A survey of technologies and algorithms for parsing and indexing multimedia databases.

Augustine Kureva Damba

This dissertation is presented as part of the requirements for the award of the Degree of Master of Computer and Information Science at the Auckland University of Technology

April 2008

Acknowledgements

I owe the success of this work to my wife Lorraine and daughter Laura. Thanks Lorraine for being so supportive and thanks Laura for being such a nice baby.

I would like to thank Dr. Russel Pears, my supervisor, for his guidance, advice and constant encouragement throughout the course of this research. I have benefited a lot from his rich database background.

Table of Contents

Attestation of Authorship

"I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the qualification of any other degree or diploma of a university or other institution of higher learning, except where due acknowledgement is made in the acknowledgements"

Yours Sincerely,

Augustine Damba

Abstract

A survey of technologies and algorithms for parsing and indexing multimedia databases.

Conventional database systems are designed for managing textual and numerical data, and retrieving such data is often based on simple comparisons of text/numerical values. The digitized representation of images, video, or data by itself does not fully represent the contents of the multimedia items, users have to manually add semantic data to fully describe content. The content-based retrieval methods take these intrinsic values into account. Content-based retrieval of multimedia database calls for content-based indexing techniques.

Querying multimedia data is costly and appropriate indexing is required. Efficient indexing of high dimensional feature vectors is important to allow content based query applications to perform efficiently on large databases. In this survey, I give an overview of the advances on algorithms for indexing multimedia data, identify the problems of processing queries in high-dimensional space, and recommend the best multimedia data features to be used for indexing. The main contribution of this research is to identify and recommend the commonly used methods of indexing multimedia data by collating available algorithms.

Chapter 1: Introduction

A multimedia database is a collection of “multimedia data items such as text, images, graphic objects, sketches, video and audio” (Adjeroh & Nwosu, 1997). The database is very different from the normal textual database in terms of both semantics and presentation. Multimedia data is so complex that it requires complex processing to drive the semantics from its contents. Multimedia data requires huge capacity storage management systems in addition to the traditional relational database systems (RDBMS) capabilities (Adjeroh & Nwosu, 1997). As the storage of the video data grows, so does the complexity of retrieval of the data.

Existing research has focused on implementing multimedia algorithms using data features such as color,shape and texture (Adjeroh & Nwosu, 1997). These common features are automatically extracted from the image data by algorithms. The extracted features are then compared to the features of supplied image for searching and indexing purposes. There is still a need for further research to identify state of the art indexing and algorithm methods that support content-based query applications.

While there have been a large number of algorithms developed and implemented, there are gaps in research regarding the effect of data features used in the algorithms on perfomance. This research aims to identify the most significant methods for indexing of high dimensional feature vectors which allow content based query applications to perform efficiently on large databases. Ideally, an algorithm should only return stored images that match a user defined criterion and no others, which I term redundant. I thus define a high performing algorithm as one that returns the least number of redundant images, saving user’s time and increasing efficiency.

While constructive research was used in the development of multimedia indexes, quantitative research syntheses will form the central part of this research review (Mosteller & Colditz, 1996).

Overview of multimedia data

Textual data is easy to represent in the database when compared to video data. Video data has features such as the video motion that are difficult to represent as simple objects. In the database, video data is stored as contiguous frames. A frame is a representation of a single image including its features. A single video will consist of many frames that represent every single image in the data. The frames can be meaningless unless their relationships to each other are maintained in the database. The video data has spatial constraints that define a relationship between video frames.

There is increased use of surveillance cameras in cities and public places such as train stations, shopping malls and at airports. Interaction with video data involves long duration operations (Adjeroh & Nwosu, 1997) for example if there is a crime, the police spend days going through the video footage. The multimedia data in the database is not very useful unless it can be retrieved from the database.

The storage of the images and video data is not a problem. However, for the images and videos to be useful, there is need for efficient methods of querying and displaying the data without losing the original quality of the images.

Content-based searching is the process of searching by the actual contents of the multimedia data (Smoliar et al, 1994). There are challenging issues to be considered when querying by video content. There are limitations to textual descriptions of the video data, which makes it very difficult to make precise requests when retrieving content (Adjeroh & Nwosu, 1997). It is difficult to query the database by scanning an image. If a person’s image is supplied, all videos with a person in it are returned. How does the query know which person to look for? In this scenario, adding image features such as the colour of the person's eyes, hair colour and other distinctive personal characteristics will surely help to narrow down the search.

The name given to a video is not a sufficient representation of the video contents. The existing methods of cataloguing a video by adding meta-data also have their limitations (Smoliar et al, 1994). The manually captured data may have errors and also most importantly the meaning of the video data to the person entering the metadata may be different from someone else's perceived meaning of the video. For example, an X-ray image taken by a dentist consists of the images as well as the text added by the dentist. The text in this context is enough to describe the tooth surgery. However, the text may not be very useful if another dentist is only looking for the contents of the image. Users want to be able to retrieve videos based on their content.

The searching mechanism should allow for quick retrieval of images. To enable fast searching, proper indexes should be in place. There is need for investigation on what indexing strategies are currently in place for supporting content-based searching.

Research objective and expected outcome

The objective of this research is to identify multimedia indexing algorithms performance bottlenecks. A high performing algorithm is one that returns semantically closest matching images in similarity matching. The expected outcome would be the identification of multimedia data features that can be used in algorithms to support efficient indexing.

Research questions

To what extent does the use of data features influence the performance of multimedia indexing algorithms?

What are the performance challenges in automatic content-based queries?

Relevance of research area to contemporary IT research and practice

There has been a growing interest in online video content with individuals posting their home videos on websites such as www.youtube.com. The intended use of the video data is for entertaining, teaching and marketing. The volume of video data posted is increasing rapidly thus efficient methods of storing the video data and retrieval are of increasing importance. The large number of indexing algorithms current researchers are working on highlights the extent of the problem. In order to support current research, I will evaluate the indexing algorithms for multimedia data, using shape, colour, texture data features and semantic information.

Research Scope

Multimedia databases may contain a wide variety of data such as "text, images, graphic objects, sketches, video and audio" (Nang and Park, 2007) There are a number of technologies and algorithms that are used for searching the multimedia data in different circumstances. Text data can easily be queried using basic SQL statements. This research will only identify technologies and algorithms that support content-based queries.

The next chapter presents a brief overview of multimedia data, which is followed by a statement of research question. The literature review comes after a brief justification of the relevance of the topic to contemporary

Information Technology research. In literature review chapter 2, I investigate the research trends that underpin performance improvements in existing indexing algorithms. The literature review ends by identifying commonly used multimedia indexing methods. A discussion on the research methods is found in chapter 3 of this report. The research methods chapter describes relevant research methods that can be used in evaluating the performance of multimedia indexes. The methods are compared and the most appropriate method is chosen. Chapter 4 provides a detailed analysis of the findings from the literature review. The report concludes with chapter 5, the conclusion, which summarises the findings on this survey research.

Chapter 2: Literature Review

This chapter acknowledges work already done which is related to multimedia indexes. I look at the current research and the improvements being made. I will also highlight any gaps that are still existing in the improvement of the indexing and parsing of multimedia databases.

Algorithm performance is driven by the architecture of the database. Some understanding of the architecture will help in utilising the best options when parsing multimedia images. In this study, I will show the importance of certain components of the video databases architecture. The diagrams will aid to show where certain algorithm functions will be executed and what components exist in the database to support high performance.

The algorithms and tools will then be scrutinised to see what features they are using when parsing video and still image data. The evaluation of existing algorithms and tools will lead to the identification of gaps or weakness in the current algorithms or current research approaches.

VDBMS Architecture

This sub-chapter describes the Video Database Management Systems (VDBMS) architecture. I look at the first attempt by Walid et al, (2002) and then describe in detail the more recent architecture.

Smoliar et al, 1994 described a simplified architecture of the VDBMS in Figure 1 below. The differences between the traditional DBMS system and the VDBMS are highlighted in the detailed description of the VDBMS by Walid et al, (2002).

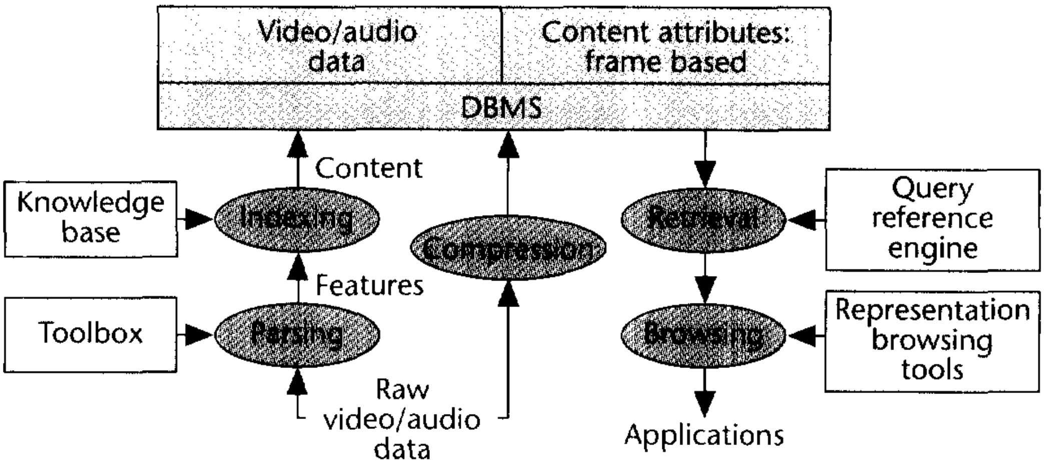

Figure 1 VDBMS architecture (Smoliar and Zhang, 1994)

The DBMS in figure 1 is responsible for “providing facilities for traditional DBMS functions like database definition and creation, data retrieval, data access and organization, data independence, privacy, integration, integrity control, version control, and concurrency support” (Adjeroh & Nwosu, 1997). Another perspective, broadly similar to that of Adjeroh et al, (1997) comes from Smoliar et al, (1994) who says that the basic functions of the DBMS are to provide data independence, integration, concurrency control, persistence privacy and integrity control.

The DBMS controls all the operations on video data. Figure 1 shows that parsing of the video or audio data is done to identify the features to use for indexing. Compressed data is committed to the database. The retrieval of video data goes through a query reference engine which in turn interacts with browsing tools. While Figure 1 is a high level representation of the VDBMS, Figure 2 explains the VDBMS architecture in more detail.

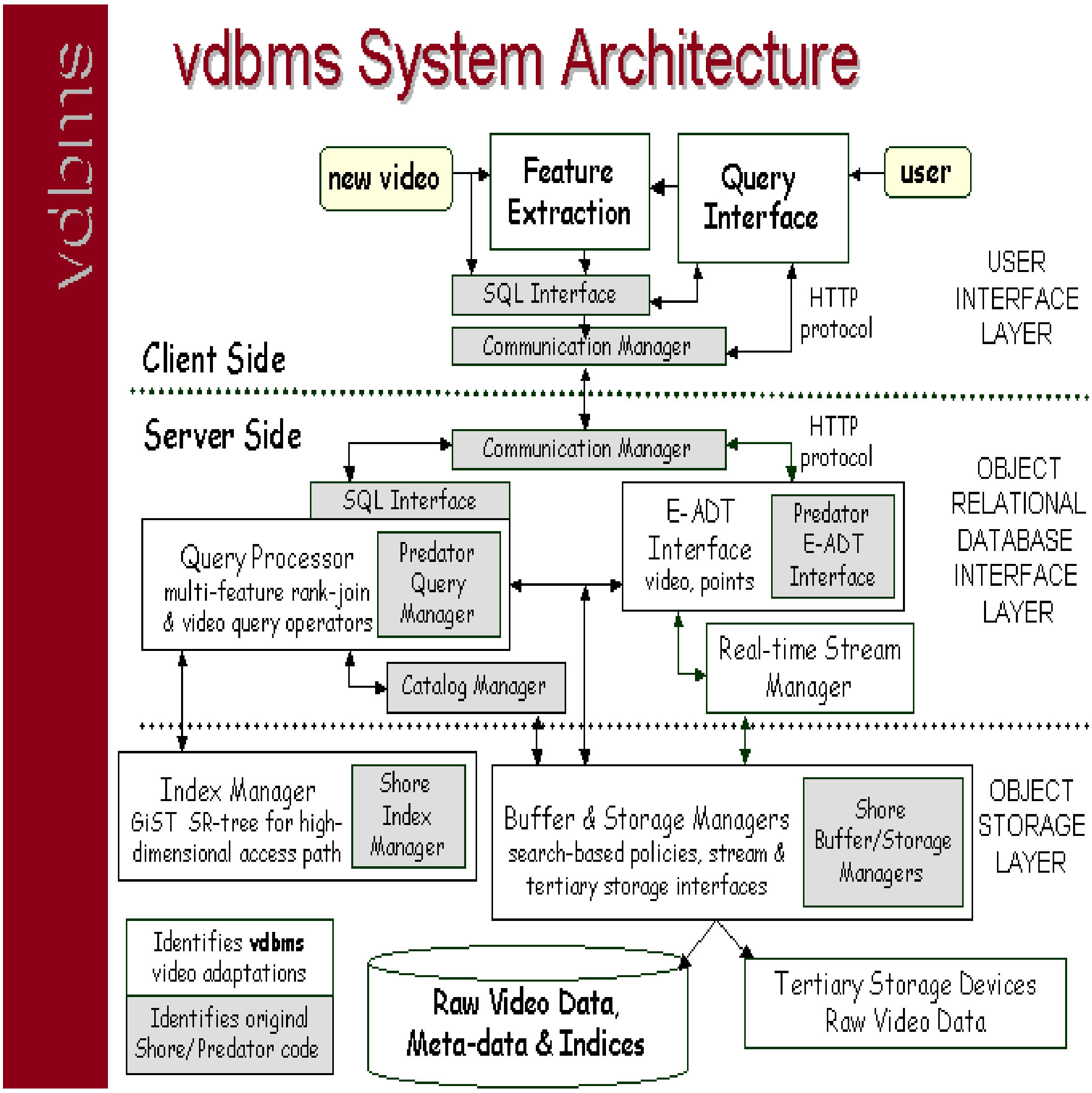

Figure 2 VDBMS layered system architecture retrieved from “A Video Database Management System for Advancing Video Database Research” Walid et al, (2002)

The User Interface Layer

The user interface layer consists of applications software that supports content-based query, search and retrieval queries. A video is preprocessed to identify video features and index video content for searching. The algorithm allows for the detection of shot boundaries by checking the color differences. A shot is a collection of one or more video frames. In the user interface layer, video shots are then processed to extract representative key frames, visual feature descriptors, camera motion classification, spatial and temporal segmentation, and the semantic annotations of domain experts. The VDBMS query interface supports “query by example, query by camera motion type, query by keywords, and SQL queries” (Walid et al, , 2002)

The user can query an image by specifying the image features to match e.g. color and the query interface automatically builds an SQL query to be passed to the server for execution.

Object Relational Database Management Layer

This layer serves to admit, schedule, monitor and serve concurrent video stream requests periodically (Walid et al, 2002). The object database management takes place on the server side and communicates with the client side through the SQL interface module. Query processing is done in this layer where the SQL query from the client layer is processed and the relevant images returned. The returned images are checked to see if they satisfy the requirements of the query. Because video data uses very large amount of space, the stream manager is implemented as multi-threaded modules in order to improve response time. The stream manager also takes care of the buffering.

The Object Storage System Layer

This layer is made up of the storage and buffer managers, index manager and database storage. The buffer manager interacts with the stream manager during large streaming of video data. The page requests are handled by the buffer manager, which ensures that all pages are correctly returned. The storage manager performs operations on requests to determine the frequently accessed data. It uses indexing algorithms to quickly identify requested data.

Major performance challenges reside in this layer; the buffering of data affects the retrieval of the video data and has huge impact on performance. The indexing of the image data needs to be done optimally. The optimization of indexing enhances the speed of returning query data. The performance of indexing depends on the buffering capabilities of the system.

The VDBMS design uses MPEG-7 standard which can handle content with various modalities such as the image, video, audio, speech, graphics, and their combinations (Chang et al, 2001). MPEG-7 defines the format of multimedia data which enable applications to reference multimedia data in a standard way. The following section describes the functions of MPEG-7 in more detail.

MPEG-7

There is a complementary relationship between MPEG-7 and the Multimedia Database Systems (Kosch 2002). It is therefore important to highlight the key features of MPEG-7 that makes it a preferred standard in databases.

MPEG-7 is an audio-visual data content standard which was formally known as the content description interface. MPEG-7 aims to describe the contents of the multimedia data. The existing standards such as MPEG-1, MPEG-2 and MPEG-4 define the coding and representation of multimedia data. The development of the MPEG-7 standard complements the existing MPEG-1, MPEG-2 and MPEG-4 standards. For MPEG-7 to be successful, the code and representation standards have to exist.

As I have noted, the VDBMS is a complex architecture, which requires specialised retrieval methods. There are interfaces that support the querying of video data. There is a greater level of difficulty when operating on video data and thus MPEG-7 standard addresses content with various modalities such as the image, video, audio, speech, graphics, and their combinations (Chang et al, 2001). MPEG-7 defines the standard tools such as the Descriptors, Description Schemes and the Data Definition Language. It defines the structural descriptions of multimedia information. The motivation for the development of MPEG-7 standard was to support and facilitate a range of applications that operate on the audio-visual data. There has been a proliferation of tools for multimedia content creation and distribution that have contributed to the availability of large amounts of audio-visual content.

The Descriptors, define the syntax and semantics of features of the audio-visual content at different levels of abstraction. The Descriptors also define shape, motion, texture and color at a low level of abstraction. At high level of abstraction, events, content genres and abstract concepts are supported. The Description Schemes use the attributes defined by Descriptors to construct complex descriptions and establish the relationships among the multimedia content features. The DDL module defines the XML schema for the Description Schemes and Descriptors which allows for flexible definitions. There are cases where the MPEG-7 Description Schemes and Descriptors cannot fully define the video content, for example in film, news. The tools that are developed depend on DDL that defines the requirements of the specialized tools that are used to describe content specific domains.

According to Chang et al, (2001), the low level features defined by MPEG-7 such as color, shape, etc can be extracted automatically from the multimedia content whereas the users manually define the high level features that define the semantic information. The MPEG-7 does not describe how the features are extracted. The applications are given the flexibility to extract the features by using different approaches but the representation of the extracted features still has to comply with the MPEG-7 standard. Furthermore the standard does not define what the extracted features should be used for.

The increased availability of multimedia content resulted in the demand for more tools and systems for supporting the indexing, searching and managing multimedia content. MPEG-7 aims to provide an interface, to facilitate the creation of new applications, between different tools and applications to improve their interoperability. The MPEG-7 standard ensures that different systems and applications that generate, distribute and manage audio-visual data are interoperable. MPEG-7 defines the syntax and semantics of various description tools (Chang et al, 2001). The content descriptions provided by MPEG-7 help the users and the applications to identify, retrieve and match multimedia information. Many organizations have contributed to the design of the interoperable frameworks and representations of metadata description.

Databases also have growing capabilities for handling MPEG-7 data formats. The VDBMS data model represents the content of all multimedia objects stored in the database. The design facilitates operations such as querying, browsing, indexing, editing and insertion of objects. The operations require use of feature extraction vectors that are obtained through MPEG-7.

The MPEG-7 standard provides the means to encode and decode descriptions to and from binary format (Kosch, 2002). XML format is used to represent the low-level multimedia descriptions. This representation allows for interoperability and databases can decide how to encode the different elements of the description (Kosch 2002). The multimedia data model can be implemented in databases using the extensible DBMS which support the XML SQL utility (Kosch 2002). Most major databases including Oracle, DB2 and SQL server, now support XML.

Multimedia Databases

A video object consists of features such as frame resolution, number of bits per pixel, compression information and duration and colour models (Grosky 1997). A frame is a data object. In traditional databases, a frame can be compared to a row/tuple. Smoliar et al, (1994) defines colour, texture and shape as the basic features of a frame. A data object can therefore qualify as a frame when it has these basic features defined by the MPEG-7 standard for multimedia data.

There are difficulties in defining the attributes of video data in advance (Oomoto et al, 1993). A scene with the same sequence of video frame may have different set of attribute definitions for each frame. This is due to the temporal nature of multimedia data (Smoliar et al, 1994). A single video may have frames with very different feature contents, that is, the colour texture in one frame is not the same as the colour texture in another frame. This makes it hard to establish the relationship between frames when automatically parsing the contents of the video.

Flickner et al, (1995) in their research about Query by Image and Video Content (QBIC) defined a scene as a single instance of a frame. An object is part of a scene. Therefore a frame consists of an object and the description of its features. Oomoto et al, (1993) say that each video frame can also be regarded as an object and a meaningful scene is a sequence of video frames. While Flickner et al, (1995) say that a scene can be a single frame, Oomoto et al, (1993) agree but the further explain that for the scene to be meaningful, it has to consist of a sequence of frames.
Oomoto et al, (1993) found that some meaningful scenes may overlap and can be found in other scenes. It is recommended that frames share such scenes.

Text data is sometimes used to identify and index video data. Text data is added to the video/image data during cataloguing. Early research confirms that text data about video contents is either manually captured or automatically generated text or meta-data from the video contents (Chang et al, 2001). The MPEG-7 standard allows for low level attributes to be automatically generated. The semantics of the multimedia data can be described through human intervention.

The nature of video data described above illustrates how difficult it is to search and retrieve data.

Gudivada et al, (1995) proposed a content-based image retrieval system (CIBR). The prototype they developed allows for retrieval by sketch, a user outlines an image and the system retrieves a similar image from the database. Gudivada et al, (1995) managed to extract object centroids and boundaries automatically; however the extraction of some logical features needed human intervention. The behaviour of the retrieval process was described as expensive and difficult. This is due to the fact that the meaning of the video content is domain specific (Gudivada et al, 1995).

Another method is to retrieve by motion whereby a video's spatio-temporal image sequences that match the defined action are returned. The system automatically scans the contents of the video and generates descriptive data about the video. The generated data is then used for the retrieval of video.

Queries are in most cases imprecise. It is difficult to find an exact match between video data attributes therefore retrieval is based on similarity rather than exact matches (Adjeroh et al, 1997).

When returned results are based on similarity, the videos returned in any one query are likely to be more than one (Grosky, 1997). This introduces performance issues like media access storage (buffering), indexing and retrieval.

In order to understand how to optimise the queries, it is important also to be aware of the features that are used in retrieving image data. Adjeroh et al, (1997) say attributes such as colour, shape, texture, spatial information, symbolic strings, have been used to index images. Smoliar et al, (1994) previously used the same image features such as shape, texture, color and semantic information to develop a prototype system that uses the features for fast indexing abilities. These features are automatically derived from the video content.

The difficulty in using content-based indexing is that the same data could mean different things to different people. Users have very diverse information needs; therefore a single feature may not be sufficient for indexing a single video/image feature (Adjeroh et al, 1997). For example, a computer can easily identify a horse in an art object but it is almost impossible for the computer to automatically determine the meaning of the art object. Only the person looking at the object is able to determine the meaning.

Adjeroh et al, (1997) argue that automatic indexing is affected by the diverse content in the multimedia data. It is difficult to identify a feature that is appropriate for indexing in any given environment. Smoliar et al, (1994) suggest that retrieval should be based on features of static images but this demands understanding of the temporal features that are important for retrieval and then features can be computed automatically.

Grosky (1997) proposes using an "automatic combination of indexing methodologies for each feature to make a single index for a complex feature based on the individual features". Grosky (1997) seems to suggest that a composite index which uniquely identifies a video frame should be used.
Smoliar et al, (1994) say that different techniques to inspect the contents of different images can be applied and the techniques are then combined by means far more sophisticated that for textual databases.

Content-based retrieval in multimedia database calls for content-based indexing techniques. The challenges to content-based indexing are mainly a result of the fact that the features of the objects are characterised by multimodal reference. Wu (1997) developed an indexing algorithm called

ContIndex, after it was found that certain semantics have to be added to the content for indexing. The tool has the capability to self-organise nodes to match the context. While Wu (1997)'s proposed tool was relevant then, recent research focuses on automatically generating the semantic data rather than reorganise nodes (Adams et al, 2003).

Adams et al, (2003) came up with a new approach for searching images by content. Adams' work was motivated by the limitations he found in the query-by-example (QBE) systems. While QBE systems allowed users to provide an example image (visual content) of the image that they seek, certain type of users were found to be more interested in searching by inputting keywords (Adams et al, 2003). This changed the focus from QBE to query-by-keyword (QBK). The objective is to develop a system for labelling and retrieval of the generic semantic concepts in video. The approach used in this project is to find the high level semantic features first and then define the atomic semantic concepts such as sky, water, soil and tree. Statistical models are then used for the automatic labelling of other video with similar atomic concepts. The QBK automatically generates textual keywords from the image content and uses the keywords for indexing. This new approach replaces the error-prone manual addition of image meta-data by users.

In the next section I discuss some tool and algorithms used in search by content. The tools were carefully chosen to show the beginning and the progression of the image parsing technologies.

Tools and Algorithms

This chapter discusses various tools and algorithms developed in order to make content based searching easier and efficient. The tools address various layers of the VDBMS architecture. In this chapter I look at algorithms that use content based searching methods.

We first look at the QBIC system that was developed by Flickner et al, (1995). The QBIC system forms the basis of content based searching. All the other tools developed afterwards strongly reference the QBIC approach and try to improve the performance of the QBIC application. All algorithms and tools were developed because the text added to images was not accurate enough to represent the actual contents of the image.

Flickner et al, (1995) developed a query by image content system to explore content-based retrieval methods, after finding that the image descriptions added by users were not fully representative of image content. The QBIC system facilitates the retrieval of image data using the image content features. Users can perform content-based queries such as querying by an image or retrieving an image from the database using a sketch of the image. Query by image content allows queries based on example or sketch images, color and semantic patterns (Flickner et al, 1995).

The QBIC system allows searching using color, texture, shape, videos and semantic information. The system consists of a database population component and the query component. Flickner et al, (1995) split the video data into three components, which are shot detection, building frames from each shot before storing it into the database and the derivation of layered coherently moving structures. The graphical query interface allows the user to generate the image features, which will be used to search the database for similar features. QBIC filters the data and compute a similarity metric.

Flickner et al, (1995) found that there were performance overheads caused by the lack of proper indexing of the image data. They found that the existing indexing was based on the textual contents of the image and not the visual content. Flickner et al, (1995) proposed that indexing of multimedia data be based on both text and visual content for better performance of queries. Two years latter Smith and Chang (1997) developed a fully automated content-based image query system called VisualSEEK.

The VisualSEEK system integrates feature-based image indexing with spatial query methods. Smith and Chang's (1997) objective of content-based image querying is to find and retrieve images that satisfy the criteria of similarity matching. Similarity matching is the retrieval from a database of images that have similar features to the features defined in the query (Smith and Chang, (1997)).

The prototype system searches by visual features of an image. As with the QBIC system, VisualSEEK allows the user to form a query by providing a sketch diagram of the image's spatial arrangements of the color regions. VisualSEEK automatically extracts and indexes color regions of an image and find the images that contain similar color. The extraction and indexing of image data is a huge improvement over traditional methods and it addresses the performance issues identified by Flickner et al, (1995) in QBIC.

VisualSEEK uses "efficient indexing techniques for color information, region sizes, absolute and relative spatial locations "(Smith and Chang, (1997)).

Although there was extensive research in using the simple image features (such as colour and shape), color histograms, efficient indexing structures and use of effective pre-filtering techniques, Smith and Chang (1997) found that the then current image indexing methods for content-based querying did not use spatial information and spatial relationships . Image spatial information includes information such as the size of the image, location and its relationship with other regions. VisualSEEK provides both: comparison of similar regions in an image and spatial query using color of image.

VisualSEEK decomposes an image into feature properties for example color region and spatial information such as size, location and relationship with other regions. If an image has different colors, VisualSEEK is able to determine which colors are close together. VisualSEEK aims to improve image retrieval by automatically extracting feature regions, query by both spatial and feature information.

Chang et al, 1998 identified the limitations of VisalSEEK (Smith and Chang, (1997)), and QBIC as the parsing of still images only. The algorithms and tools only performed well when searching still images. Although Smith et al, (1997) used spatial information and its relationship to other regions, the limitation is that the tool only scans a single image and the relationship relates to the image feature relationship within the image only rather than relationships of the image to other images in the database. Chang et al, 1998 proposed a new system for the efficient querying of video content using spatio-temporal information. Chang et al, (1998) proposed algorithms for automated video segmentation (such as motion, color, and edges) and tracking which resulted in the VideoQ system. VideoQ uses scene change detection methods to create video shots. VideoQ, unlike VisualSEEK (Smith and Chang, (1997)) and QBIC (Flickner et al, 1995), can detect abrupt and transitional scene changes.

VideoQ features include

• Automatic video object segmentation and tracking;

• A rich visual feature library including color, texture, shape, and motion;

• Query with multiple objects;

- Spatio-temporal constraints on the query;

• Interactive querying and browsing over the World-Wide Web;

• Compressed-domain video manipulation.

VideoQ introduced another data type that can be stored in multimedia databases for content based video searching. VideoQ is based on the VDBMS architecture defined by Smoliar et al, (1994). The performance of the video content searching algorithm depends on the VDBMS architecture. The current process for representing the visual content of a video in VideoQ is to split the video into shots. A shot is a collection of similar frames. Frames are used for computing similarity between video sequences.

Chang et al (1999) proposed a “tree structured key frame hierarchy” which is based on the hierarchical algorithm. The new algorithm was developed after it was found that the existing algorithms were losing visual information when representing a shot. The aim of this study is to improve the performance of spatial database indexing techniques. Chang et al, (1999)’s proposed video indexing algorithm supports content-based queries and accurately represents the features of the original video.

The tree-structured key hierarchy reduces the number of comparisons for the retrieval of an image. Chang et al, (1999) found that despite improved quality of video and indexing methods, the continuous motion information was still lost after the algorithm computations. Whilst Chang et al, (1999) improve the amount of image information retained after index computations; more work was still required to improve the retrieval of images based on content.

Kiranyaz et al, 2003 introduces MUVIS system for content-based multimedia indexing and retrieval framework. MUVIS is a content-based information retrieval system which was initially developed in 1998 to support querying image databases using semantic features such as those defined and improved in Naphade et al, 2005. The revisiting of the MUVIS system design was largely due to the major improvements in the content based searching algorithms. The MUVIS framework (Kiranyaz et al, 2003) supports video scene detection, dynamic feature extraction algorithms, audio and video encoding schemes, multimedia conversion, real-time audio and video capturing.

The MUVIS however only uses color, texture and shape as semantic features but also supports indexing, browsing and querying multimedia data such as audio, video and images. The MUVIS system supports various formats such as MPEG-4, MP3. The system defines an interface which can be integrated into third party feature extraction algorithms. The paper by Kiranyaz 2003 defines the framework and techniques for retrieval of multimedia data. The motivation for the MUVIS is that digital visual information requires massive storage and management, therefore the objective of the research is to try and minimize the problems that already exist.

The MUVIS framework is made up of four components which include VDatabase, IDatabase, DbsEditor and Mbrowser (Kiranyaz et al, 2003). The VDatabase and IDatabase define the database creation framework. DbsEditor is responsible for the indexing of the multimedia databases. DBAEditor performs the offline feature extraction processing, other major tasks include deleting, updating or adding video feature clips to the database. The DbsEditor performs the database tasks for multimedia databases.

The Mbrowser component performs the retrieval and browsing of video content. It performs key-frame browsing, scene detection, random access support for video playback and displays information related to the extracted features of the active database (Kiranyaz et al, 2003).

The MUVIS framework applies visual feature extraction to video clips and images. According to (Kiranyaz et al, 2003), feature extraction algorithms can be implemented as independent modules via the feature extraction interface (Kiranyaz et al, 2003). The algorithms need to be implemented as DLL to be recognized by the FEX API.

Kiranyaz et, al. 2003 defines the four stages in audio indexing as follows Classification and segmentation, Audio Framing, Audio Feature Extraction, Key-Framing via Minimum Spanning Tree Clustering and finally the indexing over the extracted Key-Frames.

Ünel et al, 2004, present a query optimization strategy based on a system called BilVideo. BilVideo is a spatio-temporal query optimization-reordering algorithm. The optimization strategy is made up of two parts, which are internal node reordering and leaf node reordering. The algorithms use the parent child relationship to reorder nodes. The leaf node-reordering algorithm makes use of statistical information to sort the contents of the leaf nodes. Ünel et al, (2004) believe that their query processing algorithms perform better because reordering is done in two dimensions. This was backed by the performance tests conducted on the query processor. The performance of the algorithm can be improved by adapting parallel query processing.

Semantic visual features are the content features of a video or image such as people, car, and cityscape. Mu (2006) describes a shot level video browsing method based on semantic visual features for content-based retrieval. Mu (2006) uses the semantic feature vector to calculate the score of similarity between two key frames. Key frames with similar semantic visual features are retrieved based on the score. Mu (2006) in this project compares three video retrieval and browsing systems including temporal neighbour, semantic visual feature and fused browsing systems. After the comparisons, Mu (2006) found that non-visual centric tasks are less effective and efficient compared to visual centric features browsing. Temporal neighbour browsing was found to be a common navigation method used in shot level content-based retrieval (Mu, 2006). A shot as defined by Mu (2006), represents a series of consecutive frames with no abrupt transition.

Mu (2006) also defines visual features such as color histograms, text, camera movement and face detection for defining similarity matching. Mu (2006) defines a visual similarity browsing function as one that allows a user to “find similar shots matching people or car”. Slow response was highlighted as a major bottleneck in the content-based retrieval system (Mu, 2006).

Shape representation is based on two techniques. The boundary-based methods use the contour or border of the shape object and do not make use of the interior details of the object such as color. Region-based techniques use both the boundary and the internal details of a shape object.

Shahabi and Safar (2007) compared the performance of boundary-based techniques, Fourier, grid-based method, Delaunay triangulation method, and shape representation and similarity measures in their MBC-TPVAS algorithm which was tested on a database of noise shapes and its performance was measured by running queries on the database.

Shahabi and Safar (2007) proposed a shape based retrieval method called MBC-TPVAS. MBC-TPVAS uses recall and precision to measure the effectiveness of similarity searching and retrieval using image shape points. Recall is the ratio between the total number of retrieved shape objects and the total number of shape objects in a database. Precision is similar to recall but only measures the number of similar objects retrieved and total number of shape objects retrieved (Shahabi and Safar, 2007). Precision measures the retrieval accuracy.

MBC-TPVAS is as effective as the other shape based representation methods. MBC-TPVAS does not require a lot of resources when computing shape signatures of objects; it requires low storage and does not require normalization of the objects. MBC-TPVAS directly supports query types such as “return images that have a similar shape to the sketch provided”.

The algorithms described so far focus on different aspects on content based searching. The VisalSEEK (Smith and Chang, (1997)) and QBIC (Flickner et al, 1995) systems are similar in that they are both based on still image searching. However, because the images used in the development of the algorithms are different and have different semantic features, it is difficult to compare the performance of the two algorithms. This demonstrates that performance improvements are relative to the set of images used in the development of the algorithm. VideoQ (Chang et al, 1998) searches video images which are different from still images in both format and semantics but the authors still claim they are trying to improve the current still image based algorithms. A closer look at similar algorithms and tools for video image searching shows that there is a lack of a standard framework for testing the performance of algorithms. This is because the environment in which an algorithm is developed in may be different from the environment another algorithm is run in. This lack of standard testing environment resulted in Smeaton and Over (2003) developing the TRECVID project.

Smeaton and Over (2003) describe the TRECVID project which is used for benchmarking the effectiveness of information retrieval on digital video. The TRECVID project facilitates the participation of groups in investigation of techniques for digital video information retrieval. TRECVID (Smeaton and Over, 2003) provides a test collection and a common environment for testing and comparing the effectiveness of video information retrieval tools. The TRECVID exercises benchmark the effectiveness of systems in carrying out various information retrieval on multimedia data.

Smeaton and Over (2003) describe a TRECVID exercise which aims to benchmark performance of algorithms using shot boundary detection, feature extraction and searching digital video. The TRECVID exercise involves 17 participants testing different algorithms they developed using the same testing environment but different image features.

It was found that in “Shot Boundary Detection” task, the transitions are harder to recognise than abrupt ones (Smeaton and Over, 2003)

The 17 participants also carried out a feature detection task which involves automatic extraction of high-level features from video. The purpose of extracting features is to help in navigating and searching video content. The TRECVID exercise provided a benchmark for the effectiveness of automatic feature extraction. Participants are able to exchange the output of the feature detection. Some of the features used in this exercise include outdoors, indoors, face, people, cityscape, landscape, test overlay, speech, instrumental sound and monologue. Feature detection involves returning images or video which closely match the features in the input picture, for example when search using the “People” feature, images containing a group of humans are returned.

The results from participants were found to vary in their dispersion among features. Smeaton and Over, 2003 found that some results were very poor for features such as cityscape, landscape, people, indoors, text overlay and monologue. Smeaton and Over, 2003 blames the participating groups for failing to put sufficient effort or cause operation errors. Smeaton and Over, 2003 thinks that the groups underestimated the complexity of the task. However, there was good performance in using sound and speech features.

The TRECVID project’s main objective is to provide a common testbed and evaluation metrics for allowing comparisons across systems. In conclusion, Smeaton and Over, 2003, found that the evaluation of the effectiveness of different approaches to information retrieval from video data is rapidly becoming of crucial importance.

Chapter 3: Research Methods

In this chapter I discuss research methods that may be used in determining the effect of multimedia data features on performance of data indexing. The research methods are evaluated according to their relevance to the topic. In addition to highlighting possible research methods in the performance of multimedia data indexing algorithms, this chapter justifies the choice of research method used in this research.

Survey Research

The process of carrying out a survey involves determining the objectives of the survey, the information required and the data collected (Collis & Hussey, 2003). A survey may be used in determining the searching methods users use in multimedia databases. Interviews or questionnaires can be used for gathering information about the current features used in searching multimedia data and the features that the users would prefer to use for searching. The results of the survey would help researchers to improve the performance of the queries by ensuring that the indexing of such features is implemented.

Collis & Hussey, (2003) categorised survey research into two parts, the analytical and the descriptive research methods. Analytical research aims to identify features and their relationships in a research scope. Descriptive research aims to quantify the accuracy of a particular feature (Collis & Hussey, 2003). While analytical research may be used to identify the data features used in multimedia data indexing, descriptive research methods may be used to enumerate the occurrence of a feature. The results of analytical research may not be very useful in this research topic as the objective is not to find occurrence of features but to find the effect of data features on performance of indexing algorithms.

Interviewing multimedia application users is another option that can be used to determine the performance of content-based queries; however the results of the queries would depend on the user environment.

Constructive Research

Researchers have applied constructive research (Kasanen et al, 1993) in the design and implementation of multimedia indexing algorithms (Adams et al, 2003; Donderler et al, 2003; Gudivada et al, 1995; Flickner et al, 1995; Oomoto et al, 1993). Constructive research seeks to demonstrate feasibility of an approach, produce proof of concept and build a prototype. Constructive research is used by researchers concerned with developing frameworks and models, which help to create new artifacts. As can be deduced from the definition above, constructive research can only be used in cases where a new algorithm or tool has been developed. This research method may not be necessarily used when no new algorithms are being developed. Constructive research was not used in this study because the objective of this research is not to develop new artefacts but to identify and evaluate existing tools and algorithms. There are many ways of evaluating existing algorithms and if it involves implementing the algorithms for testing purposes, experimental research method would be the appropriate method to use.

Experimental Design

Experimental research seeks to confirm a theory, explore a relationship, evaluate model accuracy and validate a measure. This research methodology can be used to verify whether an indexing algorithm works as intended. The process will involve following instructions described by the researcher and implementing the algorithm. The objective would be to evaluate model accuracy and make it possible to validate the authors' research results. The possible conclusions from using the experiment design would be that the algorithm works as expected or it does not work as described. The performance of indexing algorithms can be tested and results compared to other experiments. The set-up of the apparatus needs to be considered as a factor that may alter the results. Experimental design results on indexing performance are more accurate than results of survey research or meta-analysis. While experimental design is the favoured research method for this topic, it demands a lot of resources. It will take more time to implement and test algorithms. Resources such as the hardware and software will need to be available for the experiment. Quantitative analysis needs to be considered for this topic.

Quantitative analysis

While a significant quantum of research was done using design science research methods (March & Simth, 1995) that end with an artefact such as a prototype for demonstrating the practical implementation of the algorithms, surveys have also been applied in the area of multimedia databases (Antani , Kasturi & Jain, 2001).

Quantitative research synthesis investigates study level attributes such as the design of research, trends in results, quality and identification of gaps that require new studies. By using meta-analysis as a research method in this study, I will be able to collate the available indexing algorithms and evaluate their performance. The key features of a quantitative research synthesis are the inclusion of protocols for study identification and the criteria to be used for inclusion and exclusion of research papers.

Meta-Analysis

The method to be used in this research is meta-analysis. Meta-analysis is also know as the synthesis of research findings, it is a quantitative research method that presents the potential of finding areas of agreement in the field of science and areas that still need further research (Mosteller & Colditz, 1996).

Quantitative research is not new in the area of indexing multimedia databases. Antani, Kasturi and Jain (2001) used a survey to identify recent trends on the pattern recognition methods for abstraction, indexing and retrieval of multimedia data. They reviewed past research and used meta-analysis research method to help them support their conclusions. Idris and Panchanathan (1997) also used quantitative research synthesis method to identify image and video techniques for indexing.

Research synthesis' strength is in producing answers to questions that cannot be addressed by individual studies (Mosteller & Colditz, 1996). The research method to be used in this study will help identify the current trends in multimedia indexing and the features that are commonly used in today's algorithms. This will keep us from relying on the results of a single study. Research synthesis provides the opportunity to view the whole picture in the area of multimedia data indexing and understand the differences among the algorithms.

Rosenthal and DiMatteo (2001) say that the advantage of meta-analysis is that it prevents our reliance on just one finding and help us realize that other findings also have powerful evidence. Small features are also considered in the overall goal of a research. One of the reasons for choosing meta-analysis is that it demands focused, one-degree of freedom contrasts, research becomes more attentive to the precise formulation of the questions asked and the answers extracted.

There has been a lot of work done in the area of indexing multimedia databases for optimal query processing. The constructive research papers make individual conclusions based on the method and purpose of the research. By using meta-analysis as a research method in this study, I was able to collate the available indexing algorithms and evaluate their performance.

Rosenthal and DiMatteo (2001) say that meta-analysis allows researchers to provide conclusions from multiple rigorous researches that are more credible and accurate. One of Meta-analysis goals is to identify source of heterogeneity. Mosteller and Colditz (1996) found that the application of research synthesis has to include key issues such as choosing an outcome measure, assessing the quality and applying regression model for combining collected data.

This study identifies and evaluates algorithms and tools used for parsing multimedia data. I have collated recent research papers and identified a couple of algorithms I thought would contribute more to the objectives of this study.

A high performing algorithm is one that returns fewer images matching a user defined criterion. An algorithm which returns fewer images will be regarded as more efficient than one that returns more images. This assumption is only true in similarity matching. The major assumption here is that the fewer images returned closely match the sketch provided by the user using features such as text, color, shapes and semantic information. The algorithm that uses most features or all features identified by other researchers is more likely to be robust than an algorithm that uses some features. Papers were selected based on the assumption above. There was careful consideration for the element of bias. I have chosen studies that defined algorithms and then implemented them. The implementation of an algorithm meant that the theory was tested against practice.

Research papers were further evaluated based on the semantic features they use. I have tried to include papers that cover various semantic features used in content based searching. I have identified papers using shape, texture and colour for searching and indexing multimedia database. Furthermore, I looked at prior research that address still images and prior research that address video images. The challenges in parsing and searching these data types are varied. The next chapter will discuss the challenges that exist in querying multimedia databases and the problems in the research direction.

Chapter 4: Analysis

This research has found that meta-data (text added to an image or video) does not adequately annotate or describe an image or video. Text based description was found to be incomplete (Shahabi and Safar, 2007). Most research papers reviewed generally agree and have in most cases introduced content-based queries and indexing methods. The solutions proposed by researchers are so varied that they need scrutiny.

There has been significant research in the use of colour, shape and texture for image searching and indexing. The use of these features in content searching started in the mid 90s with Flickner et al, (1995) developing the QBIC algorithm. The tools used a combination of features including image semantic information. The major problem then was the focus on both video and still images. The parsing and indexing of video images was not robust enough. The database architecture for storing videos and images was the major bottleneck. Semantic information about the image was lost when an image is inserted into databases. The algorithms treated a video as a collection of continuous still images. Therefore in the end the video data was processed as still images. The video images ended up losing semantic relationship information. Databases did not support image data types (normally stored as a reference to a jpeg/mpeg file stored on the computer). Despite the technology challenges, the QBIC system defined the algorithms for shot detection and video frames.

The main focus of the QBIC system was on searching images and the use of a number of semantic features helped in the performance of the algorithms. The searching improved and more specific images were returned by the queries.

Similarity matching returned images of similar color, shape or text. The indexing of the images was still an issue in the QBIC system as well as in VisualSEEK (Flickner et al, 1995) where text and content data were used for the indexing of images. Smith and Chang (1997) extended these features by proposing the use of spatial information and spatial relationships.

The use of spatial relationships helps in the efficient retrieval of images. Although this is an improvement to QBIC, the concept is still based on the same principle – image content. VisualSEEK has the ability to tell the density of each color in an image. One particular problem with the VisualSEEK system which I found to be present in the QBIC system as well is that the focus is on still images and not video images. Therefore the spatial relationships and spatial information VisualSEEK and QBIC systems introduced is only limited to the relationship of features in a single image. This is a limitation in most algorithms developed in the 90s.

Research shows that the focus diverted from only being able to index and query using the contents of images or videos to being able to return the original quality of the video or image. Researchers acknowledged the fact that there was significant research in the use of semantic features for indexing and querying, therefore they decided to focus on the quality of the images. I can see that in 1998, Chang et al, proposed a video indexing algorithm, which supports content-based queries and accurately represented the features of the original video. A major limitation in Chang et al's research is the lack of suitable database architecture to support algorithms. A video image, for example, would lose the information about the relationship between the frames because the database did not have the appropriate methods to represent the spatio-temporal relationships between frames.

The VideoQ system (Chang et al, 1998) application/tool, which, is a very useful tool was a major improvement to query by image content. The tool was developed after it was found that the existing algorithms and tools were based on still images only (QBIC, VisualSEEK). VideoQ then focused on the improvement of indexing of video images. This makes it hard to compare or argue that VideoQ is indeed an improvement to the QBIC system and the VisualSEEK system. So far since the inception of QBIC, I can see that the performance of indexing algorithms improved through use of extra multimedia data features.

The problem found with current research is that either the research is using different image features, such as color, shape, text, from the ones used before or it is using different data (Smeaton and Over, 2003). There are obvious differences between video data and still data with more challenges in the former.

This research therefore acknowledges that VideoQ is an improvement to the QBIC and VisualSEEK systems but cannot compare the VideoQ system to the still image based tools and algorithms. This is an issue that needs to be highlighted in future research.

We would suggest that future research look at the algorithms and tools for parsing still images only and those for parsing video images be reviewed separately. This is only an issue when I try to compare the performance of the algorithms in terms of indexing and querying images by their contents. This research however is about the identification of tools and algorithms for parsing multimedia databases of which QBIC, VisualSEEK and VideoQ are all still relevant.

The VDBMS architecture designed by Smoliar et al, (1994) lacks the necessary details about how the video data should be handled from the application level. Smoliar et al, (1994) only define the Video/audio data and how their content attributes are stored as frames. The architectural limitations meant that researchers were limited in the ways they can design queries. There is evidence that the development of algorithms was focusing only on indexing and parsing raw video and image data (Smoliar et al, 1994).

I also found that most research was based on the textual database SQL mindset. The researchers wanted to develop content-based queries using traditional SQL syntax which was simply not designed for these types of tasks. For example, Select * from birds where color = ‘RED’, is not the sort of query that would support query by image contents. More complex queries, which make use of the semantic features and spatial relationships, needed to be used but are hard to type in on the command line.

There were interesting developments regarding the query interface for multimedia object retrieval. The user experience part was not adequately addressed. There were significant developments with regards to the user interface. The architecture defined by Walid et al, (2002) fully represents the user interface layer. The main objective was to enable users to query multimedia databases using a sketch or a copy of the image. The user can only see the query interface layer which interacts with the feature extraction, SQL interface and the communication manager. A new video goes through a feature extraction procedure, which interacts with the SQL interface. More complex queries using spatial information could be automatically generated using the information extracted from the image. Similarity matching support enabled for images with similar features to the image in the query to be returned.

In future, I am hoping that users will be able to query for images that are not only similar to the image provided but also query images that may also match the colour distributions specified. These types of requirements thus make it very difficult to formulate structured SQL-type queries. A good query interface is one that allows the user to say “give me all running horses” and the query returns images of running horses. At the moment current systems are capable of returning horses based on colour and shape, so if the shape of the horse looks like it is running then running horses are returned.

There are still gaps in the research of multimedia databases and one of them is the inability to search for disguised faces. For example in a crime database, I may not be able to compare an image of a robber with balaclava-clad face to their uncovered face. There are even more complex cases where a person removes their facial hair or undergoes plastic surgery.

I have found from research that MPEG-1, MPEG-2, MPEG-3 and MPEG-4 are underlying codec for multimedia standards, which define mainly how multimedia data must be stored as wavelets. The development of these standards improved the storage and retrieval means for multimedia databases. The compression and decompression processes resulted in major data loss. The data loss may not be detectable by the human eye but the image before compression was found to be different on a bit by bit level. This affects the overall quality of the image.

A new standard MPEG-7, addresses content with various modalities such as the image, video, audio, speech, graphics, and their combinations (Chang et al, 2001). MPEG-7 defines the standard format of multimedia data and enables applications to reference multimedia data in a standard way. This standardization allowed for interoperability of applications. As MPEG-7 defines the syntax and semantics of description tools, application are now able to use data from other applications.

Research has found that since the development of MPEG-7, there has been a proliferation of algorithms and tools based on this standard. The weaknesses I found in these tools and algorithms (discussed in chapter 2) tend to highlight MPEG-7 weaknesses such as data loss, searching mechanisms and the query and indexing methods. The definition of standards in MPEG-7 meant that the coding and decoding of multimedia data became easier and therefore quicker.

Chapter 5: Conclusion

The objective of this research was to find out whether the perfomance of multimedia indexing algorithms is dependent on the set of image data features used in indexing. Furthermore, this research identified the perfomance challenges facing automatic content-based queries on large databases. Ideally, an algorithm should only return stored images that match a user defined criterion and no others, which I term redundant. In this chapter, I will discuss the trends in algorithm development, performance improvements and the user interface developments.

Most algorithms are based on the VDBMS framework. The VDBMS architecture defines and controls the development and optimisation of tools and algorithms, which parse and index multimedia databases. The VDBMS design uses the MPEG-7 standard, which can handle content with various modalities such as the image, video, audio, speech, graphics, and their combinations (Chang et al, 2001). MPEG-7 defines the format of multimedia data, which enable applications to reference multimedia data in a standard way.

The earliest algorithms to be developed focused on content-based searching for which there was only partial support for automation. There were, however, improvements that led to fully automated content-based image feature searching and indexing. However, automatic feature extraction was performed only on still images (Smith & Chang, 1997). One major limitation of these early algorithms is that they treated video as a collection of continuous still images. This resulted in video data being processed as a series of still images. The video images thus ended up losing important semantic information about relationships between frames in a video sequence. The processing of video data demanded a lot of processing resources. Algorithm performance was affected by the increased number of features in video data. The performance of the systems on parsing video data needed improvement (Ünel et al, 2004), as there are more features in video data than in still image data.

Improvements in the database technology meant that the video data could be fully represented in the database without any loss of feature relationship information. In order to leverage such improvements in database technology, the algorithms needed to improve their support for automatic feature extraction on video data (Chang et. al 1998). Subsequent systems improved to support video scene detection (Kiranyaz et al, 2003) and were able to handle spatio-temporal relationships between video frames.

In 2003, comparing performance of systems became difficult because of differing environments and image data but there is now a test-bed for the evaluation and testing of the performance of algorithms (Smeaton and Over, 2003).

Most research studies compared the performance of indexing algorithms on still images to indexing algorithms on video images (Smith and Chang ,1997, Chang et al, 1998, Kiranyaz et al, 2003, Naphade et al, 2005, Ünel et al, 2004, Mu, 2006). Algorithms for parsing still images should be reviewed separately from video images in future research. This is because the video data has spatio-temporal features that impact the performance of parsing algorithms.

The performance of indexing algorithms improved through the use of extra multimedia data features such as shape, texture, semantic information (Chapter 2: Tool and algorithms).

There were interesting developments regarding the query interface for multimedia object retrieval. The VDBMS architecture designed by Smoliar et al, 1994 lacked the necessary detail about how the video data should be handled at the application level. Smoliar et al, 1994 only define the audio-video data and how their content attributes are stored as frames. The lack of a user interface layer was a limitation to researchers in query design. The query interfaces were not flexible, making it hard for users to specify features for extraction.

User experience was significantly improved through the development of a user interface layer in the architecture defined by Walid et al, (2002). This enabled users to query multimedia databases using a sketch or a copy of the image (Liu et al, 2007).

In future, I am hoping that I will be able to search for images by specifying their colour distributions e.g. 40% red, 20% blue and 40% white. This enables users to query for images that are not only similar to the image provided but also query images that may also match the colour distributions specified. There are also other gaps in the research of multimedia databases and which include the inability to search for disguised faces.

There has been significant research and success in the use of colour, shape and texture for image searching and indexing. In conclusion, performance of algorithms can be improved by increased number of semantic features, such as shape, motion, texture and color, used together in combination in multimedia indexing and searching/querying algorithms.

References

Adams, W.H., Iyengar, G.,Lin, C.Y., Naphade, M. R., Neti, C., 2003, Semantic indexing of multimedia content using visual, audio, and text cues, EURASIP Journal on Applied Signal Processing (2), pp 170--185.

Adjeroh, D.A. and Nwosu, K.C., 1997, Multimedia database management-requirements and issues, IEEE Multimedia, 4(3), pp 24-33.

Antani, S., Katsuri, R., Jain, R., 2001,A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video, 34(5), pp 945-965

Chang, S.F., and Chen, W., Meng, H.J., Sundaram, H., 1998, A fully automated content-based video search engine supporting multi-objects spatio-temporal queries, IEEE Trans. Circuit Syst. Video Technol., 8, pp 602-615

Chang S.F., Sull, S., Lee, S., 1999, Efficient video indexing scheme for content-based retrieval, Circuits and Systems for Video Technology, IEEE Transactions, 9(8), pp 1269-1279

Chang, S.F., Sikota, T., Purl, A., 2001, Overview of the MPEG-7 standard, Circuits and systems for video technology, IEEE transactions, 11(6), pp 688-695

Collis, J., Hussey, R. 2003, Business research: A practical guide for undergraduate and postgraduate students. New York: Palgrave Macmillan.

Donderler, E.M., Saykol, E., Ulusoy, O., Gudukbay, U., 2003, BilVideo: A Video Database Management System, IEEE Computer Society Press, 10(1), pp 66-70

Unel, G., Donderler, M., Ulusoy, O., Gudukbay, U., 2004, An efficient query optimization strategy for spatio-temporal queries in video databases, Journal of Systems and Software, 73(1), pp 113–131

Gudivada, V., Raghavan, V., 1995, Content based image retrieval systems, IEEE computer society, 28(9), pp 18-22

Grosky, W. 1997, Managing multimedia information in database systems, Communication of the ACM, 40(12), pp 72-80

Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Qian Huang D.B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D., Yanker, P., 1995, Query by image and video content: the QBIC system, IEEE Computer systems, 28(9), pp 23-32

Idris, F., Panchanathan, S., 1997, Review of image and video indexing techniques, Journal of Visual Communication and Image Representation, 8(2), pp 146

Kiranyaz, S., Caglar, K., Guldogan, E., Guldogan, O., and Gabbouj, M., 2003, MUVIS: a content-based multimedia indexing and retrieval framework, Proceedings of the seventh international symposium on signal processing and its applications, pp 1-8

Kasanen,E.L, Kari, Siitonen, Arto, 1993, The constructive approach in management accounting research, Journal of Management Accounting Research, 5, pp 243 –260

Kosch, H., 2002, MPEG-7 and multimedia database systems, ACM SIGMOD, 31(2) pp 34-39

Liu, Y., Zhang, D., Lu, G., Ma, W., 2007, A survey of content-based image retrieval with high-level semantics, Elsevier Science Inc, 40(1), pp 262-282

Lew M.S., Sebe, N., Djeraba, C., Jain, R., 2006, Content-based multimedia information retrieval: State of the art and challenges, ACM Transactions on Multimedia Computing, Communications, 2(1), pp 1-19

March, S. T., and Smith, G. F., 1995, Design and natural science research on information technology, decision support systems 15(4), pp 251-266

Mosteller, F., Colditz, G.A., 1996, Understanding Research Synthesis (Meta-Analysis), Annual Review of Public Health, 17, pp 1-23

Mu, X., 2006, Content-based video retrieval: does video's semantic visual feature matter?, Proceedings of the 29th annual international ACM, pp 679-680

Nang, J., Park, J., 2007, An efficient indexing structure for content based multimedia retrieval with relevance feedback, Proceedings of the 2007 ACM symposium on Applied computing, pp 517-524

Naphade, R. M., Kennedy, L., Kender, R. J., Chang, S., Smith, R. J. , Over, P., and Hauptmann, A. 2005, Alight scale concept ontology for multimedia understanding for TRECVID, pp 679-680

Oomoto, E. and Tanaka, K., 1993, OVID: Design and implementation of a video-object database system, IEEE transactions on knowledge and DNA engineering, 5(4), pp 629-643

Pentland, A., Picard, R.W., Sclamff, S., 1994, Photobook tools for content based manipulation of image databases, Pmc SPLE (Storage and Retrieval for Image and Video Databases II) 2185, pp 34-37

Rosenthal, R., DiMatteo, M.R., 2001, Meta-analysis: Recent Developments in Quantitative Methods for Literature, Annual Review of Psychology, 52, pp 59-82

Shahabi, C., Safar, M., 2007, An experimental study of alternative shape-based image retrieval techniques, Springer Netherlands, 23(1), pp 29-48

Smeaton, A. F., Over. P., 2003, TRECVID: Benchmarking the effectiveness of information retrieval tasks on digital video. In proceedings of the international conference on image and video, 2748, pp 451-456

Smith, J.R.,Chang, S.F., 1997, VisualSEEK a fully automated content-based image query system, ACM Multimedia, Boston, pp 87-98

Smoliar, S.W., Zhang H., 1994, Content based video indexing and retrieval, IEEE Multimedia, 1(2), pp 62-72

Wu, J.K., 1997, Content-based indexing of multimedia databases, IEEE Transactions on Knowledge and Data Engineering, 9(6), pp 978-989

Walid G. A., Ann C., Jianping F., Ahmed K., Moustafa A. Hammad, Ihab F. Ilyas, Mirette S. Marzouk, Xingquan Z, A video database management system for advancing video database research, Medical information systems.