EVENT MINING IN MULTIMEDIA DATA WITH UNSTRUCTURED RESOURCES

THESIS SUBMITTED TO
MANONMANIAM SUNDARANAR UNIVERSITY
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE OF

DOCTOR OF PHILOSOPHY
in
COMPUTER APPLICATIONS

By

V.NARAYANI
Reg. No. 3666

Under the guidance of

DR. S.P.VICTOR
Associate Professor & Head, Department of Computer Science
St.Xavier's College, Palayamkottai, Tirunelveli

MANONMANIAM SUNDARANAR UNIVERSITY

TIRUNELVELI – 627 012, INDIA

JUNE 2011

DR. S.P.VICTOR
Associate Professor & Head
Department of Computer Science
St.Xavier's College, Palayamkottai,
Tirunelveli – 627 002.

CERTIFICATE

This thesis entitled "Event Mining in Multimedia Data with Unstructured Resources" submitted by Mrs. V. Narayani, (Reg. No. 3666) for the award of Degree of Doctor of Philosophy in Computer Applications of Manonmaniam Sundaranar University is a record of bonafide research work done by her and it has not been submitted for the award of any degree, diploma, associateship, fellowship of any University/Institution.

Place: Tirunelveli.

Date:

(Signature of the Guide)

V. NARAYANI M.C.A., M.Phil
Research Scholar (Reg. No. 3666)
Department of Computer Science
St.Xavier's College,Palayamkottai,
Tirunelveli -2.

DECLARATION

I hereby declare that the thesis entitled "Event Mining in Multimedia Data with Unstructured Resources" submitted by me for the Degree of Doctor of Philosophy in Computer Applications is the result of my original and independent research work carried out under the guidance of Dr.S.P.Victor, Associate Professor and Head, Department of Computer Science, St.Xavier's College, Tirunelveli and it has not been submitted for the award of any degree, diploma, associateship, fellowship of any University/Institution.

Place: Tirunelveli.

Date: (Signature of the candidate)

ACKNOWLEDGMENTS

First and foremost I would like to express my respect, love and gratitude towards **Lord Krishna** for showering blessings upon me to carry out this research work.

I would like to express my respect and gratitude wholeheartedly to **Rev. Dr. Alphonse Manickam, S.J.,** Principal, St.Xavier's College, Palayamkottai, Tirunelveli who has given me an opportunity to do my research work in this institution.

I wish to thank my guide **Dr.S.P.Victor,** Associate Professor and Head, Department of Computer Science, St.Xavier's College, Palayamkottai, Tirunelveli for his valuable guidance and his infinite patience. Working with him has been a great experience, both satisfying and rewarding. My due course is definitely entitled with his valuable suggestions.

It is a pleasure to express my gratitude wholeheartedly to all the Staff members of the Research Section, Manonmaniam Sundaranar University and Staff members in the Research Center, St.Xavier's College for their timely help in various occasions during my research tenure.

Last but not the least; I thank my husband Mr.S.Rajkumar M.E.,(Ph.D), my parents and brother for their love and warmth throughout the course of my life.

V.NARAYANI

ACKNOWLEDGMENTS

First and foremost I would like to express my respect, love and gratitude towards **Lord Krishna** for showering blessings upon me to carry out this research work.

I would like to express my respect and gratitude wholeheartedly to **Rev. Dr. Alphonse Manickam, S.J.,** Principal, St.Xavier's College, Palayamkottai, Tirunelveli who has given me an opportunity to do my research work in this institution.

I wish to thank my guide **Dr.S.P.Victor,** Associate Professor and Head, Department of Computer Science, St.Xavier's College, Palayamkottai, Tirunelveli for his valuable guidance and his infinite patience. Working with him has been a great experience, both satisfying and rewarding. My due course is definitely entitled with his valuable suggestions.

It is a pleasure to express my gratitude wholeheartedly to all the Staff members of the Research Section, Manonmaniam Sundaranar University and Staff members in the Research Center, St.Xavier's College for their timely help in various occasions during my research tenure.

Last but not the least; I thank my husband Mr.S.Rajkumar M.E.,(Ph.D), my parents and brother for their love and warmth throughout the course of my life.

V.NARAYANI

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

SYMBOL	DESCRIPTION
$\sum$	Summation
C	contained in /subset
f(n)	function defined interms of n
f(x,y)	bivariate function
m	slope of line
O	big –oh Asymptotic notation of a curve
E’	Event mining system
| |	Modulus / Absolute value
C_f	Categorization factor
C_l	Cluster factor
$\int$	integration
Φ	empty set
Є	belongs to
I(x,y)	Image at the position x,y cartesian coordinate
G*	Contrast transformation of an image
M	Mean value
$\int\int$	Double integration
η	Primary membership function
μ	Secondary membership function
δ	Transition function
e_0	Initial event
T	Time intervlas

CHAPTER 1

INTRODUCTION

In this chapter the basic concepts related to data mining such as data warehouse, data mart, multimedia mining, data preprocessing, data analysis are described.

1.1 DATA MINING AND DATAWAREHOUSING

Data Mining is a process of discovering various models, summaries, and derived values from a given collection of data. It is an iterative process within which improvement is defined by discovery, either through automatic or manual methods. It is the search for new, valuable, and nontrivial information in large volumes of data. The two primary goals of data mining are *prediction* and *description. Prediction* involves using some variables or fields in the data set to predict unknown or future values of other variables of interest. *Description* focuses on finding patterns describing the data that can be interpreted by humans.

The goals of prediction and description are achieved by using data-mining techniques, for the following primary data-mining tasks:

1. *Classification* - discovery of a predictive learning function that classifies a data item into one of several predefined classes.

2. *Regression* - discovery of a predictive learning function, which maps a data item to a real-value prediction variable.
3. *Clustering* - a common descriptive task in which one seeks to identify a finite set of categories or clusters to describe the data.
4. *Summarization* - descriptive task that involves methods for finding a compact description for a set or subset of data.
5. *Dependency Modelling* - finding a local model that describes significant dependencies between variables or between the values of a feature in a data set or in a part of a data set.
6. *Change and Deviation Detection* - discovering the most significant changes in the data set.

Data warehouse is a collection of integrated, subject-oriented databases designed to support the decision-support functions (DSF), where each unit of data is relevant to some moment [12] in time. It is a system for storing and delivering massive quantities of data. The primary goal of a data warehouse is to increase the "intelligence" of a decision process and the knowledge of the people involved in this process.

A data warehouse can be viewed as an organization's repository of data, set up to support strategic decision-making. The function of the data warehouse is to store the historical data of an organization in an integrated manner that reflects the various facets of the organization and business. The data in a warehouse are never updated but used only to respond to queries from end users who are generally decision-makers. In many instances, an organization may have several local or departmental data warehouses often called data marts. A *data mart* is a data warehouse that has been designed to meet the needs of a specific group of users. It may be large or small, depending on the subject area.

Two aspects of a data warehouse are most important for a better understanding of its design process: the first is the specific types of data stored in a data warehouse, and the second is the set of transformations used to prepare the data in the final form such that it is useful for decision making. A data warehouse includes the following categories of data, where the classification is accommodated to the time-dependent data sources:

1. Old detail data
2. Current detail data
3. Lightly summarized data
4. Highly summarized data
5. Metadata

To prepare these five types of elementary or derived data in a data warehouse, the fundamental types of data transformation are standardized. There are four main types of transformations, and each has its own characteristics:

1. *Simple transformations* - These transformations are the building blocks of all other more complex transformations. It includes

manipulation of data that is focused on one field at a time, without taking into account its values in related fields.

2. *Cleansing and scrubbing* - These transformations ensure consistent formatting and usage of a field, or of related groups of fields. This class of transformations also includes checks for valid values in a particular field, usually checking the range or choosing from an enumerated list.
3. *Integration* - This is a process of taking operational data from one or more sources and mapping it, field by field, onto a new data structure in the data warehouse. The common identifier problem is one of the most difficult integration issues in building a data warehouse.
4. *Aggregation and summarization* - These are methods of condensing instances of data found in the operational environment into fewer instances in the warehouse environment. Summarization is a simple addition of values along one or more data dimensions; Aggregation refers to the addition of different business elements into a common total; it is highly domain-dependent.

These transformations are the main reason why we prefer a warehouse as a source of data for a data-mining process. If the data warehouse is available, the preprocessing phase in data mining is reduced, sometimes even eliminated. This preparation of data is the most time-consuming phase.

A three-stage data-warehousing development process is summarized through the following basic steps:

1. *Modelling – It represents* the time to understand business processes, the information requirements of these processes, and the decisions that are currently made within processes.
2. *Building* - To establish requirements for tools that suit the types of decision support necessary for the targeted business process; to create a data model that helps further define information requirements; to decompose problems into data specifications and the actual data store, which represent either a data mart or a more comprehensive data warehouse.
3. *Deploying* - To implement, relatively early in the overall process, the nature of the data to be warehoused and the various business intelligence tools to be employed; to begin by training users.

Data mining represents one of the major applications for data warehousing, since the sole function of a data warehouse is to provide information to end users for decision support.

Data Warehousing Overview

The goal of Data Warehousing is to generate front-end analytics that will support business executives and operational managers.

The pre-Data Warehouse zone provides data for data warehousing. Data Warehouse designers determine which data contains business value for insertion. *OLTP* (online transaction processing) databases store operational data . *Metadata* ensures the purity and accuracy of data entering into the data lifecycle process. It also ensures that data has the right format and relevancy. The terminology used to describe Meta data is "data about data".

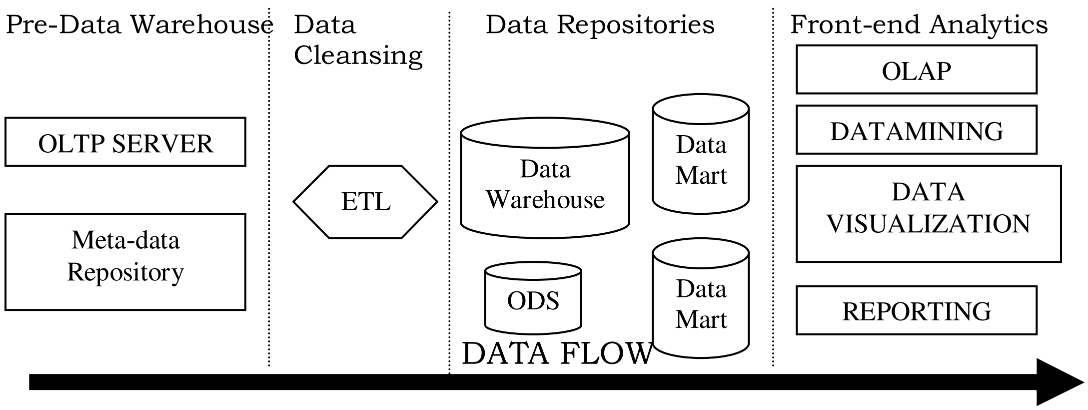


Figure 1.1: Overview of Data Warehousing Infrastructure

Before data enters the data warehouse, the extraction, transformation and cleaning (ETL) process ensures that the data passes the data quality threshold. ETLs are also responsible for running scheduled tasks that extract data from OLTPs.

Data Warehouse repository is the database that stores active data of business value for an organization. The Data Warehouse modelling design is optimized for data analysis. The variants of Data Warehouses are Data Marts and ODS (Operational Data Store). Data Marts are not physically different from Data Warehouses. Data Marts can be thought of as smaller Data Warehouses built on a departmental level rather than on a company-wide level. Data Warehouses collects data and is the repository for historical data. It is not always efficient for providing up-to-date analysis, therefore ODS are used to hold recent data before migration to the Data Warehouse. Large amounts of data in OLTPs can tie down computer resources and slow down processing.

Front-end applications are used by business users to interact with data stored in the repositories. *Data Mining* is the discovery of useful patterns in data. Data mining [25] are used for prediction analysis and classification. *OLAP*, OnLine Analytical Processing, is used to analyze historical data and slice the business information required. *Reporting* tools are used to provide reports on the data. Data are displayed to show relevancy to the business and keep track of key performance indicators (KPI). *Data Visualization* tools are used to display data from the data repository. Often data visualization is combined with Data Mining and OLAP tools. Data visualization can allow the user to manipulate data to show relevancy and patterns [24].

1.2 MULTIMEDIA MINING

Multimedia [12] Mining is a subfield of data mining that deals with an extraction of implicit knowledge, multimedia data relationships, or other patterns not explicitly stored in multimedia databases. Multimedia data refers to data such as text, numeric, images, video, audio, graphical, temporal, relational and categorical data.

Multimedia mining is a very broad and deep field, and it borrows elements from a variety of interdisciplinary fields, including the following:

- Data mining (e.g., image, video, text, and Web mining)
- Semantic Web
- Machine learning
- Speech recognition
- Computer vision
- Information filtering
- Information retrieval

Multimedia mining has three core concepts — representation, integration, and analysis. *Representation* is the process by which low-level data is converted into features and concepts. *Integration* is the process of bringing together data or information from multiple, varied sources. A*nalysis* is the process by which useful information and knowledge is extracted from the existing data or information. This process as applied to multimedia data is multimedia mining.

Data representation and analysis can be conducted without integration for single data sources, but it is usual for a particular application to be interested in data from multiple sources. The challenges when integrating data include cleaning data, converting data to a standard, and matching data on a single standard. Multimedia [17] data integration faces these challenges as well as additional challenges because of the size and diversity of data.

The Common activities that are considered as forms of data integration:

- Physical integration of multiple data feeds into one, in either storage, memory, or both.
- Coordination of data, or the standardization of data from multiple data standards and dictionaries into one standard.
- Use and integration of multiple ontologies, and data or information represented using those ontologies.
- Use and integration of syntactic and semantic [4] technologies, and data or information represented using these technologies.

These forms of integration are not unique to multimedia data, as additional challenges exist in applying these forms of integration to multimedia data because of the variety of structured, semi structured, and unstructured data [5] that is involved.

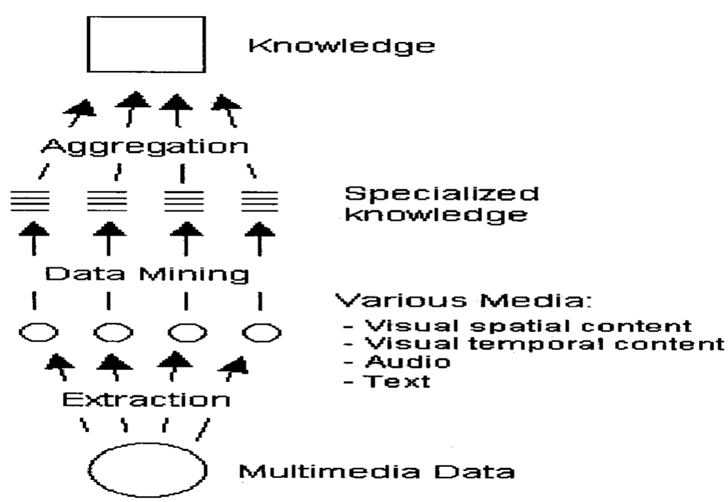


Figure 1.2: Multimedia data mining: an approach

Advances in digital [8] technologies make possible the generation and storage of large amounts of multimedia [17] objects such as images and video clips. Multimedia content contains rich information in various modalities such as image, audio, video frames, time series, etc. Advanced tools that can find characteristic patterns [24] and correlations in multimedia content are required for the effective exploitation of multimedia databases. In a video database that contains thousands of hours of video clips, users require functionalities such as content-based search or summarization to efficiently obtain the information they need. Finding characteristic patterns in various data modalities of video is essential to support these content-based functionalities. For example, to support efficient retrieval, continuous video clips need to be partitioned into segments of coherent content; for identifying relevant information, video segments need to be classified according to their content. Patterns that combine information from multiple data modalities also offer useful clues to understand the video content.

To make multimedia data accessible and useful, it is required to have advanced data mining tools to find patterns that are both meaningful and can support data mining tasks like segmentation, classification, and summarization. Interesting patterns exist in multimedia data sets other than video databases.

Finding patterns in multimedia data is a challenging problem. Finding patterns that are meaningful and understandable could benefit many data mining [25] applications like segmentation, classification, and rule discovery. The goal of segmentation, such as image segmentation, is to separate multimedia data into parts, such that each has a homogeneous characteristic. In classification, there is a need to differentiate object's different types, such as classification of video clips by their genres. Sometimes classification and segmentation are achieved at the same time: if labels of data are available from classification, then it is easy to identify the segmentation boundaries which separate data with different labels. For many multimedia applications, it is essential to discover features or patterns [24] that capture the content of an image or a video clip. A pattern discovery method is *data-independent/manual* if the method focuses on extracting the same set of patterns for every data set, and does not adapt to the properties of an individual data set. Usually, the patterns are manually designed based on knowledge on the data domain.

For finding patterns in multimedia [17] data such as images or video clips, a natural direction is to observe how people perceive and describe a multimedia object and design features that match human perception. One advantage of this approach is that it directly addresses the needs of human users of multimedia systems. There have been many manually defined features proposed for images and audio signals. These features are usually derived from psychological or human-computer-interface studies in human perception. For images, most of these features are based on the concepts of colour, texture and shape. Colour histogram is the most popular colour feature, with advanced variants such as colour moments and colour sets that improve the characterizations of the colour information. Other popular colour features include the colour correlogram and the colour structure, which combine the colour information with the layout information to provide better descriptions of the image content. Texture, such as coarseness, contrast, or directionality, is another visual cue people use to describe images. Shape information is also used in human perception. In situations where colour and texture information are not available, recognize objects based only on shape information. Many shape features have been proposed and the most popular ones are features based on the region covered by the shape, such as the area of the convex hull, the eccentricity, and the geometric moments [12]. Knowledge about human perception of colour patterns is used to design new features for image retrieval.

To achieve better performance in content-based retrieval, and the goal of understanding multimedia data, recent research has focused on finding mid-/high- level features that are associated with concepts that human used in perception and reasoning. Mining in visual database is quite different from standard alphanumeric mining. Feature vectors per image are computed for evaluation distance function on the feature space. This function is used to retrieve images from a given set. Images with distance less than a predefined threshold or within a predefined number are retrieved. These feature vectors facilitate mining by colour, texture, geometric properties, shape, volume, spatial constraints, etc.

1.3 STRUCTURED DATA

Data that resides in fixed fields within a record or file are structured data. Relational databases and spreadsheets are examples of structured data. Structured data [5] is data that is organized in a structure so that it is identifiable. The most universal form of structured data is a database like SQL or Access. SQL (Structured Query Language) allows selecting information based on columns and rows in a field. People use structured data every day. Structured data is anything that has an enforced composition to the atomic data types. Structured data is managed by technology that allows for querying and reporting against predetermined data types and understood relationships.

Data that is usually the source for a data mining process can be classified into structured data, semi-structured data, and unstructured data [5]. Most business databases contain structured data consisting of well-defined fields with numeric or alphanumeric values, while scientific databases may contain all three classes. Examples of semi-structured data are electronic images of business documents, medical reports, executive summaries, and repair manuals. The majority of web documents also fall in this category. Structured data is often referred to as traditional data, while the semi-structured and unstructured data are lumped together as non-traditional data, also called multimedia data.

Most of the current data-mining methods and commercial tools are applied to traditional data. The standard model of structured data [5] for data mining is a collection of cases. Potential measurements called features are specified, and these features are uniformly measured over many cases. Usually the representation of structured data for data-mining problems is in a tabular form, or in the form of a single relation, where columns are features of objects stored in a table and rows are values of these features for specific entities. A simplified graphical representation of a data set and its characteristics is given in Figure 1.3.

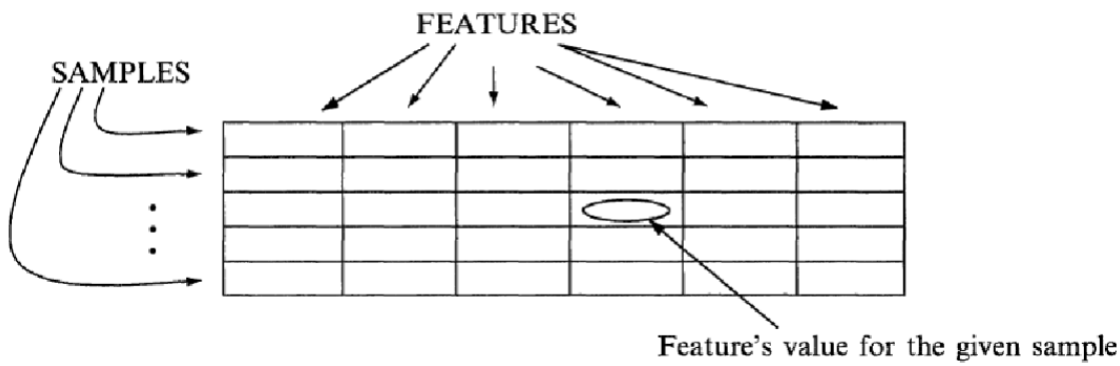


Figure 1.3: Tabular representation of a data set

In the data-mining literature, the terms samples or cases are used for rows. Many different types of features are common in data mining. Not all of the data-mining methods are equally good at dealing with different types of features. There are several ways of characterizing features. One way of looking at a feature in a formalization process is to see whether it is an *independent variable* or a *dependent variable;* i.e., whether or not it is a variable whose values depend upon values of other variables represented in a data set. This is a model-based approach to classifying variables. All dependent variables are accepted as outputs from the system for which we are establishing a model, and independent variables are inputs to the system, as represented in Figure 1.4.

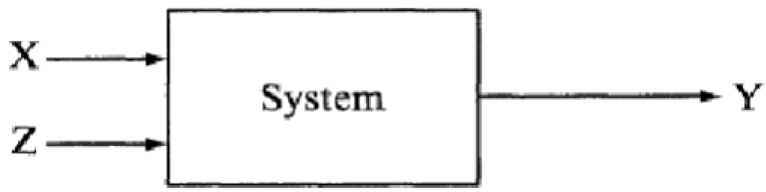


Figure 1.4: A real system, besides input variables X and outputs Y, often has unobserved inputs Z

There are some additional variables that influence system behaviour, but the corresponding values are not available in a data set during a modelling process. These are usually called unobserved variables, and they are the main cause of ambiguities and estimations in a model.

1.4 UNSTRUCTURED DATA

Data that does not reside in fixed locations are unstructured data. Examples are word processing documents, PDF files, e-mail messages, blogs and Web pages. Unstructured data [5] has no identifiable structure. Unstructured data typically includes bitmap images/objects, text and other data types that are not part of a database. It can be textual or non-textual. Textual unstructured data is generated in media like email messages, PowerPoint presentations, Word documents, collaboration software and instant messages. Non-textual unstructured data is generated in media like JPEG images, MP3 audio files and Flash video files. Some current technologies used for content searches on unstructured data require tagging entities such as names or applying keywords and meta tags. Therefore, human intervention is required to help make the unstructured data machine readable. People use unstructured data every day. They use it for creating, storing and retrieving reports, e-mails, spreadsheets and other types of documents. Unstructured data consists of any data stored in an unstructured format at an atomic level. An example of unstructured data is a video recorded by a surveillance camera in a department store. Such visual and multimedia recordings of events or processes of interest are currently gaining widespread popularity because of reduced hardware costs. This form of data generally requires extensive processing to extract and structure the information contained in it.

Two Categories of Unstructured Data

Unstructured data consists of two basic categories:

- Bitmap Objects: Inherently non-language based, such as image, video or audio files.
- Textual Objects: Based on a written or printed language, such as Microsoft Word documents, e-mails or Microsoft Excel spreadsheets.

Unstructured data refers to information that either does not have a pre-defined data model and/or does not fit well into relational tables. Unstructured information is naturally text-heavy but may contain data such as dates, numbers, and facts as well. These results in irregularities and ambiguities that make it difficult to understand using traditional computer programs as compared to data stored in fielded form in databases or annotated in documents.

The term is vague for several reasons; 1) structure, while not formally defined can still be implied and 2) data with some form of structure may still be characterized as unstructured if its structure is not helpful for the desired processing task, and 3) unstructured information might have some structure (semi-structured) or even be highly structured but in ways that are unanticipated or unannounced.

Software that creates machine-processable structure exploits the linguistic, auditory, and visual structure that is inherent in all forms of human communication. This inherent structure can be inferred from text, by examining word morphology, sentence syntax, and other small- and large-scale patterns. Unstructured information can then be enriched and tagged to address ambiguities and relevancy-based techniques then used to facilitate search and discovery. Examples of "unstructured data" may include books, journals, documents, metadata, health records, audio, video, log, files, and unstructured text such as the body of an e-mail message, Web page, or word processor document. The information contained in unstructured data is not always easy to locate. It requires that data in both electronic and hard copy documents and other media be scanned so a search application can parse out concepts based on words used in specific contexts. This is called semantic search.

1.5 DATA PREPROCESSING

Data pre-processing is an important step in the data mining process. The phrase "Garbage In, Garbage Out" is applicable to data mining and machine learning projects. Data gathering methods are loosely controlled, resulting in out-of-range values, impossible data combinations, missing values, etc. Analyzing data that has not been carefully screened for such problems can produce misleading results. Thus, the representation and quality of data is first and foremost before running an analysis.

If there is much irrelevant and redundant information present or noisy and unreliable data, then knowledge discovery [20][37] during the training phase is more difficult. Data preparation and filtering steps can take considerable amount of processing time. Data pre-processing includes cleaning, normalization, transformation, feature extraction and selection, etc. The product of data pre-processing is the final training set. Data are collected from the existing databases, data warehouses, and data marts. Data preprocessing includes two common tasks:

1. Outlier detection and removal: Outliers are unusual [42] data values that are not consistent with most observations. Outliers result from measurement errors, coding and recording errors, and, sometimes, are natural, abnormal values. Such non representative samples can seriously affect the model produced later. There are two strategies for dealing with outliers:
    a) Detect and eventually remove outliers as a part of the pre-processing phase, or
    b) Develop robust modelling methods that are insensitive to outliers.
2. Scaling, encoding, and selecting features: Data pre-processing includes several steps such as variable scaling and different types of encoding.

Data-pre-processing steps should not be considered completely independent from other data-mining phases. In data-mining process, all activities, together, could define new and improved data sets for subsequent iterations. A good pre-processing method provides an optimal representation for a data-mining technique by incorporating a priori knowledge in the form of application-specific scaling and encoding.

Data preprocessing describes any type of processing performed on raw data to prepare it for another processing procedure. There are a number of different tools and methods used for preprocessing, which includes:

1. sampling, which selects a representative subset from a large population of data;
2. transformation, which manipulates raw data to produce a single input;
3. denoising, which removes noise from data;
4. normalization, which organizes data for more efficient access; and
5. Feature extraction, which pulls out specified data that is significant in some particular context.

1.6 DATA ANALYSIS

Data analysis is a practice in which raw data is ordered and organized so that useful information can be extracted from it. The process of organizing and thinking about data is key to understanding what the data does and does not contain. There are a variety of ways in which people can approach data analysis, and it is easy to manipulate data during the analysis phase to push certain conclusions or agendas. Raw data can take a variety of forms, including measurements, survey responses, and observations.

Charts, graphs, and textual write ups of data are all forms of data analysis. These methods are designed to refine and distill the data so that readers can collect interesting information without the need to sort through all of the data on their own. Summarizing data is often critical to supporting arguments made with that data, as is presenting the data in a clear and understandable way. Data mining uses a relatively large amount of computing power operating on a large set of data to determine regularities and connections between data points. Algorithms that employ techniques from statistics, machine learning and pattern recognition are used to search large databases automatically. Data mining is also known as Knowledge-Discovery in Databases (KDD).

Data miners are statisticians who use techniques like near-neighbor models, k-means clustering [21], holdout method, k-fold cross validation, the leave-one-out method. Regression techniques are used to subtract irrelevant patterns [24], leaving only useful information. *Bayesian* is a technique that predicts the likelihood of future events by combining prior probabilities and probabilities based on conditional events. Decision trees are used to filter mountains of data. In a decision tree, all data passes through an entrance node, where it faces a filter that separates the data into streams depending on its characteristics. The data mining [25] process is a tool for uncovering patterns in a large amount of data. It involves five main steps, which include preparation, data exploration, model building, deployment, and review. Each step in the process involves a different set of techniques, but most of them use some form of statistical analysis.

Before the data mining process can begin, the researchers typically set research objectives. This preparation step usually determines what types of data need to be studied, what data mining techniques should be used, and what form the results will take. This initial step in the process may be essential to gather useful information. The next step in the data mining process is exploration. This step involves gathering the required data from an information warehouse or collection entity. Then, mining experts prepare the raw data sets for analysis. This step usually consists of gathering, cleaning, organizing, and checking all of the data for errors. This prepared data usually then enters the third step in the data mining process, model building. To accomplish this, researchers typically take small test samples of data and apply a variety of data mining techniques to them. The modelling step is often used to determine the best method of statistical analysis required to achieve the desired results. The final step in the data mining process is deployment. To do this, the techniques chosen in the model are applied to the larger data set, and the results are analyzed. The report that comes from this step usually shows the patterns found in the entire process, including any classifications, clusters, associations, or sequential patterns existing within the data set. Review is often an important final step. This phase in the process involves repeating mining models with a new data set to make sure that the main set was representative of the entire population of data. The results cannot predict trends in the larger population if the data sample does not accurately represent it.

Analysis of data is a process of inspecting, cleaning, transforming, and modelling data with the goal of highlighting useful information, suggesting conclusions, and supporting decision making. Data analysis has multiple facets and approaches, surrounding diverse techniques under a variety of names, in different business, science, and social science domains.

Data mining [25] is a data analysis technique that focuses on modelling and for predictive rather than purely descriptive purposes. Predictive analytics focuses on application of statistical or structural models for predictive forecasting or classification, while text analytics applies statistical, linguistic, and structural techniques to extract and classify information from textual sources, a species of unstructured data. Data integration is a predecessor to data analysis. Data analysis is closely linked to data visualization and data distribution. The term data analysis is sometimes used as a synonym for data modelling.

1.7 ORGANIZATION OF THE THESIS

The main contribution of the thesis is the implementation of Data mining in the resources of unstructured datum. From the careful investigations of existing measures, the computation schema and mapping schema are implemented for similarity analysis. In each sector of video, image of unstructured datum a real time framework model is analyzed with its parameters. The thesis is organized as follows:

- Chapter 2 describes the background concepts related to the research.
- Chapter 3 explains the work carried out by other researchers related to our research.
- Chapter 4 describes details regarding the video mining.
- Chapter 5 describes the event mining analysis of unstructured data.
- Chapter 6 explains the heuristical techniques of event mining on unstructured data.
- Chapter 7 explains the Univariate transformations for image mining.
- Chapter 8 gives the conclusion and directions for future work.

CHAPTER 2

MINING IN UNSTRUCTURED RESOURCES

In this chapter we discuss about some of the important concepts related to our research work such as video mining, event mining, structured and unstructured data, clustering, classification etc.

2.1 STRUCTURED RESOURCE ANALYSIS

Data that is the source for a data mining process can be classified into structured data, semi-structured data, and unstructured data. Most business databases contain structured data consisting of well-defined fields with numeric or alphanumeric values, while scientific databases may contain all three classes.

In structured data

- data is organized in semantic chunks
- similar entities are grouped together
- entities in the same group have same descriptions
- descriptions for all entities in a group
    - have same defined format
    - have a predefined length
    - are all present

- and follow same order

Examples of semi-structured data are electronic images of business documents, medical reports, executive summaries, and repair manuals. The majority of web documents also fall in this category. An example of unstructured data is a video recorded by a surveillance camera in a department store. Such visual and multimedia recordings of events or processes of interest are currently gaining widespread popularity because of reduced hardware costs. This form of data generally requires extensive processing to extract and structure the information contained in it.

Structured data is often referred to as traditional data, while the semi-structured and unstructured data are lumped together as non-traditional data (also called multimedia data). Most of the current data-mining methods and commercial tools are applied to traditional data.

Structured data analysis is the statistical data analysis of structured data. This can arise either in the form of an *a priori* structure such as multiple-choice questionnaires or in situations with the need to search for structure that fits the given data, either exactly or approximately. This structure can then be used for making comparisons, predictions, manipulations etc.

The following are the types of structured data analysis:

- Regression analysis
- Bayesian analysis

- Cluster analysis
- Combinatorial data analysis
- Geometric data analysis
- Topological data analysis
- Shape analysis
- Functional data analysis
- Tree structured data analysis
- Formal concept analysis
- Algebraic data analysis

2.2 UNSTRUCTURED RESOURCE ANALYSIS

Unstructured data [5] has no identifiable structure. Unstructured data typically includes bitmap images/objects, text and other data types that are not part of a database. Unstructured data is a generic label for describing any corporate information that is not in a database. In Unstructured Data

- data can be of any type
- not necessarily following any format or sequence
- does not follow any rules
- is not predictable
- examples include
    - text
    - video

- sound
- images

Unstructured data can be textual or non-textual. Textual unstructured data is generated in media [17] like email messages, PowerPoint presentations, Word documents, collaboration software and instant messages. Non-textual unstructured data is generated in media like JPEG images, MP3audio files and Flash video files. Some current technologies used for content searches on unstructured data require tagging entities such as names or applying keywords and meta tags. Therefore, human intervention is required to help make the unstructured data machine readable.

Garbage in, garbage out has been a saying in computing since its beginning. Working with data that has no, or little, meaning leaves information process in terrible circumstances. Companies need a single and unified view of data. This includes structured data [5] that is readily available and unstructured data that may be buried or not traceable by normal procedures. If the use of data is to be optimized in a process it must be open to analysis.

In addition to data that is coded and simply numeric as data stored in rows and columns there is data that is free-form that can be from written notes, word documents, e-mail, reports, news feeds, etc. This latter data may be critical and can affect business performance, trends and customer feedback.

2.3 CLASSIFICATION

Classification is a data mining technique used to predict group membership for data instances. Popular classification techniques include decision trees and neural networks. The form of classifier depends on the classification technique used. For example, neural network produce a set of weight as a classifier, regression form an equation as a predictor while decision tree, C4.5, CART, Rough Set and Bayesian theory generate set of rules known as rule based classifier. Rules are more interpretable by human when compared to other form of classifiers. The process of classification involves applying the rules onto a set of unseen data. There are many issues appeared in rule application process such as more than one rule match, multiple scanning of large rule base and uncertainty.

Various parametric and non-parametric methods are used to solve classification related problems. Traditional statistical methods are parametric in nature based on the assumptions about the nature of the distributions and estimate the parameters of the distributions to solve the problem. Non-parametric methods, make no assumptions about the specific distributions involved, and are therefore distribution-free.

Discriminant analysis is a technique for classifying a set of observations into two or more predefined classes. The purpose is to determine the class of an observation based on a set of variables known as predictors or input variables. The model is built based on a set of observations for which the classes are known. This set of observations is sometimes referred to as the training set. Based on the training set, the technique constructs a set of linear functions of the predictors, known as discriminant functions, such that $L = b_1x_1 + b_2x_2 + \ldots\ldots + b_nx_n + c$, where the b's are discriminant coefficients, the x's are the input variables or predictors and c is a constant.

These discriminant functions are used to predict the class of a new observation with unknown class. For a k class problem k discriminant functions are constructed. Given a new observation, all the k discriminant functions are evaluated and the observation is assigned to class i if the i^{th} discriminant function has the highest value.

Discriminant Analysis (DA), a multivariate statistical technique is commonly used to build a predictive / descriptive model of group discrimination based on observed predictor variables and to classify each observation into one of the groups. In DA multiple quantitative attributes are used to discriminate single classification variable. DA is different from the cluster analysis because prior knowledge of the classes, usually in the form of a sample from each class is required.

The common objectives of DA are

1. To investigate differences between groups;
2. To discriminate groups effectively;
3. To identify important discriminating variables;

4. To perform hypothesis testing on the differences between the expected groupings;
5. To classify new customers into pre-existing groups.

2.4 CLUSTERING

Clustering is “the process of organizing objects into groups whose members are similar in some way”. A *cluster* is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters. Clustering [21] is the process of dividing a dataset into mutually exclusive groups such that the members of each group are as "close" as possible to one another, and different groups are as "far" as possible from one another, where distance is measured with respect to all available variables.

A set of unordered observations, each represented by an n-dimensional feature vector, will be partitioned into smaller, homogenous and practical useful classes C_1, C_2, ..., C_k such that in a well-defined sense similar observations are belonging to the same class and dissimilar observations are belonging to different classes. This definition implies that the resulting classes have a strong internal compactness and a maximal external isolation. Graphically this means, that each cluster would be represented by a spherical shape in the n-dimensional feature space. However, real world cluster may not follow this model assumption.

Clustering can be done based on the observations or on the attributes. Whereas the first approach gives us object classes the later one is very helpful in order to discover redundant attributes or even attribute groups that can be summarized into a more general attribute. The basis for clustering [21] is a data table containing lines with *m* observations and rows for *n* attributes describing the value of the attributes for each observation. Cluster analysis or clustering is the assignment of a set of observations into subsets called *clusters* so that observations in the same cluster are similar in some sense. Clustering is a method of unsupervised learning, and a common technique for statistical data analysis used in many fields, including machine learning, data mining, pattern recognition, image analysis, information retrieval, and bioinformatics [23].

Clustering Algorithms

Hierarchical algorithms find successive clusters using previously established clusters. These algorithms usually are either agglomerative ("bottom-up") or divisive ("top-down").

- Agglomerative algorithms begin with each element as a separate cluster and merge them into successively larger clusters.
- Divisive algorithms begin with the whole set and proceed to divide it into successively smaller clusters.

Partitional algorithms determine all clusters at once, but can also be used as divisive algorithms in the hierarchical clustering. Density-based algorithms are devised to discover arbitrary-shaped clusters. In this approach, a cluster is regarded as a region in which the density of data objects exceeds a threshold. DBSCAN and OPTICS are two algorithms of this kind.

Many clustering algorithms [2] require the specification of the number of clusters to produce in the input data set, prior to execution of the algorithm. Barring knowledge of the proper value beforehand, the appropriate value must be determined, a problem on its own for which a number of techniques have been developed.

Advantages and Disadvantages

Cluster analysis is a good way for quick review of data, especially if the objects are classified into many groups. Cluster analysis also suggests how groups of units are determined such that units within groups are similar in some respect and unlike those from other groups.

An object can be assigned in one cluster only. In k-means clustering methods, it often requires several analysis before the number of clusters can be determined. It can be very sensitive to the choice of initial cluster centers.

2.5 SIMILARITY MEASURES

The calculation of similarity between the attributes must be meaningful. It makes no sense to compare two attributes that do not make a contribution to the considered similarity. Since attributes can be numerical and categorical or a combination of both, pay attention to this by the selection of the similarity measure. Not all similarity measures can be used for categorical attributes or can deal at the same time with numerical and categorical attributes. The variables should have the same scale level.

Similarity Measures for Images

Images can be rotated, translated, different in scale, or may have different contrast and energy but they might be considered as similar. In contrast to that, two images may be dissimilar since the object in one image is rotated by 180 degree. The concept of invariance in image interpretation is closely related to that of similarity. A good similarity measure should take this into consideration. The classical similarity measures do not allow this. Usually, the images or the features have to be pre-processed in order to be adapted to the scale, orientation or shift. This process is a further processing step which is expensive and needs some a-priori information which is not always given. Filters such as matched filters, linear filters, Fourier or Wavelet filters are especially useful for invariance under translation and rotation [34]. There has been a lot of work done to develop such filters for image interpretation in the past. The best way to achieve scale invariance from an image is by means of invariant moments [12], which can also be invariant [31] under rotation [34] and other distortions. Some additional invariance can be obtained by normalization.

Depending on the image representation similarity measures can be divided into:

- pixel (Iconic)-matrix based similarity measures,
- feature-based similarity measures, and
- structural similarity measures.

Since a CBR image interpretation system has to take into account nonimage information such as about the environment or the objects etc, similarity measures which can combine non-image and image information are needed.

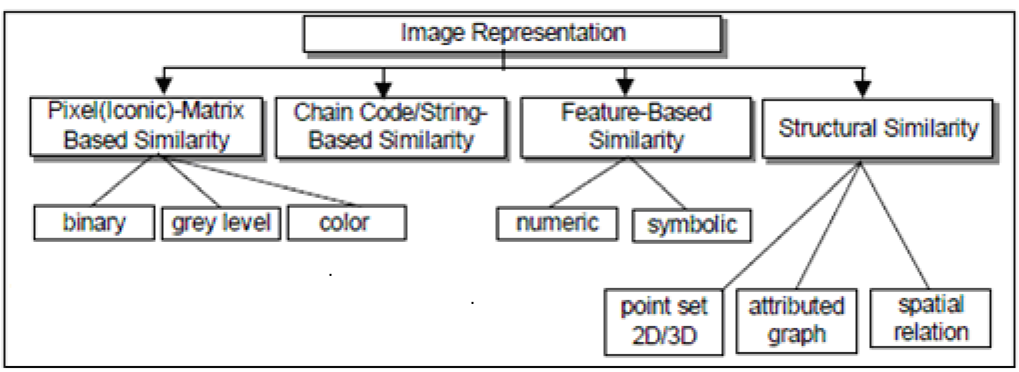


Figure 2.1: Image Representations

An important step in most clustering is to select a distance measure, which will determine how the *similarity* of two elements is calculated. This will influence the shape of the clusters, as some elements may be close to one another according to one distance and farther away according to another. For example, in a 2-dimensional space, the distance between the point ($x = 1$, $y = 0$) and the origin ($x = 0$, $y = 0$) is always 1 according to the usual norms, but the distance between the point ($x = 1$, $y = 1$) and the origin can be 2, $\sqrt{2}$ or 1 if you take respectively the 1-norm, 2-norm or infinity-norm distance.

The Common distance functions are Euclidean distance, Manhattan distance, Maximum norm etc. Mahalanobis distance corrects data for different scales and correlations in the variables. The angle between two vectors can be used as a distance measure when clustering high dimensional data. The Hamming distance measures the minimum number of substitutions required to change one member into another. Another important distinction is whether the clustering uses symmetric or asymmetric distances. Many of the distance functions listed above have the property that distances are symmetric.

2.6 MODELLING

Data modelling is a way to structure and organize data so it can be used easily by databases. Unstructured data [5] can be found in word processing documents, email messages, audio or video files, and design programs. Data modelling is routinely used in conjunction with a database management system. Data that has been modelled and made ready for this system can be identified in various ways, such as according to what they represent or how they relate to other data. The idea is to make data as presentable as possible, so analysis and integration can be done with as little effort as necessary. For an information system to be useful, reliable, adaptable, and economic, it must be based first on sound *data modelling*, and only secondarily on *process analysis*.

A model is a symbolic or abstract representation of something real or imagined. A data model helps us visualize data structures to measure how completely and accurately they reflect our information system problem space.

Data modelling is a method used to define and analyze data requirements needed to support the business processes of an organization. The data requirements are recorded as a conceptual data model with associated data definitions. Actual implementation of the conceptual model is called a logical data model. To implement one conceptual data model may require multiple logical data models. Data modelling defines not just data elements, but their structures and relationships between them. Data modelling techniques and methodologies are used to model data in a standard, consistent, predictable manner in order to manage it as a resource.

Data modelling may be performed during various types of projects and in multiple phases of projects. Data models are progressive; there is no such thing as the final data model for a business or application. Instead a data model should be considered a living document that will change in response to a changing business. The data models should ideally be stored in a repository so that they can be retrieved, expanded, and edited over time. Two types of data modelling are:

- Strategic data modelling: This is part of the creation of an information systems strategy, which defines an overall vision and architecture for information systems is defined. Information engineering is a methodology that embraces this approach.
- Data modelling during system analysis: In system analysis logical data models are created as part of the development of new databases.

Data modelling is also a technique for detailing business requirements for a database. It is sometimes called *database modelling* because a data model is eventually implemented in a database.

Conceptual, logical and physical schemes

A data model can be an external model (or view), a conceptual model, or a physical model.

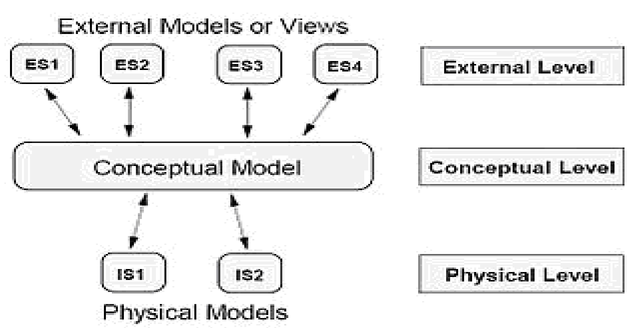


Figure 2.2: The ANSI/SPARC three level architecture

A data model *instance* may be one of three kinds:

- Conceptual schema: describes the semantics of a domain, being the scope of the model. A conceptual schema specifies the kinds of facts or propositions that can be expressed using the model.
- Logical schema: describes the structure of some domain of information. This consists of descriptions of tables and columns, object oriented classes, and XML tags, among other things.
- Physical schema: describes the physical means by which data are stored. This is concerned with partitions, CPUs, tablespaces, and the like.

Storage technology can change without affecting either the logical or the conceptual model. The table/column structure can change without affecting the conceptual model. In each case, the structures must remain consistent with the other model. The table/column structure may be different from a direct translation of the entity classes and attributes, but it must ultimately carry out the objectives of the conceptual entity class structure. Early phases of many software development projects emphasize the design of a conceptual data model. Such a design can be detailed into a logical data model. In later stages, this model may be translated into physical data model. However, it is also possible to implement a conceptual model directly.

Data modelling process

The actual database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a Data Definition Language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity. The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model these are the tables and views. In an Object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the DataBase Management System or DBMS.

Modelling methodologies

Data models represent information areas of interest. While there are many ways to create data models, only two modelling methodologies stand out, bottom-up and top-down:

- Bottom-up models are often the result of a reengineering effort. They usually start with existing data structures forms, fields on application screens, or reports. These models are usually physical, application-specific, and incomplete from an enterprise perspective. They may not promote data sharing, especially if they are built without reference to other parts of the organization.

- Top-down logical data models, on the other hand, are created in an abstract way by getting information from people who know the subject area. A system may not implement all the entities in a logical model, but the model serves as a reference point or template.

Sometimes models are created in a mixture of the two methods: by considering the data needs and structure of an application and by consistently referencing a subject-area model.

Entity relationship diagrams

There are several notations for data modelling. The actual model is called "Entity relationship model", because it depicts data in terms of the entities and relationships described in the data. An entity-relationship model (ERM) is an abstract conceptual representation of structured data. Entity-relationship modelling is a relational schema database modelling method, used in software engineering to produce a type of conceptual data model (or semantic data model) of a system, often a relational database, and its requirements in a top-down fashion.

These models are being used in the first stage of information system design during the requirements analysis to describe information needs or the type of information that is to be stored in a database. The data modelling technique can be used to describe any ontology (i.e.

an overview and classifications of used terms and their relationships) for a certain universe of discourse i.e. area of interest.

Generic data modelling

Generic data models are generalizations of conventional data models. They define standardized general relation types, together with the kinds of things that may be related by such a relation type. The definition of generic data model is similar to the definition of a natural language. For example, a generic data model may define relation types such as a 'classification relation', being a binary relation between an individual thing and a kind of thing (a class) and a 'part-whole relation', being a binary relation between two things, one with the role of part, the other with the role of whole, regardless the kind of things that are related. Given an extensible list of classes, this allows the classification of any individual thing and to specify part-whole relations for any individual object. By standardization of an extensible list of relation types, a generic data model enables the expression of an unlimited number of kinds of facts and will approach the capabilities of natural languages. Conventional data models have a fixed and limited domain scope, because the usage of such a model only allows expressions of kinds of facts that are predefined in the model.

Semantic data modelling

The logical data structure of a DBMS, whether hierarchical, network, or relational, cannot totally satisfy the requirements for a conceptual definition of data because it is limited in scope and biased toward the implementation strategy employed by the DBMS.

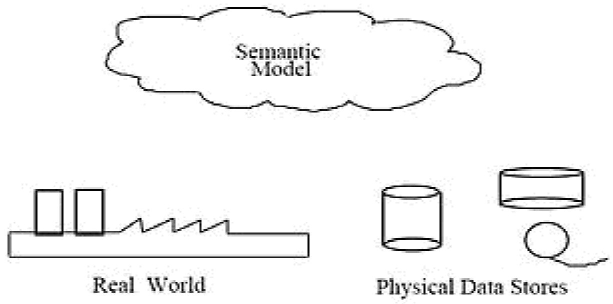


Figure 2.3: Semantic data models

The need to define data from a conceptual view has led to the development of semantic [4] data modelling techniques. That is, techniques to define the meaning of data within the context of its interrelationships with other data. As illustrated in the figure the real world, in terms of resources, ideas, events, etc., are symbolically defined within physical data stores. A semantic [4] data model is an abstraction which defines how the stored symbols relate to the real world. Thus, the model must be a true representation of the real world.

A semantic [15] data model can be used to serve many purposes, such as:

- planning of data resources
- building of shareable databases
- evaluation of vendor software
- integration of existing databases

The overall goal of semantic [4] data models is to capture more meaning of data by integrating relational concepts with more powerful abstraction concepts known from the Artificial Intelligence field. The idea is to provide high level modelling primitives as integral part of a data model in order to facilitate the representation of real world situations.

Benefits of Data Modelling

Abstraction: The act of abstraction expresses a concept in its minimum, most universal set of properties. A well abstracted data model will be economical and flexible to maintain and enhance since it will utilize few symbols to represent a large body of design. In data modelling, strong methodologies and tools provide several powerful techniques which support abstraction. Entity sub-types enable the model to reflect real world hierarchies with minimum notation. Automatic resolution of many-to-many relationships into the appropriate tables allows the modeler to focus on business meaning and solutions rather than technical implementation.

Transparency: Transparency is the property of being intuitively clear and understandable from any point of view. A good data model enables its designer to perceive truthfulness of design by presenting an understandable picture of inherently complex ideas. The data model can reveal inaccurate grouping of information, incorrect relationships between objects, and contrived attempts to force data into preconceived processing arrangements.

Effectiveness: An effective data model does the right job - the one for which it was commissioned - and does the job right - accurately, reliably, and economically. It is tuned to enable acceptable performance at an affordable operating cost.

2.7 VIDEO MINING

The rare abnormal video events [3] overlapped can be detected within the platform of data mining framework through geometric reconstruction. The supervised and unsupervised clustering instances with feature extraction will be used to support the final event detection [27]. The significant and optimal expectation of the proposed framework can be explained through a collection of cricket game videos within an over containing sample datum. Video sequences are typically segmented into shots, where each shot represents a contiguous scene with a certain context. Automatically identifying the boundary between shots is an active area of research that has made great strides. Once a shot has been identified, it is often represented by a *key frame*. The key frame is then often used for other purposes such as manual or automated annotation and shot representation during a search. Video analysis [29] techniques often decompose the analysis problem into one that can be handled using the image analysis techniques. This is fine for many applications, but it may not capture temporal information inherent in video data. Approaches do exist for capturing this information such as identifying distinctive features in successive video frames.

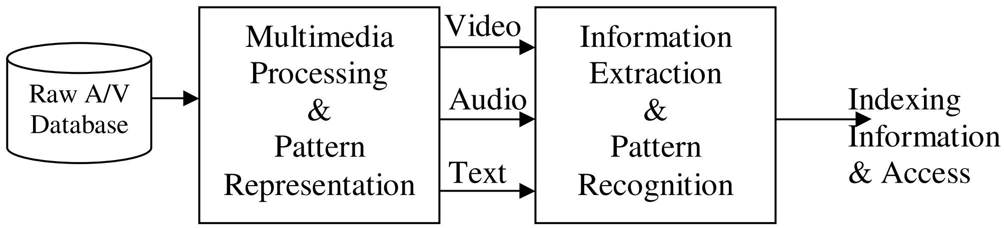


Figure 2.4: Information Extraction from Video

Video mining is unsupervised discovery of patterns [24] in audio-visual content. Such purely unsupervised discovery is readily applicable to video surveillance as well as to consumer video browsing applications. We interpret video mining as content-adaptive or blind content processing, in which the first stage is content characterization and the second stage is event discovery based on the characterization obtained in first stage. The purpose of video data mining is to discover and describe interesting patterns in data. The task becomes especially tricky when the data consist of video sequences, because of the need to analyze enormous volumes of multidimensional data.

Overview of video mining system

Personal desktop multimedia [17] mining aims to help users to search browse and manage their media contents on desktop PC easily. The key components include low-level feature extraction, short boundary detection, scene/story segmentation, video structural summary, high level semantic [4] concept detection, intelligent annotation, indexing and content based image/video retrieval. In order to analyze video content semantically, it is essential to fuse multi-modality information to bridge the gap between human semantic concepts and computer low-level features from both the video sequence and audio streams. A video mining system framework that consists of three layers is given in figure 2.5.

First, MPEG decoder decodes the input video into visual, audio and motion [7] streams. Then the shot boundary detection determines the shot boundary according to general features extracted from the visual streams, e.g., colour histogram, mean value and standard deviation of each frame's pixel intensities etc.

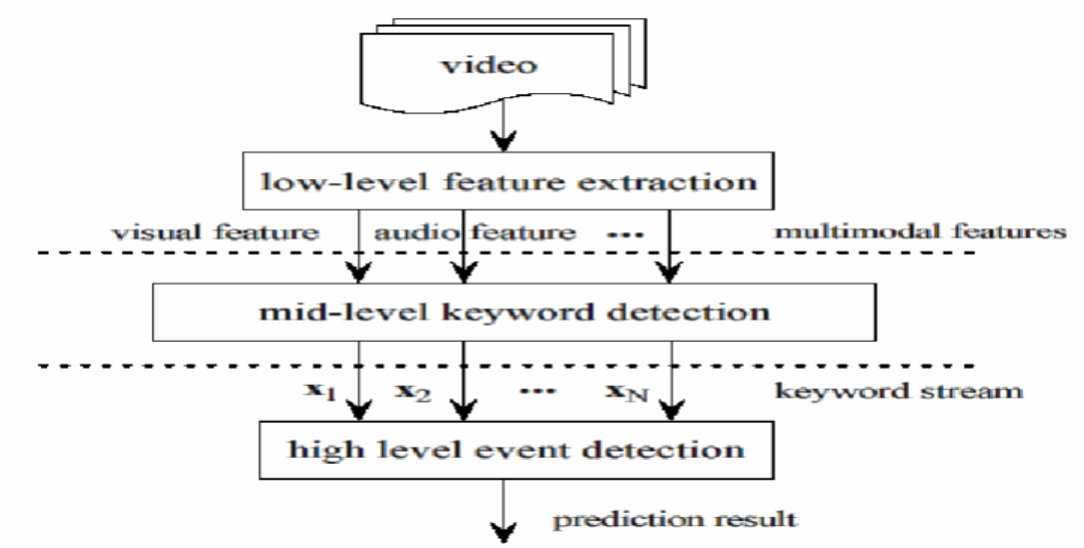


Figure 2.5: Overview of the video mining system

A few key frames are selected from each shot to represent the shot's content. After the key frames are ready, more complex features are extracted from these key frames to detect the mid-level keywords by multi-modality fusion, i.e. visual and audio keywords of face, building, road, excited speech, music and explosion etc. Finally, according to the mid-level keywords representation in time series, high-level events can be derived e.g. highlights in sports video. Similar to text mining based on parsing of word, sentence, paragraph and whole document, parsing a video often consists of four levels, i.e., frame, shot, scene and the whole video sequence. To analyze the video content semantically, shot boundary detection is a prerequisite step. In general, a shot is a set of video frames captured by a single camera in one consecutive shoot action. According to whether the transition between shots is abrupt or not, the shot boundaries are categorized into cut and gradual transition (GT). According to the characteristics of different editing effects, the GTs can be further classified into dissolve wipe, and fade out/in etc. types. The cut detector uses 2nd order derivatives of colour histogram, flash light detector, and GT filter. The GT detector uses the same features as the cut detector plus motion [7] vectors.

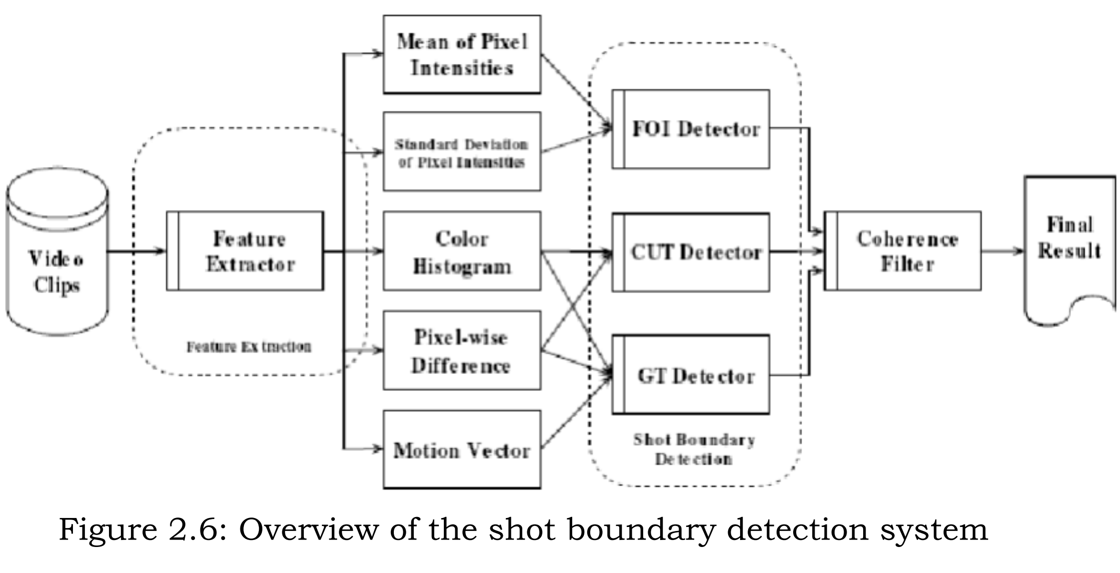


Figure 2.6: Overview of the shot boundary detection system

The overview of the shot detection in the video mining system is shown in the figure 2.6. It has 48 bins colour histogram in RGB space and 16 bins for each channel. Pixel-wise difference feature, as a supplement to colour histogram, is used to represent the spatial information. To detect flashlight effect and monochrome frame, the mean value and standard deviation of each frame's pixel intensities are calculated, at the same time, abrupt change of illumination is detected by tracking the variation of mean gray value. Besides features from uncompressed domain, motion [5] vectors from compressed domain are also extracted to reflect the global motion [7] of a frame.

Driven by the ever increasing needs of numerous multimedia and online database applications, there is a growing interest in video mining research. The goal of video mining is to discover knowledge, patterns, and events called *semantic structures* in the video data to facilitate the user's data access. There are three main research issues for video mining, i.e., summarization/abstraction, browsing/skimming, and indexing/retrieval.

Due to different production and editing styles, videos can be classified into two categories: the scripted and the non-scripted. Usually, they are associated with different video mining tasks. *Scripted videos* are produced or edited according to a pre-defined script or plan. News and movies are highly scripted videos that are composed of pre-defined segments or episodes. The events in *non-scripted videos* happen spontaneously and usually in a relatively fixed setting, such as meeting, sports, and surveillance videos.

Sports video analysis [29] has been widely studied due to its great commercial value. Although sports videos are considered as non-scripted, they usually have a relatively well-defined structure or repetitive patterns, which could help us to enhance the scripted ness of sports videos for more versatile and flexible access. In general, there are two kinds of video mining approaches: the *event based* [40] and *structure-based*. The event based approaches detect the events-of-interest or highlights, which improve the semantic [15] understanding of the video content. The structure-based approaches can parse a long video sequence into individual segments, e.g., the play/break, which can be used as a mid-level video content representation.

Although these approaches provide limited semantics, they can facilitate further high-level semantic analysis. A mid-level representation framework was proposed for semantic sports video analysis [29] [40][14] where both temporal structures and event hierarchy were involved; a mosaic-based generic scene representation was developed from video shots [6] and used to mine both events and structures. Machine learning is considered as one of the most effective approaches for semantic video analysis [29]. Hidden Markov models (HMMs) is the most popular one.

2.8 EVENT MINING

Events are real world occurrences that unfold over space and time. Event mining [28] from multimedia streams improves the access and reuse of large media collections. *Event Mining* discovers and delivers information and knowledge in a real-time stream of data, or events. Event detection [27] is a natural application for multimedia mining. It has a variety of practical applications that are becoming more widespread as data-collection instruments become more sophisticated and less expensive. In particular, video event detection systems are becoming increasingly popular for many applications.

Some examples of these systems include:

- Robotics (e.g., obstacle detection and avoidance)
- Water contaminant warning system
- Video surveillance, including motion [7] detection, unknown-object identification, and
- Detection of foot-traffic flow in retail stores.

With the rapid increase of many kinds of sports video, how to analyze and manage their contents becomes an important and interesting issue. Semantic [15] classification that identifies meaningful events from a video sequence forms the kernel technology for applications in such as sports video understanding, summarization, and retrieval. For example, the highlights can be identified to form a personalized summary of the video so that users would save a lot of time to see what he/she wants to see. Recently, many researches focus on content analysis of sports video.

Event detection [27] is of great importance for effective video indexing [36], summarization, browsing, and retrieval. However, due to the challenges posed by the so-called semantic gap issue and rare event detection, most of the existing works rely heavily on the domain knowledge with large human interference. With the introduction of an advanced temporal analysis process, the representative event patterns can be explored, discovered, and represented with little human effort. In addition, the multimodal data mining [25] technique on the basis of the decision tree algorithm is adopted to select the representative features automatically and to deduce the mappings from low-level features to high-level concepts.

Different Tasks in Event Mining

The task of associating one or more semantics from data can take one of many forms, depending on the data available and the target decision. Extending from pattern classification problems in closely related areas such as face detection and recognition, speaker identification and image segmentation [39] we present the following six tasks on event semantics.

- Detection
- Segmentation
- Recognition

- Verification or identification
- Annotation
- Discovery

The three tasks detection, recognition, and verification require a known event model or event description, while annotation requires more than one event models that may be interrelated. Finally, in current event-modelling systems some of these tasks tend to associate with particular attributes.

Event Properties

Regular or Sporadic, Usual or Unusual: Many existing research in event detection carry one or more of the common modifiers for their detection target. These modifiers constrain event detection problems differently and they exercise important influences on the algorithm being chosen. Being regular means marked or distinguished by steadiness or uniformity of action, procedure, or occurrence, while being sporadic is characterized by occasional or isolated occurrence, appearance, or manifestation.

An alternative classification of event types is usual versus unusual, or normal versus abnormal. Being unusual or abnormal not only means that the why aspect is unaccounted for, but also means that the 'what and how' aspects are unknown from a normal pool of media clips. These event properties affect our algorithm design choices. For example, regular and sporadic events in sports can both be detected with a set of trained classifiers; detecting unusual [42] events often means finding outliers that do not fit the current set of models; once discovered, we can also build models to describe the unusual events.

Applications of Event Modelling and Detection

In order to understand what an event detection [27] system should achieve, it helps to examine the uses of event metadata. Event metadata can provide semantically meaningful indexes that help decompose a task with faceted metadata and map an information-seeking task to multimedia event ontology, such as the large-scale multimedia ontology for broadcast news. This can provide additional aspects for matching and filtering information just as the faceted attributes of author, title, publisher, helps catalog, search, and promote books in libraries. Many information-seeking tasks can be mapped onto such event-based semantic metadata. For instance: 1) Active, or goal-oriented information seeking involves finding media clips that matches an existing description 2) Matching involves deciding whether or not a media [17] clip matches a given description 3) Browsing and impression formation means trying to get an idea about the content of an entire collection from an overview or random sample of the content or the thumbnails view in modern operating systems. Semantic metadata can be directly summarized into text form, tag clouds, or thumbnails, which are more intuitive and convenience for an overview than collections of media sequences unrolled in time 4) Indexing and archival is to insert metadata for items in a collection so that they can be easily found at a later time. Here, event metadata are directly applicable as additional indexes.

The benefit of semantic metadata on information seeking tasks can propagate its influence to real-world multimedia systems, for example, semantic metadata has helped generate significantly better results for automatic and interactive video search.

2.9 IMAGE MINING

Image Data Basics

An image consists of a two-dimensional array of numbers. The colour or gray shade displayed for a given picture element (pixel) depends on the number stored in the array for that pixel. The simplest type of image data is black and white. It is a binary image since each pixel is either 0 or 1. The next, more complex type of image data is gray scale, where each pixel takes on a value between zero and the number of gray scales or gray levels that the scanner can record. These images appear like common black and white photographs. Most gray scale images today have 256 shades of gray. People can distinguish about 40 shades of gray, so a 256-shade image looks like a photograph. The most complex type of image is colour. Colour images are similar to gray scale except that there are three bands, or channels, corresponding to the colours red, green, and blue. Thus, each pixel has three values associated with it. A colour scanner uses red, green, and blue filters to produce those values.

The most used features for image description are: colour, texture, shape and spatial features. Many of the existing image databases allow users to formulate queries by submitting an example image. The system then identifies those stored images whose feature values match those of the query most closely, and displays them. Colour features are usually represented as a histogram of intensity of the pixel colours. Same system, partitions the image into blocks and each block is indexed by its dominant hue and saturation values. Colour and spatial distribution can be also captured by an anglogram data structure. The most used texture features are the Gabor filters. Shape feature techniques are represented from primitive measures such as area and circularity to more sophisticated measures of various moment [16] invariants; and transformation-based methods ranging from functional transformations such as Fourier descriptors to structural transformations such as chain codes and curvature scale space feature vectors. High level image semantic representation techniques are based on the idea of developing a model of each object to be recognized and identifying image regions which might contain examples of the image objects. The system analyzed object drawings, and use grammar structures to derive likely interpretations of the scene. The user is asked to identify a possible range of colour, texture, shape or motion [7] parameters to express his or her query, which is then, refined using relevance feedback techniques.

When the user is satisfied, the query is given a semantic label and stored in a query database for later use.

Image mining deals with the extraction of knowledge, image data relationship, or other patterns not explicitly stored in the images. It uses methods from computer vision, image processing [18], image retrieval, data mining, machine learning, database, and artificial intelligence. Rule mining has been applied to large image databases. There are two main approaches. The first approach is to mine from large collections of images alone and the second approach is to mine from the combined collections of images and associated alphanumeric data.

Content-Based Retrieval of Images

In content-based retrieval, the user describes the desired content in terms of visual features, and the system retrieves images that best match the description. Content-based retrieval is therefore a type of retrieval by similarity. Image content can be defined at different levels of abstraction. At the lowest level, an image is a collection of pixels. Pixel-level content is rarely used in retrieval tasks. The raw data can be processed to produce numeric descriptors capturing specific visual characteristics called *features*. The most important features for image databases are colour, texture, and shape. Features can be extracted from entire images describing global visual characteristics or from portions of images describing local characteristics. A feature-level representation of an image requires less space than the image itself. The next abstraction level describes the semantics. A semantic-level characterization of photographic images is a complex task. Semantic content can be inferred from the lower abstraction levels. At the highest level, images are often accompanied by metadata. Metadata may contain information that cannot be obtained directly from the image, as well as an actual description of the image content.

To support content-based retrieval at any of the abstraction levels, appropriate quantities that describe the characteristics of interest must be defined, algorithms to extract these quantities from images must be devised, similarity measures to support the retrieval must be selected, and indexing techniques to efficiently search large collection of data must be adopted.

Image Feature Extraction Mechanism

The mechanism transfers low level image characteristics into high level semantic features using fuzzy [33] production rules with degree of recognition and image interpretation. For the low level image characteristics the following colour, shape and texture features are calculated.

a) Colour characteristics

The colour feature extraction procedure includes colour image segmentation. First the standard RGB image is converted as L*u*v* image, where L* is luminance, u* is redness–greenness, and v* is approximately blueness– yellowness. Twelve hues are used as fundamental colours. There are yellow, red, blue, orange, green, purple, and six colours obtained as linear combinations of them. Five levels of luminance and three levels of saturation are identified. This results that every colour is transferred into one of 180 references colours. After that clustering in the 3-dimensional feature space is performed using the *K*-means algorithm. After this step the image is segmented as N regions, each of which is presented in extended chromaticity space.

b) Texture characteristics

The Quasi-Gabor filter is explored to present the image texture features. The image is characterized with 42 values by calculation of the energy for each block defined by a combination of one of 6 frequencies (f = 1, 2,4, 8, 16 and 32) and one of 7 orientations (q = 0°, 36°,72°, 108°, 144°, 45° and 135°). Take the average value of the magnitude of the filtered image in each block.

c) Shape characteristics

For shape representations the image is converted into binary. Polygonal approximation that uses straight-line, Bezier curve and BSpline are applied. As a result the image is presented as a set of straight lines, arcs and curves.

Retrieval Based on High Level Semantic Features

Here image retrieval based on high level colour, texture, shape and semantic features are discussed.

a) Retrieval by high level colour properties

There are seven types of contrast namely Contrast of hue, Light-dark contract, Cold-warm contrast, Complementary contrast, Simultaneous contrast, Contrast of saturation, Contrast of extension. Harmony is defined as a combination of colours resulting in a gray mix that generates stability effect onto the human eyes. Non-harmonic combinations are called expressive. Fuzzy [33] production rules are used to translate the low level semantic features into sentences qualifying warmth degree, and contrasts among colours.

b) Retrieval by high level texture properties

Transforming the low level texture characteristics into high level semantic features such as texture of wood, rock, wall-paper, etc. is made by calculating the low level texture characteristic of a typical set of corresponding textures and finding the “cluster center” values which is used in the fuzzy production rules.

c) Retrieval by high level shape properties

A set of typical shapes characterizing the domain specific objects are defined. Fuzzy production rules are used for calculation similarity between the search shape and given object shape. They are obtained after image mining.

d) Retrieval by high level semantic features

A set of high level semantic [4] features which are defining in the image mining process are used. They combine high level colour, texture and shape properties and high level semantic features defined by the expert during the image mining.

2.10 UNIVARIATE TRANSFORMATIONS

Image mining is defined as the analysis of observational images to find (un)suspected relationships and to summarize the data in novel ways that are both understandable and useful. The five important steps in image mining are identification, modelling, tracking, prediction and communication with stakeholders. All these processes are discussed below.

Identification

Focusing on uncertain objects, consider first their identification. Identification requires making the step from raster to objects. That means segmentation, followed by a classification. Identification is done by applying a segmentation routine, in which both the object and the uncertainty are modelled. Image segmentation [39] includes procedures based on mathematical morphology, on edge detection and on identifying homogeneity in one band or in a set of bands. Discriminant analysis could be applied, honoring the different spectral bands and yielding posterior probabilities to the nonselected classes. Interesting results are being obtained by applying texture based segmentation for single bands, and for multiple bands. This leads to improved segmentation, again including uncertainty values. The result of this operation for an image at moment t thus is a series of nt objects $O_{t,i}$, i = 1,…,nt, that are characterized by similar pixel values, which are different from pixels values in the vicinity.

Modelling

Modelling of identified objects $O_{t,i}$ can typically be done by applying a fuzzy approach, resulting in membership functions of the objects to a class A of interest. Membership functions in a 2-dimensional space are functions μA(x, y), taking values between 0 and 1, which specify the degree to which the location (x, y) is characterized by A. A membership function is formally defined by a function η(z) for z∈[0,1], which transformed using a homeomorphism towards the membership function μA(x, y) for (x, y)⊂ A, the set of interest. Such membership functions are characterized by the steepness of their slopes, i.e. showing how rapidly it increases from zero to one, by their homogeneity and by their support and centroid. For image mining with its combined spatial/temporal variation, consider the development of fuzzy sets in time. In a proper modelling effort characteristics of membership functions are being recorded and stored on those moments that images are available. This will facilitate modelling during the next stage and help to predict or find the cause of the spatio-temporal object. The centroid, for example, has coordinates ($x_{c,t}$, $y_{c,t}$) at each moment t, and when it is present on n images is observed at the moments t_1,…, t_n. At the different moments these centroids can then be plotted and a curve can be fitted. As an alternative one could consider modelling by means of kernel densities and approximating Gaussian functions. The property that a parametric shape emerges at the expense of showing smoothness may not be a realistic property of the phenomenon under study. Alternatively, uncertainty is modelled by stochastic and probabilistic objects. Random sets concern the analysis of shapes and forms that are subject to random influences. Statistical methods are used for that purpose. One of the recently developed aspects concerns the development and implementation of operators on vague and fuzzy [33] sets. The univariate operators for a single vague set are distinguished from the bivariate operators for two vague sets. It has been shown that on the basis of α-cuts an extension of current crisp operators could be defined and implemented in a usable way. One binary variable operator is defined, being the distance between two objects. This distance is also defined in terms of the distance between α-cuts, resulting in a distance curve as a function of α. The distance decreases with increasing values of α, and integration of the distance curves provides a single value for the distance. The distance is not defined for values of α larger than the lowest of the maximum membership values of the two objects.

Tracking

After parameterization of the objects, the next stage is to model their behaviour in space and time. As long as the object is characterized by a limited set of parameter, such tracking may be relatively straightforward. However, treatment of objects that was relatively straightforward for crisp objects becomes somewhat more complicated for uncertain object: Splitting and merging of objects, their birth and death require some special attention. Let us consider the object $O_{k,1}$ at moment tk. The splitting of this object leads to two new objects $O_{k+1,1}$ and $O_{k+1,2}$ at moment tk+1. Both objects require membership functions, which are defined on the basis of the membership function at t as well as the characteristics of the new objects. An inverse operation is merging of two objects at moment tk towards one single object at moment tk+1. This requires the membership functions to be combined into one new membership function. Also the two existing centroids have to be combined into one new centroid, possibly by a weighted averaging of the centroids of the original objects. Tracking of objects thus requires a proper modelling of both splitting and merging. In addition, tracking requires to carefully consider the birth and death of objects.

Prediction

If the tracking routines are successful, one may venture into the next stage of image mining, e.g. prediction of the object in the space-time domain. When tracking a single uncertain object over time, one way to proceed is to define a parametric curve for the centroid and possibly of other parameters of the membership function and then to predict at which location the curve most likely will be at a moment t_0 beyond the moments of observation so far. In prediction, one may consider a future event, i.e. a real prediction, or one moment [34] prior to image availability, like predicting the moment that the object is born. Also, prediction is sometimes required for the object between two moments tk and tk+1. Prediction results in the values of the membership function, of the centroid, and of the associated uncertainties.

2.11 OPTIMIZATION TECHNIQUES

Optimization refers to the selection of a best element from some set of available alternatives. In the simplest case, this means solving problems in which one seeks to minimize or maximize a real function by systematically choosing the values of real or integer variables from within an allowed set. This formulation, using a real-valued objective function, is probably the simplest example; the generalization of optimization theory and techniques to other formulations comprises a large area of applied mathematics. Generally, it means finding "best available" values of some objective function given a defined domain, including a variety of different types of objective functions and different types of domains.

Multi-objective optimization

Adding more than one objective to an optimization problem adds complexity. For example, if you want to optimize a structural design, you would want a design that is both light and rigid. There will be one lightest design, one stiffest design, and an infinite number of designs that are some compromise of weight and stiffness. This set of trade-off designs is known as a Pareto set. The curve created plotting weight against stiffness of the best designs is known as the Pareto frontier. A design is judged to be Pareto optimal if it is not dominated by other designs: a Pareto optimal design must be better than another design in at least one aspect. If it is worse than another design in all respects, then it is dominated and is not Pareto optimal.

Multi-modal optimization

Optimization problems are often multi-modal, that is they possess multiple good solutions. They could all be globally good or there could be a mix of globally good and locally good solutions. Obtaining all the multiple solutions is the goal of a multi-modal optimizer.

Classical optimization techniques due to their iterative approach do not perform satisfactorily when they are used to obtain multiple solutions, since it is not guaranteed that different solutions will be obtained even with different starting points in multiple runs of the algorithm. Evolutionary algorithms are popular approaches to obtain multiple solutions in a multi-modal optimization task.

An optimization algorithm is an algorithm for finding a value x such that $f(x)$ is as small (or as large) as possible, for a given function f, possibly with some constraints on x. Here, x can be a scalar or vector of continuous or discrete values. An algorithm terminates in a finite number of steps with a solution.

An algorithm is a special case of an iterative method, which generally need not converge in a finite number of steps. Instead, an iterative method produces a sequence of iterates from which some subsequence converges to a solution.

2.12 PATTERN IDENTIFICATION

Atomic patterns are considered to be the fundamental patterns, which build a pattern language and cannot be broken down into a set of sub-patterns. Composite patterns [24] are a product of pattern integration that goes beyond a simple composition that groups patterns without any synergy. The fundamental pattern catalogue also defines a set of relations that can be treated as connections in a composition. The technique focuses on the patterns applied into a design and considers the pattern overlapping that can be present in specific design parts. Overlapping occurs when an individual design part has a role in two different patterns. Composite patterns that are identifiable with the techniques have constituents as overlapped patterns. The identification technique starts the analysis from the instantiated pattern variant in a design. The specific treatment of pattern overlapping distinguishes the approach from other existing attempts at composite pattern identification. Some of the early attempts at identifying patterns from an existing solution were built exclusively from the structural information that was constructed from a source code. The fundamental presumption is that pattern extraction is possible without additional information. Object-oriented software metrics are used for the purpose of identifying structural GoF patterns. In the case of other pattern types, false positives can occur. False positives must be detected and inspected by the user alone. Single class metrics are used to reduce the search space in a structure. Metrics such as NOA (Number of Attributes) and NOO (Number of Operations) have appeared in various configurations. Metric scores are used for the detection of candidate classes for structural patterns.

Patterns [24] can be detected with the help of basic metrics on the class structure. Patterns promote weak coupling between classes and a greater abstraction if the impact is observed on the level of an individual pattern. The process of detecting composite patterns can return different results, and should be assessed on adjusted score intervals. Design metrics, if applied properly, have proven effective as indicators of flaws and the inappropriate use of patterns in existing designs. The domain and language-independent discovery of patterns is possible with the use of formal specifications, which serve as an independent meta-layer between a specific design and conceptual artefacts. A formal specification language enables the formal definition of the patterns themselves and their application.

The goal of the identification process in all cases is to detect the patterns that can be atomic or composite. Atomic patterns are not a result of composing existing patterns. Early research dealt with the discovery of atomic patterns, which are included in existing catalogues. Finding a new extracted pattern that can be used in future designs, like any other pattern, justifies the invested effort. The application of a composite pattern increases the pattern's usage and protects a designer from the inappropriate application of several patterns. The set of patterns can be applied in a design with many variations, while the composite pattern consists of a proven solution for their application. A single pattern can appear in different designs in many variations. Pattern catalogues suggest basic forms of a pattern. It is expected that the same patterns will have a different cardinality and types of elements. This fact should be considered during the construction of the input data for the technique. The information in the applied patterns is a valuable base for further analysis.

Observing patterns as a whole in a design, it appears that all the patterns are connected through some interactions. An alternative presents the relationships that are defined by the pattern catalogue. Using these relationships between patterns, like glue in a composite, significantly reduces the possible combinations. No standardized set of pattern relations is defined when considering multiple catalogues. This reduces the space for pattern detection on an individual pattern language with the presumption that there is an appropriate set of relationships available.

The individual pattern parts in a composite should provide more pattern functionality than they provide when applied separately. The guideline for good synergy between patterns, in a composition, can be found in the level of pattern overlapping. The patterns in a composite can overlap. An individual part of such a design has various roles in used patterns. Overlapping can be observed on all the building parts of a pattern that are suggested in a pattern definition. A high level of overlapping indicates strong integration between individual patterns. Therefore we will define composite patterns as a set of patterns that are connected with the overlapping parts. When overlapping between patterns is detected, the candidates for composites can be extracted. The data needed for pattern coverage and pattern overlapping presentation requires that patterns applied in a design be conceived as sets of the connected building elements, which include classes, interfaces, methods and attributes. The methods and attributes, which are prescribed by a pattern, present the building parts for pattern classes and pattern interfaces. Classes and interfaces are referred to as the main elements of a pattern or a design, while the methods and attributes of a class or interface are referred to as sub-elements of a pattern or a design. The pattern coverage matrix precisely defines the form of instantiated patterns in individual design fragments. The matrix can be presented on a whole pattern, an element or a sub-element level of detail.

CHAPTER 3

RECENT EVOLUTION IN EVENT MINING

In this chapter we describe the event mining in detail with some of the researches that have been already carried out.

3.1 RESEARCH ON EVENT MINING

There has been major interest in recent years to classify videos according to the patterns of activity in them. Such activity-based analysis focused on supervised approaches which learn a predefined class of activities from a training set. Unsupervised approaches to video summarization dealt with problems such as shot boundary detection and scene classification. Most approaches cast it in the framework of a time-series clustering problem by breaking down the video stream into overlapping subsequences using 'sliding-windows', and then cluster these subsequences. Subsequence based approaches do not account for the fact that activities may have different temporal spans.

3.2 EVENT ANALYSIS

Classical approaches to event analysis and recognition are based on building a parametric or non-parametric model for a restricted set of pre-defined events. This approach involves an extensive and expensive training phase where the model for each event is learnt from several training examples of the event. In real world applications, one is not provided with an exhaustive set of the events that may occur in a given setting. This approach also requires the application to be retrained for every new deployment so that it is aware of all the events that are expected to occur within its field of view. These limitations have led researchers to look for unsupervised methods for video mining. But, such unsupervised approaches to mining events from videos present a few interesting challenges. First, since the system has no prior knowledge about the nature of the various events, the model for an event must be rich enough to support a wide variety of events that might occur. Second, the unsupervised approach has to recognize event boundaries and cluster consistent event subsequences into a single cluster. We show that a 'cascade of dynamical systems' is a very rich model for events and this model also provides a natural way of extracting event consistent subsequences from a long video.

Modelling Events in Videos

An event consists of a subject executing a series of action verbs in order to achieve a certain goal. Any model for events must be able to represent each of the verbs separately while simultaneously being able to detect the boundaries between them.

Modelling Action Elements

A complex event can be broken down into its constituent action elements. During each action element, the motion [7] of the actor remains consistent. In fact, it is this consistency of motion that segments an event into action elements. Therefore, each action element is modelled using a time invariant [31] dynamical system and the event is modelled as a cascade of dynamical systems.

Representation Feature

Since we are interested in mining events from video sequences, the feature extracted from each image frame must capture the motion [7] of objects in that frame. Generally, optical flow [26] serves as a good approximation to the true motion fields. Therefore, use optical-flow as the feature for representing motion in each image. In cases where optical flow computation is not robust, one can use background subtracted masks or even silhouettes as the feature. The appropriate feature to use depends on the end application.

3.3 PROBLEMS ON EVENT MINING

Event-Modelling Problems

Although we hear about systems that perform event detection, the meanings and scopes of what are being solved are very diverse. This diversity comes from two sources. The word detection can refer to several different computational tasks, some of which may be co-existing or overlapping. The properties of an event are mainly variable at different semantic [4] levels and aggregations along several principle attributes.

Such variation can transform the problem very severely. These two factors are examined in detail.

1) Different Tasks in Event Mining

The task of associating one or more semantics from data can take one of many forms, depending on the data available and the target decision. Extending from pattern classification problems in closely related areas such as face detection and recognition, speaker identification and image segmentation, the following six tasks on event semantics are presented here:

- Detection: compare data with a known event or event model; decide the presence or absence of the event.
- Segmentation: locate which part of the data corresponds to the event of interest; this specification can be in time, space, or both.
- Recognition: recover from data a description of the event containing one or more of the attributes.
- Verification or identification: confirm a specific property in the event
- Annotation: associating possibly more than one semantic [4] labels to data, possibly choosing from a semantic ontology and taking into account the relationships among the semantics.
- Discovery: find events without knowing its semantics beforehand, using the regularity or self-similarity among event instances.

These different tasks often co-occur, and the tasks - detection, recognition, and verification would require a known event model or event description, while annotation requires more than one event models that may be interrelated.

2) Event Properties: Regular or Sporadic, Usual or Unusual

Being regular means "marked or distinguished by steadiness or uniformity of action, procedure, or occurrence". Being sporadic is "characterized by occasional or isolated occurrence, appearance, or manifestation". Regular event instances can be aggregated into an event class by the steadiness or uniformity of their occurrences.

Being unusual or abnormal not only means that the why aspect is unaccounted for, but also means that the 'what and how' aspects are unknown from a normal pool of media clips.

3) Current Event Detection Problems:

Among the event detection problems currently being addressed, we can see a few salient groups with very similar problem setups. The similarities are in the semantic aspects or along event modelling components: detecting which of the five W (when, where, who, what, why) attributes, how to group or aggregate event instances, properties of target events, data format, and target applications.

Existing research works are in the following groups:

- Detecting known events and activities from one continuous capture;

- Event detection in edited sequences;
- Unsupervised event detection and pattern mining, i.e., detect unknown events;
- Event annotation in a collection of media objects.

Forms of Event Media Data

Multimedia archives are snapshots of real-world events from capturing, editing, and archiving with limited metadata and annotation. The setup for media capture imposes limits on what are available to us for data analysis; it also puts constraints for system design. Few scenarios and implicit but important assumptions are examined here.

1) Single Stream from One Continuous Take:

This scenario uses one camera and/or audio recorder, with either a fixed installation such as close-circuit surveillance or moving in space such as unmanned aerial vehicle (UAV) or lifelogs. The scope of data analysis is within a start/stop of the recording device. Thus, there is continuity in both space and time. Mapped to the 5W1H in event attributes, the offset between the media time and the real-world time is constant, and the location correspondence is either fixed or continuously changing. Such continuity enables common signal-level processing operations in the image sequences and sound such as motion [7] estimation, tracking, registration, background substraction, and mosaicing. This scenario is addressed by many visual-based event and activity modelling systems due to its high value in practice and the simple fact that the other scenarios are various combinations and compositions of the start/stop of recording.

2) Multiple Concurrent Streams:

Multiple cameras or microphones can be set up to capture multiple view points. This setup is usually found in surveillance networks and meeting recordings. These concurrent streams offer richer representation about the original scene. They are often calibrated so that the spatial and temporal correspondence can be reconstructed, until now they present additional challenges in data association for finding events from multiple sources.

3) Single Stream from Multiple Takes:

Conventional video and audio are linear media in that they contain a single sequence destined to be consumed in temporal order. Such a sequence can be obtained by concatenating segments taken at different time and/or location. Most broadcast content and its raw footage are in this category. Event semantics in these streams evident themselves not only within each shot but also in the syntactical relationship in a sequence of shots or a sequence of scenes. Shot boundaries introduce discontinuities in time and space in these streams and the reference times and locations of such discontinuities are often unknown. This unknown correspondence and the typically short shot length can prevent low-level vision algorithms, such as tracking and object segmentation, from performing strongly.

4) Media Collectives:

Real-world events are also captured in collections of loosely related media streams and objects, such as pictures from vacation trips, user-generated content around breaking news, or photo pools on community events. Each media object in the collection may be an unedited continuous capture or a produced stream. They tend to be temporally asynchronous and spatially dispersed with unknown correspondences. These collectives provide comprehensive views of the events of interest; however the appearances of different media clips are usually diverse.

Feature Representation

Feature representations are extracted from media sequences or collections, converting them into numerical or symbolic form. Such representations are convenient system representations and are fundamentals to event recognition. Good features are able to capture the perceptual saliency within the event, distinguishing it from other events. A summary of commonly used features for completeness are presented which direct the users to respective surveys on image, video, speech, and audio features. In order to structure the discussion, the features are grouped across different media types into three common categories, based on methods for computing them and their level of abstraction.

1) Low-Level Features:

Low-level features directly reflect the perceptual saliency of media signals. The procedures for computing them do not change with respect to the data collection or the event being detected. Still images are generally described in three perceptual categories, i.e., colour, texture, and shape, while image sequences introduce one more dimension of perceptual saliency, i.e., motion. Colour features are popular due to their ability to maintain strong cues to human perception with relatively less computational overhead. The main concern in reliably extracting colour information is to choose from a variety of colour spaces and achieve perceptual resemblance and colour constancy over different scene and imaging conditions. Local shapes capture obvious geometric properties in an image; this is among the most-studied image features, since psychovisual studies have showed that the human visual system performs the equivalence of edge detection. Local shapes are computed over local gray-scale or colour derivatives. Texture loosely describes an image aside from colour and local shape. It usually reflects structure and randomness over a homogeneous part of an image. Filter families and statistical models such as Gabor filters and Markov analysis are popular choices for capturing texture. Motion [7] provides information about short-term evolution in video. The 2-D motion field can be estimated from image sequences by local appearance matching with global constraints, and motion [7] can be represented in various forms of kinetic energy, such as magnitude histogram, optical flows, and motion [7] patterns in specific directions. Although colour, shape, texture, and motion can be described separately, there are features that provide integrated [13] views such as correlogram (colour and texture) or wavelets (texture and local shape).

2) Mid-Level Features and Detectors:

Mid-level features are computed using the raw signal and/or low-level features. Their computation generally involves signal- or data-domain-dependent decisions in order to manage with the change in the data domain and target semantics, and sometimes training is needed. Mid-level features capture perceptual intuitions as well as higher level semantics derived from signal-level saliency. Examples of mid-level features and detectors include: tracked objects and segmented object parts; visual concepts pertaining objects, scenes and actions, such as people, airplane, greenery. There are also mid-level features that are specific to a data domain, such as the crowd cheering detector or goal post detectors in sports videos. Features cannot only be extracted from media content; they can also come from the 5W1H in faceted metadata, i.e., structured attributes fields such as dates, location proximities, semantic distances between locations, etc.

3) Feature Aggregates for Recognition:

Feature aggregates are derived from features and detectors so that the inherent spatial-temporal structure in the media sequence can be represented as numbers/vectors/bags [9] so as to fit the data structure required by most statistical pattern recognition models. Aggregation is generally done with one or several of the following operations.

a) Accumulative statistics.

This includes histogram, moments, and other statistics over collections of points. These statistics provide simple however effective means for aggregating features over space and time. They have the advantages of being insensitive to small local changes in the content as well as being invariant [31] to coordinate shift, signal scaling, and other common transformations. The associated disadvantage is in the loss of sequential or spatial information.

b) Point selection.

The selection of possible feature detectors from candidate portions of the original signal aims to preserve perceptual saliency and provide better localization of important parts. Tracking and background substraction can be viewed as one type of selection, as well as salient parts extraction, silence detection in audio, or stop word removal.

c) Set aggregation.

This is done over the features in an image, an image sequence, audio segment, or other natural data units. The sets can be unordered or ordered, e.g., bag of words, bag-of-features, sequences, or more general graph structures.

4) Discussion about Features:

The separation made among low-level, mid-level, and feature aggregations is sometimes unclear. For example, tracking can be seen as either a mid-level feature extraction or part of the selection process. Selection and aggregation can also happen before the extraction of features, such as silence removal, stop word removal, etc. With the wide variety in feature representations, choices shall be made from domain knowledge and the event modelling task at hand, and coming up with the "optimal features" would remain an open problem.

Computational Models

In event detection, models are responsible for mapping data representations to semantic descriptions, where the descriptions are either in the forms of a discrete label or continuous states. Few observations on choosing and using models for event detection are given here.

1) Knowledge-Driven and Data-Driven Approaches:

Human perception of sensory streams is known to be both knowledge-driven and data-driven. Several well-known event recognition systems from the 1990's are mostly knowledge driven, using automaton, finite state machine, or grammar models for inference. Data-driven models range from variants of nearest neighbors to the generative and discriminative statistical models that represent complex class boundaries and encode relationships among the input and output. Nearest neighbor, or distance-based classifiers remember the primitives of known classes and classifies and then classify new examples to the nearest primitive; this has been widely used in many applications, such as face recognition or action recognition. Similar phenomena are observed with increasingly large amount of visual- and multimodal event repository being collected and made available to the research community. One example that combines knowledge and data is stochastic context-free grammar (SCFG). SCFG is initialized and weighted by data, it smoothes the HMM equivalent with nontrivial weights among unlikely or invisible parse strings and does not suffer from the lack of data to reliably estimate or even foresee unlikely paths. SCFG outperform hidden Markov models (HMM) in the human activity recognition task due to the inability of HMMs to represent a large variety of possible paths. Having weighed the pros and cons, smart combinations of knowledge and data or systematic ways to encode knowledge into data-driven models are very much desirable.

2) Generative and Discriminant Models:

Generative models produce a probability density model over all variables in a system and manipulate it to compute classification and regression functions. Discriminative models directly attempt to compute the input–output mappings for classification and regression, eschewing the modelling of the underlying distributions.

Discriminative methods-logistic regression, support vector machines (SVM), boosting-have been success in both research and practice over the last few years. Generative models - HMM, dynamic Bayesian network (DBN), linear dynamic systems - are still the models that many choose for capturing events that unfold in time. The popularity of generative models is due to two reasons: they offer to explain the data in addition to being able to complete the detection task, and they are naturally suitable to capture the structure of the data. These models with structural constraints do not suffer a search space of KL, with K the number of possible states and L the length of the sequence. Discriminative models with particularly designed feature representation and a similarity metric have also shown good detection performance in domains like computational biology and text classification. Discriminative models have also been used to model video events [3] such as story segmentation or short-term events with promising results.

3)Continuum of Supervised, Unsupervised, and Semi-Supervised Models:

A general machine learning task involves learning a mapping from input space X to output space Y: f(X) -> Y. For supervised learning, Y is known at training time, while in unsupervised learning, Y is unknown. In supervised learning, only f(X) is learned, in unsupervised learning f(X) and Y are estimated at the same time, while in semi-supervised learning a subset of Y may need to be learned together with f(X), or Y may need to be learned with certain constraints. Semi-supervised learning methods use unlabeled data to either modify or reprioritize hypotheses obtained from labeled data alone. Popular semi-supervised approaches include EM with generative mixture models, multiple instance learning, self-training, co-training, transductive support vector machines, and graph-based methods. Event detection [27] and recognition problems do not readily map to this classic supervised versus unsupervised setup, since event data are inherently structured, and both X and Y can come in different granularities. For example, a data tuple (x,y) can mean any of the following: a pixel x has label y, a region x has label y, an image x is assigned label y, at least one image x in a video sequence is assigned label y, or the entire sequence x share the label y. Therefore, the distinction of supervised and unsupervised models for event recognition is a gradually changing grayscale, rather than being black or white. The level of supervision varies depending on what kind of labeling information is available at training time. This information can include: a sequence-level label about whether an entire clip contain an event, its start/stop time, the object bounding box or parts, if it is possible or not for two events to co-occur, etc.

These diverse scenarios make various semi-supervised learning algorithms very desirable. When formulating event detection as a learning problem, deciding what to label and how the data should look can be more important than building the machinery to learn the mapping from data to label. The General components for event modelling are shown in the figure. Different types of feature extraction process can be interwoven, as can feature extraction and modelling, or data capture and feature extraction.

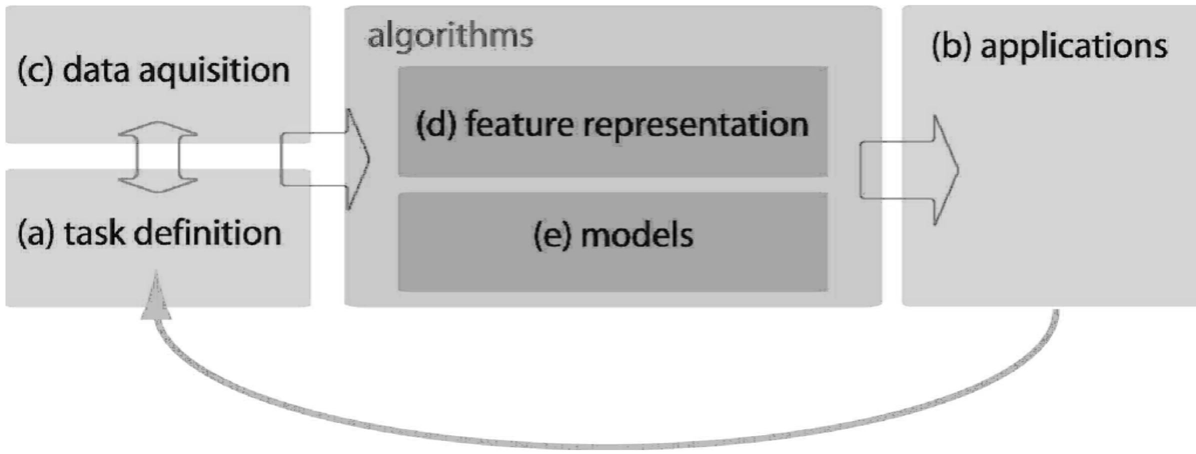


Figure 3.1: General components for event modelling

Learning similarity measures or learning to select features resides in the intersection of features and modelling. Data capture and feature extraction may be done in one shot, with embedded architecture such as smart camera systems. Implemented in hardware, such design not only improves upon software-based system on detection speed, it also makes the deployment of event detection systems easy for both everyday use and large-scale multimedia sensor networks.

3.4 OTHER RELATED WORKS

Data mining techniques have been increasingly developed to provide solutions for semantic event detection [27] in diverse types of videos. Video events [3] are normally defined as the interesting events which capture user attentions. Most current research for video event detection heavily depends on certain artifacts such as domain knowledge and priori model, and thus making them hard to be extendible to other domains or even other data sets. Though some works have been conducted to deal with the general video event extraction, they can only achieve rough detection capability due to the well-known *semantic gap* and *rare event detection* issues. The *rare event detection* issue, also known as *imbalance data set* problem, occurs in most video event detection applications. This issue is referred to as a very small percentage of positive instances versus negative instances, where the negative instances dominate the detection model training process, resulting in the degradation of the detection performance. To develop generalized event detection framework [14] applicable to different application domains, a necessary step is to relax the need of domain knowledge such as those domain-specific rules with pre-defined fixed thresholds and additional domain-based high-level features, which are often used to increase the percentage of the positive instances in the data set.

Event mining [28] is a vibrant area of research. Emerging topics of investigation include:

1. Distributed, multimodal multisource event modelling and detection, with immediate applications in analyzing surveillance and meeting recordings.
2. Event modelling and detection by explicitly modelling multiple facets also producing faceted annotation based on time and location proximity, as well as social interactions

3. Extracting faceted attributes from unstructured or semi-structured data (image, exif [22] metadata, or user-generated tags).
4. Event analysis that takes into account rich event capture and annotation - how to use and disambiguate the faceted attributes when available and how to separate the relationships among events.
5. Systematically encode and estimate domain knowledge, and use this knowledge to improve recognition, save computation, as well as to help develop systems that will generalize across different data domains.

Michael Fleischman et al. have presented a methodology for automatically learning a lexicon of hierarchical patterns of movement from unannotated video data in their paper "Mining Temporal Patterns of Movement for Video Event Recognition". These hierarchical patterns encode fine grained temporal relations and capture global information about events. In order to validate the methodology, they present a discriminative approach to event recognition in which the lexicon of hierarchical patterns is used to represent events for a tree kernel Support Vector Machine. Evaluations indicate that the patterns are informative and suggest that accurate event recognition systems may be achieved by incorporating the local information encoded in dynamic models of event recognition with the global information captured in automatically learned lexicons of hierarchical patterns of motion. The utility of hierarchical patterns does not end with video event recognition. Another area of research examines how such patterns can be used to aid human search through video data. They are examining the effectiveness of incorporating the multi-variate time series representations into a user interface in order to provide users with a static view of large amounts of dynamic video data. By colouring these time series with mined patterns users will be able to more easily perfect in on significant periods of movement, facilitating user's ability to search through large amounts of video data. In addition to data visualization, the methods described may also facilitate the learning of language to describe video events [3]. Recent work in the cognitive sciences has stressed the importance of hierarchical representations of events in models of situated word learning. The approach provides a way to learn such hierarchical structures automatically, allowing for the event representations learned from the Human Speechome Project (HSP) video data to be mapped to the speech recognized in the HSP audio data. The development of such models would advance work on the cognitive modelling of human language development as well as research on the Natural Language Interfaces for searching video data.

In their paper," Unusual Event Detection via Multi-camera Video Mining" [43] authors proposed a multi-camera mining framework for unusual event detection in surveillance video. They use two-stage training to bootstrap a probabilistic model for the usual events, and detect unusual event by thresholding the likelihood of a test event being generated by the usual event model.

In their paper, “Video event detection with combined distance-based and rule-based Data mining techniques”, [44] authors use an advanced framework that utilizes both the distance-based and rule-based data mining techniques for domain independent video event detection to address the *rare event detection* issue. Their framework is fully automatic without the need of any domain knowledge, which is achieved by data pre-processing including increasing the percentage of positive instances and reducing the feature dimension, and a decision tree classifier for event detection. The experimental results in goal event detection from multiple broadcast video data show the viability and effectiveness of the framework for general event detection.

CHAPTER 4

FILTERED VIDEO INSTANCE OPTIMIZATION

In this chapter we describe the detection of rare abnormal video events overlapped within the platform of data mining framework through geometric reconstruction♦. The supervised and unsupervised clustering instances with feature extraction will be used to support the final event detection. The significant and optimal expectation of the proposed framework can be explained through a collection containing sample datum of cricket game videos within an over. Video sequences are typically segmented into shots, where each shot represents a contiguous scene with a certain context. Automatically identifying the boundary between shots is an active area of research that has made great strides. Once a shot has been identified, it is often represented by a key frame. The key frame is then often used for other purposes such as manual or automated annotation and shot representation during a search.

4.1 VIDEO MINING - OVERVIEW

Video mining is unsupervised discovery of patterns in audio-visual content. Such unsupervised discovery is applicable to video surveillance as well as to consumer video browsing applications. Video mining can be interpreted as content-adaptive or blind content processing, in which the

♦This covers our work published "Detecting Optimization framework for rare video instance within a structured datum of filtered dataset", International Journal of Advances in Science and Technology – Vol 1, No.4, 2010,1-7.

first stage is content characterization and the second stage is event discovery based on the characterization obtained in first stage.

Video analysis [29] techniques often decompose the analysis problem into one that can be handled using the image analysis techniques. This is fine for many applications, but it may not capture temporal information inherent in video data. Approaches exist for capturing this information such as identifying distinctive features in successive video frames. A variety of standards exist to represent video data and metadata. The Motion Pictures Expert Group (MPEG) standards are probably the most well known and include MPEG-1, -2, -4, -7, -21, and others.

4.2 HARMONIC FRAMESET EXTRACTION SCHEMA

Standard machine-learning techniques can be used for analysis of multimedia data after feature extraction has been performed. There are many available machine learning techniques. In many cases multiple techniques can be combined in a modular approach to achieve more robust results. There are three general categories of analysis techniques — classification techniques, clustering techniques, and statistical techniques.

Classification techniques involve the development of models that use supervised learning to separate data into two or more discrete classes that are known in advance. Some classification techniques are artificial neural networks, Bayesian networks, Decision trees, Rule-based systems, Support vector machines.

Clustering [21] techniques involve the development of models that use unsupervised learning to separate data into two or more discrete clusters that are not known in advance. Clustering is typically performed when it is desirable to organize the data into clusters that contain similar data, but the clusters are not known in advance.

Statistical techniques involve the use of probabilities and sufficient population sample sizes to establish normal and abnormal conditions in a system and to make predictions of future events based on the current state of the system being observed. A statistical model may be as simple as one that determines the mean of a single observed parameter and flags any readings that fall outside of a certain number of standard deviations from the mean.

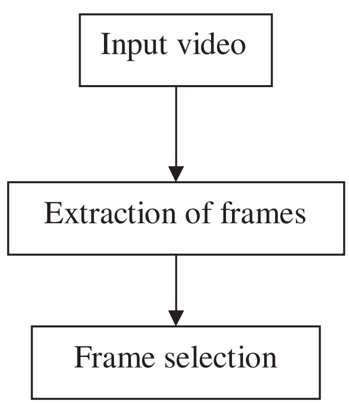


Figure 4.1: Skeleton for the proposed Model

Figure 4.2 shows the architecture of proposed data mining framework that consists of Video Parsing and Instances for learning, selection of instance, Filtered Dataset, and Final selection phases.

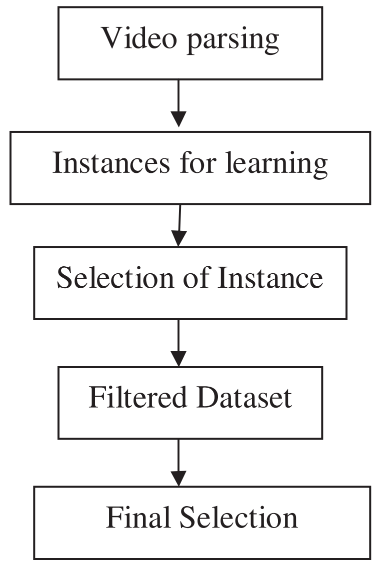


Figure 4.2: Architecture of the Proposed Model

4.3 VIDEO PARSING

Digital [8] video needs to be properly processed before it can be inserted into a video server. The processes include compressing, parsing and indexing a video sequence. Video parsing is the process of detecting scene changes or the boundaries between camera shots in a video stream. Digital video sequences are stored in a compressed format such as motion JPEG or MPEG. Content-based video indexing [36] and retrieval is a method that finds and manages the essential information of video data. It is desirable to enable viewers to get a summary of the sequence of the video data without seeing its whole sequence. An automatic video parsing is necessary for the content-based video indexing and retrieval. Video parsing involves two tasks: a video segmentation and a video indexing. The video streams are segmented into elemental units such as shots and scenes at the video segmentation stage.

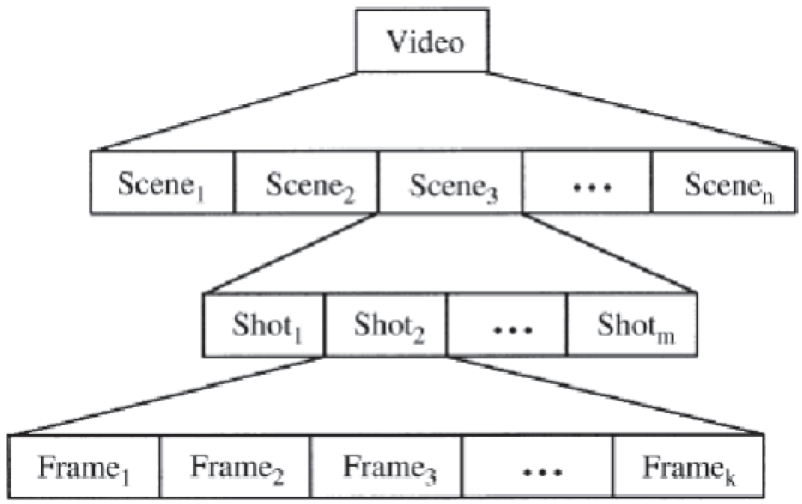


Figure 4.3: The hierarchical structure of a video

The elemental units are labeled based on their content information at the video indexing [36] stage. Generally, the video data consists of three basic units: frame, shot, and scene. A frame is an individual image. A video shot [6] can be defined as consecutive frames. Finally, a scene is comprised of a collection of one or more adjoining shots that focus on an object or several objects of interest. Shots are the basic unit for video manipulation. There are many different transitions between shots. The simplest transition is cut, but the other transitions include fade, dissolve, wipe, and so on. Camera operation analysis is useful for partitioning and indexing the video data. There are six basic camera operations. An efficient video parsing method using the shot boundary detection and the camera operation analysis is presented. For the shot boundary detection, local colour information is used. In order to reduce the computation time, an adaptive time window is utilized. Local spatio-temporal images and multilayer perceptron (MLP) are used for analyzing the camera operations. This method is reliable and fast, because it utilizes the learning algorithm with spatio-temporal information in the frame and does not have to process the entire image. Methods for shot boundary detection focus on cut detection, and on gradual transition detection.

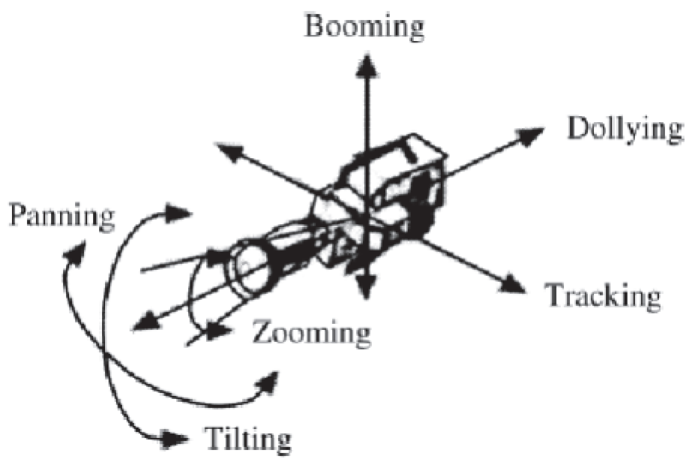


Figure 4.4: Six basic camera operations

a) Shot boundary detection

The major techniques that have been used for the shot boundary detection are pixel differences, edge difference, and histogram comparison. In these methods, if the difference between two adjacent frames exceeds a predefined threshold, then the latter frame is detected as a shot boundary.

In the easiest way to detect shot boundary, the number of pixels was counted, if the pixel's gray values are greater than a threshold. This count is compared against a second threshold to determine if the shot boundary has been found. In this method, a filter was used in order to eliminate sensitivity to camera movement and noise effects.

In the methods using edge difference, the number and position of edges in edge-detected images were compared to reduce the effects of camera motion. The percentage of edges that enter or exit between the two adjacent frames was computed. Shot boundaries were detected if the percentage exceeded the predefined threshold. This method was more accurate at detecting cuts and much less sensitive than chromatic scaling, but it needs more computing time than the other methods. The histogram comparison is the most common method used to detect shot boundaries which compute gray levels or colour histograms of the two adjacent frames. In these methods, if bin-wise difference between the two histograms of two adjacent frames exceeds a threshold, the shot boundary is detected. The histogram comparison using gray-level histogram was implemented that is very efficient for detecting shot boundaries. The above methods for the detection of shot boundaries have significant problems caused by camera motion and object movement, noise, and gradual transitions. One way to resolve these problems is to analyze the various camera operations.

b) Camera operation analysis

The method of camera operation analysis was addressed first through motion vector field analysis, by matching motion vector fields with predefined models in Hough space for various camera operations. This method was computationally expensive because it used optical flow and block matching algorithm to analyze the direction of motions in the video data. Also, it was affected by noise and the frames in sequences that were spoiled by flashlights. A camera operation analysis method using motion vectors in MPEG-type video data is used in which the predetermined prototypes of motion vectors caused by the camera operations were used. A new camera operation analysis method using X-ray images obtained by computing the average of each line and each column in successive frames was developed. The corresponding method operates on groups of *x-t* and *y-t* spatio-temporal images. An *x-t* (respectively *y-t*) spatio-temporal image is obtained by fixing the variable *y* (respectively *x*) constant in the moving image box. Up/down motions are represented in the *y-t* image and left/right motions are represented in the *x-t* image. Edge detection is first performed for all the spatio-temporal images; the intensity and the gradients of edges are calculated. Then a horizontal (respectively vertical) X-ray image is obtained by taking weighted integral for the edge images of *x-t* images (respectively *y-t* images), weighting on the intensity of edges. Some difficult problems remain in the camera operation analysis.

Discrete Cosine Transform

DCT, a derivative of DFT, employs real sinusoids as basic functions, and, when applied to natural images, has energy compaction efficiency close to the optimal KL Transform. Owing to this property and to the existence of efficient algorithms, most of the international image and video compression standards, such as JPEG, MPEG-1, and MPEG-2, rely on DCT for image compression. Because block-DCT is one of the steps of the JPEG standard and as most photographic images are in fact stored in JPEG format, it seems natural to index the DCT parameters.

a) Histogram of DC Coefficients

This technique uses the histogram of the DC image parameters as the indexing key. The construction of the DC image is illustrated in Figure 4.5.

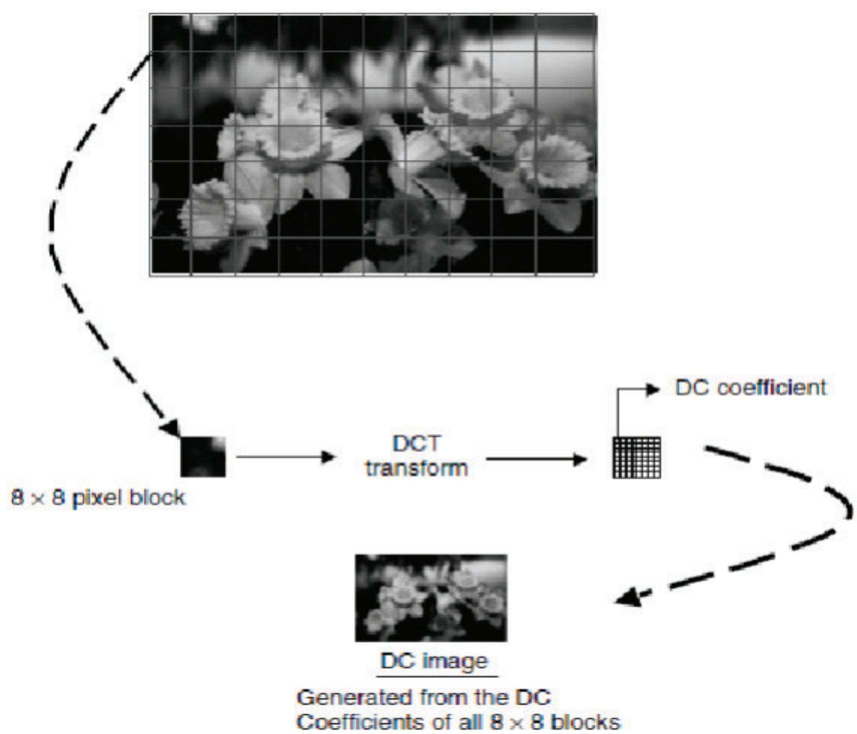


Figure 4.5: DC image derived from the original image through the DCT transform

Each image is partitioned into nonoverlapping blocks of 8×8 pixels, and each block is transformed using the two-dimensional DCT. Each resulting 8×8 block of coefficients consists of a DC value, which is the lowest frequency coefficient and represents the local average intensity, and of 63 AC values, capturing frequency contents at different orientations and wavelengths. The collection of the DC values of all the blocks is called the DC image. The DC image looks like a smaller version of the original, with each dimension reduced by a factor of 8. Consequently, the DC image also serves as a thumbnail version of the original and can be used for rapid browsing through a catalog. The histogram of the DC image, which is used as a feature vector, is computed by quantizing the DC values into N bins and counting the number of coefficients that fall within each bin. It is then stored and indexed in the database. Fifteen bins have been used to represent the colour spectrum of the DC image. These values are normalized in order to make the feature vectors invariant to scaling.

When a query is issued, the quantized histogram of the DC image is extracted from the query image and compared with the corresponding feature vectors of all the target images in the database. Similarity is typically computed using the histogram intersection method or the weighted distance between the colour histograms. The best matches are then displayed to the user. As histograms are invariant to rotation [34] and have been normalized, this method is invariant to both rotation and scaling.

b) Indexing with the Variance of the AC Coefficient

A feature vector of length 63 can be constructed by computing the variance of the individual AC coefficients across all the 8×8 DCT blocks of the image. Because natural images contain mostly low spatial frequencies, most high-frequency variances will be small and play a minor role in the retrieval. Good retrieval performance can be achieved by relying just on the variances of the first eight AC coefficients. This eight-component feature vector represents the overall texture of the entire image. The runtime complexity of this technique is smaller than that of traditional transform features used for texture classification and image discrimination.

c) Indexing with the Mean and Variance of the Coefficient

A method based on the DCT transform of 4×4 blocks produces 16 coefficients per block. The variance and the mean of the absolute values of each coefficient are calculated over the blocks spanning the entire image. This 32-component feature vector represents the texture of the whole image. A Fisher Discriminant Analysis (FDA) is used to reduce the dimensionality of the feature vector, which is then used for indexing.

d) Combining DCT Coefficients with Spatial Information

A technique for image retrieval using JPEG has been is based on the mutual relationship between the DCT coefficients of unconnected regions in both the query image and the target image. The image spatial plane is divided into even paired windows; the size of the windows is a function of the size of the image and is selected to be the smallest multiple of 8 that is less than the initial window size. These windows are randomly paired into K pairs. The average of each DCT coefficient from all the 8×8 JPEG blocks in each window is computed. The DCT coefficients of the windows in the same pair are compared. If the DCT coefficient in one window is greater than the corresponding one in the other window, a "1" is assigned; otherwise a "0" is assigned. Each window pair yields 64 binary values, and therefore each image is represented by a binary feature vector of length 64xK. The similarity of the query and target images can be determined by employing the Hamming Distance of their binary feature vectors.

e) Comparing Edges Using DCT

Many content-based indexing and retrieval methods rely on the discriminating power of edge information: similar images often have similar edge content. A technique to detect oriented line features using DCT coefficients is based on the observation that mostly horizontal, vertical, and diagonal features produce large values of DCT coefficients in vertical, horizontal, and diagonal directions, respectively. It is noted that a straight line of slope m in the spatial domain generates a straight line with a slope of approximately $1/m$ in the DCT domain. The technique can be extended to search for more complex features composed of straight-line segments.

4.4 INSTANCE LEARNING

In machine learning, instance-based learning or memory-based learning is a family of learning algorithms that, instead of performing explicit generalization, compare new problem instances with instances seen in training, which have been stored in memory. Instance-based learning is a kind of lazy learning. It is called instance-based because it constructs hypotheses directly from the training instances themselves. This means that the hypothesis complexity can grow with the data: in the worst case, a hypothesis is a list of n training items and classification takes $O(n)$. One advantage that instance-based learning has over other methods of machine learning is its ability to adapt its model to previously unseen data. Where other methods generally require the entire set of training data to be re-examined when one instance is changed, instance-based learners may simply store a new instance or throw an old instance away. A simple example of an instance-based learning algorithm is the k-nearest neighbor algorithm.

Multiple Instance Learning (MIL) is a special learning framework which deals with uncertainty of instance labels. In this setting training data is available only as pairs of bags [9] of instances with labels for the bags. Instance labels remain unknown and might be inferred during learning. A positive bag label indicates that at least one instance of that bag can be assigned a positive label. This instance can therefore be thought of as a witness for the label. Instance in negative labeled bags [9] are altogether of the negative class, so there is no uncertainty about their label.

Multi-instance (MI) classification is a supervised learning technique, but differs from *normal* supervised learning:

- it has multiple instances in an example
- only one class label is observable for all the instances in an example

Multiple-instance learning is a variation on supervised learning. Instead of receiving a set of instances which are labeled positive or negative, the learner receives a set of *bags* [9] that are labeled positive or negative. Each bag contains many instances. A bag is labeled negative if all the instances in it are negative and a bag is labeled positive if there is at least one instance in it which is positive. From a collection of labeled bags, the learner tries to induce a concept that will label individual instances correctly.

In video concept detection, most existing methods have not well studied the *intrinsic hierarchical structure* of video content. Unlike flat attribute-value data used in many existing methods, video is essentially a structured media with multi-layer representation. For example, a video can be represented by a hierarchical structure including, from large to small, *shot*, *key-frame*, and *region*. It also fits the typical Multi-Instance (MI) setting in which the "bag-instance" correspondence is embedded among contiguous layers. Such multi-layer structure and the "bag-instance" relation embedded in the structure is called as Multi-Layer Multi-Instance (MLMI). Video concept detection [1] is formulated as an MLMI learning problem in which a rooted tree with MLMI nature embedded is devised to represent a video segment. In contrast to traditional MI learning, both the Multi-Layer structure and Multi-Instance relations are leveraged simultaneously.

Using clustering techniques, the various instances can be grouped. From the available instances, required instances can be filtered to get the final event instances.

4.5 FILTERED DATASET AND VIDEO OPTIMIZATION

Categorization of videos can be achieved by exploring the concepts and meanings of the videos. This task requires bridging the gap between low-level contents and high-level concepts. Once a relationship is developed between the computable features of the video and its semantics, the user would be allowed to navigate through videos by ideas instead of the rigid approach of content matching. But, this relationship must follow the norms of human perception and abide by the rules that are most often adhered to by the creators of these videos. These rules are generally known as *Film Grammar* in video production literature. Like any natural language, this grammar also has several dialects, but is fortunately, more or less universal. The interpretation of concepts using this grammar first requires the extraction of appropriate features.

Secondly, these features or *symbols* need to be explored as in natural languages. However, the interpretation of these symbols must comply with the governing rules for video-making of a particular genre. An important aspect of this approach is to find a suitable mapping between low-level video features and their bottom-line semantics. These steps can be summarized as:

- Learn the video making techniques which are also called *Film Grammar*.
- Learn the theories and practices of film aesthetics, such as the effect of colour on the mood, the effect of music on the scene situation and the effect of post processing of the audio and video on human perception.
- Develop a model to integrate this information to explore concepts.
- Provide users with a facility to navigate through the audiovisual data in terms of concepts and ideas.

This framework is represented in Figure 4.6

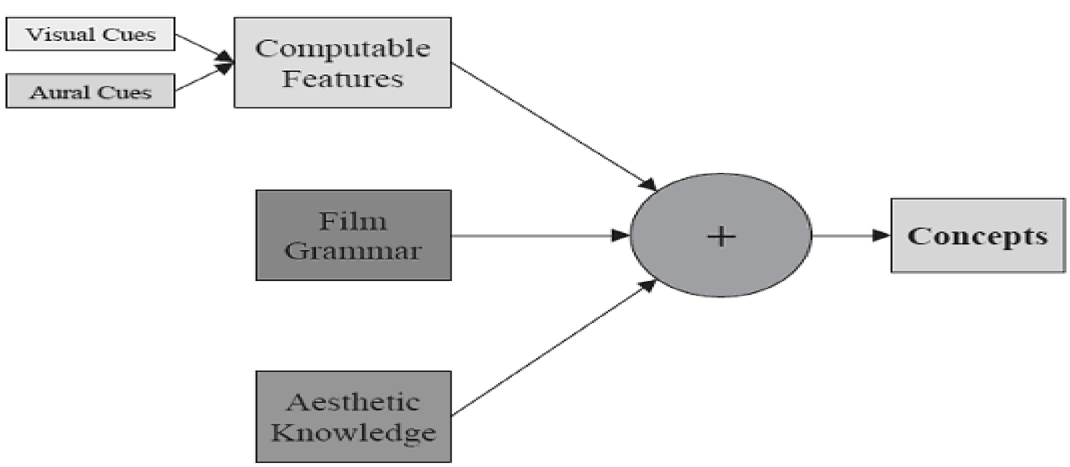


Figure 4.6: Video optimization- approach.

Film Structure

There is a strong analogy between a film and a novel. A *shot*, which is a collection of coherent image frames, is similar to a *word*. A number of words make up a *sentence* as shots make visual thoughts, called *beats*. Beats are the representation of a subject and are collectively referred to as a *scene* in the same way that sentences collectively constitute a *paragraph*. Scenes create *sequences* like paragraphs make *chapters*. Finally, sequences produce a *film* when combined together as the chapters make a *novel*. This final audiovisual product, i.e. the film, is our input and the task is to extract the concepts within its small segments in a bottom-up fashion. Here, the ultimate goal is to decipher the meaning as it is perceived by the audience.

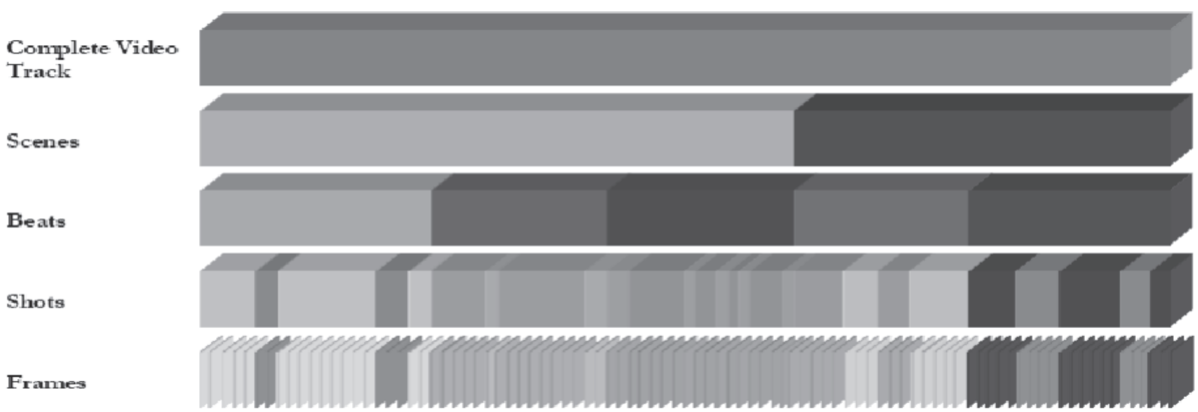


Figure 4.7: A film structure

Computable Features of a Visual Data

We define computable features of an audio-visual data as a set of attributes that can be extracted using image/signal processing and computer vision techniques. This set includes, shot boundaries, shot length, shot activity, camera motion, colour characteristics of image frames as video features. These features and methods to compute them are given here.

Shot Detection

A shot is defined as a sequence of frames taken by a single camera with no major changes in the visual content. Modified version of the colour histogram intersection was used. For each frame, a 16-bin HSV normalized colour histogram is estimated with 8 bins for hue and 4 bins each for saturation and value.

Key frame Detection

Key frames are used to represent the contents of a shot. Choosing an appropriate number of key frames is difficult since we consider a variety of videos including feature movies, sitcoms and interview shows, which contain both action and non-action scenes. Selecting one key frame may represent a static shot quite well; however, a dynamic shot may not be represented adequately. Therefore, we have developed a method to select variable number of key frames depending upon the shot activity. Each shot, S_i, is represented by a set of key frames, K_i, such that all key frames are distinct. Initially, the middle frame of the shot is selected and added to the set K_i as the first key frame. The reason for taking the middle frame instead of the first frame is to make sure that the frame is free from shot transition effects, for instance, a *diffusion* effect. Next, each frame within a shot is compared to every frame in the set *Ki*. If the frame differs from all previously chosen key frames by a fixed threshold, it is added in the key frame set, otherwise it is ignored.

Shot Length and Shot Motion Content

Shot length (the number of frames present in a shot) and shot motion content are two interrelated features. These features provide cues to the nature of the scene. For a given scene, these two attributes are generally consistent over time to maintain the pace of the movie.

Computation of Shot Motion Content

Motion in shots can be divided into two classes; *global motion* and *local motion*. Global motion [7] in a shot occurs due to the movements of the camera. These may include *pan* shots, *tilt* shots, *dolly/truck* shots and *zoom in/out* shots. Local motion is the relative movement of objects with respect to the camera, for example, an actor walking or running. Shot motion content is defined as the amount of local motion in a shot and exploits the information encoded in MPEG-1 compressed video to compute it. The horizontal and vertical velocities of each block are encoded in the MPEG stream. These velocity vectors reflect the global or local motion. Estimate the global affine motion using a least squares method. The goodness of the fit is measured by examining the difference between the actual and reprojected velocities of the blocks. The magnitude of this error is used as a measure of shot motion content.

4.6 EXPERIMENT AND RESULTS

Here we use a cricket video for our experimental analysis, where the sixes, fours, wickets are to be focused. A computer-implemented process is used for performing video concept detection on a video clip based upon a prescribed set of target concepts. The technique comprising process actions of segmenting the clip into a plurality of shots, wherein each shot comprises a series of consecutive frames that represent a distinctive coherent visual theme is used. It constructs a multi-layer multi-instance (MLMI) structured metadata representation of each shot, that are classified into three segmentations.

Segment one: Focuses on all frames of the clip.

Segment two: Focus on frames related to threshold frame.

Segment three: Focuses on particular target shot.

It can be represented through a tree structure in which a filtered scheme can be utilized.

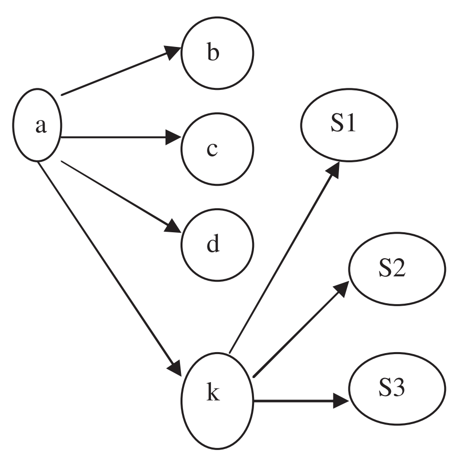


Figure 4.8 Tree structure

The nodes of the tree structure represent the following: a is root node representing the frames. Node – b represent the frame with no bat –ball touch. Node - c represent the frame with all ground shot bat ball touch. Node – d represent the frame with all short air shot bat-ball touch (expected catch). Node - k represent the frame with all long air shot bat-ball touch (expected six). S1 represents Air shot with non optimal angle and here the runs may 4/3/2/1. S2 represents Air shot with possible angle with or without fielder end (out-wicket). S3 represents Air shot with an optimal angle and force with or without fielders/unable to catch and here the run is six. The objective of this research is to focusing on all k-shots. We can visualize this as an Optimality function as follows,

Maximize

$$F(x) = ((\sum_{i=1}^{6} a_i \sum_{j=1}^{3} S_j)/6) *100$$

Where

considering a single over

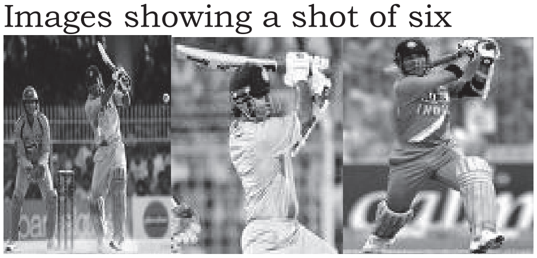
a_i =1 if run is six 0 otherwise

S_1 = 1 if run is 4/3/2/1, 0 otherwise

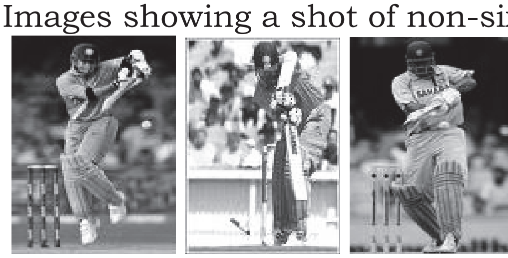
S_2 = 1 if out, 0 otherwise

S_3 = 1 if run is six, 0 otherwise

Images showing a shot of six



Images showing a shot of non-six



From the process described above we are able to locate the required output that is the frame containing sixes, fours, and wickets which are of importance in the game.

4.7 DISCUSSION

Considering the cricket video our goal is Detection of a six. Hundred cricket videos are selected from internet including world cup whose duration is 4 hrs. From the analysis we find out that there only 18 sixes. Therefore N1, the positive instances (six) is 18 and N2, negative instances (non six) is 82 and hence the Success instance is 18%. About 20 times random selection, the occurrences of Success instances is 20% approximately (six) and 80% non six.

Table 4.1: Cross validation result

No	Six	Ident.	Missed	Mis-Ident	RC	PR
1	8	6	2	2	75	75
2	8	5	3	3	80	80
3	8	7	1	1	93	93
4	8	8	0	2	100	80
5	8	7	1	3	93	70
6	7	6	1	3	93	66
7	7	7	0	2	100	71
8	7	5	2	3	86	62
9	7	4	3	4	60	50
10	7	6	1	2	93	75

Table 4.1 shows the event detection performance of proposed data mining framework, where “RC” , “PR”, and “Ident.” denote “Recall”, ”Precision”, and “Identified” respectively. The “Missed” column indicates the number of six instances that are misclassified as non six, and the “Mis-Ident.” column indicates the number of non six instances that are misclassified as six instances. The Recall and Precision for the six events can be defined as:

Recall = Identified / (Identified + Missed);

Precision = Identified / (Identified + Mis-ident).

The World cup cricket 2011 final match between India and Srilanka is the collection of unstructured video for our sample datum. According to our approach videos can be analyzed through frames. The first stage is identification of valid frames. The second stage is classifying the frame based on its unstructured behaviour. According to behaviour anomaly, the frames are clustered and named. The recognition of clusters can be mapped with behavioural similarities. A sample output screen of the video optimization is given below:

Figure 4.9: Output Screen shot

CHAPTER 5

EVENT MINING ANALYSIS OF UNSTRUCTURED DATA

In this Chapter we show an outline for handling unstructured datum♦ through the conversion of layered structured datum with its various levels of implications.

5.1 EVENT MINING

The semantics about events is essential in real time decision analysis and support. In earlier stages different approaches have been developed to build models and logical algorithms that relied on the concept of events. Nowadays several modelling techniques are available in processing events with its mining scenarios, but the process of handling unstructured datum in Event mining is a little bit tedious approach. Event mining means the following

1. Design of an event model
2. Choice of representations
3. Measurement, primary evaluation, and generation of representations
4. Secondary evaluation, aggregation of system knowledge to achieve a knowledge base

---

♦This covers our work published "Semantic Investigation of Unstructured Datum on Event Mining Analysis", International Journal of Engineering Science and Technology. - Vol. 2 (7), 2010, 2763-2769.

5. Application of the knowledge base

An event model describes the type of events that can occur in a system, and their attributes. Event models should be defined in such a way that the can be used for the set up of measurements in real world systems, which deliver formal representations of event streams taking place there.

Three types of events are internal events, send and receive events, and remote events. Since events are always defined with respect to system architecture, a generic modelling framework for distributed systems [10] is needed, which supports the definition of event streams. The choice of such a framework depends on the planned exploitation of actual event model, and in particular of the resulting, measured event streams.

Forms of Event Media Data

Multimedia archives are snapshots of real-world events from capturing, editing, and archiving with limited metadata and annotation. The setup for media capture imposes limits on what are available to us for data analysis; it also puts constraints for system design. Few scenarios and implicit but important assumptions are given here.

- Single Stream from One Continuous Take
- Multiple Concurrent Streams
- Single Stream from Multiple Takes
- Media Collectives

5.2 TRANSITION AND MODELLING

In event detection, models are responsible for mapping data representations to semantic descriptions, where the descriptions are either in the forms of a discrete label or continuous states. A few observations on choosing and using models for event detection are given here.

1) Knowledge-Driven and Data-Driven Approaches:

Human perception of sensory streams is known to be both knowledge-driven and data-driven. Several well-known event recognition systems are mainly knowledge driven, using automaton, finite state machine, or grammar models for inference. Data-driven models range from variants of nearest neighbors to the generative and discriminative statistical models that represent complex class boundaries and encode relationships among the input and output.

2) Generative and Discriminant Models:

Generative models produce a probability density model over all variables in a system and manipulate it to compute classification and regression functions. Discriminative models directly attempt to compute the input–output mappings for classification and regression, eschewing the modelling of the underlying distributions.

3)Continuum of Supervised, Unsupervised, and Semi-Supervised Models:

A general machine learning task involves learning a mapping from input space X to output space Y. For supervised learning, Y is known at training time, while in unsupervised learning, Y is unknown. In supervised learning, only f(x) is learned, in unsupervised learning f(x) and Y are estimated at the same time, while in semi-supervised learning a subset of Y may need to be learned together with f(x), or Y may need to be learned with certain constraints. Semi-supervised learning methods use unlabeled data to either modify or reprioritize hypotheses obtained from labeled data alone.

Event Detection in Edited Sequences

The meanings of a continuous, edited multimedia sequence reside both in each shot and in the syntactic relationships among adjacent shots created by the director. The most prevalent forms of such content are feature films and television broadcast archives, where generic and domain-specific events are useful for indexing, search, and summarization. A broad definition of events in produced videos includes two categories: those resulting from video production (i.e., camera or editing operations), such as shot boundaries, scene change; and those are inherent in the video content, such as changes in objects, settings, or topics.

1) Detecting Production Events:

Detection of video production effects is a natural first step towards breaking down the video understanding problem, and it has received considerable attention since the beginning of multimedia analysis. Shot boundary detection is detected as a change in colour, texture, or motion features. Several shots with consistent location and ambient sound constitute a scene. Scene changes in films can be inferred with features related to chromacity, lighting, audio features, coherence, and memory models.

2) Detecting Content Semantics in Produced Videos:

Domain-specific events inherent in video content can be further categorized into regular patterns or spontaneous events. Detection of spontaneous events from films or TV drama, such as explosion, clapping, and waterfalls has been done with global audio and video features and a probabilistic factor graph, or in a interactive frame work that helps the user to label a subset of the attributes. For generic produced videos, little can be assumed regarding the camera, scene, or objects, due to the frequent [19] transition among shots and the large variations in the scene and imaging conditions. Therefore, the analysis systems typically resort to global content features or generic mid-level features such as colour histogram, correlogram, visual, and audio classes.

Unsupervised Event Discovery

Most of the work in the previous sections detects known events. Automatically detecting unknown events can also be very useful when the user needs to explore a new collection, find new things that are unaccounted for among the set of known events, or initialize models and data annotations for more accurate modelling. This scenario has received considerable recent attention, in part because media collections have outgrown the amount of reliable annotation.

There is a large variety of problems and solutions in this area, although it is relatively new. The problems are in two categories, namely, finding regular events and/or unusual [42] events. The computational models typically involve clustering algorithms [2], association and co-occurrence mining, and dynamic graphical models, in combination with outlier-identification and model adaptation. The data domains span many raw and produced content types, such as broadcast news, sports, surveillance, lifelog, meetings, etc. Regular patterns [24] are typically found using clustering operations with various features and models. A continuous media sequence can be either presegmented into fixed length units or jointly clustered and segmented by generative models typically in the HMM/DBN family.

While event discovery usually relies on unsupervised approaches, the separation between supervised and unsupervised is, in fact, gray-scale. Here, supervision can mean knowing the onset/offset of events but not knowing the location and action of objects or knowing objects and scenes but not knowing their actions and interactions, while being unsupervised will still require domain knowledge and clever feature engineering to steer the discovery towards meaningful directions.

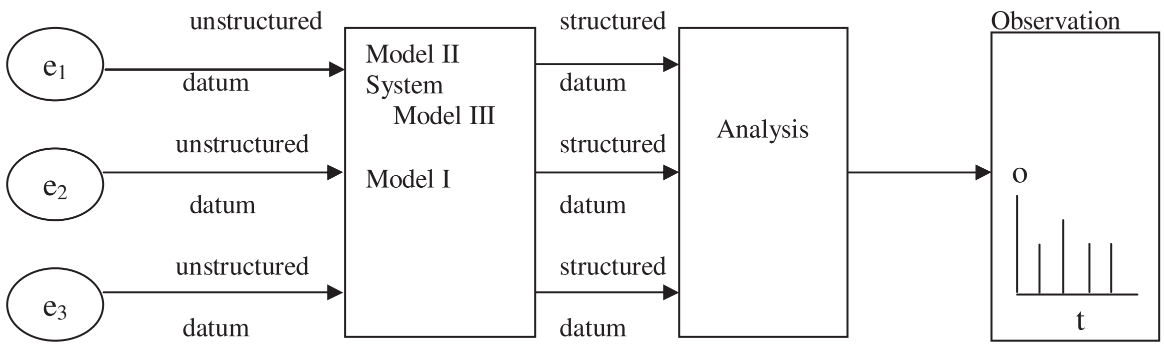


Figure 5.1: Proposed Model for handling unstructured datum

System Model 1:

Events implied by changes in observation.

Ex. Vehicle moves

Observation => green signal

System Model 2:

Future events based on changes in observation.

Ex. If signal = red => vehicle will stop

System Model 3:

Future observation based on changes of events

Ex. If more vehicle on one side then signal will be green soon.

Event Mining System:

$E' = \{ O, T, \delta, E, e_0 \}$

O -> set of all observations

T -> Time intervals

E -> Event states as a set

e_0 -> Initial event

δ -> Transition function

$\delta : OXT \rightarrow E$

where

$O = \{ O_1, O2, \ldots, On\}$

$T = \{ t_1, t2, \ldots , tn\}$

5.3 CLUSTERING

Cluster analysis is the process of grouping objects into subsets that have meaning in the context of a particular problem. The objects are thereby organized into an efficient representation that characterizes the population being sampled. Unlike classification, clustering does not rely on predefined classes. Clustering is referred to as an unsupervised learning method because no information is provided about the "right answer" for any of the objects. It can uncover previously undetected relationships in a complex data set. Many applications for cluster analysis exist. For example, in a business application, cluster analysis can be used to discover and characterize customer groups for marketing purposes.

Two types of clustering algorithms [2] are nonhierarchical and hierarchical. In nonhierarchical clustering, such as the k-means algorithm, the relationship between clusters is undetermined. Hierarchical clustering repeatedly links pairs of clusters until every data object is included in the hierarchy. With both of these approaches, an important issue is how to determine the similarity between two objects, so that clusters can be formed from objects with a high similarity to each other. Commonly, distance functions, such as the Manhattan and Euclidian distance functions, are used to determine similarity. A distance function yields a higher value for pairs of objects that are less similar to one another. Sometimes a similarity function is used instead, which yields higher values for pairs that are more similar.

Distance Functions

Given two p-dimensional data objects i = (x_{i1},x_{i2}, …,x_{ip}) and j = (x_{j1},x_{j2}, …,x_{jp}), the following common distance functions can be defined:

Euclidian Distance Function:

$$d(i,j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \ldots + |x_{ip} - x_{jp}|^2}$$

Manhattan Distance Function:

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}|$$

k-means Algorithm

The *k*-means algorithm is one of a group of algorithms called *partitioning methods*. The problem of partitional clustering can be formally stated as follows: Given *n* objects in a *d*-dimensional metric space, determine a partition of the objects into *k* groups, or clusters, such that the objects in a cluster are more similar to each other than to objects in different clusters. Recall that a partition divides a set into disjoint parts that together include all members of the set. The value of *k* may or may not be specified and a clustering criterion, typically the *squared-error criterion*, must be adopted.

The solution to this problem is straightforward. Select a clustering criterion, then for each data object select the cluster that optimizes the criterion. The *k*-means algorithm initializes *k* clusters by arbitrarily selecting one object to represent each cluster. Each of the remaining objects is assigned to a cluster and the clustering criterion is used to calculate the cluster mean. These means are used as the new cluster points and each object is reassigned to the cluster that it is most similar to. This continues until there is no longer a change when the clusters are recalculated.

Hierarchical algorithms can be either agglomerative or divisive, that is top-down or bottom-up. All *agglomerative hierarchical clustering algorithms [2]* begin with each object as a separate group. These groups are successively combined based on similarity until there is only one group remaining or a specified termination condition is satisfied. For *n* objects, *n*-1 mergings are done. *Hierarchical algorithms* are rigid in that once a merge has been done, it cannot be undone. Although there are smaller computational costs with this, it can also cause problems if an erroneous merge is done. As such, merge points need to be chosen carefully.

BIRCH

Many clustering algorithms [2], including *k*-means, require an access to the entire data set. When the data set is very large and does not fit into available memory one has to “squash” the dataset to make applications of *k*-means-like algorithms possible. The Balanced Iterative Reducing and Clustering algorithm (BIRCH) is a clustering algorithm designed to operate under the assumption "the amount of memory available is limited, whereas the dataset can be arbitrary large". The algorithm does the "squashing," or generates "a compact dataset summary" minimizing I/O cost involved in clustering [21] the dataset. BIRCH thus reduces the problem of clustering the original very large data set into the one of clustering the set of "summaries" which has the potential to be much smaller.

5.4 EXPERIMENT AND RESULTS

Consider the real time scenario of a shopping mall which contains 10 floors of which each floor can be allotted for different business purposes. The datum associated with this shopping mall is clearly an unstructured datum; we can convert this into a layered structured datum and then handle it with an optimal option. The algorithm used for clustering the dataset here is the BIRCH algorithm.

Table 5.1: Floor details in the shopping mall

Floors	Details
F10	Hotel, Ice cream Parlour
F9	Cosmetics
F8	Electronics – A/C, Laptops etc
F7	Share consultancy
F6	Jewellery
F5	Young gents
F4	Young ladies
F3	Kids
F2	Old women
F 1	Old men
GFloor	Entrance and Car parking

Data Classification & Clustering:

Consider members present in shops as family.

Table 5.2: Category table

<table>
<tr><td>1</td><td>Category – 1 – Large family:
Old men >= 4
Old women >= 4
Young ladies >= 6
Young Gents >= 6
Kids >= 8</td><td>6</td><td>Category – 6 – Unit family:
Young Gents =1
Young ladies = 1
Kid = 1</td></tr>
<tr><td>2</td><td>Category – 2 – Medium family:
2 < Old men < 4
2 < Old women < 4
3 < Young ladies < 6
3 < Young Gents < 6
4 < Kids < 8</td><td>7</td><td>Category – 7 - Isolated family:
Young Gents = 1 or
Young ladies = 1
Kids <= 2</td></tr>
<tr><td>3</td><td>Category – 3 – Small family:
Old men < 2
Old women < 2
Young ladies < 3
Young Gents < 3
Kids < 4</td><td>8</td><td>Category – 8 Component family:
a) Old men or old women or both >=2
b) Young girl or Young lady or both >=2
c) Old men or old women or both >=2 and kids >=1
d) Young gent or young lady or both >=2 and kids >=1
e) Old men or old women or both >=2 and kids >=1</td></tr>
<tr><td>4</td><td>Category – 4 – Ideal family:
Old men = 2
Old women = 2
Young ladies = 1
Young Gents = 1
Kids = 2</td><td>9</td><td rowspan="2">Category – 9 - Abnormal family:
a) kids >=1
b) Young gents >=1
c) Young ladies >=1
d) Old men >=1
e) Old women >=1</td></tr>
<tr><td>5</td><td colspan="2">Category – 5 – Classical Family:
Young Gents = 1
Young ladies = 1
Kids > 2</td></tr>
</table>

These 9 categories are put into several clusters as

Clusters = {Old men, old women, Young lady, Young gent, Kid}

Family comprises the following mixture, so that any family can fall within these sets irrespective of count

Table 5.3: Clusters table

1) Old man	17) old men, old women, young gent
2) Old woman	18) old men, old women, kids
3) Young lady	19) old men, young lady, young gent
4) Young gent	20) old men, young lady, kids
5) kids	21) old men, young gent, kids
6) old men, old women	22) old women, young lady, young gent
7) old men, young lady	23) old women, young lady, kids
8) old men, young gent	24) old women, young gent, kids
9) old men, kids	25) young lady, young gent, kids
10) old women, young lady	26) old men, old women, young lady, young gent
11) old women, young gent	27) old men, old women, young lady, kids
12) old women, kids	28) old men, old women, young gent, kids
13) young lady, young gent	29)old men, young lady, young gent, kids
14) young lady, kids	30) old women, young lady, young gent, kids
15) young gent, kids	31) old men, old women, young lady, young gent, kids
16) old men, old women, young lady	

1 day -> 15 hours working – > 8am to 11 pm

Table 5.4: Cluster Evaluation table

CL1 -> upto Rs.500	CL17 -> upto Rs.2000
CL2 -> upto Rs.500	CL18 -> upto Rs.1500
CL3 -> upto Rs.2000	CL19 -> upto Rs.3500
CL4 -> upto Rs.1000	CL20 -> upto Rs.3000
CL5 -> upto Rs.500	CL21 -> upto Rs.2000
CL6 -> upto Rs.1000	CL22 -> upto Rs.3500
CL7 -> upto Rs.2500	CL23 -> upto Rs.3000
CL8-> upto Rs.1500	CL24 -> upto Rs.2000
CL9 -> upto Rs.1000	CL25 -> upto Rs.3500
CL10 -> upto Rs.2500	CL26 -> upto Rs.4000
CL11 -> upto Rs.1500	CL27 -> upto Rs.3500
CL12 -> upto Rs.1000	CL28 -> upto Rs.2500
CL13 -> upto Rs.3000	CL29 -> upto Rs.4000
CL14-> upto Rs.2500	CL30 -> upto Rs.4000
CL15 -> upto Rs.1500	CL31 -> upto Rs.4500
CL16 -> upto Rs.3000	

$E = \sum ctf_i * clfj$ [Exp = Category factor * cluster factor]

i^{th} category with j^{th} cluster

Table 5.5: Pivot table

Category values	Pivotal value
ct_1	5
ct_2	4.5
ct_3	4
ct_4	3.5
ct_5	3
ct_6	2.5
ct_7	2
ct_8	1.5
ct_9	1

Event Mining Table:

A sample table on a Sunday.

Table 5.6: Category evaluation table1

S.No.	Category	Units
1	ct_1	11
2	ct_2	6
3	ct_3	8
4	ct_4	4
5	ct_5	7
6	ct_6	9
7	ct_7	2
8	ct_8	5
9	ct_9	1

Table 5.7: Category – Cluster relation table1

S. No.	Category	Clusters	Total Amount (Rs.)
1	ct_1	CL31,CL30,CL29,CL28,CL27,CL24,CL23,CL22,CL31,CL30,CL30	39500
2	ct_2	CL31,CL30,CL28,CL26,CL20	18000
3	ct_3	CL18,CL21,CL22,CL23,CL21,CL22,CL18,CL24	19000
4	ct_4	CL16,CL15,CL17,CL16	9500
5	ct_5	CL3,CL9,CL6,CL5,CL7,CL8,CL7	11000
6	ct_6	CL3,CL8,CL2,CL1,CL6,CL4,CL2,CL9,CL8	9500
7	ct_7	CL2,CL9	1500
8	ct_8	CL3,CL2,CL1,CL2,CL1	4000
9	ct_9	CL1	500

A sample table on a weekday.

Table 5.8: Category evaluation table2

S.No.	Category	Units
1	ct_1	8
2	ct_2	3
3	ct_3	4
4	ct_4	2
5	ct_5	4
6	ct_6	4
7	ct_7	2
8	ct_8	2
9	ct_9	1

Table 5.9: Category – Cluster relation table2

S. No.	Category	Clusters	Total Amount Rs.)
1	ct_1	CL31,CL30,CL29,CL28,CL27,CL24,CL23,CL22	27000
2	ct_2	CL31,CL30, CL20	11500
3	ct_3	CL18,CL21,CL22,CL23	10000
4	ct_4	CL16,CL15	4500
5	ct_5	CL3,CL9,CL6,CL5	4500
6	ct_6	CL3,CL8,CL2,CL1	4500
7	ct_7	CL2,CL9	1500
8	ct_8	CL3,CL1	2500
9	ct_9	CL1	500

Using Data Classification and clustering, the customers and the sales of products have been analyzed. The analysis shows that the sales of products are characterized by the nature of customers, special offers, religious and family functions, advertisements etc.

5.5 DISCUSSION

We analyze the sales for the shopping mall with various versions in a month. We obtain the results as surprising objectives. If we take the normal sales percentage as 100% then there is a noticeable increase in sales percentage on Government holidays and religious functions.

Similarly there is a obvious increase in sales during Sundays, Marriage and Aadi(Seasonal) period. Specific events such as Fathers day, Advertisements and stock clearance also make a rise in sales, whereas abnormal events (Bandh) and month end diminishes the sales. Hence we can easily perform Data Classification and clustering in a unstructured datum in which the customers and the sales of products have been analyzed in an efficient manner.

CHAPTER 6

HEURISTICAL TECHNIQUES OF EVENT MINING ON UNSTRUCTURED DATUM

In this chapter we deal with a discrete event system which produces unstructured datum♦ during its functional approach. We handle the unstructured datum towards the layered structured system with its various levels of impacts and implications.

6.1 EVENT MINING - OVERVIEW

Events are essential factors in the world of data mining [25]. Hybrid dynamic systems, stochastic systems and discrete event system are some of the entities that deal with unstructured datum in event mining systems. Event mining [28] is a challenging area of research and applications. Events are especially challenging for real-time analysis. A key to understanding events is to know what caused them and having that knowledge at the time the events happen. Another issue is the knowledge about the consequences of events. The ability to track event and consequences is an essential step toward on-line decision support and an important challenge for new algorithms for event mining [35]. Many existing enterprise systems are distributed and event-driven. Events might be described by structured and un-structured information.

♦ This covers our work published "An Approach for Handling Unstructured Datum on Event Mining with Heuristical Techniques", International Journal of Engineering Science and Technology- Vol. 2 (7), 2010, 2770-2773.

The structured information is well recognized and is stored in databases. Environmental scanning is a new term and it means the acquisition and use of the information about events, trends, and relationships in an external environment. Therefore, the methods of dealing with unstructured information about events are especially important.

Event mining might have various applications. On the business level it is the business activity monitoring (BAM). BAM is defined as a concept that provides a real-time access to critical business performance indicators to improve the speed and effectiveness of business operations. BAM involves alerts, triggers, sensors, and agents that determine a transaction or event that is meaningful. Computer networks produce a large amount of event-based data that can be collected for network analysis. These data include alerts from firewalls and intrusion detection systems (IDS), log files [38] of various software systems, routing information from the Internet and so on.

6.2 HEURISITC ANALYSIS OF ANTIVIRUS SYSTEM

Intrusion detection is the act of detecting unwanted traffic on a network or a device. An IDS can be a piece of installed software or a physical appliance that monitors network traffic in order to detect unwanted activity and events such as illegal and malicious traffic, traffic that violates security policy, and traffic that violates acceptable use policies. Many IDS tools will also store a detected event in a log to be reviewed at a later date or will combine events with other data to make decisions regarding policies or damage control. An IPS is a type of IDS that can prevent or stop unwanted traffic. The IPS usually logs such events and related information.

Technologies

Several types of IDS technologies exist due to the variance of network configurations. Each type has advantages and disadvantage in detection, configuration, and cost.

a) Network-Based

A Network Intrusion Detection System (NIDS) is one common type of IDS that analyzes network traffic at all layers of the Open Systems Interconnection (OSI) model and makes decisions about the purpose of the traffic, analyzing for suspicious activity. Most NIDSs are easy to deploy on a network and can often view traffic from many systems at once. A term becoming more widely used by vendors is “Wireless Intrusion Prevention System” (WIPS) to describe a network device that monitors and analyzes the wireless radio spectrum in a network for intrusions and performs countermeasures.

b) Wireless

A wireless local area network (WLAN) IDS is similar to NIDS in that it can analyze network traffic. However, it will also analyze wireless-specific traffic, including scanning for external users trying to connect to access points (AP), rogue APs, users outside the physical area of the company, and WLAN IDSs built into APs. As networks increasingly support wireless technologies at various points of a topology, WLAN IDS will play larger roles in security. Many previous NIDS tools will include enhancements to support wireless traffic analysis.

c) Network Behaviour Anomaly Detection

Network behaviour anomaly detection (NBAD) views traffic on network segments to determine if anomalies exist in the amount or type of traffic. Segments that usually see very little traffic or segments that see only a particular type of traffic may transform the amount or type of traffic if an unwanted event occurs. NBAD requires several sensors to create a good snapshot of a network and requires benchmarking and baselining to determine the nominal amount of a segment's traffic.

d) Host-Based

Host-based intrusion detection systems (HIDS) analyze network traffic and system-specific settings such as software calls, local security policy, local log audits, and more. A HIDS must be installed on each machine and requires configuration specific to that operating system and software.

Detection Types

a) Signature-Based Detection

IDS can use signature-based detection, relying on known traffic data to analyze potentially unwanted traffic. This type of detection is very fast and easy to configure. However, an attacker can slightly modify an attack to render it undetectable by a signature based IDS. Still, signature-based detection, although limited in its detection capability, can be very accurate.

b) Anomaly-Based Detection

IDS that looks at network traffic and detects data that is incorrect, not valid, or generally abnormal is called anomaly-based detection. This method is useful for detecting unwanted traffic that is not specifically known. For instance, anomaly-based IDS will detect that an Internet protocol (IP) packet is malformed. It does not detect that it is malformed in a specific way, but indicates that it is anomalous.

c) Stateful Protocol Inspection

Stateful protocol inspection is similar to anomaly based detection, but it can also analyze traffic at the network and transport layer and vender-specific traffic at the application layer, which anomaly-based detection cannot do.

False Positives and Negatives

It is impossible for IDS to be perfect, primarily because network traffic is so complicated. The erroneous results in IDS are divided into two types: false positives and false negatives. False positives occur when the IDS erroneously detects a problem with benign traffic. False negatives occur when unwanted traffic is undetected by the IDS. Both create problems for security administrators and may require that the system be calibrated. A greater number of false positives are generally more acceptable but can burden a security administrator with cumbersome amounts of data to sift through. However, because it is undetected, false negatives do not afford a security administrator an opportunity to review the data.

System Components

IDSs are generally made up of the following main types of components:

1. Sensors—These are deployed in a network or on a device to collect data. They take input from various sources, including network packets, log files, and system call traces. Input is collected, organized, and then forwarded to one or more analyzers.
2. Analyzers—Analyzers in an IDS collect data forwarded by sensors and then determine if an intrusion has actually occurred. Output from the analyzers should include evidence supporting the intrusion report. The analyzers may also provide recommendations and guidance on mitigation steps.
3. User interface—The user interface of the IDS provides the end user a view and way to interact with the system. Through the interface the user can control and configure the system. Many user interfaces can generate reports as well.
4. Honeypot—In fully deployed IDS, some administrators may choose to install a "honeypot," essentially a system component set up as bait or decoy for intruders. Honeypots can be used as early warning systems of an attack, decoys from critical systems, and

data collection sources for attack analyses. Many IDS vendors maintain honeypots for research purposes, and to develop new intrusion signatures. Note that a honeypot should only be deployed when the organization has the resources to maintain it. A honeypot left unmanaged may become a significant liability because attackers may use a compromised honeypot to attack other systems.

6.3 LAYERED DATA

Layered video coding was proposed as an attractive and efficient solution for the problem of video adaptation in heterogeneous multicast sessions. Using the MPEG video coding, the compressed data is composed of three types of frames - Intraframe coding (I-frame), Predictive coding (P-frame) and Bidirectional predictive coding (B-frame). I-frames are independent frames compressed using only intraframe compression. These frames are self sufficient and do not need previous or future information in order to be decoded. Iframes can be ten times bigger than other frames as they are used as references. P-frames carry the motion difference to the previous I or P frames. This coding method uses the fact that macro blocks in subsequent pictures are highly similar and usually do not change but move. B-frames are encoded based on the previous and the next I or P frame. In layered coding approaches, the raw video is encoded into a set of cumulative (or non-cumulative) layers. A basic layer contains the essential video information with a basic low quality and a set of enhancement layers are used to improve the quality of the received video. In the case of MPEG-4coding, layers can be obtained by applying three different types of scaling: Temporal scaling, Spatial scaling or Fine Granularity Scaling (FGS). In temporal scaling, the base layer carries only P or I frames. The enhancement layer adds B-frames and then increases the frame rate. Spatial scaling allows the base layer to carry a low resolution version of the video. Enhancement layer provides full resolution B-frames and then allows the decoder to display a full resolution picture. In layered multicast video, the set of layers are sent over separate multicast groups. A receiver joins, first, the basic layer to obtain the basic video quality and then adapts the rate depending on its capabilities by joining/leaving enhancement layers. Depending on the set of joined layers, heterogeneous receivers will obtain different video quality. Multi-layer approach supports video data. In multiple layer approach, layered video coding is the key technology. In layered coding, video data is divided into multiple layers. Data included in those layers are not overlapping each other. Layers are categorized into the base layer and enhancement layers. Base layer provides the minimum quality of original video data, and it is fundamental for decoding other layers. Enhancement layer provides additional data which improves video quality. Each layer has a dependency with layer directly below for decoding. Several layered coding methods have been proposed, including

MPEG-2 scalable profile, MPEG-4 scalable profile, and H.263+. Multiple layer approach uses layered encoded data to adapt to heterogeneity.

6.4 EVENT CATEGORIZATION

The Proposed categorization model is given in the following figure.

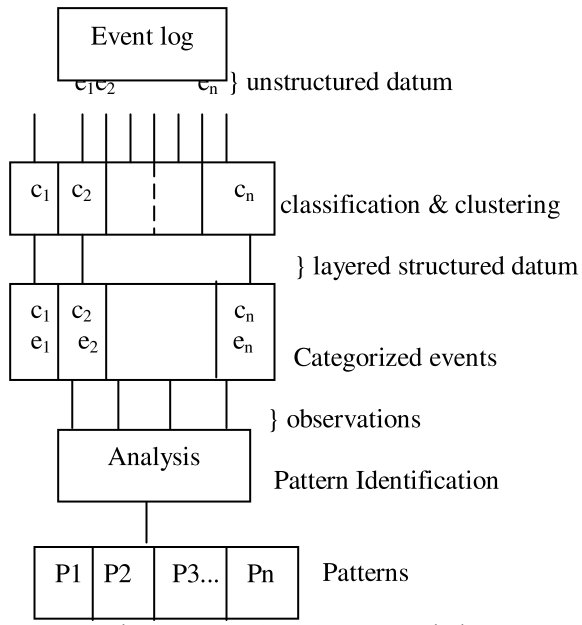


Figure 6.1: Proposed Categorization Model

The various stages in the proposed methodology are

1. Initialize the model
2. Determine the state of the model at each time interval t.
3. Update model structure and parameters.
4. Collect the observations and perform classification and clustering the unstructured datum
5. Evaluate acceptance ratio and accept or reject
6. Construct layered structured datum with different level of patterns

Event Mining System:

$E' = \{O, T, \delta, E, e_0\}$

O -> set of all observations

T -> Time intervals

E -> Event states as a set

e_0 -> Initial event

δ -> Transition function

δ : OXT -> E

where

$O = \{ O_1, O2, \ldots, On\}$

$T = \{ t_1, t2, \ldots , tn\}$

6.5 PATTERN IDENTIFICATION

Many methods for pattern recognition exist. Most of the methods fall into one of the four following categories:

a) Template matching:

Pattern to be recognized is compared with a learned template, allowing changes in scale and pose; this method can work directly on the data. For images a template is a usually a small image. Given a set of templates in the same class, one template representing the class is computed, e.g. by pixelwise averaging. In its simplest form any new pattern is compared pixelwise to the set of stored templates. The new pattern is then assigned to the class for which the correlation between the templates is highest. In practice template matching becomes more difficult as one cannot assume that two templates to be compared are near exact copies of one another. An image might have a different scale;

the object in the image might have a different pose. Therefore, substantial preprocessing is required before template matching can take place. Invariant [31] features can help in this problem.

b) Statistical classification:

Pattern to be recognized is classified based on the distribution of patterns in the space spanned by pattern features;

c) Syntactic or structural matching:

Pattern to be recognized is compared to a small set of learned primitives and grammatical rules for combining primitives. For applications where the patterns have an apparent structure these methods are interesting. They allow the introduction of knowledge on how the patterns in the different classes are built from the basic primitives.

d) Neural networks:

Pattern to be recognized is input to a network which has learned nonlinear input-output relationships; neural networks copy the way humans recognize patterns. They can be viewed as parallel computing systems consisting of an extremely large number of simple processors with many interconnections. In the human brain those simple processors are called neurons, when simulated on a computer one calls them perceptrons. In both cases, the processors have a set of weighted inputs and “fire” if all inputs are above a certain threshold. To train a neural network, input patterns are fed to the system and the expected output is defined. Then the weights for the different connections are automatically learned by the system. In most systems there are also a lot of perceptrons which are neither connected to the input or output, but are part of the so-called hidden layer. Such a neural network is called a multi-layer perceptron. For specific cases neural networks and statistical classifiers coincide. Neural networks are attractive as they can be applied in many different situations. It is, however, difficult to understand why a neural network assigns a certain class to an input pattern.

6.6 EXPERIMENT AND RESULTS

Consider the real time scenario of a shopping mall which contains 10 floors of which each floor can be allotted for different business purposes. The problem here to deal with is the usage of lift available in the mall and giving a report based on the analysis whether to improve the maintenance of the lift or to have additional ones.

Table 6.1: Skeleton table

S.No.	Time	In	Out	Floor No.	Weight
1	Various slots	No. of Persons entered in	No. of persons entered out	Towards floor no.	Ø – no persons $ - over weight

Total number of persons in = Number of Persons entered in

Total number of persons out = Number of Persons entered out

Frequency of floor no. used = Count of floor numbers frequently used

Number of Ø = Number of times the lift was empty

Number of $ = Number of times the lift was overweight

Floor no. at which $ occurs = Frequency of floor numbers were lift was overweight

Lift Data Analysis:

Lift capacity – 800 kg, 6/7 adults and 4 children maximum

The floor details are as given in the figure 5.1 in the page 124.

Lift log file:

Table 6.2: Evaluation table

S. No.	Occasions	Floor Nos.	Usage of lift
1	Govt. holidays/ Religious functions	1 to 6	High
2	Sundays	1 to 6 10	Avg High
3	Marriage /Muhoortham	1 to 5 6	High High
4	Father's day/Mothers Day	1 & 2 3 4 & 5 6	High B-Avg Avg Avg
5	Aadi / Discounts	1 to 5	High
6	Year end stock clearance	1 to 5	High
7	Govt. Rules / Budget Tax rate variational Business change rate New year	7	High
8	Electronics March - June – AC Oct – Dec – TV / Home Theatre	8	High
9	Night time	6 10	Min High
10	Cosmetics Mar- May Oct – Dec	9	High Avg

Max > 90%, High > 70% and <=90%, Avg < =70% and > 50%, B-Avg < = 50% and >= 25%, Min < 25%

From the analysis made, we can suggest to make alterations in the display of products, or make modifications in the entire floor. Also the maintenance of lift should be made frequently during holidays and functions to avoid rush during over crowds. If possible additional lift facility can be provided.

6.7 DISCUSSION

- Based on Data Reports we can advise the mall to change the products of particular floor with additional items.
- If hotel frequency is less then change the food products.
- If frequency of Ø > 0 many times, it represents the requirement of lift operator is a must. Supervision for lift operations is also required.
- If $ > 0 on both lift exceeds 15 times per month, it impacts the requirement of another lift for the shopping mall.

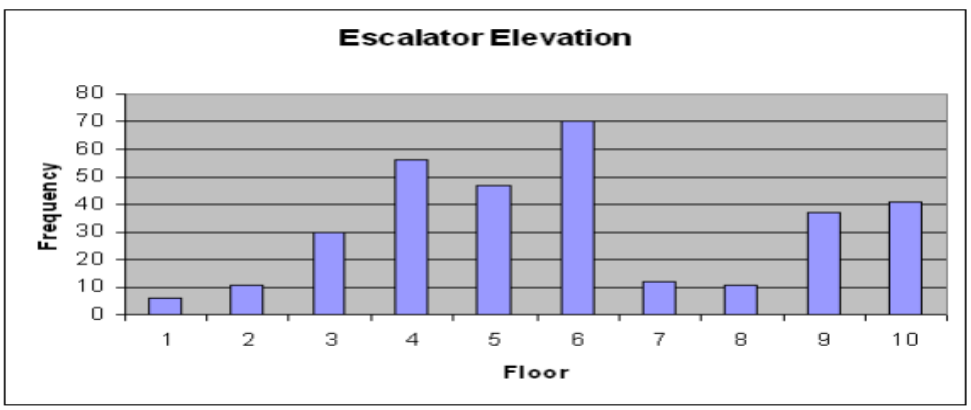


Figure 6.2: Sample chart comparison on a Akshya tritiyi (Function for mostly gold Purchase)

CHAPTER 7

UNIVARIATE TRANSFORMATION FOR IMAGE MINING

In this chapter we deal with the techniques for efficient image retrieval system♦. Nowadays Image retrieval plays a vital role in the investigation department, Passport Verification, Voters Id Checking, Document Analysis together with Biometric markings etc. The main requirement in this system is the amount of time consumption for the retrieval results.

7.1 IMAGE MINING - OVERVIEW

Image mining is a process to find valid, useful, and understandable knowledge from large image sets or image databases. Image mining combines the areas of content-based image retrieval, image understanding, data mining and databases. It deals with the extraction of knowledge, image data relationship, or other patterns not explicitly stored in the images. It uses methods from computer vision, image processing [18], image retrieval, data mining, machine learning, database, and artificial intelligence. Rule mining has been applied to large image databases. Image mining is more than just an extension of data mining to image domain.

♦ This covers our work published "Efficient Retrieval Techniques for Images Using Enhanced Univariate Transformation Approach", International Journal of Engineering Science and Technology- Vol. 2(8), 2010, 3762-3768.

Feature selection and extraction is the pre-processing step of Image Mining. This is a critical step in Image Mining. The approach here is to mine from Images – to extract patterns and derive knowledge from large collections of images, deals mainly with identification and extraction of unique features for a particular domain. Even if there are various features available, the aim is to identify the best features and thereby extract relevant information from the images.

The four major image mining steps are as follows:

1. Feature extraction - Segment images into regions identifiable by region descriptors (blobs). Ideally one blob represents one object [32]. This step is also called segmentation.
2. Object identification and record creation - Compare objects in one image to objects in every other image. Label each object with an id. This step is the preprocessing algorithm.
3. Create auxiliary images - Generate images with identified objects to interpret the association rules obtained from the following step.
4. Apply data mining algorithm to produce object association rules.

7.2 IMAGE SEGMENTATION

The basic idea of image segmentation is to group individual pixels (dots in the image) together into regions if they are similar. Similar can mean they are the same intensity (shade of gray), form a texture, line up in a row, create a shape, etc. There are many techniques available for image segmentation, and they vary in complexity, power, and area of application.

Segmentation Algorithm and Segmentation Parameters

The gray level histogram is calculated from the original image. This histogram is smoothed by some numerical functions and heuristic rules to find the cut points for the liquor and brain gray-level areas. The parameters of the function and rules are stored with the cases, and given to the segmentation unit if the associated case is selected. The following steps are performed. The histogram is smoothed by a numerical function. There are two parameters to select: the complexity of the interpolation function and the interpolation width. Then the histogram is segmented into intervals, such that each begins with a valley, contains a peak and ends with a valley. The peak-to-shoulder ratio of each interval is tested first. An interval is merged with the neighbor sharing the higher of its two shoulders if the ratio of peak height to the height of its higher shoulder is greater than or equal to some threshold. Finally, the number of the remaining intervals is compared to a predefined number of intervals. If more than this has survived, the intervals with the highest peaks are selected. The number of intervals depends on the number of classes into which the image should be segmented. The thresholds are calculated and then applied to the image.

Histogram-Based Segmentation

Histogram-based image segmentation [39] is one of the most simple and most often used segmentation techniques. It uses the histogram to select the gray levels for grouping pixels into regions. In a simple image there are two entities: the background and the object. The background is generally one gray level and occupies most of the image. Therefore, its gray level is a large peak in the histogram. The object [32] or subject of the image is another gray level, and its gray level is another, smaller peak in the histogram.

Histogram Preprocessing

Histogram-based segmentation depends on the histogram of the image. Therefore, the image and its histogram may need preprocessing before analyzing them. The first step is histogram equalization Histogram equalization attempts to alter an image so its histogram is at and spreads out over the entire range of gray levels. The result is an image with better contrast. The next preprocessing step is histogram smoothing. When examining a histogram, look at the peaks and valleys. Too many tall, thin peaks and deep valleys will cause problems. Smoothing the histogram removes these spikes while retaining the same basic shape of the histogram. Smoothing a histogram is an easy operation. It replaces each point with the average of it and its two neighbors.

Thresholding and Region Growing

Two more topics common to all the methods of image segmentation are: image thresholding and region growing. Image thresholding sets the pixels in the image to one or zero. It begins with the routine threshold image array that accomplishes this task. The difficult task is region growing.

Histogram-Based Techniques

The following present four segmentation techniques: manual technique, histogram peak technique, histogram valley technique, and adaptive technique.

a) Manual Technique

In the manual technique the user inspects an image and its histogram manually. Trial and error comes into play and the result is as good as you want it to be. Note that all image segmentations will appear rough. It is possible to perform additional processing to make the result more pleasing to the eye but that is not the purpose of segmentation. The purpose is to break the image into pieces so later computer processing can interpret their meaning. The output is for computer not human consumption. Also note how difficult it is for the computer, even with manual aid, to find objects [32] that are trivial for humans to see. Anyone could trace over the input image and outline the objects better than the segmentation process. Manual segmentation is good for fine tuning and understanding the operation. Its trial-and-error nature, however, makes it time consuming and impractical for many applications. We need techniques that examine the histogram and select threshold values automatically.

b) Histogram Peak Technique

The first technique to examine the histogram and select threshold values automatically uses the peaks of the histogram. This technique finds the two peaks in the histogram corresponding to the background and object of the image. It sets the threshold halfway between the two peaks.

c) Histogram Valley Technique

The second automatic technique uses the peaks of the histogram, but concentrates on the valleys between them. Instead of setting the midpoint arbitrarily halfway between the two peaks, the valley technique searches between the two peaks to find the lowest valley.

d) Adaptive Histogram Technique

The final technique uses the peaks of the histogram in a first pass and adapts itself to the objects [32] found in the image in a second pass. In the first pass, the adaptive technique calculates the histogram for the entire image. It smoothes the histogram and uses the peak technique to find the high and low threshold values. In the second pass, the technique segments using the high and low values found during the first pass. Then, it calculates the mean value for all the pixels segmented into background and object. It uses these means as new peaks and calculates new high and low threshold values. Now, it segments that area | again using the new values.

For images we can also consider the different ways of segmenting an image. A partition decomposes the image into fixed regions. Commonly this is either a fixed set of rectangles, or one fixed rectangle in the middle of the image, and a further partition of the remaining space in a fixed number of equal parts. Weak segmentation boils down to grouping pixels in the image based on a homogeneity criterion on colour or texture, or by connecting edges. It leads to a decomposition of the image where each region in the decomposition has a uniform colour or texture. For strong segmentation, finding specific conceptual objects [32] in the image, we again have to rely on models for each specific object, or a large set of hand-annotated examples.

Main objective of image mining is to reduce uncertainty, thus to allow making better decisions. Because of the high temporal resolution, images often are not properly validated. We may therefore doubt the quality of the derived information, requiring additional efforts for skilful mathematical and logical modelling. Fuzzy methods, statistical methods and probabilistic procedures are important in order to deal with the various characteristics, and with a proper attention to aspects of data quality.

The object has to be segmented from the original image, such that all pixels representing the objects shape have been identified as distinct from those pertaining to the rest of the image. A local diffusive segmentation method is used. All contiguous pixels, which share a given point-based characteristic of the object or are surrounded by those that do, are considered as object pixels and those outside the included region, are considered as background. The result is a group of contiguous pixels, which collectively represent the object. The boundary pixels of the object are then extracted from the segmented object pixels by a simple iterative trace, around the outside of the object that continues until the starting point is reached. This trace produces a second group of pixels collectively representing the objects exterior contour.

7.3 IMAGE ENHANCEMENT

An image processing [18] system that involves digital signal processing is shown in Figure 7.1.

Figure 7.1: Image processing system

The input image source *I(x,* y) is generally an object or a natural scene, but it may be an image produced by another system, such as a filter, a cathode ray tube (CRT) display monitor, or a video cassette recorder (VCR). The digitizer converts the input source to an electrical signal whose amplitude represents the image intensity and digitizes the electrical signal using an analog-to-digital (AID) converter.

The sequence $f(n_1, n2)$ that results from the digitizer is then processed by a digital image processing [18] algorithm. The algorithm may be implemented on a general purpose computer, a microprocessor, or special purpose hardware. The specific algorithm used depends on the objective, which may involve image enhancement, restoration, coding, understanding, or any combination of them. The result of processing is then displayed, generally for human viewing, but sometimes as an input to another system. The display typically used is a CRT monitor, but may be a photograph or VCR tape. If the result is some symbolic representation, as in image understanding, the display used can also be just a printer.

Image enhancement is the processing of images to improve their appearance to human viewers or to enhance other image processing [18] systems' performance. Methods and objectives vary with the application. When images are enhanced for human viewers, as in television, the objective may be to improve perceptual aspects: image quality, intelligibility, or visual appearance. In other applications, such as object identification by machine, an image may be preprocessed to aid machine performance. Because the objective of image enhancement is dependent on the application context, and the criteria for enhancement are often subjective or too complex to be easily converted to useful objective measures, image enhancement algorithms tend to be simple, qualitative, and ad hoc. In addition, in any given application, an image enhancement algorithm that performs well for one class of images may not perform as well for other classes. In image enhancement, the objective is to make the processed image better in some sense than the unprocessed image.

In another class of enhancement problems, a degraded image may be enhanced by reducing the degradation. Examples of image degradation are blurring, random background noise, speckle noise, and quantization noise. This area of image enhancement overlaps with image restoration. An algorithm that is simple and ad hoc, and does not attempt to exploit the characteristics of the signal and degradation, is generally considered an enhancement algorithm. An algorithm that is more mathematical and complex, which exploits the characteristics of the signal and degradation with an explicit error criterion that attempts to compare the processed image with the original undegraded image, is generally regarded as a restoration algorithm. This distinction is admittedly somewhat vague and arbitrary. Some arbitrary decisions have been necessary in dividing certain topics between this chapter and the next chapter, which deals with the image restoration problem. It is well known that the contours or edges in an object contain very important information that may be used in image understanding applications. The first step in such an application may be to preprocess an image into an edge map that consists of only edges. Since more accurate detection of edges in an image can enhance the performance of an image understanding system that exploits such information, converting an image to its corresponding edge map may be viewed as an enhancement process.

Another important class of image enhancement problems is the display of 2-D data that may or may not represent the intensities of an actual image. A low resolution image of 128 x 128 pixels may be made more visually pleasant to a human observer by interpolating it to generate a larger image, say 256 x 256 pixels. In 2-D spectral estimation, the spectral estimates have traditionally been displayed as contour plots. Although such 2-D data are not images in the conventional sense, they can be presented as images. We can display them as black-and-white images, or we can enhance them with colour so that their appearance may be improved and information conveyed more clearly. In other applications, such as infrared radar imaging, range information as well as image intensities may be available. By displaying the range information with colour, relative distances of objects in an image can be highlighted. Even good-quality images may be enhanced by certain types of distortion. For example, when an object in an image is displayed with false colour, the object may stand out more clearly to a human viewer.

Image enhancement technique is to make the image clearer so that various operations can be performed easily on the image. At first the captured RGB image is converted to the gray level image. Contrast stretching is a simple image enhancement technique that attempts to improve the contrast in an image by stretching the range of intensity values it contains to span a desired range of values, e.g. the full range of pixel values that the image type concerned allows. Low contrast images can be found due to the poor illumination, lack of dynamic range in the imaging sensor, or due to the wrong setting of the lens. The idea behind the contrast stretching is to increase the dynamic range of intensity level in the processed image. The general process of the contrast stretching operation on grayscale image is to apply the following equation on each of the pixels in the input image to form the corresponding output image pixel:

$$O(x,y) = (I(x,y)-min)(n_i/(max-min))+i \qquad (1)$$

where, O(x,y) represents the output image, I(x,y) represents the x^{th} pixel in the y_{th} column in the input image. Here, ni represents the number of intensity levels, i represents the initial intensity level, "min" and "max" represent the minimum intensity value and the maximum intensity value in the current image respectively. Here "no. of intensity levels" shows the total number of intensity values that can be assigned to a pixel. For example, normally in the gray-level images, the lowest possible intensity is 0, and the highest intensity value is 255. Thus "no. of intensity levels" is equal to 256. The contrast stretching operation is applied on the grayscale images in two passes. In the first pass the algorithm calculates the minimum and the maximum intensity values in the image, and in the second pass through the image, the above formula is applied on the pixels. Enhance the gray level image to improve its visual quality and machine recognition accuracy using the following formula:

$$G^* = \text{INTRANS}\ (F', \text{stretch}', M, E) \qquad (2)$$

Here, INTRANS performs the intensity or gray level transformations and G computes a contrast stretching transformation using the following MATLAB expression:

$$\text{Contrast} = 1./(1+ (M./(F + eps))\text{^}\ E) \qquad (3)$$

where, parameter M must be in range [0,1]. The default value for M is mean2(im2double(F)) and the default value for E is 4. Here, F is gray-level image and M is such result which is found by applying image double and median filtering operation on F. eps returns the distance from 1.0 to the next largest double-precision number, i.e. eps = 2^(− 52).

7.4 REPRESENTATION TECHNIQUES

A digital [8] image is an array of pixels, where each pixel has a colour. The basic representation for the colour of the pixel is the triple R(ed), G(reen), B(lue). There are however many other colour spaces which are more appropriate in certain tasks. Consider HSV and Lab here.

A first thing to realize is that the colour of an object is actually a colour spectrum, indicating how much a certain wavelength is present. This is the basis for defining HSV. To be precise the three components of HSV are as follows: H(ue) is the dominant wavelength in the colour spectrum. It is what you typically mean when you say the object is red, yellow, blue, green, purple etc. S(aturation) is a measure for the amount of white in the spectrum. It defines the purity of a colour distinguishing for example signal-red from pink. Finally, the V(olume) is a measure for the brightness or intensity of the colour. This makes the difference between a dark and a light colour if they have the same H and S values.

Lab is another colour space that is used often. The L is similar to the V in HSV. The a and b are similar to H and V. The important difference is that in the Lab space the distance between colours in the colour space is approximately equal to the perceived difference in the colours.

The colour is assigned to individual pixels. All these colours will generate patterns in the image, which can be small, or large. These patterns are denoted with the general term texture. Texture is of great importance in classifying different materials like the line-like pattern in a brick wall, or the dot-like pattern of sand. In an image there will be in general different colours and/or textures. This means there will be many positions in the image where there is a significant change in image data, in particular a change in colour or texture. These changes form (partial) lines called edges. The above measure gives a local description of the data in the image. In many cases global measures, summarizing the information in the image are used. Most commonly these descriptions are in the form of colour histograms counting how many pixels have a certain colour. It can however, also, be a histogram on the directions of the different edge pixels in the image.

An image histogram looses all information on spatial configurations of the pixels. If I have a peak in the histogram at the colour red, the pixels can be scattered all around the image, or it can be one big red region. Colour coherence vectors are an alternative representation which considers how many pixels in the neighbourhood have the same colour. A similar representation can be used for edge direction.

The Discrete Cosine Transform (DCT) is a transform which takes an image and computes it frequency domain description. This is exactly the same as considered for audio earlier, but now in two dimensions. Coefficients of the low frequency components give a measure of the amount of large scale structure where high frequency information gives information on local detailed information. The (Haar) wavelet transform is a similar representation, which also takes into account where in the image the specific structure is found.

Images are the most active area of research among the many forms of multimedia data types. Image data can be easily created or obtained, is voluminous, and has many possible features that can be extracted. Image analysis presents many challenging and interesting research areas, and image analysis techniques are also used for video analysis [29].

Images are commonly stored in one of two format types:

1. Pixel formats. The image is composed of many picture elements within a grid. An image in pixel format must be rectangular. Each pixel has a colour that can be represented using different schemes at varying levels of detail. Pixel formats may or may not be compressed. Common pixel formats include JPEG (JPG), GIF, PNG, TIFF, or BMP. Digital cameras take pictures in pixel format.
2. Vector formats. The image is composed of individual, scalable objects defined by mathematical equations. An image in vector format need not be rectangular. Common vector formats include SVG, WMF, or CGM. Shockwave Flash and Google Maps display images using vector format.

Images have many interesting features that can be measured and used for analysis, including:

- Colour
- Texture
- Motion
- Edges
- Shapes

The Multimedia Content Description Interface (MCDI), also known as MPEG-7, is a standard that provides mechanisms to describe and search a variety of multimedia, including both low-level features and high-level content. This broad standard is based on XML and contains the following major components:

1. Description Tools — Descriptors (D) define the syntax and semantics of each feature (metadata element); Description Schemes (DS) specify the structure and semantics of the relationships between their components.
2. A Description Definition Language (DDL) defines the syntax of the MPEG-7 description tools to allow the creation of new DS (and possibly D) and to allow the extension and modification of existing DS.
3. System Tools support binary coded representation for efficient storage and transmission, transmission mechanisms (both for textual and binary formats), multiplexing of descriptions, synchronization of descriptions with content, and management and protection of intellectual property in MPEG-7 descriptions.

These components relate to each other as follows: the *Description Tools* are used to facilitate the creation of *Descriptions*, which are sets of instantiated *Descriptors* (and their corresponding *Description Schemes*). The *Description Tools* may also be used to create new *Descriptions Schemas* and *Descriptions*, if needed. Finally, the *System Tools* are used to deploy the *Descriptions* that have been created. The actual creation and consumption of the descriptions is outside the scope of MPEG- 7, which is concerned only with the representation of the descriptions themselves.

Visual words and Image Representation

Treat images as a collection of the representative prototypes sampled from the training image corpus, and then use the resulted distribution in the descriptor space as a characterization of the image. The words in the images, called visual words [41], must be calculated to form a vocabulary of N words. Each image will be finally represented by a word histogram. The construction of visual words is processed in two steps:

- Computation of local descriptors [30] for a set of images.
- Clustering of the previous descriptors, by K-means.

Each cluster will define a visual word. There would be as many words as clusters obtained at the end of step (ii). The local descriptor computation in an image is also done in two stages

1. First detect the interest points in the image. These points are maximums of Laplace of Gaussian.
2. Then, the descriptor of the interest points is computed on the gray level gradient of the region around the point. SIFT algorithm is employed to extract the features due to its impressive performance in image recognition. The SIFT algorithm was invented for object [32] recognition and widely adopted for applications in image/video retrieval. It consists of four major stages: scalespace extrema

detection, keypoint localization, orientation assignment, and keypoint descriptor construction. First, the DoG (Difference of Gaussian) operator is convolved with the image in the scale space, and a pixel is selected as a keypoint if it is the scale-space extrema. Then, an edge orientation histogram (EOH), determined by the gradient orientations in each keypoint's neighborhood, is constructed for the keypoint. The resulted 128-dimensional vector of EOH is then employed as a distinctive local descriptor. This step is to form visual words from the local descriptors [30] computed in the first step. Perform a k-means on descriptors and take the averages of each cluster as visual word.

After building the visual vocabulary, all descriptors are assigned to their nearest cluster. Compute distances, in R, from each descriptor to representatives of previously defined clusters. An image is characterized by the frequency of its descriptors. The image corpus will be represented by a contingency table crossing images and clusters and a cell (i,j) contains the number of descriptors of the image i assigned to the cluster j.

Decision Trees and Rules

Decision trees and rules that use univariate splits have a simple representational form, making the inferred model relatively easy for the user to comprehend. However, the restriction to a particular tree or rule representation can significantly restrict the functional form of the model.

A large number of decision tree and rule-induction algorithms are described in the machine learning and applied statistics literature. To a large extent, they depend on likelihood-based model-evaluation [11] methods, with varying degrees of sophistication in terms of penalizing model complexity. Greedy search methods, which involve growing and pruning rule and tree structures, are typically used to explore the super exponential space of possible models. Trees and rules are primarily used for predictive modelling, both for classification and regression, although they can also be applied to summary descriptive modelling.

7.5 COMPONENT ANALYSIS

Component Recognition

Component recognition requires making the step from raster to objects. It is done by applying a segmentation in which objects are modelled. Image segmentation can be performed by various procedures based on mathematical morphology, edge detection. Interesting results can be obtained by applying texture based segmentation. The result of this operation for an image at moment t thus is a series of n_t objects Ot,i, i =1 …nt, that are characterized by similar pixel values, which are different from pixel values.

Component Membership

By applying fuzzy [33] approach, membership functions are obtained for identified objects $O_{t,i}$. Membership functions in a 2-

dimensional space are functions $\mu A(x,y)$, taking values between 0 and 1, which specify the degree to which the location (x,y) is characterized by A. A membership function is defined by a function $\eta(z)$ for $z \in [0,1]$, which transformed using a homeomorphism towards the membership function $\mu A(x, y)$ for $(x, y) \in A$, the set of interest. Such membership functions are characterized by the steepness of their slopes, by their homogeneity and centroid.

The centroid, has coordinates ($x_{c,t}$, yc,t) at each moment t, and when it is present on n images is observed at the moments t1,..., tn. At the different moments these centroids can then be plotted and a curve can be fitted.

The univariate operators for a single vague set from the bivariate operators for two vague sets are discussed here. It has been shown that on the basis of α-cuts an extension of crisp operators could be defined and implemented in a usable way. The following univariate operators are distinguished.

1. The length of α-cuts defines the length of a vague line object
2. $|VL|$ is inversely proportional to value of α.
3. $|VL|=f(\alpha)$
4. $\int f(\alpha)d\alpha=n$ where n is a real number.
5. $\int\int f(\alpha)d\alpha=\sum$Area of α cuts

   $\int\int f(\alpha)d\alpha = 1$ equals the area of the core,

   $\int\int f(\alpha)d\alpha = 0$ equals the area of the support.

$\int\int f(\alpha)d\alpha$ inversely proportional to α.

6. The line connecting any two extremes is equal to the extreme connection of α-cuts.
7. The Circumference of vague set is distinct with α-cuts.

One binary variable operator is defined, being the distance between two objects. This distance is also defined in terms of the distance between α-cuts, resulting in a distance curve as a function of α. The distance decreases with increasing values of α, and integration of the distance curves provides a single value for the distance. The distance is not defined for values of α larger than the lowest of the maximum membership values of the two objects.

Behaviour Analysis

Treatment of objects that was straightforward for crisp objects becomes complicated for uncertain object: Splitting and merging of objects, their birth and death require some special attention. Consider the object $O_{k,1}$ at moment tk. The splitting of this object leads to two new objects Ok+1,1 and Ok+1,2 at moment tk+1. Both objects require membership functions, which are defined on the basis of the membership function at t as well as the characteristics of the new objects. An inverse operation is merging of two objects at moment tk towards one single object at moment tk+1. This requires the membership functions to be combined into one new membership function. Also the two existing centroids have to be combined into one new centroid, possibly by a weighted averaging of the centroids of the original objects.

Pattern Mapping

The next stage of image mining is pattern mapping of the object in the space-time domain. When analyzing single uncertain object over time, one way to proceed is to define a parametric curve for the centroid and possibly of other parameters of the membership function and then to predict at which location the curve most likely will be at a moment t_0 beyond the moments of observation so far. In pattern mapping, one may consider a future event, i.e. a real mapping, or one moment prior to image availability, like mapping the moment that the object is born. Also, mapping is sometimes required for the object between two moments tk and tk+1. Pattern Mapping results in the values of the membership function, of the centroid.

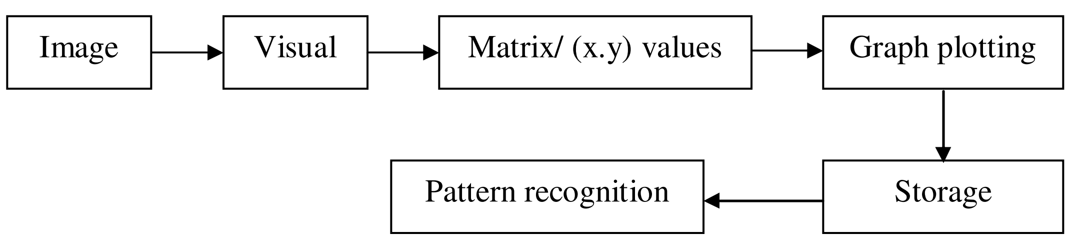


Figure 7.2 Basic Structure

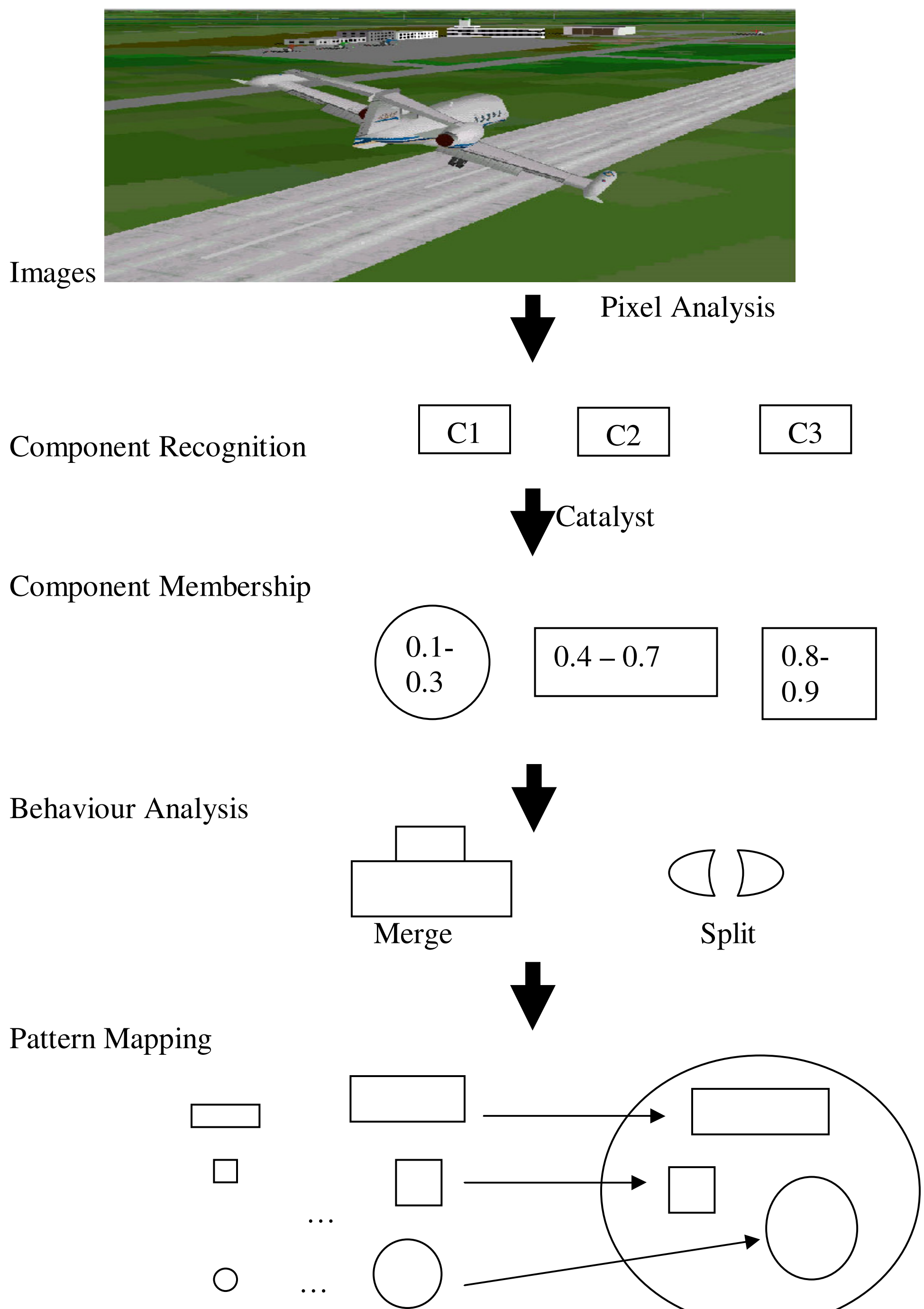


Figure 7.3 Process of Image Mining

7.6 EXPERIMENT AND RESULTS

Here the database contains 25 images divided into 5 categories. The image in query has to mapped with the images available in the database. This mapping technique is applied by the following processes – component recognition, component membership, behaviour analysis and pattern mapping.

Table 7.1: Database Description

Category	No. of Images
Flowers	5
Birds	5
Cars	5
Animals	5
Trees	5

Visual words [41] and images are displayed on the same plane. To help the user for interpretation, the screen is divided into two parts: the projection of images and/or visual words on the left, and the right part is reserved to display some selected images. The user can select one or a group of images by pointing on it. All the images found in a neighborhood of radius *r* of the interesting image will be displayed on the right of the screen. The selected images are listed on the right hand side. This gives us immediately a general summary of the content of these images.

For our example, consider we are having Flowers, Birds, Cars, Animals, and Trees as group of images in the database. These group names are the visual words and membership value is assigned to each group. Once an image is given in the query, based on the membership value, it is mapped with images in the database of each group. Once the group has been identified, then with the help of behaviour analysis, the exact image that matched with the query image will be selected from the group if it is available. Then the result is displayed with the image from database along with the query image.

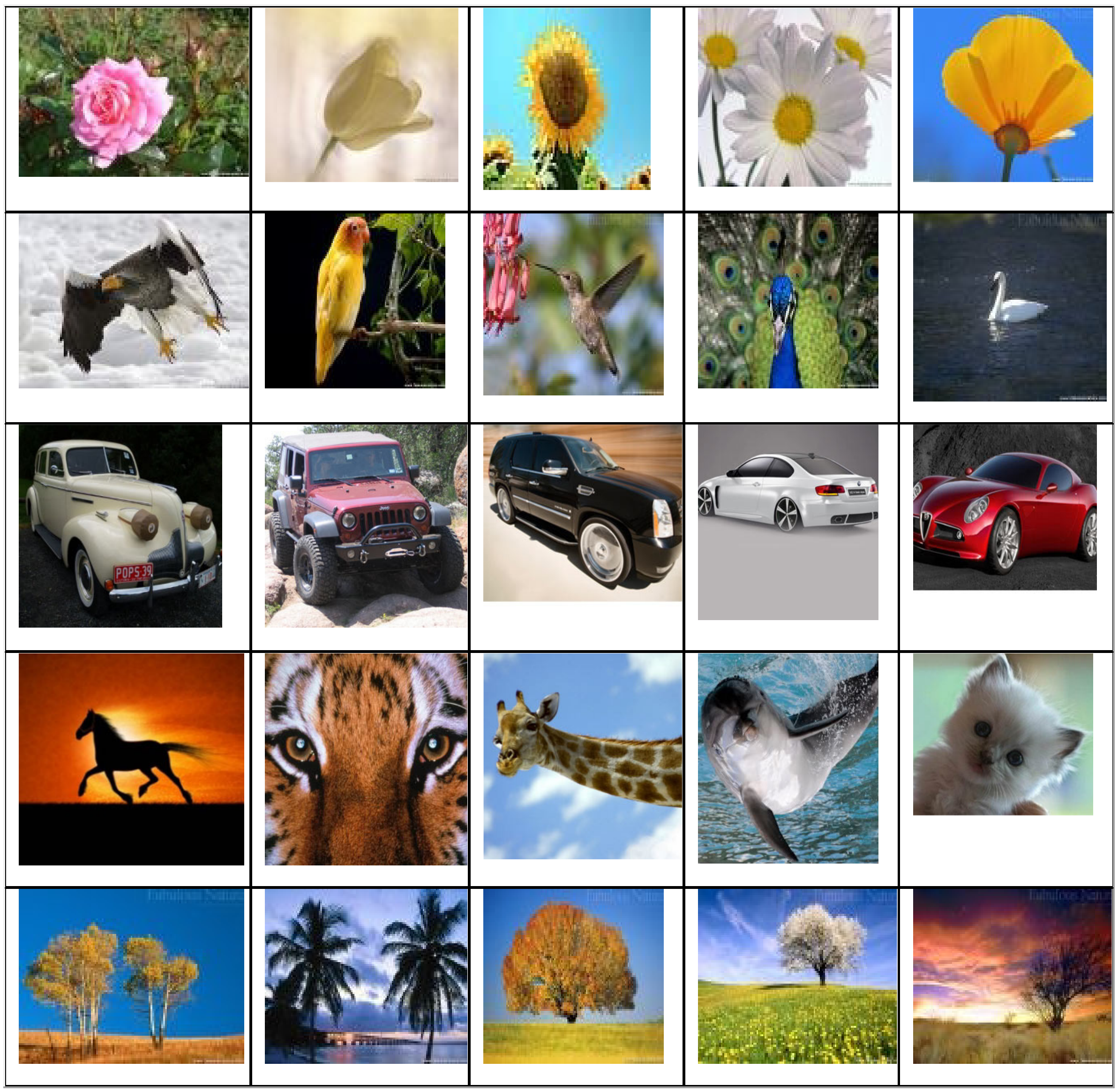


Figure 7.4: Images in Database

After the mapping technique, if the query image matches with the image available in the database, then the result is displayed as below:

Templates	Result
	
	

7.7 DISCUSSION

This concept deals with the resultant model which uses Image mining techniques for identifying image retrieval issues. It can be achieved by image preprocessing, image cleaning, image segmentation, image enhancement and image representation for image retrieval which result in analysis of multiple image data resources. In future these techniques may be very helpful in the analysis of Satellite, Medical and Electronic Circuit images with improved computational time and classification techniques. The content based image annotation and image referential approaches can be done through the performance of implicit semantic tags in the search result images, the efficiency of search accuracy is more prominent for such applications.

CHAPTER 8

CONCLUSION AND FUTURE ENHANCEMENTS

8.1 CONCLUSION

Event mining of video and image are dealt in this research work with its unstructured nature. The observations in the implementation of this unstructured datum on event mining are addressed as follows:

In this work the resultant model uses data mining techniques for identifying the abnormal event detection issues. The final model is fully self explanatory without any basic domain knowledge. It can be achieved by data preprocessing, data cleaning, and data integration for event detection which result in analysis of multiple video data resources. Nowadays Image retrieval plays a vital role in the investigation department, Passport Verification, Voters Id Checking, Document Analysis together with Biometric markings etc. The main requirement in this system is the amount of time consumption for the retrieval results. In this work the heuristical approach by partitioning the unstructured datum into structured patterns is used. Any system can be analyzed through its functional event log file using event mining techniques. These events can be further processed for efficient productivity systems. It uses the approach for handling unstructured event datum with its layered statistics. The conversion of unstructured event datum to the structured event datum plays the vital role for implementing the scales and measurements for unstructured event sets, moreover it is used to perform the analysis and establish pivotal statistics to skip from fuzziness to the crispy culture. The problems that arise on computational framework can be handled through structural analysis over similar data classifications.

8.2 CONTRIBUTIONS

In this research the following contributions have been made in the area of data mining in unstructured datum.

- In Chapter 4 Frame classification theory for video mining is performed.
- In Chapter 5 and 6 Unstructured datum is identified, classified, clustered and analysis is made and output is produced based on the problem.
- In Chapter 7 Image parameter similarity is used to perform image mining.

8.3 FUTURE WORK

The development and evaluation of various methods discussed in this research work serves as a solid foundation for continuing research in the area of data mining. The future research can be oriented in many directions as given below:

- ❖ This work can be extended to evaluate the patterns in audio data sets and text.

- ❖ Implementation of neural network and fuzzy logic over unstructured resources which requires complex analysis schema will be focused.
- ❖ Relevant feedback methods can also be implemented for satisfying user needs.

EVENT MINING IN MULTIMEDIA DATA WITH UNSTRUCTURED RESOURCES

V.NARAYANI (Reg.no.3666)
Research Scholar
Dept. of Computer Science
St.Xavier's College
Palayamkottai

DR.S.P.VICTOR
Associate Professor & Head
Dept. of Computer Science
St.Xavier's College
Palayamkottai

ABSTRACT

Data Mining comprises three fundamental tasks namely Data Identification, Data Classification, and Data Recognition. Given a datum for mining a system, initially identify its nature as structured or unstructured, then classify it with its event analyzer and finally recognize through its behavioral dynamic measures.

The purpose of video data mining is to discover and describe interesting patterns in data. The task becomes especially tricky when the data consist of video sequences, because of the need to analyze enormous volumes of multidimensional data. The supervised and unsupervised clustering instances with feature extraction will be used to support the final event detection. The significant and optimal expectation of the proposed framework can be explained through a collection of videos containing several frames.

Event Mining discovers and delivers information and knowledge in a real-time stream of data, or events. Event detection is a natural application for multimedia mining. It has a variety of practical

applications that are becoming more widespread as data-collection instruments become more sophisticated and less expensive. Events are essential factors in the world of data mining. Hybrid dynamic systems, stochastic systems and discrete event system are some of the entities that deal with unstructured datum in event mining systems. In this work there exists an analysis of discrete event system which produces database of unstructured datum during its functional approach. It also includes the concept of handling the unstructured datum towards the layered structured system with its various levels of impacts and implications.

Unstructured data refers to information that either does not have a pre-defined data model and/or does not fit well into relational tables. Unstructured text information is naturally text-heavy but may contain data such as dates, numbers, and facts as well. These results in irregularities and ambiguities that make it difficult to understand using traditional computer programs as compared to data stored in fielded form in databases or annotated in documents. The semantics about events is essential in real time decision analysis and support. In earlier stages different approaches have been developed to build models and logical algorithms that relied on the concept of events. Nowadays several modeling techniques are available in processing events with its mining scenarios, but the process of handling unstructured datum in Event mining is a little bit tedious approach. In this work, handling unstructured datum through the conversion of layered structured datum with its various levels of implications is addressed with a sample datum.

Image mining deals with the extraction of knowledge, image data relationship, or other patterns not explicitly stored in the images. It is a process to find valid, useful, and understandable knowledge from large image sets or image databases. It uses methods from computer vision, image processing, image retrieval, data mining, machine learning and artificial intelligence. Rule mining has been applied to large image databases. The main approach is to mine from large collections of images. Image mining combines the areas of content-based image retrieval, image understanding, data mining and databases. Image mining is more than just an extension of data mining to image domain.