Optimal Shot Detection and Recognition using Shiryaev-Roberts Statistics

Ranjith Ram[*] , Anup Shetty and Subhasis Chaudhuri
Vision and Image Processing Lab
Department of Electrical Engineering
Indian Institute of Technology Bombay
Powai, India
{ranjiram, anupshetty, sc} @ ee.iitb.ac.in

ABSTRACT

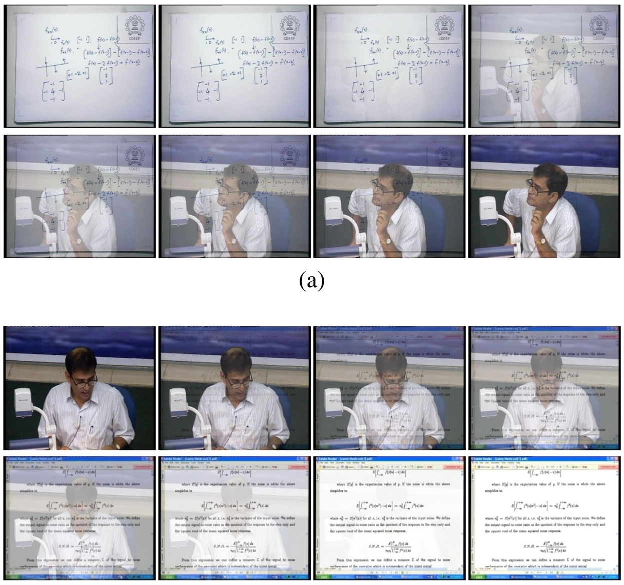
Temporal segmentation of a video into its constituent shots is the basic step towards the exploration about the organization of digital video for all higher level analysis. Video shot detection methods in the literature mostly involve heuristics and fail to perform satisfactorily under varied shot detection scenarios. Though model based shot recognition methods are popular, they are inadequate when a given test video sequence contains transitions. Not much work has been reported which deal with the changes in activities in areas where we have to recognize the activities over a long video sequence. We formulate this as a novel N-class, model based shot detection problem and present a stochastic, asymptotically optimal procedure as a solution to such a scenario, so that neither changes in content nor the types of shot transition hinder the decision making process. A hidden Markov model (HMM), trained using a few relevant features from the different classes of frame sequences is employed to achieve this goal. We present extensive experimental results to demonstrate the effectiveness of our method.

Keywords

Video segmentation, activity recognition, HMM, Shiryaev-Roberts statistics, change point detection.

1. INTRODUCTION

Video is so powerful for knowledge sharing due to its inherent ability to carry and transmit *rich* information through its constituent media. The increased use of digital video necessitates its automatic content analysis for its easy access and fast browsing. All such analysis techniques require a prior knowledge about the scene breaks present in the video. Hence the detection of scene or shot transition is the first and fundamental step of all higher level video analysis. The commonly found transitions are (a) hard cuts, (b) fades, (c) dissolves and (d) wipes. Examples of scene dissolve are shown in figure 1. Hence one has to cope with the change in video activity between two temporal segments and also with the type of transition in-between. Since the performance of scene change detection has a direct impact on the subsequent higher level video analysis, a reliable method should be adopted for temporal segmentation [4].



(a)

(b)

1: Thumbnail examples of a dissolve shot transition. (a) between writing hand and talking head and (b) between talking head and slide show.

The existing shot detection techniques can be classified into two categories : (a) threshold based methods and (b) machine learning based methods, where the former usually uses some function of frame difference for pixels, blocks or histograms [13], which relies on a suitable threshold and the latter employs machine learning approaches[1, 14], which avoids the difficulties involved in threshold selection. These methods exploit the statistical characteristics of the test data and uses an HMM model for classification. Authors in [1] use the differences in signal in both audio and video channels to train the HMM and hence the method does not work well when the scene cuts are not hard enough. Authors in [14] try to train the HMM using features derived during the transition phases. Hence they cannot handle all types of scene changes equally efficiently.

Authors in [2] use the combination of a Bayesian model for each type of transition and a filtered frame difference called structural

[*] Corresponding author

ICVGIP '10, December 12-15, 2010, Chennai, India
Copyright 2010 ACM 978-1-4503-0060-5/10/12 ...$10.00.

information for video shot detection. In [5] authors employ HMM for parsing news video for simultaneous segmentation and characterization. However, it considers both audio and visual features unlike in our work. In [3] the authors formulate a statistical model for shot detection, by using a robust metric for visual content discontinuities and by considering the shot length distribution at shot boundaries. However our method is very different in approach and always yields scene cuts with the minimum delay performance. Although we use HMM for shot detection, the proposed method learns the activities in a given scene and makes no attempt to learn the statistics during transition, making it equally amenable to deal with all types of shot changes. The key concept is that whenever the activity in a scene changes, the data statistics changes and we capture the change point optimally (with respect to detection delay) using the Shiryaev-Robert statistical test [10].

Since the proposed method is based on detecting changes in activities in the scene rather than intensity variation across consecutive frames, it can handle different types of shot changes like wipe, dissolve, etc, but the activities in the scene must be learnt before application. Hence the method is suitable for scenes that can be modeled by appropriate HMMs, like commercials, news video, sports video, instructional (educational) video, etc. In particular, we explain the proposed technique with respect to analysis of educational video due to its widespread current application in distance education. As a matter of fact, in [6], the authors make use of the transcribed speech text extracted from the audio track of video to segment lecture videos. However, this is not a reliable method as any lecture video may have different types of (audio) pauses, introducing unnecessary cuts. Development of an appropriate shot detection algorithm is badly required to analyze instructional video.

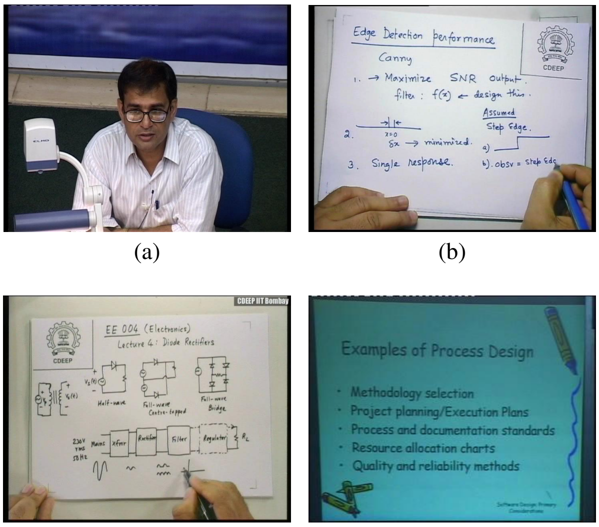
We started this paper by placing our work in the context of existing literature. In the next section we describe the method of feature selection along with our model. In section 3 we describe the theoretical background of change detection procedure and extend it to N-class transition detection which is subsequently followed by a description of the proposed algorithm in section 4. In section 5 we show detailed experiments and analyze the results and in section 6 we have the concluding remarks.

2. CHOICE OF FEATURES AND MODEL

Our objective is to classify shots and to detect the change point in between. A wide variety of videos are available in digital world and their scene compositions also show this diversity. For eg., the classes of scenes found in sports video are entirely different from those found in educational videos. These constituent shots actually evolve from the activity that is being carried out in the live scene. Typically a tennis video may contain scenes like serve, close up, crowd etc., while an instructional vieo may have instructional scenes like talking instructor, writing hand, slide show, discussion etc. Hence a general methodology of scene change detection can be evaluated only with finite classes of example videos. Since ours is a hidden Markov model based one, its training determines the class of video it can cope with. In this paper, we demonstrate the performance of our algorithm on analysing instructional videos.

Since an instructional video is produced at a live class-room environment, the activities captured by the camera are limited and can be classified into a finite number. Generally it contains a talking head, several slide transitions, hand-written portions on black board or slide and/or discussion sessions. Here we define three classes of activities: (1) Talking Head, (2) Writing Hand and (3) Slide Show, depicted in figure 2.

As one can expect, feature selection is the first task to be carried out and for that we have to compare the example frames from these classes. Motion in the first class is more, compared to that in the second which in turn, is greater than that in the third. Hence the energy of the temporal derivative in intensity space can be used as a relevant feature, which is given by


Edge Detection performance
Canny
CDEEP
EE 004 (Electronics)
Lecture 6: Diode Rectifiers
Examples of Process Design
Methodology selection
Project planning/Execution Plans
Process and documentation standards
Resource allocation charts
Quality and reliability methods

(a) (b)

(c) (d)

2: Illustration of different types of scenes in lecture video. (a) Talking Head, (b) & (c) Writing Hand and (d) Slide Show.

$$z_1(t) = \sum_{m=1}^{M}\sum_{n=1}^{N}(F_t(m,n) - F_{t-1}(m,n))^2 \qquad (1)$$

where F_t is the pixel intensity values of the current frame and F_{t-1} is those of the previous frame, M is the number of rows and N is the number of columns in the video frame.

The gray-level histogram $h_t(i)$ gives the distribution of image pixels over different intensity values. The histogram will be very sparse for the slide show class and moderately sparse for the writing hand and dense for talking head. Hence the entropy of the histogram [7] can be treated as another good feature for the detection of these activities, which is given by

$$z_2(t) = -\sum_{i=0}^{255} h_t(i)log(h_t(i)). \qquad (2)$$

Since the slide show and writing hand frames contain certain templates or white pages as background, the gray-level, corresponding to the background will be emphasized in the histogram array. Hence the mode of the histogram is chosen as the third feature, $z_3(t)$ and we use a three dimensioanl feature space, $Z_i = \{z_1(i),\, z_2(i), z_3(i)\}$ for the HMM based classification. Note that the features discussed here correspond to the analysis of lecture videos. For other types of video, appropriate features which are relevant to the scenes present, should be chosen.

Hidden Markov models are stochastic state transit models which make it possible to deal with time-scale variability in sequential data [9]. The basic characteristic of HHM is their learning ability which automatically optimizes the parameters of the model when presented with a time sequential data. HMM consists of a fixed, known number of states. Each of these states are assigned probability of transition from that particular state to any other state including itself. At every instant of time, transition from one state to another occur stochastically and similar to Markov models, state at a point in time depends only on the state at the preceding time. Each state yields a symbol according to the probability distribution assigned to the state. The present state or the sequence of states are not directly observable, hence the name hidden Markov model. These states can be inferred through the sequence of the observed symbols. Rabiner [9] formulated the observations in each state as weighted mixtures of any log-concave or elliptically symmetric probability density function with mean vector μ_{jm}, and covariance matrix $\mathbf{U}_{jm}$, for the m^{th} mixture component in state j. Gaussian mixtures were used to model the probability distribution of the observations. The weights in Gaussian mixture should sum up to one.

2.1 Learning the Activities

To apply HMMs to time-sequential data from the images $\mathbf{I} = \{I_1, I_2, \ldots, I_T\}$, the images must be transformed into observation sequences $\mathbf{Z} = \{Z_1, Z_2, \ldots, Z_T\}$ in the learning and recognition phases where $Z_n = \{z_{1,n}, z_{2,n}, \ldots, z_{J,n}\}$ and J is the dimensionality of the features. For the given example, $J = 3$. In the learning phase, for every class of activity, from each frame I_i of an image sequence, a feature vector $f_i \in \mathbb{R}^n$, is extracted, and pdfs are constructed. Training an HMM means estimating the model parameters $\lambda = (A, B, \pi)$ for a given activity by maximizing the probability of the observation sequence $\Pr(Z|\lambda)$, where A is the transition matrix containing the probabilities of transition from one state to another, B is the observation matrix, consisting the probability of observing a particular symbol in any given state and π is the initial state probability[9]. The Baum-Welch algorithm is used for estimating these parameters[9]. For each and every activity we calculate the features from the sequence of images. These features of $m = 1, 2, \ldots, M$ datasets of a single type of activity are then stacked together to get a three dimensional data matrix, $\mathbf{Z}^j_{tm}$, with other two dimensions being the time n and feature dimension j. This is then fed into Baum-Welch algorithm to get the optimized set of parameters λ for every activity.

2.2 Recognizing the Activities

In the recognition phase, from each frame of the image sequence of the test data, feature vectors are extracted in a similar manner. These vectors are compared against the models for each class of activity . For a classifier of C categories, we choose the model which best matches the observations from C HMMs $\lambda_i = \{A_i, B_i, \pi_i\}$, $i = 1, \ldots, C$. This means that when a sequence from an unknown category is given, we calculate the probability that this particular observation was from any of the category i for each HMM and select λ_{c^*}, where c^* is the category with the best match. The class that gives the highest likelihood score calculated using the forward algorithm is declared as the winner [9]. Although the HMM is a well appreciated method for activity recognition, we have used a unique feature vector. Since we are able to select the relevant features associated with the visual variation in the constituent scenes in the video, we achieve an excellent recognition accuracy.

3. OPTIMAL TRANSITION DETECTION

Hidden Markov model can be used to identify isolated activities as discussed in section 2. The test sequences that are fed into the model are required to be of a single temporal segment. If the sequence consists of more than a single scene, one followed by another, the recognition process will not give the correct result. Hence, we need some method to detect these scene changes automatically and continuously with the minimum possible delay and then segment the test sequence at the transition point. Thus, these temporally segmented sequences can then be fed into HMM individually so as to recognize that particular scene.

Consider the interesting formulation of the change point detection problem. There is a sequence of observations whose distribution changes at some unknown point in time and the goal is to detect this change as soon as possible, subject to certain false alarm constraints [10]. Let the unknown point of time at which the change occurs be ζ. At ζ, all or most of the components of the observation vector change their distribution. Let us consider that there are only two classes, c_0 and c_1 and the change takes place from class c_0 to c_1. If the change point occurs at $\zeta = k$, then the jth component of the feature vector $Z_{j,1}, \ldots, Z_{j,k-1}$ follow the distribution whose conditional density is $f^{(j)}_{c_0,i}(z_{j,i}|z_{j,1}, \ldots, z_{,i-1}),\ \ i = 1, \ldots, k-1$, while the data $z_{j,k}, z_{j,k+1}, \ldots$ have the conditional densities $f^{(j)}_{c_1,i}(z_{j,i}|z_{j,1}, \ldots, z_{j,i-1}),\ \forall\ \ i \geq k$.

Veeravalli [12] proposed the centralized sequential change point detection procedure with a stopping time τ for an observed sequence $\{\mathbf{Z}^n\}_{n\geq1}$, where $\mathbf{Z}^n = \{Z^n_1, \ldots, Z^n_I\}$ and $Z^n_i = \{z_{j,1}, \ldots, z_{j,n}\}$. Thus $\mathbf{Z}^n$ is the accumulated history of all features upto the given time instant. A false alarm is raised whenever the detection is declared before the change occurs, i.e. when $\tau < \zeta$, where τ is the computed detection time. A good change point detection procedure should give stochastically, a small detection delay $(\tau - \zeta)$ provided there are no or very few false alarms. The change point ζ is assumed to be a random variable with some prior probability distribution $p_k = P(\zeta = k)$, $k = 1, 2, \ldots$. It follows from the work of Shiryaev in [11] that if the distribution of the change point is geometric, then the optimal detection procedure is the one that raises an alarm at the first time such that the posterior probability p_n of occurrence of change point exceeds some threshold θ, where the threshold θ is chosen in such a way that probability of false alarm (PFA) does not exceed a predefined value α.

The exact match of the false alarm probability is related to the estimation of the overshoot in the stopping rule. Putting $\theta \leq 1 - \alpha$ gives an optimal solution to this problem. Now, let us assume that the prior distribution of the change point is geometric with the parameter ρ, $0 < \rho < 1$, i.e.

$$p_k = P(\zeta = k) = \rho(1-p)^{(k-1)}\ for\ k = 1, 2, \ldots \tag{3}$$

Shiryaev [10] defined the following two statistics for $k \leq n$. Let

$$\Lambda^k_n = \prod_{t=k}^{n} \prod_{j=1}^{J} \frac{f^{(j)}_{c_1,t}(z_{j,t}|Z_j^{t-1})}{f^{(j)}_{c_0,t}(z_{i,t}|Z_j^{t-1})}. \tag{4}$$

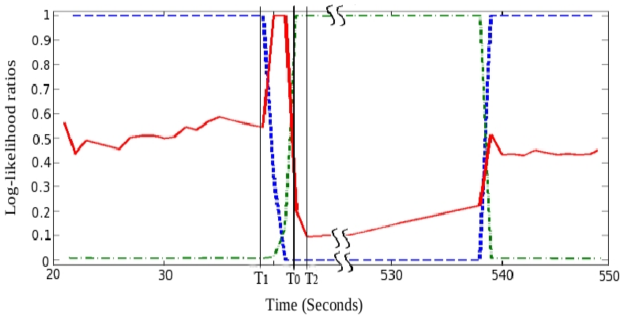
where the term on the right hand side can be interpreted as likelihood ratio and

$$R_{\rho,n} = \sum_{k=1}^{n} (1-\rho)^{(k-1-n)} \Lambda^k_n. \tag{5}$$

Taking into account that $R_{\rho,n} = p_n[(1-p_n)\rho]$, the Shiryaev stopping rule can be written in the following form

$$\nu_{\theta'} = \inf\{n \geq 1 : R_{\rho,n} \geq \theta'\},\ \theta' = \frac{\theta}{(1-\theta)\rho}. \tag{6}$$

where $\nu_{\theta'}$ denotes the time instant of shot change for a specific value of θ'. Shiryaev procedure is optimal in the iid case. However, Veeravalli in [12] showed that it is asymptotically optimal when α approaches to zero under fairly general conditions.



3: Likelihood ratios of the instructional scenes at change point. Each curve represents the likelihood normalized with respect to the current activity. The symbol $\int\int$ represent broken time axis.

4. OVERALL ALGORITHM

In the above sections we described various components that we used in our work. In this section we show how we combined each of these components so as to use it in our framework. We also show how our framework can be used for detecting the change points with minimum delay. Before the change occurs, the output of the proposed algorithm would be same as that of a HMM based recognizer. Hence, we can safely assume that we know the scene before the shot change occurs. Once the change occurs, our goal is to accurately detect the change point between the scenes with the minimum possible delay and very low false positives.

As mentioned earlier Veeravalli [12] used the Shiryaev-Roberts change point detection procedure [8] to solve a two class problem in a distributed sensor network. Unfortunately, the problem of recognizing scene changes cannot be modeled as a two-class problem as there could be several types of shots representing different activities. Hence we modify their algorithm so that it can find transitions among N different classes, thereby making it an N-class change point detection procedure. For each point in time, we calculate the likelihood ratios for every category, the denominator being the likelihood of the present scene. Hence, the highest value among all these ratios is one at any given time instant and it corresponds to that of the present scene. All other likelihood ratios are less than one. The scenario remains the same until a change point occurs. When the scene changes, the likelihood curve of the previous scene drops down and the likelihood of another scene rises. This is illustrated in figure 3 where the likelihood ratios for transition shown in figure 1 are plotted as a function of time. In the curve, change starts at frame T_1 and the transition phase gets over at frame T_2 (ground truthed manually). The detected transition frame T_0 should satisfy $T_1 < T_0 < T_2$. We declare this point T_0 as a valid change point according to the Shiryaev stopping rule. Once it has been declared that a change has occurred, the current frame is taken as first frame and the recognition process is again initialized. This is done continuously to achieve a continuous recognition of scenes and shots. Hence, keeping all things unchanged, equation (6) can be modified to

$$\nu_B = \inf\{n \geq 1 : R^{c_i}_{\rho,n} \geq \theta'\}. \tag{7}$$

for any c_i, where c_i $i = \{i = 1, 2, \ldots, N\}$s are the categories of the scenes present in the dataset. Let c_0 be the present shot already recognized. The Shiryaev statistics gets modified to

$$\Lambda^k_n(c_i; c_0) = \prod_{t=k}^{n} \prod_{j=1}^{J} \frac{f^{(j)}_{c_i,t}(z_{j,t}|Z^{t-1}_j)}{f^{(j)}_{c_0,t}(z_{j,t}|Z^{t-1}_j)}. \tag{8}$$

and

$$R^{c_i}_{\rho,n} = \sum_{k=1}^{n} (1-\rho)^{(k-1-n)} \Lambda^k_n(c_i; c_0). \tag{9}$$

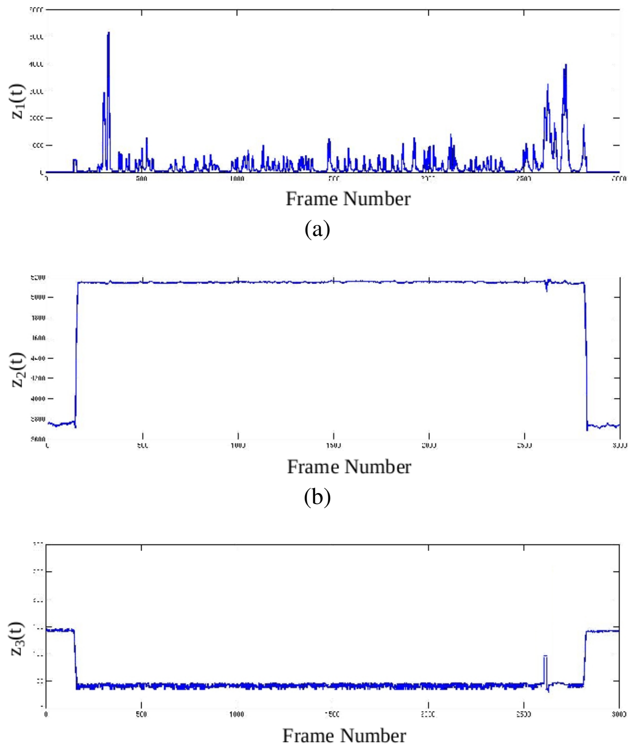
Refering to figure 3 again, we explain the formulation of our algorithm. The figure shows an example of successful detection of change point. The x-axis being the time and the y-axis is the likelihood ratios. The colored dashed curves shows the likelihood ratios of different activities. The vertical lines shows the actual transition boundaries. It is denoted by point T_1 and T_2 in time line. We can see from the plot that initially when the sequence consisted of only one activity, the blue colored dashed curve, corresponding to the activity at that instant has the highest value. At change point, the likelihood for that particular activity decreases and the likelihood for the new activity increases. Hence, the likelihood ratio for the new activity increases. As this increase in ratio crosses the threshold, we initialize the value of the first frame to current frame and again the log-likelihoods are calculated using the HMM discussed in section 2. In figure 3 we reinitialize the likelihoods at point T_0. The new activity is shown by rise in value of another curve to one.

Having arrived at the N-class transition detection solution, we now provide a step-by-step description of the proposed algorithm:
Step 1: Compute the feature vectors for a block of test frames and feed it to the trained HMM models to recognize activity.
For each subsequent frame, do:
Step 2: Compute the Shiryaev's statistics using likelihood ratios to check whether the ratio exceeds the threshold.
Step 3: If *not* go to Step 2; else reinitialize current frame as first frame and go to Step 1.

5. RESULTS AND DISCUSSIONS

We demonstrate the performance of the proposed algorithm on several lecture videos. Our experiments were aimed at finding how well our algorithm performs with respect to false alarm and detection lag when the probability of false alarm is changed. The videos used for the experiments consisted of typically three different lecture scenes, $viz.$ talking head, writing hand and the slideshow presentation, as shown in figure 2. The image sequences used for training were different from the ones used for testing. Features are calculated as explained in section 2. These features are plotted in figure 4 for a video segment which has two transitions. Figure 4(a), (b) and (c) show the features $z_1(t)$, $z_2(t)$ and $z_3(t)$, respectively.

The log-likelihood ratios for the three classes for the temporal duration as shown in figure 4 are plotted in figure 5(a). The green curve denotes talking head, the red one denotes slide show and the blue one denotes writing hand. Note that at frame number 152, the scene transits from slide show to talking head and at frame number 2823, it transits back to slideshow. For a comparative study, the simple histogram difference for the same temporal duration is plotted in figure 5(b), in which it crosses the threshold a couple of times during both the transitions, signifying certain temporal ambiguities as regards when exactly the transition takes place. This is because both the transitions are dissolves with a duration of 10 - 15 frames. For illustration, the first transition in the given temporal duration is shown in a larger time scale in figure 6. The histogram difference curve in figure 6(a) shows considerably high values during the dissolve transition and so there is an ambiguity in change detection. But the log-likelihood curves shown in figure 6(b) do not suffer from such problems and are able to fix the change point without any ambiguity. Hence our method perform quite well irrespective of the type of transition involved. This is because the algorithm

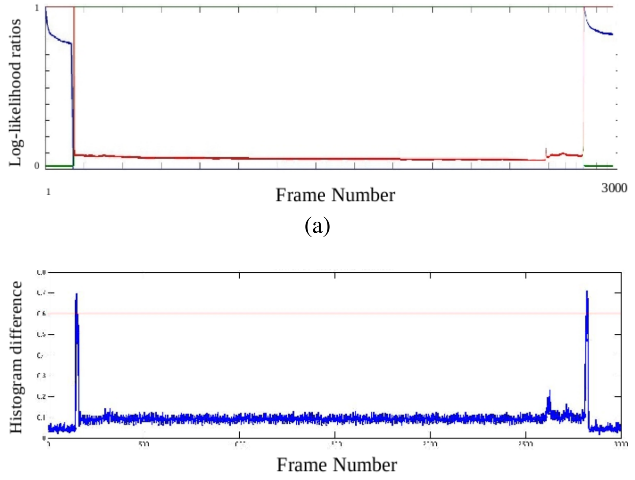


(a)

(b)

(c)

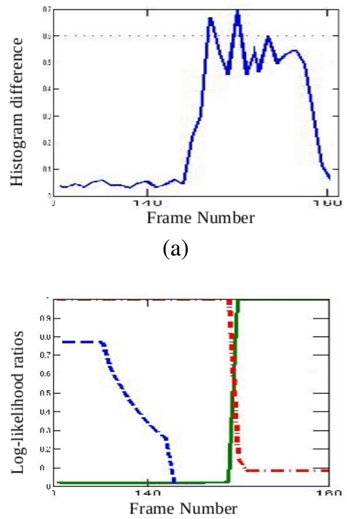
4: Plots of the features used for shot detection in a video segment. (a-c) are the features $z_1(t)$, $z_2(t)$ and $z_3(t)$, respectively.



(a)

(b)

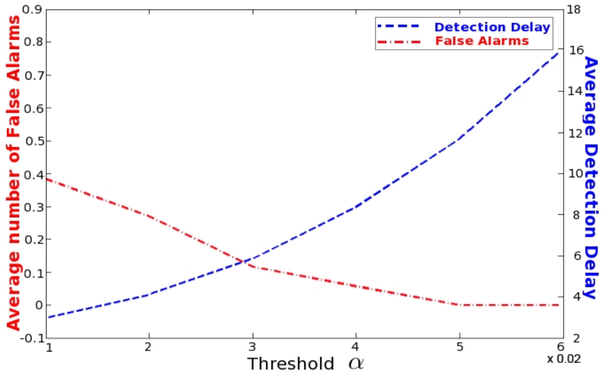
5: Plots of the log-likelihood ratios of the classes and the histogram difference for the temporal duration shown in figure 4. (a) shows the log-likelihood curves and (b) shows the histogram difference.



(a)

(b)

6: Plots of the histogram difference and the log-likelihood ratios of the classes in a larger time scale for the first transition as given in figure 5. (a) shows the histogram difference and (b) shows the log-likelihood curves.

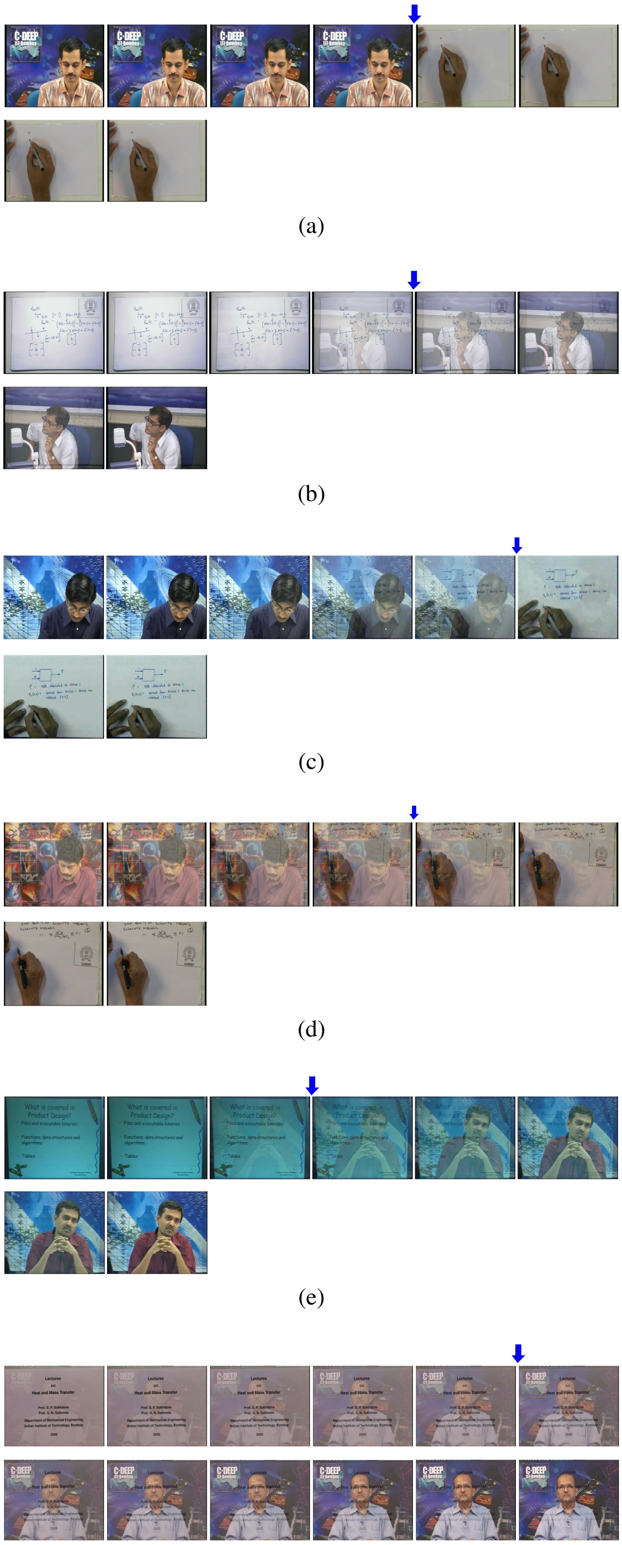


7: Plots showing average number of false alarms and detection delay for different values of threshold α.

automatically recognizes the activity in a video and continuously looks for a change point thereafter.

Our dataset consisted of forty two scene transitions having all possible combinations and different types of shot changes. The duration of the transition varied from a minimum of two frames to a maximum of seventeen frames. The ground truths were obtained through subjective reviewing. The transition detection algorithm takes the likelihoods computed by HMMs in each time frame and computes the probability of transition in linear time.

In the testing phase, the experiments were carried out for different values of threshold α. When α was kept as low as about 0.02, the results showed a large number of false alarms. We increased α in steps of 0.02. The accuracy increases as α is increased. For α = 0.07 only two out of forty two showed false alarms. The remaining sequences gave correct results with no false alarm. When the value of the threshold was further increased to 0.1, all the sequences were detected perfectly and no false alarm was seen in any of the sequences. The average number of false alarms for the experimented scenes with respect to different values of α is plotted in



(a)

(b)

(c)

(d)

(e)

(f)

8: Example results showing thumbnail frame sequence at shot boundaries for different videos and the obtained change points. The arrow marks show the scene breaks.

figure 7. The gradually decreasing plot intuitively substantiates the claim of proper working of the proposed algorithm.

On the similar lines of the above experimentations, we tested for the variations in detection delay given by (T_0 - T_1) against different values of threshold as shown in figure 7. With lower values of α though there were frequent false alarms, the detection delays were considerably small. With the increase in α, the detection delay also increases. When α was initialized to 0.02, it gave very low values of detection delays. There is a marginal increase in the detection lag when the threshold is increased to 0.05. On further increase to $\alpha = 0.1$, the average delay increases to almost twice of the detection lags at $\alpha = 0.05$. Hence, if we keep the value of α low, we get very accurate and low detection delay, but it increases the occurrence of the false alarms. When α is kept high, detection accuracy improves, but the lag increases.

Finally in figure 8 we give a few illustrative results on change detection on different sequences at shot boundaries. Here only one example thumbnail of boundary frame sequence for each video is shown and the corresponding detected scene change is marked by an arrow. Note that in figure 8(a) the scene transition is a cut while in (b) - (f) it is dissolve which occur through eight frames in (b) - (e) and through twelve frames in (f). Also, figures 8(a) - (d) show transitions between talking head and writing hand while (e) - (f) show those between talking head and slide show. Regardless of the type of scene change, our algorithm effectively identified scene breaks at the points marked.

6. CONCLUSIONS

We have presented a novel model based approach for recognizing scene transitions continuously by monitoring the likelihood functions. We proposed a framework that can automatically detect the transition in scenes and thereby separate them at the transition points so that the individual activities can be efficiently and continuously recognized with a guaranteed minimum delay. Even though the transition examples used here are taken from lecture video, our model works equally well for other videos, but what makes it different is the choice of features. The feature selection differes for various classes of videos like educational video, news video, sports video, etc. Hence the results are quite applicable to all types of video irrespective of its scene composition.

7. REFERENCES

[1] J. Boreczky and L. Wilcox. A hidden markov model framework for video segmentation using audio and image features. *Proc. ICASSP'98, Seattle, WA*, pages 3741–3744, 1998.

[2] Han and Kweon. Shot detection combining bayesian and structural information. *Proceedings of SPIE, the International Society for Optical Engineering*, 4315:509–516, 2001.

[3] Hanjalic and Zhang. Optimal shot boundary detection based on robust statistical models. *Proceedings of the IEEE International Conference on Multimedia Computing and Systems*, 2:710–714, 1999.

[4] A. Hanjalic. Shot boundary detection: unraveled and resolved ? *IEEE Transactions on Circuits and Systems for Video Technology*, 12(2):90–105, 2002.

[5] S. K. Krishna, R. Subbarao, S. Chaudhuri, and A. Kumar. Parsing news video using integrated audio-video features. In *First International Conference on Pattern Recognition and Machine Intelligence (PReMI)*, pages 538–543, 2005.

[6] M. Lin, M. Chau, J. F. N. Jr., and H. Chen. Segmentation of lecture videos based on text: A method combining multiple

linguistic features. *Proceedings of the 37th Hawaii International Conference on System Sciences,*, 2004.

[7] N. R. Pal and S. K. Pal. Object-background segmentation using new definition of entropy. *IEEE Proceedings*, 136(4):284–295, 1989.

[8] M. Pollak. Optimal detection of a change in distribution. *Annals of Statistics*, 13:206–227, 1985.

[9] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proc. IEEE*, volume 77, pages 257–286, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc.

[10] A. Shiryaev. On optimum methods in quickest detection problems. *Theory Probab. Appl.*, 8:22–46, 1963.

[11] A. Shiryaev. Optimal stopping rules. *Springer-Verlag, NY*, 1978.

[12] V. V. Veeravalli. Decentralized quickest change detection. *IEEE Transactions on Information Theory*, 47(4):1657–1665, 2001.

[13] R. Zabih, J. Miller, and K. Mai. A feature-based algorithm for detecting and classifying scene break. *Proc. ACM Multimedia 95, San Francisco, CA*, pages 189–200, 1995.

[14] W. Zhang, J. Lin, X. Chen, Q. Huang, and Y. Liu. Video shot detection using hidden markov models with complementary features. *Proceedings of the First International Conference on Innovative Computing, Information and Control (ICICIC)*, 2006.