Multimedia Tools and Applications, 20, 135–158, 2003

Framework for Synthesizing Semantic-Level Indices

ANKUSH MITTAL
LOONG-FAH CHEONG
Department of Electrical and Computer Engineering, National University of Singapore, Singapore

Abstract. Extraction of the syntactic features is a well-defined problem thereby lending them to be exclusively employed in most of the content-based retrieval systems. However, semantic-level indices are more appealing to user as they are closer to the user's personal space. Most of the work done at semantic level is confined to a limited domain as the features developed and employed therein apply satisfactorily only to that particular domain. Scaling up such systems would inevitably result in large numbers of features. Currently, there exists a lacuna in the availability of a framework that can effectively integrate these features and furnish semantic level indices.

The objective of this paper is to highlight some of the issues in the design of such a framework and to report on the status of its development. In our framework, construction of a high-level index is achieved through the synthesis of its large set of elemental features. From the large collection of these features, an image/video class is characterized by selecting automatically only a few principal features. By properly mapping the constrained multi-dimensional feature space constituted by these principal features, with the semantics of the data, it is feasible to construct high level indices. The problem remains, however, to automatically identify the principal or meaningful subset of features. This is done through the medium of Bayesian Network that discerns the data into cliques by training with pre-classified data. The Bayesian Network associates each clique of data points in the multi-dimensional feature space to one of the classes during training that can later be used for evaluating the most probable class to which that partition of feature space belongs. This framework neither requires normalization of different features or the aid of an expert knowledge base. The framework enables a stronger coupling between the feature extraction and meaningful high-level indices and yet the coupling is sufficiently domain independent, as shown by the experiments. The experiments were conducted over real video consisting of seven diverse classes and the results show its superiority over some of the standard classification tools.

Keywords: content based retrieval, syntactic features, Bayesian Network, semantic level indices, meaningful-feature selection

1. Introduction

Multimedia content can be modeled as a hierarchy of abstractions. At the lowest level are the raw pixels with unprocessed and coarse information such as color or brightness. The intermediate level consists of objects and their attributes. While the human level concepts involving the interpretation of the objects and perceptual emotion form the highest level.

Based on the above hierarchy, descriptive features in multimedia, furnished to the users of content based technology, can be categorized as either syntactic features or semantic features [10]. A syntactic feature is a low-level characteristic of an image or a video such as an object boundary or color histogram. A semantic feature, which is functionally at a higher level of hierarchy, represents an abstract feature such as the label grass assigned to a region of an image or a descriptor 'empathy of apprehension' for a video shot (a shot is a sequentially recorded set of frames representing a continuous action in time and space by a single camera [1, 14]). Succinctly the retrieval process can be conceived of as

the identification and matching of features in the user's requested pattern against features stored in the database. While extraction of the syntactic features is relatively undemanding, the semantic features are more appealing to the user as they are closer to the user's personal space. As a simple example, it is more rational to query on the 'next interesting soccer shot' than to query on the 'next zoom-in with 10 units'. However, most of the Content-Based Retrieval (CBR) systems exclusively employ syntactic features.

In this paper, we examine the evolution of CBR systems from the perspective of fulfillment of the user's requirements. Most of the CBR systems use syntactic features as the basis for matching and employ either Query-by-Example or Query-through-dialog box to interface with the user. These systems operate at a lower level of abstraction and therefore the user needs to be highly versed in the details of the CBR system to take advantage of them. Most of the work that have attempted to operate on semantic level have confined themselves to limited domain. This is because features developed and employed by them apply to a particular domain well, but are not effective in diverse domains. For example, color histogram works well when indexing is done in the domain of fabric images. But when the database contains news photograph, the color histogram would not be useful [17]. Despite extensive research there exists a lacuna in the availability of a framework that can effectively integrate these features and furnish semantic level indices. Some of the issues in the design of such a framework are highlighted and a few of them are addressed in the paper. Specifically mapping and inference on high dimensional feature space is considered. By performing a prudent synthesis of the features, each of which incorporates knowledge and built-in cues, the characterization of the class can be extracted. This is done though the medium of Bayesian Network that discerns the data into cliques by training with pre-classified data. The Bayesian Network associates each clique of data points in the multi-dimensional feature space to one of the classes during training. This association can later be used for evaluating the most probable class(es) to which that partition of feature space belongs. Experiment on large database and the comparison with the other standard classification tools (Neural networks, decision trees, support vector machines and K-nearest neighbor classifier) show the effectiveness of our approach.

This paper is organized as follows. In Section 2, review and analysis of existing CBR systems are presented. This section also aims at discussing the conventional procedure for matching two images/videos by considering popular CBR systems. In Section 3 we discuss the steps required in order to make CBR systems more serviceable to the user. The lay-out of the system developed by us and the modified Bayesian Network (MBN) used for indexing is presented in Section 4. The method of finding the probabilities of a shot belonging to one or more class(es) is also discussed in this section. Experimental results are given in Section 5. Conclusions and scope for future work follow in Section 6.

2. Review and analysis of existing CBR techniques

In this section, the progress in the field of multimedia indexing and matching techniques used in CBR systems are discussed. The drawbacks in interfacing through Query-by-example or Query-through-dialog box in systems employing syntactic features are brought out. A few works on semantic level have also been discussed.

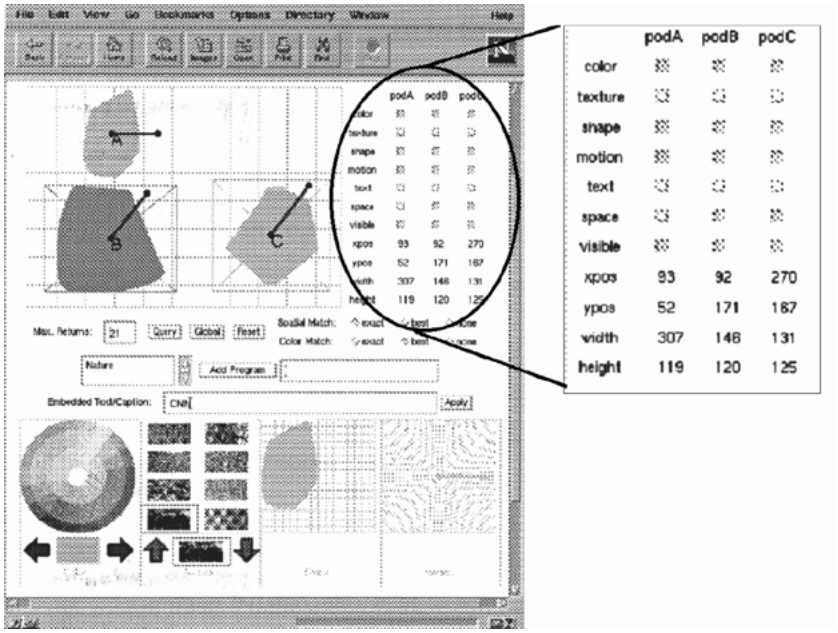

Figure 1. Query through dialog box: VisualSEEK.

2.1. Syntactic and semantic approaches

In recent years research in content-based retrieval has focused on the use of internal features of images and videos computed in an automated or semi-automated way [9, 41]. Automated analysis calculates statistics that can be approximately correlated to the content features. This is useful as it provides information without costly human interaction.

Popular automatic image indexing systems (as CHABOT [21], VisualSEEK [30]) employ user composed queries that are provided through the dialog box. An example of 'Query through dialog box' is shown for VisualSEEK (figure 1). The major drawback in such systems is that queries are more or less technical like "find all shots with a dominant object motion = 10 units/second" or "give me an image with $\{0 < R < 30, 100 < G < 255, 0 < B < 30\}$" for the case when a user is interested in a predominantly green image. The user needs to know the implementation details of the features and the search method.

The only alternative to 'Query through dialog box' was thought to be 'Query by example' technique where a user is presented with number of example images and he indicates the closest. The various features of the chosen image are evaluated and matched against the images in the database. Using some distance metric, the distance between the feature vector (i.e., vector containing the set of features) for the example image and a database image is computed. A few images which have distance less than a threshold are retrieved.

There has been a parallel of 'query by example' in the field of video indexing. The majority of work in video indexing has focused on the detection of key frames called

representative frames or R-Frames [22, 38, 40]. The R-Frames are chosen based on some predefined criteria and the user is again provided as output the choice between various R-Frames of video clips which are close to the user query.

There are a number of defects with retrieving items with 'query by example'.

1. In contrast to clearly defined text search, in image search, using 'query by example', the image can be annotated and interpreted in many ways. For example, a particular user may be interested in a waterfall, another may be interested in mountain and yet another in the sky, although all of them may be present in the same image.
2. It is reasonable for a user to wonder "why do these two images look similar?" or "what specific parts of these images are contributing to the similarity?"(see CANDID [15]). Thus a user is required to know the search structure and other details for efficiently searching the database.
3. Since there is no matching of exactly defined fields in query by example, it requires a larger similarity threshold as it usually involves many more comparisons than query via the dialog box [17]. The number of images retrieved are so many that it makes the whole task tedious and sometimes meaningless.

Researchers have recently been reviewing the appropriateness of these approaches based on syntactic features. There has been some effort in the direction of developing techniques which are based on analyzing the contents of images and videos at a higher level. Shannon et al. [29] have analyzed and looked specifically at video-taped presentations in which the camera is focused on the speaker's slides projected by an overhead projector. By constraining the domain they are able to define a "vocabulary" of actions that people perform during a presentation. In the work done by Gong et al. [39], video content parsing is done by building a priori model of a video's structure based on domain knowledge. Out of the set of recorded shots, shots pertaining to news category are retrieved and the user can define his choice with respect to them. Sudhir et al. [33] have worked in automatic classification in 'Tennis'. Their approach is based on generation of an image model for the tennis court lines and players. Automatically extracted tennis court lines and the players' location information are analyzed in a high-level reasoning module and related to useful high-level tennis play events. Similar work is found in the domain of soccer by Dennis et al. [37] where they try to extract interesting shots from a soccer match for the purpose of presenting them as highlights.

While the above works in the semantic domain disclosed the potentiality of description in semantic terms, a systematic exploration of construction of high-level indices is lacking. These works, though recognizing a need for semantic characterization of video, were focused on narrow domains.

2.2. The matching techniques

In this subsection, matching techniques used in popular CBR systems are examined. By matching technique, we mean the method of finding similarity between two sets of multimedia data, which can either be images or videos. The parameters of such a technique,

which we discuss and analyze herein, are (1) level of abstraction of features, (2) distance measures and (3) normalization of features, if supported, or else the method of relatively weighing the features.

In VisualSEEK [30] a query is specified by the colors, sizes and arbitrary spatial layouts of color regions. Each of the feature measures form different modules with each module utilizing a specific distance measure. The single region distance is given by the weighted sum of the color set ($d^{set}_{q,t}$), location ($d^{s}_{q,t}$), area ($d^{a}_{q,t}$) and spatial extent distances ($d^{m}_{q,t}$). The best match minimizes the total distance.

In JACOB [2], queries are based on color and texture measures. The user chooses a value between 0 and 1 to indicate the relative importance of a set of features over each other. Apart from this naive procedure no other technique for normalization is implemented. In QBIC (Query by Image Content) [9], the query is built on either color, texture, or shape of image objects and regions. The distance measure used for each feature is the weighted Euclidean measure where the weights reflect the importance of components of each feature. CHABOT [21] facilitates image search based on features like location, colors and concepts, examples of which are 'mostly red', 'sunset', 'yellow flowers' etc. Equal weightage is assigned in this system to all features in retrieving the image.

CANDID [15] computes a global signature for every image in a database, where the signature is derived from various image features such as localized texture, shape, or color information. The comparison functions used by CANDID to compare two different signatures $P_{Ii}(\mathbf{x})$, i = 1, 2, are the L_1 distance function, L_2 distance function, similarity function (*sim*) and normalized similarity function (*nsim*). The *sim* and *nsim* functions are:

$$
\begin{aligned}
sim(I1, I2) &= \int_R P_{I1}(\mathbf{x}) P_{I2}(\mathbf{x})\, d\mathbf{x}; \\
nsim(I1, I2) &= \frac{\int_{\mathrm{R}} \mathrm{P}_{\mathrm{I1}}(\mathbf{x}) \mathrm{P}_{\mathrm{I2}}(\mathbf{x})\, \mathrm{d}\mathbf{x}}{\left[\int_{\mathrm{R}} \mathrm{P}^2_{\mathrm{I1}}(\mathbf{x})\, \mathrm{d}\mathbf{x} \int_{\mathrm{R}} \mathrm{P}^2_{\mathrm{I2}}(\mathbf{x})\, \mathrm{d}\mathbf{x} \right]^{1/2}}
\end{aligned}
\tag{1}
$$

A common strategy can be discerned in these different CBR systems: they employ only low level features with distance measures similar to Euclidean distance, with no method to automatically generate the weights of the features. The ineffectiveness of these distance measure in producing a suitable match is well analyzed by Stricker [32]. For example, using the L1-metric to compute the distance between two images, false negatives often result, i.e., not all the images with similar color composition are retrieved. Using a metric similar to the L2-metric results in false positives, i.e., histograms with many non-zero bins are close to any other histogram and thus are retrieved always.

Moreover, these indexing systems suffer from the problem illustrated in figure 2, which depicts the motion activity distribution of a number of shots of soccer. There are essentially two divisions of time in a soccer match: one when there is lot of excitement (and therefore, motion) and another when the motion is less leading to two modes in the probability distribution of the motion activity. In the figure the mean motion activity is also shown by vertical line consisting of asterisks. A reasonable threshold (i.e., 20) is assumed and the thresholded region centered at the mean activity point is also shown. It is evident from the figure that thresholded region characterizes the contents inadequately. Thus if the

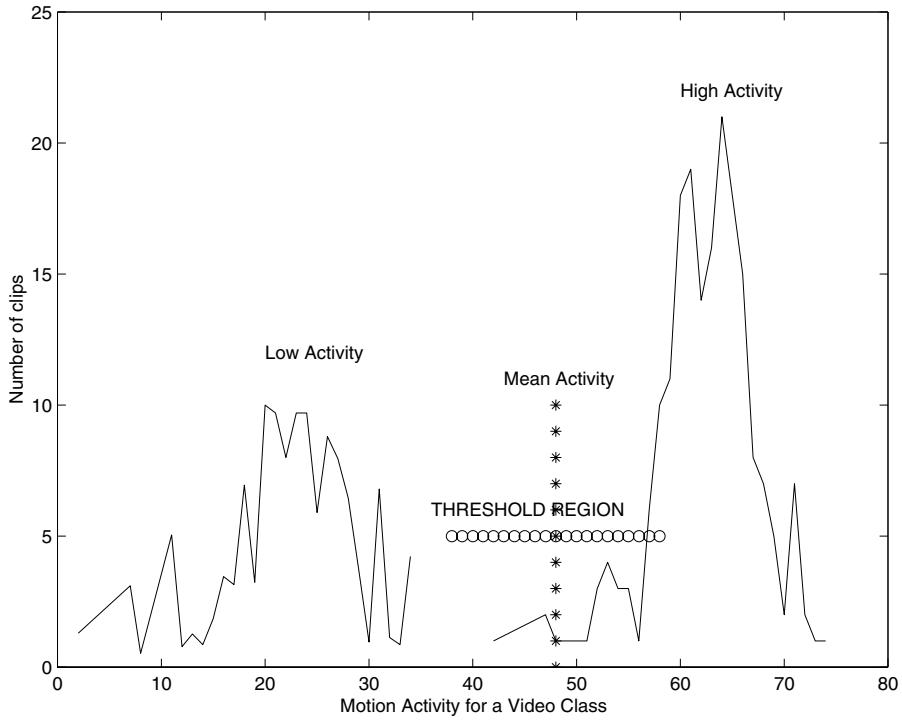

Figure 2. Problem using distance measures in multimodal distribution.

thresholded region around the mean vector is accepted as the region of interest, it would lead to large errors.

None of the indexing schemes discussed so far is capable of dealing with multimodal distribution. Another problem that may arise is that the probability distribution may not be Gaussian, even though it may be unimodal. The distance measures used by these systems inherently assume that with increasing distance from the mean vector, the probability decreases. Thus some sort of Gaussian assumption is implicitly accepted. This is the case for the Bayesian Network employed in [7] which may turn out to be ineffective.

Identifying the meaningful set of features for a given domain is yet another important issue which is unexplored. Many systems (like JACOB [2]) either resort to having the user specify the relative weights to the features or like CHABOT [21], they assign equal weightage to all features in retrieving the image or video shot. By asking the user to specify the weight of various features an injudicious assumption is made that the user is knowledgeable enough to ascertain these to a fine degree. To rely upon human experience is not a pragmatic approach when the aim is to build an integrated system with large number of classes and many features. An automatic technique is required by which more weightage is assigned to the relevant features for a class. Also the issue of dealing with diverse feature measures by normalization or otherwise has not been properly dealt with.

3. The required steps

There are a number of steps which are required in the march towards developing a generic system which can furnish semantic level indices to the user:

3.1. More meaningful features

Identification of a number of symptomatic features which are more fundamental characterization of a multimedia class is required. For example, Time-to-Collision (TTC) appears to be important in a variety of real-world tasks and is often used by the cameraman to create projection of consciousness over the motion picture producing an empathy of fear or defense. Higher order description of TTC such as magnitude of TTC, frequency of occurrence of low TTC in a video etc. would assist in characterization of a class especially in sports like ball games, rugby etc., cinema categorization like action movie and horror and in animal documentary like chase, flock of birds flying/group of animals running etc. Second example is of Spatial-Temporal categorization of motion activity where the distribution of the motion through the extent of the shot in spatial domain forms two types of shots: local motion shot (like in News, interviews etc.) and global motion shot (like in soccer, car race etc.). Similarly the motion activity in temporal domain forms division into homogeneous shots with uniform motion (like uniform panning shot etc.) and heterogeneous shots [11].

3.2. Mapping and evaluation

To associate a class with a set of feature vectors is called mapping while finding the most plausible class or classes of a given feature vector value assignment, fully specified or partially specified, is known as evaluation. Mapping would facilitate building an a priori knowledge base required to incorporate higher level features. In this step, using a large number of classified training samples, feature space of each class, with the attendant probability distribution, has to be estimated. The evaluation procedure is used when a new shot is presented to the system and a label from the pre-existing class has to be assigned to the shot.

In our work, we address the issues relating to mapping and evaluation in detail. In order to achieve these, a general and automatic algorithm is required which could evaluate relevance of a feature for each of the class. One possibility is the application of neural networks in performing the aforementioned task of mapping and classification. Neural network systems, like Multilayer Perceptrons (MLP), get trained by parameters derived from a number of pre-classified shots with a suitable activation function, and a number of layers. After sufficient training it might be possible to evaluate the closest class when presented with an unclassified instance. However, there are many complex issues in the implementation of neural networks as a classifier in multimedia indexing. MLP implementation requires magnitude scaling of inputs and outputs to make them within the reasonable limits of ± 0.6 to ± 0.8 [12]. This scaling, however, results in losing the essential distinction and variation in the input space particularly when there is a large number of features with different

scales and variation. It is also well known that the performance of neural networks is very critically dependent on the structure of the networks (for example, number of hidden layers, number of perceptrons etc. in MLP, positioning of and number of centers in Radial-Basis Function (RBF) Network. Besides these, evaluating the relevance of features and performing classification are not straight forward tasks in neural networks. Finally, it is not possible to evaluate output with partially specified input feature vectors, which would place a restriction on the functionality of the indexing system. Similar problems are encountered when using a classification approach by clustering or k-nearest-neighbor classifier. Besides the problem of normalization of features, these approaches are limited as they cannot evaluate the relevance of the features. In the ensuing sections, we present a Bayesian framework to solve these problems.

In Bayesian Networks, a probabilistic summary, which contains the conditional probability of each feature for a given class and the prior probability of the class, is stored. During training with pre-classified sample, the network updates the probabilities stored with the specified class. Bayesian networks are employed to represent causal knowledge, as for example, from diseases to symptoms in Medical diagnosis application [34]. Ferman and Tekalp [7] and Naphade et al. [20] have recently employed probabilistic framework to construct descriptors in terms of location, objects and events.

The key attribute for each image/video is a feature vector which corresponds to a point in a multi-dimensional feature space. Presuming that the retrieval system is endowed with a few indicative and descriptive features, a high level index or multimedia class is predicted to occupy a distinct representative set of sub-spaces in the multi-dimensional feature space formed by all classes in the domain. Through a process known as feature partitioning (explained in Section 4.3.1), the associative strength (or causal probability) of a particular index, with respect to such a clique, is ascertained by the degree of density of data points in the clique. Thus a non-meaningful feature having no highly populated clique for the class does not participate actively during the inference step.

3.2.1. The structure in multimedia. The assumption that meaningful high level indices would form characteristic cliques in feature space does not restrict the present work to confined domains as done by previous work in semantic domain as discussed in Section 1 as it is commonly accepted that editing effect, movement of the camera, subjects in the frame, colors, variation of lighting effects etc. are meaningfully-directed and intentional. For example, Nack and Parkes [19] show that the *motivation*, *theme realization* and *resolution* dictate creation of meaningful scene and therefore structure in choosing the film elements. Thus, a pan operation, which is described as a particular rotation of the camera on its vertical axis from a stationary point, may be used to establish the contiguity of screen space, and leads the viewer to understand and feel from this expression the 'sweep' and 'scope' of a monument valley landscape and the stagecoach crossing it [31].

Of course, the high level index or class mentioned herein should have structure and should not be built using arbitrary rules. These structures are general ones for they reflect the stable nature of the world and the ways in which viewers perceive and interact with the world. They embody the relationship of the viewers perceiving the world. Naturally, such structure is observed in choice of a field or courts where a particular game is played, the motion in

the frame (in category of local motion or global motion etc.) and so on. Previous works by Vasconcelos and Lipman [35] and Fischer et al. [8] support this reasoning. Vasconcelos and Lipman [35] have integrated shot length along with global motion activity to characterize the video stream with properties such as violence, sex or profanity suggesting that there exists a relationship between the degree of action and the structure of visual patterns that constitute a movie. Similar work is done by Fischer et al. [8] where they use the style profile of a movie to match with a few video genres. The classification module in their work is comparable to a human expert who is asked about his/her evaluation of closeness of a particular feature.

3.3. Query system

A query system which allows user constructs in common language with minimal restriction providing choices of class is required. This involves translating user language to constructs at semantic level.

The present framework can support with larger feature set common queries like "show next sports shot", "show interesting shots from a soccer match", "go to climax of the movie" etc. One interesting point to note in the above example is that zoom-in may be one of the characteristics for an interesting shot in a soccer match but a user does not need to know it. Thus a user will not be required to construct his query in low level details. Such queries are very meaningful in database labeling and video-on-demand applications, where a user can request to see video of his choice like soccer. Applications such as censoring of violence and thrill can also be designed using the framework.

4. The probabilistic framework and mapping

In this section, we first describe our framework and some useful properties of the framework, followed by a more technical description of the CBR Bayesian network, and the evaluation of the most probable hypothesis.

4.1. An overview of the Bayesian CBR system

An overview of the CBR system presented in this paper is depicted in figure 3. The shots are input to N feature extractors. During the development stage of the system, the feature vector, which is formed by combining the output from the various extraction modules, goes to the learning module where parameters for the Bayesian Network are extracted and passed on to the Bayesian Network. In the query stage, the extracted features are passed on to the Bayesian Network which generates probabilities of the given shot belonging to the various classes. A label of most probable hypothesis (MPH) is put on that shot. On receiving the user query, the Query manager translates it in terms of class or a group of classes and tries to match with the MPH label on each shot. The shots with the best matching are retrieved and displayed to the user.

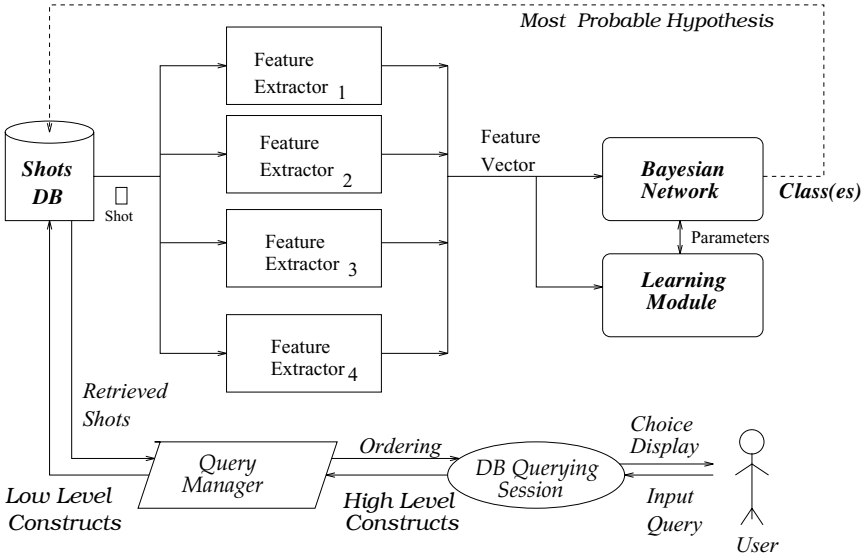

Figure 3. Proposed content based retrieval system.

4.2. Properties of our framework

Some of the useful properties of our framework are:

1. The user is not required to know low-level details like color range, motion value etc. He/she is provided with reasonably high level indices to formulate his/her query. The framework permits the direct usage of common semantic-level indices like "News", "climax" etc.
2. The evaluation of the parameters of classification and especially the relevance of each feature with respect to each class is performed automatically. The number of features can be indeed large. The features where a class shows consistency in value range are meaningful and therefore only these are selected. The feature for which the probability distribution is widely spread is not so relevant as the one that has compact distribution.
3. The framework does not make any assumptions about the PDFs of the features of a class. There exist many possibilities: unimodal with Gaussian, unimodal without Gaussian or arbitrary multimodal distribution. The indexing system works well for all the above cases.
4. The mechanism to support partial query by the user and also to label multiple indices on a single image/video shot, which give multiple perspective to one image/shot, is provided. The labels need not be exclusive in nature.
5. Since the framework is based on choosing the most probable hypothesis based on inference, there is no need for thresholding, which otherwise plays an important role in deciding the number of shots retrieved given a user query. In the absence of such

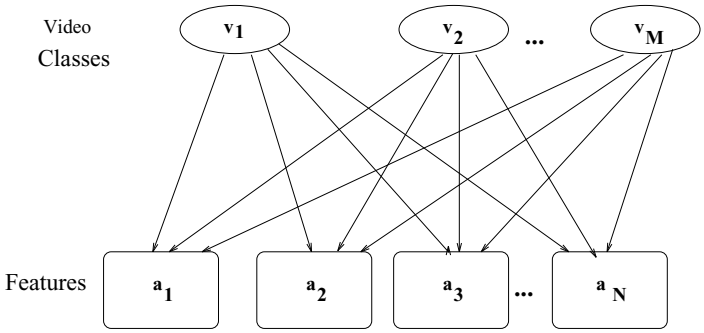

Figure 4. Schematic CBR Belief Network.

a strategy, adjusting the value of threshold to an optimum level often becomes a big concern.

4.3. The computational problem

Let the set of Video classes be denoted by V, the set of features (or attributes) by A, and the relation $C \subseteq V \times A$ represents the pairwise causal associations between Video classes and feature values (see figure 4). Video class v_i causing feature value a_j corresponds to a link in the figure. A subset of A, denoted A^+, represents the set of all feature values that are present, while $A^- = A - A^+$ represents the set of feature values assumed to be absent. The way we compute A^+ and A^- from continuous value of features is discussed later in this section. Each causal link between video class v_i and feature value a_j is associated with a number $c_{ij} \in (0,1]$, called causal strength from v_i to a_j, representing how frequently v_i causes a_j. In other words, $c_{ij} = \mathrm{P}(v_i \text{ causes } a_j \mid v_i)$. If there is no causal link between v_i and a_j, then c_{ij} is assumed to be zero. The evaluation of c_{ij} is done in the learning module, which also constructs dynamically the structure of the Network as discussed in the next Section 4.3.1.

If we let V_T be interpreted as all $v_i \in V$ that are present and all $v_k \in V - V_T$ that are not occurring, a relative likelihood measure for a hypothesis V_T given A^+ can be derived as

$$L(V_T, A^+) = \prod_{a_j \in A^+} \left[1 - \prod_{v_i \in V_T} (1 - c_{ij}) \right] \prod_{a_l \in A^-} \prod_{v_i \in V_T} (1 - c_{il}) \prod_{v_i \in V_T} \frac{\pi_{\mathrm{i}}}{1 - \pi_{\mathrm{i}}} \tag{2}$$

where π_i is the priori probability of occurrence of video class v_i.

The problem remains to identify the most probable hypothesis $V_{\max}$ with the highest L value among all possible hypotheses. Since classes are assumed to occur independent of one another, any combination is possible. The potential search space for a solution is generally extremely large (i.e., $2^{|V|}$) and thus search programs result in computational difficulty.

A lot of work has been done (for example, [25, 26]) in the field of inference to avoid this computational intractability. The key to computational efficiency when performing

inferencing in belief networks is to take advantage of the conditional independence specified by the network topology and to find ways of propagating the impact of a new evidence locally without having to calculate the entire joint distribution. For a singly connected network, inference may be performed using an efficient algorithm to propagate the effect of a new observation [25]. However, a more complex method is required for deriving inference in a multiply connected network such as the one illustrated in figure 4. The standard way of dealing with loops in multiply connected networks are clustering and conditioning. Clustering (as given in Lauritzen and Spiegelhalter [16]) involves forming compound variables in such a way that the resulting network of clusters is singly connected. Conditioning involves breaking the communication pathways along the loops by instantiating a select group of variables.

The computational complexity of these algorithms has not been completely analyzed in terms of the network topology. Nevertheless all are liable to combinatorial problems if there are many intersecting cycles. Cooper [4] has shown that the problem of inference to obtain conditional probabilities in an arbitrary belief network is NP-hard. This suggests that it will be quite useful to look for approximate or bounding methods rather than exact algorithms.

An efficient algorithm, given by Henrion [13], based on branch and bound search identifies k most probable explanations. The theory behind this is that out of hundreds of uncertain variables in a large knowledge base, search is performed on only those few that are important for the current problem using pruning. The algorithm is designed for medical applications where diseases and findings are binary, either present or absent. However, our scheme focuses on continuous feature space. The input to Bayesian Network is the feature vector, which is formed by combining the output of feature extraction modules. The output of Bayesian Network is the most probable hypothesis (a hypothesis is a set of video classes: none, one or several) also referred to as the most probable explanation (MPE).

4.3.1. Discretization of the features. To deal with continuous variables without making assumptions about the underlying probability distribution is a difficult task. Here, we devise a simple and elegant procedure that can be employed for aiding classification.

When features have numeric values, a common approach (like in [24]) is to compute $P(a_k \mid v_i)$ by assuming that within each class, the values for a feature are normally distributed about some mean. Mean and standard deviation of a class for a feature are evaluated using standard statistical approach. It was shown in Pazzani [23] that normality assumption of numeric data may lead to poor performance for many practical systems like electrical faults. He suggested the discretization of the variables into a small number of partitions for each feature.

We found that it is not optimal to have P partitions all of equal sizes as some partitions become densely populated leading to poor discrimination. Thus we propose a statistical approach for finding the boundary points. The algorithm works as follows

1. Start by discretizing each feature into fixed number of partitions (say four).
2. With the addition of a new value in a partition, check the density of partition. If it is greater than the defined threshold θ, divide the partition into two at the median value.
3. If the number of partitions becomes large, stop dividing the partitions. This would happen

when a feature is having almost the same value range for most classes and therefore not very useful for discrimination amongst classes.

Partitioning the features in the above way requires no prior assumption about the probability distribution and yet gives a good representation of the distribution of data (which are some of the desired properties of the indexing scheme). Let feature a_j be partitioned into p_j sub-features, a_{j_k} for $1 < k < p_j$. After the execution of the above discretizing step, the modified Bayesian Network (MBN) would have increased the number of leaf nodes as each partition (a_{j_k}) of a feature can be considered as a Boolean variable. At a particular instance, one and only one of the sub-features, denoted by $a^{\tau}_{j_k}$, will have the status of being present (i.e., True). The link probabilities are also redefined as c_{ij_k}. Similarly after partitioning, ${A^+}_k$ refers to the set of all $a^{\tau}_{j_k}$ present and A^-_k refers to the set of other sub-features (i.e., those which are not present).

4.3.2. Noisy-OR model of the network. The effect of multiple class on a common sub-feature is modeled by noisy OR gate. This assumes that the mechanisms whereby different classes can result in a given sub-feature assignment operate independently. Noisy OR gate is suitable when a sub-feature can be present only by the class set, V. However in real life, there is always a possibility that a sub-feature may be present even when none of the classes in V is present. Therefore we permit leakage in the noisy OR gate which assigns null hypothesis (H_ϕ) with some finite value not necessarily equal to zero. Leak probability corresponding to null hypothesis is defined as $c_{\phi j_k} = P(a_{j_k} \mid H_\phi)$. Thus the probability that a_{j_k} will occur given any combination of classes is

$$P\left(a_{j_k} \mid H\right) = 1 - \left(1 - c_{\phi j_k}\right) \times \prod_{\forall v_i \in H} \left(1 - c_{ij_k}\right) \tag{3}$$

The noisy OR model has two assumptions: all features and sub-features are conditionally independent of each other given any hypothesis, and the classes are marginally independent. Using these assumptions and letting q_{ij_k} denote 1 - c_{ij_k}, $P(H, A)$ can be calculated as follows:

$$\begin{aligned} P(H, A) &= P(H)P(A \mid H) \\ &= P(H)P(A^-_k \mid H)P(A^+_k \mid H) \\ &\qquad \text{(by conditional independence assumption)} \\ &= P(H_\phi) \times \prod_{\forall v_i \in H} \frac{\pi_{\mathrm{i}}}{1 - \pi_{\mathrm{i}}} \times \prod_{a_{j_k} \in A^-_k} q_{\phi j_k} \times \prod_{a_{j_k} \in A^-_k} \prod_{\forall v_i \in H} q_{ij_k} \\ &\quad \times \prod_{\forall a_{j_k} \in A^{\tau}_{j_k}} \left(1 - q_{\phi j_k} \prod_{\forall v_i \in H} q_{ij_k}\right) \end{aligned} \tag{4}$$

4.3.3. Finding the most probable hypothesis. To find the most probable hypothesis, a search in the space of $2^{|V|}$ possible hypotheses is required. An efficient algorithm could be implemented by requiring that a video class, v be added to a hypothesis, H (initially H_ϕ)

only when there is an increase in the posterior probability i.e. when $P(H \cup \nu \mid A)$ is greater than $P(H \mid A)$. Defining Marginal explanatory power (MEP) as follows:

$$\begin{aligned}\text{MEP}(\nu, H) &= \frac{P(H \cup \nu \mid A)}{P(H \mid A)} \\ &= \frac{P(A \mid H \cup \nu)P(H \cup \nu)}{P(A \mid H)P(H)} = \frac{P(H \cup \nu, A)}{P(H, A)}\end{aligned}$$

The MEP is a measure of the increase or decrease in the degree to which the hypothesis, H, explains feature value A due to the addition of the video class, ν. The MEP of a class ν for a hypothesis H cannot be less than the MEP of ν for any extension of H (similar deduction is made in Henrion [13]). This implies that if $\text{MEP}(\nu, H) \leq 1$ then ν can be eliminated as a path for exploring as an extension to H, since it cannot lead to a more probable hypothesis. Thus the only classes which need to be considered are those for which the MEP is greater than 1. This provides a strong pruning heuristic.

Using Eq. (4), we derive a simple form of $\text{MEP}(\nu, H)$ for the CBR Bayesian Network. Note that the first and third term are independent of the hypothesis in Eq. (4) and hence they will cancel out on taking ratio in MEP. The second term deals with prior probabilities of classes only and therefore will evaluate to $\frac{\pi(\nu)}{1-\pi(\nu)}$ which is the a priori odds to class ν.

The fourth term involves more computation than any other term in Eq. (4) as it involves multiplying over all sub-features that are not present and all classes. Fortunately enough, due to the special structure of the CBR network, it reduces to $\prod_{\forall a_{j_k} \in A^-{}_k} \; q_{\nu a_{j_k}}$ when we take the ratio.

Thus MEP reduces to the following:

$$\text{MEP}(\nu, H) = \frac{\pi(\nu)}{1-\pi(\nu)} \times \prod_{\forall a_{j_k} \in A_k^-} q_{\nu a_{j_k}} \times \frac{\prod_{\forall a_{j_k} \in A^\tau_{j_k}} \left(1 - q_{\phi j_k} q_{\nu j_k} \prod_{\forall v_i \in H} \; q_{i j_k}\right)}{\prod_{\forall a_{j_k} \in A^\tau_{j_k}} \left(1 - q_{\phi j_k} \prod_{\forall v_i \in H} \; q_{i j_k}\right)}$$

The search algorithm (pseudo code) using MEP is presented in the appendix.

5. Experiments and results

This section describes experiments using real video sequences. Details of experimental setup are first described. The working of the partitioning and inference processes are then illustrated. This is followed by the results of comparison with a few standard tools.

5.1. Experimental setup

A domain comprising of diverse video classes was selected as shown in Table 1. Figures 5 and 6 show an example frame of each type of video class. The sequences were recorded from TV using VCR and grabbed in MPEG format. The size of the training database was 3500 sequences comprising of 700 sequences for each class. The size of test database was 1400 sequences i.e., 200 sequences for each class. The frame dimension was 352×288.

Table 1. Set of features used and set of video classes.

Notation	Type	Count	Index	Type
CLR	Color	$16 * 3$	1	Basketball
FD	Feature density	16	2	Educational
HST	Histogram mapping	$16 * 3$	3	Music (MTV)
MTN	Motion	16	4	Soccer
IV	Intensity variation	$16 * 3$	5	Swimming
		176	6	Tennis
			7	Volleyball

5.1.1. The feature extraction process in our work. The set of features that were chosen is presented in Table 1. A brief description of each feature is as follows: Color is the average color of each frame of the video sequence. Feature density is the degree of details such as edges present in each frame. Histogram mapping is the weighted average of the three most prominent histogram bins constructed from each frame. Motion is the number of pixels moved in the frame and intensity variation is the average of intensity difference between the same pixel in two consecutive frames in a sequence. Both the motion and feature density are expressed as percentages.

While motion and color features are commonly used in CBR systems, intuitively, other features also seem to be important in characterization. Different type of videos shows different type of intensity variation. For example, commercials show high variation while Soccer may not, though significant motion may be present in both of them. Similarly,

Figure 5. Basketball, Educational, and MTV.

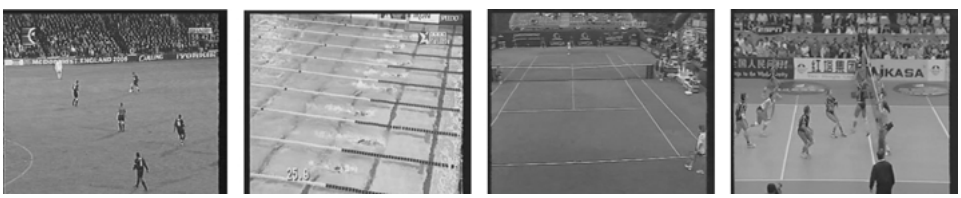

Figure 6. Soccer, Swimming, Tennis, and Volleyball.

basketball or volleyball video have higher feature density as compared to tennis video, which has more spatial smoothness.

We employ fine-grained approach in analyzing each data into segmented smaller regions. Although working with local features implies increasing the complexity in the feature extraction process and increasing the dimension of the feature space, they are still employed in our CBR system as local features provide more effective characterization of an index. Each of the properties like histogram color, motion etc. could be considered as a feature in the segmented domain, where these properties are derived for each $\mathcal{Z}$ fragments of a frame. The effectiveness of knowledge representation by fine-grained approach in multimedia indexing can be illustrated by the following examples: in soccer video, the lower half of the frame generally depicts the field and upper half may be sometimes the audience or the advertisement boards. The color of segmented lower half could be an appropriate cue. The other example concerns the degree of distribution of motion thus leading to spatial-temporal categorization of motion activity as already discussed.

5.2. *Discussion of results*

Figure 7 shows the output of our feature partitioning algorithm with the color as the feature. Eleven sub-features (i.e., A–K) of color were formed as a result of the partitioning. The probability distribution of the classes over the partitions is depicted in the figure. The X-axis

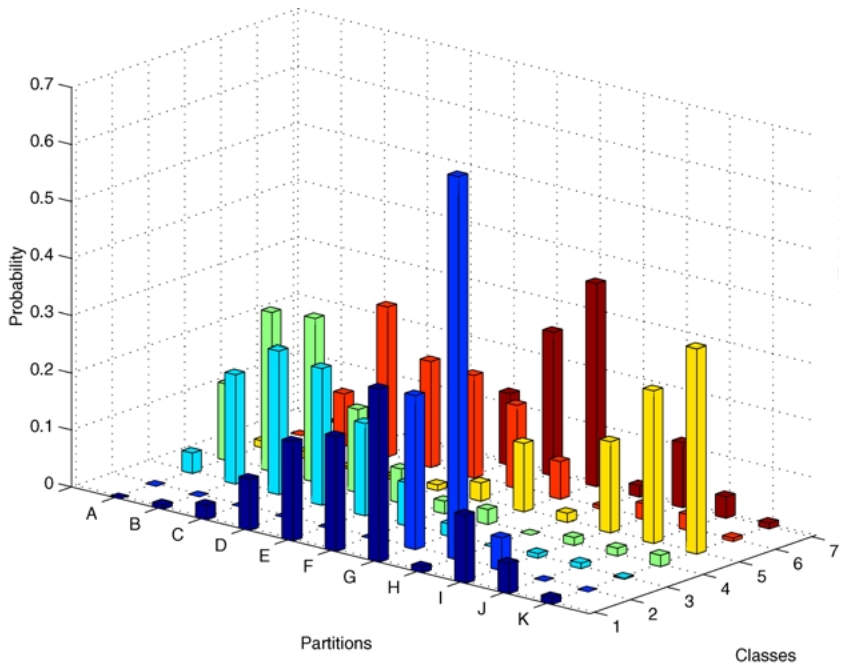

Figure 7. Probability distribution of classes over color-feature partitions.

is marked by the partitions A to K while the classes (1 to 7) are marked on the Y-axis. Thus a class distribution along the partitions can be determined by viewing parallel to the X-axis from the mark of the class. Many noteworthy observations can be made from the figure. The first is that many partitions have one or at most two dominant classes. For example, partition H has only class 2 (i.e., Educational) with probability 0.672. Therefore definite distinction exists between the classes when we consider more than one dimension. Furthermore, in the next subsection, we would find that distinction and classification could be made even when two or more classes clash on similar partitions of more than one but not all features.

The second observation from figure 7 is that the relevance of a feature for each class is appropriately calculated based on the distribution of the class over the feature space. For a class like MTV (class 3), which has wide distribution, the color feature is not very relevant. This can be inferred as the highest partition probability (or sub-feature causal strength) is only 0.252 for partition C. Therefore any sub-feature of color would assign low-probability to MTV class. On the other hand class 2, i.e. Educational, has narrow distribution and therefore a hit to a particular sub-feature where it is present will result in high probability assignment. Thus for a class like Educational, color becomes a meaningful feature.

Figure 8 depicts the probability computation during the inference phase. A video sequence belonging to Basketball (class 1) was given as an input to the system. The logarithm of the probabilities for each class are plotted against the number of features, where log(0) is treated as $\log(\epsilon)$, $\epsilon \ll 1$. With increase in the number of features, the distinction between class 1 and the other classes increases and the sequence is correctly classified as class 1. It can

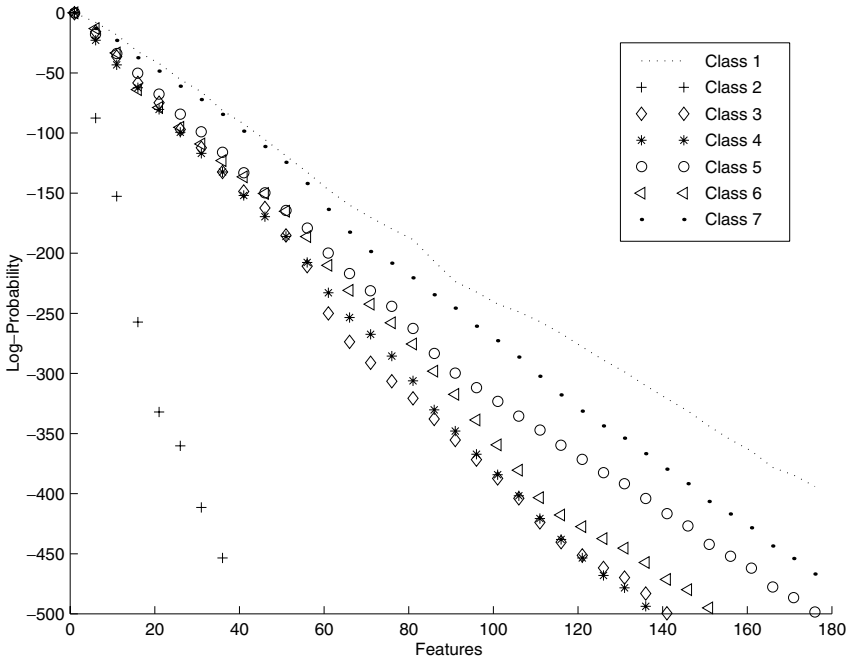

Figure 8. Probability computation with varying number of features.

be noted from this example that high dimensionality of feature space does not degrade the performance of the CBR Bayesian system. On the other hand, tools like Neural networks [12] and SVM [28] suffer from the curse of dimensionality.

5.3. Comparison

We would like to present a comparison of our approach with Artificial Neural Networks (ANN), decision trees, K-Nearest Neighbor classifier (KNN) and Support vector machines (SVM). The ideas of performing association and classification in content-based indexing is beginning to develop with the application of tools like Neural networks (for example, see Doulamis et al. [6]), decision trees (see Demsar and Solina [5]) and K-nearest neighbor classifier (see Yang and Kuo [36]). These works have different paradigms of operation than our CBR system in the sense that they do not envisage autonomous development of high level indices from the local feature extraction and the knowledge extraction processes as we do.

In ANN, a feedforward backpropagation network was used with one hidden layer consisting of 500 neurons keeping in consideration the generalization guidelines set in Haykin [12]. The training function used was gradient descent with momentum backpropagation and the transfer functions employed were {*tansig*, *logsig*} where *tansig* and *logsig* are the hyperbolic tangent sigmoid transfer function and the log sigmoid transfer function respectively. Some of the most well known decision tree algorithms are C4.5 [27] and its improved successor C5.[1] We chose C5 decision tree package for the purpose of comparison since it has many nice features like accurate and fast rule sets, and fuzzy thresholding. A refinement to KNN algorithm called distance weighted nearest neighbor algorithm [18] is used in which the contribution of each of the k neighbors are weighed according to their distance to the query point, giving greater weight to closer neighbors. The application of SVM to domain of more than two target classes is still in the development phase; nevertheless, we use the LIBSVM[2] package [3] where the iterative process is performed by treating one class as $+1$ and the others as -1, thereby getting $|V|$ SVM models.

The performance of each of these tools was evaluated on 1400 sequences. The recall rate, i.e., the proportion of relevant shots retrieved, and the precision rate, i.e., the proportion of retrieved shots that are relevant, are shown in Table 2. The recall rate of MBN (Modified Bayesian Network) is highest (89.14%) while KNN stands second in recall with 81.79%. The reasonably good performance of KNN hints that there are clusters in higher dimensional feature space; however, MBN performance is better as the boundary samples are often

Table 2. Recall rate and precision rate.

Tool	Recall (%)	Precision (%)
MBN	89.14	90.44
DT	79.29	80.10
KNN	81.79	83.63
SVM	76.86	91.07
ANN	42.14	79.27

misclassified in KNN. The precision rate of SVM is highest with 91.07% while MBN is second best with 90.44%.

The individual class errors can be further seen from Table 3–7, which show the confusion matrices for the tools. It can be seen that the educational class is most characterized by these

Table 3. Confusion matrix using MBN.

Actual class	Output class						
	1	2	3	4	5	6	7
1	172	0	13	0	5	1	9
2	0	200	0	0	0	0	0
3	2	0	182	0	6	7	3
4	1	0	24	166	1	0	8
5	4	0	2	0	187	1	6
6	6	0	32	0	0	159	3
7	0	0	7	0	10	1	182

Table 4. Confusion matrix using ANN.

Actual class	Output class						
	1	2	3	4	5	6	7
1	0	1	0	0	109	90	0
2	0	200	0	0	0	0	0
3	0	12	0	0	34	152	0
4	0	24	0	0	11	165	0
5	0	0	0	0	192	8	0
6	0	0	0	0	2	198	0
7	0	0	0	0	102	98	0

Table 5. Confusion matrix using DT.

Actual class	Output class						
	1	2	3	4	5	6	7
1	148	0	2	2	0	44	4
2	0	198	0	0	2	0	0
3	0	4	158	10	6	6	16
4	6	0	12	144	4	32	2
5	4	4	2	0	166	2	22
6	30	0	22	2	0	144	2
7	0	0	26	0	22	0	152

Table 6. Confusion matrix using KNN.

Actual class	Output class						
	1	2	3	4	5	6	7
1	152	3	9	0	0	13	23
2	0	200	0	0	0	0	0
3	2	4	146	0	1	41	6
4	0	2	8	165	0	20	5
5	11	22	0	0	148	4	15
6	0	0	35	2	0	159	4
7	3	4	0	1	2	15	175

Table 7. Confusion matrix using SVM.

Actual class	Output class						
	1	2	3	4	5	6	7
1	139	0	1	0	60	0	0
2	0	200	0	0	0	0	0
3	0	0	143	0	57	0	0
4	0	0	0	147	53	0	0
5	0	0	0	0	200	0	0
6	0	0	0	0	70	130	0
7	0	0	0	0	83	0	117

features and thus most of the shots are classified correctly. Actually, there are only few settings (like those depicting speaker, screen, students etc.) which are possible in educational videos. Table 3 depicts the performance of MBN. One significant observation is that most of the misclassified shots were assigned class 3 (MTV), which has large distribution in the feature space (especially with relation to color and motion), thus leading to misclassifications. The ANN was successful in training for only a few classes, and similarly SVM also shows a bias towards assigning class 5 as 523 video shots were classified as class 5. On the other hand, the errors in DT and KNN were more distributed.

6. Conclusion and future work

This paper seeks to bring out some inherent problems with existing content-based retrieval systems. In essence, these systems are far away from the user's space and expectation. For CBR systems to be pragmatic and serviceable to the common user, semantic level indices should be made available to the user. We have presented a framework by which various syntactic features can be synthesized through the medium of Bayesian Network. From such

synthesis high-level indices may emerge. The Bayesian Network associates each clique of data points in the multi-dimensional feature space to one of the classes during training which can later be used for evaluating the closest class.

The principal contribution of this paper is in systematically bringing out the concept of integration of features to give semantic level indices. Our framework can efficiently work with a large number of classes and hundreds of features and yet only the meaningful features would be extracted for a class. There is no need for normalization of features nor is there any need of expert knowledge base.

Our work merely forms the foundation of a functional CBR system. Many additional steps are required. We have only formed a two level Bayesian Network implying a direct coupling between the features and the semantic classes. The presence of other intermediate structures like the temporal context of a shot, or the probable emotion evoked by a shot etc. would be essential to enhance the performance of the system. In addition, a number of indicative low-level features, such as the effects of different types of camera motion, need to be identified and integrated. However, the framework we have presented here would continue to be a building block of future CBR systems.

Appendix: Algorithm to evaluate most probable hypothesis

The algorithm for finding the most probable hypothesis using pruning technique is as follows.

Let *MPH* be the most probable hypothesis examined so far, i.e., it has probability greater than any other yet examined.*Hlist* is the list of hypotheses that are candidates for extension, that is, those for which probability has been computed but whose extensions have not been examined. $EXTN\_H(H)$ is the list of classes that are the admissible extensions to the hypothesis H. They are initialized as follows:

$MPH = H_\phi$
$Hlist = \{ H_\phi \}$
$EXTN\_H(H_\phi) = \{ v \mid v \text{ is parent of } a_{j_k}, a_{j_k} \in A^{\tau}_{j_k} \}$

```
procedure search(CBR Net)
{
while(Hlist is not empty)
  {
    remove H from Hlist;
    for each class in EXTN_H(H)
       {
         compute MEP(v,H);
         if MEP < 1
             remove v from EXTN_H(H)
          else
           {
             for each extension H+ = H U v
                {
                     P(H+, A) = MEP(v,H) * P(H,A)
                     if P(H+, A) > P(MPH, A)
                            MPH:=H+
```

```
                    insert H+ in HList
                    set EXTN_H(H+)= EXTN_H(H)- v
                }
            }
        }
    }
output MPH;
}
```

Acknowledgments

Heartfelt thanks are due to Dr. P.V. Krishnan, I.I.T. Delhi, who gave the motivation for the work. Thanks are also due to CIT Multimedia Lab, NUS for providing the facility for digitizing the videos, and to R. Ramesh for valuable comments and his time.

Notes

1. http://www.rulequest.com/see5-unix.html
2. http://eewww.eng.ohio-state.edu/˜ maj/osu_11_svm/

References

1. Y.A. Aslandogan and C.T. Yu, "Techniques and systems for image and video retrieval," IEEE Transactions on Knowledge and Data Engineering, Vol. 11, No. 1, pp. 56–63, 1999.
2. M.L. Cascia and E. Ardizzone, "JACOB: Just a content-based query system for video databases," in IEEE Int. Conf. on Acoustics, Speech and Signal Processing, May 1996.
3. C.C. Chang and C.-J. Lin, "LIBSVM: Introduction and benchmarks," in Tech. Report, CS Deptt., NTU, Taiwan. http://www.csie.ntu.edu.tw/ c̃jlin/libsvm/, 2000.
4. G.F. Cooper, "Probabilistic inference using belief networks is NP-hard," Technical Report, KSL-87-27, Stanford University.
5. J. Demsar and F. Solina, "Using machine learning for content-based image retrieving," International Conference on Pattern Recognition, 1996, Vol. 3.
6. N.D. Doulamis, A.D. Doulamis, and S.D. Kollias, "A neural network approach to interactive content-based retrieval of video databases," International Conference on Image Processing, 1999, Vol. 2.
7. A.M. Ferman and A.M. Tekalp, "Probabilistic analysis and extraction of video content," in Proc. Of ICIP, 1999, Vol. 2, pp. 91–95.
8. S. Fischer, R. Lienhart, and W. Effelsberg, "Automatic recognition of film genres," in ACM Multimedia 95—Electronic Proceedings, San Francisco, California, Nov. 1995.
9. M. Flickner et al., "Query by image and video Content: The QBIC system," *IEEE Computer*, pp. 23–32, Sept. 1995.
10. V.N. Gudivada and V.V. Raghavan, "Content-based image retrieval systems," IEEE Computer, Sept. '95.
11. A. Hampapur, "Designing video data management systems," in Ph.D. Thesis, The University of Michigan, 1995.
12. S. Haykin, "Neural network: A comprehensive foundation," 2nd ed., pp. 178–210, 1999.
13. M. Henrion, "Towards efficient probabilistic diagnosis in multiply connected belief networks," Influence Diagrams, Belief Nets and Decision Analysis, pp. 385–410, 1990.
14. A.K. Jain, A. Vailaya, and X. Wei, "Query by video clip," Multimedia Systems, pp. 369–384, 1999.

15. P.M. Kelly, T.M. Cannon, and D.R. Hush, "Query by image example: The CANDID approach," in Proc. of the SPIE, Storage and Retrieval for Image and Video Databases III, Vol. 2420, pp. 238–248, 1995.
16. S.L. Lauritzen and D.J. Spiegelhalter, "Local computations with probabilities on graphical structures and their applications to expert systems," J. Royal Statistical Society, pp. 157–224, 1988.
17. S. Lendis, "Content-based image retrieval systems project," http://www.tc.cornell.edu/Visualization/Education/cs718/fall1995/landis/.
18. T.M. Mitchell, "Instance-based learning," in Machine Learning, McGraw-Hill, pp. 230–248, 1997.
19. F. Nack and A. Parkes, "The application of video semantics and theme representation in automated video editing," Multimedia Tools and Applications, pp. 57–83, 1997.
20. M.R. Naphade, T. Kristjansson, B. Frey, and T.S. Huang, "Probabilistic multimedia objects (multijects): A novel approach to video indexing and retrieval in multimedia systems," in Proc. of ICIP, 1998, pp. 536–540.
21. V.E. Ogle and M. Stonebraker, "CHABOT: Retrieval from a relational database of images," IEEE Computer, pp. 40–48, September 1995.
22. K. Otsuji and Y. Tonomura, "Projection-detecting filter for video cut detection," Multimedia Systems, Vol. 1, pp. 205–210, 1994.
23. M. Pazzani, "An interative improvement approach for the discretization of numeric attributes in bayesian classifiers," International Conference on Knowledge Discovery and Data Mining (KDD), pp. 228–233, 1995.
24. M. Pazzani, C. Merz, K. Ali, and T. Hume, "Reducing misclassification costs," International Conference on Machine Learning, 1994.
25. J. Pearl, "Probabilistic Reasoning in Intelligent Systems," Morgan Kaufmann, 1988.
26. Y. Peng and J.A. Reggia, "Abductive Inference Models for Diagnostic Problem-Solving," Springer-Verlag, 1990.
27. J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, 1993.
28. S. Raudys, "How good are support vector machines?" Neural Networks, Vol. 13, pp. 17–19, 2000.
29. X.J. Shannon, M.J. Black, S. Minneman, and D. Kimber, "Analysis of gesture and action in technical talks for video indexing," IEEE Int. Conf. on Computer Vision and Pattern Recognition, pp. 595–601, 1997.
30. J.R. Smith and S.F. Chang, "VisualSEEK: A Fully Automated Content-Based Image Query System, ACM Multimedia, Nov. 1996.
31. V. Sobchack, "Toward inhabited spce: The semiotic structure of camera movement in the cinema," Semotica, pp. 317–335, 1982.
32. M.A. Stricker and M. Swain, "Bounds for the discrimination power of color indexing techiniques," in Proceedings SPIE Storage and Retreival for Image and Video Databases II, 1994, pp. 15–24.
33. G. Sudhir, C.M. Lee, and A.K. Jain, "Automatic classification of tennis video for high-level content-based retrieval," in IEEE Workshop on Content-Based Access of Image and Video Databases, 1998.
34. B.S. Todd, R. Stamper, and P. Machpherson, "A probabilistic rule-based expert system," International Journal of Biomedical computing, pp. 129–148, 1993.
35. N. Vasconcelos and A. Lipman, "Towards semantically meaningful feature spaces for the characterization of video content," in Proc. of Int. Conf. on Image Processing, 1997.
36. Z. Yang and C.C.J. Kuo, "A semantic classification and composite indexing approach to robust image retrieval," International Conference on Image Processing, Vol. 1, 1999.
37. D. Yow, B.L. Yeo, M. Yeung, and B. Liu, "Analysis and presentation of soccer highlights from digital video," in Second Asian Conf. on Computer Vision(ACCV '95), 1995.
38. R. Zabih, J. Miller, and K. Mai, "Video browsing using edges and motion," IEEE Int. Conf. on Computer Vision and Pattern Recognition, pp. 439–446, 1996.
39. H. Zhang, A. Kankanhalli, and S.W. Smoliar, "Automatic partitioning of full motion video," Multimedia Systems, Vol. 1, pp. 10–28, 1993.
40. H.J. Zhang, Y. Gong, S.W. Smoliar, and S.Y. Tan, "Automatic parsing of news video," in Proc. of Int. Conf. On Multimedia Computing and Systems, Boston, Massachusetts, USA, May 1994, pp. 45–54.
41. H.J. Zhang, C.Y. Low, S.W. Smoliar, and J.H. Wu, "Video parsing retrieval and browsing: An integrated and content based solution," in Proc. of Multimedia '95, San Francisco, CA, USA, 1995, pp. 15–24.

Ankush Mittal received his B.Tech. and Master (by Research) degree in Computer Science and Engineering from Indian Institute of Technology, Delhi. He was a Ph.D. scholar in the department of Computer and Electrical Engineering at the National University of Singapore. Prior to this he was working as Assistant Professor in Department of Computer Science, Birla Institute of Technology, India. His research interests are in Multimedia indexing, machine learning and motion analysis. Presently, he is a visiting lecturer in Computer Science department, National University of Singapore

L-F. Cheong was born in Singapore on June 1, 1965. He received the B.Eng. degree from the National University of Singapore, and the Ph.D. from University of Maryland at College Park, Center for Automation Research, in 1990 and 1996 respectively. In 1996, he joined the Department of Electrical and Computer Engineering, National University of Singapore, where he is an Assistant Professor now. His research interests are related to the basic processes in the perception of three-dimensional motion, shape and their relationship, as well as the application of these theoretical findings to specific problems in navigation and in multimedia systems, for instance, in the problems of video indexing in large databases.