Video Classification and Shot Detection for Video Retrieval Applications

M. Kalaiselvi Geetha[*]
Dept. of Computer Science and Engineering, Annamalai University, Chidambaram, Tamil nadu, 608 002, IndiaE-mail: gee_siv@rediffmail.com [†]

S. Palanivel
Dept. of Computer Science and Engineering, Annamalai University, Chidambaram, Tamil nadu, 608 002, IndiaE-mail: spal_yughu@yahoo.com

Received: 21-07-2008
Revised: 10-02-2009

Abstract

Appropriate organization of video databases is essential for pertinent indexing and retrieval of visual information. This paper proposes a new feature called Block Intensity Comparison Code (BICC) for video classification and an unsupervised shot change detection algorithm to detect the shot changes in a video stream using autoassociative neural network (AANN) which makes retrieval problems much simpler. BICC represents the average block intensity difference between blocks of a frame. A novel AANN misclustering rate (AMR) algorithm is used to detect the shot transitions. The experiments demonstrate the effectiveness of the proposed methods.

Keywords: Video classification, Block intensity comparison code, Video indexing, Misclustering rate, Shot transition**.**

1. Introduction

At present, tremendous amount of digital multimedia database is accessible by people both in the Internet and television. The quantity of video is voluminous that it is intricate for a human to go through it to choose the video that matches his interest. Consequently, research has begun to analyze the visual media to automate the retrieval task. One approach that users employ is to look for video within specific classes or genres. Accordingly, for retrieving the video of interest, relevant organization and segmentation of the video database is important. Eventually, the objective is to break up the video stream into a set of significant and handy segments called shots. A shot can be habitually visualized as a series of interconnected or unbroken sequence of successive frames taken contiguously by a single camera. Normally, a video is produced by compiling quite a few shots by a procedure called editing. Edit process constructs different kind of transitions from one shot to another such as abrupt and gradual, may take place. A survey of techniques for automatic indexing and retrieval of video data is found in Ref.1.

1.1. *Related work*

The literature reports many approaches for video classification viz., broad genre classification[2,3,4] that categorizes video database into cartoon, sports, commercials, news, music; limited domain classification which organize[5,6,7,8] a video into different

[*] Permanent address of the author.

sub categories and at the final level, the semantic content[9,10] classification that identifies the specific events like highlights and crowd in a video sequence.

Visual media is multimodal in nature and a number of features can be used to extract the information contained in a video. Chromaticity signatures and temporal features[2] are used with HMM as the classifier model. Additional details such as objects and their attributes could improve the performance of the system. In [3] shots in a video database are classified using RBF network with one hidden node per class. RBF networks are more sensitive to the initial situation and hence using generalized RBF network is essential. Normally, the classification methods cannot be applied for digital media in compressed form. Normalized Information Distance (NID) which approximates theoretical Kolmogorov complexity[4] is used for genre classification in compressed video which reflects good performance. Classifying content in a specific genre is also of research interest particularly in sports genre. The trouble arises where a massive quantity of video material are recorded; for example during Olympic games and Asian games. A system for classifying sports video using cue detectors which are indicative of the sport type is proposed in Ref. 5. The cue detectors operate on the key frames for the presence or absence of the objects they are trying to detect. Since video data exhibit dynamic behavior, searching only on key frames may not be sufficient. An edge based sports video classification is presented in Ref. 9 which work directly on compressed video using DCT coefficients. The performance of Refs. 5,9 may be improved by combining other modalities of video. In [7] text biased methods and feature synthesis is applied in [8] for classifying news video. Fisher's Linear Discriminant technique is used8 to condense the dimensionality of the input space. The detailed information of the features is lost by the reduction procedure. Ref. 10.classifies the video by performing shot detection. They used DCT as a likelihood function with HMM model and the results are found to be promising. Most of the techniques use different types of spatial and temporal features alone where the usage of other information is also essential. In [12] the features are modeled using two different classifier methodologies viz., hidden Markov model (HMM) and support vector machines (SVM) and in C4.5 decision tree is used in [13] to build the classifier. In [14] motion descriptors such as foreground object motion and background camera motion are used.[14] Identifying motion like entry/exit; movement/rest could improve the performance of the procedure. Ref. 15 address the problem of video genre classification for five classes with a set of visual features and SVM is used for classification.

Shot transition detection is a key issue which reveals upper level visual content organization. Detection of shot boundaries provides a support for virtually all video abstraction and high-level video segmentation techniques. In addition, other research domains can also be benefited significantly from flourishing automation of shot-boundary detection processes. The shot transition detection problem and the major issues involved are described in Refs. 16,17. They analyses the performance of shot transition detection techniques using color histograms, MPEG compression parameter information, and image block-motion matching. Shots are detected using visual features and shot length.[18] RGB histogram values[19] are used to compare the average values of each color channel in every frame. Color histogram in RGB space is used in Ref. 20. An inter-frame similarity measure based on motion is obtained using a block-matching process.[21] All these shot transition methods fail to address the problem from the viewer's perspective. Human view the visual media by moving their eyes three to four times each second and incorporating information across foveation points.[22] From the observer's perspective[23] shots are detected based on a foveated representation of the video.

1.2. *Outline of the work*

Video data is made up of visual, audio and textual information. Classification of this video data is possible only if all these three information are combined together. However, it is observed that classification is achievable merely with visual information based on the notion that, for an individual to interpret a situation, visual perception alone is adequate. For example, when people talk, visual information in the form of gestures and facial expressions has a greater impact on the listener than the spoken communication themselves and their modulations.

This paper extends the previous work[24] based on visual features by integrating human perception or visual system with the intensity values of the pixels. Normally, low-level features used for video classification do not catch up the high-level semantics entrenched in the images in a video. For this reason, focus should be given on the visual context of the video or how the user perceives a video and consequently, to provide more user-intuitive classification. Generally, in a video, each video genre enjoys its own attributes using which a viewer discriminates between them. Moreover, the intensity distribution within a frame will vary between genres. In accordance with this scrutiny, this paper proposes a novel and efficient method for classification of video based on intensity distribution in a video frame. The method uses a simple procedure that divides a frame into blocks and creates a code called Block Intensity Comparison Code (BICC), which represents the intensity difference between blocks in a frame.

In video classification, another setback is with curse of dimensionality.[25] If the dimensionality of the input space is higher, more feature vectors are needed for training. A simpler, but very effective, way of dealing with high-dimensional data is to reduce the number of dimensions of the input space. One way to avoid the curse of dimensionality is by projecting the data onto a lower-dimensional space. Various techniques[11] viz., principal components analysis (PCA), linear discriminant analysis (LDA), independent component analysis (ICA) are reported in the literature. PCA also called as Karhunen-Loeve transform, the most commonly employed dimension reduction technique is used in this work.

The experiments are carried out in two phases. Initially, experiments are conducted on individual frames in a video sequence with Euclidean distance as a measure and BICC as features. Since, HMM is found to be very effective in capturing the dynamic nature of video, in the second phase, analysis is done on the video sequence with HMM as the classifier model. BICC projected on to a lower dimensional space is used as features.

After classifying the video data, the next step is to organize the video data into manageable and controllable segments to achieve retrieval efficiency. Shot transition detection makes retrieval problems much simpler and easier. The performance of a greater part of the shot transition algorithms profoundly depend on the length of the video, i.e., shot transitions in a short video sequences are more thorny to discover than longer segments. This is due to the fact that, short video segment does not enclose adequate information to make a reliable model and thus cannot identify a shot transition very well. This paper proposes a novel shot transition detection (STD) algorithm called Autoassociative Neural Network Misclustering Rate (AMR) that uses autoassociative neural network (AANN) model to detect the shots of less than 2 sec duration.

2. Feature extraction for video classification

Video is a resourceful media containing audio, text, image, texture, motion information etc. Appropriate recognition and representation of this information is needed for further processing. In that rationale, a feature is a descriptive aspect extracted from an image or a video stream. Visual data exhibit numerous types of features that could be used to recognize or represent the information it reveals. These features exhibit both static and dynamic properties. Classification or recognizing an appropriate video relies on competent use of these features that provide discriminative information useful for high-level classification. The following subsection present the description of the feature for video classification used in this study.

2.1. *Block intensity comparison code*

In this section, the method to compute the Block Intensity Comparison Code (BICC) is presented, which characterizes the intensity variations between blocks in a video frame. The method to generate the BICC is motivated by two facts:

(i) Intensity distribution of video frame in a genre is unique to that genre; Objects in a video frame are recognized by edges. Edges are perceived mainly as intensity changes in a video frame and are easily recognized during quick voluntary eye movements. Each video genre enjoys its own attributes which is unique to that genre, i.e., intensity changes/edges in a video frame is unique to a specific genre as seen in “Fig.1(a)”, Fig. 1(b)” and “Fig.1(e), Fig.1(f)”. Serial frames in “Fig.1(a), Fig.1(b)” shows a few people standing and the frames are almost similar. Sports frames in “Fig.1(e)”, Fig.1(f)” shows cricket genre which are analogous with the pitch, players and field area. These objects are recognized only as intensity variation.

(ii) Human observer visually perceives the object/video only in the area that he/she is most interested in. For example, in the sports video shown in "Fig.1(e)", "Fig.1(f)", the attention of the observer will be focused on the players and pitch area in the cricket frame rather than the rest of the surrounding field area. This distribution of intensity value will vary between genres. Hence, method to compute the intensity distribution in a frame is essential. So, analysis is necessary to compute the variation/distribution in the intensity value within a frame in a video sequence. Based on this analysis, the intensity changes between blocks of a frame are represented by block intensity comparison code (BICC). The procedure for generating BICC is given below.

To extract the BICC features, each image is divided into k x k blocks, each of size (M/k x N/k), where M x N is the size of the image. Image of size 320 x 240 is used for experimental studies. Initially, average intensity in each block is computed. For comparing the intensity distribution in a frame, the average intensity value of each block in a frame is compared with every other block in the frame. Consider 5 x 5 representation of a video frame. "Eq. (1)" illustrates the generation of proposed BICC feature. *i* and *j* represents the *i*th and *j*th blocks in a frame. BICC is generated using

$$y\left[(i-1)\,25+j-\frac{i\,(i+1)}{2}\right]=\begin{cases}1 \text{ if } x(i)\ >x(j)\\ 0 \ \text{ otherwise}\end{cases}$$

$$1\le i\le 25\ ,\ \ 2\le j\le 25\ \ \text{and}\ \ i<j, \qquad (1)$$

where *x(i), x(j)* are the average intensities of the *i*th and *j*th blocks respectively. To generate the BICC, for example, the frame is divided into 5 x 5 blocks to generate the feature vector. "Fig.1(a)", "Fig.1(b)" shows the 5 x 5 representation of frames from the two different serials and "Fig.1(e)", "Fig.1(f)" shows the 5 x 5 representation of two different sports genre. Each block in a frame is compared with every other block to generate BICC using "Eq. (1)". For example, if the image is divided into 5 x 5 blocks, then "Eq. (1)" generates 300 dimensional feature vectors. First element in the feature vector compares the intensity of 1st and 2nd block, second element compares the intensity of 1st and 3rd block and so on. Similarly, last element in the feature vector compares the intensity of 24th and 25th blocks. It is clear from "Fig.1(c)", "Fig.1(d)" that the BICC generated for the serial genre shown are almost similar. "Fig.1(g)", "Fig.1(h)" shows BICC generated for the sports genre of sports frames shown in "Fig.1(e)", "Fig.1(f)" and BICC generated are found to be similar. It is also seen that BICC generated for serial genre differ completely from the sports genre. Thus, "Fig. 1(c)", "Fig.1(d)" and "Fig.1(g)", "Fig.1(h)" demonstrates the high discriminating power of BICC to differentiate the genres. The distance or error between the two comparison codes P = (p_1, p_2, ···, p_n).and Q= (q_1, q_2, ···, q_n) can be calculated using

$$d=\sum_{k=1}^{n}(p_k\oplus q_k) \qquad (2)$$

where $\oplus$ is the bit-wise exclusive-OR operation.

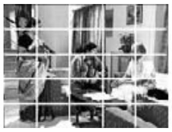

Fig.1(a)

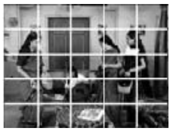

Fig. 1(b)

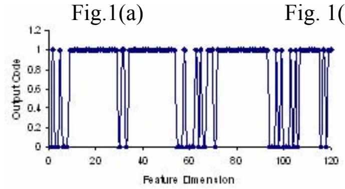

Fig.1(c) BICC for Fig. 1(a)

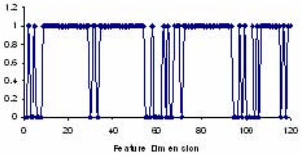

Fig. 1(d) BICC for Fig. 1(b)

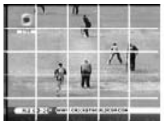

Fig. 1(e)

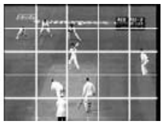

Fig. 1(f)

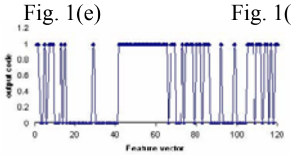

Fig. 1(g) BICC for Fig.1(e)

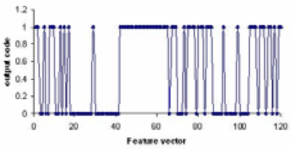

Fig. 1(h) BICC for Fig. 1(f)

Published by Atlantis Press

2.2. *Principal component analysis*

PCA is a useful statistical procedure that has found importance in many fields, and is a well known technique for finding patterns in data of high-ceilinged dimension. It is a way of identifying patterns in data, and expressing the data in such a way to highlight their similarities and differences. Since it is difficult to identify patterns in data of lofty dimension, PCA is a powerful contrivance for investigating the data. The other main advantage of PCA is that once these patterns are found, the data can be compressed by reducing the number of dimensions, without much loss of information. PCA 'combines' the essence of attributes by creating an alternative, smaller set of variables. The initial data can then be projected onto this smaller set. Suppose that $x_1, x_2, \cdots, x_P$ are P training vectors, each belonging to one of N classes $\{\zeta_1, \zeta_2, \cdots, \zeta_N\}$. Then, the training vector, x_p, can be projected to lower dimension vector y_p, using an orthonormal linear transform given by $y_p = WTx_p$. The transformation matrix (W) can be obtained from the eigenvalues and eigenvectors of the covariance matrix (Σ) of the input data. A more detailed discussion can be obtained from Ref. 11..

2.3. *Hidden Markov model*

The hidden Markov model consists of two stage stochastic processes: One is an unobservable Markov chain with a finite number of states, an initial state probability distribution and a state transition probability matrix, and the other is a set of probability density functions associated with each state. The probability density function can be either discrete (discrete HMM) or continuous (continuous HMM). HMM assumes that an input observation sequence of feature vectors follows a multi-state distribution. It is expressed by the initial state distribution (π), the state transition probabilities (A), and the observation probability distribution in each state (B). In HMM training, it estimates the parameter set $\lambda = (A,B, \pi)$, for each class based on the training sequences. It enters a new state based on the transition probability depending on the previous state. After making the transition depending on the current state, an output symbol is produced based on the probability distribution. Ref. 26 gives an overview on HMM.

3. Feature extraction for shot transition detection

Color histogram method is commonly used in video processing applications, which can also provide motion invariant representation of video. Further, they are found to be exceptionally competent in capturing the global information, with low computational complexity and hence are employed in this work. Normally, digital images are represented in RGB color space. In the present work, 24 bits/pixel images (8 bits for R, G and B components each) are used. An n dimensional histogram features are extracted from the video frames by quantizing the RGB color space into n bins. "Fig. 2" shows the 64 bin histogram for the sports image.

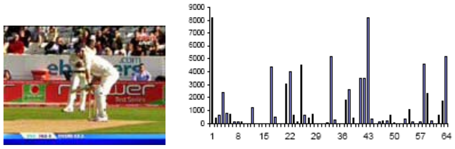

Fig. 2 Sports frame and its 64 bin histogram

3.1 Autoassociative neural network (AANN)

For detecting a shot transition (ST), the analysis focuses on the characteristics of adjacent frames. Hence, it is clear that the model used must be capable of finding the variations that exist between the consecutive video frames. Support vector machine (SVM) is good at this type of learning since the training involves optimization over entire pattern in linear space. Since the video data is made up of shots, it can be viewed as a non-linear model. GMM can be used to confine the distribution of the data, but the components are considered to be Gaussian and the number of mixtures is also set in advance. AANN captures the distribution of the data points depending on the constraints imposed by the structure of the network; just as the number of mixtures and Gaussian functions do in the case of GMM and hence are employed in this work.

Autoassociative neural network models are feedforward neural networks performing an identity mapping of the input space, and are used to capture the distribution of the input data8. Let us consider the five layer aL bN cN bN aL AANN model[27] shown in "Fig. 3", which has three hidden layers. a,b,c are integers that indicate the number of units used in that layer. In this network, the second and fourth layers have more units than the input layer. The third layer has fewer units than the first or fifth. The processing units in the first and third hidden layer are nonlinear, and the units in the second compression/hidden layer can be linear or nonlinear. As the error between the actual and the desired output vectors is minimized, the cluster of points in the input space determines the shape of the hypersurface obtained by the projection onto the lower dimensional space. The nonlinear output function for each unit is tanh (s), where s is the activation value of the unit. The network is trained using standard back propagation algorithm. AANN captures the distribution[28,29] of the input data depending on the constraints imposed by the structure of the network, just as the number of mixtures and Gaussian functions do in the case of Gaussian mixture models (GMM). The detailed overview of distribution capturing ability of AANN is explained in Ref. 27.

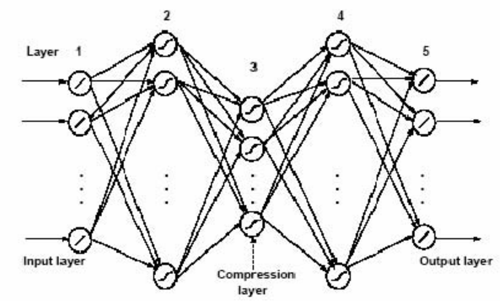

Fig. 3 AANN Network Structure

3.2 Autoassociative neural network misclustering rate (AMR) algorithm

Many of the existing methods[29,30] require manual assistance for marking the shot transitions from video stream for training. That is they need an approved set of predetermined data to train the classifiers before they can be used for detection. AANN is a supervised training algorithm.[27] On the other hand, the proposed AMR shot detection algorithm does not require training samples. Hence, there is no need for separate training data for the proposed shot detection method. The approach is discussed as follows.

Since the work concentrates on shot transition, extracting features from each frame in the video clip becomes essential. Initially a video clip of *n* frames is considered. The extracted histogram features of the video frames from the window w are used to train the AANN model. The approach begins with the assumption that, there is an ST located at frame *p* of the window *w* as shown in “Fig. 4”. If the features come from different shots, they have significant differences and hence AANN cannot cluster these data[27] properly. Conversely, if the features come from a single shot, AANN can effectively cluster these data. The novelty of the proposed approach is that it uses the same histogram features for testing AANN model. Thus AANN is said to perform unsupervised clustering. The AMR algorithm for detecting ST is summarized as follows:

(i) Histogram features are extracted from each frame in a video.

(ii) Select a new frame *p* in the video, where *p* is the frame that is the assumed ST point. Each new *p* is assumed as the ST point and the algorithm tries to find the existence of ST.

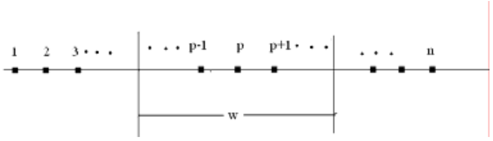

Fig. 4 Windowing approach

(iii) The histogram features extracted from w video frames (*w*/2 frames on both sides of *p*) as shown in “Fig. 4” are used for training AANN model. Each of the features is given as input to the AANN model. The output of the model is compared with the input, to compute the normalized squared error. The normalized error (e) for the feature vector (y) is given by,

$$e = \frac{\|y - o\|^2}{\|y\|^2} \tag{3}$$

where **o** is the output vector given by the model. The error (e) is transformed into a confidence score (c) using c = exp(-e). The average confidence score is calculated for the *w* feature vectors.

(iv) If the average confidence score from the AANN model is less than a threshold indicates that assumed p^{th} frame position is the shot transition point.

A ’hypothetical ST’ is assumed at frame p and tested to determine whether it matches with the characteristics of an ST. After one ‘hypothetical ST’ has been tested, w is moved one frame to the right and step (iii) of the AMR algorithm begins again. These steps are iteratively performed until the window reaches the end of the video stream.

4. Experimental results and discussion

This section analyzes the performance of the proposed BICC for video classification and AMR algorithm for detecting ST.

4.1. *Video classification*

Experiments are conducted with 12 hrs. of video data (10 hrs. for training and 2 hrs. for testing)recorded using a TV tuner card at 25 frames/second, from various television channels at different timings to ensure variety of data. Cricket video is used for sports genre. For cartoons, video data from various cartoon channels are collected. Care is taken to record commercials on different types of products, while recording the commercials. News data is collected from various news channels and Serials from different regional language channels. In the first phase, individual frames of the video genre are used for the experiments. A total of 12,500 frames are used for training and a separate 5,000 frames are used for testing from the recorded data. Block intensity comparison code (BICC) is obtained for all the training and testing frames. Then, exclusive-OR matching is used to find the distance between two BICCs as described in Section. 2.1. The average distance between two different video genres is shown in “Table 1”.

If the data lies in high dimensional space, then an enormous amount of data is required to learn distributions. For dimension reduction, PCA is employed on the BICC features and average Euclidean distance between different video genres using projected BICC is shown in “Table 2”. The performance is improved using PCA. However, in view of the fact that the video data exhibit dynamic behavior, the analysis of the video based on individual frames only will not be sufficient. The dynamic nature of video necessitates the extraction of features from individual frames. Video is processed at 12 frames/sec for extracting BICC features. To confine the time changing nature of video information, video segments are experimented with HMM as the classifier model. Approximately, 10 hours of video (3277 clips) is used for training using the projected BICC as features. Moreover, a separate test video of 2 hours (684 clips) is used to validate the performance of the classifier model. Since there is no specific algorithm/method to fix up the number of states to be used in HMM, the model is tested using diverse states by varying the feature vector dimension and the duration of the video sequences. The confusion matrix for BICC with HMM is presented in “Table 3”.

Table 1 Exclusive-OR matching performance of video genres

	Cartoon	Sports	Comm*	News	Serials
Cartoon	0.34	1.72	1.43	1.84	1.27
Sports	1.53	0.21	1.58	1.43	1.86
Comm*	1.67	1.84	0.32	1.75	1.34
News	1.56	1.34	1.76	0.52	1.78
Serials	1.41	1.79	1.56	1.34	0.26

*- Commercial

Table 2 Average Euclidean distance between projected BICC of video genres

	Cartoon	Sports	Comm*	News	Serials
Cartoon	0.23	1.92	1.63	1.94	1.64
Sports	1.64	0.17	1.64	1.72	1.92
Comm*	1.72	1.92	0.21	1.85	1.74
News	1.81	1.56	1.82	0.32	1.83
Serials	1.72	1.81	1.74	1.86	0.14

*- Commercial

Table 3 Confusion matrix for projected BICC with HMM (in %)

	Cartoon	Sports	Comm*	News	Serials
Cartoon	97.6	0	2.4	0	0
Sports	0	98.7	0	1.3	0
Comm*	1.4	0	98.6	0	0
News	0	0	0	100	0
Serials	0	0	2.8	0	97.2

*- Commercial

The difficulty with HMM is that, it demands a huge volume of training data in high dimensional space. Since the video data exhibit time varying patterns, various experiments are conducted to investigate the performance of the features. Experiments are carried out by varying the dimensions of the feature vector and duration of video sequences. Precision (P) and recall

(R) are the commonly used evaluation metrics in the information retrieval field. To simultaneously assess the number of false alarms and mis-detections, the work used the geometric mean (F-score) as the combined measure of Precision (P) and recall (R) for calculating accuracy which is defined as follows:

$$F_{\alpha} = \frac{2PR}{P + R} \tag{4}$$

where α is weighting factor. For harmonic mean of P and R, α=0.5 is used. Fig. 9 shows the overall performance accuracy for each genre while varying the feature vector dimension. Increase in the dimension of feature vector results in blocks of smaller size. Hence, the intensity variation between adjacent blocks is not significant and thus the feature vector extracted in such a case degrades the performance as shown in "Fig. 5". Further, the improved performance of the model with the increase in the duration of video sequence is shown in "Fig.6". As seen, if the duration of video is less than 10 sec, the HMM is not able to capture the sequence information needed for video classification. If the duration of video is greater than 20 sec, there is no further improvement in the classification performance of HMM. It is evident from Fig. 6 that the proposed BICC with HMM doesn't necessitate the video sequence of much longer length as it gives better performance with 10 sec video. Moreover, it is seen that BICC furnishes exceptional performance even with a lesser amount of data. Experiments are also conducted to demonstrate the efficiency of the proposed BICC with other visual features viz., edge features, motion features, color histogram features[24] and other existing features that are potentially useful for video genre classification, such as shot length[31], the number of faces[32], color moments[33] and color correlogram.[34] The results are evaluated with F-score. Evidently, the proposed approach is found to outperform these features as illustrated and "Table. 4".

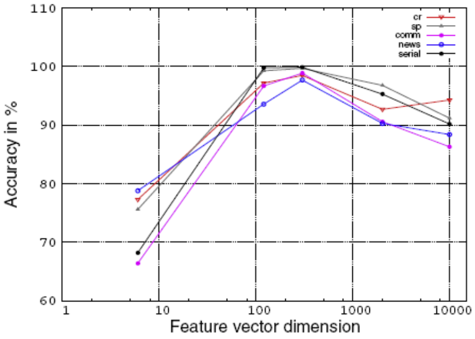

Fig. 5. BICC performance for various feature dimension

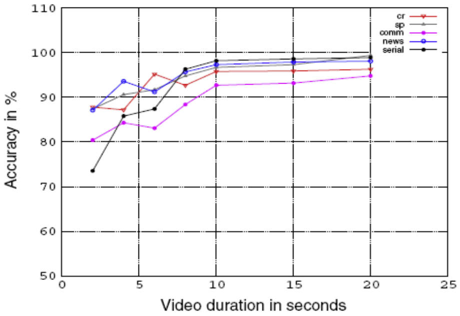

Fig. 6 BICC with HMM for different duration

After classifying the video data into respective genres, the next task is to detect the shot transitions. The data set under consideration consist of 5461 transitions, comprising of 4585 cuts and 876 gradual transitions. This section presents a detailed discussion on the experiments conducted.

Table 4. Performance of various visual features

Classification Scheme	Precision (%)	Recall (%)	Accuracy (%)
Edge	79.2	59.6	66.07
Motion	75.2	49.2	59.46
Histogram	83.6	70.17	76.29
Color Correlogram	96.74	84.87	90.41
Shot length	94.6	81.14	87.36
Faces	93.8	75.6	83.72
Color moments	92.6	81.14	86.49
Prj.BICC+HMM	99.2	97.6	98.4

In general, two kinds of error measurements are used to evaluate STD systems. They are i) mis-detection that refers to the situation where the algorithm fails to locate actual ST and ii) false alarm which refers to ST located by the system that are not actual ST. To evaluate the performance of the AMR based automatic STD algorithm, the experimental results are compared with the manually generated targets. Since an ST is normally coupled with a time scale rather than a single point, ST target is defined as [ST_{start}, ST_{end}]. To minimize problems caused by the different standards of comparison, the experimentally detected ST that fall in the [$ST_{start} - \Delta_t$, $ST_{end} + \Delta_t$] target interval are regarded as correct detection, where Δ_t is a commonly accepted measurement used for analysis of STD. In the experiments conducted, a deviation of 0.2 sec (5 frames on either side) is allowed to detect the correct detection.

4.2.1 *Exploring the best factors for experimental setup*

Experiments are performed to decide the best factors of the proposed algorithm. The window size, number of bins and AMR threshold are set according to the results obtained from the experiments. The structure of AANN model used is (aL bN cN bN aL), where, L - Linear unit, N- Nonlinear unit (tanh (s)), and a, b, c are integers varied to evaluate the performance. The network structure (64L 90N 10N 90N 64L) gives optimal performance. F-score measure as defined by "Eq. (4)" is used for evaluating the results obtained,

4.2.2 *Setting up the parameters*

This section illustrates the effect of varying the number of dimension of the histogram features (number of bins) for various window sizes and AMR threshold. For conducting experiments, *n* dimensional histogram features are extracted using n bins where, *n* = 8, 27, 64 and 125. The performance of number of bins and the window sizes are evaluated using F-score as given in "Eq. (4)". A high F-score is obtained with 64 bins against a window size of 50 as shown in "Fig. 7". False and mis-detections relative to window size and number of bins are shown in "Fig. 8" and "Fig. 9" illustrate the. The window size of 50 with 64 bins appreciably reduced the number of false alarms and mis-detections. Consequently, this study used window size of 50 with 64 bins for optimal ST detection.

Similarly, experiments are carried out to examine the effect of selecting different window sizes and AMR threshold. AMR threshold of 0.8 and a window size of 50 significantly reduced the number of false alarms and mis-detections. Therefore, this study used adjacent windows with a size of 50 frames with the AMR threshold setting of 0.8 for a reliable ST detection.

4.2.3. *Classifying the shot transition types*

After setting the threshold and window size as in Sec. 4.2.2, the next task is to classify the ST; whether the detected ST is abrupt or gradual transition. The abrupt transitions are sudden changes where as for the gradual transition; an image slowly appears/disappears in a solid colored screen. For estimating the decision threshold, a range of values for which a transition has been already detected are examined using variance measure, along with their ground truth information. Variance is calculated using "Eq. (5)" for the range of scores obtained for the window under consideration, to decide the type of transition.

$$\sigma^2 = \frac{\sum_{i=1}^{w} (x_i - \mu)^2}{w} \qquad (5)$$

where, w is the window size, μ is the mean, x_i is the ith frame confidence score. For classifying the ST, the calculated variance are examined and found that, GT reports very low variance when compared to abrupt transition.

The results obtained with the proposed AMR algorithm is compared with already existing algorithms like edge change ratio (ECR) and histogram difference (HD) methods.[35] An example of cut and dissolve detection for the cricket video with the proposed AMR algorithm along with ECR and HD is shown in "Fig 10". As seen in Fig. 10 the abrupt changes are detected as narrow fall downs as shown as c_1, c_2 and c_3. Gradual changes are seen as parabolic portions as shown from d_1 to d_2. As seen, the proposed technique is able to detect a cut with a shot length of less than 2 sec duration. Though, HD is able to detect dissolve, it couldn't differentiate cut and dissolve, which reveals the supremacy of the proposed algorithm. Examples of cut and dissolve transitions is shown in "Fig. 11(a)" and "Fig. 11(b)".

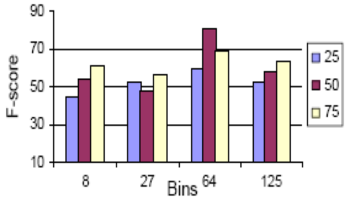

Fig. 7 Performance of the method by varying window size and bins

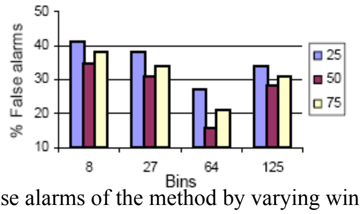

Fig. 8 False alarms of the method by varying window size and bins.

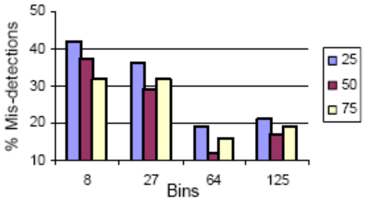

Fig. 9 Mis detections shown by the method while varying window size and bins.

In order to demonstrate the supremacy of the proposed AMR algorithm, another set of experiments are conducted where, the algorithm is applied on a separate test set comprising of large number of video clips with a total of 342 transitions. They are collected from various categories which include cartoon, cricket, football, commercial, and news. “Table 5” shows the performance comparison of the proposed AMR algorithm with ECR and HD methods which demonstrate the efficiency of the proposed AMR algorithm.

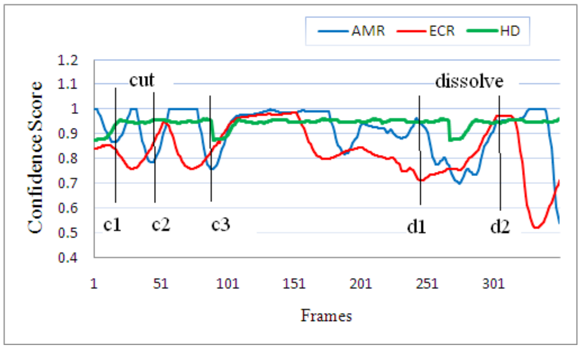

Fig. 10 Comparison of AMR with ECR and HD in detecting cut and dissolve

The time consumption of the proposed AMR algorithm for shot transition detection is measured on a PC with Intel P4 2.53 GHz processor. For detecting shot transitions in a video of 1 min. the algorithm took 85sec. while using the window size of 50 frames which gives better performance. Shot detection time includes extracting and analyzing feature statistics. This shows that the time consumption of the proposed AMR algorithm is comparable with the existing algorithms on the computer used.

4.2.4. *Accuracy computation*

Each detected shot transition point is compared with the ground truth shot transition point S_g to obtain minimum distance. Normalized average error is the average minimum distance. The accuracy of the approach is calculated by measuring the distance from manually generated ground truth by,

$$\theta = \frac{1}{S_a}\sum_{i=1}^{S_a} \min_{1\le j\le S_g} \left| x_i - T_j \right| \qquad (6)$$

where, S_a is the number of shots found by the algorithm, S_g is the number of shots in the ground truth, T_j is the j^{th} ST in the ground truth and x_i is the i^{th} ST detected by the AMR algorithm. The normalized average error for the AMR algorithm is found to be 0.16, while ECR and HD methods produce 0.42 and 0.37 respectively using “Eq. (6)”.

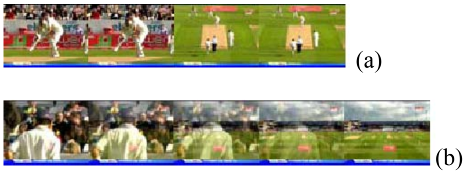

Figure 11 Shot transition detection (a) Cut (b) Dissolve

5. Conclusion

Classification and shot detection form the basis for retrieval algorithms. This paper proposed a new feature called BICC for video classification. The feature gives an exceptional performance of 98.4% with HMM. The results are compared with other existing methods and the proposed method demonstrates its effectiveness in classifying the genres. After classifying the genres, the approach tried to detect the shot transitions using a novel approach called AMR algorithm. The proposed AMR algorithm is unsupervised and the experiments conducted demonstrated the efficiency of the approach.

Further, the classification approach required a video of 10 sec duration for better performance and the proposed AMR algorithm is used to detect only cut and dissolve transitions. Though the results are promising in detecting cut and dissolve, the modern digital video demonstrates numerous types of shot transitions with the advancement in the digital technology. Therefore, method to detect other shot changes like fade in and fade out, zoom, pan, tilt and special camera effects produced using modern digital technology are necessary. Future work involves using much lesser amount of video for classification and detecting other types of shot transitions like fade in, fade out etc., by combining other modalities of video like text and audio.

Table 5. Performance comparison of various algorithms D –Correct detections M – Mis-detections F – False detections

Test Video		AMR algorithm						Edge change ratio						Histogram difference				
		Cut		Dissolve				Cut		Dissolve				Cut		Dissolve		
	D	M	F	D	M	F	D	M	F	D	M	F	D	M	F	D	M	F
Cartoon	47	4	2	16	6	3	12	5	4	8	6	4	18	7	7	6	7	4
Cricket	55	3	1	12	5	2	17	7	3	6	5	3	13	8	8	7	8	6
Football	57	3	2	24	2	2	13	5	7	11	8	6	15	7	9	8	6	3
Commercial	51	4	2	27	3	2	28	9	4	16	6	4	12	8	17	12	7	5
News	42	2	0	7	3	2	11	7	3	9	4	3	9	4	8	7	5	3
Total	252	16	7	86	19	11	89	33	21	50	29	20	67	34	49	40	33	21
Average Error	0.11			0.36			0.42			0.56			0.44			0.64		

References

1. Brunelli, R., O. Mich, & C. Modena. A survey on the automatic indexing of video data, *J. of Visual Communication and Image Representation,* 10(2) (1999) 78- 112.
2. Cheng Lu, Mark Drew, S., & James Au. An automatic video classification system based on a combination of HMM and video summarization, *Int. J. of Smart Engineering System Design*, 5(1) (2003) 33-45.
3. Gillespie, W.J., Nguyen, D.T. Video classification using a tree-based RBF network, *IEEE International Conference on Image Processing,* 3, (2005) 465-468.
4. Kaabneh K., Abdullah A., & Al-Halalemah A.Video classification using normalized information distance, *Proceedings of the Geometric Modelling and Imaging-New Trends (GMAI '06),* pp. 34-40.
5. Edward Jaser, Josef Kittler & William Christmas. Hierarchical decision making scheme for sports video categorization with temporal postprocessing, proceedings of the *IEEE computer society conference on Computer Vision and Pattern Recognition (CVPR 04),* pp. 908-913.
6. John M. Gauch, Abhishek Shivdas.. Finding and identifying unknown commercials using repeated video sequence detection, *Computer Vision and Image Understanding*, 103(1), (2006) 80-88.
7. Peng Wang, Rui Cai & Shi-Qiang Yang. A hybrid approach to news video classification with multi-modal features, *Proceedings of ICICSPCM*, (Singapore 2003), pp. 15- 18.
8. Ming-yu Chen, Alexander Hauptmann. Multi-modal classification in digital news libraries, *Proceedings of the* 2004 *JointACM/IEEE Conference on Digital Libraries (JCDL 04),* 212-213.
9. Lee M.H., Nepal S., & Srinivasan U. Edgebased semantic classification of sports video sequence, *Int. Conf. Multimedia and Expo*, 1, (2003), 157-160.
10. Kolekar M.H., Sengupta S, Hidden Markov model based video indexing with discrete cosine transform as a likelihood function, *Proceedings of the IEEEINDICON Conference*, (2004), 20 - 22.
11. Duda, Hart, & Stork. (2000). *Pattern Classification*, Second Edition, (John Wiley & Sons, Inc.), India.
12. Vakkalanka Suresh, Krishna Mohan, C., Kumaraswamy, R., & Yegnanarayana, B. Combining multiple evidence for video classification, *IEEE Int. Conf. Intelligent Sensing and Information Processing (ICISIP-05)*, India. 187-192.
13. Truong, B.T., Venkatesh, S., & Dorai, C, Automatic Genre Identification for content based Video categorization, *Int. Conf. Pattern Recognition*, 4, (2000) 230-233.
14. Roach, M.J., Mason, J.S.D., & Pawlewski, M., Video genre Classification using Dynamics, *IEEE International Conference on Acoustics, Speech,and Signal Processing, (ICASSP 01)*, 3, 1557-1560.

15. Vakkalanka Suresh, Krishna Mohan, C., Kumaraswamy, R., & Yegnanarayana, B., Content-Based Video Classification using SVM, *Int. Conf. on Neural Information Processing*, (2004), 726-731.

16. Ullas Gargi, Rangachar Kasturi, and Susan H. Strayer, Performance Characterization of Video-Shot-Change Detection Methods, *IEEE Transactions on Circuits and Systems for Video Technology*, 10, (2000), 1-13.

17. Jinhui Yuan, Huiyi Wang, Lan Xiao, Wujie Zheng, Jianmin Li, Fuzong Lin, and Bo Zhang, A Formal Study of Shot Boundary Detection, *IEEE Transactions on Circuits and Systems for Video Technology*, 17, (2007), 168-186.

18. Hanjalic. A., Shot-boundary detection: unraveled and resolved, *IEEE Transactions on Circuits and Systems for Video Technology*, 12, (2002), 90-105.

19. Jacobs. A., Miene A., Ioannidis G.T., and Herzog. O., Automatic shot boundary detection combining color, edge, and motion features of adjacent frames. *TRECVID 2004 Workshop NotebookPapers*, (2004), 197-206.

20. Mas, J., Gabriel Fernandez, Video shot boundary detection using color histogram. TRECVID 2003 Conference, (2003).

21. Lupatini, G.. Saraceno, C and Leonardi. R., Scene break detection: a comparison. *8th International Workshop on Research Issues in Data Engineering*, Vol. 3441, (1998).

22. A. L. Yarbus, *Eye Movements and Vision*. (New York Plenum, 1967).

23. Giuseppe Boccignone, Angelo Chianese, Vincenzo Moscato, and Antonio Picariello, Foveated Shot Detection for Video Segmentation, *IEEE Transactions on Circuits and Systems for Video Technology*, 15, (2005), 365-377.

24. Kalaiselvi Geetha, M., Palanivel, S. HMM based Video classification using static and dynamic features, *IEEE International Conference on Computational Intelligence and Multimedia Applications*, (2007), 277-281.

25. Bellman, R.. *Adaptive Control Processes: A Guided Tour*, (Princeton University Press, 1961).

26. 26. Rabiner, L. R., Juang, B.H., A tutorial on Hidden Markov Models, *EEE ASSP Magazine*, 4-15, (1986).

27. Yegnanarayana, B. and. Kishore, S. P., AANN: an alternative to GMM for pattern recognition, *Neural Networks*, 15, (2002), 459-469.

28. Simon Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd edition, (Prentice Hall, 1998).

29. Jess Bescs, Guillermo Cisneros, Jos M. Martnez, Jos M. Menndez, and Julin Cabrera, A Unified Model for Techniques on Video-Shot Transition Detection, *IEEE Transactions on Multimedia*, 7, (2005), 293-307.

30. Qi, Y., Hauptmann. A., and Liu. T., Supervised classification for video shot segmentation, *IEEE Conf. Multimedia Expo,* 2, (2003), 689-692.

31. Yi-Hua Zhou Yuan-Da Cao Long-Fei Zhang Hong- Xin Zhang, An SVM-based soccer video shot classification, Proceedings of 2005 *International Conference on Machine Learning and Cybernetics*, 9, (2005), 5398-5403.

32. K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. *European Conference on Computer Vision, I,* (2004), 69-81.

33. L. Csink and Sz. Sergyán, Color Content-based Image Classification, *Proc. of the 5th Slovakian-Hungarian Joint Symposium on Applied Machine Intelligence and Informatics*, (2007), 427-434.

34. J. Huang, S.R. Kumar, M. Mitra, W.J. Zhu, and R. Zabih, Image Indexing Using Color Correlograms, *International Journal of Computer Visiun*, (1999), 245-268.

35. R Lienhart. Comparison of automatic shot boundary detection algorithms. *SPIE Conf. on Storage and Retrieval for Image & Video Databases VII*, 3656, (1999), 290–301.