

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

DIMUSE: AN INTEGRATED FRAMEWORK FOR DISTRIBUTED MULTIMEDIA
SYSTEM WITH DATABASE MANAGEMENT AND SECURITY SUPPORT

A dissertation submitted in partial fulfillment of the

requirements for the degree of

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE

by

Na Zhao

2007

To: Interim Dean Amir Mirmiran
College of Engineering and Computing

This dissertation, written by Na Zhao, and entitled DIMUSE: An Integrated Framework for Distributed Multimedia System with Database Management and Security Support, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this dissertation and recommend that it be approved.

Xudong He

Nagarajan Prabakar

Keqi Zhang

Mei-Ling Shyu

Shu-Ching Chen, Major Professor

Date of Defense: July 19, 2007

The dissertation of Na Zhao is approved.

Interim Dean Amir Mirmiran
College of Engineering and Computing

Dean George Walker
University Graduate School

Florida International University, 2007

ACKNOWLEDGMENTS

I would like to extend my sincere gratitude and appreciation to my Ph.D. advisor, Professor Shu-Ching Chen, for his guidance, support, suggestions and encouragement while this dissertation was being conducted. I am also indebted to Professors Xudong He, Nagarajan Prabakar of the School of Computing and Information Sciences, Professor Keqi Zhang of Department of Environmental Studies and International Hurricane Research Center, and Professor Mei-Ling Shyu of the Department of Electrical and Computer Engineering, University of Miami, for accepting the appointment to the dissertation committee, as well as for their suggestions and support.

The financial assistance I received from the School of Computing and Information Sciences and the Dissertation Year Fellowship from University Graduate School is gratefully acknowledged.

I would also like to thank all my friends and colleagues whom I have met and known while attending Florida International University. In particular, I would like to thank Chengcui Zhang, Min Chen, Kasturi Chatterjee, Khalid Saleem, Fausto Fleites, Michael Armella, Hsin-Yu Ha and other members of the Distributed Multimedia Information System Laboratory for their generous help. My special thanks go to Mr. Frank Oreovicz in Purdue University for his help with English writing and presentation checking.

Finally, my utmost gratitude goes to my husband Cheng Xu, my father Guoqing Zhao, my mother Jiyun Cang, and brother Kang Zhao, for their love, support and encouragement, which made this work possible.

ABSTRACT OF THE DISSERTATION

DIMUSE: AN INTEGRATED FRAMEWORK FOR DISTRIBUTED MULTIMEDIA SYSTEM WITH DATABASE MANAGEMENT AND SECURITY SUPPORT

by

Na Zhao

Florida International University, 2007

Miami, Florida

Professor Shu-Ching Chen, Major Professor

With the recent explosion in the complexity and amount of digital multimedia data, there has been a huge impact on the operations of various organizations in distinct areas, such as government services, education, medical care, business, entertainment, etc. To satisfy the growing demand of multimedia data management systems, an integrated framework called DIMUSE is proposed and deployed for distributed multimedia applications to offer a full scope of multimedia related tools and provide appealing experiences for the users.

This research mainly focuses on video database modeling and retrieval by addressing a set of core challenges. First, a comprehensive multimedia database modeling mechanism called Hierarchical Markov Model Mediator (HMMM) is proposed to model high dimensional media data including video objects, low-level visual/audio features, as well as historical access patterns and frequencies. The associated retrieval and ranking algorithms are designed to support not only the general queries, but also the complicated temporal event pattern queries. Second, system training and learning methodologies are incorporated such that user interests are mined efficiently to improve the retrieval performance. Third, video clustering techniques are proposed to continuously increase the searching speed and accuracy by architecting a more efficient multimedia database structure. A distributed video management and retrieval system is designed and implemented to demonstrate the overall performance. The proposed approach is further

customized for a mobile-based video retrieval system to solve the perception subjectivity issue by considering individual user's profile. Moreover, to deal with security and privacy issues and concerns in distributed multimedia applications, DIMUSE also incorporates a practical framework called SMARXO, which supports multilevel multimedia security control. SMARXO efficiently combines role-based access control (RBAC), XML and object-relational database management system (ORDBMS) to achieve the target of proficient security control.

A distributed multimedia management system named DMMManager (Distributed MultiMedia Manager) is developed with the proposed framework DIMUSE to support multimedia capturing, analysis, retrieval, authoring and presentation in one single framework.

TABLE OF CONTENTS

CHAPTER	PAGE
CHAPTER I. INTRODUCTION AND MOTIVATION.....	1
1.1 Significance and Impact of Multimedia System Research.....	3
1.2 Proposed Solutions.....	6
1.3 Contributions.....	8
1.4 Scope and Limitations of the Proposed Prototype.....	11
1.5 Outline of the Dissertation.....	12
CHAPTER II. LITERATURE REVIEW.....	14
2.1 Multimedia Data Modeling, Indexing and Data Structures.....	14
2.2 Multimedia Retrieval Methodologies.....	15
2.2.1 Keyword-based Retrieval.....	15
2.2.2 Content-based Retrieval.....	16
2.2.3 Challenges in Multimedia Retrieval.....	17
2.3 Multimedia Security Solutions.....	18
2.4 Prototype Multimedia Management Systems.....	20
2.4.1 Content-based Multimedia Retrieval Systems.....	20
2.4.2 Multimedia Presentation Authoring and Rendering Systems.....	32
CHAPTER III. OVERVIEW OF THE FRAMEWORK.....	35
3.1 Multimedia Database Modeling and Retrieval Module.....	39
3.1.1 Image Database Modeling and Retrieval using MMM.....	39
3.1.2 Video Database Modeling and Retrieval using HMMM.....	39
3.1.3 Online Learning and Offline Training via HMMM.....	40
3.1.4 Video Database Clustering.....	41
3.2 Multimedia Presentation Module.....	42
3.2.1 Presentation Design with MATN Model.....	42
3.2.2 Presentation Rendering with JMF and SMIL.....	43
3.3 Security Management Component.....	43
3.3.1 Security Policy and Role Managing.....	43
3.3.2 Security Checking.....	44
3.3.3 Multimedia Data Managing and Processing.....	44
3.4 Multimedia Application and System Integration.....	45
3.4.1 DMManager: Distributed Multimedia Manager.....	45
CHAPTER IV. MULTIMEDIA DATABASE MODELING AND RETRIEVAL.....	47
4.1 Introduction.....	47
4.2 Overall Framework.....	49
4.3 Hierarchical Markov Model Mediator (HMMM).....	53
4.4 Two-level HMMM Model.....	56
4.4.1 Video shot level MMM.....	57
4.4.2 Video-level MMM.....	61
4.4.3 Connections between first level MMMs and second level MMM.....	62
4.4.4 Initial Process for Temporal Event Pattern Retrieval.....	64
4.5 Video Database Clustering and Construction of 3rd Level MMM.....	67
4.5.1 Overall Workflow.....	67

4.5.2	Conceptual Video Clustering	69
4.5.3	Constructing the 3 rd level MMM model.....	72
4.5.4	Interactive Retrieval through Clustered Video Database.....	74
4.5.5	Experimental Results for Video Clustering	76
4.6	Conclusions	78
CHAPTER V. MULTIMEDIA SYSTEM TRAINING AND LEARNING		80
5.1	Introduction	80
5.2	Related Work.....	82
5.3	Automate Offline Training using Association Rule Mining	84
5.3.1	Overall Process.....	85
5.3.2	Automated Training using ARM.....	87
5.3.3	Experimental Results for Automated Learning Mechanism	89
5.4	Online Relevance Feedback	91
5.4.1	Anticipant Event Pattern Instance	91
5.4.2	Affinity Instances for A	92
5.4.3	Feature Instances for B.....	93
5.4.4	Updated Similarity Measurements and Query Processing.....	94
5.4.5	Experimental Results for System Learning Techniques	95
5.5	Application: A Mobile-based Video Retrieval System	99
5.5.1	Introduction	99
5.5.2	Related Work	101
5.5.3	System Architecture	104
5.5.4	MoVR: Mobile-based Video Retrieval	107
5.5.5	HMMM-based User Profile	109
5.5.6	Fuzzy Associated Retrieval.....	114
5.5.7	Implementation and Experiments.....	118
5.5.8	Summary	124
CHAPTER VI. SECURITY SOLUTIONS FOR MULTIMEDIA SYSTEMS.....		125
6.1	Introduction	125
6.2	SMARXO Architecture.....	126
6.3	Multimedia Access Control.....	127
6.3.1	Multimedia Indexing Phase.....	129
6.3.2	Security Modeling Phase.....	130
6.3.3	DBMS Management Phase	134
6.4	Security Verification	135
6.5	Conclusions	137
CHAPTER VII. MULTIMEDIA SYSTEM INTEGRATION		138
7.1	System Overview.....	138
7.2	Multimedia Data Collecting	140
7.3	Multimedia Analysis and Indexing.....	140
7.3.1	Image Analysis and Indexing.....	140
7.3.2	Video Analysis and Indexing.....	141
7.4	Multimedia Retrieval.....	142
7.4.1	Content-based Image Retrieval	142
7.4.2	Video Data Browsing and Retrieval.....	144
7.5	Multimedia Presentation Module	147

7.5.1	Multimedia Presentation Authoring	147
7.5.2	Multimedia Presentation Rendering.....	151
7.5.3	Presentation Rendering via JMF Player	152
7.5.4	Presentation Rendering via SMIL Language	153
7.6	Conclusions	154
CHAPTER VIII. CONCLUSIONS AND FUTURE WORK		155
8.1	Conclusions	155
8.2	Future Work.....	157
LIST OF REFERENCES		160
VITA.....		169

LIST OF TABLES

TABLE	PAGE
Table IV-1. HMMM is an 8-Tuple: $\Lambda = (d, \mathbf{S}, \mathbf{F}, \mathbf{A}, \mathbf{B}, \mathbf{\Pi}, \mathbf{O}, \mathbf{L})$	54
Table IV-2. 3-level HMMM model	56
Table IV-3. Feature list for the video shots	59
Table V-1. Experimental results for ARM-based feedback evaluations.....	91
Table V-2. Average accuracy for the different recommendations	122
Table VI-1. Comparison of multimedia security techniques	137
Table VII-1. Example mappings to the graphical query language.....	145
Table VII-2. MATN structures for 13 temporal relationships	149
Table VII-3. MATN design buttons & functionalities.....	150

LIST OF FIGURES

FIGURE	PAGE
Figure II-1. CIRES interface with sample images	21
Figure II-2. WebSeek interfaces (a) sample catalog (b) image retrieval results with relevance feedback	22
Figure II-3. Query interface of VDBMS.....	22
Figure II-4. User interface of the Goalgle soccer video search engine	24
Figure II-5. User interface for IBM VideoAnnEx Tool.....	25
Figure II-6. Query interface of IBM MARVEL	26
Figure II-7. User interface for CuVid	27
Figure II-8. User interface for Youtube video search	28
Figure II-9. User interface for Google video search	29
Figure II-10. User interface for Yahoo! video search.....	30
Figure II-11. User interface for AOL TRUVEO video search.....	31
Figure II-8. LAMP interface with the synchronization graph of a news-on-demand presentation	32
Figure II-9. Views layout and user interface for T-Cube.....	33
Figure II-10. Structured media authoring environment of Madeus	34
Figure III-1. Overall framework and components of DIMUSE.....	37
Figure IV-1. Overall framework of video database modeling and temporal pattern retrieval utilizing HMMM, online learning, offline training and clustering techniques	51
Figure IV-2. Three-level construction of Hierarchical Markov Model Mediator.....	55
Figure IV-3. An example result of a temporal pattern query.....	65
Figure IV-4. HMMM-based soccer video retrieval interface	66
Figure IV-5. Overall workflow for the proposed approach	68
Figure IV-6. The proposed conceptual video database clustering procedure	73
Figure IV-7. Lattice structure of the clustered video database	75

Figure IV-8. Result patterns and the traverse path.....	75
Figure IV-9. Comparison of the average execution time.....	76
Figure IV-10. Soccer video retrieval system interfaces (a) query over non-clustered soccer video database (b) query over clustered soccer video database	77
Figure V-1. Two feedback scenarios for the soccer video goal event retrieval.....	82
Figure V-2. Overall process for the automated training	86
Figure V-3. System interfaces for the Mobile-based Video Retrieval System.....	89
Figure V-4. Online learning procedure of temporal based query pattern retrieval	95
Figure V-5. User-centered soccer video retrieval and feedback interface	97
Figure V-6. Online training experimental results for Query 1.....	98
Figure V-7. Soccer video retrieval and feedback results for Query 2. (a) first round event pattern retrieval; (b) third round event pattern retrieval.....	98
Figure V-8. Mobile-based video retrieval system architecture.....	105
Figure V-9. Overall framework of mobile-based video retrieval system	109
Figure V-10. Generation of individual user’s affinity profile.....	111
Figure V-11. Fuzzy weight adjustment tool (a) generalized recommendation; (b) personalized recommendation; (c) fuzzy associated recommendation	114
Figure V-12. Mobile-based soccer video retrieval interfaces (a) initial choices (b) retrieval by event (c) retrieval by pattern	119
Figure V-13. Mobile-based soccer video retrieval results (a) video browsing results (b) video retrieval results (c) video player.....	121
Figure V-14. Experimental comparison of different recommendations	123
Figure VI-1. Example of image object-level security (a) original image (b) segmentation map (c) hiding a portion of the image	126
Figure VI-2. SMARXO architecture.....	127
Figure VI-3. Extended RBAC definitions in SMARXO	128
Figure VI-4. XML examples of multimedia hierarchy (a) example for image objects (b) example for video hierarchy.....	129

Figure VI-5. XML examples of the fundamental roles (a) example of subject roles (b) example of object roles	131
Figure VI-6. Example requirements for video scene/shot-level access control.....	132
Figure VI-7. XML examples of the optional roles (a) example of temporal roles (b) example of IP address roles.....	132
Figure VI-8. Security policies (a) formalized security policy (b) XML example on policy roles.....	134
Figure VI-9. Algorithm for security verification in SMARXO	136
Figure VII-1. The multimedia management flow of DMMManager.....	139
Figure VII-2. Multimedia presentation authoring tool.....	143
Figure VII-3. The key-frame based video retrieval interface with a shot displayed.....	144
Figure VII-4. Soccer retrieval interface with example temporal query and results	146
Figure VII-5. The user interface for MATN model design.....	151
Figure VII-6. The rendered multimedia presentation played by the JMF player.....	153
Figure VII-7. The rendered multimedia presentation played by the web browser	153

LIST OF DEFINITIONS

DEFINITION	PAGE
Definition IV-1: Markov Model Mediator (MMM) [Shyu03].....	53
Definition IV-2: Hierarchical Markov Model Mediator (HMMM).....	54
Definition IV-3: $SV(v_i, v_j)$, the similarity measure between two videos, is defined by evaluating the probabilities of finding the same event pattern Q^k from v_i and v_j in the same query for all the query patterns in QS	70
Definition IV-4: Assume CC_m and CC_n are two video clusters in the video database D . Their relationship is denoted as an entry in the affinity matrix A_3 , which can be computed by Equations (IV-22) and (IV-23). Here, SC is the function that calculates the similarity score between two video clusters.....	73
Definition V-1: An HMMM-based User Profile is defined as a 4-tuple: $\Phi = \{\tau, \hat{A}, \hat{B}, \hat{O}\}$,	110
Definition VI-1: An Object Hierarchy $OH = (O, OG, \leq_{OG})$, where O is a set of objects and $OG = O \cup G$ with G is a set of object groups. \leq is a partial order on OG called the dominance relation, and $O \subseteq OG$ is the set of minimal elements of OG with respect to the partial order. Given two elements $x, y \in OG$, $x \leq y$ iff x is a member of y	132
Definition VI-2: Given the octets named I_1, I_2, I_3, I_4 , the IP address segment expression IP can be defined as $IP = \sum_{j=1}^n x_j \cdot I_j \triangleright y_j \cdot I_d$, where $n = 4$, $0 \leq x_j \leq 2^8 - 1$, $0 \leq y_j \leq 2^8 - 1$, $x_j, y_j \in N$, $x_j + y_j \leq 2^8 - 1$ for $j = 1, \dots, 4$, $I_d \in \{I_1, I_2, I_3, I_4\}$	133
Definition VI-3: Object Entity Set: $OES(o) = \{o\} \cup \{s : s \in o\}$	135

CHAPTER I. INTRODUCTION AND MOTIVATION

With the rapid evolution of technologies and applications for consumer digital media, there has been an explosion in the complexity and amount of digital multimedia data being generated and persistently stored. This revolution is changing the way people live, work, and communicate with each other, and is impacting the operations of various organizations in distinct areas, such as government services, education, medical care, business, entertainment, etc. To solve the related problems, a large number of papers have been published recently on multimedia techniques and multimedia systems. However, the issues related to analysis, modeling, specification, and design of distributed multimedia systems and applications are still challenging both researchers and developers.

In comparison to traditional text and data, multimedia objects are typically very large and may include images, video, audio and some other visualization components. Due to the specific characteristics of the multimedia data, many subsequent research issues arise within the fields of multimedia analysis, storage, retrieval, transmission, presentation, and security protection. Generally, the following aspects should be considered when a multimedia system is designed.

First, a distributed architecture is required for the construction of a large-scale multimedia system. Multimedia data is storage consuming and may be distributed through the network and allocated at distinct computers. Accordingly, the multimedia applications should be capable of managing the distributed multimedia data in the network environment. The systems should allow multimedia data to be transmitted through the networks or other connections easily.

Second, content based retrieval is one of the major issues which should be considered in the multimedia applications. As multimedia data is rich in semantic information, intermediate processing and semantic interpretation become much more helpful, especially for handling images, audio and video data. Manual annotation of multimedia data for content based retrieval is cumbersome, error prone, and prohibitively expensive. To make it feasible, multimedia analysis

techniques are developed to automatically extract the visual/audio features and obtain the semantic understanding for the multimedia content. Thereafter, an advanced multimedia modeling framework should be constructed to combine these features, semantic annotations, along with the user perceptions for content based retrieval purposes.

Third, user feedback should be deployed to refine the retrieval performance by satisfying diverse user interests. Undoubtedly, the distinct background, situation and interest of different users inevitably call for individual views into a semantic understanding of the multimedia data and therefore produce user centered meta-data. Accordingly, multimedia retrieval, summarization, ranking composition, delivery, and presentation need to be designed to satisfy users' requirements and preferences.

Fourth, it is a challenging yet rewarding task to provide security support for a large scale multimedia management system. For some of the multimedia content generated in medical, commercial and military fields, it may only be partially exposed to the general public or should not be accessible at all. Hence, it is critical to develop a user-adaptive framework for the data access control to provide enhanced security support in multimedia database design and multimedia system development.

In recent years, emerging ubiquitous multimedia applications have been developed to fulfill various kinds of demands for multimedia analysis, retrieval, and usage. However, most of these systems can only provide one or few functionalities for multimedia data management. For example, some systems are concerned with the production of multimedia material; some systems handle mainly multimedia analysis, annotation and retrieval issues, while some others only provide the functionalities for multimedia presentation design. What is lacking is an integrated framework for the construction of a comprehensive distributed multimedia system, which can support a full scope of functionalities.

In this research, an integrated framework called DIMUSE (DIstributed Multimedia SystEm) is proposed for distributed multimedia applications including multimedia data capturing, analysis, database modeling, content-based retrieval, presentation authoring and rendering, etc. In addition to a complete set of multimedia searching and editing tools, another attractive aspect of DIMUSE is that user interactions and perceptions are fully considered in the system learning and training process for providing innovative multimedia experience to the users.

The remainder of this chapter is organized as follows. The next section discusses the detailed significance and impact of multimedia system research to develop an integrated framework for the database modeling, information retrieval, and security support in the distributed multimedia system. In Section 1.2, the proposed solutions are introduced for constructing such an integrated framework. Section 1.3 presents the main contributions of this research. The scope and limitations of this proposed framework are further explored in Section 1.4. Finally, Section 1.5 summarizes the outline of this dissertation.

1.1 Significance and Impact of Multimedia System Research

The popularity of digital media is growing fast in all aspects of the market including traditional broadcasting, new media enterprise, and World Wide Web. It results from the convergence of many factors, including the general affordability of multimedia capturing, management and distribution devices, and the pervasive increase in network bandwidth. The digital media is expected to play a critical role in enhancing the value of traditional computer applications. Correspondingly, there is a growing demand for efficient technologies for retrieving semantic information and extracting knowledge from multimedia content.

The methods for describing and retrieving semantic information from multimedia content can inevitably enable or enhance applications and services both for commercial users and end users. Such kinds of applications include, but are not limited to the following areas: automatic and semi-automatic annotation, multimedia data indexing, content based image/video retrieval,

collaborative filtering, digital media sharing, personalized adaptation, and multimedia presentation delivery.

One ultimate goal of multimedia system research is to offer appealing multimedia experiences to users considering their own preferences and information needs. In multimedia system research, users' interests and perceptions not only have to be taken into account, but in fact must become the essential concern at all parts of multimedia system design. This process involves far more than merely physical storage or technical analysis of multimedia data. It should also consider the user abilities, device capabilities, network characteristics, etc. Moreover, the interactive multimedia user interfaces must be developed to address the content based retrieval facilities, authoring and display environment, as well as the management functions for user access control.

The major research issues in this dissertation can be outlined as follows:

- (1) Hierarchical multimedia data modeling issue. In the development cycle of a multimedia management system, one of the most crucial issues is how to proficiently model, accumulate, and manage the multimedia data, along with their metadata, features, and other related information. In addition to the source data, it should also be able to proficiently model the multimedia objects in a hierarchical way considering their temporal and/or spatial relationships, such as video shots, video key frames, and segmented image objects.
- (2) Semantic concept mining, storage, and retrieval issues. An efficient content-based retrieval operation is necessary to offer querying functionality to a multimedia database management system (MMDBMS). In content-based multimedia retrieval, the “semantic gap” denotes the gap between the rich meaning and interpretation that the users anticipate the database systems to associate their queries for searching and browsing multimedia data. This issue needs to be addressed to make multimedia information pervasively

accessible and reusable upon the original concepts and meanings represented by the digital media data. The critical difficulty here is how to efficiently derive and facilitate semantic annotations which require knowledge and techniques from assorted disciplines and domains, even though many of them are outside of the traditional computer science fields.

- (3) User-centered learning issue. As different users may eventually have diverse interests, users' perceptions need to be taken into account when modeling the underlying database. Two kinds of methods can be considered to improve the retrieval performance. One is to trigger the online learning algorithm which can handle interactions of a single user, which may pose restrained performance due to the limited size of positive feedback. An alternative solution is to learn general user perceptions via feedback from different users. The training process is initiated only when the number of feedbacks reaches a certain threshold. This can improve the overall training performance but it becomes a manual process to decide the threshold and initiate the training process. This issue should be further investigated to discover the best scheme on system learning process.
- (4) Security management issue. There is a growing concern about security and privacy of the distributed multimedia contents over the Internet or local networks. It may involve multiple levels of access control requirements when accessing multimedia contents in a distributed environment. Moreover, composing multimedia documents brings together multimedia objects that exist in various formats. The security requirement varies for different types of multimedia objects. Hence, a security model is desired for a distributed multimedia management system that allows creation, storage, indexing and presentation for the multi-level secured multimedia contents.
- (5) System integration issue. In addition to the querying capability, the development of an abstract semantic model is also essential for an integrated robust MMDBMS. The model

should be powerful enough to support multimedia presentation synchronization and utilize optimal programming data structures for the implementation. A proficient semantic model is anticipated to model not only multimedia presentations, but also the temporal and/or spatial relations of different media streams. In addition, this semantic model should be able to help the integration of different components with various functionalities for the purpose of developing an advanced multimedia system. For instance, this model can be deployed and integrated in this system to generate multimedia presentations by synchronizing user preferred multimedia data which are retrieved from various multimedia browsing or retrieval modules.

1.2 Proposed Solutions

In response to the above-mentioned research problems, a set of models, methodologies and techniques are proposed, implemented and applied to fulfill the requirements of the distributed multimedia management system. In the current system implementation, we employ a multi-threaded client/server architecture that can run on Windows, Unix or Linux platforms. The system is developed by using C++, Java, and an object-relational database called PostgreSQL [PostgreSQL]. In this distributed multimedia management system (DMMS), a database engine is implemented to support image feature extraction, video shot segmentation, content-based image and video queries, data management, file delivery, and multimedia presentations supports, etc. The client application utilizes a variety of user interfaces, which allow for the browsing, retrieval and composition of the media contents from various domains (e.g., sports, hurricane, medical, etc.) from its respective data stored within the database. Particularly, the following techniques are proposed and developed to address the great challenges aforementioned.

(1) Video database modeling and retrieval via HMMM

To efficiently manage a large multimedia archive, a promising solution should incorporate high-level semantic descriptions for multimedia content processing, management, and

retrieval. In this research, the Hierarchical Markov Model Mediator (HMMM) mechanism is proposed to efficiently store, organize, and manage low-level features, multimedia objects, and semantic events along with high-level user perceptions, such as user preferences, in the multimedia database management system (MMDBMS). In order to archive all valuable data, HMMM also adopts multi-disciplinary techniques, such as content-based analysis, audio feature extraction, video shot detection and segmentation algorithms, machine learning methodologies, and relevance feedback techniques. Basically, HMMM can help to bridge the semantic gap between concept-based and the content-based retrieval approaches to the underlying multimedia database model. By employing the proposed HMMM mechanism, high-dimensional multimedia data can be efficiently organized, indexed and managed. Moreover, the temporal relationships between the video shots are naturally integrated in HMMM such that the proposed mechanism can offer the capability to execute not only the traditional event queries but also the complicated temporal pattern retrieval towards the large scale video database quickly and accurately. Moreover, an advanced video clustering technique is proposed and implemented to cluster the videos based on not only the low level features, but also the high level semantic meanings and the user perceptions.

(2) User interaction support by offline training and online learning mechanisms

In this research, online learning and offline training strategies are designed and incorporated in the HMMM mechanism such that high-level user perceptions and preferences as well as the low-level visual/audio features can be considered. Further research is conducted to combine these two techniques to gain the best tradeoff in both performance and speed. In particular, a user adaptive video retrieval framework called MoVR [Zhao07a] is proposed for efficient multimedia data searching and management in the mobile wireless environment. In this framework, individual user profiles are designed to learn personal interests, while general user access history is also recorded to accumulate the common knowledge and preferences. Fuzzy

association concept is adopted here such that users can choose the best combination between general user perceptions and individual user interests to find their anticipated results easily.

(3) Security management via SMARXO

A framework called SMARXO [ChenSC04b] is proposed to perform multilevel multimedia security for multimedia applications. This technique efficiently combines Role Based Access Control (RBAC), XML [XML] and Object-Relational Database Management System (ORDBMS) to achieve the target of proficient security control. By using this framework, the system can check and follow the security policies by evaluating the user's subject role, object role, temporal role, and spatial role. Hence, inappropriate accesses are strictly prohibited and the sensitive multimedia data are efficiently protected. With the designed security management interfaces, administrators are capable of creating, deleting, and modifying all kinds of access control roles and rules. Meanwhile, security information retrieval becomes very convenient because all the protection related information is managed by XML.

(4) System development and integration in DMMManager

In this research, a distributed multimedia management system named DMMManager (Distributed MultiMedia Manager) is developed with the proposed framework DIMUSE to support multimedia capturing, analysis, retrieval, authoring and presentation in one single framework. The distributed client/server architecture is adopted in DMMManager such that multiple requests from different clients can be handled simultaneously. A set of core components are efficiently integrated in DMMManager so that the user can complete various tasks including video/audio capturing, content-based image/video retrieval, multimedia presentation design and rendering, and security management.

1.3 Contributions

In this dissertation, an integrated framework called DIMUSE is proposed for multimedia application design. In DIMUSE, a variety of advanced techniques are proposed, implemented and

integrated to develop a large scale distributed multimedia system with both powerful database management capabilities and enhanced security protection.

The major contributions of this research can be outlined as follows:

- The proposed Hierarchical Markov Model Mediator (HMMM) mechanism offers a hierarchical structure to assist in the proficient construction of the multimedia database. With the power of HMMM, the proposed video database modeling mechanism, we can narrow down the semantic gap between the content/concept based retrieval approaches with the comprehensive multimedia database modeling. First, HMMM naturally incorporates the temporal relationship between semantic events such that complicated temporal pattern queries can be executed. Second, HMMM helps to retrieve more accurate patterns quickly with lower computational costs. Third, the multimedia retrieval approaches associated with HMMM integrate the feedback and learning strategies by considering not only the low-level visual/audio features, but also the high-level semantic information and user preferences.
- In this research, the proposed interactive video retrieval framework incorporates a conceptual video clustering strategy. The proposed framework can reuse the cumulated user feedback to perform the video clustering process, such that the overall system can not only learn the user perceptions, but also get more efficient multimedia database structure via adopting video clustering technique. As the HMMM mechanism helps to traverse the most optimized path to perform the retrieval, the proposed framework can only search several clusters for the candidate results without traversing all the paths to check the whole database. That is, the proposed video clustering technique can be conducted to further reduce the searching time especially when dealing with the top- k similarity retrievals. Meanwhile, the clustering technique helps to further improve the database structure by adding a new level to model the video clusters.

- DIMUSE employs a strategy to accommodate advanced queries by considering the high level semantic meaning. First, it is capable of searching semantic events or event patterns considering their popularity by evaluating their access frequencies in a large number of historical queries. Second, users can choose one or more example patterns with their anticipated features from the initial retrieved results, and then issue the next round of queries. It can search and re-rank the candidate patterns which involve similar aspects with positive examples reflecting the user's interests. Third, it is worth mentioning that this approach supports both online learning and offline training such that the system can efficiently learn the individual user preferences in real time, while continuously improving the overall performance to gain the long term benefits. In this research, the offline training mechanism is further improved and automated by adopting the association rule mining technique. That is, the training process can be automatically invoked for certain videos by evaluating the historical queries and feedbacks.
- In order to accommodate various constraints of the mobile devices, a set of advanced techniques are developed and deployed to address essential issues in the proposed mobile-based video retrieval system. First, HMMM-based user profiles are created to integrate seamlessly with a novel learning mechanism. It can enable the "personalized recommendation" for an individual user by evaluating his/her personal histories and feedbacks. Second, the fuzzy association concept is employed in the retrieval process such that the users gain control of the preference selections to achieve reasonable tradeoff between the retrieval performance and processing speed. Third, virtual clients are designed to perform as a middleware between server applications and mobile clients. This design helps to reduce the storage load of mobile devices, and to provide greater accessibility with their cached media files.

- Several significant access control techniques are incorporated in SMARXO to satisfy the complicated multimedia security requirements. First, SMARXO incorporates efficient multimedia analysis mechanisms to acquire meaningful visual/audio objects or segments. Second, XML and object-relational databases are adopted such that proficient multimedia content indexing can be easily achieved. Third, a dominant access control model is upgraded and embedded to suit the specific characteristics of multimedia data. Moreover, XML is also applied to organize all kinds of security related roles and policies. Finally, and most importantly, all of these techniques are efficiently organized such that multi-level multimedia access control can be achieved in SMARXO easily.

1.4 Scope and Limitations of the Proposed Prototype

The proposed prototype has the following assumptions and limitations:

- (1) The current design and deployment of the HMMM model partly relies on the precision and correctness of automatic pre-filtering and semantic event annotation algorithms. It assumes that we can get reasonably good event annotation results automatically as the inputs. However, in a real scenario, most of the event annotation and learning algorithms are constructed by considering the domain knowledge, while the precision of a particular algorithm cannot be guaranteed for all the video samples. The annotation accuracy could actually be affected by all kinds of noisy data. In other words, semantic information extraction is still an open research issue and a very challenging task.
- (2) For the content based video retrieval (CBVR) approach, the user feedback and system training techniques are proposed and implemented. However, the performance of the offline training algorithm is dependent on the number of historical queries. The assumption is made such that enough feedback is provided from multiple users. Moreover, the coverage of user feedback also counts. The greater the number of the media files involved in the historical feedback, the better the performance of the system

after the training process. If the selected query sample has not been touched in any historical feedback, the system retrieval performance can hardly be improved for this specific query. On the contrary, a small number of historical records can be used in online learning but they cannot dramatically improve the retrieval performance for the whole database.

- (3) The proposed security management framework is suitable for large scale companies and organizations because it can support multiple levels of security assurance while considering all kinds of possible security roles and complicated access control rules. However, it could be somewhat complicated to learn and operate. There is still lacking a general semantic model to describe and formalize the security access control processes.
- (4) Although the present system supports distributed architecture with multiple servers and multiple clients, the current research mainly focuses on one central database. The existing database in DMManager contains a huge amount of multimedia data (around 15G), and is continuously expanding. Therefore, more efforts should be made to manage the “distributed” multimedia database. Future researches are expected to manage several separated databases which are allocated at distributed servers through the network. These databases can be connected, indexed and linked together by adopting the concepts of HMMM model.

1.5 Outline of the Dissertation

This dissertation is organized as follows.

Chapter II gives a literature review of the approaches of multimedia modeling and indexing strategies, retrieval methodologies, system training methodologies, as well as the security solutions for a multimedia system. A set of prototype systems and applications are also reviewed.

In Chapter III, the overall framework of the proposed approach is described for the distributed multimedia management systems. A series of modules are presented in detail to further advance the understanding of the proposed prototype.

Chapter IV mainly presents advanced solutions for the multimedia database modeling and content based retrieval. The Hierarchical Markov Model Mediator (HMMM) model is introduced and formalized. Related issues are further discussed, for example: construction of a 2-level HMMM model, video clustering method and construction of the 3rd level HMMM, etc.

Chapter V focuses on the user interactions through multimedia system offline training and online learning techniques. First, an innovative method is proposed to automate the offline system training by using the association rule mining method. Second, the online learning scheme is designed based on the HMMM database model to improve retrieval performance in real time. Third, a user adaptive video retrieval system, called MoVR is proposed and developed in a mobile wireless environment.

Chapter VI presents a security model called SMARXO for distributed multimedia applications via utilizing RBAC, XML and Object-Relational database. The different phases in security control management are described in detail.

The system integration issues are covered in Chapter VII by presenting the distributed multimedia management system – DMMManager. The system modules are described and the system interfaces are demonstrated to show the functionalities.

Finally, the conclusions and future work are summarized in Chapter VIII.

CHAPTER II. LITERATURE REVIEW

Although the recent development in multimedia analysis and distribution techniques has made digital media more accessible than ever before, it still lacks a comprehensive and effective solution for database modeling and retrieval. In this chapter, the existing approaches and methodologies in multimedia system research are summarized. The detailed discussions focus on the areas of multimedia (especially video) database modeling, indexing, content-based multimedia retrieval, and security management, etc.

2.1 Multimedia Data Modeling, Indexing and Data Structures

It is much more difficult to index and search multimedia databases, in which information is implicitly represented by pixel colors, motion vectors, and audio samples, than the traditional text documents. Therefore, the most common approach adopted in multimedia data management is to first extract the media content representations and then to apply data indexing or clustering techniques on them for fast media retrieval.

There is a rapid proliferation of visual processing and analysis techniques to extract the media content representations for media management, indexing and retrieval. Take video database as an example, some researchers have conducted experimental studies to identify the salient objects and their motions. For example, [ChenSC03c] presents a learning based algorithm to track the vehicles and identify their spatio-temporal information in the transportation data. The extracted salient objects and their trajectories can be indexed for video retrieval.

For the purpose of high-dimensional multimedia data indexing and modeling, many techniques have been proposed. In [DeMenthon03], the pixel regions are represented by high-dimensional points, and these points are assigned labels and stored into a single binary tree for k-nearest neighbor retrieval. The authors in [Fan01] proposed a multilevel video modeling and indexing approach, called MultiView, which consists of a set of separate indices for the clusters,

where each cluster is connected to a single root node. [Fan04] describes their later work, named ClassView, where the database indexing structure includes a set of hash tables for different visual concept levels, and a root hash table containing the information about all semantic clusters. Another new layered approach for multimedia content representation and storage for search or retrieval was introduced in [Huang00].

Currently, there also exist approaches focusing on the clustering techniques for the video data management. For example, a hierarchical clustering method for sports video was presented in [Ngo01]. Two levels of clusters are constructed where the top level is clustered by the color feature and the bottom level is clustered by the motion vectors. [Odobez03] presents a spectral clustering method to group video shots into scenes based on their visual similarity and temporal relationships. In [Xie03], algorithms are proposed for unsupervised discovery of the video structure by modeling the events and their stochastic structures in video sequences by using Hierarchical Hidden Markov Models (HHMM). However, most of the existing research works produce the clusters mainly on low-level and/or mid-level features, and do not consider high-level concepts or user perceptions in the clustering procedure. This gives rise to the problem of “semantic gap.”

2.2 Multimedia Retrieval Methodologies

2.2.1 Keyword-based Retrieval

The early adopted multimedia retrieval solutions were to query upon the textual data. For this purpose, traditional multimedia databases basically store the textual descriptors along with the source media data such that textual-based multimedia queries can be conveniently performed. Such texture descriptors are mainly extracted based on the use of annotations. For instance, there exist some video query approaches with the use of event annotations that are generally described as time-dependent information or values that are synchronous with the source data. These approaches either support semantic queries and some basic temporal queries, or deploy event-

based indexing via the inclusion of the event name, start time, and end time. For instance, IBM TRL's MPEG-7 authoring system [IBM_TRL] deploys event-based indexing and retrieval. Additionally, SMOOTH [Kosch01] and GOALGLE [Snoek03] support the semantic queries and some basic temporal queries for soccer event retrieval.

However, in practice, it is difficult to perform correct and comprehensive annotations automatically by utilizing machine interpretation techniques due to the inherent complexity of media content. Alternatively, manual annotations can be performed. However, this process involves some uncertainty because of the subjectivity of human perception and the limitation of keywords and information loss. As an attempt to address this issue,

[Detyniecki00] introduces a method of fuzzy annotations by embedding certainty values in the XML file. Another issue of manual annotation is that it requires tremendous manual effort, which becomes infeasible with the fast growth of media data.

2.2.2 Content-based Retrieval

Different from the traditional keyword-based search technologies, the content-based indexing and retrieval approaches automatically extract features such as color, texture, shape, etc. It provides more powerful search abilities and becomes a focus in this research area. Many approaches have been proposed in both the academia and industry, such as IBM's QBIC system [Flickner95], Virage's VIR engine [Virage], PhotoBook [Pentland94], and Fotofile [Kuchinsky99]. However, these systems focus only on content-based image retrieval (CBIR). There are also some projects that aim to offer video data solutions. In these projects, the video data is analyzed and segmented to facilitate the browsing functionality upon the video structure data, e.g., video segment, scene, shot, frame, etc. For example, in the Multimedia Analysis and Retrieval System (MARS) [Rui97], the role of a table of contents (ToC) is employed to structure a video into a set of scenes. In addition, an interactive content-based video browser is presented in [Guillemot03], which supports a hierarchical navigation of video over the Internet through

multiple levels of key frames. There also exist some systems which support queries for both images and videos, such as VisualSEEK [Smith96] and VISMAP [ChenW01].

In these content-based retrieval systems, Query by Examples (QBE) is mainly adopted as the query approach. QBE focuses on retrieval based on low-level or mid-level visual/audio features. Given an example image or video clip, the system aims to retrieve similar multimedia objects with similar features (color, shape, etc.). In [Aref02], a video-enhanced database system called VDBMS is proposed to support feature-based medical video data retrieval. [ChenL03] describes a system which applies image retrieval techniques to query videos by setting up the links between videos and images. IBM's video retrieval system MARVEL [IBM_Marvel] supports QBE in both the low-level feature space and the high-level model-vector space. In addition, the authors in [Ianeva04] present a probabilistic multimedia retrieval model, which can capture correlations in time and space to improve the precision of the QBE approach. However, QBE approaches have their own limitations because the users may not have the image/video example at hand when issuing the queries. In addition, QBE will not perform well if the query example is not taken with an appropriate angle or scale.

2.2.3 Challenges in Multimedia Retrieval

In terms of video data retrieval, the most recent researches mainly focus on semantic events retrieval. The existing event-based and object-based video retrieval applications may encounter a problem since event detection and object segmentation require manual annotations of video events, salient objects, and their boundaries. Ideally, the semantic content of the video data can be mined automatically by utilizing various machine interpretation techniques, and therefore the videos can be automatically annotated. However, based on the current technology development and general experiences, these kinds of complicated data abstractions are not feasible in practice. Instead, the computer may perform automatic or semi-automatic annotation with limited semantic interpretation.

The individual user's background calls for a different view on multimedia data. Therefore, another critical challenge is to design the multimedia system and retrieval algorithm such that individual user interests can be learned and satisfied. For this purpose, relevance feedback (RF) is utilized in Content Based Image Retrieval (CBIR) to bridge the semantic gaps and provide more accurate results based on users' responses [Rui98]. Several recent studies have incorporated this technique in video retrieval. [Amir05] presents a video retrieval system utilizing relevance feedback for multimodal formulations. [Yan03] describes a negative pseudo-relevance feedback (NPRF) which can extract information from the retrieved items that are not similar to the query items. Relevance feedback helps to refine the multimedia search results. However, the existing RF approaches do not incorporate an efficient methodology for multimedia database modeling that has the capability to model all layers of multimedia objects and consequently offer multi-modal video retrieval to satisfy individual user's interests.

2.3 Multimedia Security Solutions

In the earlier times, Mandatory Access Control (MAC) and Discretionary Access Control (DAC) were the only two known access control models available. That is, if an access control model was not MAC, it had to be a DAC and vice versa. In a computer system employing MAC, the administrator sets the system security policies which entirely determines the access right granted. Even for the user who creates the resource, he/she cannot grant less restrictive access to it than that specified by an administrator. As for DAC, the basic access control policies are defined to objects in a file system. The users are permitted to entirely determine the access granted to their created resources. That is, unauthorized users can be granted access through accident or malice of the users.

Compared with MAC and DAC, Role-Based Access Control (RBAC) is a newer and alternative security solution to restricting system access to authorized users. The fundamental feature of RBAC is to support the administration of large numbers of privileges on system

objects, and reduce the effort to define and manage complex security policies. With RBAC, roles are created for various characters based on their job functions. For a specific role, the permissions to perform certain operations are assigned to it and the member of this role can acquire these permissions to perform the particular system operations. Since the permissions are not assigned to users directly, the management of individual users becomes easier. It is simply a matter of assigning appropriate roles to the users.

Sandhu et al. [Sandhu96] summarizes and categorizes the traditional RBAC models into four families: RBAC0 – base model; RBAC1 – hierarchical model; RBAC2 – constraint model; and RBAC3 – combined model. Traditional RBAC models have many restrictions on the access control modeling. Therefore, numerous extended RBAC models have emerged to handle those unresolved security issues. By evaluating the traditional RBAC approaches, it has been found that several issues still remain open. First, temporal constraints may not be considered when setting the roles. Second, the locations of users are not restricted. Third, most security applications can only handle access control on multimedia files without taking care of multimedia contents. Fourth, it lacks a hierarchical architecture for the roles and therefore the role management will become complicated when the user number increases manifold.

Traditional RBAC models [Sandhu96] have many restrictions on access control modeling. Therefore, numerous extended RBAC models have emerged to handle those unresolved issues. In [Bertino01], the Temporal Role-Based Access Control (TRBAC) model, which brings the basic temporal dependencies, is proposed but it cannot handle several useful temporal variables including the constraints on user-role and role-permission assignments. The Generalized Temporal Role-Based Access Control (GTRBAC) model [Joshi05] is proposed later to solve this problem. Recently, this model was extended to an XML based version called X-GTRBAC [Bhatti05], which incorporates the content- and context-aware dynamic access control requirements of an enterprise. However, these models only improved the control capability on

temporal constraints. Moyer et al. propose the Generalized Role-Based Access Control (GRBAC) model which leverages the traditional RBAC by incorporating subject roles, object roles, and environment roles [Moyer01]. But they only introduce the temporal constraints in the environment roles, and it can only handle access control on multimedia files without taking care of multimedia contents. Another Generalized Object-Composition Petri-Net Model (GOCPN) is proposed in [Joshi02], which mainly focuses on the modeling of documents to allow secure accesses to a multimedia database management system. GOCPN utilizes a mandatory access control (MAC) approach which cannot fully perform complicated roles, role hierarchies, temporal constraints, and IP address restrictions. The comparison between our proposed security model with the aforementioned research works is discussed in Chapter VI.

2.4 Prototype Multimedia Management Systems

2.4.1 Content-based Multimedia Retrieval Systems

2.4.1.1 CIRES: Content Based Image REtrieval System

By evaluating a combination of higher level and lower level visual clues, CIRES [Iqbal02] is developed to support content-based image retrieval, including the queries ranging from scenes of purely natural objects such as sea, sky, trees, etc., to images containing conspicuous structural objects such as buildings, towers, bridges, etc. Figure II-1 shows the interface in CIRES system. In the lower level analysis, a channel energy model is utilized to represent the image texture, while the color histogram techniques are also employed. In order to describe the structural content of an image, the perceptual organization is considered in the higher level analysis to extract fractional energies in various spatial-frequency channels.

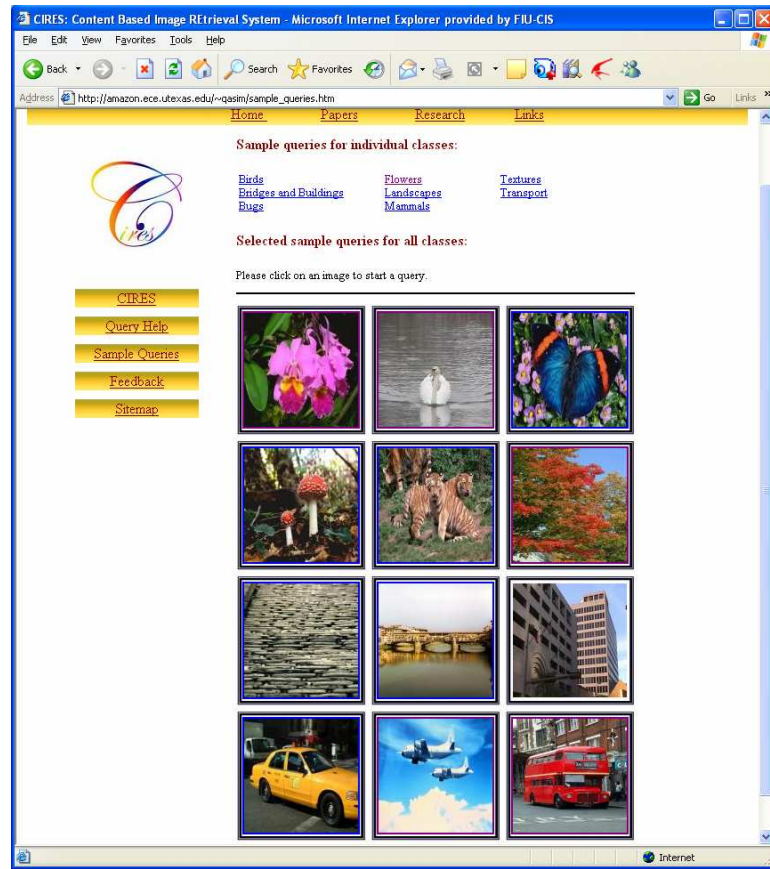
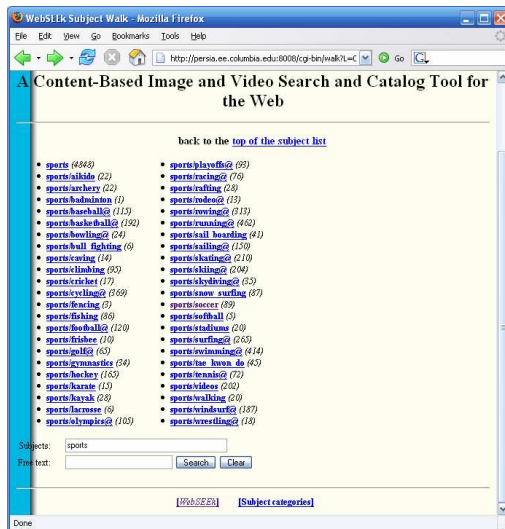


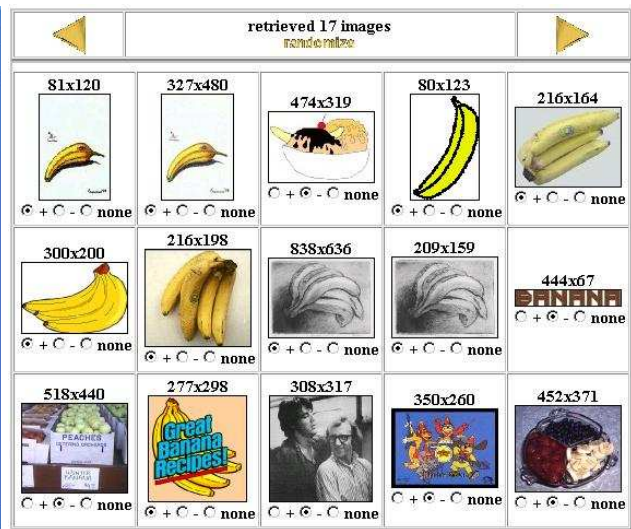
Figure II-1. CIRES interface with sample images

2.4.1.2 WebSEEK: Content Based Image and Video Catalog and Search Tool for the Web

WebSEEK [WebSEEK] is a web-based image/video catalog and search tool developed by Columbia University. This system combines text-based and color based queries through a catalog of images and videos collected from the Web. The user can initiate a query by choosing a subject from the available catalogue or entering a topic (as shown in Figure II-2(a)). There are several selections available such that query results may be used for a color query in the whole catalogue or for sorting the result list by decreasing color similarity to the selected item. Users can also possibly modify an image/video color histogram manually before reiterating the search. Furthermore, this system also adopts the relevance feedback technique for finer grain refinement of query results (as shown in Figure II-2(b)).



(a)



(b)

Figure II-2. WebSeek interfaces (a) sample catalog (b) image retrieval results with relevance feedback

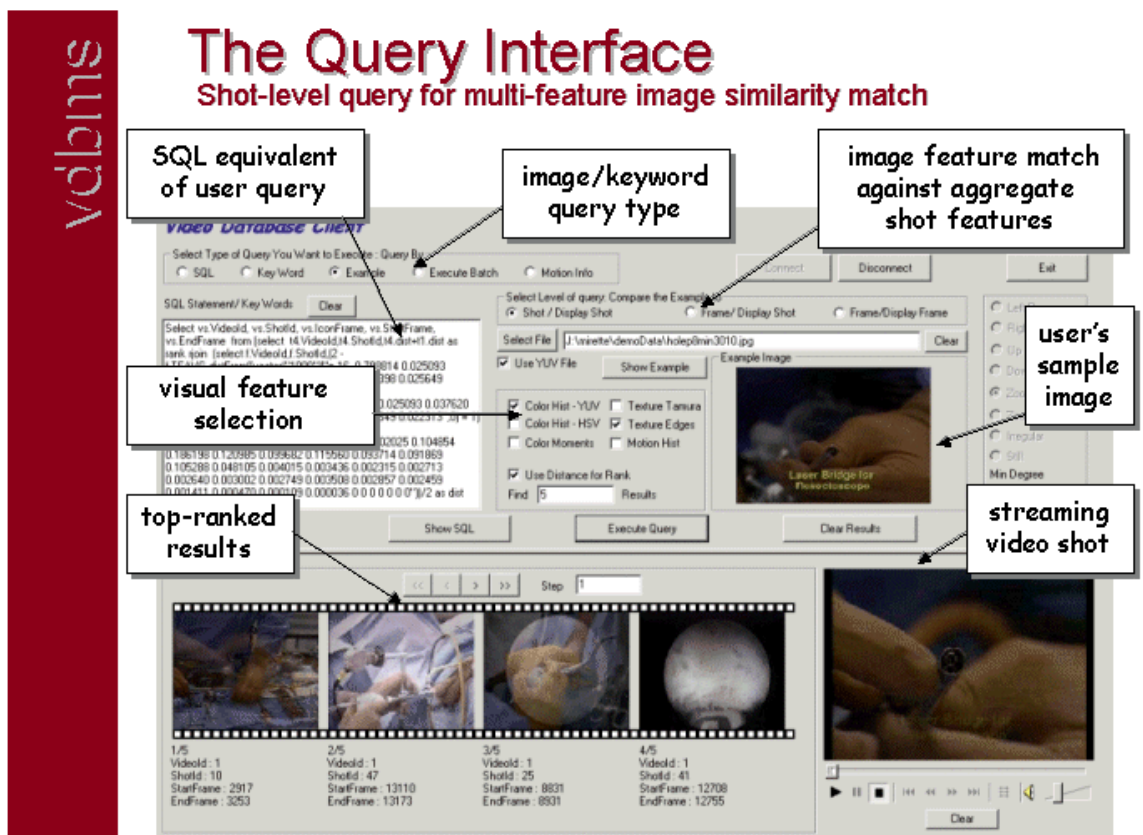


Figure II-3. Query interface of VDBMS

2.4.1.3 VDBMS: A Medical Video Retrieval System

In the VDBMS (Video Data Based Management System) Project [Aref02][Aref03] developed at Purdue University, a video-enhanced database system is proposed to support a series of functionalities for video database management, including video content preprocessing, representation and indexing, video and meta-data storage, feature-based video retrieval, buffer management, and continuous video streaming. As a multi-discipline video retrieval system, VDBMS supports both search-by-content and search-by-streaming. Furthermore, VDBMS has also been developed as a research platform via incorporating new techniques. For instance, two query operators are implemented: rank-join and stop-after algorithms.

In addition, VDBMS employs a method to define and process video streams through the query execution engine such that the continuous queries are supported to realize the requests as fast-forward, left outer join, and region-based blurring. Here the window-join algorithm works as the core operator for continuous query processing. The query interface of VDBMS is illustrated in Figure II-3.

2.4.1.4 Goalgle: Soccer Video Search Engine

Goalgle [Snoek03] is a prototype search engine for soccer video. As illustrated in Figure II-4, browsing and retrieval functionalities are provided by means of a web based interface. Goalgle allows users to retrieve video segments from a collection of prerecorded and analyzed soccer matches by selecting the specific players, events, matches, and/or text. In [Snoek05], the author expands their research to a time interval multimedia event (TIME) framework for semantic event classification in multimodal video contents. Three machine learning techniques are studied and compared: C4.5 decision tree, maximum entropy, and support vector machine.

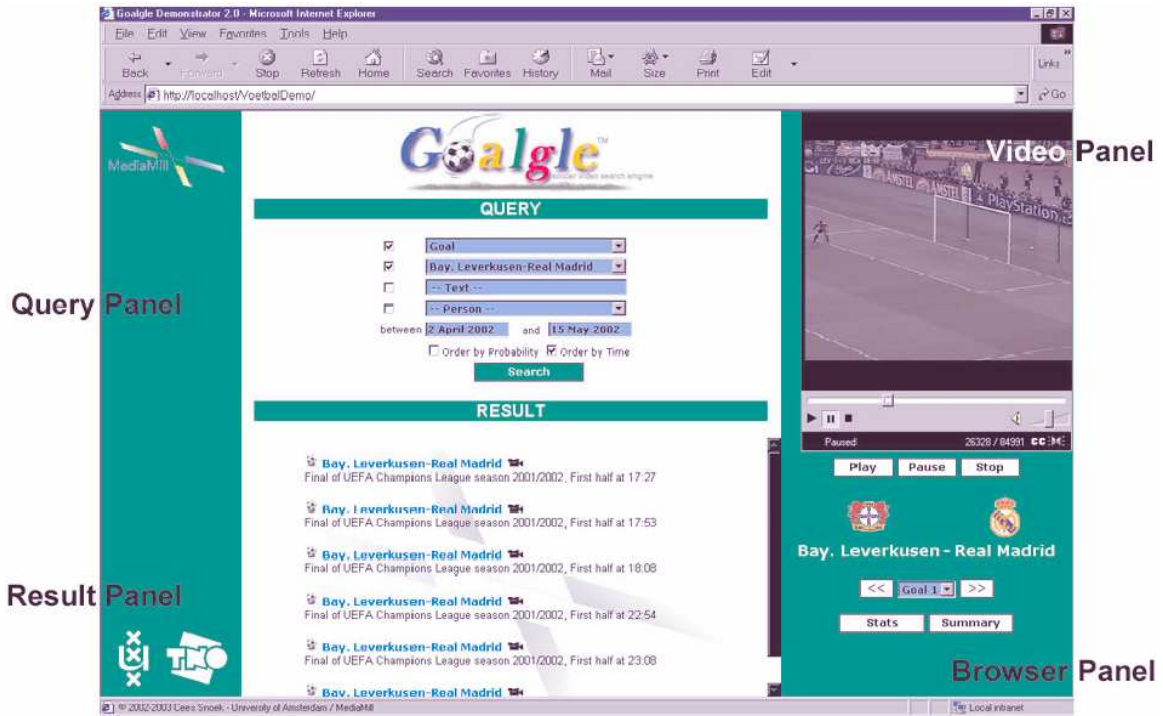


Figure II-4. User interface of the Goalge soccer video search engine

2.4.1.5 IBM VideoAnnEx: Video Annotation Tool

Figure II-5 shows the IBM VideoAnnEx annotation tool [IBM_VideoAnnEx], which is developed to assist authors in the task of annotating video sequences with MPEG-7 metadata. Each shot in the video sequence can be annotated with static scene descriptions, key object descriptions, event descriptions, and other lexicon sets. The annotated descriptions are associated with each video shot and are stored as MPEG-7 descriptions in an output XML file. VideoAnnEx can also open MPEG-7 files in order to display the annotations for the corresponding video sequence. The annotation tool also allows customized lexicons to be created, saved, downloaded, and updated.

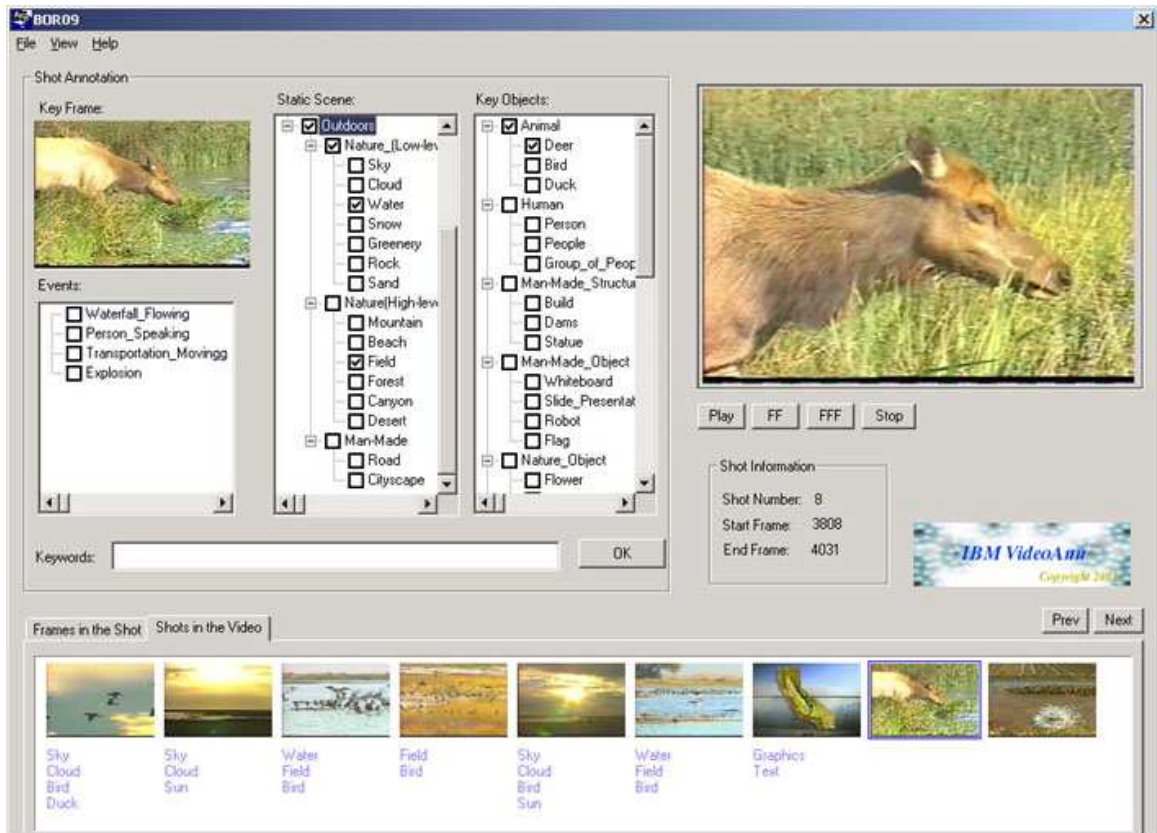


Figure II-5. User interface for IBM VideoAnnEx Tool

2.4.1.6 IBM MARVEL

The IBM video retrieval system MARVEL [IBM_Marvel] is developed to organize the growing amounts of online multimedia data by using machine learning techniques to automatically label the multimedia contents. It supports query by example in low level spaces as well as high level model-vector space. In this research work, the time-consuming and error-prone processes of metadata labeling are replaced with a semantic-based machine learning approach. It is claimed that only 1-5% of the content is required to be manually annotated as the training examples. Multimodal features are employed for automatic annotating, for example visual clues, sounds, speech transcripts, etc. The MARVEL multimedia analysis engine and the MARVEL multimedia search engine are implemented to provide the internal supports. Figure II-6 shows the online system interfaces of IBM MARVAL.

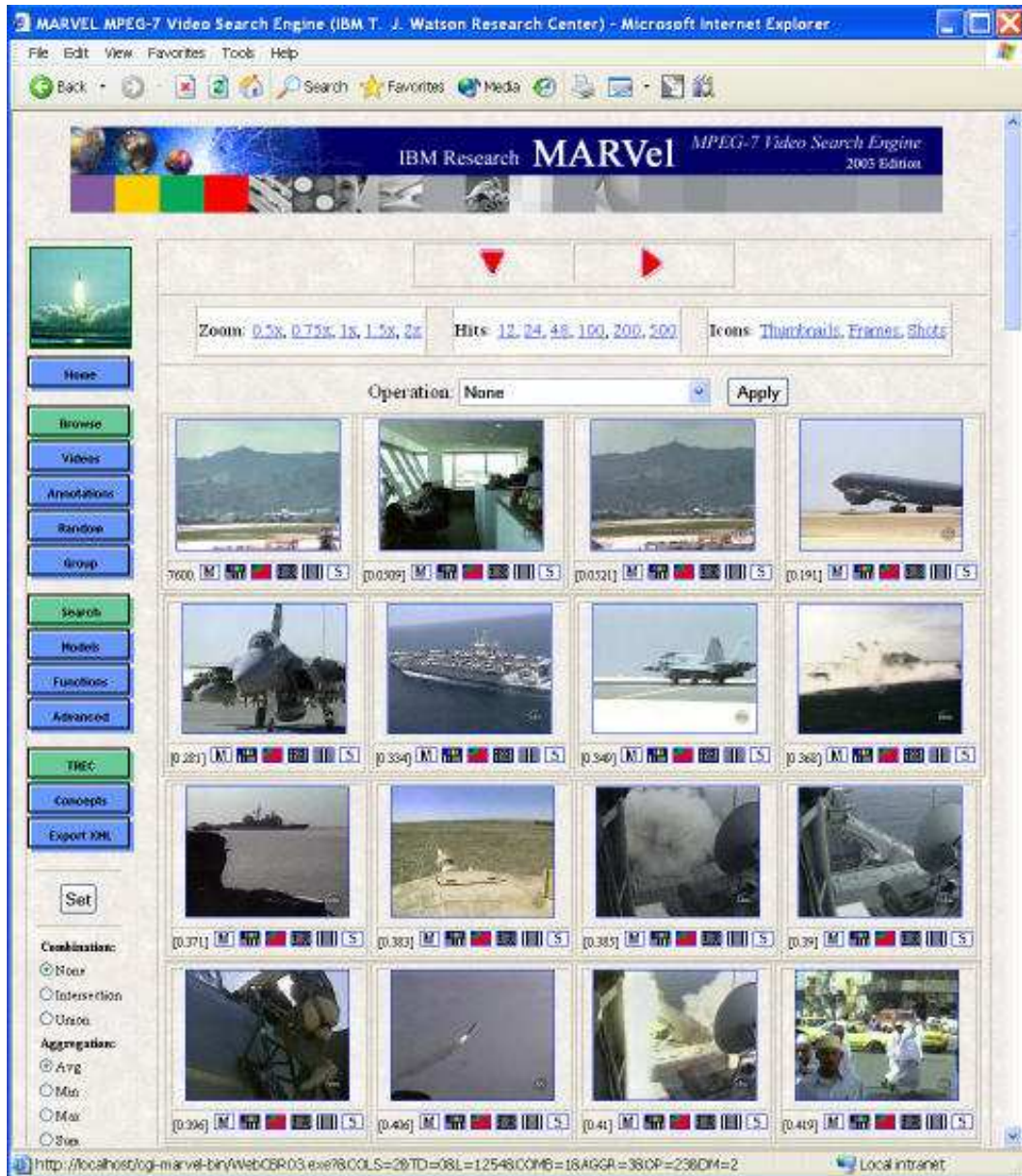


Figure II-6. Query interface of IBM MARVEL

2.4.1.7 CuVid: News Video Search System

The Columbia DVMM lab created the CuVid [CuVid] system for the 2005 TRECVID interactive search evaluation. It integrates a search engine for broadcast news video by employing the advanced techniques such as video story segmentation, semantic concept detection, duplicate detection, multimodal retrieval, and interactive browsing interfaces. Specifically, the story

segmentation algorithm considers the information bottleneck principle and the fusion of visual features and prosody features extracted from the speech. Moreover, a parts-based approach is utilized to detect the duplicate scenes across various news sources. The online retrieval interface of CuVid is shown in Figure II-7.

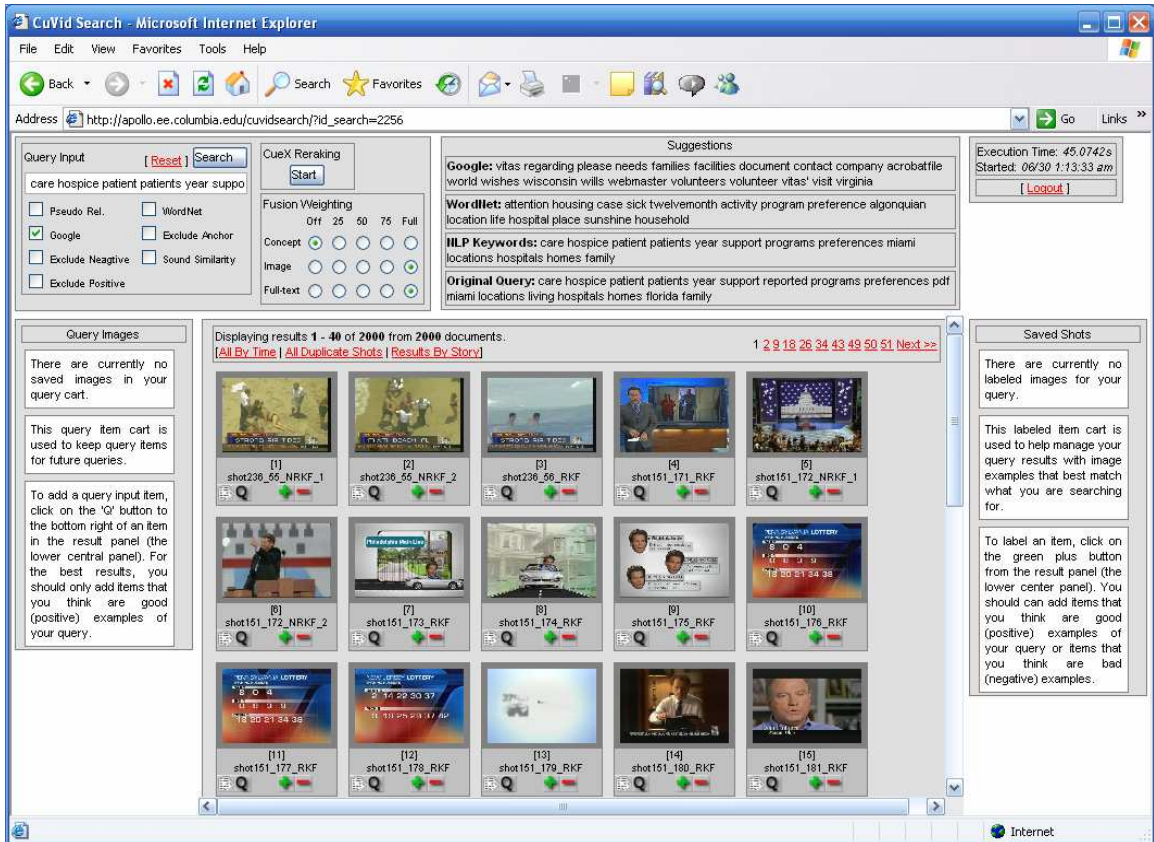


Figure II-7. User interface for CuVid

2.4.1.8 Youtube

Youtube is a video sharing website created in early 2005 where consumers can upload, view and share video clips [Youtube]. It uses Adobe Flash technology to display a wide variety of video content, including movie clips, TV clips, and music videos, as well as amateur content such as short videos which are created and edited by users. Unregistered users can watch most videos on the site, while registered users are permitted to upload an unlimited number of videos. In YouTube's second year, functions were added to enhance user ability to post video comments

and subscribe to content ratings. Each uploaded video features a series of tags that are user inputted and these tags are indexed for keyword-based searches. As demonstrated in Figure II-8, the results can be sorted by their posting time, viewing counts, or rating scores.

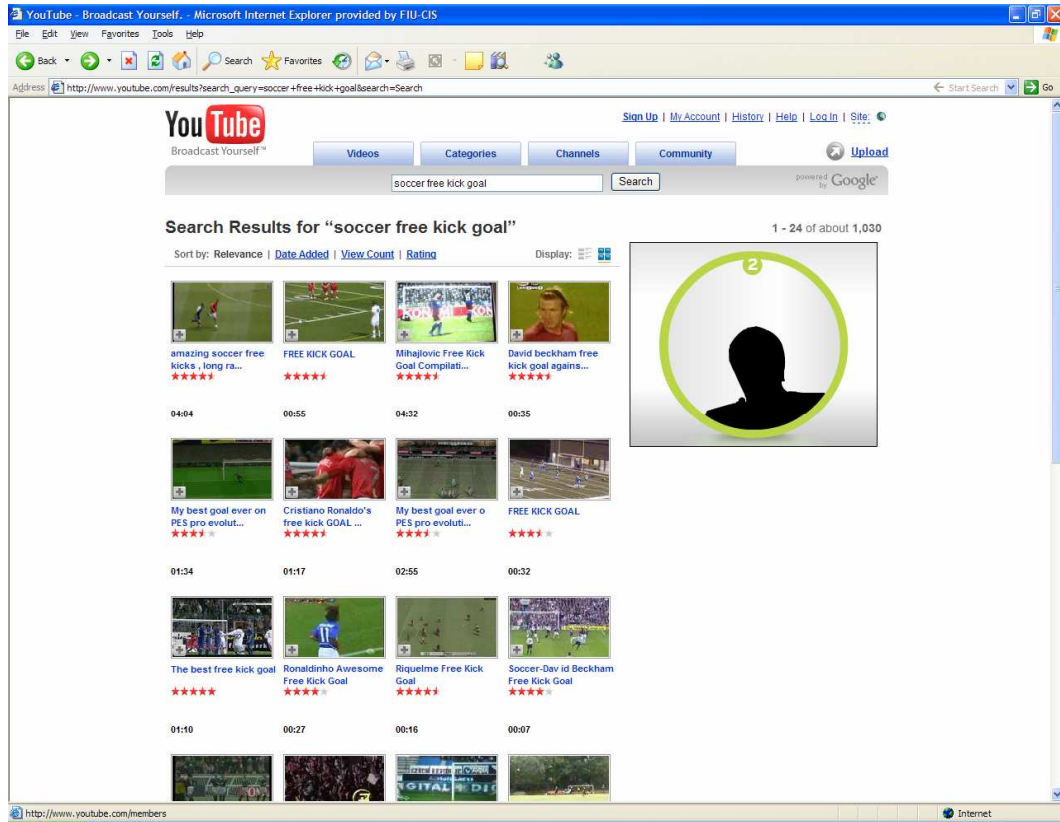


Figure II-8. User interface for Youtube video search

2.4.1.9 Google Image/Video Search Engine

As a company specializing in Internet searches and online advertising, Google indexes billions of web pages and offers users convenient tools to search for information through the use of keywords and operators. Google has also employed web search technology in other search services, including image search and video search. The new Google Video has been transformed into a video search service that provides links to online and offsite video content [GVideo]. It can index media from YouTube as well as an assortment of other video hosting sites, including

Metacafe, MySpace, and BBC. Previews are available next to search results that are hosted on YouTube or Google Video, and thumbnail snapshots are available for content hosted by other providers. Much like Google image search, a frame with relevant Google-provided functionality appears at the top of the window when the user clicks through a search result. Figure II-9 shows the video searching results of “soccer free kick goal”, where the keywords are usually found from the video descriptions and associated web pages.

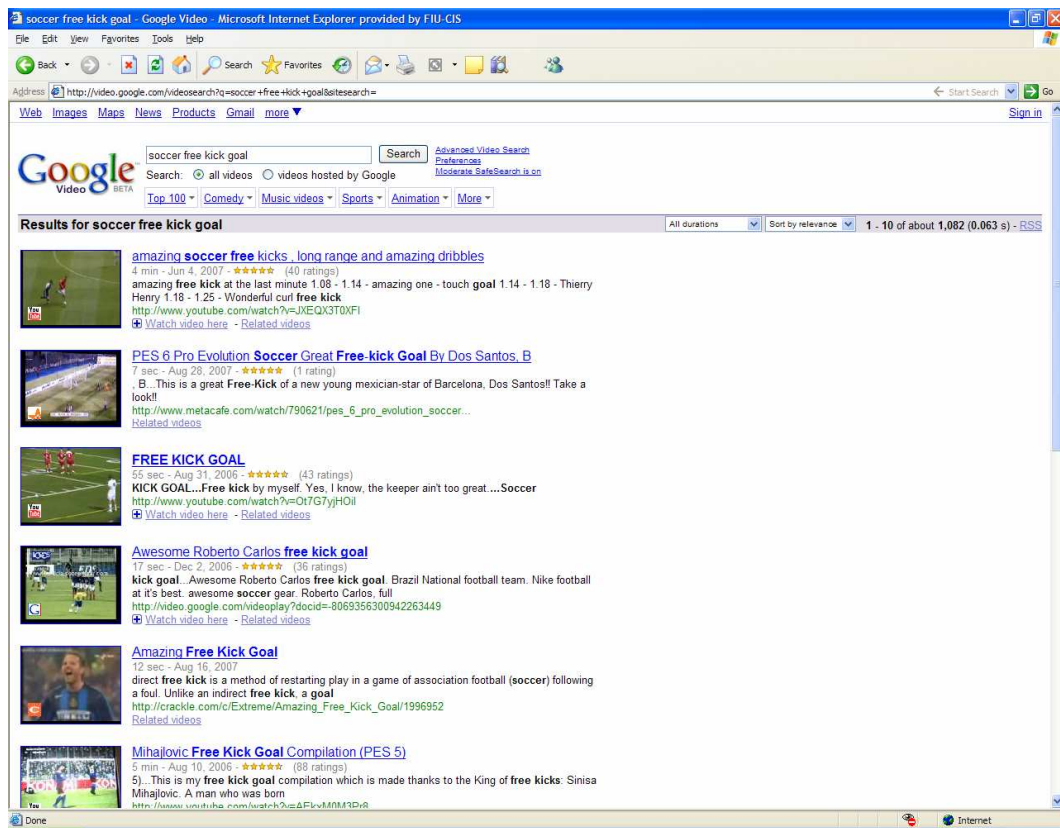


Figure II-9. User interface for Google video search

2.4.1.10 Yahoo! Image/Video Search Engine

Originally Yahoo! started as a web directory of other websites, organized in a hierarchy index of pages. Over time, Yahoo! designed and developed its own web crawler and search engine. In late 2007, Yahoo! Search was updated with a more modern appearance with Search Assist added, which can automatically suggest and offer related search terms as they are typed.

The keyword-based image/video search functions are also supported by Yahoo! Search based on the web searching techniques. A combination of factors are used in Yahoo! Video Search [YVideo] to enable users to find and view different types of online video, including movie trailers, TV clips, news footage, and independently produced video. These factors include Yahoo!'s media crawling and ranking technology, its content and media relationships, as well as its support for Media Really Simple Syndication (Media RSS), a self-publishing specification for audio and video content. Yahoo! Video Search supports open standards in the creation and syndication of content. By supporting Media RSS, Yahoo! Video Search hopes to foster openness and choice for independent video publishers looking to promote their content. Figure II-10 shows the Yahoo! Video Search results for a query with keywords “Soccer Free Kick Goal”.

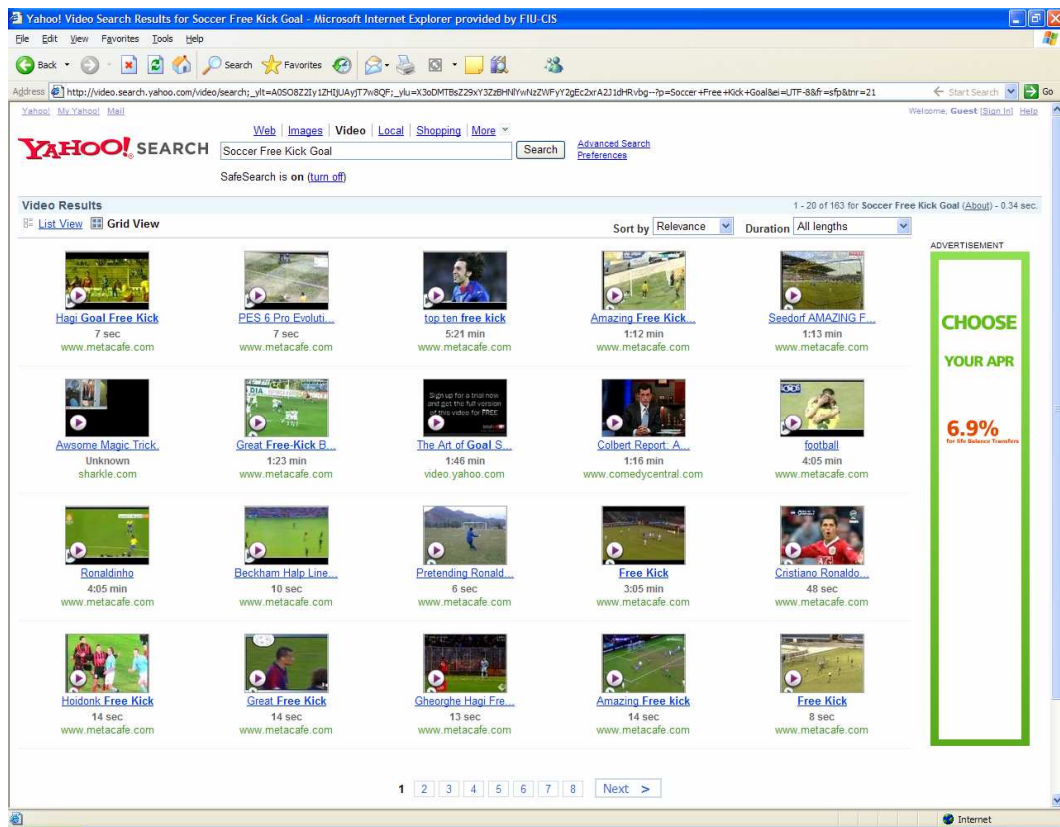


Figure II-10. User interface for Yahoo! video search

2.4.1.11 AOL Truveo Video Search Engine

AOL Truveo [Truveo] operates under the idea that users do not merely search for video by entering specific words or phrases, as they would when starting a regular web search. Instead, Truveo assumes that people do not always know exactly what they are looking for in online video searches, so browsing through content can help to retrieve unexpected but welcome results. Truveo provides useful interfaces to support video browsing. It repeatedly displays spot-on results when users are looking for a video about a specific subject, or provides a variety of other video clips that are similar to encourage users to view more results. As shown in Figure II-11, one useful feature of Truveo is the way it shows results: by sorting clips into neatly organized categories, such as Featured Categories, Channels, and Tags. These buckets spread out on the page in a grid-like manner, giving users more to see in a quick glance.

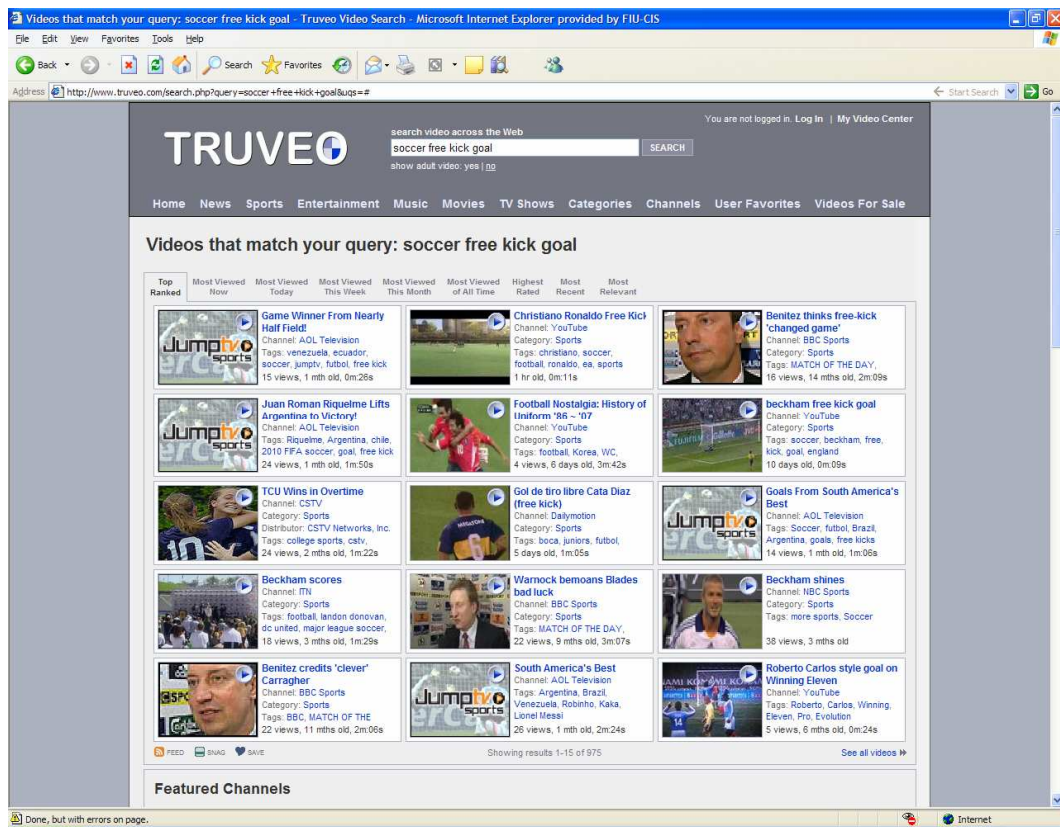


Figure II-11. User interface for AOL TRUVEO video search

2.4.2 Multimedia Presentation Authoring and Rendering Systems

2.4.2.1 LAMP: Laboratory for Multimedia Presentations Prototyping

Gaggi et al. [Gaggi06] propose a system called LAMP, a prototyping environment which allows an author to set up and test a complex hypermedia presentation. The media editing tool in LAMP is implemented based on a graph notation, where the nodes are media objects and the edges are the synchronization relations between them. An execution simulator is also included to test the presentation dynamics by manually triggering the related events. Finally, a player is developed to display the presentation and visually interpret the synchronized schema. In Figure II-12, the LAMP interface is shown with an example synchronization graph for news-on-demand presentation.

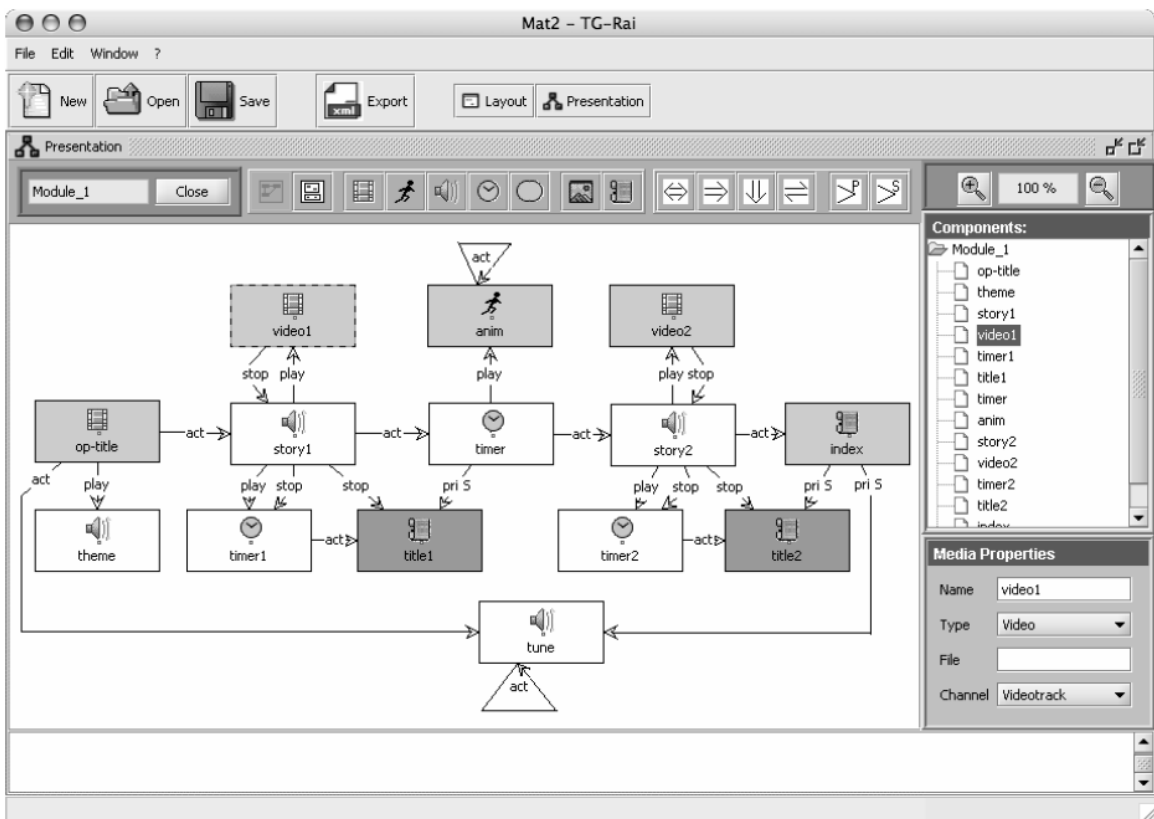


Figure II-12. LAMP interface with the synchronization graph of a news-on-demand presentation

2.4.2.2 T-Cube: Multimedia Authoring System for E-Learning

Ma et al. [Ma03] introduce a rich media authoring system, T-Cube, which has been designed by and used at the University of Trier for eLearning. By using T-Cube, multimedia content can be constructed and presented to students with either offline (CD/DVD/download) or online (in real time or on demand) usage. The multimedia-based teaching contents, including video, audio and screenshot, are recorded and encoded at the classroom and simultaneously published on the Internet. Figure II-13 illustrates an example of the layout design and the generated presentation interface for T-Cube.

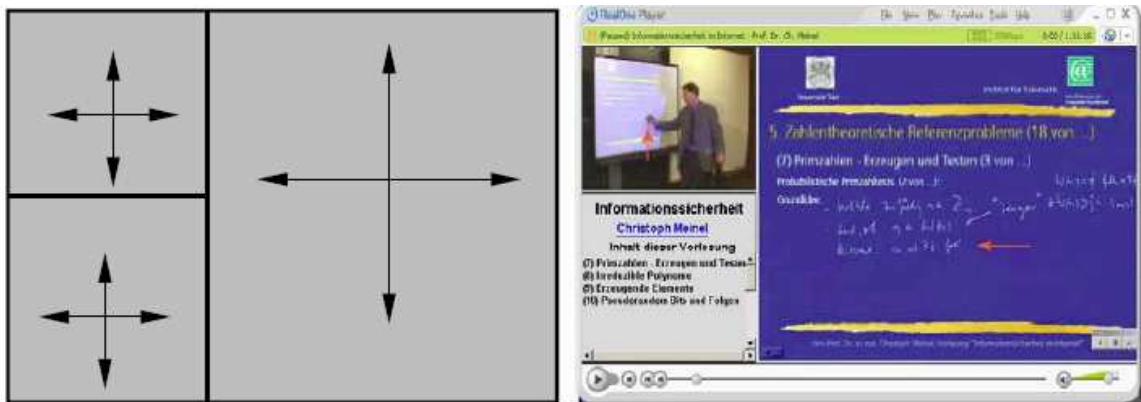


Figure II-13. Views layout and user interface for T-Cube

2.4.2.3 Madeus: Structured Media Authoring ENvironment

In the paper [Jourdan98], Madeus is developed to help in editing media documents that contain fine-grained synchronizations in the temporal, spatial and spatiotemporal dimensions. A semiautomatic tool is integrated in the system that analyzes, generates and allows the editing of the content description of video media. Madeus employs a document model that is based on structured, temporal interval-based and region-based models. Figure II-14 illustrates the system interface for the timeline-based multimedia presentation authoring environment.

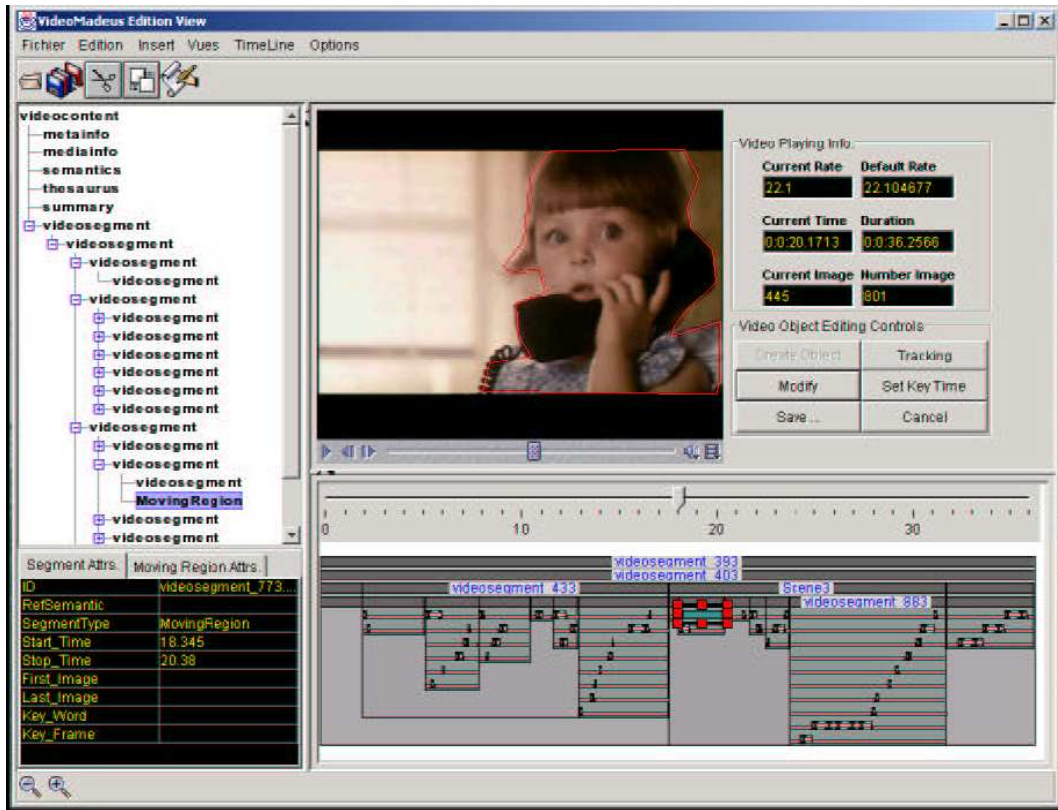


Figure II-14. Structured media authoring environment of Madeus

CHAPTER III. OVERVIEW OF THE FRAMEWORK

This chapter mainly provides an overview of DIMUSE, the proposed framework for the design of a distributed multimedia system. Three major modules have been incorporated in the proposed framework: multimedia retrieval module, multimedia presentation design and rendering module, and security management module. In order to efficiently and effectively integrate them together, a set of client-side interfaces as well as server-side components are implemented and connected.

As illustrated in Figure III-1, users are required to provide their ID and password for logging-in purposes. A security checking component is designed to check the user role and security rule by sending the request to the server-side security checker. After passing the security check, general users are allowed to access both the multimedia retrieval and presentation modules (shown as green links), while the administrators are also permitted to access the security management module (shown as orange links). Users are able to traverse between the multimedia presentation design environment and the multimedia retrieval components (e.g., content-based image retrieval, video browsing, soccer video event pattern retrieval, etc.). The system provides the flexibility for users to search their anticipated media files and download the data back to the presentation design interface. These source media files are listed and users can preview and then choose their preferred material to architect diverse kinds of multimedia presentations. These designed presentation models can be rendered to a real presentation and displayed on the Java media player or a web browser. However, not all the multimedia information can be accessed or displayed completely because of the security assurance issues. The proposed security management module actually takes charge of the multimedia data accessing control of the whole application, including both of the previously mentioned modules. Therefore, the user requests generated in retrieval module and presentation module may receive three kinds of responses. First, if the user has full access to some specific media file under the specific environment, he/she

can view and download the source data in both of the retrieval and presentation modules. Second, if the user is not allowed to access some specific media file, no matter what the reason is, the system basically rejects the request and no data will be offered. Third, if the user has partial accessibility permission to the specific media files, these files will be processed such that restricted parts are hidden and not accessible by the user. However, users can still view the processed media files for the un-sensitive parts and download this processed data file to the presentation. The only difference here is that the restricted data objects are not shown in the final presentation.

The whole framework adopts multi-thread client/server architecture, where a set of network protocols can be utilized for the transmission of multimedia information, including both requests and media data. These protocols include TCP/IP, UDP, HTTP, RTP, etc. In the server-side, an object-relational database is designed for multimedia data storage and management purposes. In DIMUSE, multiple categories of data are stored, which include:

- (1) Source media data, including text, audio, image, and video data.
- (2) Processed multimedia data, including the meta data, image objects, video segments, etc.
- (3) Low level visual/audio features.
- (4) High-level semantic information, including affinity relationships, and user access frequencies and access patterns, etc.
- (5) Security roles and rules, including user roles, object roles, environmental roles (i.e., temporal roles and IP address roles), and security policy rules.

In DIMUSE, the Markov Model Mediator (MMM) [Shyu03] and HMMM model is adopted to model multimedia related database. The security-related information is mainly stored in XML.

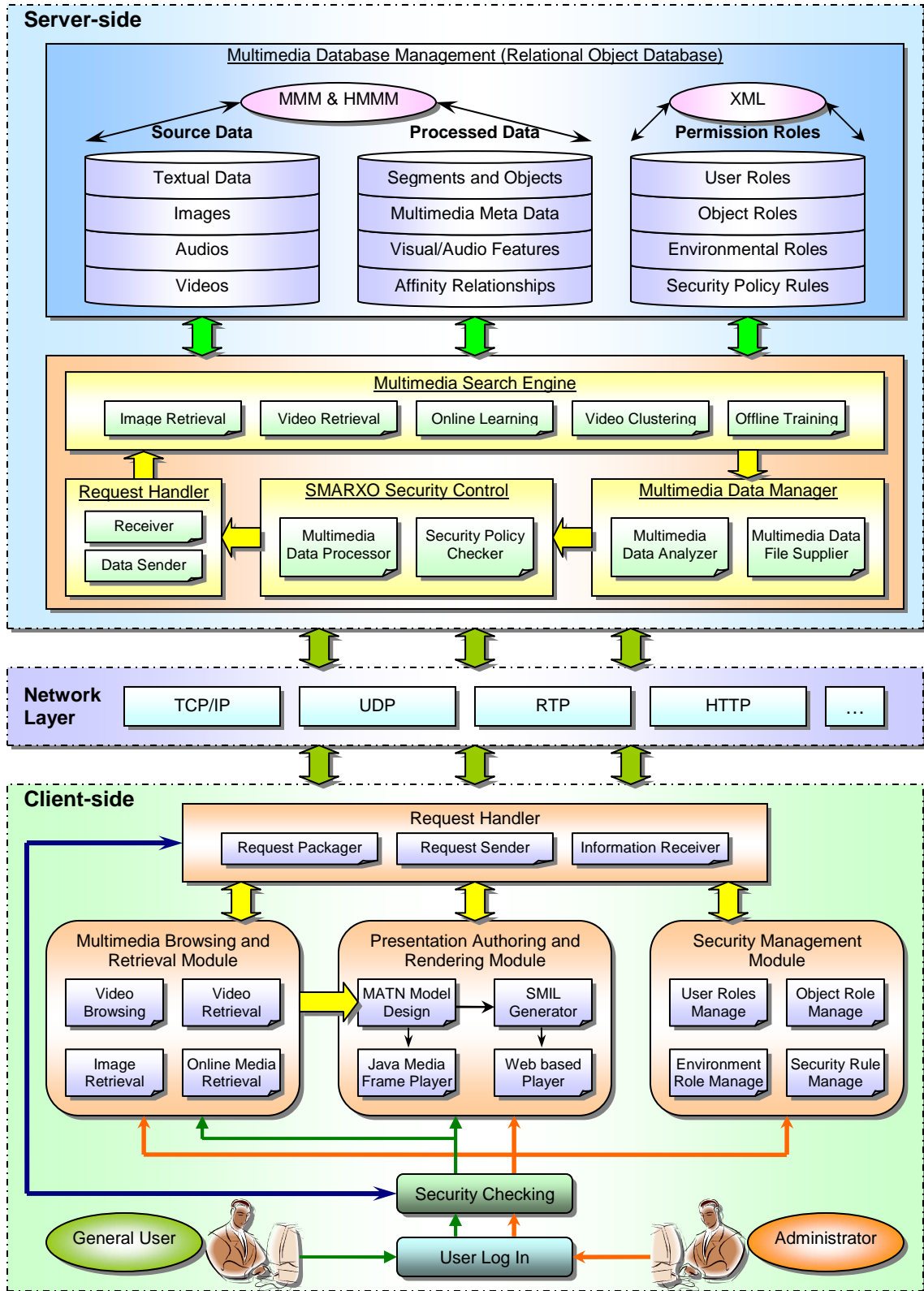


Figure III-1. Overall framework and components of DIMUSE

Server-side engines are developed according to the diverse functionalities with database access and computation-intensive operations.

- (1) A request handler is implemented for receiving the request and sending back the results.
- (2) A multimedia search engine is developed to support content-based image retrieval, video event/pattern queries, online learning, offline training, as well as the video clustering, etc.
- (3) The multimedia data manager is designed, where the data analyzer is responsible for some analysis functions, such as feature extraction and video segmentation. The multimedia data/file supplier is capable of catching the requested data/file from the database.
- (4) The SMARXO security control component is also incorporated in the server-side. This sub-module is utilized to manage the access control roles and rules, perform security checking, and process the media data upon the security constraints.

In the client-side, a set of user-friendly interfaces as well as some managing functions are included to fulfill the diverse requirements.

- (1) Multiple media retrieving interfaces are facilitated to support various methods of data accessing, including video browsing, content-based image retrieval, conceptual video retrieval, and web-based multimedia retrieval.
- (2) The multimedia presentation component incorporates a Multimedia Augmented Transition Network (MATN) [ChenSC00a][ChenSC01b] based presentation design environment. The system can provide two possible methods for presentation rendering: JMF player and web-based player. The SMIL interpreter is also included for the efficient converting from MATN model to HTML+SMIL scripts.
- (3) Security management module offers the interfaces for user role managing, object role managing, temporal role managing, IP address role managing, as well as security rules managing.

3.1 Multimedia Database Modeling and Retrieval Module

The major challenges of content-based multimedia retrieval include not only the difficulty of extracting features and generating semantic indexes for hierarchical multimedia contents, but also the incapability of discovering hidden personalized user interests.

3.1.1 Image Database Modeling and Retrieval using MMM

As a well established mathematical construct, the Markov Model Mediator (MMM) [Shyu03] is applied to model complicated images as well as the retrieval engine in the content based image retrieval component of DIMUSE. The development of the MMM supported image databases are accomplished by employing the object-relational database. This module is not the focus of this dissertation so please refer to paper [Shyu04d] for more details.

3.1.2 Video Database Modeling and Retrieval using HMMM

By extending MMM to a multiple level description, an innovative database modeling mechanism called Hierarchical Markov Model Mediator (HMMM) is proposed in this research for video database modeling, storage and retrieval purposes. In order to model hierarchical media objects, HMMM is composed with multiple levels of MMM models which are connected effectively and efficiently.

The dream of pervasive multimedia retrieval and reuse will not be realized without incorporating semantics in the multimedia database. In this research, HMMM integrates low-level features, semantic concepts, and high-level user perceptions for modeling and indexing multiple-level video objects to facilitate temporal pattern retrieval. A variety of multimedia objects in different levels are modeled with the state sequences associated with their transition probabilities by incorporating the temporal meanings and/or their affinity relationships. Different from the existing database modeling methods, this proposed approach carries a stochastic and dynamic process in both search and similarity calculation. In the retrieval of semantic event patterns, HMMM always tries to traverse the most optimized path, and therefore it can assist in retrieving

more accurate patterns quickly with lower computational costs. Moreover, HMMM supports feedbacks and learning strategies, which can proficiently assure the continuous improvements of the overall performance.

3.1.3 Online Learning and Offline Training via HMMM

In DIMUSE, an innovative method is proposed and developed to capture the individual user's preferences by considering the low-level features as well as the semantic concepts and relationships. With the hierarchical and stochastic design for video database modeling, the proposed framework supports not only the general concept-based retrieval methods, but also the complicated temporal event pattern queries. In the online learning approach, a set of MMM instances are created for the user with distinct preferences, and the system is capable of learning and then generating the updated results to satisfy the special information requirements. With the proposed online learning mechanism, the retrieval and ranking of video events and the temporal patterns can be updated dynamically in real time to satisfy individual user's interests and information requirements.

Moreover, user feedback is efficiently accumulated for the offline system training process such that the overall retrieval performance can be enhanced periodically and continuously. That is, the overall system can always remain as a learning mechanism since the access patterns and frequencies from various users can be proficiently stored and analyzed for the long-term offline system training.

The offline training process is normally initiated only when the number of feedbacks reaches a certain threshold. This could improve the performance but it becomes a manual process to decide the threshold and initiate the training process. To address this challenge, we propose an advanced training method by adopting the association rule mining technique [Zhao07b], which can effectively evaluate accumulated feedback and automatically invoke the training process. Training is performed per video rather than for all videos in the database, making the process

more efficient and robust. In addition, it can further improve the semantic models in the video database and continuously improve retrieval performance in the long run. As an example, the proposed method is applied to a soccer video retrieval system and the experimental results are analyzed.

Further, we applied the Hierarchical Markov Model and system training mechanism in a mobile-based video retrieval system (MoVR). We developed innovative solutions for personal video retrieval and browsing through mobile devices with the support of content analysis, semantic extraction, as well as user interactions. HMMM-based user profiles were designed to capture and store individual user's access histories and preferences such that the system can provide the "personalized recommendation." We also employed the fuzzy association concept to empower the framework so that the users can make their choices of retrieving content based solely on their personal interests, general users' preferences, or anywhere in between. Consequently, the users gain control in determining the desirable level of tradeoff between retrieval accuracy and processing speed. A mobile-based soccer video navigation system was implemented and examined to demonstrate the performances of the proposed MoVR framework.

3.1.4 Video Database Clustering

To accommodate the requirements of multi-disciplinary video retrieval in the distributed multimedia applications, a conceptual video database clustering technique is proposed, implemented and incorporated in DIMUSE.

As mentioned above, the video database is modeled by HMMM, which is a hierarchical learning mechanism and supports both online and offline training. Actually, the cumulated historical queries and the associated user feedbacks can be reused to update the affinity relationships of the video objects as well as their initial state probabilities. Correspondingly, both the high level semantics and user perceptions are employed in the video clustering strategy. The associated retrieval algorithm is also proposed to search the top- k patterns with traversing the

minimum number of clusters. This technique assists to cluster the related media data to improve the retrieval performance. With the clustering information, the database structure can be further refined by adding a new level of MMM to model the clusters. Furthermore, the computation costs in the query processing can be significantly reduced.

3.2 Multimedia Presentation Module

A multimedia presentation is a delivery medium of a collection of media streams which are constrained by temporal synchronization relationships among each other. An abstract model called Multimedia Augmented Transition Network (MATN) [ChenSC00a][ChenSC01b] is adopted in DIMUSE as the presentation model. This component is one of the key modules in our distributed multimedia management system. However, the multimedia presentation module and its related techniques are not the major contributions of this dissertation. This module will only be presented in system integration section of Chapter VII and the reader is referred to the book of [ChenSC00a] for more details related to MATN model.

3.2.1 Presentation Design with MATN Model

An MATN model is composed with a group of states connected by directed arcs with marked multimedia strings. By combining structure-based authoring with well-defined graphic-based notations, MATN offers great flexibility for users to design a complicated multimedia presentation with synchronization of the heterogeneous multimedia objects. MATN supports the specification of temporal constraints for multimedia content, and these temporal requirements can be satisfied at runtime. MATN also provides a good data structure for the implementation to control multimedia playback.

A group of features are implemented in the MATN-based presentation design environment such that users can easily add or delete the presentation states, adjust the temporal constraints for each arc, design a sub-network to accommodate diverse conditions, etc. In addition, the MATN file format is designed by considering the MATN structures and embedded

information. The file saving and opening functions are also developed to store and resume the user-designed MATN based presentations.

3.2.2 Presentation Rendering with JMF and SMIL

A presentation rendering component is implemented and integrated in DIMUSE to convert the designed MATN model to a multimedia scenario perceivable to the users. Basically, there are two approaches provided to fulfill different requirements based on diverse environments.

One approach is to synchronize and display the presentation in a client-side player which is implemented by using Java Media Framework (JMF) technologies. JMF provides superior techniques for rendering the presentation models into a stand-alone application in a runtime environment. Four kinds of distinct media players are developed to exhibit the text, image, audio, and video. Since the MATN model captures the spatial and temporal relationships, they can be interpreted and utilized to control the players.

The other approach is to convert the designed MATN model to SMIL languages, which can be displayed in the web browser directly. SMIL notations can be combined with the HTML file. Therefore, an SMIL template is deployed in the system such that the MATN structure can be interpreted into the SMIL+HTML format. The SMIL-based scripts can be displayed wherever the web browser is available. This approach is specifically suitable for the online-based multimedia applications.

3.3 Security Management Component

3.3.1 Security Policy and Role Managing

The main objective of security policy and role manager is to deal with the various access control roles and rules. In the proposed security framework, four kinds of roles are defined to handle a request behavior in the multimedia applications. First, as the most fundamental feature in RBAC, subject roles are defined to recognize the users' role in the application. As the

permissions are granted to the roles, the users with the same role are permitted to perform the same set of operations. Second, object roles are facilitated to control the access of not only the source media data, but also the embedded objects and segments. Third, temporal roles are responsible to control the effective time of the access functionalities. Fourth, spatial roles are designed to restrict unauthorized accesses from alien computers based on the checking of their IP addresses. By combining all these four roles, the security access policies are defined as access control rules. These control information are designed to be stored in the XML format so that they can be easily retrieved and viewed.

3.3.2 Security Checking

Upon receiving an access request, the security checker will first validate the user ID and Password and identify its subject role. As operation time and operator's IP address can be easily obtained, the temporal roles and spatial roles are also checked. For the requested media data and objects, the object roles are considered. Based on the security checking results, the system responds to the request with the following three possibilities: First, the user's access with certain media data is denied; Second, the user is allowed to access and perform certain operations for the requested media data; Third, the user is allowed to access or operate partial contents of the requested media data. Within the third condition, the system will perform media data processing to hide the restricted parts (objects, segments, etc.) from the source media data and show the processed multimedia data to the user.

3.3.3 Multimedia Data Managing and Processing

The multimedia data manager is responsible of managing the media source data, along with their extracted objects or segments. For the purpose of supporting multi-level security, multimedia data are required to be stored in a hierarchical way. The recent multimedia data processing techniques can help in the multimedia indexing phase to extract the multimedia objects from the source data. For instance, image segmentation can help to identify the image

objects; video decoding, shot detection and scene detection can assist to achieve meaningful video shot sequences. In addition, users are allowed to manually identify and define their target multimedia objects or segments. In case of a multimedia document containing restricted objects, the multimedia data manager takes charge to perform the data processing such that the restricted parts (e.g. image objects, video segments) are hidden while the users are still capable of viewing the remaining parts of the source media.

3.4 Multimedia Application and System Integration

3.4.1 DMMManager: Distributed Multimedia Manager

Based on the proposed framework, a distributed multimedia system called DMMManager is developed. DMMManager adopts a multi-threaded client-server architecture. In the server-side, an object relational database called PostgreSQL [PostgreSQL] is employed to store the media source data, meta data, features, and the other information. A database engine is developed with C++ to support computation intensive processes, such as query processing, feature extraction, media supply, online and offline training, etc. The client-side application is developed with Java, which provides a variety of user-friendly interfaces for users to issue the multimedia queries, download the retrieved media files, design and view multimedia presentations, etc. DMMManager is capable of supporting a full scope of multimedia management functionalities by efficiently integrating multiple modules together. It is also utilized as a test bed for our recent multimedia researches. This current application stores totally 10,000 color images, around 50 videos along with more than 10,000 video shots. A series of client-side interfaces are designed to rank the content-based image retrieval (CBIR) and content-based video retrieval (CBVR) query results by similarity scores. Moreover, users are allowed to mark the resulting media object with positive or negative labels. When the number of accumulated feedbacks reaches a threshold, the offline training process will be triggered to refine the underlying affinity relationship matrix to improve the overall retrieval performance.

Particularly, a soccer video retrieval system named SoccerQ [ChenSC05a] is developed and integrated in DMManager to support not only the basic queries but also the complicated temporal event / event pattern queries for a soccer video database. The client-side interfaces integrate the video browsing panels and soccer event query in a common framework. The client-side applications can collect the user requests based on their anticipated soccer events with the associated temporal relationships, then construct a request message and send it to the servers. The server-side database engine extracts the related parameters from the received request, retrieves the desired video sequences and finally returns the video clips to clients. In this research, the SoccerQ system is further updated and expanded to incorporate new techniques such as video database modeling and clustering.

CHAPTER IV. MULTIMEDIA DATABASE MODELING AND RETRIEVAL

This chapter addresses the research issues involved in the multimedia database modeling and retrieval module of DIMUSE, which offers a variety of functionalities for users to search for and access their favorite media files. An innovative mechanism called Hierarchical Markov Model Mediator (HMMM) is proposed for managing multiple levels of media objects. As an example, the most basic 2-level HMMM model and the associated retrieval algorithm are introduced for temporal event pattern retrieval. Furthermore, a conceptual video clustering strategy is proposed to improve the overall retrieval performance and reduce the computation time by constructing the 3rd level HMMM model. A soccer video retrieval system has been developed and employed as a test bed for all these newly proposed techniques.

4.1 Introduction

Due to the rapid propagation of multimedia applications that require data management, it becomes more desirable to provide effective multimedia database modeling and retrieval techniques capable of representing and searching the rich semantics in media data. In the existing content-based multimedia retrieval approaches, there are four essential challenges to be addressed.

The first challenge is to bridge the “semantic gap” between the multi-modal visual/audio features and the rich semantic, which means that the users anticipate the database systems to associate their queries for searching and browsing purposes based on the semantic concepts represented by the digital media data. The semantic interpretations are required to be derived and facilitated efficiently by utilizing assorted methodologies and techniques from various disciplines and domains, even though many of them do not belong to the traditional computer science fields.

The second emerging challenge is to proficiently model and search for the multimedia objects by considering their temporal and/or spatial relationships. It is anticipated that a

generalized database modeling mechanism can be designed to incorporate all the related multimedia information to support not only the basic retrieval methods, but also the complicated temporal event pattern queries (i.e., to retrieve the video clips containing a user-designed sequence of semantic events that follow some specific temporal relations). Here, semantic event annotations are used to recognize real-world representation of the video shots, also referred to as events or concepts.

Another crucial problem is to incorporate high-level user perceptions in the database modeling and retrieval process. When performing multimedia retrieval, different users may eventually have diverse interests, leading to separate preferences for the anticipated multimedia objects. Therefore, multimedia summarization, retrieval, and ranking should focus on satisfying the individual user's interest and information requirements. Hence, users' perceptions need to be taken into account when modeling the underlying database and designing the retrieval algorithm.

Finally, an additional important research topic is to mine and cluster the multimedia data, especially to accommodate the requirements of video retrieval in a distributed environment. With the recent advances in multimedia technologies, the number of multimedia files and archives increases dramatically. Since the multimedia databases may be distributed geographically through the local network or world-wide Internet, the associated workloads could be quite expensive when dealing with complicated video queries. In particular, semantic-based video retrieval is multi-disciplinary and involves the integration of visual/audio features, temporal/spatial relationships, semantic events/event patterns, high-level user perceptions, etc. Therefore, it is expected to utilize a conceptual database clustering technique to index and manage the multimedia databases such that the related data can be retrieved together and furthermore the communication costs in the query processing can be significantly reduced.

In this chapter, an integrated and interactive framework is proposed for video database modeling and retrieval approaches to efficiently and effectively organize, model, and retrieve the

content of a large scale multimedia database. In this proposed work, the semantic descriptions and user preferences are successfully applied to enhance the performance not only for multimedia content management but also database clustering and conceptual video retrieval. In order to achieve the goal, this newly proposed framework includes a variety of advanced techniques.

First, for the purpose of data processing and concept mining, this framework adopts multi-disciplinary techniques, such as content-based image analysis, audio feature extraction, video shot detection and segmentation algorithms, data mining, and machine learning. Second, the Hierarchical Markov Model Mediator (HMMM) mechanism is introduced to efficiently store, organize, and manage low-level features, multimedia objects, and semantic events along with high-level user perceptions (such as user preferences) in the multimedia database management system (MMDBMS). Third, innovative feedback and learning methods are proposed to support both online relevance feedback and offline system training such that the system can learn the common user perception as well as discover the individual user requirements. Fourth, a clustering strategy is also proposed to group video data with similar characteristics into clusters that exhibit certain high level semantics. This proposed approach is able to reuse the cumulated user feedback to perform video clustering, such that the overall system can learn the user perceptions and also construct more efficient multimedia database structure by adopting the video clustering technique. For evaluation purposes, a soccer video retrieval system utilizing the proposed framework is developed.

4.2 Overall Framework

In general, multimedia data and metadata can be categorized into three groups: entities, attributes, and values, where the description of an entity is composed of the combinations of attributes and their corresponding values. One of the significant characteristics of video data is that video entities may pose various temporal or spatial relationships. Accordingly, users are normally interested in specific semantic concepts and the associated temporal-based event

patterns when querying a large scale video archive. However, some of the current computer vision and video/audio analysis techniques only offer limited query processing techniques on textual annotations or primitive low-level or mid-level features. Although a variety of researches have begun to consider retrieval of semantic events and the salient objects, a comprehensive database modeling technique is lacking to support the access and query on the temporal-based event patterns.

In this study, a temporal event pattern is defined as a sequence of semantic events that follow some specific temporal relations. Here, a semantic event annotation is used to mark real-world situations of the video shot, also referred to as events. For instance, in a soccer video, the events such as “goal”, “corner kick”, “free kick”, “foul”, “goal kick”, “yellow card”, and “red card” are considered. An example temporal pattern query can be expressed as follows: “A user wants to search for a specific soccer video segment with the following temporal patterns. At first, a goal event resulting from a free kick happens. After that, a corner kick occurs at some point in time, followed by a player change, and finally another goal shot event happens.”

In our earlier studies, we proposed various approaches in the multimedia area, especially video data mining, indexing and retrieval. In [ChenSC03a][ChenSC04a], the methodologies were proposed to identify the “goal” and “corner kick” events. Moreover, a temporal query model related graphical query language was introduced in [ChenSC05a] to assure the soccer event queries with the support on temporal relationships. In this proposed approach, the Markov Model Mediator (MMM) is extended to the Hierarchical MMM mechanism such that the multiple-level video entities and their associated temporal or affinity relationships can be efficiently modeled to answer this type of temporal pattern query.

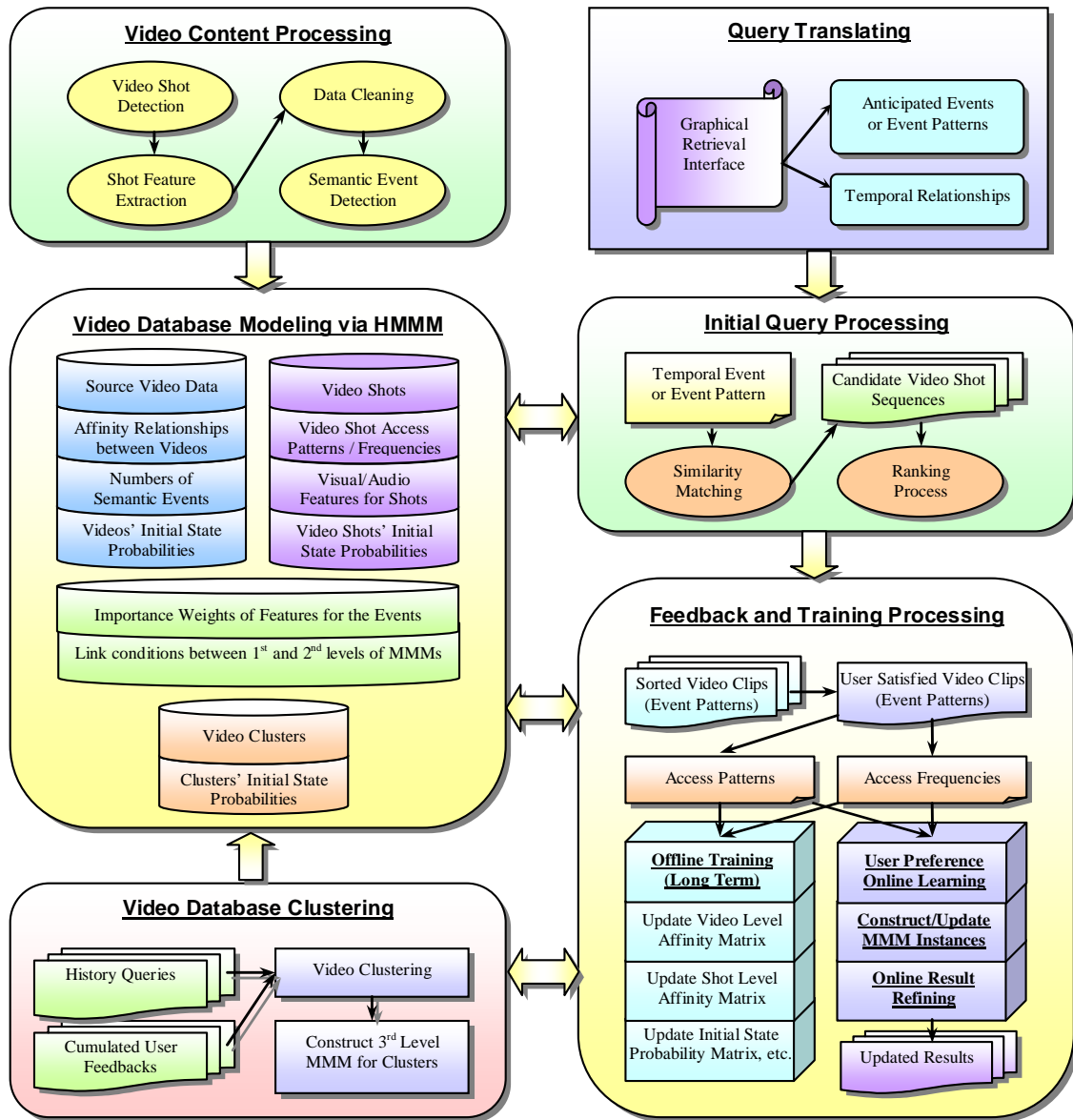


Figure IV-1. Overall framework of video database modeling and temporal pattern retrieval utilizing HMMM, online learning, offline training and clustering techniques

As illustrated in Figure IV-1, our proposed framework consists of six major stages.

- 1) The first step is to process the video data by utilizing multi-disciplinary techniques for video shot boundaries detection and shot features extraction. After the data cleaning procedure, data mining techniques are employed to detect the semantic events. The algorithms for soccer event detection can be found in [ChenSC05a].

- 2) Secondly, in the Video Database Modeling module, HMMM is employed to model the extracted features, detected events, segmented video shots along with the original source data. The proposed three-level HMMM model is capable of managing the hierarchical multimedia objects (i.e., video clusters, videos, video shots) as well as their associated affinity relationships. However, it should be noted that initially only the first two levels of MMM models are constructed. The third level MMM model is constructed after video clustering.
- 3) Once a temporal pattern query is issued via the graphical retrieval interface, the Query Translator analyzes the user requirements and encodes the query to a set of expected events and their associated temporal patterns.
- 4) These requests are then sent to the server-side query processing component (Initial Query Processing) as inputs. The similarity matching process is then executed to achieve the candidate video shot sequences and finally they are sorted according to the similarity scores.
- 5) With these initial results retrieved, users are allowed to choose their preferred patterns by marking them as positive. With the online learning mechanism, the system can refine the query results and rank them in real-time capturing a user's specific perceptions. Moreover, these historical queries with feedback are accumulated in the database for future usage. As illustrated in the right-lower box, the HMMM mechanism can be trained by considering the stored user feedback for continuous system learning. The multimedia system training and learning strategies will be further discussed in the next chapter.
- 6) Finally, these historical access patterns and frequencies are also utilized in the video clustering mechanism as demonstrated in the left-lower box. After this process, the HMMM-based database model can be updated by adding the third level of MMM

model for the generated video clusters. As the system learns user knowledge, all of these updates can help to enhance the overall retrieval performance and reduce the computation costs.

4.3 Hierarchical Markov Model Mediator (HMMM)

The Markov Model Mediator (MMM) [Shyu03] is a well-established mathematical construct capable of modeling complicated multimedia databases and can efficiently collect and report information periodically. MMM has been successfully applied in several applications such as content-based image retrieval [Shyu04b][Shyu04c][Shyu04d] and web document clustering [Shyu04a].

Definition IV-1: Markov Model Mediator (MMM) [Shyu03]

An MMM is represented by a 5-tuple $\lambda = (S, F, A, B, \Pi)$, where S is a set of states which represents distinct media objects; F includes a variety of distinct features; A denotes the states transition probability distribution, where each entry actually indicates the relationship between two media objects, which can be captured through the off-line training processes; B represents the low-level feature values of media objects; and Π is the initial state probability distribution, which indicates the likelihood of a media object being selected as the query.

Here, a media object may refer to an image, a salient object, a video shot, etc., depending on the modeling perspective and the data source. A and Π are used to model user preference and to bridge the semantic gap, which are trained via the affinity-based data mining process based on the query logs. The basic idea of the affinity-based data mining process is that the more two media objects Obj_m and Obj_n are accessed together, the higher relative affinity relationship they have, i.e., the probability that a traversal choice to state (media object) Obj_n given the current state (media object) is in Obj_m (or vice versa) is higher. Details about the training and construction processes of the MMM parameters can be found in [Shyu04b].

In this research, MMM is extended to multiple level descriptions and utilized for video database modeling, storage and retrieval purposes. In particular, the Hierarchical Markov Model Mediator (HMMM) is designed to model various levels of multimedia objects, their temporal relationships, the detected semantic concepts, and the high-level user perceptions. The formal description of an HMMM is defined as below.

Definition IV-2: Hierarchical Markov Model Mediator (HMMM)

An HMMM is represented by an 8-tuple $\Lambda = (d, \mathbf{S}, \mathbf{F}, \mathbf{A}, \mathbf{B}, \mathbf{\Pi}, \mathbf{O}, \mathbf{L})$, as shown in Table IV-1.

Table IV-1. HMMM is an 8-Tuple: $\Lambda = (d, \mathbf{S}, \mathbf{F}, \mathbf{A}, \mathbf{B}, \mathbf{\Pi}, \mathbf{O}, \mathbf{L})$

Tuple	Representation
d	Number of levels in an HMMM.
$\mathbf{S}(S_n)$	The group of multimedia object sets in different levels, where $n = 1$ to d .
$\mathbf{F}(F_n)$	The sets of distinct features or semantic concepts of the specific multimedia objects, where $n = 1$ to d .
$\mathbf{A}(A_n)$	The group of state transition probability matrices. The higher the entry is, the tighter the relationship that exists between the target objects, where $n = 1$ to d .
$\mathbf{B}(B_n)$	The group of feature/concept matrices of different-level MMMs, where $n = 1$ to d .
$\mathbf{\Pi}(\Pi_n)$	The initial state probability distributions, where $n = 1$ to d .
$\mathbf{O}(O_{i+1} \rightarrow F_{i+1} \times F_i)$	The weights of importance for the lower-level features in F_i when describing the higher level feature concepts in F_{i+1} , where $i = 1$ to $d-1$.
$\mathbf{L}(L_{i+1})$	Link conditions between the higher level states and the lower level states, where $i = 1$ to $d-1$.

Each of the MMM models incorporates a set of matrices for affinity relationships, features/concepts, and initial state probability distributions. Let $|S_n|$ denote the size of S_n , which means the number of the n^{th} level MMMs (or state sets).

- $S_n = \{S_n^g\}$, where $1 \leq g \leq |S_n|$. Here, S_n^g represents the state set of the g^{th} MMM in the n^{th} level. Since the modeling descriptions of the MMM models in each level are the same

and to simplify the notation, S_n is generically used to represent one member in S_n , i.e., the set of states in the current MMM model of interest and thus g is ignored.

- $A_n = \{A_n^g\}$, where $1 \leq g \leq |S_n|$. $A_n^g (A_n^g \rightarrow S_n^g \times S_n^g)$ is designed as the affinity matrix for the g^{th} MMM in the n^{th} level. It describes the affinity relationship between pairs of states in S_n^g . Similarly, A_n is generically used to represent any member in A_n .
- $B_n = \{B_n^g\}$, where $1 \leq g \leq |S_n|$. $B_n^g (B_n^g \rightarrow S_n^g \times F_n)$ contains the feature values or number of semantic events for the states in S_n^g . Similarly, B_n is generically used to represent any member in B_n .
- $\Pi_n = \{\Pi_n^g\}$, where $1 \leq g \leq |S_n|$. Π_n^g includes the initial state probabilities for the states in S_n^g . Similarly, Π_n is generically used to represent any member in Π_n .

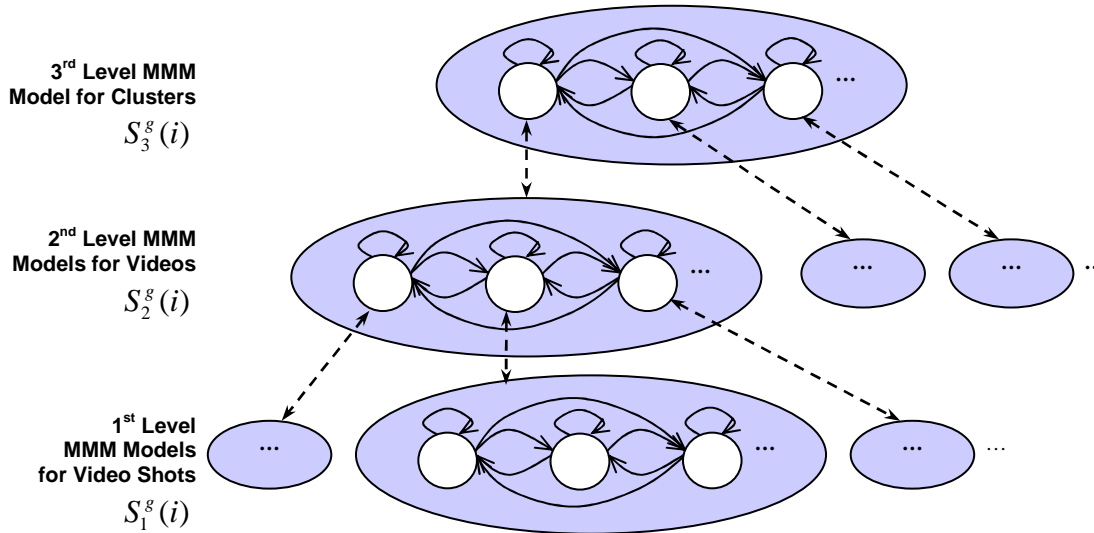


Figure IV-2. Three-level construction of Hierarchical Markov Model Mediator

In this proposed approach, we utilize a three-level HMMM model to manage the hierarchical video database. As demonstrated in Table IV-2, the MMM models of different levels

in the 3-level HMMM describe distinct objects and represent different meanings. Though the general description is the same, the matrices in different levels represent slightly dissimilar meanings to reflect the various natures of distinct multimedia objects. In the first level MMM ($d = 1$), the states represent the video shots, which are the elementary units in the video database to describe the continuous action between the start and the end of a camera operation. The feature set (F_1) consists of low-level or mid-level visual/audio features. In the second level MMM ($d = 2$), the states describe the set of videos in the database, and the feature set (F_2) contains the semantic events detected in the video collection. While in the third level MMM ($d = 3$), the states represent the set of video clusters.

Table IV-2. 3-level HMMM model

	1 st Level MMM	2 nd Level MMM	3 rd Level MMM
S	State set of video shots	State set of Videos	State set of video clusters
F	Low level visual/audio features	Semantic events (concepts)	-
A	Temporal based state transition probability between video shots	Affinity relationship between videos	Affinity relationship between video clusters
B	Formalized feature values	Annotated event numbers	-
Π	Initial state probability distribution for video shots	Initial state probability distribution for videos	Initial state probability distribution for video clusters

4.4 Two-level HMMM Model

In this section, the first two levels of the HMMM model are constructed in the beginning to model the source video and their associated video shots. More specifically, the fundamental level of the MMM model consists of a series of consecutive video shots. It needs to be noted that the events are referred to as shot-level video clips in this research. It is merely a choice of representation rather than a statement about the actual duration of a specific event. Thus, one local MMM is designed for the video shots in each video; while the second level MMM models

are constructed to model the videos in a cluster or database and thus they incorporate all the corresponding lower level MMM models.

4.4.1 Video shot level MMM

As we stated before, the matrices for affinity relationship, feature, and initial state probability distributions at different levels may hold slightly dissimilar meanings although the general depictions are the same. In the most fundamental level, the states (S_I) represent the video shots, which are the elementary units in the video database and describe the continuous action between the start and end of a camera operation. The feature set (F_I) for the video shot level MMM consists of low-level or mid-level visual/audio features.

4.4.1.1 A_1 : temporal-based relative affinity matrix

A_1 represents the temporal-based affinity relationship between the video shots in the video shot-level MMM. Let $S_1(i)$ and $S_1(j)$ (where $0 < i < j$) represent two specific video shots with certain semantic events, and if they are frequently accessed together in one temporal event pattern, they are said to have a higher affinity relationship. Hence, their temporal based affinity relationship value from $S_1(i)$ to $S_1(j)$ will be larger.

1) Initialization of A_1

Assume there are N video shots $\{s_1, s_2, \dots, s_N\}$ in video v , and all of these video shots follow the temporal sequence, i.e., $T_{s_1} < T_{s_2} < \dots < T_{s_N}$, where T_{s_i} is the occurrence time of video shot s_i . When searching for the temporal event pattern, the system will search for the video shots by following their temporal sequences. If the system goes from state s_i to state s_j , it must follow the rule of $T_{s_j} \geq T_{s_i}$. Therefore, for state s_i , there are $(N-i+1)$ possible states that the system will transit to. Accordingly, A_1 can be initialized as follows.

$$A_1(i, j) = \begin{cases} 1/(N - i + 1); & \text{where } 1 \leq i \leq N, 1 \leq j \leq N, j \geq i. \\ 0; & \text{where } j < i. \end{cases} \quad (\text{IV-1})$$

2) Update of A_1

By adopting HMMM, users are allowed to provide their feedback to the system. The video shot sequences similar to the anticipated temporal event pattern will be marked as “Positive” patterns which are used to capture the user preferences to refine the system retrieval capability for the future update. A matrix AF_1 is defined to capture the temporal-based affinity relationships among all the annotated video shots using user access patterns and access frequencies. For the y^{th} pattern R_y , $access_1(y)$ represents its access frequencies, and $use_1(i, y)$ equals 1 if s_i (the i^{th} video shot) was accessed in the y^{th} pattern R_y . Moreover, both s_m and s_n should belong to this “Positive” temporal pattern, and s_m should occur before s_n or they should occur at the same time. Let q be the number of positive patterns on the shot level, AF_1 can be calculated as below:

$$aff_1(m, n) = A_1(m, n) \times \sum_{y=1}^q use_1(m, y) \times use_1(n, y) \times access_1(y), \quad (\text{IV-2})$$

where $s_m \in R_y, s_n \in R_y, T_{s_m} \leq T_{s_n}$.

Each entry of $aff_1(m, n)$ in AF_1 indicates the frequency of s_m and s_n being accessed together in the first level MMM, and consequently the probability of these two video shots being accessed together in the temporal patterns. A_1 can then be updated via normalizing AF_1 per row and thus MMM represents the relative affinity relationships among all the video clips in the database. Let $A_1(m, n)$ be the element in the (m, n) entry in the first level MMM, then

$$A_1(m, n) = \frac{aff_1(m, n)}{\sum_{j=1}^N aff_1(m, j)}. \quad (\text{IV-3})$$

For the sake of efficiency, the training system can only record all the user access patterns and access frequencies during a training period, instead of updating A_1 matrix on-line every time. Once the number of newly achieved feedbacks reaches a certain threshold, the update of A_1 matrix can be triggered automatically. All the computations should be done offline.

Table IV-3. Feature list for the video shots

Category	Feature Name	Feature Description
Visual Features	<i>grass_ratio</i>	Average percent of grass areas in a shot
	<i>pixel_change_percent</i>	Average percent of the changed pixels between frames within a shot
	<i>histo_change</i>	Mean value of the histogram difference between frames within a shot
	<i>background_var</i>	Mean value of the variance of background pixels
	<i>background_mean</i>	Mean value of the background pixels
Audio Features	<i>volume_mean</i>	Mean value of the volume
	<i>volume_std</i>	Standard deviation of the volume, normalized by the maximum volume
	<i>volume_stddev</i>	Standard deviation of the difference of the volume
	<i>volume_range</i>	Dynamic range of the volume, defined as $(\max(v)-\min(v))/\max(v)$
	<i>energy_mean</i>	Mean RMS energy
	<i>sub1_mean</i>	Average RMS energy of the first sub-band
	<i>sub3_mean</i>	Average RMS energy of the third sub-band
	<i>energy_lowrate</i>	Percentage of samples with RMS power less than 0.5 times the mean RMS power
	<i>sub1_lowrate</i>	Percentage of samples with RMS power less than 0.5 times the mean RMS power of the first sub-band
	<i>sub3_lowrate</i>	Percentage of samples with RMS power less than 0.5 times the mean RMS power of the third sub-band
	<i>sub1_std</i>	Standard deviation of the mean RMS power of the first sub-band energy
	<i>sf_mean</i>	Mean value of the Spectrum Flux
	<i>sf_std</i>	Standard deviation of the Spectrum Flux, normalized by the maximum Spectrum Flux
	<i>sf_stddev</i>	Standard deviation of the difference of the Spectrum Flux, which is normalized too
<i>sf_range</i>	Dynamic range of the Spectrum Flux.	

4.4.1.2 B_1 : visual/audio feature matrix

We consider both the visual and audio features in the feature matrix B_1 for the video shot level MMM constructions. As shown in Table IV-3, there are a total of 5 visual and 15 audio features [ChenSC03a].

1) Normalization of B_1

The initial values of the features need to be normalized to achieve more accurate similarity measures. To capture the original value of a feature in a video shot, we define a temporal matrix BB_1 whose rows represent the distinct video shots while the columns denote all the distinct features. The entry of $BB_1(i, k)$ denotes the original value of the k^{th} feature of the i^{th} video shot, where $1 \leq k \leq K$, K is number of features and $1 \leq i \leq N$, N is the number of video shots. Our target is to normalize all of the features to fall between $[0, 1]$:

$$B_1(i, k) = \frac{BB_1(i, k) - \min_{j=1}^N(BB_1(j, k))}{\max_{j=1}^N(BB_1(j, k)) - \min_{j=1}^N(BB_1(j, k))}, \text{ where } 1 \leq i \leq N, 1 \leq k \leq K. \quad (\text{IV-4})$$

4.4.1.3 Π_1 : initial state probability matrix for shots

The preference of the initial states for queries can be achieved from the training data set. For any video shot state $s_m \in S_1$, the initial state probability is defined as the fraction of the number of occurrences of video shot s_m as the initial state can traverse with respect to the total number of occurrences for all the initially traversed video shot states in the video database from the training data set. The Π_1 can thus be constructed as below, where π_m is defined as the initial state probability for video shot s_m .

$$\Pi_1 = \{\pi_m\} = \frac{\sum_{y=1}^N use_1(m, y)}{\sum_{l \in S_1} \sum_{y=1}^N use_1(l, y)}. \quad (\text{IV-5})$$

4.4.2 Video-level MMM

The purpose of constructing video-level MMM is to cluster the videos describing similar events. A large video archive may contain various kinds of videos, such as news videos, movies, advertisement videos, and sports videos. The second level MMM is constructed such that the system is able to learn the semantic concepts and then cluster the videos into different categories.

4.4.2.1 A_2 : relative affinitive matrix for videos

Based on the information contained in the training data set, the affinity relationships among the video sets in the database can be captured, i.e., the higher the frequency of two videos being accessed together, the closer they are related to each other. The relative affinity matrix A_2 is constructed in two steps as follows:

First, a matrix AF_2 is defined to capture the affinity measures among all the videos by using user access patterns and access frequencies. After that, each entry $aff_2(m, n)$ in AF_2 indicates the frequency of the two videos v_m and v_n being accessed together in the 2nd level MMM, and consequently how closely these two videos are related to each other. Let q' be the number of queries on the video level.

$$aff_2(m, n) = \sum_{y=1}^{q'} use_2(m, y) \times use_2(n, y) \times access_2(y). \quad (IV-6)$$

The matrix A_2 can then be obtained via normalizing AF_2 per row and thus represents the relative affinity relationships among all the M videos in the database (D).

$$A_2(m, n) = \frac{aff_2(m, n)}{\sum_{j=1}^M aff_2(m, j)}, \text{ where } 1 \leq m \leq M \text{ and } 1 \leq n \leq M. \quad (IV-7)$$

Please note that A_1 and A_2 are different since A_1 considers the temporal relationships as well, while A_2 does not.

4.4.2.2 B_2 : event number matrix for videos

Matrix B_2 includes the event numbers of each video, where each row represents a video and each column denotes one semantic event. Assume there are a total of M videos in the database, where the video v_i ($1 \leq i \leq M$) contains the set of C events denoted as $\{e_1, e_2, \dots, e_C\}$, and $B_2(i, j)$ means the number of the j^{th} event (e_j) in v_i . B_2 does not need to be normalized and the integer values are kept.

4.4.2.3 Π_2 : initial state probability matrix for videos

In the video-level, the access patterns and access frequencies for videos in use_2 (instead of use_1) are used to construct the matrix Π_2 .

4.4.3 Connections between first level MMMs and second level MMM

4.4.3.1 $O_{1,2}$: weight importance matrix

The weight importance matrix ($O_{1,2}$) is required to denote the relationship between the features for video shots and the specific semantic events. This matrix is utilized to adjust the characteristic influences by learning the features of the annotated events. In $O_{1,2}$, each row represents an event concept, while each column represents a feature. The value in $O_{1,2}$ means the weight of importance of the corresponding feature for the specific event concept.

1) Initialization of $O_{1,2}$

Let each multimedia object have K features $\{f_1, f_2, \dots, f_K\}$ and C events $\{e_1, e_2, \dots, e_C\}$. We define the initial value for each feature in an event concept to be $1/K$, which means they carry the same weight importance.

$$O_{1,2}(i, j) = \frac{1}{K}, \text{ where } 1 \leq i \leq C, 1 \leq j \leq K. \quad (\text{IV-8})$$

2) Update of $O_{1,2}$

Once a group of N video shots $\{s_1, s_2, \dots, s_N\}$ consisting of the same event concept e_i ($1 \leq i \leq C$) are known, the standard deviations of the K features for all the N video shots can be

calculated as $\{Std_{i,1}, Std_{i,2}, \dots, Std_{i,K}\}$, where $Std_{i,k}$ represents the standard deviation of the i^{th} event and k^{th} feature ($1 \leq i \leq C, 1 \leq k \leq K$). Equations (IV-9)-(IV-11) can be employed to compute $O_{1,2}$. The larger the $O_{1,2}$ value is, the more important this feature is when calculating the similarity score with the specified event.

$$O'(i,k) = \frac{1}{Std_{i,k}}, \text{ where } 1 \leq i \leq C, 1 \leq k \leq K. \quad (\text{IV-9})$$

$$O_{1,2}(i,k) = \frac{O'(i,k)}{\sum_{k=1}^K O'(i,k)}; \text{ and} \quad (\text{IV-10})$$

$$O_{1,2}(i,k) = \left(\frac{1}{Std_{i,k}} \right) / \left(\sum_{k=1}^K \frac{1}{Std_{i,k}} \right). \quad (\text{IV-11})$$

4.4.3.2 B_1' : mean value of the features per event

In matrix B_1' , the row represents an event (concept), and the column denotes the visual and audio features. Assume that for the event e_i ($1 \leq i \leq C$), a set of N video shots $\{s_1, s_2, \dots, s_N\}$ are identified as e_i , where these video shots are not necessarily consecutive shots. Let $B_1(s_j, f_k)$ represent the normalized value for video shot s_j and feature f_k , the mean value of the features f_k ($1 \leq k \leq K$) for e_i can be calculated as follows.

$$B_1'(e_i, f_k) = \frac{\sum_{j=1}^N B_1(s_j, f_k)}{N}, \text{ where } 1 \leq i \leq C, 1 \leq k \leq K. \quad (\text{IV-12})$$

4.4.3.3 $L_{1,2}$: link conditions matrix

To facilitate the connections between the local MMM model and the second level MMM model, the link conditions matrix $L_{1,2}$ is designed. Let $\{v_1, v_2, \dots, v_M\}$ be the M videos and $\{s_1, s_2, \dots, s_N\}$ be the N video shots, if s_j belongs to v_i , $L_{1,2}(v_i, s_j) = 1$ (where $1 \leq i \leq M, 1 \leq j \leq N$). Otherwise, $L_{1,2}(v_i, s_j) = 0$.

4.4.4 Initial Process for Temporal Event Pattern Retrieval

Given a temporal pattern with C events $Q = \{e_1, e_2, \dots, e_C\}$ sorted by the temporal relationships such that $T_{e_1} \leq T_{e_2} \leq \dots \leq T_{e_C}$, the initial retrieval process is presented as below. Here, we assume there are M videos $\{v_1, \dots, v_M\}$ in the multimedia database archive, and there are total K non-zero features $\{f_1, f_2, \dots, f_K\}$ of the query sample. Here, $1 \leq K \leq 20$ since 20 features are used. Without any online feedback or video clusters, the initial retrieval process includes the following steps.

- **Step 1.** Initializes the flag parameters as $i=1$, $t=1$, and $y=1$.
- **Step 2.** Checks matrix B_2 and/or matrix A_2 to search for video v_i which contains event e_t . This video should have a close affinity relationship with the previous video if it is available.
- **Step 3.** Checks the link condition matrix $L_{1,2}$ and/or matrix A_1 to find the specified video shot s_t which is annotated as event e_t or similar to event e_t . This video shot should also have a strong connection to the previous video shot.
- **Step 4.** Calculates the edge weight $w_t(s_t, e_t)$ using Equations (IV-13) and (IV-14), which is defined as the edge weight from the current state s_t to the target event e_t at the evaluation of the k^{th} feature (f_k) in the query, where $1 \leq k \leq K$ and $1 \leq t \leq C$.

$$\text{At } t=1, w_1(s_1, e_1) = \Pi_1(s_1) \times \text{sim}(s_1, e_1). \quad (\text{IV-13})$$

When $1 \leq t < C$:

$$w_{t+1}(s_{t+1}, e_{t+1}) = w_t(s_t, e_t) \times A_1(s_t, s_{t+1}) \times \text{sim}(s_{t+1}, e_{t+1}). \quad (\text{IV-14})$$

Equation (IV-15) defines the similarity function to measure the similarity between s_t and e_t based on all of the non-zero features in $\{f_1, f_2, \dots, f_K\}$.

$$sim(s_t, e_t) = \sum_{k=1}^K (O_{1,2}(e_t, f_k) \times \frac{(1 - |B_1(s_t, f_k) - B_1'(e_t, f_k)|)}{B_1'(e_t, f_k)}), \quad (IV-15)$$

where $s_t \in S_1, 1 \leq k \leq K, 1 \leq t \leq C$.

In each traversal, the system will choose the optimized path to access the next possible video shot states similar to the anticipated events. At the end of one video, the next possible video candidate will be selected by checking the higher-level affinity and feature matrices.

- **Step 5.** $t = t + 1$. If $t > C$, all the events in this pattern have been traversed and therefore the similarity score of the whole candidate pattern should be computed as indicated in Step 6. Otherwise, the system goes to Step 3 to continue checking the next video shot candidate which most closely matches the next event. Note that the traversal path should be recorded in the whole process.
- **Step 6.** Assumes a candidate video shot sequence is defined as $R_y = \{s_1, s_2, \dots, s_C\}$, the final similarity score can be calculated as:

$$SS(Q, R_y) = \sum_{t=1}^C w_t(s_t, e_t). \quad (IV-16)$$

- **Step 7.** $i = i + 1; y = y + 1$. Checks if $i > M$. If yes, all the candidate video sets are checked and the system goes to Step 8. If no, the system goes to Step 2 and checks matrices A_2 and B_2 to find the next video candidate.
- **Step 8.** There are $y - 1$ candidate patterns. The system ranks the candidate video shot sequences according to the similarity scores.
- **Step 9.** Finally, a list of $y - 1$ sorted video shot sequences is retrieved as the output.

Free Kick & Goal → Corner Kick → Player Change → Goal



Figure IV-3. An example result of a temporal pattern query



Figure IV-4. HMMM-based soccer video retrieval interface

As illustrated in Figure IV-3, the key frames of a set of retrieved temporal event patterns are displayed below the temporal pattern query to show an example of the results.

A soccer video retrieval system has been developed for the evaluation of the proposed approach. In the current approach, the proposed HMMM mechanism is utilized to model the multimedia database. Two levels of MMM models are constructed to model 54 soccer videos which are segmented into 11,567 video shots. Among these video shots, 506 of them are annotated as semantic events. Figure IV-4 shows the client-side interface of the system, where the left-bottom part shows the interactive panels where a user can issue the queries. The right side panel demonstrates the resulting patterns sorted by their similarity scores. In this case, the target pattern is issued with a goal shot followed by a free kick, and therefore 8 patterns (including 16 shots) are displayed, where the magenta box marked the 3rd pattern. The left-upper panel displays the video shot which is chosen by the user. Finally, by using the drop down menu below the key frames, users are able to select their preferred video shots/patterns, and their feedback can be sent back to the server-side for further improvement of the retrieval performance.

4.5 Video Database Clustering and Construction of 3rd Level MMM

In this section, an integrated and interactive video retrieval framework is proposed to efficiently organize, model, and retrieve the content of a large scale multimedia database. The core of our proposed framework is a learning mechanism called HMMM (Hierarchical Markov Model Mediator) [Zhao06a] and an innovative video clustering strategy [Zhao06b]. HMMM models the video database, while the clustering strategy groups video data with similar characteristics into clusters that exhibit certain high level semantics. The HMMM structure is then extended by adding an additional level to represent the clusters and their relationships.

The proposed framework is designed to accommodate advanced queries via considering the high level semantic meaning. First, it is capable of searching semantic events or event patterns considering their popularity by evaluating their access frequencies in the large number of historical queries. Second, the users can choose one or more example patterns with their anticipated features from the initial retrieved results, and then issue the next round of queries. It can search and re-rank the candidate patterns which involve similar aspects with the positive examples reflecting the user's interests. Third, video clustering can be conducted to further reduce the searching time especially when dealing with the top- k similarity retrievals. As the HMMM mechanism helps to traverse the most optimized path to perform the retrieval, the proposed framework can only search several clusters for the candidate results without traversing all the paths to check the whole database.

4.5.1 Overall Workflow

Figure IV-5 demonstrates the overall workflow of the proposed framework. In this framework, the soccer videos are first segmented into distinct video shots and their low-level video/audio features are extracted. A multimedia data mining approach is utilized to pre-process the video shots to get an initial candidate pool for the potential important events. After that, a set of initial event labels will be given to some of the shots, where not all of these labels are correct.

All the data and information will be fed into this framework for event pattern searching and video retrieval purposes. The videos included in the candidate pool are modeled in the 1st level of MMM (Markov Model Mediator) models, whereas all the videos are modeled in the 2nd level. After initializing the 1st level and 2nd level of the MMM models, users are allowed to issue event or event pattern queries. Furthermore, users can select their event patterns of interest in the initial results and re-issue the query to refine the retrieval results and their rankings. This step is also recognized as online learning. These user-selected shot sequences are stored as positive patterns for future offline training.

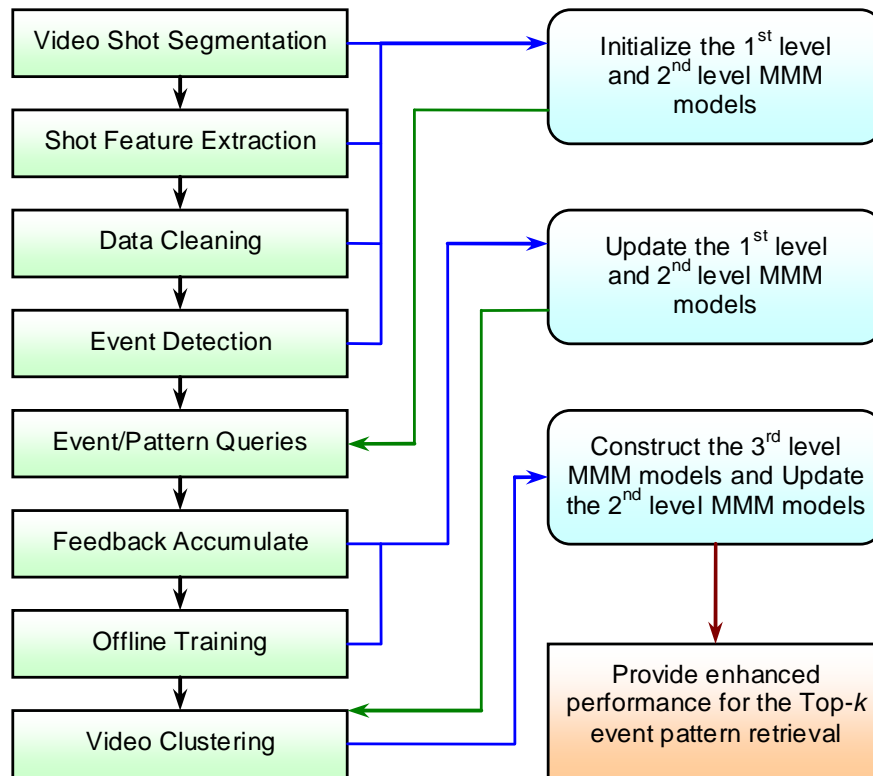


Figure IV-5. Overall workflow for the proposed approach

After a certain number of queries and feedback, the proposed framework is able to perform the offline training. The historical queries and user access records are utilized to update the affinity relationships of the videos/video shots as well as their initial state probabilities. Thereafter, both the semantic events and the high level user perceptions are employed to

construct the video clusters, which are then modeled by a higher level (3rd level) of the MMM model. In the meanwhile, the 2nd level MMM models are divided into a set of sub-models based on the clustered video groups.

The clustered database and the updated HMMM mechanism are capable of providing appealing multimedia experience to the users because the modeled multimedia database system learns the user's preferences and interests interactively by reusing the historical queries .

4.5.2 Conceptual Video Clustering

4.5.2.1 Similarity Measurement

In this proposed framework, a video is treated as an individual database in a distributed multimedia database system, where its video shots are the data instances in the database. Accordingly, a similarity measure between two videos is defined as a value indicating the likeness of these two videos with respect to their conceptual contents. It is calculated by evaluating their positive events and event patterns in the historical queries. If two videos consist of the same event(s) and/or event pattern(s) and are accessed together frequently, they are considered closely related and their similarity score should be high.

Assume there are H user queries issued through the video retrieval framework, where the set of all the query patterns is denoted as QS . In order to refine their retrieved results in real-time, the users mark their preferred event patterns as “positive” before making the next query. By evaluating the issued query sets and their associated positive patterns, the similarity measure is defined as follows.

Let v_i and v_j be two videos, and $X=\{x_1, \dots, x_m\}$ and $Y=\{y_1, \dots, y_n\}$ be the sets of video shots belonging to v_i and v_j ($X \subseteq v_i, Y \subseteq v_j$), where m and n are the numbers of annotated video shots in v_i and v_j .

Denote a query with an observation sequence (semantic event pattern) with C semantic events as $Q^k = \{e_1^k, e_2^k, \dots, e_C^k\}$, where $Q^k \in QS$. Let R^k be the set of G positive patterns that a user has selected from the initial retrieval results for query Q^k . This can be represented by a matrix of size $G \times C$, $G \geq 1$, $C \geq 1$. As shown in Equation (IV-17), each row of R^k represents an event shot sequence that the user marked as positive, and each column includes the candidate event shots which correspond to the requested event in the query pattern.

$$R^k = \begin{Bmatrix} \{s_1^1, s_2^1, \dots, s_C^1\} \\ \{s_1^2, s_2^2, \dots, s_C^2\} \\ \dots \\ \{s_1^G, s_2^G, \dots, s_C^G\} \end{Bmatrix}. \quad (\text{IV-17})$$

Based on the above assumptions, the video similarity function is defined as below.

Definition IV-3: $SV(v_i, v_j)$, the similarity measure between two videos, is defined by evaluating the probabilities of finding the same event pattern Q^k from v_i and v_j in the same query for all the query patterns in QS .

$$SV(v_i, v_j) = \left(\sum_{Q^k \in QS} P(Q^k | v_i) P(Q^k | v_j) \right) \times FA(H). \quad (\text{IV-18})$$

where $1 \leq k \leq H$, and $FA(H)$ is an adjusting factor. $P(Q^k | v_i)$ and $P(Q^k | v_j)$ represent the occurrence probabilities of finding Q^k from v_i and v_j , where the occurrence probability can be obtained by summing the joint probabilities over all the possible states [Rabiner93]. In order to calculate this value, we need to select all the subsets with C event shots from the positive pattern set R^k , which also belong to v_i or v_j . That is, $X' = \{x_1', x_2', \dots, x_C'\}$ and $Y' = \{y_1', y_2', \dots, y_C'\}$, where $X' \subseteq X$, $X' \in R^k$, $Y' \subseteq Y$, $Y' \in R^k$. If these patterns do not exist, then the probability value is set as 0 automatically.

$$P(Q^k | v_i) = \sum_{\text{all } X'} P(Q^k, X' | v_i) = \sum_{\text{all } X'} P(Q^k | X', v_i) P(X' | v_i). \quad (\text{IV-19})$$

Assume the statistical independence of the observations, and given the state sequence of $X' = \{x_1', x_2', \dots, x_{C'}\}$, Equation (IV-20) gives the probability of X' given v_i .

$$P(X' | v_i) = \prod_{t=1}^{C-1} P(x_t' | x_{t+1}') P(x_1') = \prod_{t=1}^{C-1} A_1(x_t', x_{t+1}') \Pi_1(x_1'). \quad (\text{IV-20})$$

Here, $P(x_t' | x_{t+1}')$ represents the probability of retrieving a video shot x_{t+1}' given that the current video shot is x_t' . It corresponds to the $A_1(x_t', x_{t+1}')$ entry in the relationship matrix. $P(x_1')$ is the initial probability for video shot x_1' , i.e., $\Pi_1(x_1')$. Equation (IV-21) gives the probability of an observation sequence (semantic event pattern) Q^k .

$$P(Q^k | X', v_i) = \prod_{t=1}^C P(e_t^k | x_t'). \quad (\text{IV-21})$$

where $P(e_t^k | x_t')$ indicates the probability of observing a semantic event e_t^k from a video shot x_t' . This value is computed by using a similarity measure by considering low-level and mid-level features. However, in this approach, since the users have already marked these video shots as the events they requested and preferred, the probability of observing the semantic events is simply treated as 1.

4.5.2.2 Clustering Strategy

Considering a large scale video database, it is a significant issue to cluster similar videos together to speed up the similarity search. As we stated before, a two-level HMMM has been constructed to model video and video shots. Furthermore, a video database clustering strategy which is traversal-based and greedy is proposed.

As illustrated in Figure IV-6, the proposed video database clustering technique contains the following steps. Given the video database D with M videos and the maximum size of the video database cluster as Z ($Z \geq 2$), the mechanism:

- a) Initializes the parameters as $p=0$; $n=0$, where p denotes the number of videos being clustered, and n represents the cluster number.
- b) Sets $n = n + 1$. Searches the current video database D for the video v_i with the largest stationary probability $\Pi_2(v_i)$, and then starts a new cluster CC_n with this video ($CC_n = \{ \}$; $CC_n \leftarrow CC_n \cup \{v_i\}$). Initializes the parameter as $q=1$, where q represents the number of videos in the current cluster.
- c) Removes v_i from database D ($D \leftarrow D - \{v_i\}$). Checks if $p = M$. If yes, output the clusters. If no, goes to step d).
- d) Searches for v_j , which has the largest $A_2(v_i, v_j) \times SV(v_i, v_j)$ in D . Adds v_j to the current cluster CC_n ($CC_n \leftarrow CC_n \cup \{v_j\}$).
- e) $v_i \leftarrow v_j$, where v_i represents the most recent clustered video. Every time when a video is assigned to a cluster, it is automatically removed from D ($D \leftarrow D - \{v_i\}$).
- f) $p++$ and $q++$. Checks if $p=M$. If yes, outputs the clustering results. If no, checks if $q=Z$. If yes, goes to step b) to start a new cluster. If no, goes to step d) to add another video in the current cluster.
- g) If there is no un-clustered video left in the current database, it outputs the current clusters.

4.5.3 Constructing the 3rd level MMM model

In this research, the HMMM model is extended by the 3rd level MMM to improve the overall retrieval performance. In the 3rd level MMM ($d = 3$), the states (S_3) denote the video clusters. Matrix A_3 describes the relationships between each pair of clusters.

Definition IV-4: Assume CC_m and CC_n are two video clusters in the video database D . Their relationship is denoted as an entry in the affinity matrix A_3 , which can be computed by Equations (IV-22) and (IV-23). Here, SC is the function that calculates the similarity score between two video clusters.

$$SC(CC_m, CC_n) = \frac{\sum_{v_i \in CC_m} (\Pi_2(v_i) \times \max_{v_j \in CC_n} (A_2(v_i, v_j) \times SV(v_i, v_j)))}{M}, \quad (IV-22)$$

where $CC_m \in D, CC_n \in D$.

$$A_3(CC_m, CC_n) = \frac{SC(CC_m, CC_n)}{\sum_{CC_j \in D} SC(CC_m, CC_j)}. \quad (IV-23)$$

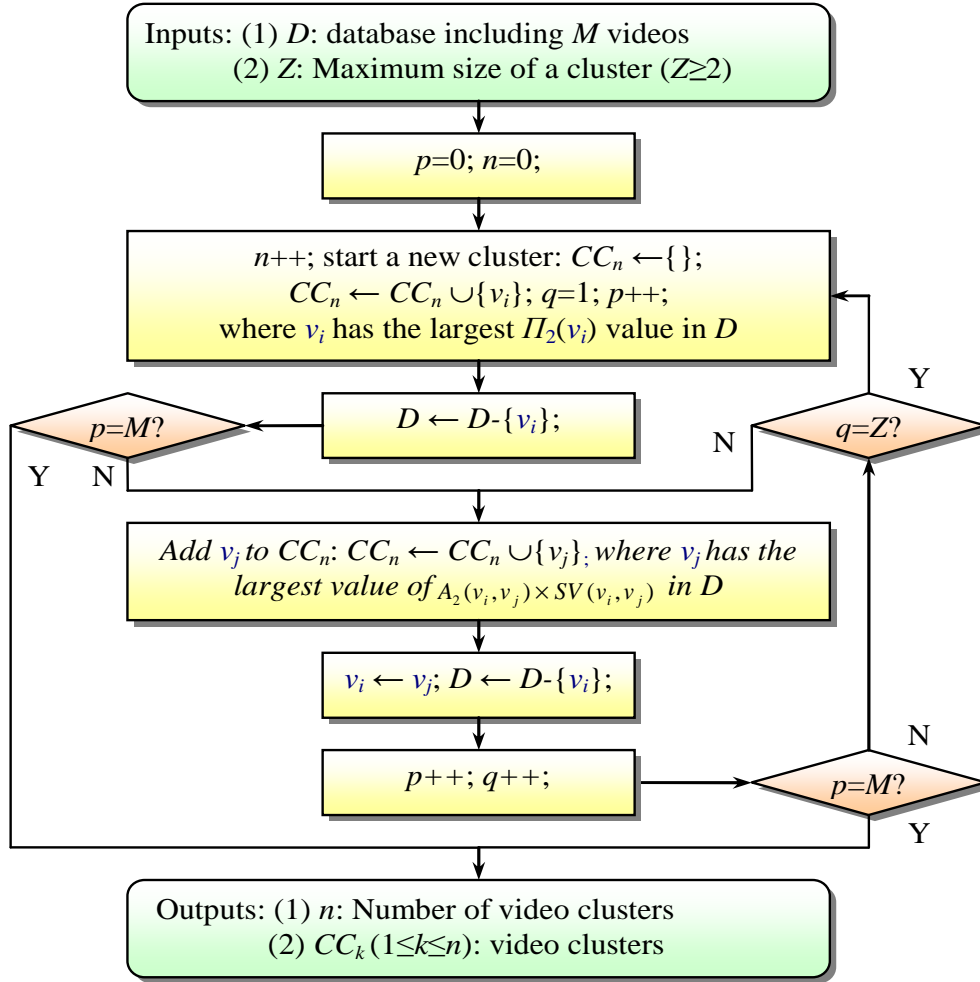


Figure IV-6. The proposed conceptual video database clustering procedure

The matrix Π_3 can be constructed to represent the initial state probability of the clusters. The calculation of Π_3 is similar to the ones for Π_1 and Π_2 . In addition, matrix $L_{2,3}$ can also be constructed to illustrate the link conditions between the 2nd level MMMs and the 3rd level MMM.

4.5.4 Interactive Retrieval through Clustered Video Database

Given an example shot sequence $Q' = \{s_1, s_2, \dots, s_C\}$ which represents the event pattern as $\{e_1, e_2, \dots, e_C\}$ such that s_i describes e_i ($1 \leq i \leq C$), and they follow the temporal sequence as $T_{s_1} \leq T_{s_2} \leq \dots \leq T_{s_C}$. Assume that a user wants to find top- k related shot sequences which follow similar patterns. In our proposed retrieval algorithm, a recursive process is conducted to traverse the HMMM database model and find the top k candidate results. As shown in Figure IV-7, a lattice-based structure for the overall video database can be constructed. Assume the transitions are sorted based on their edge weights [Zhao06a], and the retrieval algorithm will traverse the edge with a higher weight each time. For example, in Figure IV-7, we assume that the edge weights satisfy $w(s_1, s_2) \geq w(s_1, s_4) \geq w(s_1, s_7)$. The algorithm can be described as below.

1. Searches for the first candidate cluster, first candidate video and first candidate video shot by checking matrices Π_3, Π_2, B_2, Π_1 and B_1 .
2. If the pattern is not complete, continues search for the next event (video shot) via computing the edge weights by checking A_1 .
3. If the candidate pattern has been completed, goes back state by state and checks for other possible paths. Also checks if there are already k candidate patterns being retrieved. If yes, stops searching and goes to Step 6.
4. If there are no more possibilities in the current video, then marks this video with a “searched” flag and check A_2 and B_2 to find the next candidate video.
5. If all the videos are “searched” in the current cluster, then marks the current cluster as “searched” cluster and check A_3 to find the next candidate video cluster.

6. Once k patterns are retrieved, or there are no more possibilities in the database, ranks the candidate patterns via calculating the similarity scores [Zhao06a] and outputs the candidate patterns.

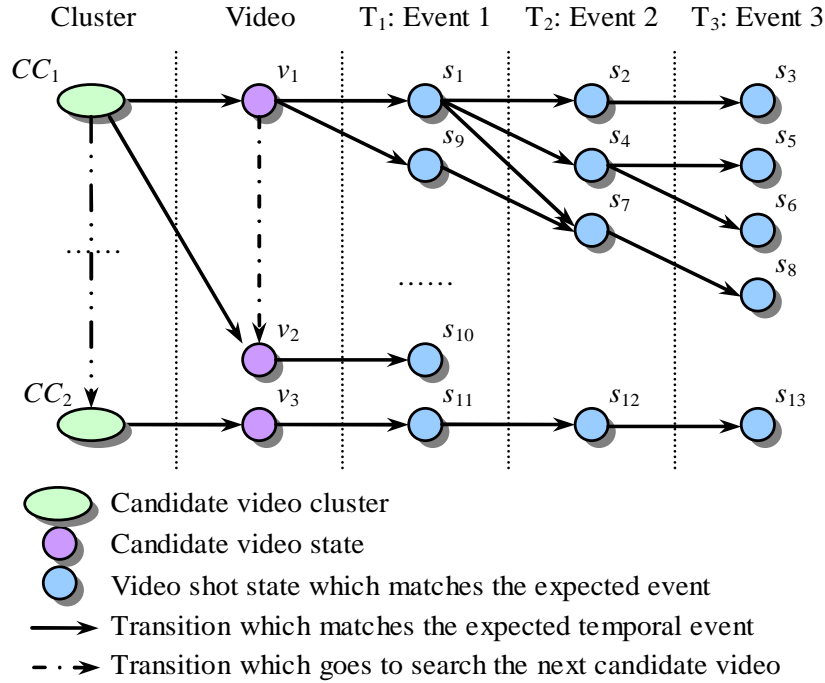


Figure IV-7. Lattice structure of the clustered video database

R	Cluster	Video	Event 1	Event 2	Event 3
1	CC_1	v_1	s_1	s_2	s_3
2	CC_1	v_1	s_1	s_4	s_5
3	CC_1	v_1	s_1	s_4	s_6
4	CC_1	v_1	s_1	s_7	s_8
5	CC_1	v_1	s_9	s_7	s_8
6	CC_2	v_3	s_{11}	s_{12}	s_{13}

Figure IV-8. Result patterns and the traverse path

As shown in the Figure IV-8, the yellow cells include the paths the algorithm has traversed. Furthermore, we designed a function to fill in the missed cells by copying the correspondent shots in the previous candidate patterns. Finally, six complete candidate patterns are generated. Once k candidate patterns are generated, the system does not need to traverse any

other clusters or videos. Therefore, it significantly reduces the searching spaces and accelerates the searching speed.

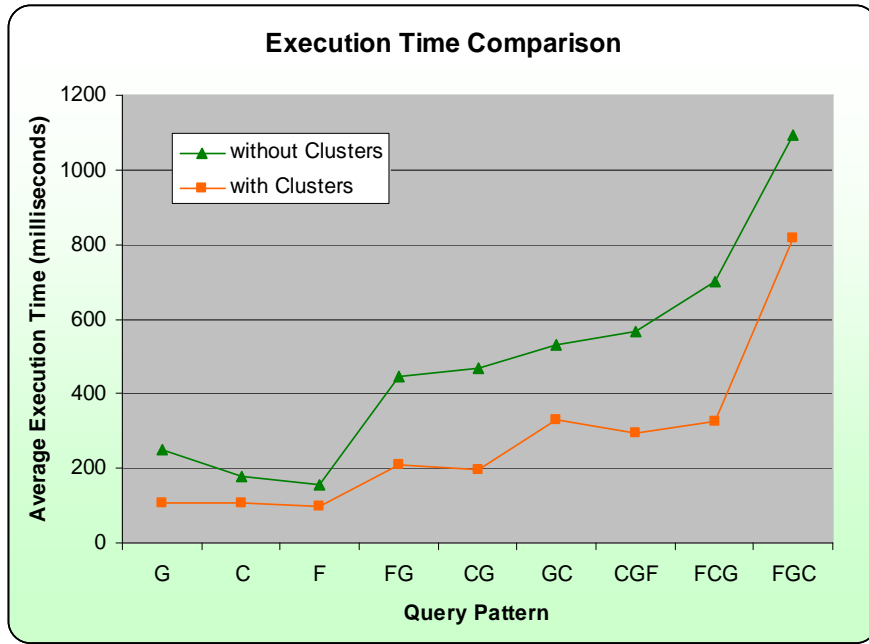
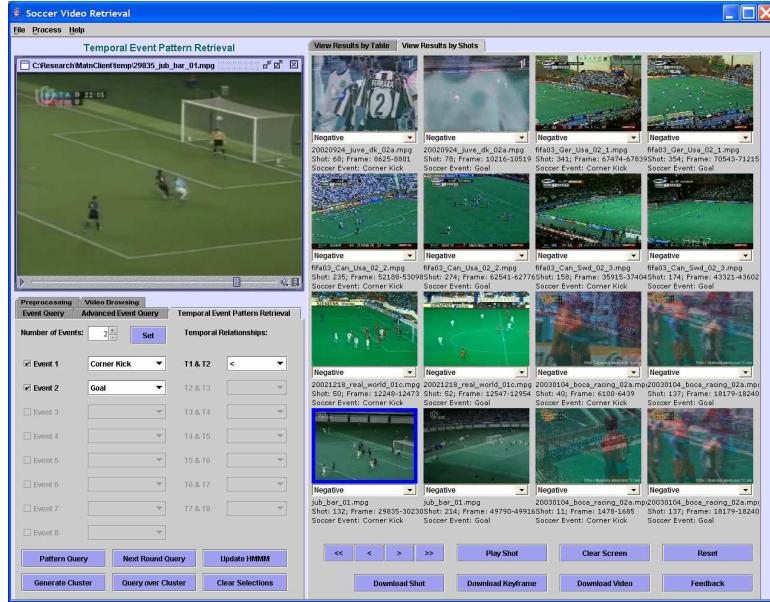


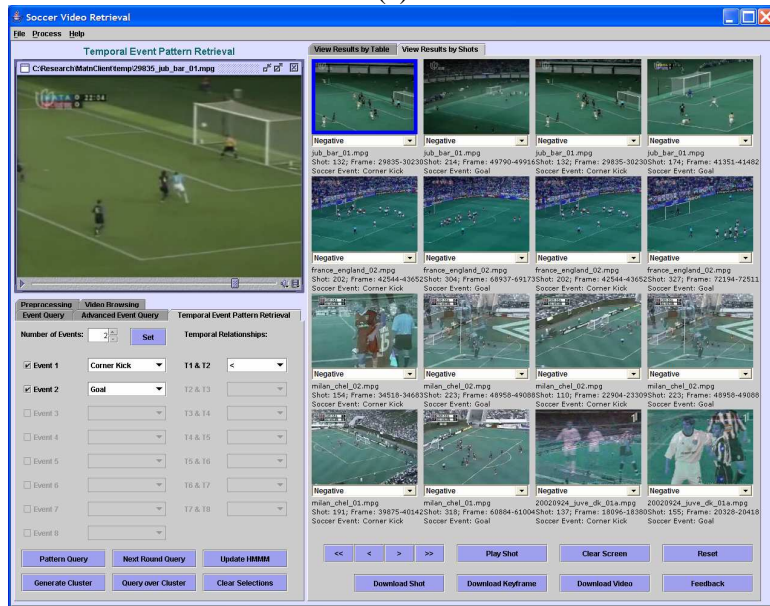
Figure IV-9. Comparison of the average execution time

4.5.5 Experimental Results for Video Clustering

We have built up a soccer video database with 45 videos which contain 8977 video shots. A retrieval system has also been implemented for the system training and experimental tests. Totally, 150 sets of historical queries were issued and user feedbacks were returned with their preferred patterns, which cover all of the 45 videos and 259 distinct video shots. In the clustering process, we defined the cluster size as 10 and the expected result pattern number as $k=60$. As shown in Figure IV-9, we use letters “G”, “F”, and “C” to represent “Goal”, “Free kick”, “Corner kick” events, respectively. Therefore, the x-axis represents different query patterns, e.g., “G” means a query to search for “Goal” Events; “FG” means a query to search for the event pattern where a “Free kick” followed by a “Goal”; and “CGF” means a query pattern of a “Corner kick” event, followed by a “Goal” and then a “Free kick”, etc. For each query pattern, we issued 10 queries to compute the average execution time in milliseconds.



(a)



(b)

Figure IV-10. Soccer video retrieval system interfaces (a) query over non-clustered soccer video database (b) query over clustered soccer video database

As illustrated in Figure IV-9, the query patterns with fewer event numbers will be executed in less time as expected. In addition, the execution time of the system with clusters is less than that of the system without clusters, indicating that our proposed approach effectively groups relevant videos in the video clusters so that only the relevant clusters and their member

videos will need to be searched. Therefore, the searching space is dramatically decreased, and the execution of the queries becomes faster.

For the query pattern (“Corner kick” followed by a “Goal”), Figure IV-10(a) demonstrates the first screen of retrieval results over the non-clustered soccer video database; while Figure IV-10(b) shows the query results over the clustered soccer video database. It can be clearly seen that the query results in the same cluster represent the similar visual clues, which are mined from the historical queries/feedbacks, and correspondingly represent user preferences.

4.6 Conclusions

In this chapter, an HMMM-based multimedia data modeling mechanism is proposed to develop a user-interactive multimedia retrieval framework. User feedbacks are adopted in this mechanism to perform both hierarchical learning and conceptual-based video clustering. Specifically, the definition of HMMM is formalized in this chapter, while the construction and basic learning methods are also given for each of the three levels in HMMM. Further, several sets of retrieval procedures and ranking algorithms are designed and presented in detail to meet different conditions of video database, i.e., database before clustering, clustered database, etc. A soccer video retrieval system is developed and employed for the experimental tests in the different stages of the whole process. The results show that our proposed approach helps accelerate the retrieval speed while providing decent retrieval results.

The major contributions of this proposed research include the following aspects. First, the HMMM mechanism offers a hierarchical structure to assist the proficient construction of a high-dimensional multimedia database. It also helps to bridge the semantic gap between the concept-based and the content-based retrieval approaches to the comprehensive multimedia database modeling. The temporal relationship between the semantic events is naturally incorporated in HMMM such that complicated temporal pattern queries can be executed. Second, this framework integrates the feedback and learning strategies by considering not only the low-level visual/audio

features, but also the high-level semantic information and user preferences. In addition, the proposed framework is designed to accommodate advanced queries via considering the high level semantic meaning. Finally, the video clustering can be conducted to further reduce the searching time especially when dealing with the top- k similarity retrievals. As the HMMM mechanism helps to traverse the most optimized path to perform the retrieval, the proposed framework can only search several clusters for the candidate results without traversing all the paths to check the whole database. Hence, more accurate patterns can be retrieved quickly with lower computational costs.

It is worth mentioning that this approach supports not only offline training, but also online learning. In this chapter, we only introduce the basic offline training method, which tries to updated A and H matrices based on a large number of historical queries and feedbacks from multiple users. The overall retrieval performance can be refined continuously to gain long term benefits. However, this method has its own disadvantages: this offline training method lacks efficiency, cannot meet individual user preferences, and needs a manual process to trigger. In fact, this research framework can support more powerful online learning methods and the offline training method can also be enhanced. Further details will be investigated and discussed in Chapter V.

CHAPTER V. MULTIMEDIA SYSTEM TRAINING AND LEARNING

Semantic retrieval of media objects may well extend textual consequents to include all forms of multimedia. However, it is a challenging task for a multimedia system to perform content-based retrieval on multi-dimensional audio/visual data, and it is even harder to refine the retrieval results iteratively and interactively based on user preferences.

This chapter mainly discusses the system learning mechanisms, which contain both offline system training and online relevance feedback [ChenSC07]. First, an innovative method is proposed to automate the offline system training by using the association rule mining method. Second, online relevance feedback is then introduced to update the HMMM database model and provide refined results in real time. Finally, this chapter addresses the learning issues in designing and implementing a user adaptive video retrieval system, called MoVR, in a mobile wireless environment. Particularly, HMMM-based user profiles are designed and developed for learning individual user preferences as well as general user perceptions. The fuzzy association concept is utilized in the retrieval and ranking process such that users can get the flexibility to achieve their anticipated refining results in terms of different knowledge sets.

5.1 Introduction

Users are usually interested in specific semantic concepts and/or the associated temporal-based event patterns when querying a large scale video archive. In this study, a temporal event pattern is defined as a series of semantic events with some particular temporal relations. In soccer video retrieval, an example temporal event pattern query can be expressed as “Search for those soccer video segments where a goal results from a free kick.” Using the algorithm proposed in the previous chapter, the system should be able to search for the video clips that contain the desired pattern and rank them based on a certain similarity measurement method. However, not all of the returned video clips will be chosen by the user as positive results. The possible reasons are (1)

some video clips may not exactly match the requested events due to the accuracy constraints of the automatic event annotation algorithm, and (2) though some video clips match the correct event pattern, they do not satisfy the user's particular interests. Furthermore, the ranking initially may not reflect the user expectations. Thus, the system should allow user feedback and learn from it to filter out inaccurate results as well as refine searching & ranking performance.

Once the initial retrieval results are returned and displayed, users should be allowed to provide their feedback through the client-side interface. Different people may have different perspectives when evaluating the similarity of the retrieved results and their expected video clips. Taking the query pattern with only one event as an example, Figure V-1 illustrates two possible scenarios for distinct users' feedbacks. Given a query to search for the goal shots, a set of results are returned and 10 of them are shown in Figure V-1. One user may want to find a goal possibly resulting from a corner kick so that the 1st, 5th, and 7th key frames marked in red rectangles are selected as the samples of interest to provide the feedback (shown in Figure V-1(a)). Another scenario is shown in Figure V-1(b), where the other user may want a specific set of results in some series of soccer video games (e.g., "FIFA Women's World Cup 2003" in this example) which represent the similar visual clues. The anticipated key frames of this user are marked in the blue rectangles as shown in Figure V-1(b).

In general, a set of possible properties can be used to simulate a user's selections, e.g., low-level visual and audio features, high-level semantic concepts, and possibly the temporal information. As stated above, it is anticipated that the relevance feedback can be supported by the video retrieval system, and therefore the next round of results can be generated and ranked in real-time based on the individual user's perspectives. Furthermore, the massive amount of feedback from multiple users should also be considered to improve the overall performance of the video retrieval mechanism in the long run. In this section, we will discuss the online relevance of feedback performance, as well as the procedures for the off-line system training.



(a)



(b)

Figure V-1. Two feedback scenarios for the soccer video goal event retrieval

5.2 Related Work

One of the most challenging tasks in multimedia information retrieval is to perform the training and learning process such that the retrieval performance of the multimedia search engine can be refined efficiently and continuously. In general, existing multimedia system training and learning mechanisms can be categorized into online learning and offline training.

Relevance feedback (RF) [Rui98] is designed to bridge the semantic gap and provide more accurate results based on the user's responses. Incorporating RF is an online solution for improving retrieval accuracy, especially for still-image applications. However, existing RF approaches have several constraints and limitations such that it is difficult to employ RF in video retrieval approaches. For example, it does not incorporate any methodology to model all layers of multimedia objects and, consequently, it does not offer efficient learning for multimodal video retrieval to satisfy general users' interests. In addition, as mentioned by Muneesawang and Guan

[Muneesawang03], RF does not offer a decent solution for video database representation to incorporate sequential information for analytic purposes. Research efforts have been conducted to extend and refine the RF method for video retrieval and learning purposes. Several multimedia system training approaches try to utilize other possible learning mechanisms such as Support Vector Machine (SVM) and Neural Network techniques. For example, a template frequency model was proposed and a self-learning neural network was employed to implement an automatic RF scheme by Muneesawang and Guan [Muneesawang03]. Yan et al. [Yan03] describe a negative pseudo-relevance feedback (NPRF) approach to extract information from the retrieved items that are not similar to the query items. Unlike the canonical RF approach, NPRF does not require the users to make judgments in the retrieval process, as negative examples can be obtained from the worst matching results. In Bruno et al. [Bruto06], a query-based dissimilarity space (QDS) was proposed to cope with the asymmetrical classification problem with query-by-examples (QBE) and RF, where system learning in QDS is completed through a simple linear SVM. However, this linear-based method failed to satisfy the complicated requirements for content-based video retrieval and learning.

For offline training algorithms, the current research mainly focuses on one-time training using certain kind of data sets or classification information. For some cases, user feedback is not the major data source in system training. For instance, Hertz et al.

[Hertz03] introduced a learning approach using the form of equivalence constraints which determine whether two data points come from the same class. It provides relational information about the labels of data points rather than the labels themselves. An automatic video retrieval approach was proposed by Yan et al. [Yan04] for the queries that can be mapped into four predefined user queries: named persons, named objects, general objects, and scenes. It learns the query-class dependent weights utilized in retrieval offline. This kind of offline training processes is time-consuming, not fully automatic, and limited to pre-defined query types.

In summary, most of the current online learning algorithms mainly deal with interactions with a single user. Due to the small amount of feedback, they can be performed in real-time, but the performance could only be improved to a limited extent, especially when handling a large-scale multimedia database. On the other hand, some offline training methods try to learn the knowledge from not only collected user feedback, but also some other training data sources. The performance could be better as it considers more training data, but the major drawback is that a manual process needs to be executed to initiate the training process. Moreover, since training is performed through the entire database, it becomes a tedious task and can only run offline.

5.3 Automate Offline Training using Association Rule Mining

User feedback is widely deployed in recent multimedia research to refine retrieval performance. In the previous chapter, we showed an HMMM mechanism designed to support offline training. For the sake of efficiency, the training system is designed to record all the user access patterns and access frequencies during a training period. Once the number of new feedbacks reaches a certain threshold, the system will trigger the matrix update procedure automatically. All the calculations are executed offline. This procedure helps to refine the overall system performance in the long run.

This method can improve the performance but it becomes a manual process to decide the threshold and initiate the training process. To address this challenge, we propose an advanced training method by adopting the association rule mining technique, which can effectively evaluate accumulated feedback and automatically invoke the training process. Training is performed per video rather than for all videos in the database, making the process more efficient and robust. In addition, it can further improve semantic modeling in the video database and continuously improve retrieval performance in the long run. As an example, the proposed method is applied to a mobile-based soccer video retrieval system and the experimental results are analyzed.

In this section, the association-rule mining (ARM) technique [Agrawal93] [Agrawal94] is applied to automate the training process. The automated training process has the following advantages. First, the multimedia system is updated to check the threshold effectively in real time and initiate the training automatically using accumulated user feedback. In other words, no manual process is required. Second, the overall training process becomes more efficient and robust since only part of the video database that contains enough historical retrieval data and positive patterns needs to be updated. Finally, the training process can further improve semantic video database modeling and continuously improve system retrieval and ranking performance in the long run.

5.3.1 Overall Process

Figure V-2 shows the overall process of the proposed method. When a user issues a query pattern, the background server executes the query and ranking process such that the system can return to the user with the ranked video clips that match the query pattern. The user is allowed to choose his/her favorite video clips as positive patterns and issue the feedback. The server engine receives the feedback and accumulates them in the video database. The system then checks if the number of new feedbacks reaches a pre-defined threshold. If yes, a background checking mechanism is invoked automatically to evaluate the feedback using ARM. Otherwise, the next round query is performed. For efficiency purposes, the system will check if there is any video containing enough positive patterns. If no video satisfies the qualification, then the training process will not proceed. Otherwise, the system will initiate the training process on certain video(s) based on the evaluation results.

After the training, the positive patterns used in training the video database model will be removed from the untrained feedback data set and, accordingly, new counts begin for the next query. After the underlying video database has been updated, all the users can have the opportunity to achieve the refined ranked results based on the trained video database models.

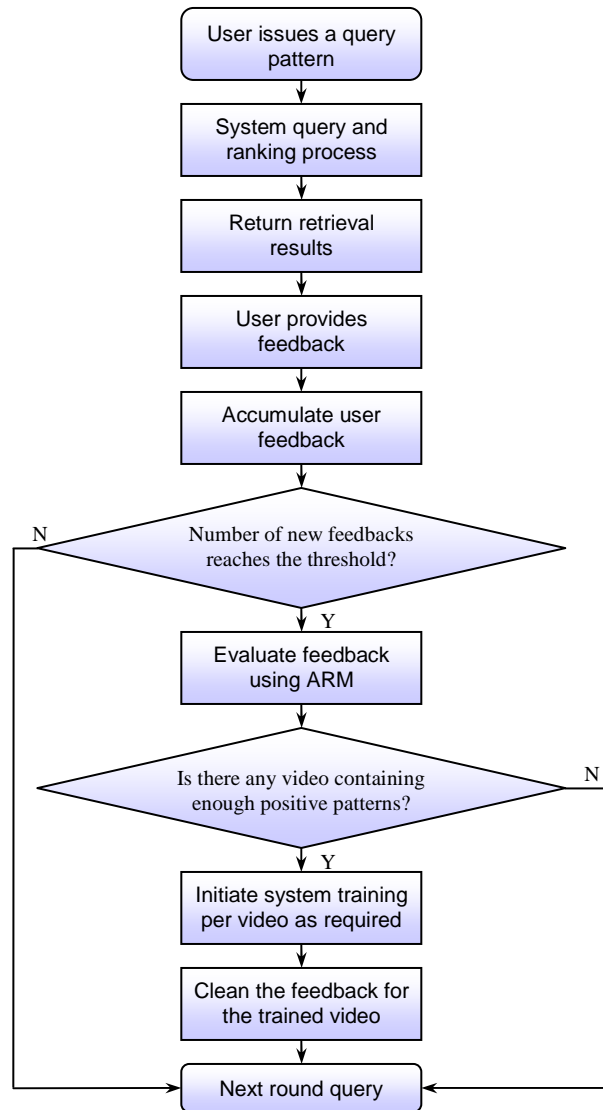


Figure V-2. Overall process for the automated training

Although this figure only shows the process sequence for one user, the system actually collects feedback from multiple users. Only one evaluation measure is calculated for each video and updated with the accumulated feedback from the common users. Of course, mutually exclusive issues should be considered such that when the feedback from one user invokes system training for a certain video, the thread which processes the request from another user should be aware of this situation and certain actions should be restricted to avoid any conflicts.

It is also worth mentioning that this proposed framework can be easily adjusted and applied to image retrieval applications. The basic idea is to evaluate if there are enough positive image patterns in a local image database. That is, the training process is performed on independent local image databases rather than the overall image repository.

5.3.2 Automated Training using ARM

The challenge for such a multimedia training process is to determine a suitable threshold value to invoke model re-training for a video v . Because the support measure used in ARM [Agrawal93] [Agrawal94] can well capture the percentage of data tuples for which the pattern is true, we investigate how to best adopt this concept for the purpose of inspecting whether the underlying HMMM model for a particular video v needs to be re-trained.

As first introduced by Agrawal et al. [Agrawal93], ARM is designed to discover items that co-occur frequently within a data set. Given a set of transactions in market basket analysis applications, where each transaction contains a set of items, an association rule is defined as an expression $X \Rightarrow Y$, where X and Y are sets of items and $X \cap Y = \emptyset$. The rule implies that the transactions of the database containing X tend to also contain Y . In ARM, the support constraint concerns the number of transactions that support a rule. The support value is defined to be the fraction of transactions that satisfy the union of items in the consequent and antecedent of the rule.

This idea can be mapped and applied to mine the association rules in the positive feedback. Here, each positive event pattern is treated as a transaction, and the historical access pattern database is defined as the set of all transactions. To satisfy our requirements, we modify the definition of target rules and define them as two itemset association rules which follow certain temporal sequences. For example, $s_m \Rightarrow s_n$ can be treated as a target rule, where s_m and s_n are video shots in video v and $T_{s_m} < T_{s_n}$. Accordingly, the support measure can be defined as below:

$$Support(m, n) = \frac{Count(s_m \Rightarrow s_n)}{NumTrans}, \quad (V-1)$$

where $Count(s_m \Rightarrow s_n)$ returns the number of positive event patterns that contain the rule of $s_m \Rightarrow s_n$, and $NumTrans$ represents the total number of all temporal event patterns (transactions) in the data set of positive feedbacks which were not used in the previous training process.

In this application, we are more concerned with the number of all rules in a certain video than the number of a specific rule. For a given video v , we can sum all the counts for the identified temporal rules to get this number as $\sum_{s_m} \sum_{s_n} Count(s_m \Rightarrow s_n)$, which can also be represented as $\sum_{s_m} \sum_{s_n} Support(m, n) \times NumTrans$.

In our video retrieval and training system, a novel means for representing the percentage of s_m and s_n in video v that are accessed in the positive pattern R_y with $T_{s_m} < T_{s_n}$ can be utilized to define an ‘‘Evaluation’’ measurement for each video for checking purposes.

$$Evaluation(v) = \frac{\sum_{s_m} \sum_{s_n} Support(m, n) \times NumTrans}{\sum_{s_m} \sum_{s_n} aff_1(m, n)} \quad (V-2)$$

Equation V-2 captures the percentage of s_m and s_n appearing in the positive temporal patterns versus the overall affinity relationship between them. If this percentage reaches a certain value, it indicates that such a relationship should be reflected more frequently in the model training process. Here, the threshold is defined as H to see if the video is ready for the next round of training.

- When $Evaluation(v) < H$: The database model for video v will not be trained and the feedback is simply accumulated in the server-side.
- When $Evaluation(v) \geq H$:

$$\begin{aligned} \text{aff}_1(m, n) &= A_1(m, n) \times \text{Support}(m, n) \times \text{NumTrans}, \\ \text{iff } s_m \in R_y, s_n \in R_y, T_{s_m} \leq T_{s_n} \end{aligned} \quad (\text{V-3})$$

The *aff* values are then utilized for updating the corresponding affinity relationship matrix and initial state probability matrix for the particular video *v*.

5.3.3 Experimental Results for Automated Learning Mechanism

The proposed approach is applied to a distributed multimedia system environment with simulated mobile clients, which will be further introduced in Section 5.4. As shown in Figure V-3, after a user issues the event pattern query, the system will search, rank the results, and return the key frames to the user. Due to the limited size of wireless devices, each screen is designed to show up to six candidate video clips as presented in Figure V-3(a). By clicking the user preferred key frame, the corresponding video segment will be displayed as shown in Figure V-3(b) and the user can provide positive feedback by using the upper right button to trigger the choice of “I like it!”.



Figure V-3. System interfaces for the Mobile-based Video Retrieval System

For ARM, we use the source codes for the Apriori algorithm from [Apriori], which provides an efficient program (Borgelt et al. [Borgelt02][Borgelt03]) to find association rules and frequent itemsets with the Apriori algorithm. Apriori is designed to operate on the data sets containing transactions (i.e., the positive event patterns in the historical feedback). The current system contains 45 videos and around 10,000 video shots. The ARM-based system evaluation results are recorded in Table V-1. Experimental results for ARM-based feedback evaluations. Initially, the system performs ARM-based evaluation every 50 historical queries. However, it seems that the knowledge captured in the first 50 historical queries is not enough and no video needs to be trained. When it reaches 100 historical queries, more association rules are discovered. For example, there are 240 distinct items (positive video shots), 588 transactions (positive event patterns), 44 identified association rules, and accordingly 1 video passing the evaluation threshold. The database model of this video is trained separately. Next, all the positive feedback patterns that are in this trained video are removed from the unused feedback dataset. Then, the system starts ARM-based evaluations per 100 new feedbacks. The same procedures are applied when the number of queries reaches 200 and 300, where three videos and two videos are trained, respectively. The system is designed to conduct training per video, rather than for all the videos in the database. We believe such a design can improve semantic modeling in the second layer and lead to further improvements in the overall retrieval performance. For example, based on 200 historical queries, the system evaluates all videos and determines that three videos need to be trained. The historical data that are already used for the training process will then be excluded from the next round of evaluation. When the number of historical queries reaches 300, the database models for two more videos are required to be trained. Compared with system training which needs to update the MMM models for 45 videos, the proposed approach reduces the training time by approximately 900 percent, while achieving a similar degree of performance improvement.

Table V-1. Experimental results for ARM-based feedback evaluations

Num of Historical Queries	Items	Patterns (Transactions)	Num of All Rules	Videos need to be trained
50	149	286	18	0
100	240	588	44	1
200	268	1069	196	3
300	274	1559	185	2

5.4 Online Relevance Feedback

Our designed retrieval algorithm is capable of achieving the result events or patterns which match with the user-designed patterns. However, different users may have their own preferences and identify only partial results as their favorites based on their personal judgments. Hence, online relevance feedback functionalities can be incorporated in the proposed video retrieval system, which is realized by creating and updating the specific MMM instances for each individual user who has distinct preferences. The MMM instances are built upon the structure and values of the existing constructed HMMM model and they are used in online system learning in order to satisfy each specific user’s requirements.

Assume for a query pattern $Q = \{e_1, e_2, \dots, e_C\}$, where $T_{e_1} \leq T_{e_2} \leq \dots \leq T_{e_C}$, a set of temporal patterns are retrieved and G of them $\{R_1, R_2, \dots, R_G\}$ are marked as “Positive” by the user. Here, $R_y = \{\tilde{s}_1^y, \tilde{s}_2^y, \dots, \tilde{s}_C^y\}$ represents the y^{th} “Positive” pattern, where \tilde{s}_i^y is defined as the i^{th} video shot in the y^{th} “Positive” pattern, $1 \leq i \leq C$, and $1 \leq y \leq G$. The related MMM instances are constructed as below.

5.4.1 Anticipant Event Pattern Instance

Sometimes users may be interested in more than one event in some specific temporal position although they do not specifically identify them in the initial query pattern. Therefore, the system should be able to capture and learn the underlying possibilities for these additional events. In our proposed approach, the system creates an instance for the anticipated event patterns by

checking the “Positive” video shots. Assume there are totally m additional events $AE = \{e_1', e_2', \dots, e_m'\}$ at time $T_{e_1'} \leq T_{e_2'} \leq \dots \leq T_{e_m'}$ in the user’s feedback on the G positive patterns. The numbers of their occurrences are denoted as $p'(e_i')$, where $0 < p'(e_i') \leq G$ and $1 \leq i \leq m$. Therefore, the actual query pattern is expanded from $Q = \{e_1, e_2, \dots, e_C\}$ to $Q' = \{e_1, e_2, \dots, e_{C+m}\}$. For an event $e_i \in Q'$, if $e_i \in Q$, its weight is $p_i = 1$. Otherwise, if $e_i \in AE$, $p_i = p'(e_i')/G$. This means that a newly detected event holds the weight based on its occurrence frequency in the “Positive” patterns. These additional events and their occurrence probabilities are used in the next round of retrieval to get the updated similarity scores and perform the new similarity ranking process.

Given a simple example, for a goal shot retrieval example, the users marked ten results as “Positive”, where six of the video shots are also “Corner Kick” shots, two of them are also marked as “Free Kick” shots, and the other two only have the annotation of “Goals”. Therefore, we can extract a set of two additional events $\{e_1', e_2'\}$ which are not mentioned in the query pattern, where e_1' denotes “Corner Kick” and e_2' represents “Free Kick”. Their occurrence probabilities are 6/10 and 2/10, respectively. Accordingly, in the next round of retrieval, these two user-preferred events should be included for comparison purposes. However, since the extracted additional events are not specified by the user, they are used only for the similarity measurement calculation.

5.4.2 Affinity Instances for A

In this framework, the affinity instances are retrieved and updated for the purpose of user preference learning. Assuming the G positive patterns come from M' distinct videos ($M' \leq G$), one affinity instance (A_1^*) for each of these M' videos can be constructed. Accordingly, M' affinity instances can be generated. In each affinity instance, the rows represent the positive video shots, and the columns represent all the shots in this video. The affinity values of the “Positive”

video shots are extracted from A which the corresponding rows are from the existing affinity relationship matrices. Let $FP(i,j)$ be the number of positive feedback patterns which contain the temporal sequence like $\{\dots, s_i, s_j, \dots\}$ and v' be one of the M' positive videos, the affinity instance A_1^* for video v' is generated as follows.

$$A_1^*(i, j) = \frac{A_1(i, j) \times (1 + FP(i, j))}{\sum_{s_k \in v'} (A_1(i, y) \times (1 + FP(i, y)))}, \quad (\text{V-4})$$

where $s_i \in v', s_j \in v', T_{s_i} \leq T_{s_j}, s_i \in R_y, 1 \leq y \leq G$.

Let $FV(i,j)$ denote the number of feedbacks where two videos v_i and v_j are accessed together. Accordingly, the higher-level affinity instance A_2^* is generated as shown below.

$$A_2^*(i, j) = \frac{A_2(i, j) \times (1 + FV(i, j))}{\sum_{v_x \in D} A_2(i, x) \times (1 + FV(i, x))}, \text{ where } 1 \leq i \leq M, 1 \leq j \leq M. \quad (\text{V-5})$$

5.4.3 Feature Instances for B

The initial search for the target events or patterns tries to compare the features of the candidate video shots with the mean values for the features of the target events (B'). Since users may potentially have their own preferences on some kind of visual or audio features, feature instances are also constructed. In other words, once the feedback is issued, the features of positive shots will be taken into account for the next round of similarity measurement. As shown in Equation (V-6), the feature instance matrix B_1^* is constructed for B by calculating the mean values for the visual/audio features of the ‘‘Positive’’ video shots.

$$B_1^*(i, j) = \frac{\sum_{y=1}^G B_1(\tilde{s}_i^y, f_j)}{G}, \text{ where } 1 \leq i \leq C+m, 1 \leq j \leq K. \quad (\text{V-6})$$

5.4.4 Updated Similarity Measurements and Query Processing

By considering the low-level features, high-level semantic concepts, as well as the user's perceptions, the specific preferences can be efficiently captured and learned such that the new requirements are incorporated into the feedback-based similarity measurement procedure. Let $R'_y = \{s_1, s_2, \dots, s_{C+m}\}$ be the y^{th} candidate pattern which matches the query event pattern Q' , the similarity score of R'_y can be calculated by Equations (V-7) to (V-11).

$$dis^*(\bar{s}_t, e_t) = \left(\sum_{k=1}^K (O_{1,2}(e_t, f_k) \times (B_1(\bar{s}_t, f_k) - B_1^*(e_t, f_k))^2) \right)^{1/2}, \quad (\text{V-7})$$

where $\bar{s}_t \in S_1, 1 \leq t \leq C + m$.

$$sim^*(\bar{s}_t, e_t) = \frac{1}{1 + dis^*(\bar{s}_t, e_t)} \quad (\text{V-8})$$

where $\bar{s}_t \in S_1, 1 \leq t \leq C + m$.

$$w_1^*(\bar{s}_1, e_1) = \Pi_1(\bar{s}_1) \times sim^*(\bar{s}_1, e_1) \times p_1. \quad (\text{V-9})$$

$$w_{t+1}^*(\bar{s}_{t+1}, e_{t+1}) = \begin{cases} w_t^*(\bar{s}_t, e_t) \times A_1^*(\bar{s}_t, \bar{s}_{t+1}) \times sim^*(\bar{s}_{t+1}, e_{t+1}) \times p_{t+1}, & \text{where } 1 \leq t < C + m, \bar{s}_{t+1} \in R_y; \\ w_t^*(\bar{s}_t, e_t) \times A_1(\bar{s}_t, \bar{s}_{t+1}) \times sim^*(\bar{s}_{t+1}, e_{t+1}) \times p_{t+1}, & \text{where } 1 \leq t < C + m, \bar{s}_{t+1} \notin R_y. \end{cases} \quad (\text{V-10})$$

$$SS^*(Q', R'_y) = \sum_{t=1}^{C+m} w_t^*(\bar{s}_t, e_t). \quad (\text{V-11})$$

Figure V-4 illustrates the procedure of the concept-based video retrieval and online system learning. After the first round of retrieval, the users provide their feedbacks and then the system tries to extract the affinity instances and feature instances from the positive multimedia objects. The anticipated event pattern is also generated by including more possible events. The retrieval process on the right hand side follows in a similar way as the initial query processing (as shown in Figure V-4) except the following three aspects: (1) When calculating the weights, the system will check the affinity instances in the user's profile. As shown in Equation (V-10), if the video shot is not positive in R_y , it will not be included in A_1^* , and therefore the system will go to

A_1 to get the affinity value. (2) The features used for similarity measurements are changed from B' to B_1^* . (3) If there are additional events detected, the system will calculate the similarity score for the generated pattern R_y' .

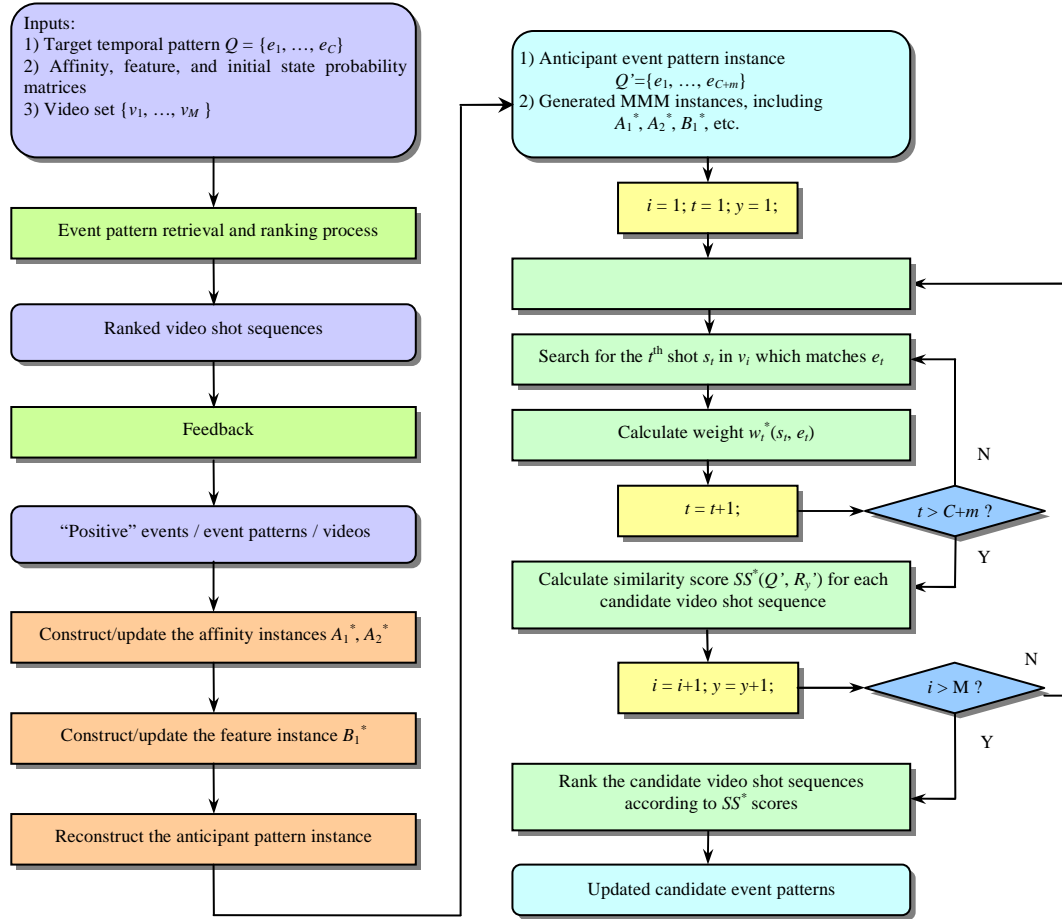


Figure V-4. Online learning procedure of temporal based query pattern retrieval

5.4.5 Experimental Results for System Learning Techniques

The current video database contains 54 soccer videos, which are segmented into 11,567 video shots and 506 of them are important semantic events. In the experiments, three query examples are used to demonstrate the results of the retrieval method and how the online relevance feedback method improves the precision of the results and their rankings.

- *Query 1*: Search for the goal events.
- *Query 2*: Find the patterns containing a goal after a free kick.
- *Query 3*: Search for a video segment with the temporal pattern containing a sequence of four events, namely “Goal”, “Player Change”, “Corner Kick”, and “Free Kick”.

Figure V-5 demonstrates the soccer video retrieval and user feedback interface with Query 3 being issued. The results are returned on the right hand side of the panel and sorted from left to right and top to down. As marked in the purple box, four video shots form a candidate temporal event pattern. By double clicking the interested key frame, the video shots are displayed in the upper-left corner of the screen. Users are able to select their preferred video shots by using the drop down menu below the key frames.

Video retrieval is not merely the comparison of low-level visual or audio features. Moreover, people may have totally different interests when they search for their target video shots. For example, a total of 72 results are retrieved for Query 1, and all of them are proved to be goal shots. The initial ranking is the same for all the users. However, users have different feedbacks even through the initial results. Some users may choose the ones in their interested series of soccer games, some users may prefer the goals resulting from the corner kicks, while others may be interested in the “exciting” goal shots where the term of “exciting” is defined subjectively based on their own judgments. It is a very complicated task to learn high-level user perceptions since even the users themselves may not be able to describe their own requirements. However, our proposed mechanism is capable of refining the video event retrieval results by modeling not only low-level features but also high-level concepts and user preferences by utilizing the HMMM mechanism and MMM instances.



Figure V-5. User-centered soccer video retrieval and feedback interface

In our experiments, 20 users participated, with most of them not computer specialists. Figure V-6 shows the number of user preferred video shots in two rounds of retrieval. From the first 16 candidate video shots initially shown, the average number of user preferred video shots is 5.5. After the first round of feedback, it reaches 8. It is noticeable that most of the users achieve better results after the first round of feedback. Few of the users have very strict requirements when they choose the goal shots, and therefore the improvement gain is limited even after the feedback and learning process.

As for Query 2, there are 75 pairs of results retrieved from the video database. However, based on our experiments, we observed that the number of similar patterns is very small. Therefore, an example instead of numbers is used to demonstrate the performance. Figure V-7(a) shows the initial result where the patterns in red boxes are selected as “Positive”. They were

ranked as the 11th and 15th candidate patterns initially. As illustrated in Figure V-7(b), after two rounds of feedbacks, all the similar patterns in the same series of soccer videos are successfully retrieved and ranked as the 1st and 8th.

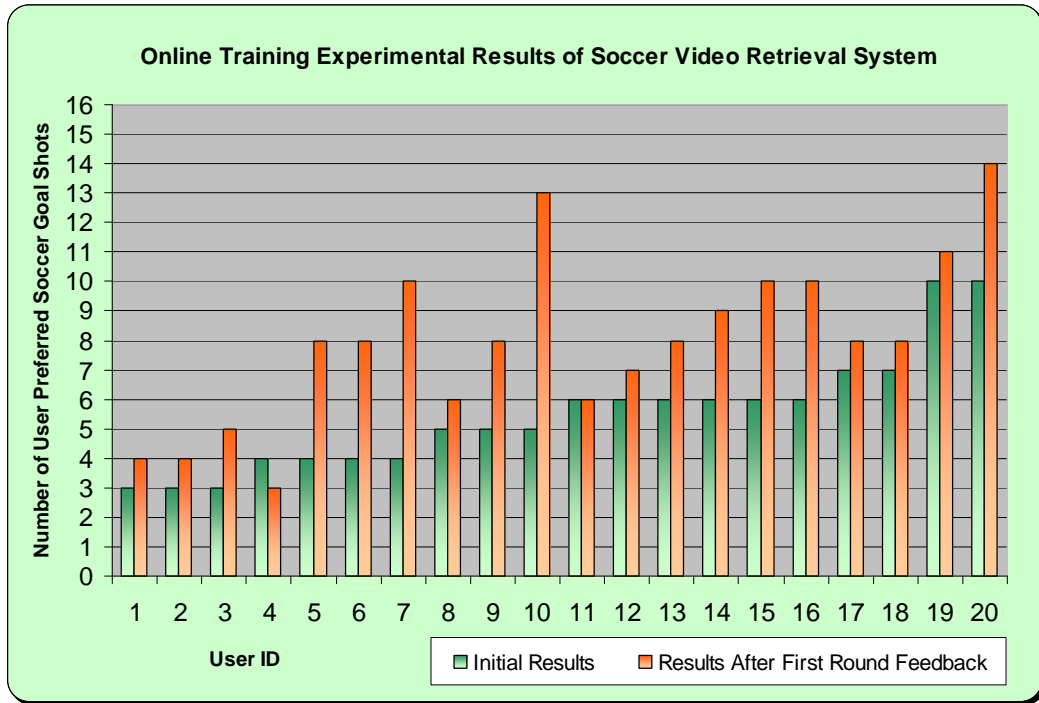


Figure V-6. Online training experimental results for Query 1



Figure V-7. Soccer video retrieval and feedback results for Query 2. (a) first round event pattern retrieval; (b) third round event pattern retrieval.

5.5 Application: A Mobile-based Video Retrieval System

5.5.1 Introduction

Nowadays, handheld mobile devices including cell phones and Personal Digital Assistants (PDAs) have become increasingly popular and capable, which creates new possibilities for accessing pervasive multimedia information. The new generations of mobile devices are no longer used only for voice communication; they are also used frequently to capture, manipulate, and display different audiovisual media contents.

With the rapid emergence of wireless network technologies, such as GSM, Satellite, Wireless Local Area Network (WLAN), and 3G, it is much easier now to transfer large sized multimedia items to mobile clients. However, multimedia mobile services still suffer not only the constraints of small display sizes, but also a limitation in terms of power supply, storage space, processing speed, etc. The navigation of multimedia content on handheld devices is always restricted in the limited time periods with minimized numbers of interactions. Meanwhile, large sized multimedia data such as video cannot be stored permanently on the mobile devices due to the limited memory.

Consider the following typical scenario in a mobile-based multimedia application. Sports fans wish to watch sports video through their cell phones. However, it is both unaffordable and sometimes unnecessary for them to watch the entire game, which would take a long time, occupy huge memory, and exhaust the power supply. Thus, a better solution is to offer them the capability of browsing and retrieving only short video clips containing interesting event shots. Given a huge collection of sports videos, there exist many challenges to accomplish this task.

- It is quite difficult to segment the videos properly and annotate the semantic events automatically. Although advanced techniques offer great capabilities in extracting multimodal visual and audio features from all kinds of videos, the “semantic gap” remains a crucial problem when bridging these low-level or mid-

level features with high-level rich semantics. Even with the best event annotation algorithms, there is hardly sufficient guarantee in terms of the correctness and completeness of the semantic interpretation results. This motivates the modeling of high-level semantic abstractions by utilizing the existing annotation results, their features, and user feedbacks.

- It is critical to address the database modeling issue, especially when considering the temporal and/or spatial relationships between the multimedia objects. It should be able to support not only the basic retrieval methods, but also the complicated temporal event pattern queries.
- There is an emerging need for supporting individual user preferences in multimedia applications. It is well-known that people have diverse interests and perceptions towards media data. Thus, it is desirable to incorporate user feedbacks with the purpose of training the retrieval system.
- In the meanwhile, we may want to reduce the number of user interactions to alleviate the burden on the users and to accommodate the restrictions of the mobile devices. It is thus desirable to keep track of user actions and accumulate knowledge about user preferences.
- The system architecture should be designed to reduce the size of data to be transferred and to minimize the requirement of data storage for the mobile devices.
- The mobile-based retrieval interface should be user-friendly, easy to operate, and capable of offering sufficient information and choices for the users.

Therefore, an efficient and effective multimedia content management and retrieval framework will be essential for the evolution of mobile-based multimedia services.

This section addresses the issues of designing and implementing a user adaptive video retrieval system, called MoVR, in a mobile wireless environment. Innovative solutions are developed for personal video retrieval and browsing through mobile devices with the support of content analysis, semantic extraction, as well as user interactions. First, a stochastic database modeling mechanism called Hierarchical Markov Model Mediator (HMMM) is deployed to model and organize the videos, along with their associated video shots and clusters, in a multimedia database to offer support for both event and complicated temporal pattern queries. Second, HMMM-based profiles are designed to capture and store individual user's access histories and preferences such that the system can provide a "personalized recommendation."

Third, the fuzzy association concept is employed to empower the framework so that the users can make their choices of retrieving content based solely on their personal interests, general users' preferences, or anywhere in between. Consequently, users gain control in determining the desirable level of tradeoff between retrieval accuracy and processing speed. In addition, to improve the processing performance and enhance the portability of client-side applications, the storage consumption information and computationally intensive operations are supported in the server-side, while mobile clients mainly target to manage the retrieved media and user feedbacks for the current query. In order to provide more efficient accessing and information caching for the mobile devices, we also designed the virtual clients at the server-side computers to keep some relevant information that mobile users require. To demonstrate the performance of the proposed MoVR framework, a mobile-based soccer video navigation and retrieval system is implemented and tested.

5.5.2 Related Work

Video browsing and retrieval in mobile devices is an emerging research area. Due to the constraints of mobile devices in terms of their power consumption, processing speed and display

capability, more challenges have been encountered than in traditional multimedia applications and many research studies have been conducted to address various issues.

To reduce the viewing time and to minimize the amount of interaction and navigation processes, a variety of studies in academia and industry have worked on the summarization of video contents. For instance, in [Gong01], Singular Value Decomposition (SVD) of attribute matrix was proposed to reduce the redundancy of video segments and thus generate video summaries. Clustering techniques were also used to optimize key frame selection based on visual or motion features to enhance video summarization [Babaguchi01]. In industry, Virage has implemented preliminary video summarization systems for NHL hockey videos using multimodal features [Virage]. However, a major issue remains in terms of the semantic gap between computable video features and the meaning of the content as perceived by the users. For this purpose, metadata about the content was used and it played an active role in video retrieval [Sachi05]. For instance, ontologies have been proposed in [Jokela00] to perform intelligent queries and video summarization from metadata. In [Tseng02], a video semantic summarization system in the wireless/mobile environments was presented, which includes an MPEG-7 compliant annotation interface, a semantic summarization middleware, a real-time MPEG1/2 video transcoder for Palm-OS devices, and an application interface on color/black-and-white Palm-OS PDA. Metadata selection component was also developed in [Lahti06a][Lahti06b] to facilitate annotation. However, automatic media analysis and annotation is still far from mature and purely manual media annotation is extremely time-consuming and error prone [Davis04]. Alternatively, semantic event detection frameworks have been proposed to facilitate video summarizations. Hu et al. used similar video clips across different sources of news stations to identify interesting news events [Hu01]. In our earlier studies [ChenSC03a][ChenSC04a], an effective approach was proposed for video event detection facilitating multimedia data mining technique and multimodal feature analysis.

In terms of video retrieval in mobile devices, “Query by Example” (QBE) is a well-known query scheme used for content-based audio/visual retrieval, and many systems have been developed accordingly [Ahmad06] [Sachi05]. However, in most existing approaches, the similarity estimation in the query process is generally based on the computation of the (dis)similarity distance between a query and each object in the database and followed by a rank operation [Ahmad06]. Therefore, especially for large databases, it may turn out to be a costly operation and the retrieval time becomes unreasonably long for a mobile device. In addition, temporal pattern query, where a sequence of temporal events is of interest, is not well supported in these studies.

In essence, the aforementioned approaches contribute to address some restrictions of video browsing and retrieval in mobile devices. However, they fail to accommodate individual user preferences with their diverse interests and perceptions towards video data. In the literature, relevance feedback [Rui98] has been widely adopted in the content-based retrieval research society to address user preference issue. [Coyle04] presents a mechanism for learning the requested feature weights based on user feedbacks to improve the recommendation ranking. In addition, [Dubois01] studied the application of fuzzy logic in the representation of flexible queries and in expressing a user’s preference learned from feedback in a gradual and qualitative way. In [Doulamis99], fuzzy classification and relevance feedback techniques were applied for processing the video content to capture user preference. A similar idea was also proposed in [Kang06], where a fuzzy ranking model was developed based on user preference contained in feedbacks. However, a common weakness of most relevance feedback methods is that the feedback process has no “memory” [LiQ01] so the user feedbacks conducted in the past fail to help the future queries. Therefore, the retrieval accuracy does not improve over time in the long run.

In addition, mobile devices have limited luxury to support frequent interactions and real-time feedback learning. Alternatively, user profiling has been extensively used in information filtering and recommendation. In [ChenL98], a personal agent called WebMate is devised which learns the user profile incrementally and facilitates browsing and searching on the web. [Martin02] also presented a study of the role of user profiles using fuzzy logic in web retrieval processes. John et al. [John01] developed a prototype information retrieval system using a combination of user modeling and fuzzy logic. [Gibbon04] presented the idea of extracting relevant video clips based on a user profile of interests and creating personalized information delivery systems to reduce the storage, bandwidth, and processing power requirements and to simplify user interactions.

In our framework, a common profile which represents the general knowledge on the semantics of video data is constructed. Such a general user profile serves as the semantic indexes of video data to speed up the retrieval process. Meanwhile, a user profile is set up for each user to characterize the personalized interest. In addition, multilevel video modeling and temporal pattern queries are well supported in our approach.

5.5.3 System Architecture

In this proposed research, the traditional client/server system architecture is adopted but enhanced to accommodate the requirements for the mobile-based multimedia services. In order to provide the maximum support and optimized solution, the following criteria are strictly followed in the system design. First, storage consumption information and computationally intensive operations are handled on the server-side. Second, mobile clients are solely required to maintain the minimized data to enable the retrieval process. Third, the system should reduce the load for the wireless network, and at the same time increase the data transfer speed for the multimedia data.

In the server-side database, a huge amount of multimedia data are stored and managed by employing the Hierarchical Markov Model Mediator (HMMM) mechanism. The video database contains not only the archived videos, video shots, and clusters, but also the numerical values which represent their affinity relationships, features, and access histories, etc.

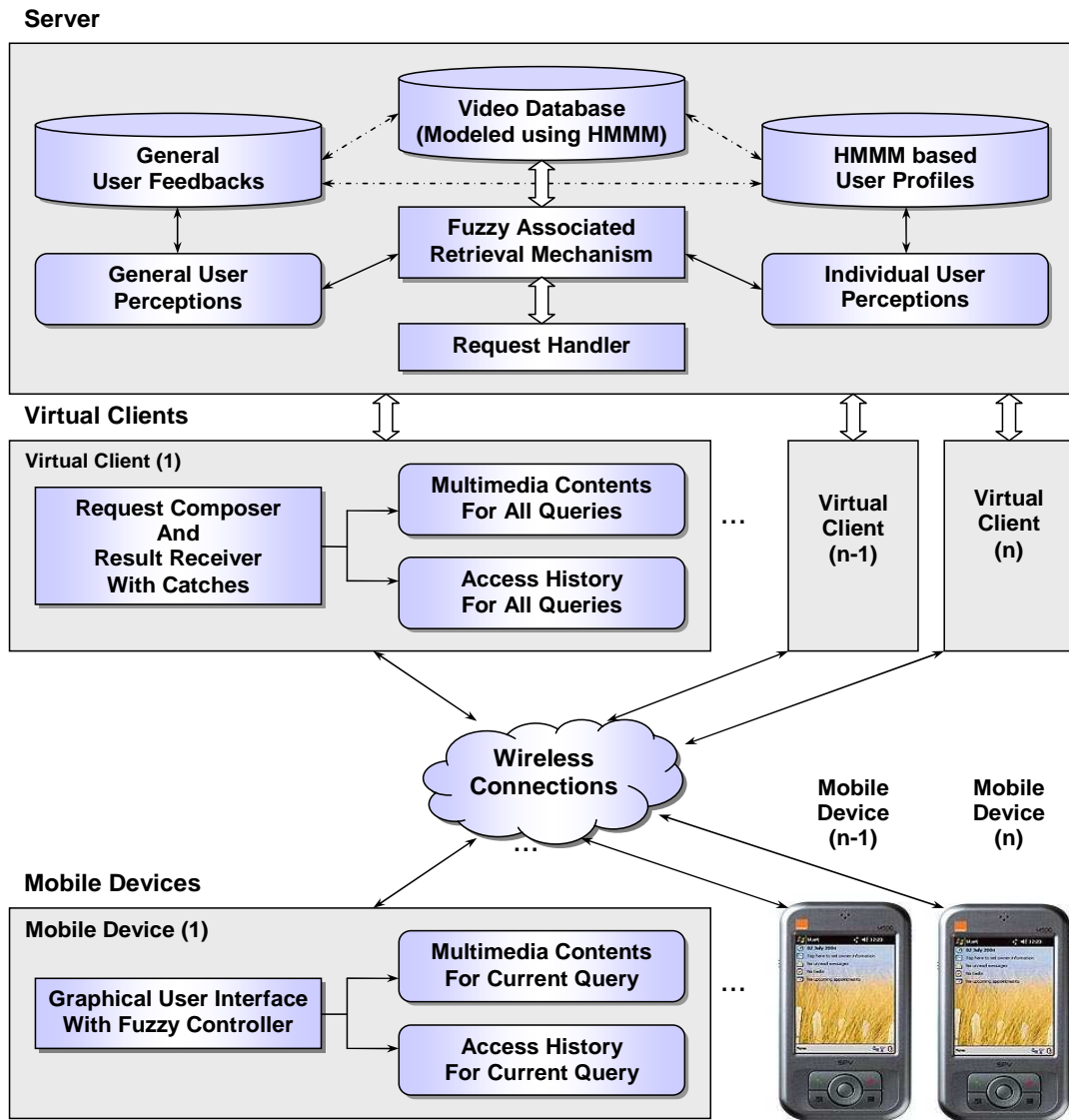


Figure V-8. Mobile-based video retrieval system architecture

As shown in Figure V-8, the database for general user feedback is developed, which consists of the positive access events or patterns from a whole group of different users. The individual user feedback can also be extracted from this database to develop the HMMM-based individual user profiles, which will be explained in the later sections. These access histories are utilized by the system to learn both the general user perceptions and individual user interests. Based on the fuzzy weight provided by the mobile user, the fuzzy associated retrieval algorithm is able to make a compromise between these two models of perceptions intelligently, retrieve the video clips, and make the ranking recommendations accordingly.

The request handler is designed to interpret the request packages and to respond to the mobile devices by sending back the retrieved and ranked results. The client-side applications on mobile devices do not need to keep storage of all the accessed media data. Alternatively, they mainly target to manage the retrieved media and user feedbacks for the current query, which includes the key frames of video shots shown on the current screen, and the video clips requested for the current operation. The mobile-based graphical interface is designed for the video retrieval system, which allows the user to easily compose and issue the event or temporal pattern based queries, to navigate through and watch the collection of retrieved results, and to provide the feedback.

In order to promote the mobility and manageability of this system, a new layer with “virtual clients” is designed and incorporated in the server-side applications to extend the dynamic computing and storage capability of the mobile clients. The virtual client is designed to represent the mobile user state in the mobile-based video retrieval system. Each virtual client is customized to a distinct mobile user who accesses the video retrieval system. It contains a communication component that consists of the requests by checking and collecting the messages and commands sent from the mobile devices. The communication component can also receive the multimedia data results from the server. Since we want to reduce the data size stored in the

mobile devices, the virtual client is designed to cache all the related multimedia content and access histories for the corresponding mobile user.

Generally speaking, the proposed virtual client solution can deliver improved flexibility, scalability and cost benefits over the traditional client/server models. Mobile users gain efficiency and productivity because they can access the multimedia resources without worrying about their storage limitation.

5.5.4 MoVR: Mobile-based Video Retrieval

In this chapter, the MoVR framework is proposed for the mobile-based video retrieval system development. This framework can support not only basic event queries, but also the complicated queries towards some temporal event pattern, which consists of a set of important events followed by a certain temporal sequence. More importantly, this framework is capable of providing both personalized recommendations and generalized recommendations. Users can also specify a fuzzy weight parameter if their query interests have not yet been clearly formed. The system will make the adjustment and generate different retrieval results based on the fuzzy associated queries. In essence, MoVR is designed to provide not only powerful retrieval capabilities but also a portable and flexible solution to mobile users.

As shown in Figure V-9, the overall framework of MoVR includes three main processing phases on the server-side.

Phase 1 is for video data preprocessing. It consists of the following steps. The first step is to process the source video data to detect the video shot boundaries, segment the video, and extract the shot-based features. Data cleaning and event annotation algorithms are then applied to detect the anticipated semantic events by employing the extracted shot-level features and multimodal data mining scheme. The components in Phase 1 are processed offline, which is not the focus of this chapter.

Phase 2 is to model the video databases. As shown in the top-center box, the HMMM mechanism is deployed to model the multilevel video entities, all kinds of features, along with their associated temporal and affinity relationships. These process steps are also performed offline. Such an HMMM database model will be updated periodically during the learning process by utilizing user feedback.

Phase 3 includes the system retrieval and learning processes which are mainly performed online in real time to interact frequently with the virtual clients and the mobile clients. Once a user issues a query requesting a certain semantic event or a temporal event pattern, such information will be sent to the virtual client, where it is packed and passed to the server for processing. After this point, the process will be slightly different for a first time user or a revisiting user on the server-side.

For the former case, the HMMM model with initial settings will be adopted and the system performs general similarity matching and ranking processes. In contrast, for a revisiting user, his/her user profile stored in the server-side will be retrieved and used for more advanced retrieval functionalities. Accordingly, an enhanced algorithm is developed on the server-side to handle these fuzzy associated video retrieval and ranking tasks. The retrieved video clips are ranked and sent back to the virtual clients. Though all the results are cached for fast retrieval, only a portion of them are actually delivered to the mobile devices in default. Users may issue feedback towards the query results through their mobile devices, which will be sent to the virtual client so that this feedback can be organized and temporarily stored. After that, the feedback will be delivered to the server for the construction or update of the user profiles. Moreover, real-time online learning is also supported so that the system will yield refined results based solely on the feedback for the current query. Essentially, two innovative techniques, user profiling and fuzzy association, are adopted and integrated with HMMM intelligently for server-side applications, which are introduced in the following sections.

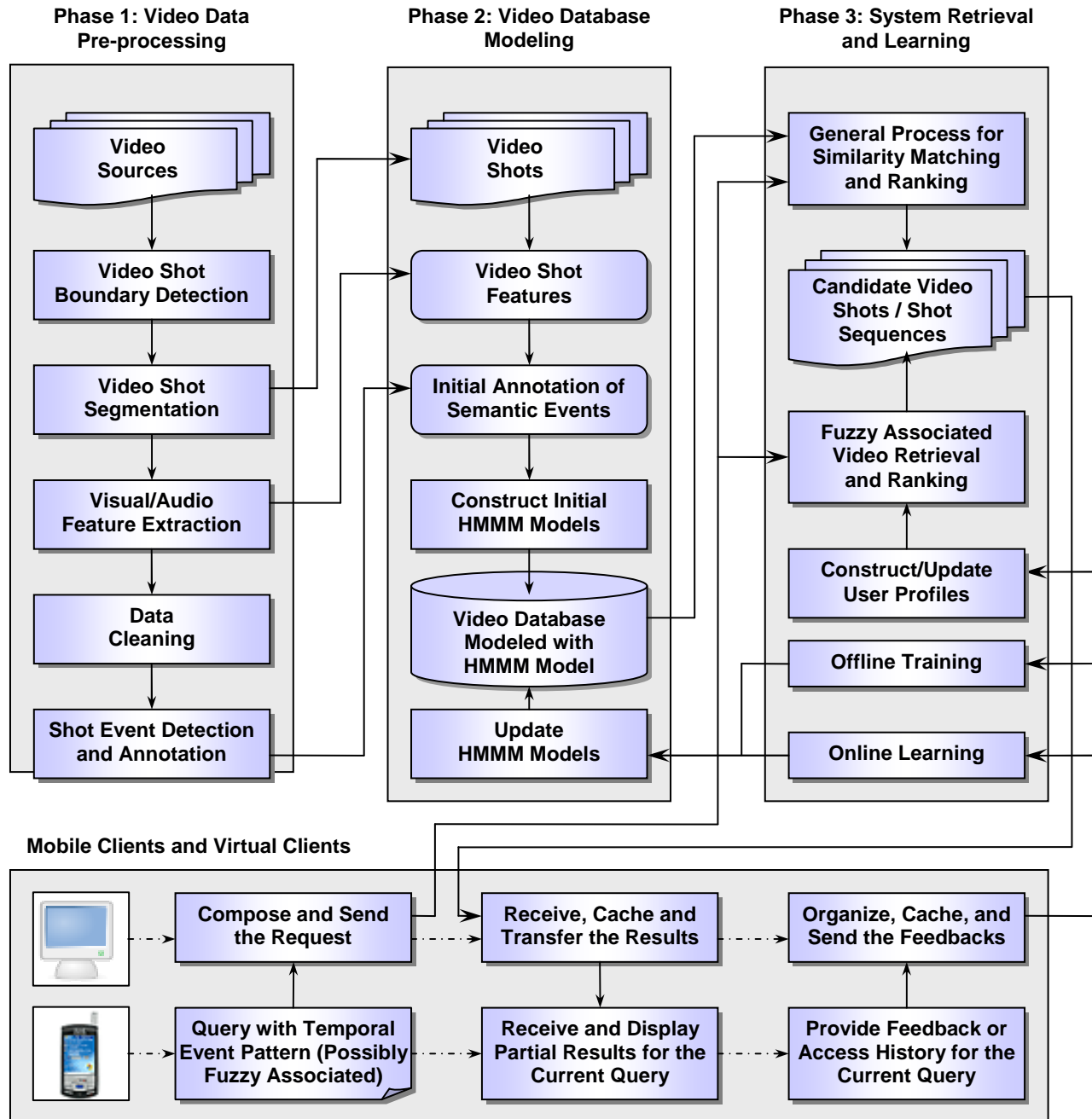


Figure V-9. Overall framework of mobile-based video retrieval system

5.5.5 HMMM-based User Profile

One of the major challenges in multimedia retrieval is to identify and learn personalized user interests. The underlying reason for this challenge is that a user's query interests can hardly be expressed precisely by using query examples or keywords. In addition, different users tend to

have diverse opinions or perceptions towards even the same query and intend to seek for different results and rankings. For example, given a query for soccer goal shots, different users may be interested in different retrieval requirements:

- followed by a corner kick;
- in the female soccer videos;
- with exciting screams;
- etc.

In this research study, the constructed HMMM model can serve as a “general user profile” which represents common knowledge of the multimedia data and the related semantics. On the other hand, an HMMM-based “individual user profile” is also constructed for each mobile user, which is mainly constructed based on learning the individual user’s query history and access patterns. The definition is described as follows:

Definition V-1: An HMMM-based User Profile is defined as a 4-tuple: $\Phi = \{\tau, \hat{A}, \hat{B}, \hat{O}\}$,

where

- τ : represents the identification of a mobile user;
- \hat{A} : Affinity profile, which incorporates a set of affinity matrices $\hat{A} = \{\hat{A}_n^g\}$ that describes the relationships between the user accessed media objects and all the media objects. Here $1 \leq n \leq d$, and $1 \leq g \leq |\lambda_n^g|$.
- \hat{B} : Feature profile, which represents the feature measurements based on the positive feedbacks of certain events and/or event patterns;
- \hat{O} : Feature weight profile, which consists of the feature weights obtained by mining and evaluating the users’ access history.

5.5.5.1 Affinity Profile

The affinity profile \hat{A} is designed to model the affinity relationships among the multimedia objects that are related to users' historic query/feedback logs. The proposed solution tries to minimize the memory size that a user profile would occupy. As illustrated in Figure V-10, the system will check the query logs and access histories for the purpose of constructing the affinity profile.

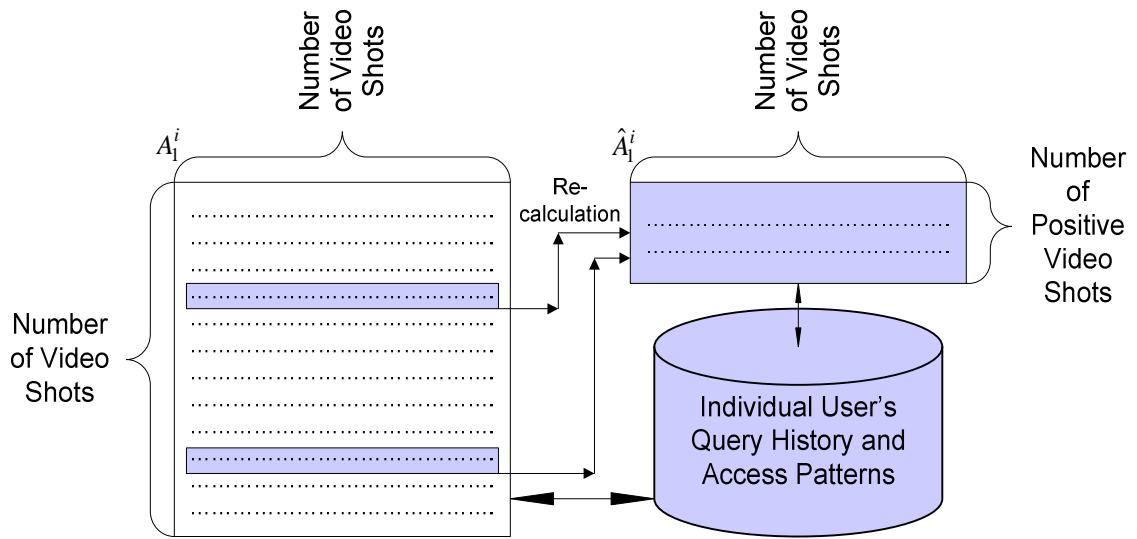


Figure V-10. Generation of individual user's affinity profile

Taking affinity matrix A_1^j as an example, it describes the temporal-based affinity relationships among the video shots of the j^{th} video. The system will find the “positive” video shots that the user has accessed before and the corresponding rows are extracted from the original matrix (A_1^j). These values are then updated and used to create a new matrix \hat{A}_1^j in this user's affinity profile. Similarly, in the second-level user affinity profiles, the rows represent the accessed videos which include at least one positive video shot, and the column includes all the videos in the cluster.

For a mobile user, his/her query log and access history include the set of issued queries as well as the associated positive feedbacks. We define matrices UF_n to capture the individual user's access frequencies for the n^{th} level objects in the multilevel HMMM database. For example, let $UF_1(i, j)$ represent the number of positive feedback patterns which contain the temporal sequence as $\{\dots, S_1^g(i), S_1^g(j), \dots\}$. $UF_2(i, j)$ denotes the number of positive patterns where both video v_i and v_j are accessed together, and $UF_3(i, j)$ denotes the number of positive patterns which contain the video shots across both video cluster CC_i and CC_j . For the affinity matrix \hat{A}_n^g , the corresponding affinity profile is computed and updated as below.

$$\hat{A}_n^g(i, j) = \frac{A_n^g(i, j) \times (1 + UF_n(i, j))}{\sum_x A_n^g(i, x) \times (1 + UF_n(i, x))}. \quad (\text{V-12})$$

Where $1 \leq n \leq d$, $d = 3$, and $S_n^g(x)$ represents all the possible states in the same MMM model with $S_n^g(i)$ and $S_n^g(j)$. In addition, when $n = 1$, states $S_1^g(i)$ and $S_1^g(j)$ also need to follow the certain temporal sequence where $T_{S_1^g(i)} \leq T_{S_1^g(j)}$.

5.5.5.2 Feature Profile

Feature profiles are constructed to describe the distinct searching interests for each user by modifying the target feature values. As discussed in our previous paper [Zhao06a], an event feature matrix B_1^i was computed based on the annotated events. However, the annotated results may not be fully correct or complete. Further, the users may have their particular interests when looking for a certain event. To address these issues, a feature profile \hat{B}_1 is proposed. Specifically, in the profile matrix \hat{B}_1 , each row represents an event, and each column represents a feature. Let f_k represent the k^{th} feature, where $1 \leq k \leq K$, and K is the total number of features. Given \tilde{z}_m as a subset of all the positive shots with event type e_m , and letting $B_1(\tilde{z}_m(i), f_k)$ denote the feature

values for video shot $\tilde{z}_m(i)$, Equation V-13 defines \hat{B}_1 . If there is no positive shot with event type e_m ($|\tilde{z}_m|=0$), the corresponding row is copied from the event feature matrix B_1 in the constructed HMMM model.

$$\hat{B}_1(e_m, f_k) = \begin{cases} \frac{\sum_{i=1}^{|\tilde{z}_m|} B_1(\tilde{z}_m(i), f_k)}{|\tilde{z}_m|}, & \text{where } |\tilde{z}_m| \geq 1, 1 \leq |\tilde{z}_m|, 1 \leq k \leq K; \\ B_1(e_m, f_k), & \text{where } |\tilde{z}_m| = 0, 1 \leq k \leq K. \end{cases} \quad (\text{V-13})$$

5.5.5.3 Feature Weight Profile

In the literature, many approaches used Euclidean distance, relational coefficients, etc. to determine the similarity measure between two data items in terms of their feature values. However, the effectiveness of different features might vary greatly from each other in expressing the media content, so it is essential to apply feature weights in measuring the similarity between multimedia data objects. In HMMM, a matrix $O_{1,2}$ is used to describe the importance of lower level visual/audio features F_1 when describing the event concepts F_2 . The initial values of all its entries are set to be equal, which indicates that all the features are considered to be equally important before any user feedback is collected and any learning process is performed. Once we obtain the annotated event set, the feature weights will then be updated as introduced in our previous work [Zhao06a].

This research mainly focuses on the feature weight profile, which is constructed based on the mobile user's individual access and feedback histories. As users can provide positive feedback on their favorite video shots, the basic idea is to increase the weight of similar features among the positive video shots, while decreasing the weight of dissimilar features among them. For this purpose, we use the standard deviation $Std(e_m, f_k)$ to measure the distribution condition of feature f_k ($1 \leq k \leq K$) on the video shots containing event e_m ($1 \leq m \leq C$), where C represents

the number of distinct event concepts. A large standard deviation indicates greater scatter of the data points. Accordingly, when there is more than one positive shot for event e_m , ($\tilde{z}_m(i) > 1$), the value of $1/Std(e_m, f_k)$ can be employed to measure the similarity of the features, which in turn indicates the importance of this feature in terms of evaluating event e_m . However, this solution does not apply when there is no positive shot or only one positive shot ($\tilde{z}_m(i) \leq 1$), so we would borrow the corresponding feature weights for event e_m from matrix $O_{1,2}$. The feature weight profile is thus defined as follows.

$$Std(e_m, f_k) = \sqrt{\frac{\sum_{i=1}^{|\tilde{z}_m|} (B_1(\tilde{z}_m(i), f_k) - \hat{B}_1(e_m, f_k))^2}{|\tilde{z}_m| - 1}}, \quad (V-14)$$

Where $\tilde{z}_m(i) > 1$, $1 \leq m \leq C$, and $1 \leq k \leq K$.

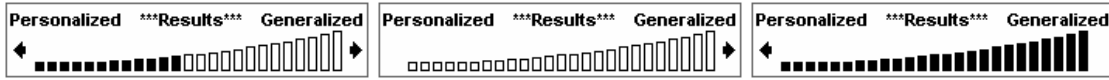


Figure V-11. Fuzzy weight adjustment tool (a) generalized recommendation; (b) personalized recommendation; (c) fuzzy associated recommendation

$$\hat{O}_{1,2}(e_m, f_k) = \begin{cases} \frac{1/Std(e_m, f_k)}{\sum_{j=1}^K (1/Std(e_m, f_j))}, & \text{where } |\tilde{z}_m| > 1, 1 \leq m \leq C, \text{ and } 1 \leq k \leq K \\ O_{1,2}(e_m, f_k), & \text{where } |\tilde{z}_m| \leq 1, 1 \leq m \leq C, \text{ and } 1 \leq k \leq K \end{cases} \quad (V-15)$$

5.5.6 Fuzzy Associated Retrieval

Fuzzy logic has been noted for its ability to describe and model vagueness and uncertainty, which are inherent in multimedia information retrieval. In this proposed framework, fuzzy logic is adopted to model the uncertainty of users' retrieval interests. Specifically, users are allowed to make their choices of retrieving content based solely on general user perceptions, personalized interests, or anywhere in between, which leads to a different level of tradeoff

between retrieval accuracy and processing speed. We use a fuzzy weight parameter $\rho \in [0,1]$ to measure the uncertainty that users may pose when issuing the video queries. As shown in Figure V-11, we use an interactive gauge on the mobile device interface for the users to adjust ρ . By choosing the personalized interest (as shown in Figure V-11(b)), $\rho = 0$, the system will evaluate the user's profile and retrieve the video clips based on the learned knowledge with respect to the user's previous access patterns. On the other hand, if the generalized recommendation mode is selected (Figure V-11(a)), i.e., $\rho = 1$, the system will comply with the common knowledge learned from the complete query log collected across different users. Therefore, the most popular video clips will be retrieved with higher ranks. Assuming that we have already performed video clustering [Zhao06b] through the database, the generalized recommendation mode is normally more efficient as satisfactory results can generally be retrieved by checking the related clusters.

Let $Q = \{e_1, e_2, \dots, e_C\} (T_{e_1} \leq T_{e_2} \leq \dots \leq T_{e_C})$ be a query pattern and $\bar{s}_t \in S$ be a candidate video shot for the event e_t ($1 \leq t \leq C$); the system can adjust the retrieval algorithm and provide three kinds of recommendations, namely, generalized recommendation, personalized recommendation, and fuzzy weighted recommendation according to the fuzzy weight issued by the user. The details are addressed below.

5.5.6.1 Generalized Recommendation

When a generalized recommendation (Figure V-11(a)) is selected, the matrices in the constructed HMMM model will be used as the common user profile to perform the stochastic retrieval process. First, as shown in Equation (V-16), the weighted Euclidean distance $dis(\bar{s}_t, e_t)$ is calculated by adopting the general feature weight $O_{1,2}(e_t, f_k)$, which is then used to derive the similarity measurements (see Equation (V-17)). Here, $B'_1(e_t, f_k)$ denotes the extracted mean

value of feature f_k with respect to event e_t based on the learned general users' common knowledge.

$$dis(\bar{s}_t, e_t) = \sqrt{\sum_{k=1}^K (O_{1,2}(e_t, f_k) \times (B_1(\bar{s}_t, f_k) - B_1'(e_t, f_k))^2)}, \quad (V-16)$$

Where $\bar{s}_t \in S_1, 1 \leq k \leq K, 1 \leq t \leq C$.

$$sim(\bar{s}_t, e_t) = \frac{1}{1 + dis(\bar{s}_t, e_t)}, \quad (V-17)$$

Where $\bar{s}_t \in S_1, 1 \leq t \leq C$.

Next, the edge weights are calculated based on Equations (V-18) and (V-19). When $t = 0$, the initial edge weight are calculated by using the initial state probability and similarity measure between state s_t and event e_t . It is worth mentioning that the system tries to evaluate the optimized path to access the next possible video shot state which is similar to the next anticipated events. Therefore, the edge weight from state s_t to s_{t+1} ($1 \leq t \leq C$) is calculated by adopting the affinity relationship as well as the similarity between the candidate shot s_{t+1} with the event concept e_{t+1} .

$$w_1(\bar{s}_t, e_1) = \Pi_1(\bar{s}_1) \times sim(\bar{s}_1, e_1). \quad (V-18)$$

$$w_{t+1}(\bar{s}_{t+1}, e_{t+1}) = w_t(\bar{s}_t, e_t) \times A_1(\bar{s}_t, \bar{s}_{t+1}) \times sim(\bar{s}_{t+1}, e_{t+1}), \text{ where } 1 \leq t \leq C. \quad (V-19)$$

After one round of traversal, the system retrieves a sequence of video shots R_y which match the desired event pattern Q . The next step would be the calculation of the similarity score. Here, $SS(Q, R_y)$ is computed by summing up all the edge weights, where a greater similarity score indicates a closer match.

$$SS(Q, R_y) = \sum_{t=1}^C w_t(\bar{s}_t, e_t) \quad (V-20)$$

5.5.6.2 Personalized Recommendation

In case the user prefers a personalized recommendation (Figure V-11(a)), the overall process steps are similar except that the matrices used are mainly from HMMM-based user profiles.

$$\hat{dis}(\bar{s}_t, e_t) = \sqrt{\sum_{k=1}^K (\hat{O}_{1,2}(e_t, f_k) \times (B_1(\bar{s}_t, f_k) - \hat{B}_1(e_t, f_k))^2)} \quad (V-21)$$

$$\hat{sim}(\bar{s}_t, e_t) = \frac{1}{1 + \hat{dis}(\bar{s}_t, e_t)} \quad (V-22)$$

Where $\bar{s}_t \in S_1, 1 \leq t \leq C$.

When calculating the edge weight $\hat{w}_{t+1}(\bar{s}_{t+1}, e_{t+1})$, there could be two conditions. If the candidate video shot has been accessed by this user and marked as ‘‘Positive’’ ($\bar{s}_{t+1} \in R_y$), the user’s affinity profiles should include this video shot and, therefore, the formula takes the affinity value from the user’s personal affinity profile (\hat{A}_1). Otherwise, there is no record for the video shot in the user profile ($\bar{s}_{t+1} \notin R_y$), and the system will pick the value from the affinity matrices (A_1) of the constructed HMMM.

$$\hat{w}_1(\bar{s}_t, e_t) = \Pi_1(\bar{s}_1) \times \hat{sim}(\bar{s}_1, e_1) \quad (V-23)$$

$$\hat{w}_{t+1}(\bar{s}_{t+1}, e_{t+1}) = \begin{cases} \hat{w}_t(\bar{s}_t, e_t) \times \hat{A}_1(\bar{s}_t, \bar{s}_{t+1}) \times sim(s_{t+1}, e_{t+1}), & \text{where } 1 \leq t < C, \bar{s}_{t+1} \in R_y \\ \hat{w}_t(\bar{s}_t, e_t) \times A_1(\bar{s}_t, \bar{s}_{t+1}) \times sim(s_{t+1}, e_{t+1}), & \text{where } 1 \leq t < C, \bar{s}_{t+1} \notin R_y \end{cases} \quad (V-24)$$

$$\hat{SS}(Q, R_y) = \sum_{t=1}^C \hat{w}_t(\bar{s}_t, e_t) \quad (V-25)$$

5.5.6.3 Fuzzy Associated Recommendation

Alternatively, if the user is uncertain and thus chooses a fuzzy weight parameter $\rho \in (0,1)$ to describe his/her interest (as illustrated in Figure V-11(c)), the system will make the

adjustment on the edge weights and accordingly, the optimized path and the similarity score would possibly be changed as defined in the following equations.

$$\tilde{w}_t(\bar{s}_t, e_t) = \rho \times w_t(\bar{s}_t, e_t) + (1 - \rho) \times \hat{w}_t(\bar{s}_t, e_t), \text{ where } 1 \leq t \leq C. \quad (\text{V-26})$$

$$\tilde{S}(Q, R_y) = \sum_{t=1}^C \tilde{w}_t(\bar{s}_t, e_t). \quad (\text{V-27})$$

After getting the candidate video shot sequences, they will be ranked based on their similarity scores and sent back to the client.

5.5.7 Implementation and Experiments

A mobile-based soccer video retrieval system is developed based on the proposed MoVR framework, which consists of the following components:

- A soccer video database is constructed and maintained in the server-side by using PostgreSQL [PostgreSQL]. Totally, 45 soccer videos along with 8977 segmented video shots and corresponding key frames are stored and managed in the database.
- Server-side engine is implemented by using C++. This module contains not only the searching and ranking algorithms, but also a set of other computationally intensive techniques, including video shot segmentation, HMMM database modeling, user profile generation and updating, etc.
- The virtual client application is implemented with Java J2SE [J2SE]. It works as middleware between server engine and mobile clients, where data communication is mainly fulfilled by using UDP and TCP.
- The user interface on the mobile device is developed by using Sun Java J2ME [J2ME] Wireless Toolkit [JavaWTK]. We try to make it portable, flexible, and

user friendly with simple but effective functions. The user can easily issue event/pattern queries, navigate key frames, play interested video clips, and provide feedbacks.

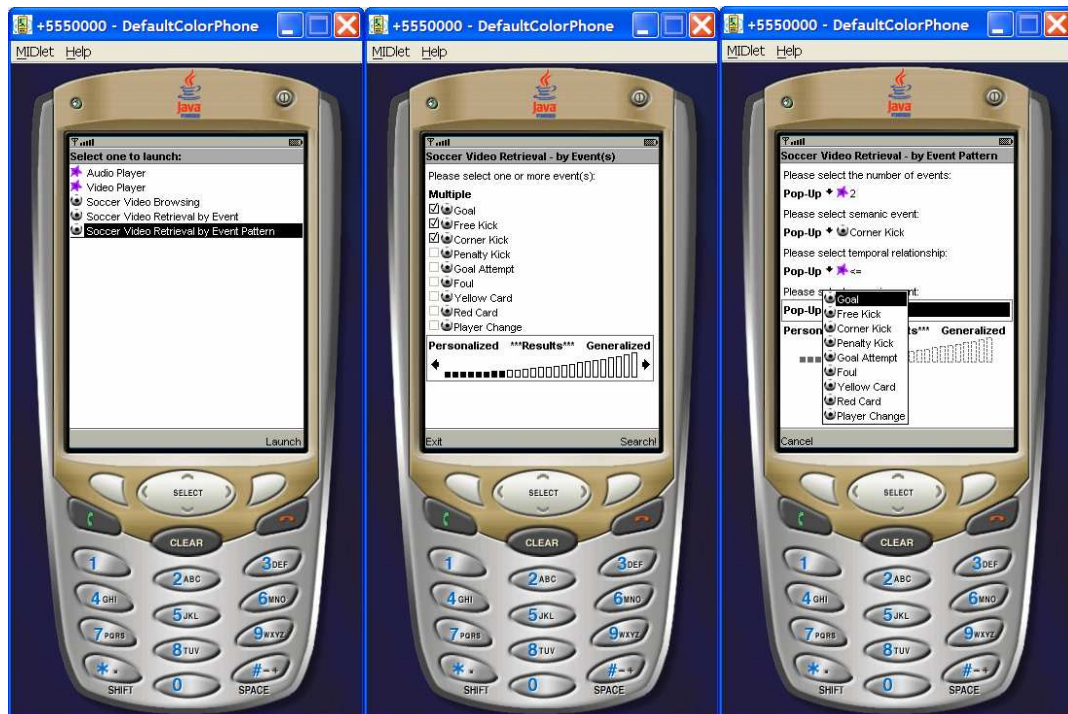


Figure V-12. Mobile-based soccer video retrieval interfaces (a) initial choices (b) retrieval by event (c) retrieval by pattern

Figure V-12 and Figure V-13 show the user query interfaces of the MoVR soccer video retrieval system.

- In Figure V-12(a), the initial choices are displayed, which include “Soccer Video Browsing,” “Soccer Video Retrieval by Event,” and “Soccer Video Retrieval by Event Pattern,” etc. The user can use the upper-center button to move up/down to select the target menu and then push the left-upper button to launch the selected application.

- Figure V-12(b) shows the query interface for the single-event queries. It allows the user to choose one or more events with no temporal constraints. For instance, in this figure, the user chooses “Goal,” “Free Kick,” and “Corner Kick,” which means the video clips with either one of these three events are of interest. Under the event list, there is a gauge control which allows the user to change the fuzzy weight parameter between two extremes: personalized recommendations and generalized recommendations. The upper-left button can be used to exit this component and go back to the main menu, while the upper-right button can be used to issue the query.
- Figure V-12(c) illustrates the interface for the temporal event pattern retrieval. The user can use the popup lists to choose the event number to define the size of the query pattern. Then it is allowed to choose the events one by one, along with the temporal relationship between two adjacent events. Taking this figure as an example, the user first sets the event number as 2, and then chooses the pattern as “Corner Kick \leq Goal,” which means that the user wants to search for the video clips with a “Corner Kick” followed by a “Goal”. These two events could also occur in the same video shot (when temporal relationship is set to “=”), which can be called a “Corner Goal”.

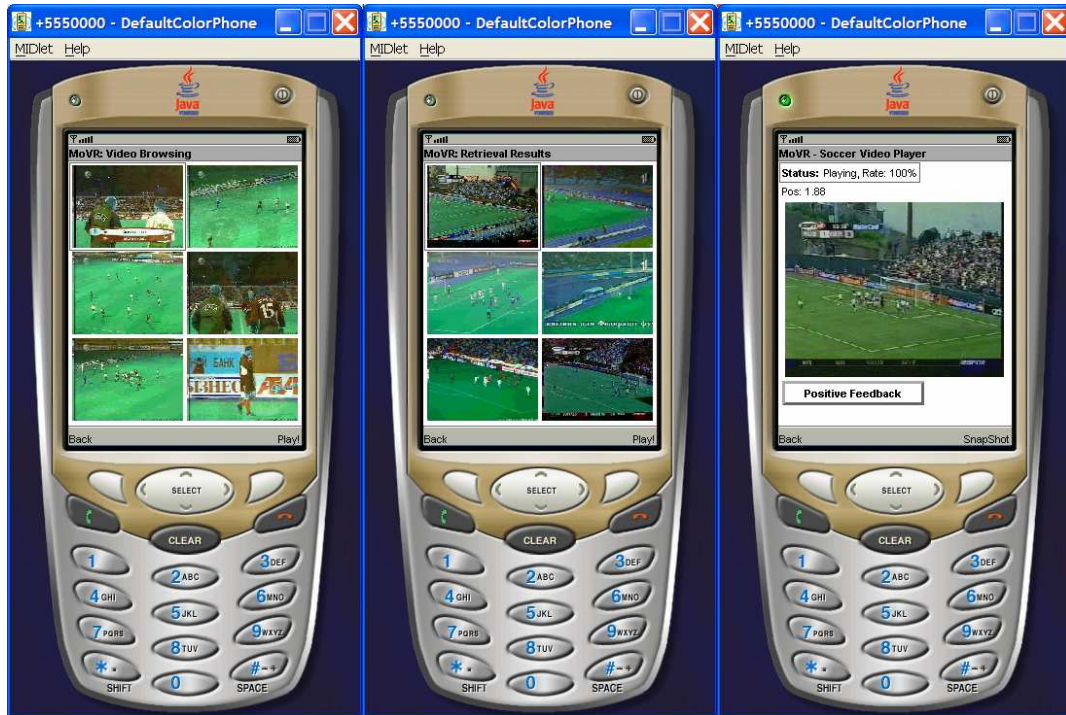


Figure V-13. Mobile-based soccer video retrieval results (a) video browsing results (b) video retrieval results (c) video player

- The returned key frames are displayed as shown in Figure V-13(a) and Figure V-13(b). Due to the limitation of screen display size, only six key frames are shown in the first screen, where each key frame represents the first frame for each of the returned video clips. Users can choose their key frame of interest and then trigger the “Play!” button to display the corresponding video clip, which may include one (for event query) or more video shots (for event pattern query). Note that Figure V-13(a) shows the video browsing results with the key frames displayed for consecutive video shots in one video. Figure V-13(b) illustrates the results for an event query targeting the video shots with corner kicks. These video shots are retrieved from different soccer videos and are ranked from left to right, and from top to down based on their similarity scores.

- The video player interface, which is shown in Figure V-13(c), contains a button called “Positive Feedback,” which can be selected to send back a positive feedback if the user is satisfied with the current video clip. A “Snapshot” functionality is also provided such that the user can capture a video frame from the video.

Table V-2. Average accuracy for the different recommendations

ID	Query	Generalized Recommendations	Fuzzy Weighted Recommendations	Personalized Recommendations
1	Goal	30.6%	38.9%	61.1%
2	Free Kick	19.4%	50.0%	58.3%
3	Corner Kick	27.8%	58.3%	72.2%
4	Goal < Goal	36.1%	55.6%	86.1%
5	Free Kick <= Goal	36.1%	50.0%	66.7%
6	Corner Kick <= Goal	33.3%	47.2%	72.2%
7	Free Kick < Corner Kick	25.0%	36.1%	52.8%
8	Corner Kick < Free Kick	36.1%	44.4%	52.8%
9	Corner Kick <= Goal < Free Kick	16.7%	38.9%	63.9%
10	Free Kick <= Goal < Goal	25.0%	36.1%	55.6%

In our experiments, a total of 300 historical queries were used for the construction of HMMM model, where the system learned the common knowledge from the general users. As shown in Table V-2, we performed the test for ten sets of distinct queries, including three single-event queries, five two-event pattern queries, and two three-event pattern queries. For example, Query 9 “Corner Kick <= Goal < Free Kick” means a pattern with a corner kick, followed by a goal, and then a free kick, where corner kick and goal may possibly occur within the same video shot. For each of these ten queries, three sets of tests are performed and each of them represents distinct user interests. In each of these tests, the user profile is constructed based on 30 historical

queries and all three possible recommendation methods are testified. Twelve top-ranked video clips shown in the first two screens (with six results each) are checked, which is called “scope”. Thus, “accuracy” here is defined as the percentage of the number of user satisfied video clips within the scope. Finally, the average accuracy is computed based on these tests.

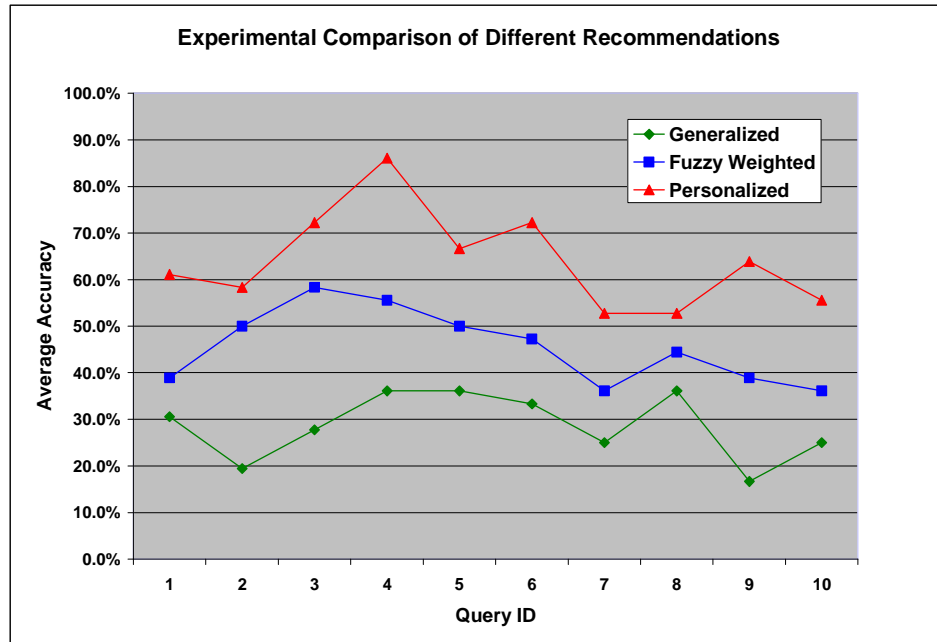


Figure V-14. Experimental comparison of different recommendations

Figure V-14 illustrates the comparison of the average accuracy values across these three kinds of recommendations. As we can see from this figure, the results based on “Generalized” recommendation have the lowest average accuracy; ”Personalized” recommendation offers the best results, while “Fuzzy Weighted” recommendation has the performance in between. In general, the personalized recommendation does offer better results by learning individual user preferences by using the HMMM-based profile. Meanwhile, though the generalized recommendation may not fully satisfy the individual users, it represents the common knowledge learned from the general users and requires shorter processing time. By adopting fuzzy weighted recommendations, the users are offered more flexibility in video retrieval.

5.5.8 Summary

With the proliferation of mobile devices and multimedia data sources, there is a great need for effective mobile multimedia services. However, with their unique constraints in display size, power supply, storage space as well as processing speed, the multimedia applications in mobile devices encounter great challenges. In this section, we present MoVR a user adaptive video retrieval framework in the mobile wireless environment. While accommodating various constraints of the mobile devices, a set of advanced techniques are developed and deployed to address essential issues, such as the semantic gap between low-level video features and high-level concepts, the temporal characteristics of video events, and individual user preference, etc. Specifically, a Hierarchical Markov Model Mediator (HMMM) scheme is proposed to model various levels of media objects, their temporal relationships, the semantic concepts, and high-level user perceptions. The HMMM-based user profile is defined, which is also integrated seamlessly with a novel learning mechanism to enable the “personalized recommendation” for an individual user by evaluating his/her personal histories and feedbacks. In addition, the fuzzy association concept is employed in the retrieval process such that the users gain control of the preference selections to achieve reasonable tradeoff between the retrieval performance and processing speed. Furthermore, to improve the processing performance and enhance the portability of client-side applications, storage consumption information and computationally intensive operations are supported in the server side; whereas the mobile clients are solely required to maintain the minimized size of data to enable the retrieval process. The virtual clients are designed to perform as a middleware between server applications and mobile clients. This design helps to reduce the storage load of mobile devices and to provide greater accessibility with their cached media files. Finally, a mobile-based soccer video retrieval system is developed and tested to demonstrate the effectiveness of the proposed framework.

CHAPTER VI. SECURITY SOLUTIONS FOR MULTIMEDIA SYSTEMS

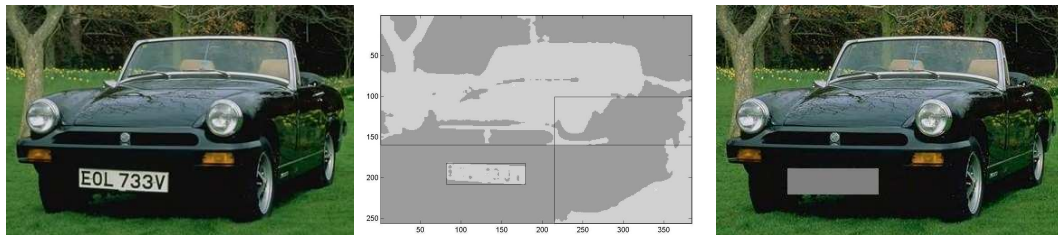
This chapter discusses the security module of DIMUSE, which is responsible for the authority checking to guarantee the security assurance of the multimedia information retrieved and used by the distributed multimedia applications. In order to satisfy the intricate requirements of multimedia security, a framework called SMARXO is presented to support multi level security access control by combining several imperative techniques: Role-Based Access Control (RBAC), eXtensible Markup Language (XML), and Object-Relational Database Management System (ORDBMS).

6.1 Introduction

With the rapid development of various multimedia technologies, more and more multimedia data are generated in the medical, commercial, and military fields, which may include some sensitive information that should not be accessed by, or can only be partially exposed to, general users. Therefore, user-adaptive multimedia data access control has become an essential topic in the areas of multimedia database design and multimedia application development for information security purposes. RBAC (Role-Based Access Control) is a good candidate for user authorization control. However, most of the existing RBAC models mainly focus on document protection without fully considering all the possible environmental constraints. Although it is claimed that some extended models are able to offer protection on multimedia files, there are still some unsolved problems. For instance, Figure VI-1(a) shows an image which can be accessed, but the “plate” object inside should not be displayed. If a user requests this image, he/she can only view the partial image as shown in Figure VI-1(c).

The focal goal of our security research can be outlined as constructing a framework to control the access to multimedia applications, files, and furthermore the visual/audio objects or segments embedded in the multimedia data. In this chapter, we develop a framework named

SMARXO (Secured Multimedia Application by adopting RBAC, XML and ORDBMS). Several significant techniques are proficiently mixed in SMARXO to satisfy the complicated multimedia security requirements. First, efficient multimedia analysis mechanisms can be used to acquire the meaningful visual/audio objects or segments. Second, XML and object-relational databases are adopted such that proficient multimedia content indexing can be easily achieved. Third, we upgrade and embed a dominant access control model which can be tailored to the specific characteristics of multimedia data. XML is also applied to organize all kinds of security-related roles and policies. Finally, and most importantly, these techniques are efficiently organized such that multi-level multimedia access control can be achieved in SMARXO without any difficulty.



**Figure VI-1. Example of image object-level security
(a) original image (b) segmentation map (c) hiding a portion of the image**

6.2 SMARXO Architecture

There are three phases available in order to build up the complete security verification architecture for multimedia applications. Figure VI-2 illustrates the SMARXO architecture. The multimedia data, extracted features, and furthermore the XML documents are all organized in the ORDBMS. Once a user, including the administrator, logs in to the system and requests the multimedia data, the security checker verifies the user's identification and the related permission. The multimedia manager responds based on the security checking results. The source multimedia data may need to be processed in order to hide the object-level or scene/shot-level information. In addition, through this framework, the administrators are capable of creating, deleting, and modifying the user roles, object roles, temporal roles, IP address roles, and security policies.

Since all the protection-related information is managed by XML, security information retrieval becomes very convenient.

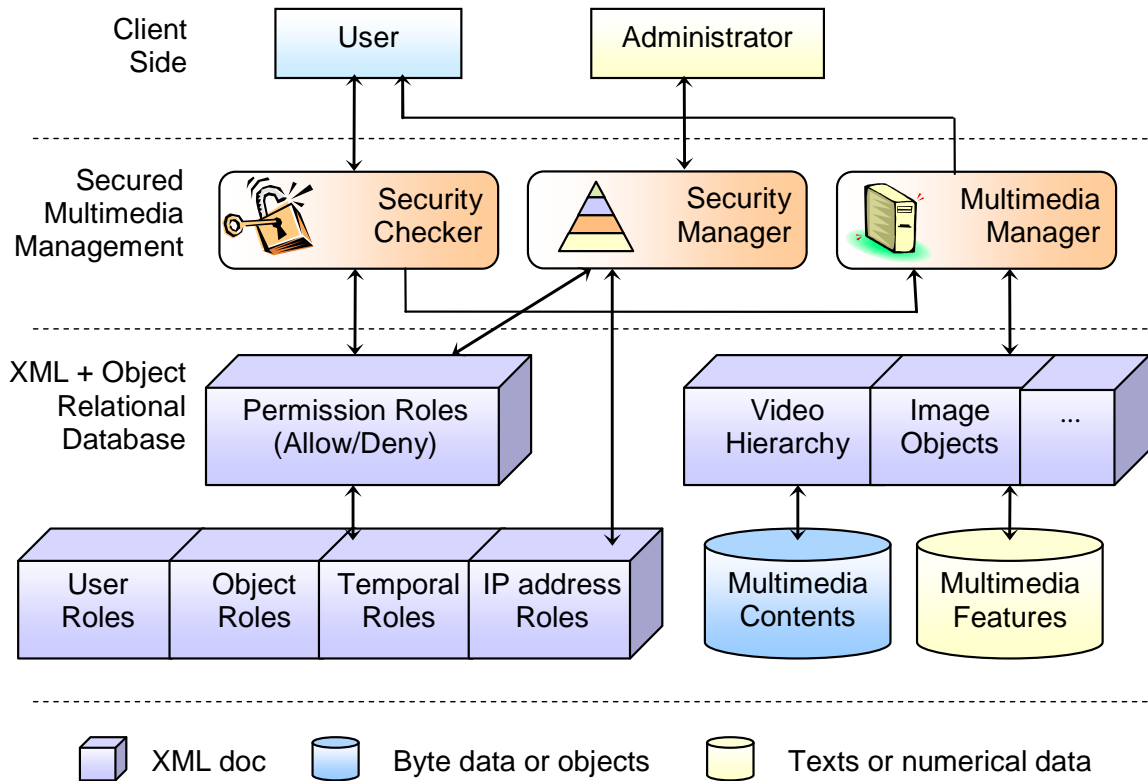


Figure VI-2. SMARXO architecture

6.3 Multimedia Access Control

The traditional RBAC methods need to be extended to perform superior access control functionalities such as the temporal and IP address control, object-level and scene/shot-level access control, etc. Based on the formal definition of traditional RBAC in [Sandhu96], the extended formal definitions are given in Figure VI-3. Compared with the traditional RBAC model, we also introduce the object roles, temporal roles, and IP address roles. The associated rules are defined such that these advanced roles can be combined to perform inclusive access control.

Sets:

U :	Users	(*) O :	Objects
Ru :	User Roles	(*) Ro :	Object Roles
S :	Sessions	(*) Rt :	Temporal Roles
P :	Permissions	(*) Ri :	IP address Roles

Rules:

- 1) $UA \subseteq U \times Ru$: user-role assignment
- 2) $RH \subseteq Ru \times Ru$: a partial order of role hierarchy
- 3) $PA \subseteq P \times Ru$: a basic permission-user role assignment
- 4) (*) $OA \subseteq O \times Ro$: object-role assignment
- 5) (*) $OP \subseteq P \times Ro$: a permission-object assignment
- 6) (*) $R \subseteq Ru \times Rt \times Ri$: an assembled role set with environmental constraints
- 7) (*) $OPA \subseteq OP \times R$: an advanced permission-role assignment
- 8) $user: S \rightarrow U$, a function mapping a session to a user
- 9) $u_roles: S \rightarrow 2^{Ru}$, a basic function mapping a session to a set of user roles
- 10) (*) $roles: S \rightarrow 2^R$, an advanced function mapping a session to a set of roles
- 11) $permissions: Ru \rightarrow 2^P$, mapping a user role to a set of permissions
- 12) $permissions': Ru \rightarrow 2^P$, mapping a user role to a set of permissions with role hierarchies
- 13) (*) $permissions'': R \rightarrow 2^{OP}$, mapping an assembled role to a set of permissions
- 14) (*) $permissions''': R \rightarrow 2^{OP}$, mapping an assembled role to a set of permissions with role hierarchies
- 15) $permissions(r) = \{p: P \mid (r, p) \in PA\}$
- 16) $permissions'(r) = \{p: P \mid \exists r' \leq r \cdot (r', p) \in PA\}$
- 17) (*) $permissions''(r) = \{p: OP \mid (r, p) \in OPA\}$
- 18) (*) $permissions'''(r) = \{p: OP \mid \exists r' \leq r \cdot (r', p) \in OPA\}$

(Note: the ones marked with * are advanced features of SMARXO)

Figure VI-3. Extended RBAC definitions in SMARXO

6.3.1 Multimedia Indexing Phase

In order to support multi-level security, the multimedia data are required to be stored hierarchically. For instance, by applying image segmentation techniques on Figure VI-1(a), the corresponding segmentation map (as shown in Figure VI-1(b)) can be achieved. Each extracted object is bounded with a rectangle. The extraction results may help people identify the meaningful objects and compute the associated bounding boxes. Both the original image and the image object information can be stored in the ORDBMS. If a specific security policy requires some portions of the target image to be hidden from the user, the system can retrieve the sub-object's attributes and process the original image to hide those portions (e.g., the protected "plate" in Figure VI-1(c)). XML can be adopted to index the image object information by a 6-tuple element: $\langle o_id, o_name, o_x, o_y, o_width, o_height \rangle$, which are the object id, object description, x and y coordinates of the top-left point, and the object width and height, respectively. Such an example can be found in Figure VI-4(a).

<p>(a)</p> <pre> <ImageObjects> <Image imgid='i001'> <Object o_id='i001o01'> <o_name>TAG</o_name> <o_x>40</o_x> <o_y>80</o_y> <o_width>8</o_width> <o_height>50</o_height> </Object> <Object o_id='i001o02'> <o_name>CAR</o_name> ... </Object> </Image> ... </ImageObjects> </pre>	<p>(b)</p> <pre> <VideoHierarchy> <Video v_id='v01'> <Event e_id='e01'> <Scene c_id='c01'> <Shot s_id='s01'> <frame_s>1</frame_s> <frame_e>89</frame_e> </Shot> ... </Scene> ... </Event> ... </Video> ... </VideoHierarchy> </pre>
--	--

Figure VI-4. XML examples of multimedia hierarchy
(a) example for image objects (b) example for video hierarchy

By utilizing video decoding, shot detection, and scene detection techniques, the specific video can be automatically segmented and diverse levels of the video objects can be achieved: frame, shot, scene, and event. For the purpose of video indexing, we can furthermore apply XML to store this kind of video hierarchy information. As shown in Figure VI-4(b), the start frame and end frame numbers of the shots are stored to mark the segmentation boundaries. In SMARXO, a “shot” is treated as the fundamental unit to store the video data for efficiency purposes. Hence, shot-level security can be performed easily by displaying the accessible shots and skipping those prohibited shots. In addition, users are allowed to manually identify their target multimedia objects or segments by giving the corresponding parameters.

6.3.2 Security Modeling Phase

In most multimedia applications, a request behavior can be briefly recognized by a 4-tuple: $\langle who, what, when, where \rangle$. The meaning of this request is that some user requests some data at some time by using some computer. As discussed before, most of the related research work can only control accesses by the “*who*” and “*what*” attributes. Few models can support security verification on the “*when*” attribute. By contrast, our framework supports all of them.

6.3.2.1 User Roles

User roles, also recognized as “Subject roles”, are the most fundamental feature of RBAC. In addition to the basic requirements, SMARXO supports one more specific feature on user authorization. When the administrator creates a new user account, he/she can choose the default property of this user from two options. One is to initially grant all the access abilities to this user, and then assign the roles which deny this user’s access to some object. The other option is to disable the user from accessing by default. Then the permission roles can be granted to this account. In Figure VI-5(a), the user “Bailey” in the “Professor” group is assigned the default value “Allow”; while the user “Smith” in the “Student” group is assigned the default property “Deny.”.

<p>(a)</p> <pre> <SubjectRoles> <UserGroup default='Allow'> <Group g_id='Professor'> <User u_id='Bailey'> <Password>abc</Password> </User> ... </Group> </UserGroup> <UserGroup default='Deny'> <Group g_id='Student'> <User u_id='Smith'> <Password>321</Password> </User> ... </Group> ... </UserGroup> </SubjectRoles> </pre>	<p>(b)</p> <pre> <ObjectRoles> <o_group id='Shots_a' > <scene s_id='s02'> <shot>2</shot> <shot>3</shot> <shot>4</shot> ... </scene> ... </o_group> <o_group id='Shots_b'> <shot>6</shot> <shot>12</shot> ... </o_group> ... </ObjectRoles> </pre>
---	--

**Figure VI-5. XML examples of the fundamental roles
(a) example of subject roles (b) example of object roles**

6.3.2.2 Object Roles

Sometimes, the user may not be able to access one or more segments/objects of a multimedia file. However, he/she should be able to access other parts of this file. The object roles are facilitated to satisfy this requirement. Figure VI-6 illustrates a video shot sequence stored in the database. User A cannot access shots 2, 3, 4; while User B cannot access shots 6 and 12. However, A and B should be allowed to view the other shots of this video except for their prohibited segments. SAMRXO supports this kind of access control by modeling both the object roles and multimedia hierarchy information. Figure VI-5(b) depicts an XML example for the object roles. Furthermore, in order to efficiently organize plentiful object roles, we introduce the object-role hierarchy which is defined as follows.

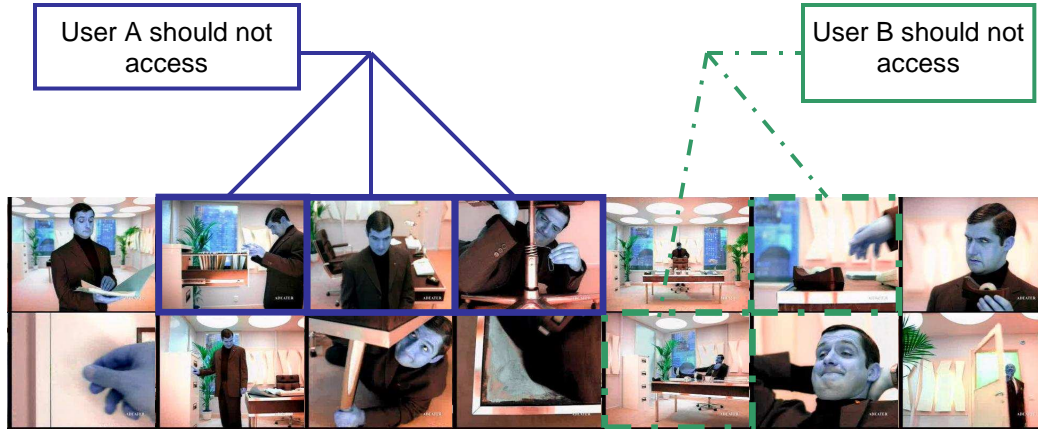


Figure VI-6. Example requirements for video scene/shot-level access control

Definition VI-1: An Object Hierarchy $OH = (O, OG, \leq_{OG})$, where O is a set of objects and $OG = O \cup G$ with G is a set of object groups. \leq is a partial order on OG called the dominance relation, and $O \subseteq OG$ is the set of minimal elements of OG with respect to the partial order. Given two elements $x, y \in OG$, $x \leq y$ iff x is a member of y .

<p>(a)</p> <pre> <TemporalRoles> <tGroup e_id='Holiday'> <Holiday h_id='Thanksgiving'> <Month>11</Month> <WeekNo>4</WeekNo> <WeekDay>4</WeekDay> </Holiday> ... </tGroup> <tGroup e_id='OfficeHour'> <H_interval> <H_start>9</H_start> <H_end>17</H_end> </H_interval> </tGroup> ... </TemporalRoles> </pre>	<p>(b)</p> <pre> <SpatialRoles> <ipGroup ipg_id='University'> <ipUniv ipu_id='FIU'> <ipDept ipd_id='SCS'> <seg1_fix>131</seg1_fix> <seg2_fix>94</seg2_fix> <seg3_fix>133</seg3_fix> <seg4_start>1</seg4_start> <seg4_end>255</seg4_end> </ipDept> ... </ipUniv> ... </ipGroup> ... </SpatialRoles> </pre>
---	--

Figure VI-7. XML examples of the optional roles
(a) example of temporal roles (b) example of IP address roles

6.3.2.3 Temporal Roles

In a multimedia application, data may be available to the users at certain time periods but unavailable at others. In order to achieve this target, the temporal constraints can be generally formalized with the following attributes: year, month, week number, week day, hour, minutes, etc. As shown in Figure VI-7(a), “Thanksgiving” is depicted with three attributes, which means that Thanksgiving is the fourth Thursday of November. The other temporal role named “OfficeHour” illustrates that the office hours are from 9 o’clock to 17 o’clock every day.

6.3.2.4 Spatial Roles

Even for the same user, he/she may be able to access the multimedia data only by using some specific computers. The IP addresses can be used to embed this kind of constraint by identifying the different networks and clients. Usually, an IP address appears in the equivalent dotted decimal representation such as 10.0.0.1 and each octet in it ranges from 0 to 255. By checking the associated IP address, the server can judge whether this access is allowed. For this purpose, we define the IP address segment for the related role modeling.

Definition VI-2: Given the octets named I_1, I_2, I_3, I_4 , the IP address segment expression IP can be defined as $IP = \sum_{j=1}^n x_j \cdot I_j \triangleright y_j \cdot I_d$, where $n = 4$, $0 \leq x_j \leq 2^8 - 1$, $0 \leq y_j \leq 2^8 - 1$, $x_j, y_j \in N$, $x_j + y_j \leq 2^8 - 1$ for $j = 1, \dots, 4$, $I_d \in \{I_1, I_2, I_3, I_4\}$.

The symbol \triangleright identifies the set of starting points of the intervals. For example, $131 \cdot I_1 + 94 \cdot I_2 + 133 \cdot I_3 + (1 \cdot I_4 \triangleright 254 \cdot I_4)$ stands for the segment between 131.94.133.1 and 131.94.133.255. It can be modeled by XML as shown in Figure VI-7(b), which also means this segment is under the role of University→FIU→SCS.

6.3.2.5 Security Rules

The security policies in traditional policies are basically classified into two categories. One is “Allow Policy” which means that certain users can access certain objects; the other is

“Deny Policy” which means that certain users cannot access certain objects. SMARXO introduces “Partial Allow Policy” which means that the user can only access partial data of this object. The definition of security policy is given in Figure VI-8(a) with a 5-tuple. Figure VI-8(b) gives a policy example which means that the “Student” can access “Shots_a” in “Holiday” by using the machines of “SCS.”

<p>(a) A security policy can be a 5-tuple: <Ru, Ro, Rt, Ri, Acc> Where: Ru: a user role; Ro: an object role; Rt: a temporal role; Ri: an IP address role; Acc: accessibility, the value can be Allow, Deny, or PartiallyAllow.</p>	<p>(b) <PolicyRoles> <policy p_id='p01'> <Ru>Student</Ru> <Ro>Shots_a</Ro> <Rt>Holiday</Rt> <Ri>SCS</Ri> <Acc>Allow</Acc> </policy> ... </PolicyRoles></p>
--	---

Figure VI-8. Security policies
(a) formalized security policy (b) XML example on policy roles

6.3.3 DBMS Management Phase

In this framework, the multimedia features, XML documents, and the multimedia contents are stored into an ORDBMS. By efficiently managing the XML segments in the ORDBMS, the XML documents can be easily updated when editing the security policies or the multimedia hierarchy information. Moreover, all the contents prepared in XML can be searched easily and accurately. In other words, it is very convenient for the administrator to retrieve the security policies by performing XML queries in the ORDBMS. Furthermore, ORDBMS provides some valuable functionality to store the byte data and large objects. Therefore, the images as well as the video shots can be professionally managed.

6.4 Security Verification

Based on an access request, the system will first check the user ID and password, and then check the user roles, object roles, temporal roles, and IP address roles consequently. After that, the security policy checks are performed on the “Object Entity Set” (OES) of the requested object o that includes both the object itself and all the entities s (segments or sub-objects belong to o).

Definition VI-3: Object Entity Set: $OES(o) = \{o\} \cup \{s : s \in o\}$.

Figure VI-9 depicts the security verification algorithm. A brief function “ $p_check(o)$ ” is presumed to check if the user can access object o in the specified time from some specified computer. Three kinds of results can be formalized as follows:

1. The access will be denied iff $p_check(o) = FALSE$.
2. A user can access the original multimedia data o iff

$$\forall t \in OES(o)[p_check(t) = TRUE],$$

where t can be any entity including o and all o 's sub-objects.

3. A user can access the processed multimedia data o' where the prohibited sub-objects are removed from o iff

$$(p_check(o) = TRUE) \wedge (\exists s \in o[p_check(s) = FALSE]),$$

where s can be any sub-object or segment which belongs to o .

Input: An Access Request $\langle id, pwd, time^*, ip_addr^*, object \rangle$

Output: (1) *FALSE*: Access is denied;

(2) *object*: Complete multimedia data as requested;

(3) *object'*: Processed multimedia data without the protected objects.

Algorithm *security_check*(*id, pwd, time**, *ip_addr**, *object*):

```
1) BEGIN
2) if (id, pwd)  $\notin U$            //Verify user identity
3)   return FALSE;
4) else
5)   if (get_user_role(id)) //Check user-role assignment
6)     u_role = get_user_role(id);
7)   else u_role = id;
8)   if (get_object_role(object)) //Check object-role assignment
9)     o_role = get_object_role(object);
10)  else o_role = object;
11)  if (get_temporal_role(time)) //Check temporal-role assignment
12)    t_role = get_temporal_role(time);
13)  else t_role = time;
14)  if (get_IPaddr_role(ip_addr)) //Check IP address role assignment
15)    ip_role = get_IPaddr_role(ip_addr);
16)  else ip_role = ip_addr;
17)  if (check_permission(u_role, o_role, t_role, ip_role)=DENY)
18)    return FALSE;
19)  else
20)    for all sub_object  $\notin$  object //Check permission on the sub-objects
21)      if (check_permission(u_role, sub_object, t_role, ip_role)=DENY) {
22)        object' = security_process(object) //Process multimedia data
23)        return object'; } //User can access the processed object
24)      else
25)        return object; //User can access the complete object
26) END
```

(**Note:** Features marked with * are advanced ones but optional in SMARXO.)

Figure VI-9. Algorithm for security verification in SMARXO

6.5 Conclusions

In this chapter, a practical framework – SMARXO is proposed to provide multilevel multimedia security for multimedia applications. RBAC, XML and ORDBMS are efficiently combined to achieve this target. In SMARXO, the RBAC model is enhanced and utilized to manage complicated roles and role hierarchies. Moreover, the multimedia documents are indexed and modeled such that access control can be facilitated on multi-level multimedia data.

Compared with the other existing security models or projects, SMARXO can deal with more intricate situations. First, the image object-level security and video scene/shot-level security can be easily achieved. Second, the temporal constraints and IP address restrictions are modeled for access control purposes. Finally, XML queries can be performed such that the administrators can proficiently retrieve useful information from the security roles and policies. The comparison among SMARXO and these existing security models/approaches is depicted in Table VI-1.

Table VI-1. Comparison of multimedia security techniques

Support	RBAC ₃	TRBAC	GTRBAC	GRBAC	GOCPN	SMARXO
Access Control	Yes	Yes	Yes	Yes	Yes	Yes
Role Hierarchy	Yes	Yes	Yes	Yes	No	Yes
Temporal Constraints	No	Yes	Yes	Yes	No	Yes
IP address Restrictions	No	No	No	No	No	Yes
Security on Multimedia Data	No	No	No	Yes	Yes	Yes
Security on Multilevel Objects	No	No	No	No	Yes	Yes

CHAPTER VII. MULTIMEDIA SYSTEM INTEGRATION

During the last decade, the rapid development of technologies and applications for consumer digital media has led to the desire to capture, store, analyze, organize, retrieve, and display multimedia data. Accordingly, various multimedia data management systems have been developed to fulfill these requirements. However, most of these systems mainly focus on one or few functionalities. Some systems are concerned with the production of multimedia material; some systems handle multimedia analysis and retrieval issues; while some other systems only provide the functionalities to synchronize various multimedia files into a presentation. In this chapter, a distributed multimedia management system called DMMManager [ChenSC03b] is presented, which tries to integrate the full scope of functionalities for multimedia management including multimedia data capturing, analysis, indexing, organization, content-based retrieval, multimedia presentation design and rendering.

7.1 System Overview

In order to provide the ability for handling simultaneous accesses of multiple users, a multi-threaded client/server architecture is designed and deployed in DMMManager. The server and the client are developed by using C++ and Java, respectively, and can run on multiple platforms. On the server-side, a huge amount of multimedia data is organized and stored, and all the computation intensive functions are arranged on the server-side to fully utilize the server's computation power. Accordingly, a database engine is implemented to support file supply, feature extraction, query processing, training computations, etc. The client-side applications provide several user-friendly interfaces for the users to issue queries, check retrieval results, and edit the multimedia presentation conveniently, gather various commands issued by the users, and package them into different categories. The TCP protocol is utilized in the client and server communications. Upon receiving the requests, the server analyzes the message, identifies what

kind of operations the user did, decides which component(s) need(s) to be run next, and then activates the corresponding operations. When the complete multimedia contents are required for the playback of media streams, UDP protocol is adopted to transfer the multimedia data.

DMMManager

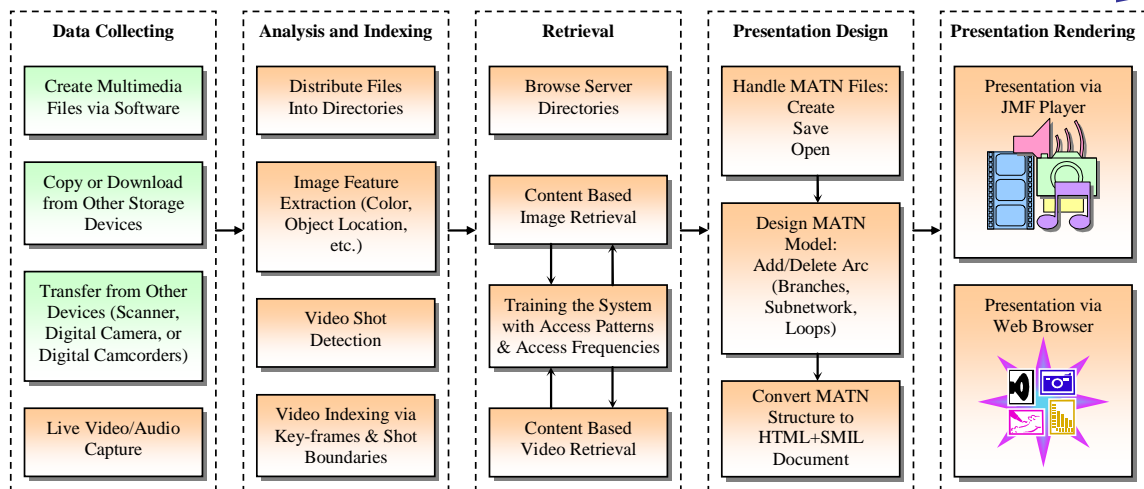


Figure VII-1. The multimedia management flow of DMMManager

This section mainly describes how DMMManager deals with a set of multimedia management issues and generates a multimedia presentation. The multimedia management flow is illustrated in Figure VII-1, where the colored boxes contain the functionalities supported by DMMManager and the operational sequence goes from data collecting to presentation rendering.

The rest of this chapter is organized as follows. In the next section, multimedia data gathering methods are addressed and the live video/audio capture component is introduced. In Section 7.3, multimedia data analysis and indexing are discussed. Section 7.4 describes three major retrieval methods in DMMManager. The Multimedia Augmented Transition Network (MATN) [ChenSC97][ChenSC00a][ChenSC00c] model and the corresponding presentation design module are introduced in Section 7.5. Section 7.5 discusses the presentation rendering techniques as well. Finally, the conclusions are summarized in Section 7.6.

7.2 Multimedia Data Collecting

Recent developments of the multimedia capture devices, data compression algorithms, large capacity storage and high bandwidth networks have helped create the overwhelming production of multimedia content. DMManager can handle the multimedia data generated or collected from multiple sources. As shown in Figure VII-1, the data can be created via software, copied or downloaded from other storage devices, transferred from other devices, or captured from live video/audio. To capture live video/audio, the video/audio capture hardware such as a web camera is used to capture consecutive scenes, where the users can decide when to begin the capturing process. Then the data gathered are encoded into video files with MPEG or AVI formats. Implemented by using Java Media Framework (JMF) [JMF], decoding and monitoring the live video/audio are also supported. In DMManager, the captured raw video/audio data are initially stored in the client-side and the user may use the upload function to transfer them to the database at the server so that they can be processed and analyzed for future retrieval.

7.3 Multimedia Analysis and Indexing

In order to address and access the desired multimedia data efficiently, large-scale digital archives need to be analyzed and indexed in the database at the server. In DMManager, different multimedia files are categorized and stored into a set of directories based on their media types and contents. For example, the image files can be classified into “animals”, “flowers”, “sports”, etc.; while the video files can be categorized as “movies”, “news”, “advertises”, etc.

7.3.1 Image Analysis and Indexing

Due to the explosion of image files and the inefficiency of text-based image retrieval, Content-Based Image Retrieval (CBIR) approaches are implemented in DMManager. Currently, DMManager provides the functionalities to extract low-level features (such as color) and mid-level features (such as object location) from the images, where the HSV color space is

used to obtain the color features and the SPCPE algorithm [ChenSC00b] is applied for object location features. It is worth mentioning that our system is flexible so that the functional components to extract other features such as texture and shape can be easily plugged in.

The extracted features together with the images are indexed and stored in the database in the server, where each image is classified into a certain category and has its own domain name and a unique ID. In DMMManager, the image metadata and feature data sets can be stored in the text files or in the Microsoft Access database, while the latter one can provide more support on the image database management and speed up the retrieval process.

7.3.2 Video Analysis and Indexing

Different from images, videos are continuous media with the temporal dimensions and consume a huge amount of storage. Therefore, instead of sequential access to the video content, which requires tremendous time, video summarization, a process of extracting abstract representations that compresses the essence of the video, becomes a challenging research topic. DMMManager adopts an efficient way to summarize video by utilizing the video shot detection method proposed in [Yilmaz00], which automatically segments the video into shots and extracts the key frames in a meaningful manner. Here, a video shot is defined as a continuous view filmed by one camera without interruption and the first frame of the detected shot is considered as the key frame representing this shot.

The procedure of video analysis and indexing is as follows. First, the video is segmented into frames. Then the features of each frame are extracted for comparison. Note that each frame can be considered as an image, and therefore, the image feature extraction functionalities implemented in the image analysis stage can be used to extract the features from the frames. By considering the relative spatial-temporal relationships between video objects [ChenSC01a], the key frame representations and shot boundaries can be achieved. Finally, the video is segmented

into smaller video clips (shots) based on these shot boundaries. These shot files together with their boundary information and key frames are stored in the server for future video retrieval. Both AVI and MPEG formats are supported in this video shot detection and segmentation process.

7.4 Multimedia Retrieval

Since the multimedia data are categorized and stored in the directories based on their media types and the defined categories, DMMManager provides a directory-based retrieval functionality, where all the data can be browsed in a hierarchical manner. In order to facilitate the users to select and acquire the desired files, a file supply functionality is implemented, which is used by a content-based image retrieval component and a key-frame based video retrieval component to allow the client to browse, download and upload files from/to the server.

7.4.1 Content-based Image Retrieval

Since the trained CBIR subsystem can provide users with their desired images more accurately and more quickly, it will greatly facilitate the design of multimedia authoring and presentation by using the retrieved images. In DMMManager, this MMM-based CBIR system is connected with the multimedia presentation authoring tool based on the Multimedia Augmented Transition Network (MATN) model [ChenSC00a].

With a multi-threaded client/server architecture, this system can support multiple clients to issue queries and offer feedback simultaneously. In the server-side, a database engine is implemented to support image query processing, file supply, training computations, feature extraction and indexing of images. There are 10,000 color images as well as their feature set stored in our database. The client-side provides a content-based image retrieval user interface, which allows the browsing, query and feedback of the image contents.

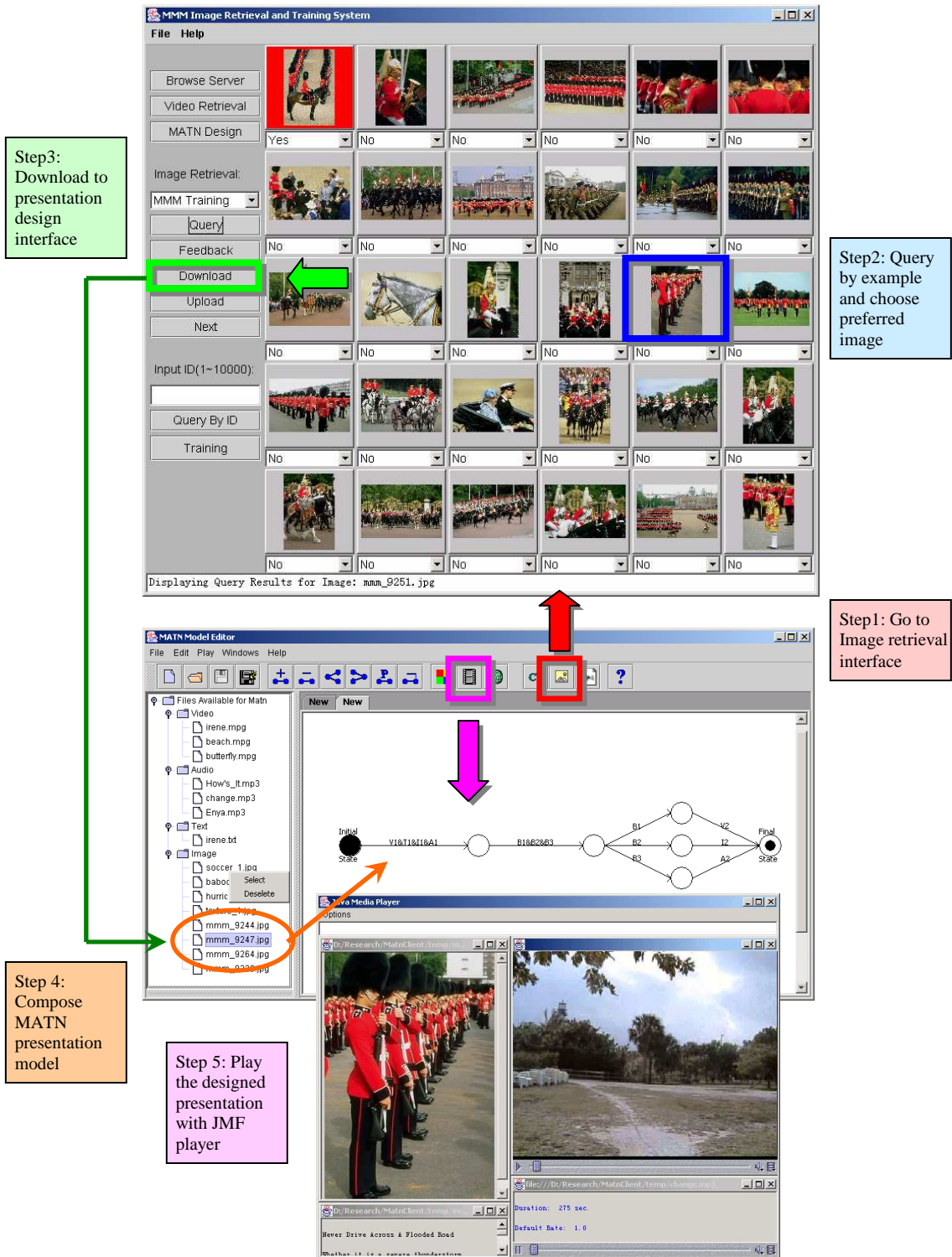


Figure VII-2. Multimedia presentation authoring tool

As shown in Figure VII-2, after the user finds the desired images using the image retrieval system, he/she may use the “Download” function to add them into the presentation material tree, which can be employed later in multimedia presentation designs. Basically, our CBIR system can help users find their images of interest more accurately and more quickly such that multimedia presentation design becomes much easier.

7.4.2 Video Data Browsing and Retrieval

7.4.3.1 Key-frame Based Video Browsing

In [Yoshitaka99], the “Query By Example” (QBE) approach for video retrieval is proposed, where the video data are considered as a set of images without temporal interrelations. Therefore, the image retrieval method can be performed for video retrieval purposes. Different from the QBE video retrieval system, DMMManager adopts the key-frame based searching method, which considers the temporal relationships among the video data and is much more powerful for video queries.

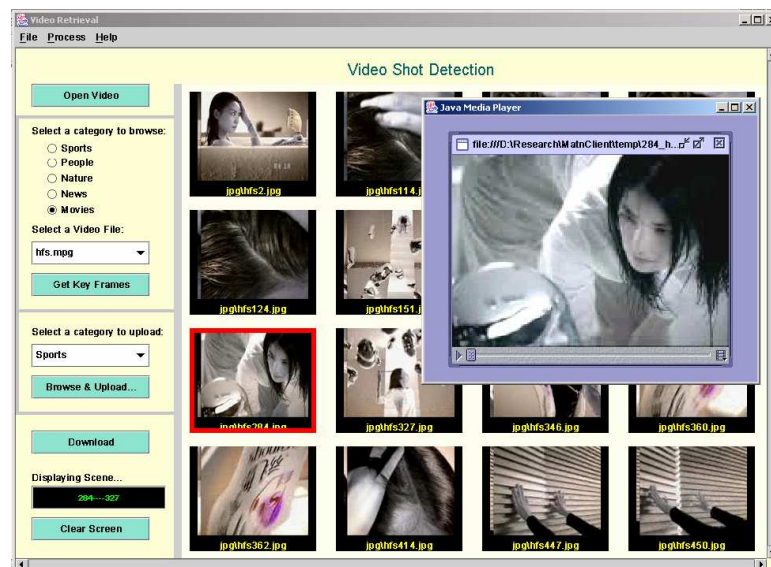


Figure VII-3. The key-frame based video retrieval interface with a shot displayed

Each video file is divided into shots, which are stored and organized in the server-side with their shot boundary information and key frames. Each key frame corresponds to a unique

shot file. The video retrieval process is defined as follows. First, users select the favorite domain. The system lists the names of all the video files in this category accordingly. Based on this list, users can select the desired video file. Then the key frames of the selected video will be displayed. From the key frames, users can easily know the contents of the whole video without previewing it. Once the users double-click a key-frame, the corresponding video shot will be displayed via a JMF player as shown in Figure VII-3. Both the original video file and the video shots can be downloaded for future multimedia presentation design and rendering.

7.4.3.2 SoccerQ: A Soccer Video Retrieval System

A soccer video retrieval system named SoccerQ [ChenSC05a] is integrated in DMManager to facilitate soccer event searching and browsing. A graphical query language is designed for specifying the relative temporal queries. Basically, the user is allowed to specify the search target and search space. After that, different objective events and the temporal relationship model types can be chosen. The double-thumbed slider is utilized for the event position, duration, or range specification. To further quantify the related parameters, the user is allowed to input the number and specify the unit. As mentioned earlier, the minute, second, and shot units are provided. In addition, the sentential operators are provided such that different query rules can be combined. Those operators include: the negative operator “not”, the conjunctive operator “and”, and the disjunctive operator “or”. Given the following two queries, the corresponding visual query specifications are listed in Table VII-1.

Table VII-1. Example mappings to the graphical query language

Events and Relationships:	Parameters (unit)	Graphical Query Filter
<i>Query 3:</i> $R_A \theta V$, $\theta = \text{starts}$ $A = \text{“Goal”}$	$R_{A0} = 0$ $R_{Af} = 10$ (minutes)	
<i>Query 4:</i> $A \theta R_B$, $\theta = \text{“starts”}$ $A = \text{“Corner kick”}$ $B = \text{“Goal”}$	$R_{B0} - A_0 = 0$ $R_{Bf} - A_f = 2$ (minutes)	

Query 1: “Find all the soccer videos from the database where there is a goal occurrence in the first 10 minutes of the video.”

Query 2: “Find all the corner kick shots from all the female soccer videos where the corner kick resulted in a goal event occurring in 2 minutes.”

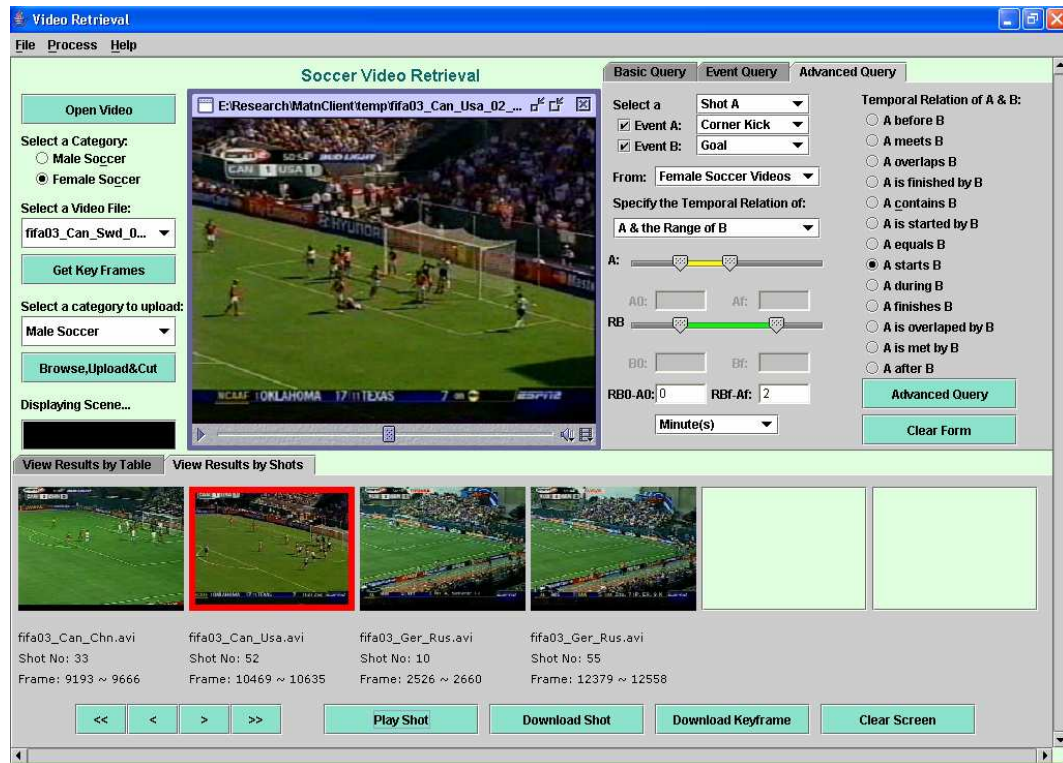


Figure VII-4. Soccer retrieval interface with example temporal query and results

The client-side application integrates both soccer event query and video browsing panels in a common framework. When a query is issued, the related parameters are transferred to the server-side. Accordingly, the server-side database engine performs computation-intensive functionalities including graphical query translation, query processing, video supply, etc. The huge amount of multimedia data is organized by adopting the PostgreSQL database.

As shown in Figure VII-4, the query criteria are specified in the graphical query panel in the upper-right corner. The illustrated query example is to find all the “corner kick” shots from all the female soccer videos and the “corner kick” shot can cause a “goal” event occurring in 2 minutes. The key frames of four result shots are displayed in the video browsing panel. The video

shot can be displayed by double clicking the corresponding key frame. Finally, the retrieved video or video shots can be downloaded to the client-side and be composed and then displayed in a multimedia presentation.

7.5 Multimedia Presentation Module

7.5.1 Multimedia Presentation Authoring

Compared to the traditional text or numerical data, the multimedia data are far more complicated because they usually contain spatial and/or temporal relationships. Therefore, in order to compose the multimedia objects from distributed sources into a sophisticated presentation, it is very important to develop abstract semantic models which meet the following requirements [Bertino98][ChenSC00a]: First, the specification of temporal constraints for multimedia content must be supported by the devised model. Second, the model must ensure that these temporal constraints can be satisfied at runtime. Finally, the model should be a good programming data structure for implementation to control multimedia playback. Currently, the researches on the multimedia conceptual models lead to four different directions: Timeline-based models, Script-based models, Graph-based models, and Structure-based models.

Our DMMManager adopts an abstract semantic model called “Multimedia Augmented Transition Network” (MATN) model [ChenSC00a], which combines the structure-based authoring with well-defined graphic based notations. Typically, the MATN model can offer great flexibility for the designers to synchronize the heterogeneous multimedia objects into presentations. MATN can not only model the sequential multimedia scenarios, but also sketch the complicated presentations, which support user interactions, structure reusability, and quality of services. Basically, an MATN model consists of a group of states connected by the directed arcs. The multimedia strings are marked as the labels of the arcs. The most fundamental components of the MATN model are defined as follows:

- State: Denotes the starting situation of the next multimedia stream and/or the ending situation of the previous multimedia stream. Each state is identified with its own name.
- Arc: An arc connects two states and has its own time duration. When the time duration is over, a transition occurs and the next stream combined with the next arc will be imported and displayed.
- Multimedia input string: A regular expression which describes the spatio-temporal combination of diverse multimedia streams and how they will be displayed. The single media object can be represented by its type (“T”: text; “I”: image; “V”: video; “A”: audio) and a unique number. For example: “T1”, “V2”, etc. The multimedia streams can be expressed by the connection of the multiple objects linked by the symbol “&”, e.g. “T1&V2”, etc.
- Feature set: Each multimedia object has its individual feature set, which contains a great deal of useful information for the presentation rendering, such as: the corresponding multimedia file’s path and name, duration, starting time, ending time, display window location and size, etc. The feature sets are embedded in the MATN and can be edited easily.

As shown in Table VII-2, the MATN models are constructed to model Allen’s 13 kinds of temporal relationships [Allen83], where the letters “A” and “B” denote two diverse multimedia objects. It can be easily seen from the table that the MATN model has a powerful expressive capability so that it is able to represent all the 13 kinds of temporal relationships correctly. In addition, MATN introduces some advanced terms and related functionalities:

Table VII-2. MATN structures for 13 temporal relationships

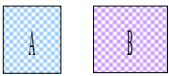
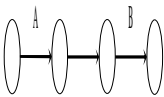
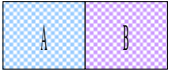
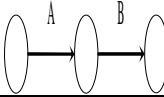
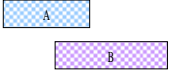
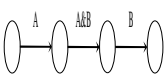
Temporal Relationships	Timeline Representation	MATN Structure
A is BEFORE B B is AFTER A		
A MEETS B B is MET by A		
A OVERLAPS B B is OVERLAPED by A		
A DURING B B CONTAINS A		
A STARTS B B is STARTED by A		
A FINISHES B B is FINISHED by A		
A is EQUAL to B (B is EQUAL to A)		

- **Branches:** Users can design the multiple branches and select the favorite branch in the real presentation. Subsequently, the corresponding part will be rendered at runtime. Therefore, the indeterminacy and user interactions can be handled.
- **Loops:** The loops can be designed and the user can decide the repetition times for the corresponding portions of the scenario. Accordingly, the presentation structure is simplified when some streams need to be displayed repeatedly.
- **Sub-network:** The previously designed MATN structures can be reused as a portion when authoring the new scenario. With only one notion (e.g., P1) on the arc, the presentation will open the related MATN file and follow its structure. When a sub-network is encountered during the interpretation of an MATN model, the control of the main presentation is passed to the sub-network level in a way that the presentation flow within

the sub-network is inserted seamlessly into the main presentation. Consequently, reusability is ensured and the presentation can be designed in a hierarchical way.

- Condition/Action Table: The table is used to store the information which cannot be implied in the states or arcs. Different actions can be carried out depending on whether a certain condition is met or not. Hence, the quality of service (QoS) control can be supported without difficulty. For example, if the available bandwidth is lower than a certain threshold, the compressed version of the video can be delivered instead of the original one.

Table VII-3. MATN design buttons & functionalities

Button Symbol	Button Functionality	Button Symbol	Button Functionality
	Add an <i>Arc</i>		Create <i>Branches</i> (The number of branches ≥ 2)
	Delete an <i>Arc</i>		Merge the available <i>Branches</i> (The number of branches ≥ 2)
	Add a <i>Sub-network</i>		Create a <i>Loop</i>

As discussed before, the MATN model can be formally defined as an eight-tuple: $\langle \Sigma, \Gamma, Q, \psi, \Delta, S, F, T \rangle$. Because this model is more capable of modeling synchronization relationships such as concurrency, alternatives, looping, etc., we implemented it to facilitate the design of multimedia presentation in DMMManager. Also the MATN is utilized as the internal data structure of this module. Figure VII-5 shows the user interface of the MATN model design module. The MATN presentation model can be easily created, edited, saved and opened through this interface. As mentioned, during the multimedia retrieval process, users can download the retrieved multimedia files. From this figure, we can see that the corresponding file names are

categorized and listed within a tree-view in the left side of the window. Users may preview any listed multimedia object and select one of them or their combinations to construct the MATN model. Several buttons are designed to provide the functionalities to assemble the MATN structure, which can be found in Table VII-3. In addition, more edit functions, such as delete function, are also developed for users to modify the structure.

Figure VII-5 shows an MATN example that contains a set of filled circles, arcs and the corresponding arc labels. The video file named “beach.mpg” is previewed as shown in the left-bottom corner. The pop up window works for the “Add an arc” function. By clicking the “Enter” button, the selected files will be synchronized into a multimedia stream and the corresponding arc will be generated and added after the current final state (or the selected state).

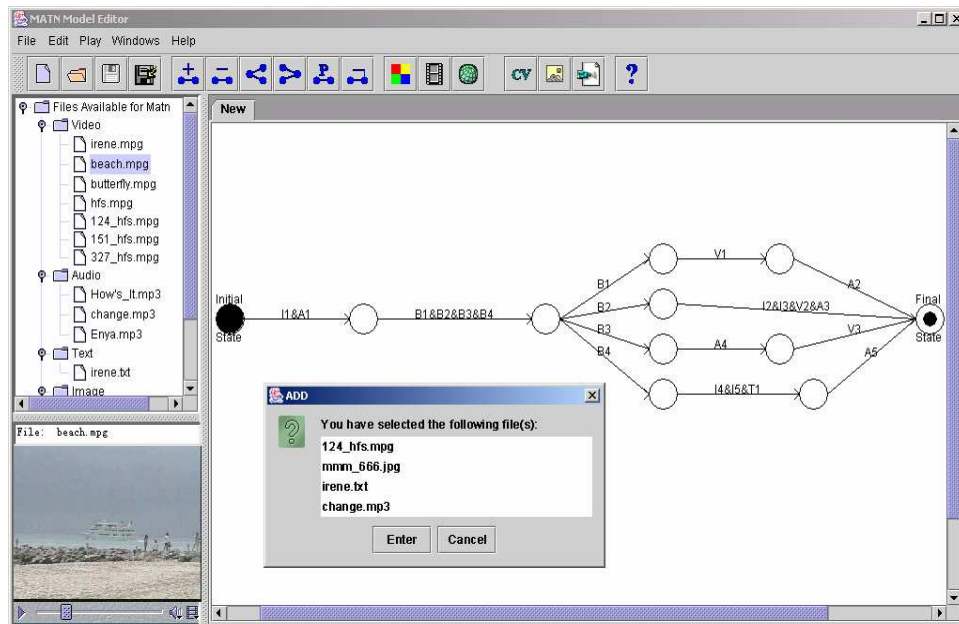


Figure VII-5. The user interface for MATN model design

7.5.2 Multimedia Presentation Rendering

In order to convert the designed MATN model to a multimedia scenario perceivable to users, a presentation rendering component is implemented in DMMManager. In this presentation rendering layer, two approaches are offered to fulfill different requirements in different

environments. One is to display the multimedia presentation in a client-side application, namely JMF player, implemented with Java Media Framework (JMF) [JMF]. When the JMF player is not available, the other approach can be adopted to convert the MATN structure into an HTML file with SMIL [SMIL] notations which can be displayed in the web browser directly.

7.5.3 Presentation Rendering via JMF Player

The Java Media Framework provides superior techniques for rendering these kinds of multimedia presentation models into a stand-alone application in a runtime environment. In DMMManager, four kinds of distinct media players are implemented to exhibit the text, image, video and audio data, respectively. The synchronization information and the spatial and/or temporal relationships can be easily retrieved from the MATN model and used to control these players. As indicated before, the MATN presentation model can be used to create a clean and integrated structure while enclosing a large amount of valuable information, which plays an important role in the rendering process. By double-clicking the state, the time duration specification, which is employed to control the start and end of a certain media stream, can be checked and modified. In addition, each multimedia object involved in the presentation has a feature set, which is used to display the presentation and control the layout. Moreover, the user can choose to render the whole or any portion of the MATN model by indicating the start and ending state. Figure VII-6 demonstrates a presentation rendering example via JMF players. Four different types of multimedia objects (image, video, audio, text) gathered from distributed sources are displayed concurrently.

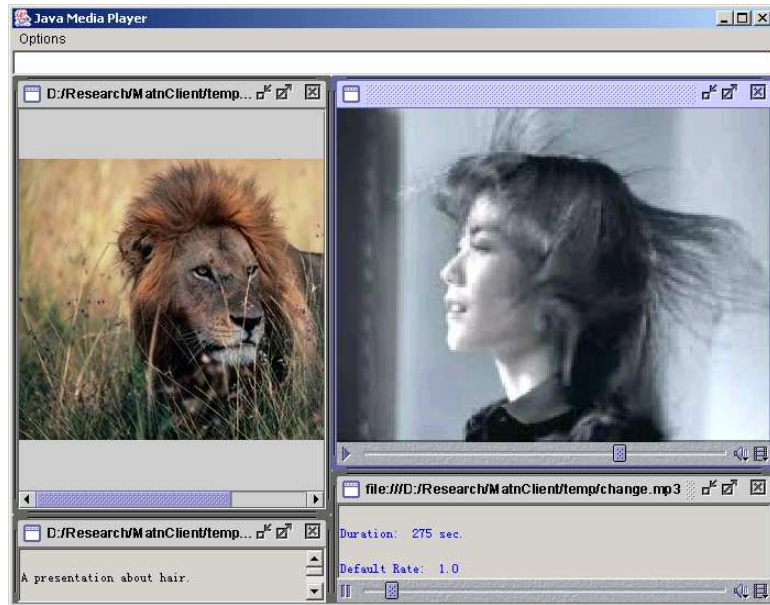


Figure VII-6. The rendered multimedia presentation played by the JMF player

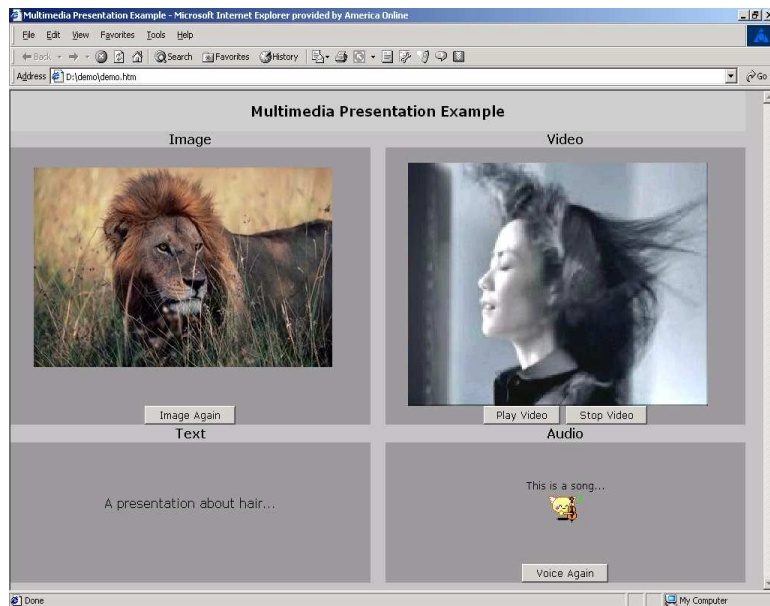


Figure VII-7. The rendered multimedia presentation played by the web browser

7.5.4 Presentation Rendering via SMIL Language

As a script-based model, SMIL has some disadvantages. For example, close to the form of a programming language, it is hard for general users to learn and use. However, one of its benefits is that the SMIL notations can be combined with the general HTML documents so the

presentation can be displayed directly within web browsers such as Internet Explorer. In DMMManager, users can utilize the easy-to-use functions to design the MATN presentation model in a graphical environment. Then a component is implemented to interpret the MATN structure to an HTML+SMIL document. Attributes such as file locations and durations and the synchronization construction contained in the well-structured MATN model can be translated into HTML+SMIL files so that the presentation can be displayed wherever a web browser is available. Thus, the heavy burden of learning a programming language is avoided while the same conceptual structure, temporal relations, and synchronization controls can be maintained correctly. Such an HTML+SMIL presentation scenario is demonstrated in Figure VII-7. As we can see, the presentation contents are the same as the ones played by the JMF player.

7.6 Conclusions

In this section, a multimedia management system, called DMMManager, is introduced, which adopts a multi-threaded client/server architecture and is capable of handling multiple clients. Both the multimedia database and the computation intensive functions are maintained or deployed on the server-side. User-friendly interfaces are developed and connected to each other so that the user can switch among them easily. A set of key components are implemented in DMMManager to collect multimedia data, analyze and index them, retrieve data as well as design and realize the final multimedia presentation. Those well-designed components are flexible, extensible and easy to maintain, which guarantees their reusability. The proposed system also allows some level of openness in its architecture, such that a new multimedia application can be easily plugged in.

CHAPTER VIII. CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

In this research, we have designed an integrated framework named DIMUSE for a distributed multimedia system with database management and security assurance. In order to provide more appealing multimedia experiences to the users, a set of novel technologies are proposed, implemented, and integrated in DIMUSE.

First, an enhanced multimedia database model called HMMM is proposed to support concept-based video queries, especially temporal event pattern queries. By using HMMM, various multimedia objects in different levels are modeled by state sequences associated with their transition probabilities by incorporating the temporal meanings. The video retrieval procedure to search the specified temporal patterns is designed as a stochastic process which always tries to traverse the optimal path, thus guaranteeing the most efficient retrieval performance even in a large scale video database. Additionally, a ranking algorithm is proposed which considers the visual/audio features, temporal relationships, and user preferences when sorting the candidate video sequences.

Second, a conceptual video clustering strategy is designed to couple with the HMMM mechanism for further improvement on the overall retrieval performance. The cumulated user feedbacks are reused in the video clustering process. With learning of historical query results, the system groups the videos by not only considering the low level features, but also taking the high level semantics and user preferences into account. The HMMM-based database model is constructed to support the conceptual video database clustering. With the clustered database, the retrieval process becomes faster and more efficient. At the same time, the multimedia database structure is further improved by adding a new level to model the video clusters.

Third, an innovative solution is provided to capture and model individual user's preferences by including high-level concepts and relationships, as well as low-level features. In the proposed online learning strategy, a set of MMM instances are created for the independent user with distinct preferences. To satisfy an individual user's needs, the system is designed to capture, learn, and then generate the updated results to satisfy the special information requirements. Additionally, the overall system can always remain as an offline learning mechanism since the access patterns and frequencies from various users can be proficiently stored and analyzed for the long-term offline system training. With this promising technique, this approach can accommodate the interest of a particular user while it can also take advantage of the common knowledge of most users. To automate the offline training process, we also propose an advanced training method by adopting the association rule mining technique, which can effectively evaluate accumulated feedback and automatically invoke the training process. Training is performed per video rather than for all videos in the database, making the process more efficient and robust.

To facilitate all of these newly proposed techniques, a soccer video retrieval system is presented to demonstrate the efficiency of the database modeling mechanism, the temporal pattern retrieval algorithm, the video clustering strategy, as well as the performance of online and offline system learning. A set of experimental tests are conducted to validate the performance of these new techniques. In addition to this, MoVR, a user adaptive video retrieval framework in the mobile wireless environment, is presented. While accommodating various constraints of the mobile devices, a set of advanced techniques are developed and deployed to address essential issues. For instance, an HMMM-based user profile is defined, which is also integrated seamlessly with a novel learning mechanism to enable the "personalized recommendation" for an individual user by evaluating his/her personal histories and feedbacks. In addition, the fuzzy association concept is employed in the retrieval process such that the users gain control of the preference

selections to achieve a reasonable tradeoff between retrieval performance and processing speed. With all of these multimedia searching and browsing components, legal users are capable of finding, accessing, and downloading media files of their interest for any future usage such as designing a multimedia presentation.

Besides, to deal with security and privacy issues in distributed multimedia applications, DIMUSE also incorporates a practical framework called SMARXO, which supports multilevel multimedia security control. SMARXO efficiently combines Role-Based Access Control (RBAC), XML and Object-Relational Database Management System (ORDBMS) to achieve the target of proficient security control. By using this framework, administrators are capable of creating, deleting, and modifying the user roles, object roles, temporal roles, IP address roles, and security policies. Meanwhile, security information retrieval becomes very convenient because all the protection-related information is managed by XML.

The proposed framework DIMUSE efficiently integrates all the above-mentioned techniques. In this framework, the MATN based multimedia presentation component plays as an important role in the system integration because the media data retrieved from other components are downloaded to this environment and users can design their own presentations with their preferred data. Furthermore, the security module takes charge of the information assurance and privacy protection for the other two modules with creation, storage, indexing and presentation for the multi-level secured multimedia contents. With the efficient integration mentioned above, the proposed framework is able to create a powerful, secure and user friendly multimedia application. In addition, all of the proposed techniques can also work individually to achieve some specific goals.

8.2 Future Work

On the basis of current research results, the future work is proposed accordingly as listed below.

1. The current prefiltering process is conducted by heavily relying on the domain knowledge and human experiences on the feature values. A prospective research issue here is to perform the prefiltering process by using boosted machine learning algorithms. Boosting refers to the general problem of producing a very accurate prediction rule (strong classifier) by combining moderately inaccurate classifiers, also called “weak classifiers”. The intuitive idea of a boosting algorithm is to alter the distribution over the domain in a way that increases the probability of the harder parts of the space, thus forcing the weak learner to generate new hypotheses that make fewer mistakes on these parts. A good candidate boosting algorithm is Adaboost [Freund97][Freund99], which can be used to boost the performance of the machine learning algorithms. Further study can be conducted to choose the best machine learning method and adjust the Adaboost method to perform the prefiltering process without manual effort.
2. The proposed HMMM mechanism will be further generalized by modeling more video shots and refining the initialization method for the temporal affinity relationships. The similarity measurement can be further recalibrated by updating the feature measurement and importance weight measurement. For example, the information gain ratio could be one solution to initialize the feature weight. Besides, the matrices and their formulas to link different levels can be further improved and justified.
3. Considering the flexibility and scalability capability of HMMM model, it can actually be enhanced to further incorporate ontology in the higher level descriptions. There are many questions that need to be answered: How to define the ontology? What’s the difference and relationship between ontology and general semantic events/concepts? What’s the purpose for modeling ontology and what kind of functionalities can we perform by adding ontology? The similarity measurement functions should also be updated when considering ontology in the HMMM mechanism.

4. A semantic model is desired to formalize the security access control process in the distributed environment. Composing multimedia documents in a distributed heterogeneous environment (e.g., Internet) involves integrating multimedia objects from multiple security domains that may employ different access control policies for media objects. Therefore, a security model for distributed document management system is required to allow the management of secure multimedia documents. The HMMM model can be considered in this security modeling task.
5. Online Video Retrieval System. In the recent years, Internet has become a dynamic huge repository for various multimedia data. Most of the current web search engines (e.g., Google Video [GVideo], Yahoo! Video [YVideo]) utilize web crawlers to collect video data and apply a text-based searching algorithm on meta data or file names for video retrieval, which cannot fully discover the real semantic meanings of the online videos. The proposed HMMM mechanism can actually be used and advanced to solve this problem. As we can extract the low level features and obtain the user access histories through a web log, the visual/audio feature values and affinity relationships between online videos can be used to refine content based video retrieval (query by example) performance.

LIST OF REFERENCES

- [Agrawal93] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD)*, USA, 1993, pp. 207–216.
- [Agrawal94] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," In *Proceedings of 20th International Conference Very Large Data Bases (VLDB)*, 1994, pp.487–499.
- [Ahmad06] I. Ahmad, S. Kiranyaz, F. A. Cheikh, and M. Gabouj, "Audiobased Queries for Video Retrieval over Java Enabled Mobile Devices," In *Proceedings of SPIE (Multimedia on Mobile Devices II), Electronic Imaging Symposium*, San Jose, California, USA, 2006, pp. 83–93.
- [Allen83] J. F. Allen, "Maintaining Knowledge about Temporal Intervals," *Communications of the ACM*, Nov. 1983, vol. 26, no. 11, pp. 832–843.
- [Amir05] A. Amir, M. Berg, and H. Permuter, "Mutual Relevance Feedback for Multimodal Query Formulation in Video Retrieval," In *Proceedings of the 7th ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2005, Singapore, pp. 17–24.
- [Apriori] Source code of Apriori algorithm (written in C) for frequent item set mining/association rule induction. <http://fuzzy.cs.uni-magdeburg.de/~borgelt/software.html#assoc>
- [Aref02] W. G. Aref, A. Catlin, J. Fan, A. K. Elmagarmid, M. A. Hammad, I. F. Ilyas, M. S. Marzouk, and X. Zhu, "A Video Database Management System for Advancing Video Database Research," In *Proceedings of the International Workshop on Multimedia Information Systems (MIS)*, Tempe, Arizona, USA, 2002, pp. 8–17.
- [Aref03] W. G. Aref, A. C. Catlin, A. K. Elmagarmid, J. Fan, M. A. Hammad, I. Ilyas, M. Marzouk, and T. Ghanem, "Video Query Processing in the VDBMS Testbed for Video Database Research." In *Proceedings of the 1st ACM International Workshop on Multimedia Databases (ACM MMDB)*, 2003, pp. 25–32.
- [Babaguchi01] N. Babaguchi, Y. Kawai, and Y. Kitahashi, "Generation of Personalized Abstract of Sports Video," In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, Tokyo, Japan, 2001, pp. 800–803.
- [Bertino98] E. Bertino and E. Ferrari, "Temporal Synchronization Models for Multimedia Data," *IEEE Transaction on Knowledge and Data Engineering*, July-August 1998, pp. 612–630.
- [Bertino01] E. Bertino, P. A. Bonatti, and E. Ferrari, "TRBAC: A Temporal Role-Based Access Control Model," *ACM Transaction on Information and System Security (TISSEC)*, August 2001, vol. 4, no. 3, pp. 191–233.
- [Bhatti05] R. Bhatti, J. B. D. Joshi, E. Bertino, and A. Ghafoor, "X-GTRBAC: An XML-based Policy Specification Framework and Architecture for Enterprise-Wide Access Control," *ACM Transactions on Information and System Security*, 2005, vol. 8, no. 2, pp. 187–227.

- [Borgelt02] C. Borgelt and R. Kruse, “Induction of Association Rules: Apriori Implementation,” In *Proceedings of 15th Conference on Computational Statistics (Compstat)*, Germany, 2002.
- [Borgelt03] C. Borgelt, “Efficient Implementations of Apriori and Eclat,” In *Proceedings of 1st Workshop of Frequent Item Set Mining Implementations (FIMI)*, USA, 2003.
- [Bruto06] E. Bruno, N. Moënne-Loccoz, and S. Marchand-Maillet, “Asymmetric Learning and Dissimilarity Spaces for Content-based Retrieval,” in *Proceedings of International Conference on Image and Video Retrieval (CIVR)*, USA, 2006, pp. 330–339.
- [ChenL98] L. Chen and K. Sycara, “WebMate: A Personal Agent for Browsing and Searching,” In *Proceedings of the 2nd International Conference on Autonomous Agents and Multi-Agent Systems*, 1998, pp. 132–139.
- [ChenL03] L. Chen, M. T. Özsu, and V. Oria, “Modeling Video Data for Content Based Queries: Extending the DISIMA Image Data Model,” In *Proceedings of 9th International Conference on Multi-Media Modeling*, Taiwan, January 2003, pp. 169–189.
- [ChenSC97] S.-C. Chen and R. L. Kashyap, “Temporal and Spatial Semantic Models for Multimedia Presentations,” In *Proceedings of 1997 International Symposium on Multimedia Information Processing*, December 1997, pp. 441–446.
- [ChenSC00a] S.-C. Chen, R.L. Kashyap, and A. Ghafoor, *Semantic Models for Multimedia Database Searching and Browsing*, Kluwer, 2000.
- [ChenSC00b] S.-C. Chen, S. Sista, M.-L. Shyu, and R.L. Kashyap, “An Indexing and Searching Structure for Multimedia Database Systems,” *IS&T/SPIE Conference on Storage and Retrieval for Media Databases*, 2000, pp. 262–270.
- [ChenSC00c] S.-C. Chen, M.-L. Shyu, and R. L. Kashyap, “Augmented Transition Network as a Semantic Model for Video Data,” *International Journal of Networking and Information Systems, Special Issue on Video Data*, 2000, pp. 9–25.
- [ChenSC01a] S.-C. Chen, M.-L. Shyu, C. Zhang, and R.L. Kashyap, “Identifying Overlapped Objects for Video Indexing and Modeling in Multimedia Database Systems,” *International Journal on Artificial Intelligence Tools*, vol. 10, no. 4, December 2001, pp. 715–734.
- [ChenSC01b] S.-C. Chen and R. L. Kashyap, “A Spatio-Temporal Semantic Model for Multimedia Presentation and Multimedia Database Systems,” *IEEE Transaction on Knowledge and Data Engineering*, July/August, 2001, pp. 607–622.
- [ChenSC03a] S.-C. Chen, M.-L. Shyu, C. Zhang, L. Luo, and M. Chen, “Detection of Soccer Goal Shots Using Joint Multimedia Features and Classification Rules,” In *Proceedings of the Fourth International Workshop on Multimedia Data Mining (MDM/KDD), in conjunction with the ACM International Conference on Knowledge Discovery & Data Mining (SIGKDD)*, Washington, DC, USA, 2003, pp. 36–44.
- [ChenSC03b] S.-C. Chen, M.-L. Shyu, and N. Zhao, “MediaManager: A Distributed Multimedia Management System for Content-Based Retrieval, Authoring and Presentation,” In

Proceedings of the 9th International Conference on Distributed Multimedia Systems, Miami, FL, USA, 2003, pp. 17–22.

- [ChenSC03c] S.-C. Chen, M.-L. Shyu, S. Peeta, and C. Zhang, “Learning-Based Spatio-Temporal Vehicle Tracking and Indexing for Transportation Multimedia Database Systems,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, no. 3, September 2003, pp. 154–167.
- [ChenSC04a] S.-C. Chen, M.-L. Shyu, M. Chen, and C. Zhang, “A Decision Tree-based Multimodal Data Mining Framework for Soccer Goal Detection,” In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, Taipei, Taiwan, R.O.C., 2004, pp. 265–268.
- [ChenSC04b] S.-C. Chen, M.-L. Shyu, and N. Zhao, “SMARXO: Towards Secured Multimedia Applications by Adopting RBAC, XML and Object-Relational Database,” In *Proceedings of the ACM Multimedia 2004 Conference*, New York, USA, October 10-16, pp. 432–435.
- [ChenSC05a] S.-C. Chen, M.-L. Shyu, and N. Zhao, “An Enhanced Query Model for Soccer Video Retrieval Using Temporal Relationships,” In *Proceedings of the 21st International Conference on Data Engineering (ICDE)*, Tokyo, Japan, 2005, pp. 1133–1134.
- [ChenSC07] S.-C. Chen, N. Zhao, and M.-L. Shyu, “Modeling Semantic Concepts and User Preferences in Content-Based Video Retrieval,” *International Journal of Semantic Computing*, in press.
- [ChenW01] W. Chen and S. F. Chang, “VISMMap: An Interactive Image/Video Retrieval System Using Visualization and Concept Maps,” In *Proceedings of International Conference on Image Processing (ICIP)*, Greece, October 2001, pp. 588–591.
- [Coyle04] L. Coyle and P. Cunningham, “Improving Recommendation Ranking by Learning Personal FeatureWeights,” In *Proceedings of the 7th European Conference on Case Based Reasoning*, Madrid, Spain, 2004, pp. 560–572.
- [CuVid] CuVid Columbia Video Search System.
<http://apollo.ee.columbia.edu/cuvidsearch/login.php>
- [Davis04] M. Davis and R. Sarvas, “Mobile Media Metadata for Mobile Imaging,” In *Proceedings of the IEEE International Conference on Multimedia and Expo*, Taipei, Taiwan, R.O.C., 2004, pp. 1707–1710.
- [DeMenthon03] D. DeMenthon and D. Doermann, “Video Retrieval using Spatio-Temporal Descriptors,” In *Proceedings of the 11th ACM International Conference on Multimedia (ACM MM)*, Berkeley, CA, USA, 2003, pp. 508–517.
- [Detyniecki00] M. Detyniecki, “Browsing a Video with Simple Constrained Queries over Fuzzy Annotations” In *Proceedings of the International Conference on Flexible Query Answering Systems (FQAS'2000)*, Warsaw, Poland, pp. 282–287.

- [Doulamis99] A. D. Doulamis, Y. S. Avrithis, N. D. Doulamis, and S. D. Kollias, "Interactive Content-based Retrieval in Video Databases Using Fuzzy Classification and Relevance Feedback," In *Proceedings of the IEEE International Conference on Multimedia Computing and Systems (ICMCS)*, Florence, Italy, 1999, pp. 954–958.
- [Dubois01] D. Dubois, H. Prade, and F. Sedes, "Fuzzy Logic Techniques in Multimedia Database Querying: A Preliminary Investigation of the Potentials," *IEEE Transactions on Knowledge and Data Engineering*, May 2001, vol. 13, no. 3, pp. 383–392.
- [Fan01] J. Fan, W. Aref, A. Elmagarmid, M.-S. Hacid, M. Marzouk, and X. Zhu, "Multiview: Multi-Level Video Content Representation and Retrieval," *Journal of Electrical Imaging*, 2001, vol. 10, no. 4, pp. 895–908.
- [Fan04] J. Fan, X. Zhu, A. K. Elmagarmid, W. G. Aref, and L. Wu, "ClassView: Hierarchical Video Shot Classification, Indexing, and Accessing," *IEEE Transactions on Multimedia*, 2004, vol. 6, no. 1, pp. 70–86.
- [Flickner95] M. Flickner et al., "Query by image and video content: The QBIC system," *Computer*, Sept. 1995, pp. 23–32.
- [Freund97] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *Journal of Computer and System Sciences (JCSS)*, 1997, vol. 55, no. 1, pp. 119–139.
- [Freund99] Y. Freund and R. E. Schapire, "A Short Introduction to Boosting," *Journal of Japanese Society for Artificial Intelligence*, 1999, vol. 14, no. 5, pp. 771–780.
- [Gaggi06] O. Gaggi and A. Celentano, "A Laboratory for Prototyping and Testing Multimedia Presentations," *International Journal of Software Engineering and Knowledge Engineering*, 2006, vol. 16, no. 4, pp. 615–642.
- [Gibbon04] D. Gibbon, L. Begeja, Z. Liu, B. Renger, and B. Shahrray, "Multimedia Processing for Enhanced Information Delivery on Mobile Devices," In *Proceedings of the Workshop on Emerging Applications for Wireless and Mobile Access*, New York, USA, 2004.
- [Gong01] Y. Gong and X. Liu, "Summarizing Video by Minimizing Visual Content Redundancies," In *Proceedings of the IEEE International Conference on Multimedia and Expo*, Tokyo, Japan, 2001, pp. 788–791.
- [Guillemot03] M. Guillemot, P. Wellner, D. Gatica-Perez, and J.-M. Odobez, "A Hierarchical Keyframe User Interface for Browsing Video over the Internet," In *Proceedings of the Ninth IFIP TC13 International Conference on Human-Computer Interaction (Interact 2003)*, Zurich, September 2003.
- [GVideo] Google Video Search. <http://video.google.com/>
- [Hertz03] T. Hertz, N. Shental, A. Bar-Hillel, and D. Weinshall, "Enhancing Image and Video Retrieval: Learning via Equivalence Constraints," In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003, pp. 668–674.

- [Hu01] J. Hu, J. Zhong, and A. Bagga, "Combined Media Video Tracking for Summarization," In *Proceedings of ACM Multimedia*, Ottawa, Canada, 2001, pp. 502–505.
- [Huang00] Q. Huang, A. Puri, and Z. Liu, "Multimedia Search and Retrieval: New Concepts, System Implementation, and Application." *IEEE Transactions on Circuits and Systems for Video Technology*, Aug. 2000, vol. 10, no. 5, pp. 679–692.
- [Ianeva04] T. Ianeva, A.P. de Vries, and T. Westerveld, "A Dynamic Probabilistic Multimedia Retrieval Model," In *Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME)*, Taipei, Taiwan, 2004, pp. 1607–1610.
- [IBM_Marvel] IBM Marvel: MPEG-7 Multimedia Search Engine.
<http://www.research.ibm.com/marvel/>
- [IBM_TRL] IBM TRL's MPEG-7 Authoring System.
http://www.trl.ibm.com/projects/digest/authoring_e.htm
- [IBM_VideoAnnEx] IBM VideoAnnEx Video Annotation Tool.
<http://www.research.ibm.com/MediaStar/VideoAnn.html>
- [Iqbal02] Q. Iqbal and J. K. Aggarwal, "CIRES: A System for Content-based Retrieval in Digital Image Libraries," In *Proceedings of International Conference on Control, Automation, Robotics and Vision (ICARCV)*, Singapore, December 2-5, 2002, pp. 205-210.
- [J2ME] Java 2 Platform, Micro Edition (J2ME). <http://java.sun.com/javame/>
- [J2SE] Java 2 Platform, Standard Edition (J2SE). <http://java.sun.com/javase/>
- [JavaWTK] Sun Java Wireless Toolkit. <http://java.sun.com/products/sjwtoolkit/>
- [JMF] Java Media Framework, <http://java.sun.com/products/java-media/jmf/>
- [John01] R. I. John and G. J. Mooney, "Fuzzy User Modeling for Information Retrieval on the World Wide Web," *Knowledge and Information Systems*, Feb. 2001, vol. 3, no. 1, pp. 81–95.
- [Jokela00] S. Jokela, M. Turpeinen, and R. Sulonen, "Ontology Development for Flexible Content," In *Proceedings of the 33rd Hawaii International Conference on System Sciences*, 2000, pp. 160–169.
- [Joshi02] J. B. D. Joshi, K. Li, H. Fahmi, B. Shafiq, and A. Ghafoor, "A Model for Secure Multimedia Document Database System in a Distributed Environment," *IEEE Transactions on Multimedia: Special Issue of on Multimedia Databases*, vol. 4, no. 2, June 2002, pp. 215–234.
- [Joshi05] J. B. D. Joshi, E. Bertino, U. Latif, and A Ghafoor, "Generalized Temporal Role Based Access Control Model," *IEEE Transactions on Knowledge and Data Engineering*, January 2005, vol. 17, no. 1, pp. 4–23.

- [Jourdan98] M. Jourdan, N. Layada, C. Roisin, L. Sabry-Ismaïl, and L. Tardif, "Madeus, an Authoring Environment for Interactive Multimedia Documents," In *Proceedings of ACM Multimedia'98*, Bristol, UK, September 1998, pp. 267–272.
- [Kang06] B.Y. Kang, D.W. Kim, and Q. Li, "Fuzzy Ranking Model Based on User Preference," *IEICE Transactions on Information and Systems*, June 2006, vol. E89D, no. 6, pp. 1971–1974.
- [Kosch01] H. Kosch, L. Böszörményi, A. Bachlechner, B. Dörflinger, C. Hanin, C. Hofbauer, M. Lang, C. Riedler, and R. Tusch, "SMOOTH - A Distributed Multimedia Database System," In the *Proceedings of 27th International Conference on Very Large Data Bases (VLDB'2001)*, Rome, Italy, pp. 713–714.
- [Kuchinsky99] A. Kuchinsky, C. Pering, M. Creech, D. Freeze, B. Serra, and J. Gwizdka, "Fotofile: A Consumer Multimedia Organization and Retrieval System," In *Proceedings of ACM CHI Conference*, May 1999, pp. 496–503.
- [Lahti06a] J. Lahti, M. Palola, J. Korva, U. Westermann, K. Pentikousis, and P. Pietarila, "A Mobile Phone based Context-aware Video Management Application," In *Proceedings of SPIEIS&T Electronic Imaging (Multimedia on Mobile Devices II)*, San Jose, California, USA, 2006, vol. 6074, pp. 83194.
- [Lahti06b] J. Lahti, K. Pentikousis, and M. Palola, "MobiCon: Mobile Video Recording with Integrated Annotations and DRM," In *Proceedings of IEEE Consumer Communications and Networking Conference (IEEE CCNC)*, Las Vegas, Nevada, USA, 2006, pp. 233–237.
- [LiQ01] Q. Li, J. Yang, and Y. T. Zhuang, "Web-based Multimedia Retrieval: Balancing out between Common Knowledge and Personalized Views," In *Proceedings of 2nd International Conference on Web Information System and Engineering*, 2001, pp. 100–109.
- [Ma03] M. MA, V. Schillings, T. Chen, and C. Meinel, "T-Cube: A Multimedia Authoring System for Learning," In *Proceedings of E-Learning 2003*, Phoenix, AZ, 2003, pp. 2289–2296.
- [Martin02] M.J. Martin-Bautista, D.H. Kraft, M.A. Vila, J. Chen, and J. Cruz, "User Profiles and Fuzzy Logic for Web Retrieval Issues," *Special Issue of Journal of Soft Computing*, 2002, vol. 6, pp. 365–372.
- [Moyer01] M. J. Moyer and M. Ahamad, "Generalized Role-Based Access Control," In *Proceedings of the 21st International Conference on Distributed Computing Systems (ICDCS 2001)*, April 2001, pp. 391–398.
- [Muneesawang03] P. Muneesawang and L. Guan, "Automatic Relevance Feedback for Video Retrieval," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Hong Kong, 2003, vol. 3, pp. 1–4.
- [Ngo01] C.-W. Ngo, T.-C. Pong, and H.-J. Zhang, "On Clustering and Retrieval of Video Shots," In *Proceedings of the 9th ACM International Conference on Multimedia*, Ottawa, Canada, 2001, pp. 51–60.

- [Odobez03] J.-M. Odobez, D. Gatica-Perez, and M. Guillemot, "Video Shot Clustering using Spectral Methods," In *Proceedings of 3rd International Workshop on Content-Based Multimedia Indexing (CBMI)*, Rennes, France, 2003, pp. 94–102.
- [Pentland94] A. Pentland, R.W. Picard, and S. Sclaroff, "Photobook: Tools for Content-Based Manipulation of Image Databases," In *Proceedings of Storage and Retrieval for Image and Video Databases II, SPIE*, Bellingham, Washington, 1994, vol. 2185, pp.34–47.
- [PostgreSQL] PostgreSQL: An Object-Relational Database.
<http://www.postgresql.org/>
- [Rabiner93] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*, Prentice Hall, 1993, ISBN: 0130151572.
- [Rui97] Y. Rui, T. S. Huang, and S. Mehrotra, "Content-based Image Retrieval with Relevance Feedback in MARS," In *Proceedings of the IEEE International Conference on Image Processing*, volume II, 1997, pp. 815–818.
- [Rui98] Y. Rui, T. S. Huang, and S. Mehrotra, "Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval," *IEEE Transactions on Circuit and Video Technology, Special Issue on Segmentation, Description, and Retrieval of Video Content*, 1998, vol. 18, no. 5, pp. 644–655.
- [Sachi05] A. Sachinopoulou, S.-M. Mäkelä, S. Järvinen, U.Westermannl, J. Peltola, and P. Pietarila, "Personal Video Retrieval and Browsing for Mobile Users," In *Proceedings of SPIE Multimedia on Mobile Devices*, San Jose, California, USA, 2005, pp. 219–230.
- [Sandhu96] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role Based Access Control Models," *IEEE Computer*, vol. 29, no. 2, February 1996, pp. 38–47.
- [Shyu03] M.-L. Shyu, S.-C. Chen, M. Chen, C. Zhang, and C.-M. Shu, "MMM: A Stochastic Mechanism for Image Database Queries," In *Proceedings of the IEEE Fifth International Symposium on Multimedia Software Engineering (MSE2003)*, Taichung, Taiwan, ROC, December 10-12, 2003, pp. 188–195.
- [Shyu04a] M.-L. Shyu, S.-C. Chen, M. Chen, and S. H. Rubin, "Affinity-Based Similarity Measure for Web Document Clustering," In *Proceedings of the 2004 IEEE International Conference on Information Reuse and Integration (IRI)*, Las Vegas, Nevada, USA , 2004, pp. 247–252.
- [Shyu04b] M.-L. Shyu, S.-C. Chen, M. Chen, and C. Zhang, "A Unified Framework for Image Database Clustering and Content-based Retrieval," In *Proceedings of the Second ACM International Workshop on Multimedia Databases (ACM MMDB)*, Arlington, VA, USA , 2004, pp. 19–27.
- [Shyu04c] M.-L. Shyu, S.-C. Chen, M. Chen, and C. Zhang, "Affinity Relation Discovery in Image Database Clustering and Content-based Retrieval," In *Proceedings of ACM Multimedia 2004 Conference*, New York, USA, pp. 372–375.

- [Shyu04d] M.-L. Shyu, S.-C. Chen, and C. Zhang, "A Stochastic Content-Based Image Retrieval Mechanism," Edited by Sagarmay Deb, *Multimedia Systems and Content-based Image Retrieval*, Idea Group Publishing, 2004, ISBN: 1-59140-265-4, pp. 302–320.
- [SMIL] Synchronized Multimedia Integration Language (SMIL). <http://www.w3.org/TR/smil20/>
- [Smith96] J. R. Smith and S. F. Chang, "VisualSEEK: A Fully Automated Content-based Image Query System," In *Proceedings ACM International Conference of Multimedia*, Boston, Nov 1996, pp. 87–98.
- [Snoek03] C. G. M. Snoek and M. Worring, "Goalgle: A Soccer Video Search Engine," In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2003)*, Baltimore, USA, July 6-9, 2003.
- [Snoek05] C. G. M. Snoek and M. Worring, "Multimedia Event based Video Indexing using Time Intervals," *IEEE Transactions on Multimedia*, vol. 7, no. 4, 2005, pp. 638–647.
- [Truveo] AOL Truveo Video Search. <http://www.truveo.com/>
- [Tseng02] B.L. Tseng, C. Lin, and J. Smith, "Video Summarization and Personalization for Pervasive Mobile Devices," In *Proceedings of the IS&T/SPIE Symposium on Electronic Imaging: Science and Technology Storage & Retrieval for Image and Video Databases*, 2002, SPIE vol. 4676, pp. 359–370.
- [Virage] Virage Search Engine. <http://www.virage.com>
- [WebSEEk] WebSEEk: A Content-Based Image and Video Search and Catalog Tool for the Web. <http://persia.ee.columbia.edu:8008/>
- [Xie03] L. Xie, S.-F. Chang, A. Divakaran, and H. Sun, "Unsupervised Discovery of Multilevel Statistical Video Structures Using Hierarchical Hidden Markov Models," In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, July 2003, vol. 3, pp. 29–32.
- [XML] Extensible Markup Language (XML) 1.0 (Second Edition) – W3C Recommendation 6 October 2000. <http://www.w3.org/TR/2000/REC-xml-20001006.pdf>
- [Yan03] R. Yan, A. G. Hauptmann, and R. Jin, "Negative Pseudo-Relevance Feedback in Content-based Video Retrieval," In *Proceedings of the Eleventh ACM International Conference on Multimedia (ACM MM)*, 2003, Berkeley, CA, USA, pp. 343–346.
- [Yan04] R. Yan, J. Yang, and A. Hauptmann, "Learning Query-Class Dependent Weights in Automatic Video Retrieval," In *Proceedings of ACM Multimedia 2004*, USA, 2004, pp. 548–555.
- [Yilmaz00] A. Yilmaz and M. Shah, "Shot Detection Using Principal Coordinate System," In *Proceedings of IASTED Internet and Multimedia Systems and Applications Conference*, Las Vegas Nevada, November 2000, pp. 168–173.

- [Yoshitaka99] A. Yoshitaka and T. Ichikawa, "A Survey on Content-Based Retrieval for Multimedia Databases," *IEEE Transactions on Knowledge and Data Engineering*, January/February 1999, vol. 11, no. 1, pp. 81–93.
- [Youtube] Youtube. <http://www.youtube.com/>
- [YVideo] Yahoo Video Search.
<http://video.search.yahoo.com/>
- [Zhao06a] N. Zhao, S.-C. Chen, and M.-L. Shyu, "Video Database Modeling And Temporal Pattern Retrieval Using Hierarchical Markov Model Mediator," In *Proceedings of the First IEEE International Workshop on Multimedia Databases and Data Management (IEEE-MDDM)*, in conjunction with the 22nd IEEE International Conference on Data Engineering (ICDE), 2006, Atlanta, Georgia, USA.
- [Zhao06b] N. Zhao, S.-C. Chen, M.-L. Shyu, and S. H. Rubin, "An Integrated and Interactive Video Retrieval Framework with Hierarchical Learning Models and Semantic Clustering Strategy," In *Proceedings of the 2006 IEEE International Conference on Information Reuse and Integration (IEEE-IRI)*, September 2006, Hawaii, USA, pp. 438–443.
- [Zhao07a] N. Zhao, S.-C. Chen, and M.-L. Shyu, "User Adaptive Video Retrieval on Mobile Devices," accepted for publication, Edited by Laurence T. Yang, Agustinus Borgy Waluyo, Jianhua Ma, Ling Tan and Bala Srinivasan, *Mobile Intelligence: When Computational Intelligence Meets Mobile Paradigm*, John Wiley & Sons Inc.
- [Zhao07b] N. Zhao, S.-C. Chen, and S. H. Rubin, "Automated Multimedia Systems Training Using Association Rule Mining", In *Proceedings of the 2007 IEEE International Conference on Information Reuse and Integration (IEEE-IRI)*, August 13-15, Las Vegas, USA, pp. 373–378.

VITA

NA ZHAO

May 31, 1981 Born, Linyi, Shandong, P. R. China

July 2001 B.E., Computer Science and Application
Northeastern University, P. R. China

April 2003 M.S., Computer Science
Florida International University, Miami, Florida, USA

2003-2007 Doctoral Candidate in Computer Science
Florida International University, Miami, Florida, USA

PUBLICATIONS AND PRESENTATIONS

Zhao, N., Chen, M., Chen, S.-C., and Shyu, M.-L., (2007). "User Adaptive Video Retrieval on Mobile Devices," accepted for publication, Edited by Laurence T. Yang, Agustinus Borgy Waluyo, Jianhua Ma, Ling Tan and and Bala Srinivasan, *Mobile Intelligence: When Computational Intelligence Meets Mobile Paradigm*, John Wiley & Sons Inc.

Chen, S.-C., Zhao, N., and Shyu, M.-L., (2007). "Modeling Semantic Concepts and User Preferences in Content-Based Video Retrieval," *International Journal of Semantic Computing*, in press.

Zhao, N., Chen, S.-C., and Rubin, S. H., (2007) "Automated Multimedia Systems Training Using Association Rule Mining", In *Proceedings of the 2007 IEEE International Conference on Information Reuse and Integration (IEEE-IRI)*, August 13-15, Las Vegas, USA, pp. 373–378.

Zhang, K., Chen, S.-C., Singh, P., Saleem, K., and Zhao, N., (2006). "A 3D Visualization System for Hurricane Storm Surge Flooding," *IEEE Computer Graphics and Applications (IEEE CG&A)*, vol. 26, Issue 1, pp. 18–25.

Zhao, N., Chen, S.-C., Shyu, M.-L., and Rubin, S. H., (2006). "An Integrated and Interactive Video Retrieval Framework with Hierarchical Learning Models and Semantic Clustering Strategy," In *Proceedings of the 2006 IEEE International Conference on Information Reuse and Integration (IEEE-IRI)*, September 2006, Hawaii, USA, pp. 438–443.

Chatterjee, K., Saleem, K., Zhao, N., Chen, M., Chen, S.-C., and Hamid, S., (2006). "Modeling Methodology for Component Reuse and System Integration for Hurricane Loss Projection Application," In *Proceedings of the 2006 IEEE International Conference on Information Reuse and Integration (IEEE-IRI)*, September 2006, Hawaii, USA, pp. 57–62.

Zhao, N., Chen, S.-C., and Shyu, M.-L., (2006). "Video Database Modeling and Temporal Pattern Retrieval using Hierarchical Markov Model Mediator," In *Proceedings of the First IEEE International Workshop on Multimedia Databases and Data Management (IEEE-MDDM)*, in conjunction with *IEEE International Conference on Data Engineering (ICDE)*, April 2006, Atlanta, Georgia, USA.

Singh, P. A., Zhao, N., Chen, S.-C., and Zhang, K., (2005). "Tree Animation for a 3D Interactive Visualization System for Hurricane Impacts," In *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME)*, July 2005, Amsterdam, The Netherlands, pp. 598–601.

Chen, S.-C., Shyu, M.-L., and Zhao, N., (2005). "An Enhanced Query Model for Soccer Video Retrieval Using Temporal Relationships," In *Proceedings of the 21st International Conference on Data Engineering (ICDE)*, April 5-8, 2005, Tokyo, Japan, pp. 1133–1134.

Shyu, M.-L., Haruechaiyasak, C., Chen, S.-C., and Zhao, N., (2005). "Collaborative Filtering via Association Rule Mining from User Access Sequences," In *Proceedings of the International Workshop on Challenges in Web Information Retrieval and Integration (WIRI)*, in conjunction with *The 21st International Conference on Data Engineering (ICDE)*, April 8-9, 2005, Tokyo, Japan, pp. 128–133.

Chen, S.-C., Shyu, M.-L., and Zhao, N., (2004). "SMARXO: Towards Secured Multimedia Applications by Adopting RBAC, XML and Object-Relational Database," In *Proceedings of the 12th Annual ACM International Conference on Multimedia (ACM-MM)*, October 2004, New York, USA, pp. 432–435.

Chen, S.-C., Hamid, S., Gulati, S., Zhao, N., Chen, M., Zhang, C., and Gupta, P., (2004). "A Reliable Web-based System for Hurricane Analysis and Simulation," In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics 2004 (SMC)*, October 2004, Hague, The Netherlands, pp. 5215–5220.

Chen, S.-C., Shyu, M.-L., Zhao, N., and Zhang, C., (2003). "Component-Based Design and Integration of a Distributed Multimedia Management System," In *Proceedings of the 2003 IEEE International Conference on Information Reuse and Integration (IEEE-IRI)*, October 2003, Las Vegas, Nevada, USA, pp. 485–492.

Chen, S.-C., Shyu, M.-L., Zhao, N., and Zhang, C., (2003). "An Affinity-Based Image Retrieval System for Multimedia Authoring and Presentation," In *Proceedings of the 11th Annual ACM International Conference on Multimedia (ACM-MM)*, November 2003, Berkeley, CA, USA, pp. 446–447.

Chen, S.-C., Shyu, M.-L., and Zhao, N., (2003). "MediaManager: A Distributed Multimedia Management System for Content-Based Retrieval, Authoring and Presentation," In *Proceedings of the 9th International Conference on Distributed Multimedia Systems (DMS)*, September 2003, Miami, Florida, USA, pp. 17–22.