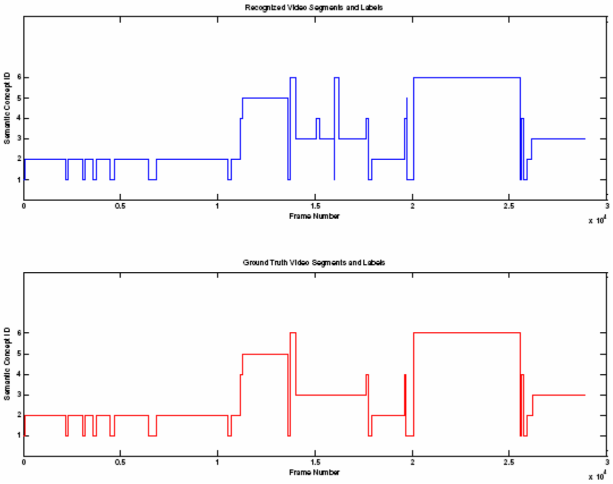

Fig. 4 Frame Number vs. Recognized/Ground Truth Video Segments and Labels

4. EXPERIMENT RESULTS

Two hours of TV broadcast video streams were recorded and digitized at a frame resolution of 180×120 pixels and frame rate of 30 frames per second. We chose the category of TV broadcast videos since they contain a large number of fairly diverse video segments which do not show a bias for or against the proposed video segmentation and indexing approach.

Sixteen minutes of the video was reserved for testing. For generation of training data, the remainder of the video data were manually segmented into the six semantic categories, i.e. *News Anchor*, *News*, *Commercial*, *Program Header*, *Weather Forecast* and *Sports News* and denoted by semantic concepts 1 through 6 respectively. A multi-dimensional feature vector was extracted for each image frame in the training video. For each of the six semantic units, a left-to-right HMM with continuous emission of observations was trained using feature vector sequences derived from the training video. To estimate the 2-gram video program model, the training video was manually labeled with labels selected from the aforementioned six semantic concepts.

The performance measurements for the above single-pass video segmentation and indexing scheme comprised of two aspects, boundary detection performance and video segment classification performance. To measure the performance of video segment boundary detection, parameters such as insertion rate, deletion rate and boundary detection accuracy [14] were used. These parameters are defined as follows. The insertion rate denotes the fraction of unassigned boundaries in the detected boundaries. The deletion rate denotes the fraction of missed boundaries in the ground truth sequence boundaries. The boundary detection accuracy measures the average shift (in terms of number of frames) between the detected boundary and actual boundary locations.

$$R_{insertion} = \frac{boundaries_{inserted}}{boundaries_{detected}} \qquad (4.1)$$

$$R_{deletion} = \frac{boundaries_{missed}}{boundaries_{actual}} \qquad (4.2)$$

$$Accuracy_B = \frac{\sum \Delta frames}{boundaries_{actual}} \qquad (4.3)$$

$\sum \Delta frames$ is the amount of shift (measured in terms of number of frames) between the detected boundary location and the actual boundary location.

To measure the video segment classification accuracy, the correct video segmentation classification rate is defined as

$$Accuracy_C = \frac{S_{correct}}{S_{correct} + S_{false}} \qquad (4.4)$$

where $S_{correct}$ is the number of correctly indexed segments and S_{false} is the number of incorrectly indexed segments. In our experiments, some incorrectly indexed segments were observed to be very short. These short segments contained a very small fraction of the total number of image frames in the video stream. Thus, the measure $Accuracy_c$ by itself does not reflect the classification performance because it treats these very short and incorrectly classified segments on par with the relatively long and incorrectly classified segments. Hence in order to measure the number of image frames that are correctly classified, we used a frame-based measure to determine the fraction of correctly classified image frames in the total number of frames as follows:

$$Accuracy_F = \frac{frames_{correct}}{frames_{correct} + frames_{false}} \qquad (4.5)$$

where $frames_{correct}$ and $frames_{false}$ are the numbers of correctly classified and incorrectly classified image frames respectively. The algorithm provided by the author of [14] was used to compute performance measures in equations (4.1)-(4.4).

In Fig. 4, the recognized semantic label sequence and the ground truth semantic label sequence are plotted against the frame number for the entire test video segment. The single-pass video segmentation and indexing algorithm is observed to detect most of the segment boundaries and label them correctly, except for some portions of the *commercial* segment which are incorrectly classified. In Tables 1 and 2, the numerical measures of performance for the single-pass video segmentation and indexing algorithm are tabulated. Fig. 4 shows that most of the inserted boundary detection and false segment classification occurs during the *commercial* segment (semantic concept ID=3). This is because of the complex nature of the content of TV commercials. In TV commercial programs, there could be large video segments that are visually similar to those found in the other semantic units such as *News Anchor* and *Sports* and hence classified incorrectly.

5. CONCLUSIONS

Well structured video programs, such as TV news broadcasts and sports videos can be modeled at both, the program level and the video semantic unit level, analogous to the language level and acoustic word level in the case of speech recognition.

For each video semantic unit, an HMM is established to model the image feature vector emission process within the scope of the semantic unit. In our work, only semantic concepts with clear meanings are used. Transitional scenes are not modeled using HMMs. The 2-gram video program model defines the transition probabilities amongst the video semantic units. With increasing complexity of real video programs, a domain knowledge-dependent definition of a video program model becomes untenable. In our approach, the data-driven maximum likelihood estimation of the 2-gram program model from training data yields very good results. The use of the Tamura features improves the accuracy of temporal segmentation and indexing. The individual HMMs for the semantic units are concatenated based on the video program model. The data-driven single-pass video segmentation and indexing algorithm determines the optimal path through the HMM concatenation. In doing so, the video stream is segmented and indexed in a single pass. Experimental results show very good accuracy in terms of both boundary detection and segment classification. The proposed scheme is scalable and extensible since the program model can be altered by addition, deletion and modification (via retraining) of the HMMs corresponding to the relevant semantic concepts without having to retrain or alter the HMMs corresponding to the other semantic concepts.

Future research would apply the single-pass video segmentation and indexing algorithm to a wide collection of video data to test its robustness and extensibility. The effectiveness of the *n*-gram language model for $n > 2$ needs to be further explored in the context of modeling of video programs.

6. REFERENCES

[1] B.L. Tseng, C.Y. Lin and J.R. Smith, "Using MPEG-7 and MPEG-21 for Personalizing Video", *IEEE Multimedia*, Vol. 11(1), pp. 42-53 2004.

[2] M.M. Yeung and B. Liu, "Efficient Matching and Clustering of Video Shots", *Proc. ICIP95*, Vol. 1, pp. 338-341, Sep. 1995.

[3] A.M. Ferman and A.M. Tekalp, "Efficient Filtering and Clustering Methods for Temporal Video Segmentation and Visual Summarization", *Jour. Visual Comm. Image Rep.*, 9(4), pp. 336-351, 1998.

[4] R. Zabih, J. Miller and K. Mai, "A Feature-based Algorithm for Detecting and Classifying Production Effects", *Multimedia Systems*, 7(2), pp. 119–128, 1999.

[5] M. Ardebilian, X. Tu and L. Chen, "Robust 3D Clue-based Video Segmentation for Video Indexing", *Jour. Visual. Comm. Image Rep.*, 11(1), pp. 58–79, 2000.

[6] S.V. Porter, M. Mirmehdi and B.T. Thomas, 'Video Cut Detection using Frequency Domain Correlation", *Proc. ICPR*, Barcelona, Spain, pp. 413–416, Sep. 2000.

[7] J. Huang, Z. Liu and Y. Wang, "Joint Scene Classification and Segmentation Based on Hidden Markov Model", *IEEE Trans. Multimedia*, pp. 538-550, June 2005.

[8] B. Li and M.I. Sezan, "Event Detection and Summarization in Sports Video*", Proc. CBIVL*, No. 8, pp. 132–138, 2001.

[9] S. Eickeler and S. Müller, "Content-based Video Indexing of TV Broadcast News using Hidden Markov Models", *Proc. ICASSP*, pp. 2997-3000, March 1999.

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

[11] P. Howarth and S. Ruger, "Evaluation of Texture Features for Content-Based Image Retrieval", *CIVR* 1004, LNCS 3115, pp. 326-334, 2004.

[12] T. Deselaers, D. Keysers and H. Ney, "FIRE - Flexible Image Retrieval Engine: ImageCLEF 2004 Evaluation", *Proc. CLEF 2004 Workshop*, Bath, UK, pp. 535-544, September 2004.

[13] P.F. Brown, V.J. Pietra, P.V. deSouza, J.C. Lai, R.L. Mercer, "Class-Based n-gram Models of Natural Language", *Computational Linguistics*, 18(4), pp. 467-479, 1992.

[14] S. Eickeler and G. Rigoll, "A novel error measure for the evaluation of video indexing systems", *Proc. OCASSP*, Istanbul, Turkey, 2000.

Table 1. Performance Measures for Video Segmentation

Actual Boundaries	Detected Boundaries	Inserted Boundaries	Deleted Boundaries	Insertion Rate (%)	Deletion Rate (%)	$Accuracy_B$ (Frame)
29	36	7	0	7/36=19.4	0/29=0	55/29=1.9

Table 2. Performance Measures for Video Segment Classification

Correctly Classified Segments	Incorrectly Classified Segments	$Accuracy_C$ (%)	Total Number of Frames	Correctly Classified Frames	$Accuracy_F$ (%)
32	5	86.5	28898	514	28384/28898=98