8

Feature Based Digital Video Indexing

Arun Hampapur
Virage Inc
9605 Fcranton Road, Suite 240, San Diego,CA92121,U.S.A
Email: arun@virage.com, Phone 619-587-4080, Fax 619-587-4071
Ramesh Jain
Department of Electrical and Computer Engineering
University of California at San Diego, La Jolla, CA 92093-0407
Email: jain@ece.ucsd.edu, Phone 619-534-8639
Terry E Weymouth
Department of Electrical Engineering and Computer Science
University of Michigan, 1101 Beal Ave, Ann Arbor, MI 48109-2110
Email: weymouth@eecs.umich.edu, Phone 313-763-6985, Fax 313-763-1260

Abstract

Indexing video data is essential for providing content based access. Indexing has typically been viewed either from a manual annotation perspective or from an image sequence processing perspective. This work proposes a methodology for designing video indexing schemes which use low level *machine derivable* indices to map into the set of application specific *desired video indices*. The indexing procedure uses image sequence processing and application requirements analysis to arrive at the low level and desired indices. The mapping is created based on the domain constraints. A mapping efficacy measure is presented. Experimental results of indexing video using *image motion features* are presented.

Keywords: Digital Video, Video Indexing, Video Databases, Content based retrieval, partial index, Video Processing, Image Sequence Analysis, Video Classification, Video Production

1 INTRODUCTION

Digital Video technology promises to make video the ubiquitous mode of communication in the future. There are several technological hurdles that need to be crossed before these promises are met. One of the major hurdles is the ability to access video based on its content. This requires content based indices into the video data. The process of attaching content based labels to video is referred to as *video annotation or indexing*. This process can be performed manually by an operator who views the video. The indexing effort is directly proportional to the granularity of video access. For example, the indexing effort for video library applications which model video by its title is lesser

S. Spaccapietra et al. (eds.), *Visual Database Systems 3*

than the indexing effort for a multimedia authoring application which indexes video based on the content and style of the shots used to compose the video. Thus as applications demand finer grain access to video, automation of the indexing process becomes essential. Given the current state of art in computer vision, pattern recognition and image processing, reliable and efficient automation is possible for low level video indices like scene changes and image motion properties etc. But most applications demand a much higher level of content based access. This paper explores the use of low level image measurements to derive higher level content based indices with the help of domain constraints.

The proposed approach begins to tackle the problem of content based access in a systematic way. This work defines the problem of video indexing and proposes a design methodology for video indexing schemes. The indexing schemes presented use low level image sequence features in a feature based classification formalism to arrive at a machine derived index. Domain constraints are used to design a mapping from low level machine derived index to the desired video index. An efficacy measure is proposed to evaluate this mapping. An example feature based indexing scheme based on the image motion feature to generate a shot framing (Bordwell, 1980) video index is presented.

The organization of the paper is as follows. A short literature survey is presented in section 2. Section 3 discusses the problem of video indexing and provides a formal definition. The design procedure for feature based video indexing is discussed in Section 4. Section 5 presents the example design for a video database to support authoring and editing applications. The mapping between the machine derived index and the desired video index is presented in section 6. The results of applying indexing scheme to cable television feed are presented in section 7. A summary of the work concludes the paper in section 8.

2 LITERATURE REVIEW

Existing work on content based video access and video indexing can be grouped into three main categories.

High Level Indexing: The work by Davis (Davis 1993, Davis 1994) is an excellent instance of high level indexing. This approach uses a set of predefined index terms for annotating video. The index terms are organized based on a high level ontological categories like *action, time ,space, etc.* Davenport et al (Davenport, 1991) have proposed a structured model for a shot. Smith et al (Smith, 1992) have proposed a video annotation system based on a layered annotation approach.

The high level indexing techniques are primarily designed from the perspective of manual indexing or annotation. The index terms used ,data structures and user interfaces are geared towards a scenario where an operator views and indexes the video. This approach is suitable for dealing with small quantities of new video and for accessing previously annotated databases. Ignoring the need for automation in video insertion is one of the main limitations of these approaches.

Low Level Indexing: These techniques provide access to video based on properties like color, texture etc. These techniques can be classified under the label of low level indexing. Ioka et al

(Ioka, 1993) have presented techniques for retrieving video sequences based on the estimation of motion vectors in video sequences and using these vectors as the key for retrieving video. Gong et al (Gong,1994) have applied color based image indexing techniques to retrieving key frames of video sequences. Nagasaka et al (Nagasaka, 1991) present video search based on object appearances. Arman et al (Arman 1994) have presented a key frame based browsing technique for video. This approach uses image moments to measure key frame similarities. The key frames are used as models of the video shots.

The driving force behind this groups of techniques is to extract data features from the video data, organize the features based on some distance metric and to use similarity based matching to retrieve the video. Their primary limitation is the lack of semantics attached to the features. Hence from the user perspective, the utility of such approaches becomes limited.

Domain Specific Indexing: A number of researchers have worked on the areas of domain specific video indexing and retrieval. One of the pioneering efforts in the area is by Swanberg et al (Swanberg, 1992, 1993). They have presented work on finite state data models for content based parsing and retrieval of news video. Smoliar et al (Smoliar, 1994) have also presented work on parsing news video.

These techniques over come the limitations of both the above categories. They use the high level structure of video to constrain the low level video feature extraction and processing. These techniques are effective in their intended domain of application. The primary limitation of these techniques is their narrow range of applicability.

Apart from the above research efforts work by Akutsu et al (Akutsu, 1992, 1994) has considered the cinematographic structure of video and the low level video feature based techniques to extract some of the cinematographic properties. The work by Akutsu et al is a very balanced approach to the problem of video indexing. How ever they have not completely utilized the structuring that exits in video information.

All the above work has viewed video in isolation and has not considered video as a component of a video database system. The proposed approach views video indexing from a video database perspective while utilizing the structure inherent in video to derive the indices. Preliminary results from this research have been reported in (Hampapur, 1995-A).

3 VIDEO INDEXING

The term indexing as used in database literature (Date, 1975), (Korth, 1986) refers to the ordering of data based on a particular attribute. This is referred to as the *search key* . For example, a telephone directory is indexed on the name attribute using *lexicographic* ordering. Video indexing serves an analogous purpose in video databases. A video index requires *video search keys* based on which the index is generated. The process of choosing search keys and designing representations for video data based on the content and the typical queries is called video data modeling. This problem has been addressed in (Hampapur, 1995-B).

The concept of a video index is illustrated below using an example. Consider a television news cast. The typical structure of a television news cast (Zettl, 1984), (Millerson, 1975) involves a segment in which the headlines are presented by the news anchor. This is followed by introduction of

each individual report by the anchor followed by the actual report itself. The news concludes with a commentary on the news by the anchor. A video index into the news cast will provide *temporal intervals of video* with *associated descriptions* of the temporal interval. The following are examples of video indices into a newscast. Depending on the type of video and its content, video indices can be of many different types (Hampapur, 1995-B).

: { Temporal-Interval = News cast} with
: {Description = (date, anchor-person-name, lead-story-name, etc) }. (1)
: {Temporal-Interval =News report } with
: {Description= (story-titles, reporter-name, production-style, etc) } (2)
: { Temporal-Interval= Shots } with
: {Description= (transition-effects, cinematographic-properties, visual-properties) } (3)

With the example indices presented, the newscast video can be accessed in terms of news casts of a particular date, or news reports with a particular topic, video shots (Bordwell, 1980), (Hampapur, 1994-A), (Hampapur 1994-B), (Hampapur, 1995-C), video shots with specific cinematographic or visual properties, etc. There are many other possible video indices. The ones presented here are meant to illustrate the idea of a video index.

A feature in the context of this paper is a low level measurable property of the image sequence. Given any video, a feature based index into the video will provide *temporal intervals of video* with *associated feature based descriptions* of the temporal interval. The typical temporal intervals for a feature based index are continuous image sequences or shots. The following are some typical feature based indices.

: { Temporal-Interval=Shot } with
: { dominant-color = blue, dominant-texture = random
: motion-type = local, flow-direction= left to right, etc } (4)

A feature based index describes the video in terms of low level measurable properties of the image sequence. The feature based descriptions of video are domain independent and can be extracted from any video and are independent of the domain or type of the video.

The focus of this paper is on how to generate the desired video indices (equation 3) based on the feature based indices (equation 4) which can be automatically extracted from the video. The paper has addressed the issues of how to arrive at a mapping between feature based indices and video indices, how to quantitatively evaluate the mapping between the two sets of indices. A motion based index is used to map into a cinematraphic property based video index.

3.1 The Digital Video Model

The model used for video data depends on the purpose for which the data is being used. The design of video data models for database applications has been addressed in (Hampapur, 1995-C), (Jain 1995). Swanberg et al (Swanberg, 1992), (Swanberg 1993) model video in terms of the temporal events in the content of the video. They use a finite state machine for representing this model. Hampapur et al (Hampapur, 1995-B) have modeled video in terms of the *editing* process of video. They

use the model for video segmentation. The video model used in the indexing process is based on (Hampapur, 1995-C), (Jain 1994). The structure of the model is shown in equation 5.

$$\begin{aligned} \mathcal{V} &: \text{Video Interval: } [t_b, t_e] \\ &: \text{Temporal Relations: } \mathcal{R} \\ &: \text{Feature Count: } n \\ &: \text{Type: } (\omega_0, \omega_1, \ldots, \omega_n) \in \Omega \\ &: \text{Features: } (\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \ldots \mathcal{F}_n) \end{aligned} \tag{5}$$

Video Interval: t_b, t_e represent the begining and end of the temporal extent of the video respectively. All the other elements of the video model refer to this time interval.

Temporal Relation: $\mathcal{R} = ((r_1, \mathcal{V}_1), (r_2, \mathcal{V}_2), \ldots (r_k, \mathcal{V}_k))$ is a set of k temporal relationships. r_i is one of the thirteen possible relationships between time intervals listed in (Allen, 1983). Thus every video model can maintain temporal relationships to k other video models.

Feature Count: n is the number of features used to describe the video segment being modeled.

Feature Type: ω_i is the feature type for feature $\mathcal{F}_i$. The different possible types of features are discussed in (Hampapur, 1995-C), (Jain 1995).

Features: $(\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3, \ldots \mathcal{F}_n)$ these are the different features that are used to describe the video being modeled. The actual structure of the features is discussed in (Hampapur, 1995-C), (Jain 1995). The number of features used in a video model and the type depend on the exact nature of the application. Any feature $\mathcal{F}_i$ takes on a label l where $l \in \mathcal{L}_i = (l_{i1}, l_{i2}, ...l_{ik})$. Thus the features are qualitative in nature and take labels form a predefined finite set.

3.2 Definition of Video Indexing

Existing literature on video indexing implicitly defines video indexing as *the process of extracting from the video data the temporal location of a feature and its value.* A similar but more explicit for of this definition is adopted in this work.

Video Indexing: Given a video data model $\mathcal{V}$ and a video interval $v = [t_b, t_e]$

$$\forall_{i=0}^{n} \mathcal{F}_i \text{ assign } l_{ji} \in \mathcal{L}_i \text{ based on } v \tag{6}$$

where n is the number of features in the video data model, $\mathcal{F}_i$ is the i^{th} feature, $\mathcal{L}_i$ is the set of labels that P_i can take.

Video Indexing as Classification Given the above definition of indexing, the problem of indexing video data can be reformulated as a problem of classifying a particular interval of video into one of the predefined categories. This reformulation is possible since the features in the video data model take only a finite set of values. The classification problem is defined as follows. Given a video interval $v = [t_b, t_e]$ and a label set $\mathcal{L} = (l_1, l_2, \ldots, l_k)$

$$\text{Assign to } v \leftarrow l_i \text{ where } l_i \in \mathcal{L} \tag{7}$$

3.3 Definition: Partial Video Index

Let $\mathcal{I}_1$ be an index on $\mathcal{VDB}$ based on a video model $\mathcal{V}_1$ and $\mathcal{I}_2$ be another index on $\mathcal{VDB}$ based on a video model $\mathcal{V}_2$. Then $\mathcal{I}_2$ is called a partial index on $\mathcal{VDB}$ with reference to $\mathcal{I}_1$ if $\mathcal{V}_2 < \mathcal{V}_1$ where the ordering of video models is based on some linear criteria. The criteria for ordering video models is presented in section 4.1. Here $\mathcal{V}_1$ is referred to as the *complete video data model* and $\mathcal{V}_2$ is referred to as the *partial video data model.* The labels in the complete video data model are called the *complete label set* and the labels of the partial video model are called the *partial label set*

The idea of a partial index is motivated by the fact that *using machine vision techniques it is possible to assign to a video shot a* ***machine derived label*** *which covers a group of* ***complete labels.*** The *machine derived* label set is a partial label set with reference to the *complete label set.* For example, if the complete labels set included labels like *Human-Subject-Motion: (Left to Right, Right to Left)* the machine derived label set could be *Object-Motion: (Left-to-Right, Right-to-Left).* A machine derived label like *object-motion* will cover the label *Human-Subject-Motion* however it may also include other objects like *vehicles.*

The reason machine derived labels are at a lower level of abstraction is due to the fact that it is fairly straight forward and computationally inexpensive to design a reliable analysis technique which can extract labels like *direction of object motion* automatically. However recognizing a human figure in an arbitrary video sequence and estimating its direction of motion is currently not within the reach of most machine vision techniques. Thus using automatic analysis to derive a partial index is a feasible approach to automating the indexing procedures.

A video data record with a partial index provides lesser information about the video data than the same record with a complete video index. Since a record with partial index will have fewer features and the total number of labels among these features will be smaller than that for a complete video index. A quantitative measure of the effectiveness of a partial video index is provided in section 4.1. There are several advantages of the partial index process including *feasibility, constrained processing cost* and *ability to incorporate techniques into prototype designs.*

4 DESIGNING FEATURE BASED INDEXING SCHEMES

This section presents the detailed procedure for designing feature based indexing schemes. A schematic of the steps involved is presented in figure 1. Each of the steps in the design procedure is discussed below:

User Specification: These are the inputs to the design process. The two main inputs are the *purpose of the video database* and the *computational constraints* in terms of the amount of computational resources available. These factors dictate the complexity of the design that can be adopted, the structures associated with the video data, the physical storage to be used and several other system parameters.

Design Steps: These are the actual steps in the design process. They include the design of the video data model, feature section, feature based class design and feature based classifier design.

Data Model Design: This step typically requires the purpose of the video data base, database expertise and content expertise (knowledge about the content of the video data and the

application). The design of the video data model is driven by the application. For example, the video data model for a video library will be entirely different from the data model for a database to support video editing.

Primary Feature Selection: The primary feature is defined as a measurement made directly from the video data. For example, the difference image between two consecutive frames of video is a primary feature, while the average difference pixel value is not. The computational constraints of the user along with computer vision or image processing expertise are needed to make the feature choice. Typically the initial design iterations will result in the choice of simple features like difference images, while later design iterations will result in the choice of more complex features to perform the feature based indexing.

Secondary Feature Selection: Secondary features are derived from some combination of primary and secondary features. Typical examples would be average value of the difference image, variance of the difference image, filtered difference images, etc. The choice of the secondary feature requires the use of vision expertise and the video data model.

Feature Based Model Design: This is a model similar in structure to the video data model, but the instantiation of this model is based on the automatic processing of the video data. The typical procedure for arriving at the feature based model is to design video data classes based on a qualitative classification. The qualitative classification process relies on the secondary features that are extracted.

Feature Based Classifier Design: (Duda, 1975) This step involves the actual design of the algorithms and code for the feature based classification process. This step relies on the vision and software engineering expertise available.

Design Evaluation Step: This is one of the key steps in the design process, and provides a comparison of the feature based model against the video data model. This measure of design efficacy should be linear and monotonic, i.e. if the feature based data model exactly matches the video data model the measure of goodness (called fbi efficacy) should be one and should decrease linearly to zero as the mis-match between the video data model and feature based model increases. One such measure is proposed in section 4.1.

Design Outputs: These are the outputs of the design process.

FBI Efficacy: This is a measure of goodness of the design and is a number which lies between (0,1) with 0 indicating an ineffective feature based indexing scheme, and 1 indicating a perfect feature based indexing mechanism.

Feature Extractors: These are the functions for extracting measurements from the video data. Typical examples include, difference images, flow, dominant colors, color distribution maps, texture maps, etc.

Discriminant Function: This is the function which combines the output of the various features to arrive at a feature based class for a given video interval.

The flow diagram of figure 1 shows the steps involved in designing a feature based indexing system. The next step is to apply the feature based indexing to actual data in the application domain

and to evaluate the experimental performance of the system. Typically this will lead to refined user specifications and a redesign of the *indexing scheme.*

4.1 Efficacy of feature based indexing

This section presents a measure which evaluates the effectiveness of the design. The problem of evaluating the effectiveness of the feature based model as compared to the video data model can be mapped into a problem of rating the *goodness of the mapping between the classes in the feature based model and the video data model.* Let $\mathcal{V}$ be a video data model. Let $\mathcal{V}$ have a single feature $\mathcal{F}$. Let $\mathcal{L}$ be the set of labels that can be assigned to the feature $\mathcal{F}$. Let $\mathcal{V}_f$ be the feature based video model. Let $\mathcal{V}_f$ have one feature $\mathcal{F}_f$ which can take on labels from $\mathcal{L}_f$. The problem of evaluating the goodness of the mapping between $\mathcal{V}_f, \mathcal{V}$ can now be defined as the problem of evaluating the mapping between $\mathcal{L}_f, \mathcal{L}$. Figure 2 shows the range of mapping between the two sets. There are three mappings shown in the figure (top, middle, bottom). The number in the left bottom corner for each mapping indicates its ranking. The set (oval) on the left indicates the feature based index (automatically derived partial index), and the oval on the right represents the ideal or desired index. The mapping shows how the machine derived index relates to the desired or ideal index. The top mapping (figure 2) has an efficacy measure of 1.0, indicating that it is perfect. The middle (figure 2) mapping has an intermediate value for the efficacy measure, since here the feature based index provides a sub grouping on the ideal index. The bottom mapping (figure 2) is the worst, since the feature based index does not provide any information about the ideal index.

A measure which behaves in this manner is used to characterize the goodness of a feature based indexing scheme. Such a measure can be designed by ranking the mappings between two sets. A *one to one* and *onto* mapping is given the highest ranking and a mapping in which one element in the feature based index covers all elements in the ideal index receives the lowest rank. In the following derivation $Card$ stands for cardinality of a set, i.e. the number of elements in the set.

$$Card(\mathcal{L}_f) = k \tag{8}$$

$$Card(\mathcal{L}) = n \tag{9}$$

Each element in $\mathcal{L}_f$ can be treated as a subset of $\mathcal{L}$. Let S_i be a subset of $\mathcal{L}$ where

$$S_i = (l \in \mathcal{L} : l^{-1} = l_f \in \mathcal{L}_f) \tag{10}$$

$\mathcal{L}$ has k subsets where $k = Card(\mathcal{L}_f)$.

$$\bigcap_{i=0}^{i=k} S_i = \emptyset \text{ and } \bigcup_{i=0}^{i=k} S_i \neq \mathcal{L} \tag{11}$$

The sets S_i have a null intersection but don not necessarily constitute a partition of $\mathcal{L}$.

$$\text{Let } \mathcal{O} = \{l \in \mathcal{L} \ : \forall_{i=0}^{i=k} S_i \ l \notin S_i \} \tag{12}$$

$\mathcal{O}$ is the set of omitted labels in $\mathcal{L}$. Let $o = Card(\mathcal{O})$. Let $\mathcal{C}$ be the set of covered labels in $\mathcal{L}$. Let $c = Card(\mathcal{C})$. Now

$$\mathcal{C} = \mathcal{L} - \mathcal{O} \text{ and } Card(\mathcal{C}) \ + \ Card(\mathcal{O}) \ = \ Card(\mathcal{L}) \ \ c + o = n \tag{13}$$

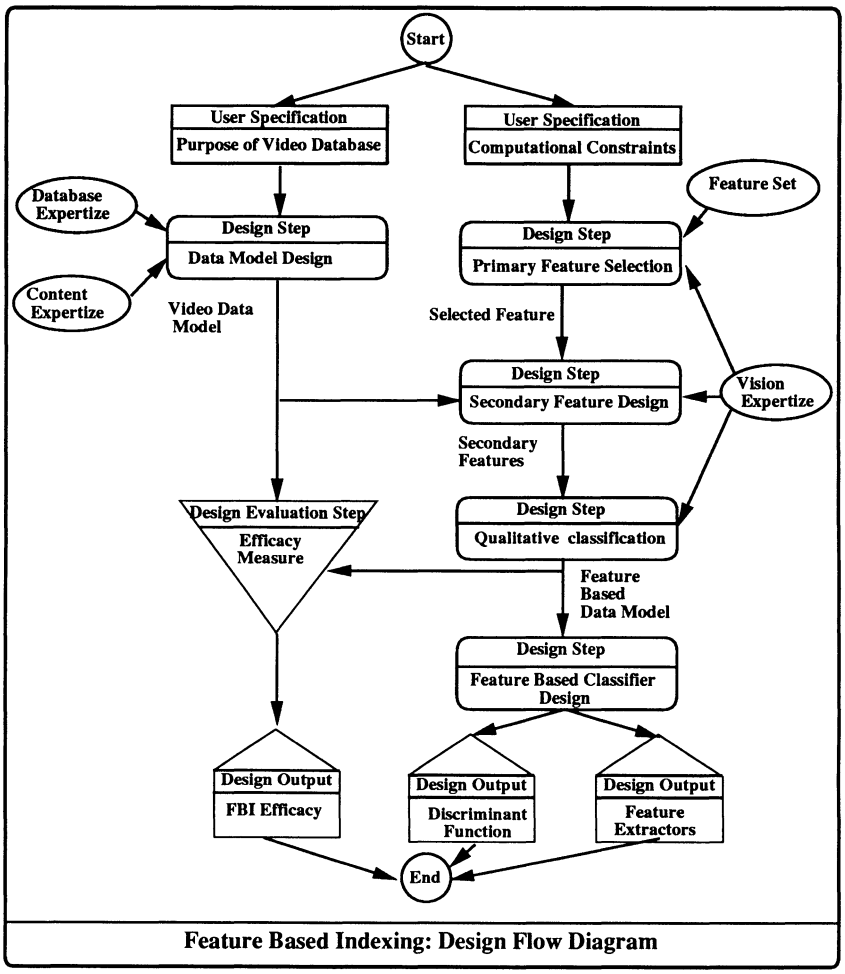

Figure 1: Feature based Indexing: Design Procedure

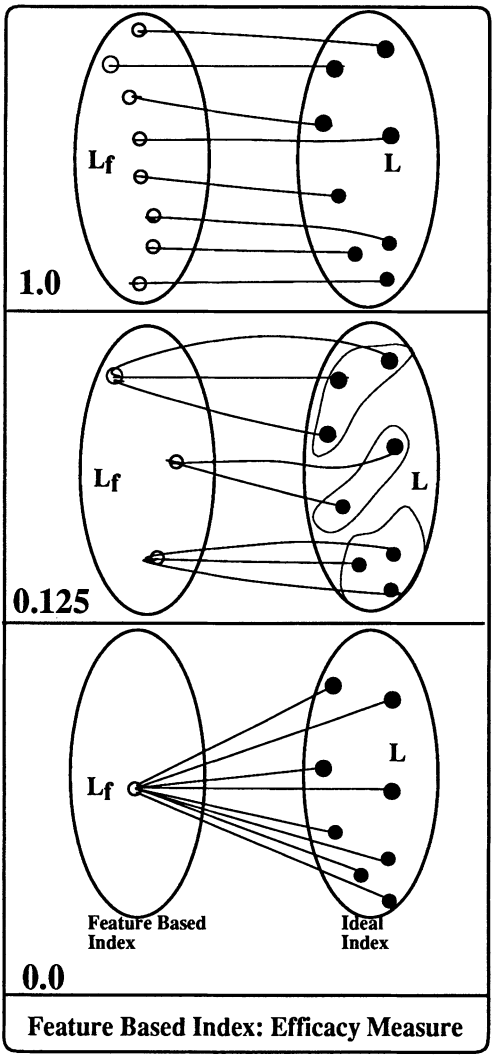

Figure 2: Feature based Indexing: Efficacy Measure

Let f be the fraction of labels covered in $\mathcal{L}$. $f = \frac{c}{n}$. f is a measure of coverage of $\mathcal{L}$ by $\mathcal{L}_f$ but does not measure the discrimination provided by the feature based data model. The measure of discrimination is the nature of the distribution of the members of C among the sets S_i. Such a measurement is provided by the variance of the cardinalities of S_i.

$$\forall_{i=0}^{i=k} H(i) = Card(S_i) \tag{14}$$

Let σ^2 be the variance of H and $\sigma^2_{\max}$ be the maximum variance of H.

$$\sigma^2 = \frac{\sum_{i=1}^{i=k} \left(Card(S_i) - \frac{c}{k}\right)^2}{n} \tag{15}$$

$$\sigma^2_{\max} = \frac{c(k-1)}{k} \tag{16}$$

$$\text{Let } \delta = \frac{\sigma^2}{\sigma^2_{\max}} \tag{17}$$

δ is a measure of the uniformity of H. $\delta = 0$ for a perfectly uniform distribution and $\delta = 1$ for a distribution in which $\mathcal{L}$ is covered by one label in $\mathcal{L}_f$. In addition to the uniformity of the distribution, for the ideal case the mapping between $\mathcal{L}_f$ and $\mathcal{L}$ is one-to-one and onto. For such a mapping $c = k$, the mean of H is a good measure for this $\mu = \frac{c}{k}$. Let ε represent the efficacy measure of the mapping.

$$\varepsilon = \frac{(1-\delta)\frac{c}{n}}{\frac{c}{k}} = \frac{k(1-\delta)}{n} \tag{18}$$

5 EXAMPLE DESIGN

This section presents an example of the design process illustrated in section 4. The example design presented here emphasizes the Feature based model design step. Further details of the design process can be found in (Hampapur, 1995-C). The presentation in this section lists the name of the step with reference to figure 4, presents a brief discussion and presents the output of the step.

User Specification: Purpose The purpose of the video database in this example is assumed to be a database to support video editing, multimedia authoring, etc, where video is typically reused from earlier footage.

Design Step: Data Model Design The system specification that can be derived from the purpose are listed below:

Access Granularity: Editing applications typically access video at the granularity of shots (continuous camera operation image sequences).

Typical Query Patterns: The typical nature of queries will include content queries and production style queries.

The video is assumed to be segmented into shots. There are a number of techniques available for segmenting video into shots (Hampapur, 1995-B) The current design example will limit

queries to production style queries, specifically to shot framing queries (Zettl, 1984). This simplification is done to keep the presentation simple. The video data model is:

$$\begin{aligned} \mathcal{V}_{ap} &: \text{Video Interval: } \textit{Shot} \\ &: \text{Temporal Relations: } \emptyset \\ &: \text{Feature Count: } 2 \\ &: \text{Type: } (\omega_0, \omega_1) \\ &: \text{Features: } (\mathcal{F}_0, \mathcal{F}_1) \\ &: \text{where } \mathcal{F}_0 = \text{Cinematographic Label Set, } \mathcal{F}_1 = \text{Content Labels} \end{aligned} \tag{19}$$

Feature	Type	Range
$\mathcal{F}_{00}$ = Framing Distance	ω_0 = Qualitative	$\mathcal{L}_{00}$ = Long, Medium, Close Up
$\mathcal{F}_{01}$ = Framing Angle	ω_1 = Qualitative	$\mathcal{L}_{01}$ = High, Eye, Low
$\mathcal{F}_{02}$ = Framing Motion	ω_2 = Qualitative	$\mathcal{L}_{02}$ = Tracking,Panning $\mathcal{L}_{02}$ = Object,Camera-Static,Null
$\mathcal{F}_{03}$ = Shot Purpose	ω_3 = Qualitative	$\mathcal{L}_{03}$ = Establishing, Zoom, $\mathcal{L}_{03}$ = Tracking, Conversation

Table 1: Complete Video Model: Features and Labels

The different features used in the above model of video are the common terminology used to describe a shot in film literature (Zettl, 1984). Thus given a video database constructed based on this model of video, it can be searched based on attributes like *shot distance, shot angle.*

Design Step: Primary Feature Selection This step relies heavily on the vision expertise available. Given the shot framing model of video the primary feature was chosen to be ***Difference Image Sequence (Jain, 1979) (Jain 1984).***

Design Step: Secondary Feature Design : This step requires a vision expertise and the study of the video production process and a study of experimental video shots. Based on all the above factors the following were chosen to be the set to secondary features.

Thresholded Difference Image Sequence: This image is generated by thresholding the difference image. The threshold is chosen to be a small value which eliminates difference pixels generated due to camera optics, digitization effects and small variations in lighting.

Area of Thresholded Difference Images: This is the number of pixels above the selected difference image threshold.

Connected Component Difference Images: The thresholded difference image is filtered and a connected component analysis is used to obtain a grouped component representation of the difference image.

Component Metrics: Different component measurements like size, shape, locations within the image etc are measured from the component image.

Design Step: Feature Based Data Model Design: Figure 3 shows a tree representation of the feature based data model used. The feature based model was designed based on an organization of the secondary features to yield a good classification of the data. The feature based data model is presented below. The process of computing the feature based model is presented in figure 4

$$\begin{aligned} \mathcal{V}_p \ &: \text{Video Shots: } [t_b, t_e] \\ &: \text{Temporal Relations: } \emptyset \\ &: \text{Feature Count: } 3 \\ &: \text{Type: } (\omega_0, \omega_1, \omega_2) \\ &: \text{Features: } (\mathcal{F}_0, \mathcal{F}_1, \mathcal{F}_2) \end{aligned} \tag{20}$$

Feature	Type	Range
$\mathcal{F}_0$ = Temporal Motion Variation	ω_0 = Qualitative	$\mathcal{L}_0$ = Homo, Hetero
$\mathcal{F}_1$ = Cumulative Motion Magnitude	ω_1 = Qualitative	$\mathcal{L}_1$ = Motion, Still
$\mathcal{F}_2$ = Spatial Motion Distribution	ω_2 = Qualitative	$\mathcal{L}_2$ = Local, Global

Table 2: Partial Video Index Features

The definition of the various labels in the feature based data model is provided below.

Homogeneous Shots (HO): A shot which has uniform motion properties along most of its temporal extent. For example, a panning shot, or a tracking shot, etc.

Heterogeneous Shots (HT): A single shot in which different temporal regions have distinct properties, or that the motion properties of the shot change over the duration of the shot. Typically such shots are called *long takes*. A long take could initially be a still close up shot, then zoom out to a medium shot, and end as a tracking long shot all within one camera take.

Motion Shots (MS): Shots which incorporate a significant amount of image motion.

Still Shots (SS): Shots which incorporate a negligible amount of image motion.

Localized Motion shots (LM): Shots in which the motion is localized to some parts of the image space of the shot. Such shots correspond to static camera moving object shots.

Global Motion Shots (GM): Shots in which the motion is significant in all portions of the image space.

Figure 4 shows the procedure used to compute the feature based index. The first step is to extract difference images from the video sequence. The difference images are thresholded

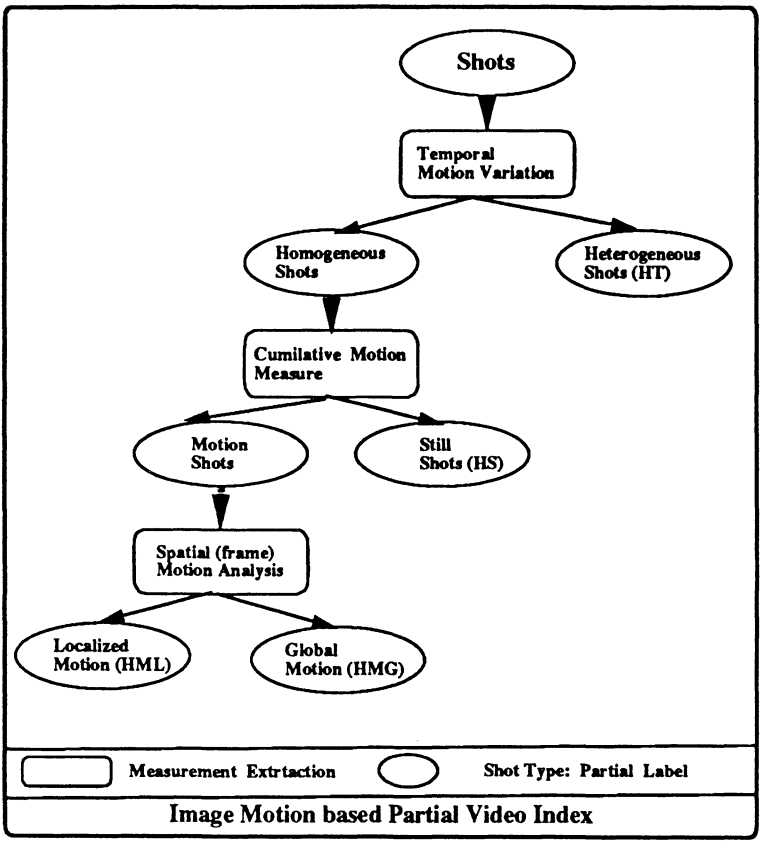

Figure 3: Difference Image based partial video data model

to eliminate noise. The thresholded difference images are grouped into components using a standard connected component labeling algorithm. The first decision in the tree (figure 3) is made based on the *variance of the total image motion area* over the length of the shot. This separates out *homogeneous* from heterogeneous shots. The grouping of *still* and *motion* shots is based on the thresholding the total motion area within a frame and using the frames to vote for the classification. The grouping into *local* and *global* motion shots is based on the *average component size* of the components in a frame.

6 MAPPING BETWEEN VIDEO MODELS

Given a partial video model $\mathcal{V}_p$ (equation 20) and a complete video model $\mathcal{V}$ (equation 19) a mapping between the two sets is necessary in order to use the partial video data model in the indexing procedure. Such a mapping between the *image motion based partial index* and the *shot framing based index* is presented in figure 5. The derivation of this mapping is based on two factors:

Partial Indexing Techniques: A thorough knowledge of the techniques used to derive the partial index and the meaning of the labels of the partial index is necessary in order to be able to create the mapping.

Video Content and Production: As the video models are domain specific it is necessary to have a good understanding of the content of the video and the techniques used to produce the video data.

Given that the application domain is *multimedia authoring and video production* the knowledge necessary to create this mapping is a knowledge of machine vision techniques (Jain-1995) and a knowledge of film production. The study of film and video production presented in (Hampapur, 1995-C) is used as a basis for arriving at this mapping. The mapping shown in figure 5 is based on the following:

HMG: Homogeneous global motion shots cover panning shots and establishment shots. Typically panning shots and establishment shots are used to provide the viewer with a panoramic view of the scene. Hence the camera is panned which causes homogeneous motion through the extent of the image.

HS: Homogeneous still shots typically cover still shots, shots of very small motion like medium shots of people talking etc.

HML: Homogeneous local motion shots cover medium shots with static camera. This is because a static camera precludes global image motion unless the objects are very close to the camera.

HT: Heterogeneous shots cover shots in which the framing parameters change over the length of the shot. For example, if a shot changes from long shot to close up, or if the camera begins motion in the middle of the shot, etc.

7 EXPERIMENTS

The feature based indexing scheme described in this paper, uses a set of low level labels derived by video processing algorithms as indicators of higher level features that are actually required by the video model. There are two factors that need to be evaluated in this approach:

Validity of Mapping: This is the problem of evaluating how faithfully does a particular machine derived label indicate or map to a desired label. This factor depends to a very great extent on the nature of the video data. Characterizing the mapping between the feature based video labels and the higher level of video features requires an extensive analysis of video data. This requires a classification of video based on different production styles and assessing the mapping for each of these production styles. This paper does not report on these experiments.

Reliability of Machine derived Index: This factor pertains to how well the low level video processing algorithms are able to extract the video classes. This is dependent on two factors, namely, the design of low level classes and the performance of the feature based classification scheme. The reliability of the feature based index can be measured in terms correctness of the labels derived by the video processing algorithms as compared to a manually assigned label set. This result can be expressed as a confusion matrix of the feature based classes. The experimental procedure and results are presented in the remainder of the section.

7.1 Procedure

The experimental procedure followed for evaluating the reliability of the machine derived index is presented in this section. A brief description of each of all of the steps involved are presented. Further discussions of some of the steps are presented in later sections.

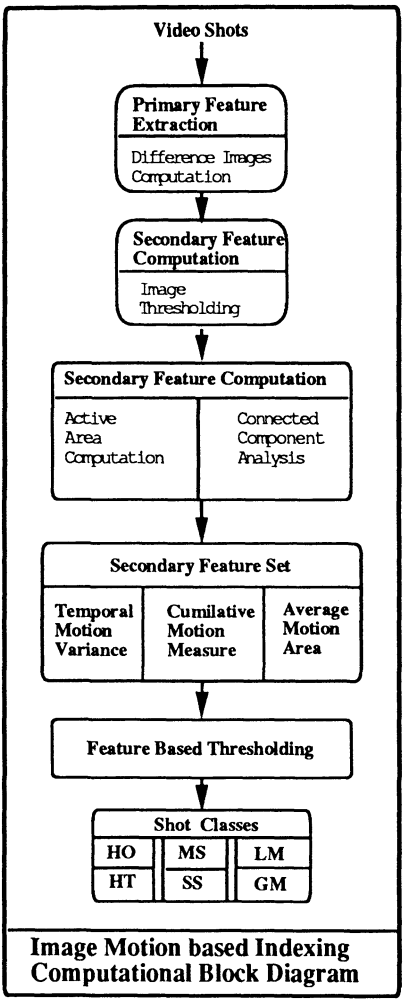

Figure 4: Image Motion based Indexing: Computational Block Diagram

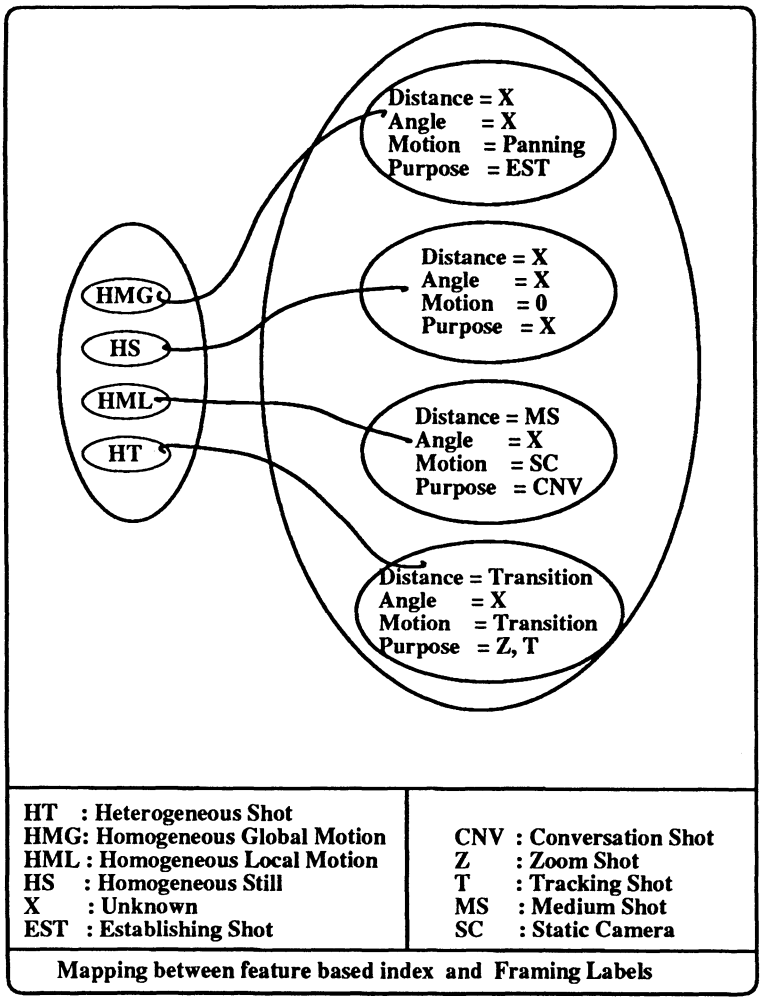

Figure 5: Mapping between machine derived features and video model features

Experimental Data: The experimental data used was taken from commercial cable television feed here in Ann Arbor, MI. The data included a wide variety of cable television programming including, news videos, comedy shows, sitcoms, music videos, commercials, sports telecasts, etc. The data is stored on a video disk.

Shot Segmentation: The video data taken from cable television was segmented into shots. The segmentation was performed using the video segmentation techniques presented in (Hampapur, 1995-C). Each shot of video is represented as a time interval (begining and ending frame numbers on the video disk).

Training Set: This is the set of shots used to tune or train the feature based indexing algorithm. The set chosen here must be representative of all the feature based classes. Further discussion of the training set is presented in section 7.2

Threshold Selection or Training: This is the process of selecting the various thresholds used by the feature based labeling algorithm. This procedure for training is presented in section 7.2

Experimental Set: This is the set of shots on which the the feature based labeling algorithms are applied. This set is chosen randomly from the set of experimental shots that is available.

Feature based labels: These are the labels that are derived by applying the feature based labeling to the experimental data set.

Manual labels: Manual labels are derived by viewing each shot manually and assigning a label. The rules for manual labeling are presented in section 7.3.

Confusion Matrix Computation: The confusion matrix is a $n \times n$ matrix. n is the number of feature based classes. The diagonal entries in the matrix indicate the percentage of shots that were correctly classified. The off-diagonal entries indicate the percentage of misclassification performed by the feature based indexing algorithm. The ideal confusion matrix would be an identity matrix.

7.2 Training

The process of training or threshold selection involves the selection of a set of shots called the *training set*, and using these shots as a basis for choosing the thresholds for the feature based indexing algorithm. The algorithm processes the video frames to extract from it the difference image between consecutive frames. The difference images pass through several computational steps and are finally converted into connected component measurements . These measurements are compared to thresholds in a sequential manner to arrive at the final classification of the shots. Training is the process of choosing these thresholds based on example data. The set of shots used for training are presented in figure 6. The training procedure for selecting the thresholds is outlined below.

1. To each shot in the training set assign a correct label (manual label) based on viewing the shot using the procedure outlined in section 7.3.

2. Run the feature based classification algorithm on all the shots in the training set and record the features.

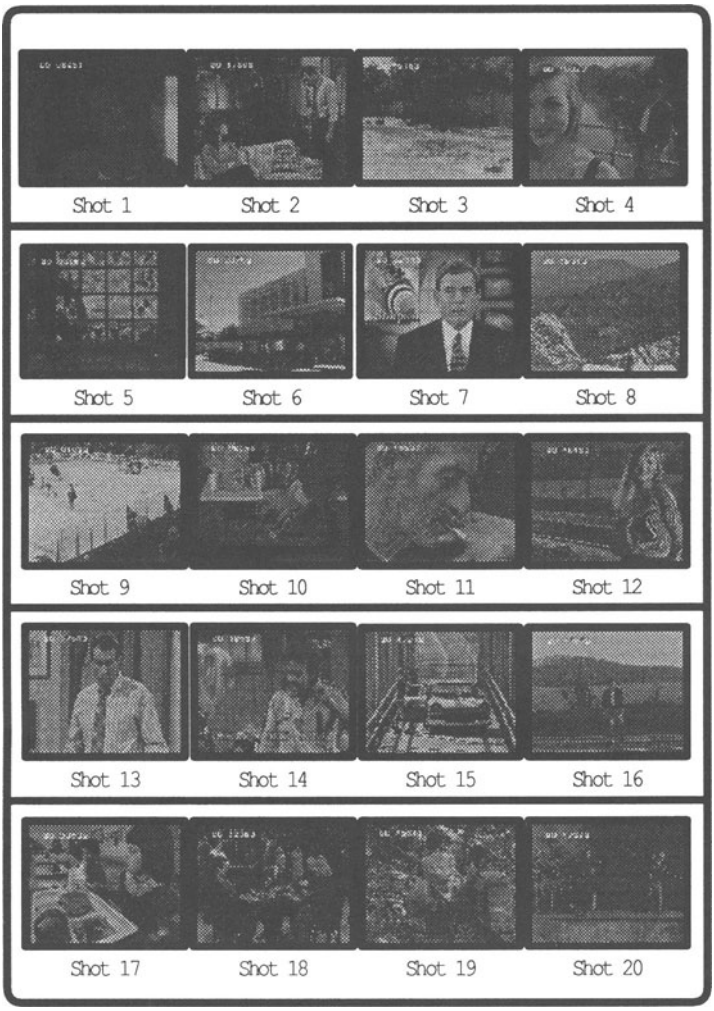

Figure 6: Training set for feature based indexing experiments

3. Choose the thresholds for each of the features which gives the best result in the training set. There are several automatic techniques for choosing these thresholds (Duda, 1975). In the experiments reported here the thresholds were manually chosen.

Figure 6 shows the first frames of each of the shots in the training set. Table 3 shows the actual cinematographic labels for each of the shots in the training set. Also shown are the manual labels assigned to each shot and the corresponding machine derived labels. The confusion matrix for the training set is presented in table 4. Given the set of shots used the confusion matrix of table 4 was the best result that could be achieved by appropriate choice of thresholds.

7.3 Ground Truth Derivation

This section presents the rules or the procedure to be followed by a human operator to label the shots with the feature based labels. The operator views each shot individually and assigns it a label based on the set of rules presented below.

Heterogeneous Shots: If the motion within the shot changes significantly over time, the shot would be termed heterogeneous. The amount of motion within the shot should be significant. Typically heterogeneous shots tend to be shots of long duration. Homogeneous shots exhibit a uniform motion behavior through the temporal extent of the shot.

Still Shots: These are shots with minimum motion in them, they typically include complete stills and conversation shots in front of a static camera, like an anchor person shot.

Local Motion Shots: These are shots where the motion is localized to different parts of the image. Typically this includes static camera shots with single and multiple moving objects.

Global Motion Shots: These are shots where the motion is distributed uniformly over the entire frame of the image. Typically, these include camera panning shots and tracking shots.

7.4 Test Set and Results

The test set for the experiments presented consists of shots taken from cable television programming. These shots were chosen at random from the experimental data set. The set consisted of seventy shots. The shot labeling algorithm was applied to all the shots and the results were recorded. Each of the shots was also assigned a manual label based on the ground truth derivation procedure presented in section 7.3. Based on these results the confusion matrix for these shots is shown in table 5.

7.5 Interpretation of Results

The results presented for the test set of shots is interpreted in this section. The interpretation presented here attempts to analyze the source of the errors. There are two main sources of error that contribute to the total error in the confusion matrix of table 5:

Operator Error: This is the error that arises due to a mislabeling of the shots by the operator. The primary reasons for this error are

Column 1	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7
Shot Label	Distance	Angle	Motion	Objects	FBI Manual	FBI Machine
Shot 1	CU	Eye	SC	1 →2	HT	HoS
Shot 2	LS →MLS	Eye →High	SC	1	HT	HT
Shot 3	LS →CU	High	SC	1	HT	HT
Shot 4	MS →CU	Eye	SC	2	HT	HoS
Shot 5	MS	Eye	SC	1	HoS	HoS
Shot 6	LS	Low	SC	1	HoS	HoS
Shot 7	MS	Eye	SC	1	HoS	HoS
Shot 8	XLS	High	SC	1	HoS	HoS
Shot 9	LS	High	Panning	Multiple	HMG	HMG
Shot 10	CU →MS	X →Eye	Panning	1	HMG	HT
Shot 11	XCU	High	SC	1	HMG	HML
Shot 12	MS	High	Hand Held	1	HMG	HT
Shot 13	MS	Eye	SC	1	HML	HML
Shot 14	MS	Eye	SC	2	HML	HML
Shot 15	LS	Eye	SC	2	HML	HML
Shot 16	XLS	Eye	SC	1	HML	HoS
Shot 17	MS	High	SC	Multiple	HML	HoS
Shot 18	MS	High	SC	Multiple	HML	HML
Shot 19	MS	High	SC	Multiple	HML	HT
Shot 20	LS	Eye	SC	Multiple	HML	HML

CU: Close Up Shot.	HT: Heterogeneous Shot
MS: Medium Shot.	HoS: Homogeneous Still Shot
MLS: Medium Long Shot	HMG: Homogeneous Motion Global
LS: Long Shot	HML Homogeneous Motion Local
XLS: Extreme Long Shot	
Eye: Eye Level Shot.	
High: High Angle Shot	
Low: Low Angle Shot	
SC: Static Camera	
Panning: Camera pans	

Table 3: Manual Derived Index and Ground truth for feature based Index

Labels	Hetero	Still	Global	Local	Total
Hetero	2	2	0	0	4
Still	0	2	0	0	4
Global	2	0	1	1	4
Local	1	2	0	5	8

Labels	Hetero	Still	Global	Local
Hetero	50%	50%	0	0
Still	0	100%	0	0
Global	50%	0	25%	25%
Local	12.5%	25%	0	62.5%

Table 4: Confusion Matrix for Training Set

Labels	Hetero	Still	Global	Local	Total
Hetero	10	0	0	7	17
Still	0	12	0	3	15
Global	4	1	14	5	24
Local	2	4	0	21	27

Labels	Hetero	Still	Global	Local
Hetero	60%	0%	0	40
Still	0	80%	0	20
Global	16%	4%	60%	20%
Local	10%	15%	0	75%

Table 5: Confusion Matrix for Experimental Set

- Error in applying the classification rules to the data.
- Data items which lie close to class boundaries.
- Operator fatigue and other reasons.

In the results presented it is estimated that about 20 % of the error is due to misclassification by the operator.

Machine Classification Error: These are errors that arise due to the mislabeling of the shots by the algorithm. The primary sources of this error are:

- The different classes in the classification scheme are not separable based on the measurements being made.
- The choice of thresholds are not correct, as the training set may on be representative of the test data set.
- The features being used for classification are not appropriate for the classification task.

Specifically the confusion between the classes in table 5 arises because of the following reasons. The list presented below is itemized by the correct class name followed by the erroneous class name.

Heterogeneous →Local: The primary reason for heterogeneous motion shots being mislabeled as local motion shots is *the motion is not detected by the difference image operator* due to the fact that sequence has many areas which have similar or smooth intensity properties. This is the problem of non uniform distribution of motion information through the image.

Local →Still: The primary reason for local motion shots being classified as still shots is *the choice of thresholds for differentiating between motion and still shots.* These shots which are misclassified typically lie near the boundary between the still and motion shots.

Global →Heterogeneous: The primary reason for global motion shots being misclassified as heterogeneous shots is that *in some frames in the shot the difference image operator does not pick up the motion.* This causes a larger variance in the motion over the length of the shot causing it to be misclassified as a heterogeneous shot. The other reason for this misclassification is due to the fact that many of the shots manually labeled as global were *close up hand held camera shots* this causes the motion between frames to be jerky and hence causes the labeling as heterogeneous.

Global →Local: The primary reason for global shots being misclassified as local shots is that *the difference image operator does not pick up the motion.* In the case of the specific shot used in the experimental set were low illumination shots which caused the non response of the difference image operator.

Local →Still: The primary reason for local motion shots being misclassified as still shots is the *choice of threshold.* These shots lie close to the boundary between motion and still shots.

One of the reoccuring reasons for the error is the inadequacy of the motion measurement feature used. This problem can be remedied by using other motion meaurements like *optic flow.* However in many cases the motion information is not available through out the image, in such cases using a better motion measurement operator does not provide much gain. Other types of analysis which perform motion measurement based on large scale object structure will have to be applied. These techniques tend to be very computationally expensive.

The confusion between heterogeneous and other classes occurs mainly due to the fact that the temporal analysis of the shot is local. Using feature variations over longer temporal intervals can effectively reduce this confusion.

In summary, the results of the classification process can be improved by using more complex image analysis techniques. From the perspective of video processing algorithms this is a tradeoff that needs to be made depending on the nature of the application. Since the use of more complex image analysis techniques implies additional computational effort per frame. Given the voluminous nature of video data, a small increase in the computation per frame can result in a significant performance loss for the complete system, which could directly affect the usability of the system.

8 SUMMARY

A novel approach to feature based indexing of video is the focus of this paper. The paper presents a methodology for designing feature based indexing schemes. This method uses the *purpose of the video database* and the *computational constraints* to design an indexing scheme. The method yields a low level video feature based classification scheme, a mapping between the machine derived index and the desired index and an efficacy measure for the machine derived index. The motivations behind this approach to indexing are the limited amount of computation that can be invested per

frame of video and the in-feasibility of deriving high level labels based on automatic video processing techniques. An example of image motion based video indexing was presented. This indexing scheme was implemented and tested on video data taken from cable television feed.

9 REFERENCES

Arman , Depommier and Hsu and Chiu. (1994),Content-based Browsing of Video, Proceedings Second Annual ACM MultiMedia Conference.

Allen. (1983) Maintaining Knowledge about Temporal Intervals, Communications of the ACM.

Akutsu and Tonomura and Hashimoto and Ohba. (1992) Video indexing using motion vectors, Proceedings of SPIE: Visual Communications and Image.

Akutsu and Tonomura. (1994), Video Tomography: An efficient method for camerawork extraction and motion analysis, Proceedings Second Annual ACM MultiMedia Conference.

Bordwell and Kristin. (1980), Film Art: An Introduction, Addison-Wesley Publishing Company.

Date. (1975), An Introduction to Database Systems, The Systems Programming Series, Addison-Wesley Publishing Company.

Davis. (1993), Media Streams: An Iconic Visual Language for Video Annotation, IEEE Symposium on Visual Languages.

Davis (1994), Knowledge Representation for Video, Working Notes: Workshop on Indexing and Reuse in Multimedia Systems, American Association of Artificial Intelligence.

Duda and Hart. (1973), Pattern Classification and Scene Analysis, A Wiley-Interscience Publication, John Wiley and Sons.

Davenport, Smith and Pincever. (1991), Cinematic Primitives for Multimedia, IEEE Computer Graphics & Applications.

Gong, Zhang, Low, Smoliar and Chua. (1994), Indexing and Retrieving Images based on color features, Technical Report, Institute of Systems Science, National University of Singapore.

Hampapur, Jain and Weymouth. (1994-A) Digital Video Indexing in Multimedia Systems, Prooceed ings of the Workshop on Indexing and Reuse in Multimedia Systems, American Association of Artificial Intelligence.

Hampapur, Jain and Weymouth. (1994-B) Digital Video Segmentation, Prooceedings of the ACM conference on MultiMedia.

Hampapur, Jain and Weymouth. (1995-A) Indexing in Video Databases, IS & T/SPIE Symposium on Electronic Imaging Science & Technology.

Hampapur, Jain and Weymouth. (1995-B) Production Model based Digital Video Segmentation, Journal of Multimedia Tools and Applications.

Hampapur. (1995-C), Designing Video Data Management Systems, Doctoral Thesis, The University of Michigan, Ann Arbor.

Ioka, Kurokawa. (1993) Estimation of Motion Vectors and their application to scene retrieval, Technical Report, IBM Research, Tokyo Research Laboratory.

Jain and Nagel. (1979), On the analysis of accumulative difference pictures from image sequences of real world scenes, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 1, No 2.

Jain. (1984), Difference and Accumulative Difference Pictures in Dynamic Scene Analysis, Image and Vision Computing, Vol 2, No 2.

Jain and Hampapur. (1994), Metadata in Video Databases, Sigmod Record: Special Issue On Metadata For Digital Media.

Jain, Kasturi and Schunck. (1995) *To be published* Introduction to Machine Vision, McGraw Hill.

Konigsberg. (1989), The Complete Film Dictionary, Penguin Books.

Korth and Silberschatz. (1986), Database System Concepts, McGraw Hill Book Company.

Millerson. (9175), The technique of Television Production, Hastings House Publishers.

Nagasaka and Tanaka. (1991), Automatic Video Indexing and Full-Video Search for Object Appearances, 2nd Working Conference on Visual Database Systems.

Smith and Davenport. (1992), The Stratification System: A Design Environment for Random Access Video, Workshop on Networking and Operating System Support for Digital Audio and Video

Swanberg, Shu and Jain. (1992), Architecture of a Multimedia Information System for Content-Based Retrieval, Proceedings of the Audio Video Workshop.

Swanberg, Shu and Jain. (1993), Knowledge Guided Parsing in Video Databases, Electronic Imaging: Science and Technology.

Smoliar, Zhang and Wu. (1994) Using Frame Technology to Manage Video, Prooceedings of the Workshop on Indexing and Reuse in Multimedia Systems.

Zettl. (1984), Television Production Handbook, Wadsworth Publishing Co.

10 BIOGRAPHY

Arun Hampapur received his B.Eng in Electronics and Communication Eng from the University of Mysore, India in 1987. He received his Masters degree in Electronics and Control Eng from the Birla Institute of Technology and Science, Pilani, India in 1989. He received his M.S in Computer Eng from Louisiana State University in 1991 and Doctorate from the University of Michigan, Ann Arbor in 1995. His dissertation work is on the Design of Video

Data Management Systems. He is currently employed as Senior Software Engineer at Virage Inc, San Diego. His research interests include video databases, digital video processing, image databases, multimedia systems, computer vision, image processing and robot navigation systems.

Ramesh Jain is currently a Professor of Electrical and Computer Engineering at the University of California at San Diego. Before joining UCSD he was a Professor of Electrical Engineering and Computer Science at the University of Michigan, Ann Arbor. He has also been affiliated with Stanford University, IBM Almaden Research Labs, General Motors Research Labs, Wayne State University, University of Texas Austin. His research interests are in multimedia information systems, image databases, machine vision and intelligent systems. Ramesh is a fellow of IEEE, AAAI and Society of Photo-Optical Instrumentation Engineers. Currently he is the Editor-in-Chief of IEEE Multimedia and is on the editorial boards of Machine Vision Applications, Pattern Recognition and Image and Vision Computing. He received his Ph.D from IIT Kharagpur in 1975 and his B.E from Nagpur University in 1969.

Terry Weymouth received his dissertation in May of 1986 from the University of Massachusetts, where he was involved in computer vision research. Dr. Weymouth is an Senior Research Scientist in the Electrical Engineering and Computer Science department of the University of Michigan. He is a member of IEEE, ACM and AAAI. His research interests include video and image databases, computer vision, robot navigation, biomedical image processing and distributed collaboratory systems.