The Multimedia Metadata Community

(http://www.multimedia-metadata.info)

Presents the

Proceedings of the

9th Workshop on Multimedia Metadata (WMM'09)

Toulouse, France, March 19-20, 2009

Edited by

Romulus Grigoras, IRIT - University of Toulouse
Vincent Charvillat, IRIT - University of Toulouse
Ralf Klamma, RTWH Aachen University
Harald Kosch, University of Passau

Foreword

We have the pleasure to organize the 9th Workshop on Multimedia Metadata which comes to support the work done by the Multimedia Metadata Community (http://www.multimedia-metadata.info). The 2009 workshop has a special focus on context-aware mobile multimedia services and is held in conjunction with the 13th French Multimedia Conference on Compression and Representation of Audiovisual Signals (CORESA 2009). Both events share joint panels and sessions bringing researchers and industry experts to fruitful discussions.

This year's program features invited renowned researchers. Timo Ojala, from the University of Oulu, will give an invited talk on the opportunities that mobile devices equipped with wireless data transmission provide for context-aware and ubiquitous computing. Christian Timmerer (ITEC Klagenfurt) and Stéphane Pateux (Orange Labs), researchers working actively within multimedia standardisation bodies, are animating a discussion on the current state of MPEG standards and the acceptance status in industry. They will also provide insights into future evolutions of these standards.

We have a quality programme composed of 9 regular papers and a doctoral symposium featuring 3 papers. Our first thanks go to the reviewers, who provided timely and thorough reviews. Their suggestions allowed authors to better their contributions.

Naturally, our thanks also go to the joint sponsors of CORESA'09 and WMM'09: CNES, Orange Labs, Noveltis, Fitting Box, IRIT, INP Toulouse, Grand Toulouse and Région Midi-Pyrénées. Their financial and logistic support has been essential to the organisation of our workshop.

We wish you a productive and enriching workshop and an excellent stay in Toulouse.

Your workshop co-chairs,

Romulus Grigoras, IRIT - University of Toulouse
Vincent Charvillat, IRIT - University of Toulouse
Ralf Klamma, RTWH Aachen University
Harald Kosch, University of Passau

Case studies on context-aware mobile multimedia services

Timo Ojala

MediaTeam Oulu research group
Department of Electrical and Information Engineering
University of Oulu
FI-90014 University of Oulu
timo.ojala@ee.oulu.fi

Abstract: This paper explores the design, implementation and evaluation of context-aware mobile multimedia services by presenting six case studies on different application domains. The case studies highlight the opportunities that mobile devices equipped with wireless data transmission provide for context-aware and ubiquitous computing. Special emphasis is placed on empirical evaluation of the services in the field in form of a user evaluation with genuine end users in real environment of use.

1 Introduction

The research community has introduced a large volume of context-aware systems [BDR07], which have adopted different definitions of the slippery notion of "context" [Do04]. The systems typically involves mechanisms for context acquisition, preprocessing, representation, management and utilization in an application. The raw context data is acquired from different types of sources or "sensors", where the term "sensor" has to be understood very broadly. The raw context data may be subjected to different forms of preprocessing, for example to deal with missing data or to elicit higher level abstractions such as logical relationships or activities. Different models have been developed for representing context data in formats understood by computers and users, ranging from simple key-value pairs to extensive ontologies [Bo07]. Different types of centralized and distributed context management architectures have been proposed for storing and distributing context data to applications. They then utilize context for various purposes such as to adapting the user interface or to retrieving contextually relevant information.

When evaluating context-aware mobile systems, we have to make a distinction whether we evaluate the system architecture [OSW07] or the user interface [KS03]. Evaluating the user interface or usability of mobile systems is difficult due to extra interaction and evaluation challenges in the mobile domain. The interaction challenges include the mobile context of use, rich device functionality, small device size, lack of direct manipulation, and lack of standardization in handset and software design.

The evaluation challenges relate to data collection (technical, social and legal constraints) and uncontrollable variables (e.g. weather, ambient noise and attention destructors). Important decisions to be made regarding evaluation include what is evaluated (interface metaphors, mental models or UI elements), how (empirical vs analytical) and where (lab vs field). While a lab experiment does not provide a proper simulation of the true mobile context and the real-word factors affecting behavior and performance are missing, a field experiment in turn is time consuming, expensive, and suspect to data collection difficulties and uncontrollable external variables.

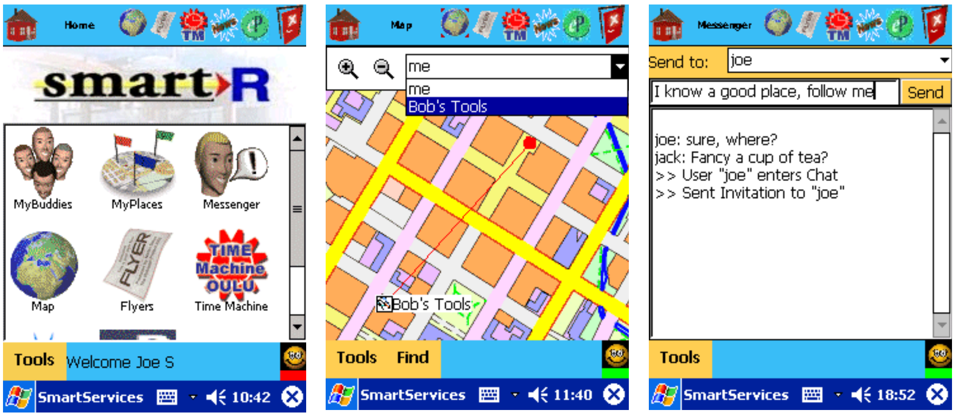
The six case studies presented in this paper demonstrate the utilization of context data in different types of context-aware mobile multimedia services. Each service employs a simple key-value context model and centralized context management architecture for storing or retrieving contextually relevant information. First five case studies originate from the Rotuaari – Context-Aware Mobile Multimedia Services research project [Ro03]. Each of them involved empirical evaluation in the field in form of a user evaluation with real users in true environment of use. It is a fundamental usability assessment method providing direct information about how the system is used and what are the exact problems [Ni94]. The sixth case study, panOULU Luotsi, is a joint effort of the Wireless Cities project [Wi05] and the panOULU network [pa09]. Some of the services are now in “production“ use, either as a public service or as a commercial product.

2 SmartRotuaari

SmartRotuaari was an early demonstration of the new mobile multimedia services that emerging wireless broadband Internet would eventually facilitate. SmartRotuaari comprised of a wireless multi-access network, SmartWare architcture for deploying context-aware mobile multimedia services, a web-based CPI (Content Provider Interface) for content management and a collection of functional prototype services [Oj03]. The design and implementation of the system started in 2002, leading to empirical evaluation in form of a large-scale field trial executed at downtown Oulu in fall 2003. SmartRotuaari was motivated by the needs of both companies (i.e. mobile service providers, technology providers, retailers etc.) and consumers as end users of mobile services. While companies have the most comprehensive knowledge about their business (“market pull”), some of them, especially smaller retailers, are not necessarily aware of the possibilities offered by the new technology (“technology push”). These aspects were studied in form of surveys, joint workshops and one-to-one discussions with local businesses.

The multi-access network comprised of a GPRS network and a WLAN (IEEE 802.11b). The WLAN emulated the emerging wireless broadband Internet access with much higher data transfer rate and lower latencies than GPRS. We built the WLAN ourselves, managing to install 11 WLAN access points around downtown Oulu by the start of the first field trial in late August 2003. Later the WLAN network was expanded, contributing to the founding of the panOULU network in October 2003 [pa09], which eventually proved to be the most valuable outcome of the project.

The SmartWare architecture included server components for service access, user positioning, instant messaging, real-time presence management and database access. The architecture facilitated using different context attributes represented as key-value pairs in service provisioning: *time* (e.g. time range when a service was active), *location* (absolute and relative locations of the user and/or a service provider, for example a service could be triggered if the user was within a specific distance of a service provider, location of the user could be provided by GPS, WLAN positioning or manual entry), *weather* (real-time weather observation, temperature and wind speed divided to non-overlapping ranges), *user profile* (e.g. age, marital status, education, occupation, income, personal interests entered upon registration) and *user presence* (availability and mood set to any of the seven predefined values).



Send to: joe

I know a good place, follow me

joe: sure, where?
jack: Fancy a cup of tea?
>> User "joe" enters Chat
>> Sent Invitation to "joe"

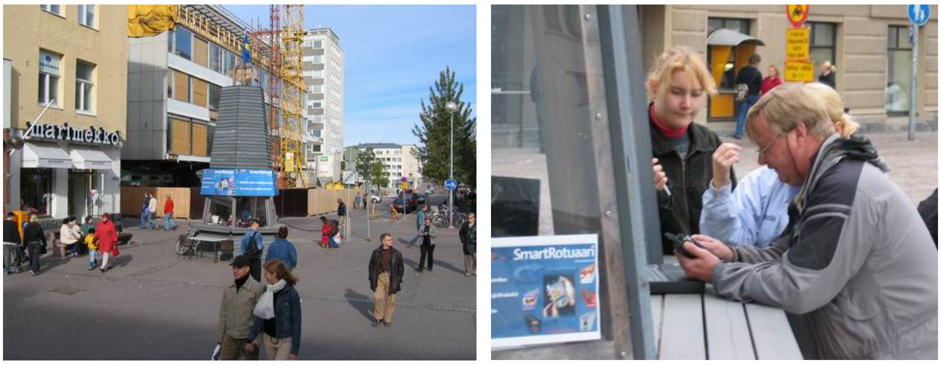
(a) (b) (c)

Figure 1. Example UI screenshots of the SmartRotuaari services: (a) "desktop"; (b) map-based guidance; (c) personal communication.

The prototype services were implemented with a monolithic Java client for PDA's. They included a service directory, map-based guidance, personal communications, Time Machine Oulu (see Section 3), mobile ads (see Section 5), personalized news and mobile payment. The "desktop" shown in Fig. 1(a) provided access to individual services. Map-based guidance (Fig. 1(b)) provided visualization of the location of a place relative to the user's current location shown as red dot. A place could refer to any entry in the service directory, a 'buddy', or a location of personal interest specified earlier by the user. Personal communications was supported in form of peer-to-peer and group chat, which were implemented as simple text-based chat (Fig. 1(c)).

The user could maintain a list of ‘buddies’ and invite them to a chat. Further, the user could set his presence status (‘mood’) to any of the predefined alternatives. The user could choose whether his location and/or presence status were shown to his ‘buddies’ and other users. Personalized news feed was provided so that the user could designate whether (s)he found a particular article interesting or not. A personalization engine then updated the user’s profile accordingly and provided first those of the incoming news that matched the user’s interest profile. Mobile payment was implemented with an external micropayment server, which facilitated payments for on-line content with real money.

A subset of the prototype services, service directory, map-based guidance, mobile ads and TimeMachine Oulu, were included in the field trial running from late August 2003 till the end of September 2003. The field trial was coordinated from an office established in a small hut placed at the very heart of downtown Oulu (Fig. 2). The office was staffed with researchers, who persuaded passers-by to sign up as test users, helped test users in creating a user profile and in using the iPAQ and services, and collected feedback via a questionnaire and occasional interviews. Each test user was awarded with a voucher to a nearby café after the test session.



(a) (b)

Figure 2. (a) Field trial office at the Rotuaari pedestrian street; (b) a test user is signing up.

Recruiting voluntary test users from the general public proved rather difficult. Eventually we had few hundred test users, of which 193 respondents filled in an extensive questionnaire. The rather small spatial coverage of the WLAN network was the most serious technical limitation. The test users had difficulties in understanding that they could lose connectivity indoors or when leaving the downtown, which led to various usability problems. Nevertheless, the field trial provided a very valuable lesson in organizing a large field experiment in a public laboratory.

3 TimeMachine Oulu

TimeMachine Oulu provided a dynamic, interactive and context-aware 3D virtual model nhof historical Oulu, to be used with the web browser of a WLAN-equipped PDA [POO03]. The development of the model was motivated and facilitated by the availability of high quality historical data of the buildings in Oulu between two devastating fires in 1822 and in 1882. The 3D virtual model was generated dynamically from the building database using VRML. The model was simplified (Fig. 3), not photorealistic, adapted to the limited rendering power, network bandwidth and storage of the mobile device, and the simple lighting model of the VRML. The user could "time travel" in the model by increasing/decreasing the "current year". The model was context-aware, placed at the current location of the user determined by WLAN positioning of the mobile device. The model was interactive in the sense that you could move around in the model and access the objects, for example to ask who lived in a particular building in the "current" year.

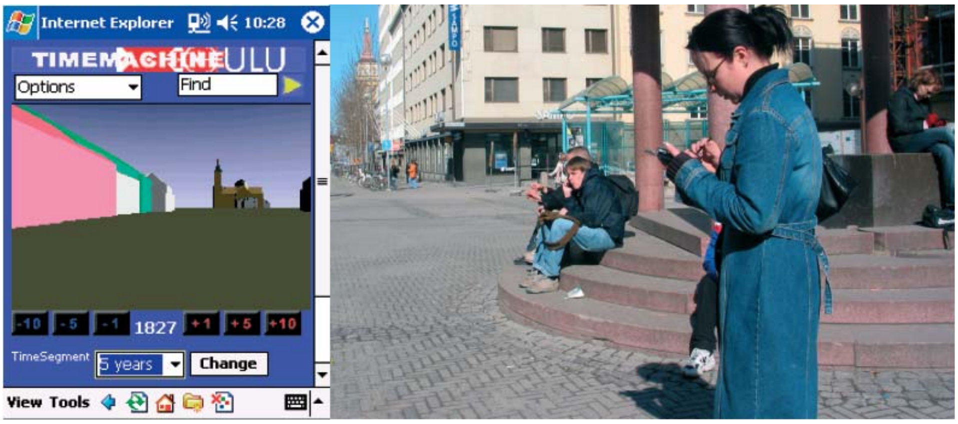


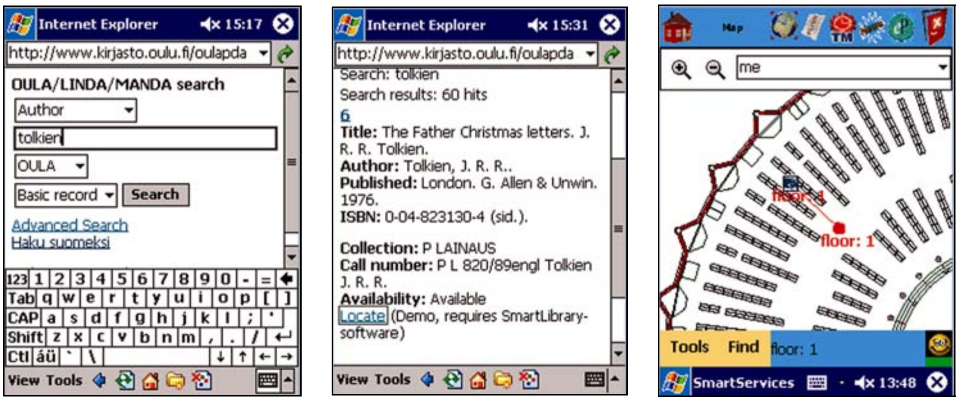
Figure 3. Two views of a particular location at downtown Oulu in 1827 (left) and 2003 (right). The old church from 1826 is still standing in the background. The building in front of it is only "recently" constructed.

TimeMachine was tested with a small task-based user-based evaluation at downtown Oulu. Ten test users were asked to complete four tasks using TimeMachine: T1: Go to year 1826; T2: Move around the church; T3: Find any red, yellow or green building and name its owner and insurance number; and T4: Find out the width and length of mayor Appelgren's house. After completing the tasks the test users completed a questionnaire addressing various aspects of the perceived usability and user experience. All ten users were able to complete tasks T1, T2 and T3 successfully, while seven were able to complete task T4. Seven users thought that TimeMachine was slow and six users found the visual quality of TimeMachine sufficient. To conclude, TimeMachine was a prime example of an interesting application made possible by high quality historical content.

4 SmartLibrary

SmartLibrary was a location-aware mobile library service, assisting library patrons in finding books and other objects. Back in 2003, SmartLibrary was the first OPAC (Online Public Access Catalogues) search interface tailored for mobile devices atop of Voyager, a widespread library management system. The traditional solution in libraries is to classify the books into holdings and shelf classes. This solution works well if the user is familiar with the shelf classification. For larger libraries, however, there can be tens of holdings, hundreds of classes and thousands of shelves. This results in especially novice library users consulting the library personnel for personal guidance, which consumes the library's resources.

SmartLibrary version 1 [ARO03] provided map-based guidance to the target bookshelf on a PDA equipped with WLAN (IEEE 802.11b) connectivity. The user's location could be estimated with WLAN positioning, which enabled dynamic guidance of the user towards the desired book. The service was a completely software-based solution, which could be provisioned atop a WLAN installed for wireless Internet access, without any additional hardware. SmartLibrary version 1 was deployed in the main library of University of Oulu on top of OULA, an online catalog based on Voyager. PDA's web browser was used to browse the OULA-pda, a web interface tailored for small devices such as PDA's. The query results provided by the OULA-pda were augmented with a link, which started a separate Java-based guidance application. Fig. 4(a) illustrates the definition of a query for a book authored by Tolkien. Fig. 4(b) shows the presentation of the entries matching the query. By clicking the "Locate"- link of the book of interest the user got access to the map-based guidance visualized in Fig. 4(c). In the map view the red dot denotes the location of the user, while the rectangular icon shows the shelf where the requested book resides.



Search: tolkien
Search results: 60 hits
6
Title: The Father Christmas letters. J. R. R. Tolkien.
Author: Tolkien, J. R. R..
Published: London. G. Allen & Unwin. 1976.
ISBN: 0-04-823130-4 (sid.).

Collection: P LAINAUS
Call number: P L 820/89engl Tolkien J. R. R.
Availability: Available
Locate (Demo, requires SmartLibrary-software)

(a) (b) (c)

Figure 4. (a) The query definition UI of OULA-pda. (b) The results of the query. (c) Visualization of the map-based guidance to the shelf containing the book searched for.

We conducted a task-based user evaluation of the SmartLibrary service with 32 randomly selected library patrons. Each user was given two tasks: first, to find a certain book by Tolkien, and second, any issue of a particular economic science periodical. Half of the users completed the first task using public desktop library terminals providing shelf classification, and the other half using SmartLibrary providing map-based guidance. The terminal was changed between the tasks. After the tasks were completed, the users were asked to fill in a questionnaire containing multiple-choice questions and space for feedback. The users were asked which of the two methods they would prefer for finding books in the library and why. All males and 64 % of females chose map-based guidance, which they also found less laborious to use than shelf classification. As expected, the assessment of the laboriousness of shelf classification correlated heavily with the users' experience in using it. The main usability problems were related to the guidance application, which was slow and worked only in particular PDA models. Switching between the web-based OULA-pda and the separate guidance application was considered awkward. The users also had difficulties in orienting themselves on maps, due to poor graphics of the maps.

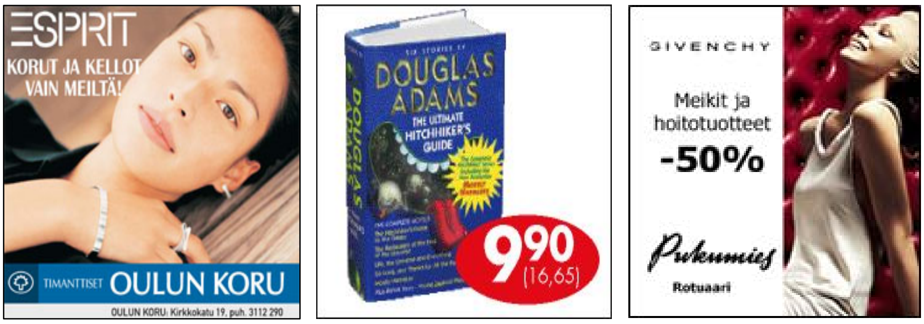
SmartLibrary version 2 [Ai04] was re-designed to address the problems of the first version. The user interface was provided for the (X)HTML browsers in desktops, PDAs and high-end mobile phones, hence integration with web-based OPACs would be seamless. The graphics of the floor plan maps were designed to be simple and clear. Different symbols on the map were color-coded: walls and other fixed structures were drawn with black, book shelves with blue, and tables with yellow. Target areas and their names were superimposed on the map. The users could also position themselves relative to pre-defined landmarks, e.g. a circulation desk and stairs, shown on the map upon request. A separate web-based CPI (Content Provider Interface) was provided for the purpose of maintaining information of shelf-classes and landmarks.

SmartLibrary version 2 was tested with a task-based user evaluation, where library patrons and staff were asked to find three books with different means: with the user's own way, SmartLibrary with a public desktop terminal, SmartLibrary with a PDA over WLAN connectivity, and SmartLibrary with a mobile phone over GPRS data connection. Most novice users preferred SmartLibrary over the shelf classification, whereas more experienced patrons and library staff preferred the shelf classification. The users considered SmartLibrary most useful on public desktop terminals, i.e. just map-based guidance without the WLAN positioning of the user.

SmartLibrary is still in 'production' use at the libraries of the University of Oulu in two different forms. Map-based guidance is provided as a web service for locating shelf-classes, study rooms, equipment and other resources, and it has become quite popular among library patrons. Also, an online search web interface tailored for mobile devices is provided. However, the WLAN positioning of the mobile device has been discarded, due to the limited range of supported WLAN devices and high maintenance cost.

5 Mobile advertising

Three different context-aware systems for permission-based mobile advertising were developed in the Rotuaari project. The first advertising system was part of the SmartRotuaari system described in Section 2. A mobile ad was a SMIL 2.0 compliant multimedia message delivered to a PDA over WLAN connection. The ad was authored and activated by the advertiser using a web-based CPI. The advertiser defined the ad profile, i.e. the context attributes (time, relative location, weather, user profile and/or user mood) that were required for the ad to “trigger“. Similarly, the user could define whether (s)he wanted to receive ads and from what relative range from her/his current location. If there was a match, the ad was delivered to the user‘s device. 18 local companies were recruited to produce free ads promoting their products and services, of which 12 eventually provided ads (Fig. 5) during the field trial. The test users‘ feedback supported our own observation of the limited added value provided by the bulk of the mobile ads, which were content to just provide the name and address of a store, for example. [Oj03]



(a) (b) (c)

Figure 5. Example mobile ads: (a) jewelry store advertising a jewelry brand; (b) discount offer from a bookstore; (c) cosmetics discount ad.

In the second advertising system ad delivery was based on a telco grade MMSC and the user device was a mobile phone equipped with GPRS data connection. The mobile ads were again created with a web-based CPI, including tailoring of the content for different user devices and the specification of the ad profile. If there was a match between the ad profile and the user profiles, the ad was delivered either as MMS messages or as XHTML Mobile Profile pages with WAP Push. The advertising system was employed in two large field trials in 2004 and in 2005, and in a third smaller field trial in 2006. For example, in the 2004 trial 44 businesses were recruited as advertisers, producing 81 different ads that were delivered 11370 times in total to 610 test users. The advertising system was hampered by technical difficulties such as variations in the implementation of MMS players in different phones. [Ko07a]

The third mobile advertising system called B-MAD was based on Bluetooth and WAP Push [Aa04]. A Bluetooth sensor was configured to periodically scan for the globally unique Bluetooth device addresses of passing by Bluetooth user devices. The sensor sent the BT addresses of detected devices over a WAP connection to an ad server, together with a location identifier. The ad server mapped the BT addresses to the user phone numbers and checked from the database if there were any ads associated with the location that had not been delivered to the user. The undelivered ads were delivered as XHTML Mobile Profile pages using WAP Push. User’s location was the only context attribute used for triggering ad delivery. The B-MAD was evaluated with a small-scale field trial where Bluetooth sensors were placed in eight stores providing 11 ads in total. Two ads from a particular store were temporally related so that if the user stayed nearby the store for a certain time after receiving the first ad, (s)he would receive a second ad including a gift certificate. 35 test users were asked to walk a designated route bypassing the stores and the delivery of ads was logged in the server. Due to the long Bluetooth scanning delay the average positioning latency was 25 seconds and the sensor was not guaranteed to detect every Bluetooth device passing by at walking speed. Further, ad delivery with WAP Push took almost 12 seconds on average. The combined total latency of 37 seconds meant that at times the user received the ad fairly far away from the sending store.

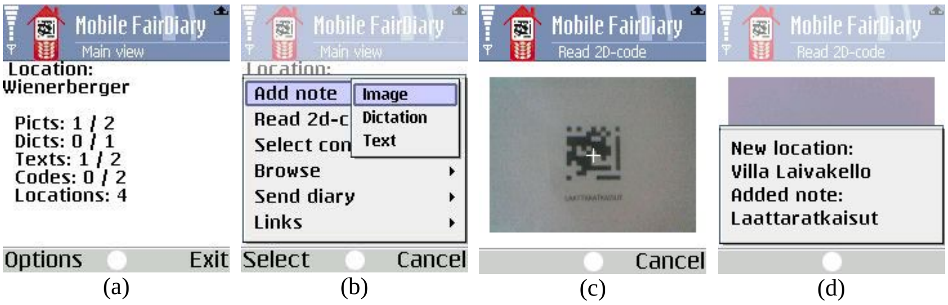
The three mobile advertising systems and their evaluation in five field trials provided us and advertisers valuable lessons in mobile advertising. The advertisers and the few agencies they used for authoring mobile ads on their behalf had great difficulties in understanding how personal channel a mobile device is. They were mostly content to just reproduce the ads they used in printed mass media. They were also reluctant to personalize the ads and consequently the customers found the ads rather useless. At the beginning we also provided too many context attributes to choose from, which led to sparse context spaces with few hits per ad. We learned that there was no room for technical glitches, which were partly our own doing and partly due to problems in commercial products. Further, we learned that being so much ahead of time was not necessarily a bonus, as other stakeholders had difficulties in buying our vision. Nevertheless, some advertisers reported that they managed to create new business by participating in our trials.

6 Mobile Fair Diary

Mobile Fair Diary (MFD) was designed to allow a housing fair visitor to make a personalized digital recording of his/her visit to the fairground for later use [Ko07b]. In a typical housing fair a large number of different types of buildings and their décor are on display for a given period of time. Typically, a visitor spends a day at the crowded fairground under a hectic schedule, visiting possibly dozens of houses and exhibition booths. Most of the offered information is made available as paper brochures, resulting in a pile of paper being carried home for later use. The housing fair also typically comes with an accompanying website containing online information about the houses and other things. Further, many visitors are equipped with notebooks and a personal digital camera for taking photos of interesting objects.

We can identify many problems in the conventional setup of a housing fair. Firstly, it does not support the user in collecting, from a range of different sources (e.g., brochures, the web and personal digital cameras), the specific detailed information of an object that is of personal interest to the user in a manner which would incorporate automatic hyperlinking between different sources. This is a very acute problem to be solved, as the total amount of information available at a housing fair is immense, but only a fraction of it may be relevant to a particular user. Secondly, the big pile of brochures and miscellaneous photos does not support efficient retrieval of relevant information at a later date, when that information would be needed in designing a kitchen renovation, for example. Thirdly, the current setup does not support efficient sharing of information with other users, for example with friends who might also be planning to renovate their kitchens. Fourthly, the many month long marketing period before the fair requires that all printed brochures are produced well ahead of the houses and their interior/exterior being finalized. This means that computer generated models are used in brochures instead of actual photos, or that illustrative photos are omitted altogether.

The MFD was designed to allow a housing fair visitor to capture the relevant information and to support more timely dissemination of information.The MFD combined a mobile phone and a desktop PC into a novel hybrid interface for collecting, storing and sharing information. The phone application was used for taking context-aware notes such as visual codes, photos, dictations and text. The notes were uploaded onto a website where they were presented in a contextual order for browsing, organizing and sharing with friends to be used on a later occasion. The website also automatically linked user-made notes with ready-made content packages of houses and related online resources.



Location:
Wienerberger

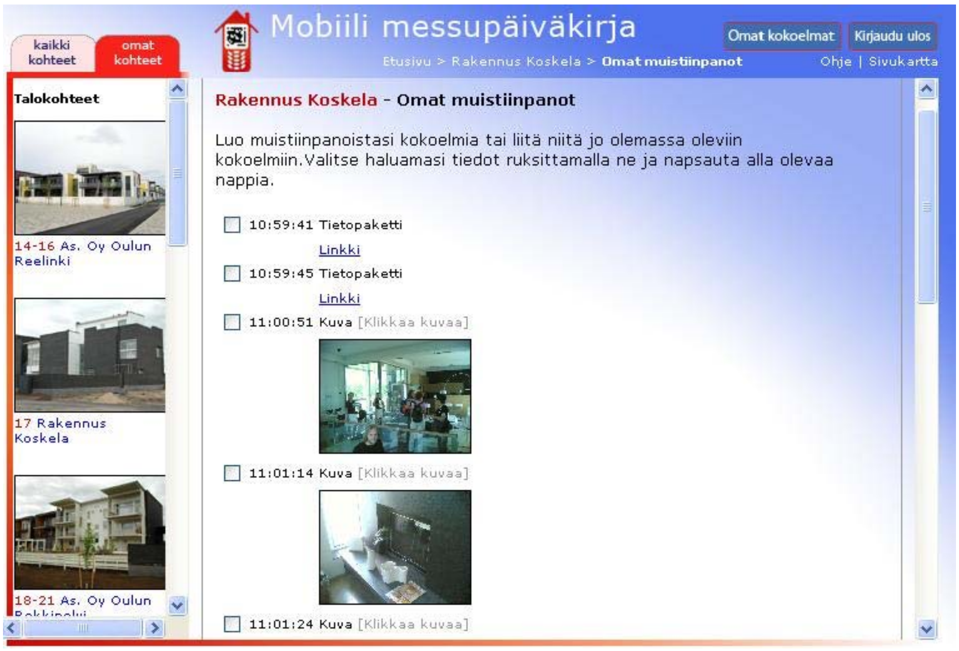
Picts: 1 / 2
Dicts: 0 / 1
Texts: 1 / 2
Codes: 0 / 2
Locations: 4

New location:
Villa Laivakello
Added note:
Laattaratkaisut

Figure 6. Screenshots of the UI of the MFD phone application: (a) main view; (b) adding a note; (c) a visual code is read; (d) a 2nd level visual code has been read and location is updated.

When starting the phone application for the first time, the user was required to type in a phone number or an email address or both so that the user could be delivered the user ID and password needed for accessing the website. After typing in the contact information, the user was presented with the main view showing the number of entries made at the current location and the total number of entries (Fig. 6(a)). The phone application had dedicated user interfaces for reading visual codes, taking photos, making dictations and writing text memos (Fig. 6(b)). The phone application also facilitated browsing and deleting the notes.

When a note was taken, it was associated with related contextual metadata comprising of the current location and a timestamp. The current location was indicated by reading a 2-D visual code of known locations with the phone's camera (Fig. 6(c)). Upon reading a visual code the user location was automatically updated (Fig. 6(d)). The MFD employed a nested two-level hierarchy of visual codes so that a 1st level code designated a house or an outdoor object such as a piece of art. The 2nd level code associated with a 1st level code of a house designated objects inside the house, for example a room or a piece of furniture. The 1st level codes were placed at the entrances, which were supposed to be read upon entering the houses. If the user forgot to read the 1st level code at the entrance, but then read a 2nd level code indoor, the phone application automatically recognized the corresponding 1st level code associated with the 2nd level code and updated the location accordingly. Furthermore, the user could also change the location by selecting it manually from the list of 1st level codes. The phone application was configurable so that an XML file specifying all visual codes and their UI labels was automatically downloaded when the application was started for the first time. The notes and their metadata were uploaded to a server at the end of the visit to the housing fair.



Rakennus Koskela - Omat muistiinpanot

Luo muistiinpanoistasi kokoelmia tai liitä niitä jo olemassa oleviin kokoelmiin.Valitse haluamasi tiedot ruksittamalla ne ja napsauta alla olevaa nappia.

□ 10:59:41 Tietopaketti
Linkki

□ 10:59:45 Tietopaketti
Linkki

□ 11:00:51 Kuva [Klikkaa kuvaa]

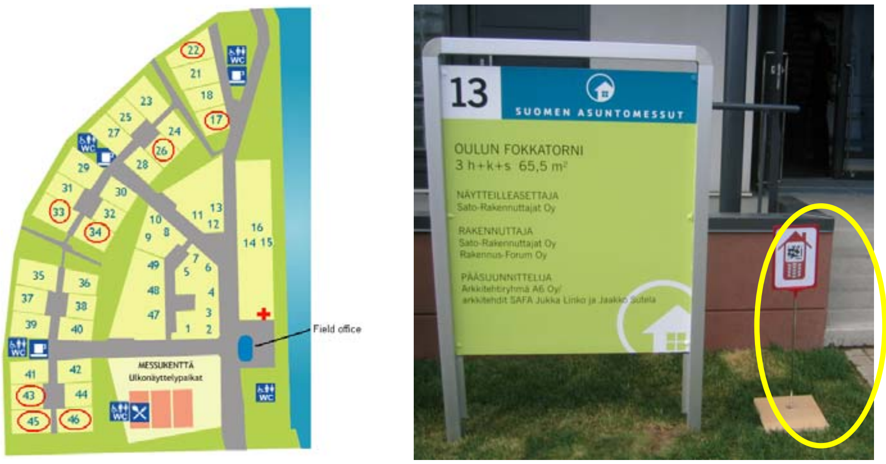
□ 11:01:14 Kuva [Klikkaa kuvaa]

□ 11:01:24 Kuva [Klikkaa kuvaa]

Figure 7. The MFD website presented notes in contextual order with hyperlinks to related information.

The MFD website facilitated fluent browsing, organizing and sharing of the notes taken with the mobile phone. The notes were presented in contextual order so that the objects corresponding to the 1st level codes read by the mobile phone (i.e., the houses and outdoor objects visited) were visualized in the left-hand frame with the title and thumbnail image in chronological order (Fig. 7). One of them was designated as the current object of interest, for which the right-hand frame then showed all notes taken in chronological order. The website also automatically augmented the notes of a particular object with hyperlinks to the online content package provided by us for that object and to other related online resources such as the website provided by the fair organizers. The user could rearrange the notes into any order that the user saw fit and deleting any needless notes. Furthermore, the user could compose personal collections of notes augmented with textual annotations. The user could also download the diary from the website to a local file as a ZIP-package.

The MFD was empirically evaluated in a large-scale field trial at the national Finnish Housing Fair in July-August 2005. The annual fair is the most prominent activity undertaken by The Finnish Housing Fair Co-operative Organization, a non-profit association, whose mission is to improve the quality of construction in Finland. Different types of buildings and their décor were on display for a period of 30 days, after which the actual occupants of the houses moved in. The fairground consisted of over fifty houses and apartment buildings constructed on an area of 13 hectares (Fig. 8(a)). We were granted permission to place a 1st level visual code at the entrance of each house and apartment building (Fig. 8(b)) and on outdoor art pieces. Furthermore, a limited number of 2nd level visual codes were allowed (3-8 per house, 43 in total) inside the eight houses circled in Fig. 8(a).



(a) (b)

Figure 8. (a) A map of the fairground, where the circled houses were equipped with 2nd level visual codes indoors; (b) A 1st level visual code placed at an entrance of an apartment building.

The field trial was coordinated from an office located near the main entrance to the fairground. If people had a compatible phone, they could acquire the phone application themselves by sending an SMS to a given phone number, downloading the application from the web, or visiting the field office. If their phone was not compatible they could loan one, without charge, for the duration of their visit at the fairground. Each test user was rewarded with ice cream upon returning to the desk and uploading their diary to the website. Qualitative data was collected through an online questionnaire available at the website and by interviewing ten randomly selected users. The test users were motivated to fill in the questionnaire with a raffle. Quantitative data comprised of the statistics of the notes and their metadata, together with the log of the users' actions on the website.

During the one-month trial 349 test users uploaded their diary to the website and 169 (48.4%) of them filled in the online questionnaire. Our small sample had similar profile to that of the whole Housing Fair clientale of 121110. The 349 diaries contained 27258 notes. 302 (86.5%) of the 349 diaries contained more than 10 notes, which can be regarded as actual usage of the MFD. An individual user took about 78 notes on average, over half of the users took more than 50 notes and 25 (7.2%) heavy users took more than 200 notes. The 27258 notes were of different types as follows: 24044 photos, 1869 indoor objects (2nd level visual codes), 831 dictations and 514 text notes. The large number of photos is simply explained by the fact that a photo has clearly the highest information value with respect to the amount of work needed. The low number of text notes is explained by the difficulty of typing notes with the 12-key keypad of a mobile phone. The low amount of dictations is at least partially explained by the challenging social setting, for it is understandably difficult to dictate given the crowd and ambient noise at the housing fair.

Here I present just some key observations from the data while the reader should see [Ko07b] for in-depth analysis. When we conducted a statistical test between the numbers of different notes and selected respondent attributes (gender, age, prior experience in using smart phones, prior experience in using the Internet), the only statistically significant finding was that females took more text notes. This means that, for example, the lack of any prior experience in using smart phones had no effect on the usage of the phone application, which testifies to a successful design in terms of learnability and efficiency. When we tested for any statistically significant differences in the questionnaire data with respect to respondent demographics, the following findings were made. Female users thought that the service was easier to learn (average score of 4.43 on 5-point Likert scale) and more self-evident to use (4.38) than male users (4.11 / 4.08). Female users were also more confident that the service was useful (4.60) than male users (4.35). Female users also felt that they gained more advantage from using the service (4.32 vs. 3.97). Quite surprisingly, the users with no prior experience of smart phones felt that the MFD was easy to learn (4.61) than those with prior experience (4.36). They also felt that the service was more self-evident to use (4.36 vs 4.10). As a whole, the data demonstrated excellent user satisfaction and identified unconventional enthusiastic user groups for a mobile service such as middle-aged women.

The MFD concept was commercialized by the startup company founded by the researchers of the project and is now available as a commercial product Entre Exhibitor.

7 panOULU Luotsi

panOULU Luotsi (~ Pilot in English, [pa09b]) is a location-based information mash-up provided for the users of a large municipal wireless network in the City of Oulu in northern Finland [Ku08]. The panOULU (public access network Oulu) network is provided by a consortium of five public organizations and four ISPs [Oj08][pa09]. As of now, the network totals about 1050 WLAN (IEEE 802.11a/b/g) access points, which provide indoor and outdoor coverage in locations deemed relevant for public access. The city center and its immediate surroundings are blanketed with a large outdoor WLAN mesh network, but otherwise the coverage is provided in hotspot manner. In its coverage area panOULU provides open (no login/authentication/registration) and free (no payment) wireless internet access to the general public, as long as you have a WLAN-equipped device. In September 2008, 15127 devices used the network, totaling 370000 sessions and 13.9 million online minutes. Up to 40% of the users in a given month are visitors, as determined from the usage patterns and by detecting devices that had not been seen in the network before. The proportion of multi-mode mobile handsets equipped with WLAN radios has been growing steadily so that they make up ~25% of the devices today. The usage of the network is still very much nomadic, not mobile, as only ~5% of sessions can be considered mobile.

The motivation behind Luotsi was to provide one obvious access point to useful information, which was before fragmented across several web sites and presented in different formats. Thus, the user had to implicitly know what information s/he needed and where to get it from. In many cases the user had to resort to multiple services before acquiring the necessary information. Some providers had made efforts to gather more comprehensive data in their site, but the data was still rather limited and outdated at times due to the lack of administration. Further, while one site might offer a map on the location of the desired target, another might not, and the user would have to manually enter the address in a map service, thus leaving the service s/he was originally using. The motivation was equally supported by the increasing coverage and usage of the panOULU network. The over 15000 users represent a considerable population in a city of 130000 people. Further, large proportion of them are visitors, who need up-to-date information of the services, places and events in a foreign city. In addition to providing a single access point to topical and relevant local information, other design goals included re-using existing information feeds instead of reinventing them, no client application should be needed besides a web browser, and the information should always be up-to-date. The design goals could be achieved with a web mash-up, which builds on the aggregation of various information feeds and real-time positioning of the user.

The high-level software architecture of panOULU Luotsi is shown in Figure 9. To access the vast amount of distributed information and to bypass the problem of outdated data on different sites, all content into Luotsi is acquired in form of XML-based feeds such as, but not limited to RSS or ATOM. The varying presentation formats of the feeds are dealt by the XML aggregator illustrated as the DataMapper in Figure 9, which maps different heterogeneous information feeds into the Luotsi database without any changes to the application source code.

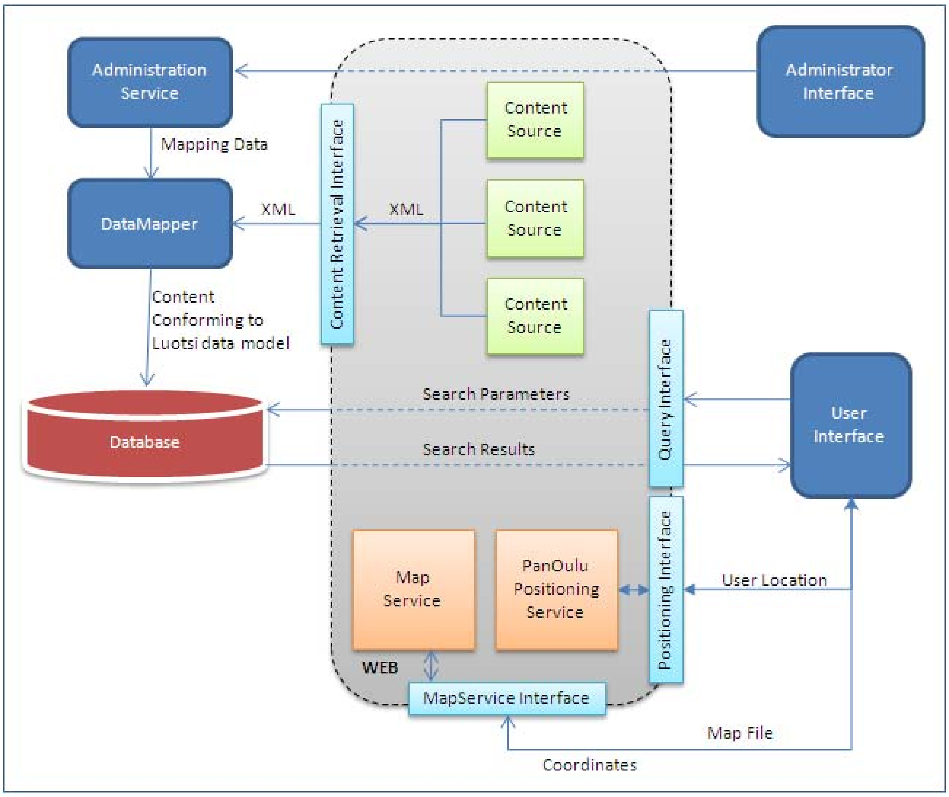


Figure 9. The high-level software architecture of panOULU Luotsi.

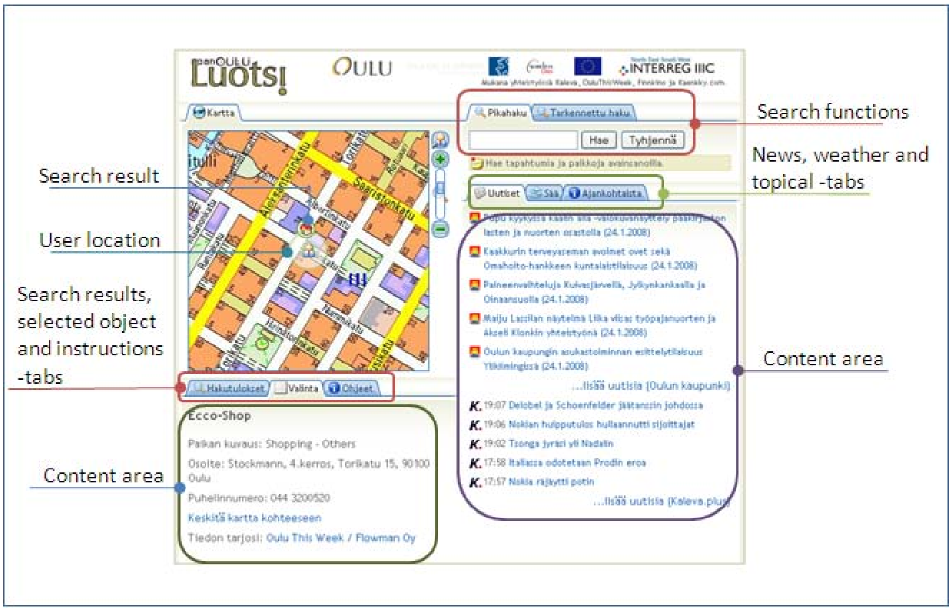
Luotsi has an administrator interface for registering any XML-based feed to be registered in the system. When registering a feed, the administrator receives a parsed presentation of how the feed is formulated (the XML tags). The administrator manually maps the attributes to corresponding fields in the Luotsi data model. For example, two feeds may describe a place with different XML structures such as <name></name> and <placeName></placeName>. Both are mapped to 'place_name' field in Luotsi data model. The mapping needs to be done only once for a feed unless, of course, the structure of the feed changes. The mapping rules are saved in a configuration file. The registered feeds are then automatically updated periodically by their provider, thus ensuring that all data is always up-to-date.

The location of the user is estimated as the location of the WLAN access point the user device is currently connected to. Typically, the user's actual location is within a 50 meter radius of location of the access point. This network-based AP ID positioning has several benefits over other methods such as GPS positioning: the user device needs no additional software or hardware, the first location estimate is obtained very quickly, and it works also indoors provided network coverage is available.

Currently, registered feeds include movie listings from Finnkino (the main movie provider in Finland), City of Oulu's event calendar and news feed, fast food restaurants from kaenkky.com (a local website listing and reviewing all fast food restaurants in Oulu region), service directory and local sights listings from ouluthisweek.net (local service provider producing both online and printed information services), and the latest news from Kaleva (the local main newspaper). Aggregated, these feeds provide a very comprehensive directory of services and events that both local people and visitors might need when moving around Oulu. Luotsi also provides a location aware weather report based on a cluster of micro weather stations installed around the city. The weather data provided by the stations is available as a public web service. Luotsi automatically determines the weather station nearest to the user's current location and displays the data in easy-to-read manner, thus adding another element to the mash-up.

The feeds are retrieved periodically by the Content Retriever, mapped by the DataMapper into Luotsi data model based on the rules in the configuration file and the mapped data is stored in the Database. The Database serves as a proxy (cache), thus making later queries much faster than if the data was always fetched from the individual feeds and mapped on the fly. The Database is a mySQL database with a table structure conforming to Luotsi data model. The data model is flexible to support the needs of varying feed structures. Fields not found from a feed are left blank, and cross-referencing is supported. This is useful if, for instance, an event-item does not have explicit location information (e.g. coordinates), but contains a logical name or street address, which can be mapped to the coordinates using the City of Oulu's geodetic service. Similarly, relevant tags of the emerging geoRSS format (e.g. <geo:lon> and <geo:lat>) could be mapped to a location in our data model, although the current content feeds do not use them.

The panOULU Luotsi web user interface is illustrated in Figure 10. It is implemented with the jQuery AJAX library using the model-view-controller design paradigm. The presentation layer is separated from the application logic stored in separate PHP files that are called from the UI. The layout is divided into two columns so that the left column shows the location of the user and objects of interest. The user's current location is presented by placing a 'guess circle' around the icon representing the user on the map, indicating that the user's actual location is within the guess circle. Object categories are presented with their specific distinct icons, e.g. a plate indicates a restaurant. The map file is retrieved by the MapServiceInterface component from the City of Oulu's online map service, which offers an open web inteface for retrieving maps. The map is initially centered at the user's location, but it can be freely moved in any direction after that. The maps outside the current view are fetched asynchronously in the background, so that when the user moves off the current map view, map loading times are minimized. The map also supports a 4-level zoom, allowing the user to zoom in or out at any point. Luotsi is primarily intended to be used via the panOULU network, which facilitates the AP ID positioning. However, Luotsi is a public web service, and thus can be used via any other access network, as well. If the positioning fails because the user is not connected to panOULU network, the user is asked to enter his/her current or desired location manually, e.g. as a street address or pointing a location on the map.



Ecco-Shop

Paikan kuvaus: Shopping - Others

Osoite: Stockmann, 4.kerros, Torikatu 15, 90100 Oulu

Puhelinnumero: 044 3200520

Keskitä kartta kohteeseen

Tiedon tarjosi: Oulu This Week / Flowman Oy

Pupu kyykyssä kaalin alla -valokuvanäyttely pääkirjaston lasten ja nuorten osastolla (24.1.2008)

Kaakkurin terveysaseman avoimet ovet sekä Omahoito-hankkeen kuntalaistilaisuus (24.1.2008)

Paineenvaihteluja Kuivasjärvellä, Jylkynkankaalla ja Oinaansuolla (24.1.2008)

Maiju Lassilan näytelmä Liika viisas työpajanuorten ja Akseli Klonkin yhteistyönä (24.1.2008)

Oulun kaupungin asukastoiminnan esittelytilaisuus Ylikiimingissä (24.1.2008)

...lisää uutisia (Oulun kaupunki)

K. 19:07 Delobel ja Schoenfelder jäätanssin johdossa

K. 19:06 Nokian huipputulos hullaannutti sijoittajat

K. 19:02 Tsonga jyräsi yli Nadalin

K. 17:58 Italiassa odotetaan Prodin eroa

K. 17:57 Nokia räjäytti potin

...lisää uutisia (Kaleva.plus)

Figure 10. The browsing interface of panOULU Luotsi.

Luotsi also employs automatic proximity-based search functionality. Once the user location has been established, the system automatically displays five nearest objects on the map. The objects found with the proximity-based search are hidden once the user makes a search for something else. Below the map is a window that can display different information based on the selected tab. Initially, the content area displays the ‘instructions’ tab with general information on the use of the service, but once the user searches for something or selects one of the automatically shown objects, the content changes to either a list of search results, or details on the selected object, respectively.

We have been logging the usage of Luotsi since its launch in October 2007. By September 2008, a total of 6658 sessions was recorded, averaging 555 sessions a month. About one third of the sessions were from devices connected to the panOULU network, two thirds from elsewhere in the Internet.

8 Conclusion

I conclude with a brief discussion on some important lessons that I have learned from the case studies and related research efforts. The lessons may also help to understand why commercially relevant context-aware applications remain sparse, most notably GPS navigators and few other location-based applications, despite the long-term enthusiasm for context-aware computing in the research community.

Design for maximum usefulness. With usefulness I refer to the combination of utility and usability [Ni94]. A system with high utility satisfies some concrete need(s) of the end user. Their identification calls for understanding the “situation of concern”, where things are not quite right and which could be resolved with the proposed system. We can then quantify our success in system design and implementation by measuring or predicting the usability of the system with various usability factors. The Mobile Fair Diary was a prime example of a service satisfying a concrete need of capturing the relevant information from a housing fair. Consequently, the test users expressed high subjective satisfaction despite the assorted design flaws.

Content is king. The design and implementation of useful services requires high quality content, which can be expensive and difficult to obtain. The same applies to contextual data, as well. If you have to produce the content yourself, then you have to allocate sufficient resources for that purpose. The TimeMachine Oulu is a good example of an application made possible by high quality historical data. Similarly, the Google Maps with its extensive map data and functional APIs have enabled thousands of web mash-ups.

Understand existing business practices and value networks. Our difficulties with mobile advertising were partly due to the concept being simply too far ahead of the state-of-the-art in business-to-customer advertising. At the same time we failed to motivate and educate the advertisers to use the mobile advertising channel in the “correct” manner. We had similar difficulties with the Mobile Fair Diary, which we offered to the housing fair as plain “technology push” without any consideration for a potential business model. Consequently, the many commercial actors of the existing value network involved in the deployment were not motivated, which led to delays and problems in the design and implementation of the service and the execution of the field trial. Finally, the research community seems often happily forgetting that a long-term production deployment of a service calls for a viable business model. A fundamental shortcoming in all presented case studies was that the user did not have to pay anything for using the services, though most of them can be envisioned to be provided as free services.

Do not forget maintenance. The responsibilities do not end with the first deployment of a service. On the contrary, the subsequent maintenance can require lots of manpower and other resources, depending on the scalability, fault tolerance and maintainability of the service infrastructure. Maintenance of context data can be difficult, as well. For example, even if you manage to somehow acquire high quality user profile data, it is well known that users are generally not motivated to keep the data up to date. Based on our own experience research organizations and projects are traditionally not well equipped to do maintenance on long-term basis. For example, we simply ran out of resources to keep the WLAN positioning system based on signal strength fingerprints functional. Similarly, maintaining dedicated server resources and components for long-term “production” deployments has proven challenging, as the resources allocated per project basis disappear when the project runs out. Thus, take maintenance into account in designing your deployment and resourcing.

Share your infrastructure. The research community values novel contributions, not high quality engineering which is a prerequisite for successful experimental field deployments. Consequently, there is lots of needless “reinventing the wheel”, where a particular application problem is tackled the umpteenth time with a slightly incremental “novel” approach justifying a scientific publication, but not much else. This could be alleviated by researchers sharing their implementations, assuming they would be of sufficiently high quality for that purpose and no IPR issues would prevent sharing. The panOULU network is a prime example of how sharing your infrastructure can benefit the whole community. [RS05]

Empirical evaluation in the field pays off. I am a big fan of empirical user evaluation of prototype implementations in true environment of use by genuine end users. Despite its methodological pitfalls and practical challenges it remains the only way to involve all factors affecting overall user acceptance, most importantly the actual usage context. Field trials are expensive and time consuming for many reasons. In terms of engineering cost there can be a big gap between a one-time steering group demo and months-long deployment exposed to the diverse general public. Setting up a field trial in public space may require dealing with lots of bureaucracy and logistic challenges. A field trial inevitably requires lots of manpower for different tasks such as help desk, content production, technical support and maintenance. A field trial may expose you to public scrutiny, which is not always a pleasant experience. A large-scale field trial assumes using off-the-shelf mass market technology, thus approving its limitations in comparison to more advanced research equipment. Further, involving the general public as test users can be very difficult, as people are simply too busy with their real lives and reluctant to install your research software in their mobile phones. However, if you despite all these warnings wish to engage in a field trial, the following recommendations may prove useful. First establish your theoretical framework: what is your situation of concern, what are your objectives, what are your hypotheses, what is being evaluated and how, what data is needed for the evaluation, what are your evaluation criteria and their theoretical foundations? Think carefully about your service offering: it is typically much more difficult to sell a large service portfolio than an individual well-focused service. When designing the field trial, plan well ahead and prepare for the unexpected. Allocate sufficient time for the integration testing of your deployment before the start of the field trial. Remember to budget ample resources for PR, help desk, technical support and maintenance during the trial. Have a proper PR strategy to recruit people as test users, to manage the expectations of the general public, media and other stakeholders, and harden your skin for public scrutiny. Make sure that you obtain the data needed for your theoretical framework and store it in a safe place. Allocate ample time for analysing the data after the field trial, for otherwise your hard work can go waste, which would be a real shame. Finally, try to decide before the field trial whether your service(s) are going to be available after the conclusion of the field trial or not. If some remain, then you have to allocate resources for their maintenance. The real proof of a successful trial deployment is when the users continue to work with a system after the trial is officially over [Wa06].

References

[Aa04] Aalto, L; Göthlin, N; Korhonen, J; Ojala, T: Bluetooth and WAP Push based location-aware mobile advertising system. Proc. Second International Conference on Mobile Systems, Applications and Services, Boston, MA, 49-58, 2004.

[ARO03] Aittola, M; Ryhänen, T; Ojala, T: SmartLibrary - Location-aware mobile library service. Proc. Fifth International Symposium on Human Computer Interaction with Mobile Devices and Services, Udine, Italy, 411-416, 2003.

[Ai04] Aittola, M; Parhi, P; Vieruaho, M; Ojala, T: Comparison of mobile and fixed use of SmartLibrary. Proc. 6th International Conference on Human Computer Interaction with Mobile Devices and Services, Glasgow, Scotland, 383-387, 2004.

[BDR07] Baldauff, M; Dustdar, S; Rosenberg, F: A survey on context-aware systems. International Journal of Ad Hoc and Ubiquitous Computing 2(4):263-277, 2007.

[Bo07] Bolchini, C; Curino, C; Quintarelli, E; Schreiber, F; Tanca, L: A data-oriented survey of context models. SIGMOD Record 36(4):19-26, 2007.

[Do04] Dourish, P: What we talk about when we talk about context. Personal and Ubiquitous Computing 8(1):19-30, 2004.

[KS03] Kjeldskov, J; Stage, J: New techniques for usability evaluation of mobile systems. International Journal of Human-Computer Studies 60(5-6): 599-620, 2003.

[Ko07a] Komulainen, H; Mainela, T; Tähtinen, J; Ulkuniemi, P: Retailers' different value perceptions of mobile advertising service. International Journal of Service Industry Management 18(4):368-393, 2007.

[Ko07b] Korhonen, J; Ojala, T; Ristola, A; Kesti, M; Kilpelänaho, V; Koskinen, M; Viippola, E: Mobile Fair Diary - Hybrid interface for taking, browsing and sharing context-aware notes. Personal and Ubiquitous Computing 11(7):577-589, 2007.

[Ku08] Kukka, H; Ojala, T; Tiensyrjä, J; Mikkonen, T: panOULU Luotsi: A location based information mash-up with XML aggregator and WiFi positioning. Proc. 7th International ACM Conference on Mobile and Ubiquitous Multimedia, Umeå, Sweden, 2008.

[Ni94] Nielsen, J: Usability Engineering. Morgan Kaufmann, San Francisco, 1994.

[OSW07] Oh, Y; Schmidt, A; Woo, W: Designing, developing, and evaluating context-aware systems. Proc. 2007 International Conference on Multimedia and Ubiquitous Engineering, Seoul, Korea, 1158-1163, 2007.

[Oj03] Ojala, T; Korhonen, J; Aittola, M; Ollila, M; Koivumäki, T; Tähtinen, J; Karjaluoto, H: SmartRotuaari - Context-aware mobile multimedia services. Proc. 2nd International Conference on Mobile and Ubiquitous Multimedia, Norrköping, Sweden, 9-18, 2003.

[Oj08] Ojala, T; Hakanen, T; Salmi, O; Kenttälä, M; Tiensyrjä, J: Supporting session and AP mobility in a large multi-provider multi-vendor municipal WiFi network. Proc. Third International Conference on Access Networks, Las Vegas, NV, 2008.

[pa09a] panOULU network. http://www.panoulu.net.

[pa09b] panOULU Luotsi, http://luotsi.panoulu.net.

[POO03] Peltonen, J; Ollila, M; Ojala, T: TimeMachine Oulu - Dynamic creation of cultural-spatio-temporal models as a mobile service. Proc. Fifth International Symposium on Human Computer Interaction with Mobile Devices and Services, Udine, Italy, 342-346, 2003.

[RS05] Sharp, R; Rehman, K: The 2005 UbiApp Workshop: What makes good application-led research? IEEE Pervasive Computing 4(3):80-82, 2005.

[Ro03] Rotuaari – Context-Aware Mobile Multimedia Services research project, 2003-2006, http://www.rotuaari.net.

[Wa06] Want, R: Build what you use. IEEE Pervasive Computing 5(3):2-3, 2006.

[Wi05] Wireless Cities project, 2005-2007, http://www.wirelesscities.org/.

Content-Based Image Retrieval Systems - Reviewing and Benchmarking

Harald Kosch
Chair of Distributed Information Systems,
University Passau, Germany
Paul Maier
Chair for Image Understanding and Knowledge-Based Systems,
Technical University Munich, Germany

Abstract

The last detailed review of Content-based Image Retrieval Systems (CBIRS) is that from Veltkamp and Tanase [VT02], updated in 2002. Since then, many new systems emerged, other systems were improved, but many systems are no longer supported. This paper reconsiders the systems described by Veltkamp and Tanase and proposes in addition for a selection of existing CBIRS a quantitative comparison.

For this purpose we developed a benchmarking system for CBIRS based on how accurate they could match ideal (ground truth) results. As measure we introduced the $\widetilde{Rank}_{WRN}$, which we developed based on the *Normalized Avarage Rank*. Our measure allows fair and accurate comparisons (due to multiple similarity values) of different CBIRS with respect to different queries. None of the other benchmarks allow this comparison.

The choice for a system to be compared quantitatively was motivated by availability of the source or runtime code and the liveness (i.e., active development on the CBIRS). The benchmarked systems were SIMBA, SIMPLIcity, PictureFinder, ImageFinder, Caliph&Emir, VIPER/GIFT and Oracle Intermedia. Results show that VIPER/GIFT performs best in the settings of our benchmark, Caliph&Emir is second and SIMBA third. The default parameter settings of the systems tend to show the best results. MPEG-7 descriptors, which are used in Caliph&Emir, show a good performance in the benchmarking.

1 Introduction

Research in Content-Based Image Retrieval is devoted to develop techniques and methods to fulfil image centered information needs of users and manage large amounts of image data. It is nowadays an even more vivid research area than when Veltkamp and Tanase provided an overview of the CBIR landscape [VT02], which showed the state-of-the-art at that time. Since then there has been much development in multiple directions. The question arises how those projects evolved: which have been concluded, which canceled, which are still active or have spawned new projects. Yet another question the report by Veltkamp and Tanase didn't answer is how the systems perform. This question alludes to the broad topic of benchmarking information retrieval systems in general and content-based image retrieval systems specifically. A number of research papers have addressed this topic ([MMS$^+$01b, MMW06, MMS01a]), focusing on how to establish standards for benchmarking in this area.

We tried to follow the paths of the systems described in [VT02] and reviewed new developments. Assuming a bird's eye perspective, we found that on the one hand many of the older projects were finished or have not been further developed. For many of these efforts it remains unclear what further developments have occurred, if any. On the other hand there are a lot of new projects. Among them we found a tendency to implement standards like MPEG-7 as well as a simplification of user interfaces. Of the recent systems none expects the user to provide image features as a query, for example. Most of the efforts try to address the problem of CBIR in a general fashion, they don't focus on specific domains. The projects are often spin-offs or specialisations of other, general CBIR systems or frameworks. Since presenting all of our findings would exceed the scope of this article we just present an overview of the systems we see as "live": active and ongoing projects, which are either under active development or research, or are being used in products. Furthermore, the availability of the source or runtime code for testing was an obvious prerequisite. The complete report can be requested from Paul Maier[1].

In this article we focus on the mentioned systems and their development as well as several new systems and on benchmarking a small subset of these systems. We don't review the development of CBIR as a whole. In their recent work Datta et al. ([DJLW08]) provide a broad overview of recent trends, ideas and influences in CBIR.

The rest of the article is organized as follows: section 2 provides the overview of "live" projects. Also, it identifies the systems chosen for the benchmark. Section 3 explains how the systems were benchmarked, particularly our new measure $\widetilde{Rank}_{WRN}$ is introduced. Section 4 discusses the benchmark results and section 5 finally sums up and provides an outlook for future work.

2 Reviewed Systems

Table 1 shows an overview of all "live" systems. The leftmost column lists the systems, sorted alphabetically. The second column holds references for the systems, then follow the columns showing the features: support of texture, color, shape, search by keywords, interactive relevance feedback, MPEG-7 descriptors [DK08] (for example *DominantColor*), whether they are specifically designed to be used as web search engine, whether they use classification techniques or segment images into regions.

The column *designed for web* is primarily a matter of scalability: a web search engine must be able to handle millions of images, performing searches on vast databases within seconds. But it also means that some kind of image accumulation mechanism, such as a web crawler, must be provided.

The column *classification* specifies if a system uses some kind of classification mechanism, clustering images and assigning them to classes. The last column, *image regions*, indicates whether a system applies some kind of image segmentation. This is not the same as *shape*: image regions might be used to identify shapes in an image, but they could also simply be a means of structuring the feature extraction (see for example VIPER in section 4.5).

One can recognize that keyword search is very often supported as well as relevance feedback. There seems to be a consensus that image retrieval in general cannot solely rely on content-based search but needs to be combined with text search and user interaction.

MPEG-7 image descriptors are still seldom used, but especially new systems or new versions of systems tend to incorporate these features. Many of the systems are designed to work on large image sets (> 100 000 images in the database), but they are rarely designed specifically as a web search engine. Currently, only Cortina seems to be designed that way.

Seven CBIR systems have been chosen to for our benchmark:

> SIMBA, SIMPLIcity, PictureFinder, ImageFinder, Caliph&Emir, VIPER/GIFT, Oracle Intermedia

The systems were chosen following three criteria: (1) availability and setup: we needed to obtain and setup the CBIRSs in order to benchmark them within our own environment. Setup failed with some

[1] Contact by email: maierpa@in.tum.de

System	reference	texture	color	shape	keywords	interactive relevance feedback	MPEG-7 support	designed for web	classification	image regions
Behold	[YHR06]	*	*		*					*
Caliph&Emir	[LBK03]	*	*		*		*			
Cortina	[QMTM04]	*	*		*	*	*	*	*	
FIRE	[DKN04]	*	*		*	*				*
ImageFinder	[att07]	?	?	?					?	*
LTU ImageSeeker	[LTU07]	*	*	*	?	?			*	*
MUVIS	[GK04]	*	*							
Oracle Intermedia	[oim07]	*	*	*	*[2]					
PictureFinder	[HMH05]	*	*	*	*	*				*
PicSOM	[KLLO00]	*	*	*		*	*			*
QBIC	[FSN$^+$01]	*	*	*	*	*				
Quicklook	[CGS01]	*	*	*	*	*				
RETIN	[CGPF07, FCPF01]	*	*			*				
SIMBA	[Sig02]	*[3]	*							
SIMPLIcity	[LWW00, WLW01]	*	*	*					*	*
SMURF	[VV99]		*	*						
VIPER/GIFT	[Mül02]	*	*			*				*

Table 1: Overview of all live CBIR systems. For each CBIRS (rows), its features are listed (columns). A star ('*') in a column marks the support for a feature and a question mark ('?') if the support is unknown. If a system doesn't support the feature, the table entry is left empty.

[2] Keyword search in Oracle Intermedia depends on the design of the database being used to store the images. Generally, multimedia data can be arbitrarily combined with e.g. text data and searched with corresponding SQL queries.

[3] Whether a feature in SIMBA is actually a color or texture feature depends on the kernel function used.

obtained systems, which were therefore excluded; (2) visual similarity only: in our benchmark we focus on visual similarity search. Therefore, systems were required to only use visual similarity properties for their search. We excluded additional features, like text search or semantic analysis (e.g., automatic annotation); (3) query by example: this method was required to be supported by all CBIRSs.

There are a number of interesting systems which comply with the mentioned requirements, but have not been tested for other reasons. There is for example the Blobworld [CTB$^+$99] system, which turned out too time-consuming to setup. Three other interesting systems which have not been tested mainly due to time constraints are PicSOM [KLLO00], Quicklook^2 [CGS01] and FIRE [DKN04][4].

3 Benchmark Structure and Measurement

A new benchmark was developed to test and compare the chosen systems. This section outlines the aim of the benchmark, describes its overall structure, states which images where used in the database and how a ground truth was build from those images. Finally, our performance measure is introduced, as well as two new methods to compare CBIR systems based on these measures.

3.1 Benchmark Methods

In recent years efforts were made to analyse the possibilities for benchmarking CBIR systems and to propose standard methods and data sets ([MMS$^+$01b, MMW06, MMS01a, ben]). Since 2003 the *cross language image retrieval track* ImageCLEF ([ima05]) is run every year as part of the *Cross Language Evaluation Forum*. The focus of this track is on image retrieval from multilingual documents. A review of evaluation strategies for CBIR is presented in [DJLW08]. Still there seems to be no widely accepted standard for CBIR benchmarking, neither a standard set of performance measures nor a standard data set of images for the benchmarks.

A CBIR system has many aspects which could be subjected to a benchmark: the retrieval performance of course, its capability to handle large and very large image sets and the indexing mechanism to name just a few. In this work we focus on the retrieval performance aspect.

Aim Of The Benchmark We compared CBIRSs to each other based on their *accuracy*, i.e. their ability to produce results which accurately match ideal results defined by a ground truth. We assume a CBIRS does a good job if it finds images which the user himself would choose. No other queues were used, like e.g. relevance feedback. We used *query by example* only, no query by keyword, sketch or others. Also, no optimization for the image sets was done. The intent was to test the systems without providing them with prior knowledge of the image database. We concentrated on visual similarity only, the benchmark does not address any other aspects, e.g. resource usage and scalability. Those aspects, while of at least equal importance, where left out mainly due to time constraints.

The basic requirement of a CBIRS, is to fulfil a user's information need: finding useful information in form of images. We assume a CBIRS fulfils an information need if it finds *visually similar* images. Accordingly, we define *similarity* of images as follows

Definition: Image Similarity Two or more images are taken to be *similar* if spontaneously viewed as visually similar by human judges.

Accuracy in this work is simply defined through the ideal order imposed on result images by similarity: Images more similar to a query are appear before less similar images in the result. If the actual

[4] The authors of PicSOM and Quicklook^2 both offered to run the tests within their environments and send back the results. FIRE is freely available as open source software.

ordering in the result violates this ideal ordering accuracy decreases. In the following we characterize a system's performance by its accuracy.

3.2 Image Database

The images used for our benchmark are divided into domains of roughly similar images. The intention was to have a fairly representative sample of images, roughly grouped by some sort of image type. The following list explains which image domain represents which common type of image (and lists the query images):

Band: [5] Images of a street band. All the images are grey-scale. They represent black and white photographs and typical images showing a group of people.

Blumen (flowers): [6] Photographs of single flowers, taken in a natural environment. Typical examples are photographs of single (natural) objects with nature as background (meadows, rocks etc.).

Kirschen (cherry): [7] Photographs of blossoming cherry trees, taken in some kind of park. This is representative for trees, bushes, colorful images, mixed with artificial structures like pathways and buildings.

Textur (texture): [8] These are texture images. That means, they are photographs of various surface structures and intended for use as texture in computer graphics. Typical examples are texture-only images, like photographs of tissue, cloth, wall paint and the like.

Urlaub (vacation): [9] Photographs from someones vacation. Typical examples are vacation/leisure time photographs.

Verfälschung-Wind (distortion-wind): A graphical filter (creating a motion blurr which looks like wind) was applied to a source image (which is used as query image), creating a sequence of images with increasing filter strength. This domain will also be referred to as *Verfälschung* for short.

Each domain additionally contains several images from all other domains to create a "background noise".

Image Sources

The images have been taken from several sources which provided them for free use. The sources are indicated as footnotes in the above list except for *Verfälschung*. The images of this domain are altered versions of an image which is distributed together with Caliph&Emir [LBK03].

The images from the *University of Washington* and *Benchathlon.net* seem to be public domain, no licence is given. *Benchathlon.net* clearly states that up-loaders should ensure that images are not "copyright encumbered". The terms for images from *ImageAfter* allow free use except for redistributing them through an online resource like *ImageAfter*[10].

[5] `http://www.benchathlon.net/img/done/512x768/0175_3301/3787/`

[6] `http://www.benchathlon.net/img/todo/PCD1202/`

[7] `http://www.cs.washington.edu/research/imagedatabase/groundtruth/cherries/`

[8] `http://www.imageafter.com`

[9] `http://www.cs.washington.edu/research/imagedatabase/groundtruth/cannonbeach/`

[10] For the licence see the web page at `http://www.imageafter.com`

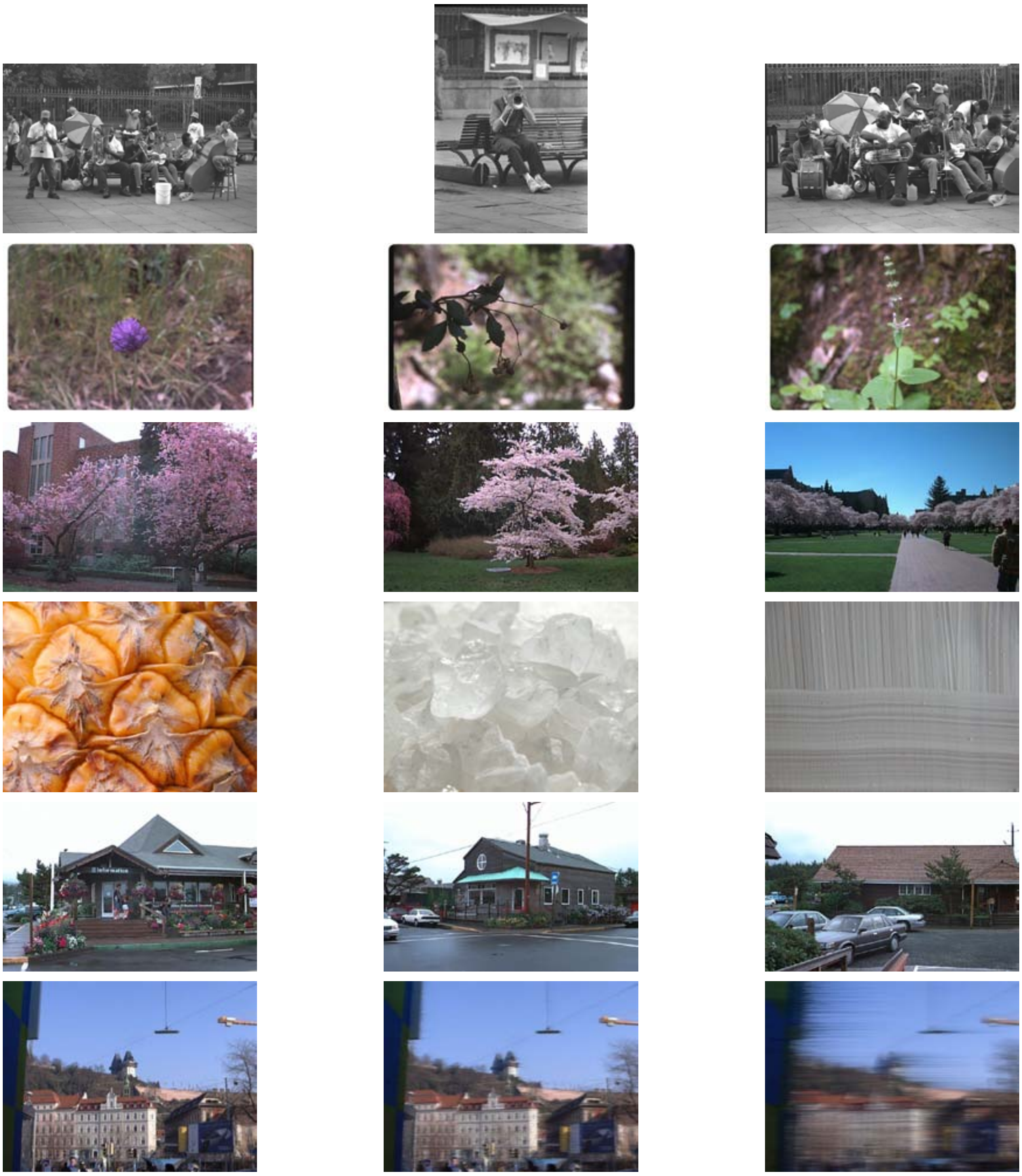


Figure 1: Example images from the domains. Each row shows 3 images from the same domain. From top to bottom: Band, Blumen, Kirschen, Textur, Urlaub, Verfälschung. For the domain Verfälschung the original image is shown left most, then two images with increasing distortion to the right.

3.3 The Benchmark

The CBIRS are benchmarked by running and evaluating a set of well defined queries on them. A query is simply an image for the search engines to look for similar images. The search is done within the images of the query domain. The reason is that building the ground truth was manageable this way, since only the images within a domain needed to be evaluated for similarity. Query images were chosen such that a reasonable number of similar results was possible.

Each CBIRS has a number of parameters which allow to adapt and modify it in various ways. Changes in those settings can improve or worsen the result for certain or even all queries. Consequently, we benchmarked different parameter settings, or *psets*, for each CBIRS. For example, for VIPER/GIFT one pset uses texture features only, another is restricted to color features. In the following, we refer to a CBIRS with a specific *pset* as *system.* We ran the benchmark on 23 psets overall. From these 20 were fully evaluated; the 3 psets of SIMPLIcity had to be excluded due the missing-images-problem explained below.

Psets are named with letters of the alphabet starting with 'b'[11]. For example, Caliph & Emir has three psets, b,c,d. Across different CBIRS psets have nothing in common, i.e., two psets c for two different CBIRS only share the letter. The only exception is pset b, which for all CBIRS is their standard setup. In the results section, the psets for each CBIRS will be described in detail.

3.4 Ground Truth

The ground truth for this benchmark has been established through a survey among 5 randomly chosen students. The students had no prior knowledge of CBIR techniques and were not familiar with the image sets (other than one could expect from fairly normal photographs). 14 queries have been defined, for each the students had to compare the query image with all images from the query's domain. They were asked to judge the similarity from 0 (no similarity) up to 4 (practically equal images)[12]. The ground truth thus consists of tuples of query image and result image which are mapped to a similarity value. This value is the mean of all similarity values from the different survey results and thus is an element of $[0,4]$.

3.5 Performance Measures

The information in this section has mainly been excerpted from [MMS$^+$01b, MMW06].

From the domain of text-based information retrieval a number of methods are known which can also be applied to content-based image retrieval. One of the simplest, but most time-consuming methods is *before-after user comparison*: A user chooses the best result from a set of different results to a predefined query. 'Best' in this context means the most relevant result.

Then, a wide variety of single valued measures exist. *Rank of the best match* measures whether the most relevant image is under the first 50 or the first 500 images retrieved. In [MMS$^+$01b] the *Normalized Average Rank* (NAR) is proposed as a performance measure for CBIR systems. We use it as basis for our own measure, which will be explained later. Popular and a standard in text retrieval are the measures *precision* and *recall*:

[11] A spread sheet has originally been used to describe the parameter sets, and since the first column 'a' was taken all sets start with 'b'.

[12] The motivation behind the 5 similarity values is simply the assumption that human judges with no prior knowledge of CBIR or familiarity with the images can fairly easy differentiate 5 (or maybe 6 or 7) similarity levels, but not more. This bears some similarity to a Likert scale, although it was not the aim to implement it.

$$precision = \frac{\text{No. relevant images retrieved}}{\text{Total No. images retrieved}},$$
$$recall = \frac{\text{No. relevant images retrieved}}{\text{Total No. relevant images in the collection}}$$

They are good indicators of the system performance, but they have to be used in conjunction, e.g. precision at 0.5 recall or the like. An entirely different method is *target testing*, where a user is given a target image to find. During the search, the number of images is recorded which the user has to examine before finding the target. The *binary preference measure* keeps track of how often non-relevant images are misplaced, i.e. before relevant images. Further single valued measures are *error rate, retrieval efficiency* and *correct and incorrect detection*. For the sake of brevity these are only listed here, for detailed explanations the reader is referred to [MMS$^+$01b].

A third category of measures can be subsumed as graphical methods. They are graphs of two measures, the most popular being *precision vs. recall graphs*. They are also a standard measure in IR and can be readily interpreted by many researchers due to their popularity. Another kind of graphs are *precision/recall vs. number of images retrieved* graphs. The drawback of these two graphical methods is their dependency on the number of relevant images. *Retrieval accuracy vs. noise* graphs are used to show the change in retrieval performance as noise is added to the query image. A number of other graph methods exist, again the reader is referred to [MMS$^+$01b] for further reading.

3.6 Worst Normalized Rank

The afore mentioned measures all use binary similarity/relevance values, i.e. only the distinction relevant/not relevant or similar/not similar is made. In contrast, our measure uses multiple similarity values and thus allows fair and accurate comparisons of different CBIRSs with respect to different queries. None of the other benchmarks allow this comparison.

To account for five degrees of similarity we developed the new measure *Worst Normalized Rank* $\widetilde{Rank}_{WRN}$, based on the NAR measure [MMS$^+$01b]. NAR is defined as

$$\widetilde{Rank} = \frac{1}{NN_R}\left(\sum_{i=1}^{N_R} R_i - \frac{N_R(N_R-1)}{2}\right) = \frac{1}{NN_R}\left(\sum_{i=1}^{N_R} R_i - \sum_{i=1}^{N_R} i\right)$$

N: no. documents total

N_R: no. relevant documents total

R_i: i-th relevant document's rank in result list

For a given query result, a ranked list of images, $\widetilde{Rank}$ essentially computes a distance between the query result and the ideal result for that query. It does so by summing up the ranks R_i of the similar images and subtracting the sum of ranks of the ideal result, which is $\sum_{i=1}^{N_R} i$ [13]. Thus, the best possible measured value is always 0, while the upper bound for the worst value is 1. The measure was chosen because it expresses CBIRS performance (for a specific query result) in a single value and is fairly easy to adapt. In the following, the term NAR will be used in a general fashion when addressing properties which hold true for both the original measure and the modified one. When referring exactly to one of them, the expressions $\widetilde{Rank}$ for the original measure and $\widetilde{Rank}_{WRN}$ for the modified measure will be used.

As a first step $\widetilde{Rank}$ is adapted to account for multiple similarity values in the following way:

[13] since the ideal result is simply the list with all relevant images at the beginning, thus having ranks from 1 to N_R

$$\widetilde{Rank_m} = \frac{1}{NN'_R}(\sum_{i=1}^{N} \frac{R_i s_i}{maxS} - \sum_{i=1}^{N} \frac{i * s_i}{maxS}) = \frac{1}{NN'_R}(\sum_{i=1}^{N'_R} \frac{R_i s_i}{maxS} - \sum_{i=1}^{N'_R} \frac{i * s_i}{maxS})$$

N**:** like above

i**:** images are sorted by similarity (most similar has lowest i), thus the ideal ranking would be $R_i = i$

s_i**:** similarity value for image i

N'_R**:** no. of relevant images, that is, all images with $s_i > 0$

$maxS$**:** maximum similarity value, five in this case

To account for multiple similarity values s_i, the ranks R_i are weighted with $\frac{s_i}{maxS} \in [0,1] : \sum_{i=1}^{N} \frac{R_i s_i}{maxS}$. The same is done for the ideal result: $\sum_{i=1}^{N} \frac{i * s_i}{maxS}$. The reasoning is that more similar images, when misplaced, should yield a larger distance from the ideal result. Note that the original measure is obtained if one chooses a ground truth with similarity values $\{0, maxS\}$: non-similar images yield $R_i * 0 = 0$, similar images $R_i * \frac{maxS}{maxS} = R_i$.

This also explains why it doesn't matter whether one sums up to N_R/N'_R or N: the non-similar images can be left out and the remaining images are exactly the N_R/N'_R similar images for $\widetilde{Rank}$ and $\widetilde{Rank_m}$ respectively. Note that the s_i need to be sorted by similarity, i.e. as stated above the highest s_i have the lowest i. Finally, the distance is normalized over the number of images in the DB and the number of similar images for the given query, N and N_R/N'_R.

$\widetilde{Rank_m}$ allows to compare CBIRS based on multiple similarity degrees. However, the measurements are not comparable across different queries or different image sets: The worst possible measurement depends on the ratio of N_R to N, i.e. it differs from query to query and image database to image database. Additionally, the worst value depends on the similarity distribution for a given query: it's a difference whether for a given query we have one very similar image and four moderately similar images or the other way round.

Therefore, as a second step, we introduce the *Worst Result Normalization*: We additionally normalize the performance measure over the worst possible result for the given query and image set. The worst result for a query is simply the reverse ideal result, i.e. the most similar images are ranked worst and appear at the end of the list while all non-similar images are ranked first. The such modified measure is then:

$$\widetilde{Rank_{WRN}} = \frac{\frac{1}{NN'_R}\sum_{i=1}^{N'_R} \frac{R_i s_i}{maxS} - \sum_{i=1}^{N'_R} \frac{i * s_i}{maxS}}{\frac{1}{NN'_R}\sum_{i=1}^{N'_R} \frac{(N-i) s_i}{maxS} - \sum_{i=1}^{N'_R} \frac{i * s_i}{maxS}} = \frac{\sum_{i=1}^{N'_R} R_i s_i - \sum_{i=1}^{N'_R} i * s_i}{\sum_{i=1}^{N'_R} (N-i) s_i - \sum_{i=1}^{N'_R} i * s_i}$$

The factor $\frac{1}{NN'_R}$ is left out, as well as $\frac{1}{maxS}$. $\widetilde{Rank_{WRN}}$ yields values in $[0,1]$ with 0 being the ideal and 1 the worst result. This makes it possible to compare the performance of systems across different queries and image sets.

Missing Images Problem

NAR requires that every similar image ($s_i > 0$) in the searched data set or image domain is ranked by the tested CBIRS, or in other words the result list must contain all similar images (*Recall* = 1). Since the CBIRS does not know beforehand which images are similar (unless it's a perfect system), all images must be returned. All search engines which have been tested provide some sort of threshold parameter to adjust the result list length, so in theory this was no problem. In practice, however, a number of systems failed to return all images, probably due to bugs. The result is that some systems did produce rather bad results: they are ranked worse than they probably would have been had they returned all images.

To deal with this problem, a number of approaches have been tested, but none of them led to a satisfying solution. In order to get comparable measures the result lists with missing images were manually completed by simply adding those images, giving them the worst possible rank. The worst possible rank is the rank of the last image in a list of all images in the domain, or simply the number of images in the domain. This is a worst case assumption punishing systems which leave out similar images rather hard. But any other solution would be based on illegal assumptions about how the CBIRS in question would have ranked the missing images, giving it an unfair edge.

The bottom line is that some systems produce results which are not really comparable. A solution might be to use an all together different performance measure than NAR which would allow to measure performance based on result lists of arbitrary length. How much this problem affected the various CBIRS is explained in subsections 4.2 through 4.6.

3.7 Comparing CBIR Systems

The performance measures introduced in the previous section allows one to measure the quality of single query results. To compare systems to one another, one needs to somehow merge these measurements into a single value representing the overall performance of the system, s.t. a ranking of the systems can be build. We used two methods, a ranking method and a scoring method.

3.7.1 Ranking

The ranking method is reminiscent of rankings used in sports competitions, where athletes have to perform in different disciplines. The single queries are analogous to the sport disciplines, the systems to the athletes. First, the systems are ranked only based on single queries. Each system then has a rank for each query or within each discipline. Now, for each system the worst and the best ranking is discarded and an average rank is computed from the rest. This average rank is then used to build an overall ranking of all systems. Systems might receive the same rank in the end, but it is unlikely. Figure 2 shows the resulting rank list.

The method can be slightly modified by simply not discarding the worst and best ranking when computing the average rank. This results in a slightly more distinct ranking (i.e. less systems are ranked equally), but essentially it doesn't differ much from the first method.

3.7.2 Scoring

The ranking produces a clear result, but doesn't say anything about the distance of systems between one another. How close, for instance, are the first three systems? We used a scoring to answer this question. A value in $[0,1]$ is computed for each system, where 0 means worst and 1 best. To be more precise, 0 represents a hypothetical system that scores lowest on all queries and 1 such a system that scores highest on all queries.

As with the ranking, in the first step, performance is evaluated separately for each query. Each system's performance on a given query, computed through $\widetilde{Rank}_{WRN}$, is mapped onto the interval $[0,1]$ using the following simple formula:

$$sc_i = \frac{r_i - b}{w - b}$$

The terms are

r_i: $\widetilde{Rank_{WRN}}$ result of system i for the current query

b: best result for the current query

w: worst result for the current query

sc_i: score of system i for the current query

The resulting scores sc_i show the performance of system i in relation to all other systems for the given query, with the worst system having $sc_i = 0$ and the best $sc_i = 1$. This preserves the relative distances from the original measures. Now, the overall score for each system is simply the mean of all per-query-scores of this system. Figure 2 shows the so computed scores for all systems.

The scoring method essentially produces the same ordering among the systems as the ranking method, yet there are some noticeable differences. Some systems which are clearly better than other systems within the scoring fall behind those systems within the ranking. The ranking essentially just counts how often a system wins or looses against the other systems, while within the scoring it is also important by how far the system wins or looses. So while one victory is easily out-weighted by lots of losses in the ranking, in scoring a clear victory might still out-weight a number of close losses. In other words, within the ranking, a system is the better the more often it ranks high with the single queries and within the scoring, a system is the better the greater the distance to the other systems is within the single queries.

4 Results

Figure 2 shows the ranking and scoring for all systems. As can be seen, the VIPER/GIFT CBIRS in its standard configuration (pset b) clearly leads the field, followed by Caliph & Emir, SIMBA and then PictureFinder and Oracle Intermedia. As mentioned in section 3.7.2 one can see that the two comparison methods slightly differ: in the ranked list, for example Caliph&Emir pset d is behind SIMBA pset e, whereas in the scoring Caliph&Emir pset d clearly has a better score than the SIMBA system.

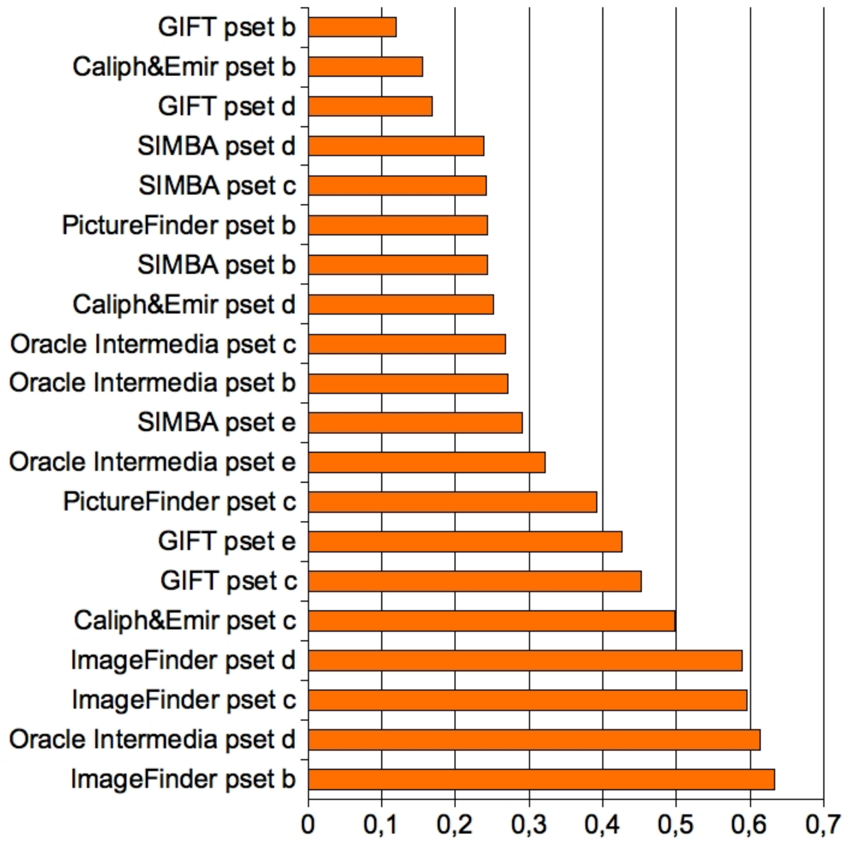
ImageFinder's performance is rather bad compared with the other systems. The most probable cause for this is that ImageFinder needs to be adjusted to the task at hand, a more detailed discussion is given in section 4.1.

Considering the different parameter sets one can see the clear tendency for the standard sets (psets b) to outperform the other sets. Only in the scoring, several CBIRS (Oracle Intermedia, ImageFinder, SIMBA) score better with a different set than the default one, but the difference is small, if not marginal. An explanation for this would be that the default settings already are an optimized parameter set for the respective CBIRS. The changes that we applied (with no explicit parameter optimization in mind) to create the other parameter sets most likely produce a less optimal set.

Table 2 shows an additional ranking of the best five systems per domain. This ranking shows that GIFT/VIPER also leads within four out of the six per domain rankings. Caliph&Emir also shows good performance scoring second within the three domains *Band*, *Textur* and *Urlaub*. The ranking method employed here is the one which also includes worst and best rankings (as explained in section 3.7.1)[14].

A comparison of all queries, based on the average results over all CBIRSs, showed that the *Blumen* queries seem to be the hardest ones. The easiest query seems to be the one from the *Verfälschung* domain, but it exhibits quite a big standard deviation.

[14] As mentioned in section 3.7.1 this method doesn't really differ much from the other ranking method. The reason for applying this method is basically that some domains only have two queries to begin with and thus no worst and best per-query ranking can be left out.



1. GIFT pset b
2. GIFT pset d
3. Caliph&Emir pset b
4. SIMBA pset b, SIMBA pset c
5. PictureFinder pset b, SIMBA pset d
6. Oracle Intermedia pset b,
7. Oracle Intermedia pset c
8. SIMBA pset e
9. Caliph&Emir pset d
10. Oracle Intermedia pset e
11. PictureFinder pset c
12. GIFT pset e
13. GIFT pset c
14. Caliph&Emir pset c
15. ImageFinder pset b
16. ImageFinder pset d
17. Oracle Intermedia pset d
18. ImageFinder pset c

Figure 2: Left: Scoring of all systems. Right: Ranked list of all systems.

Blumen 1. SIMPLIcity pset c 2. Oracle Intermedia pset b, PictureFinder pset b 3. Oracle Intermedia pset c 4. Oracle Intermedia pset e 5. GIFT pset d	**Urlaub** 1. GIFT pset b, GIFT pset e 2. Caliph&Emir pset d 3. SIMPLIcity pset b 4. GIFT pset d, SIMPLIcity pset c 5. Caliph&Emir pset c	**Kirschen** 1. GIFT pset b 2. SIMBA pset c 3. SIMPLIcity pset b 4. GIFT pset d, SIMBA pset b, SIMBA pset d 5. SIMPLIcity pset d
Band 1. GIFT pset d 2. Caliph&Emir pset b 3. PictureFinder pset b 4. GIFT pset b 5. SIMBA pset e	**Textur** 1. SIMBA pset b, SIMBA pset c 2. Caliph&Emir pset b, SIMBA pset e 3. SIMBA pset d 4. Oracle Intermedia pset e 5. Oracle Intermedia pset b	**Verfälschung** 1. GIFT pset b 2. SIMPLIcity pset c 3. GIFT pset d 4. Oracle Intermedia pset b 5. Caliph&Emir pset b, SIMBA pset b, SIMBA pset c

Table 2: The five best systems per domain.

4.1 SIMPLIcity and ImageFinder

The SIMPLIcity CBIRS implements some interesting technologies, namely the region based feature extraction, the Integrated Region Match measure and the classification of images in order to improve the search performance. In the setting pset c[15] in the *Blumen* domain, it is the best system showing a high potential. Unfortunately, for most of the other domains and psets results were distorted by missing images. Therefore we decided to exclude SIMPLIcity from the overall comparison and just to include it in the per-domain ranking (figure 2).

ImageFinder provides some kind of basic image search/recognition/match platform which a user modifies and tweaks to her specific needs. That is, compared to the others, ImageFinder is not solely meant for CBIR. Its vast numbers of parameters need to be optimized for CBIR, which was out of this works scope. Therefore, the overall results for ImageFinder are rather poor. However, for the hardest of all queries, *Blumen-10*, ImageFinder produced a good result (6th place out of 20), which shows its potential[16].

[15] For pset c of SIMPLIcity, the program *classify* was called with parameter -f 5.

[16] The result was produced with ImageFinder pset c: An unsupervised filter was used, and Images were processed with the Laplace 5 filter.

4.2 SIMBA

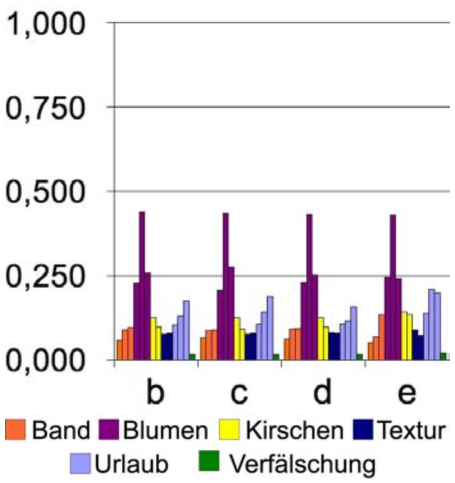


Figure 3: $\widetilde{Rank}_{WRN}$ for all SIMBA psets on all queries. Queries are grouped by pset and color coded by domain.

SIMBA uses *invariant feature histograms*, which are invariant against rotations and translations of parts of the image, e.g. objects. The feature extraction is based on kernel functions. The size of the kernel influences the size of objects SIMBA "recognizes" in images, i.e., with a big kernel, features are invariant only against rotation and translation of bigger objects [Sig02, page 24].

For SIMBA we created three psets besides the default set. Psets c and d were first tests and turned out less meaningful. Pset e provides a bigger kernel function than the default set. Specifically, kernel $(40,0),(0,80)$ was used for all color channels.

The results show that SIMBA performs well and is third in our benchmark. It shows poor within the *Blumen* domain (best pset is e with rank 14) (see table 2) and very good within the domain *Textur,* SIMBA pset b. From figure 3 one recognizes that the bigger kernel of pset e results only in minor changes in the domain *Band* with respect to the default set.

4.3 PictureFinder

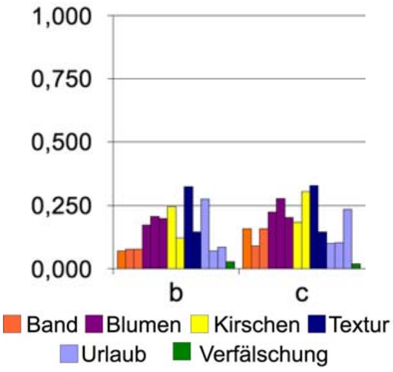


Figure 4: $\widetilde{Rank}_{WRN}$ for PictureFinder psets b and c on all queries. Queries are grouped by pset and color coded by domain. The results of the *Urlaub* domain are the 2.,3. and 4. bar counting from right.

PictureFinder models images with sets of polygons of different sizes, colors and textures. Basically, the polygons represent prominent forms in the images. PictureFinder can thus be seen as shape focussed, which might explain the good results in the *Blumen* and *Band* domains.

We modified one parameter, which determines whether the position of a polygon influences the similarity assessment or not. The default setting (pset b) was that the polygon position is important: roughly speaking, two images are more similar if they contain similar prominent forms in similar positions. Accordingly, in pset c the position was not important: only size, color and texture of polygons determined similarity.

Regarding the position as important (pset b) shows generally better results than not doing so (pset c). An exception is an outlier in domain *Urlaub*. The reason might be that for this query there exists lots of similar images (according to GT) with similar prominent forms in *different* positions. As for pset c it has been generally worse: we suspect that too many less similar images (GT), which however contain similar forms in other positions, are seen as similar to the query by PictureFinder.

Missing images in the domain *Textur* distorted these results somewhat, causing the outlier.

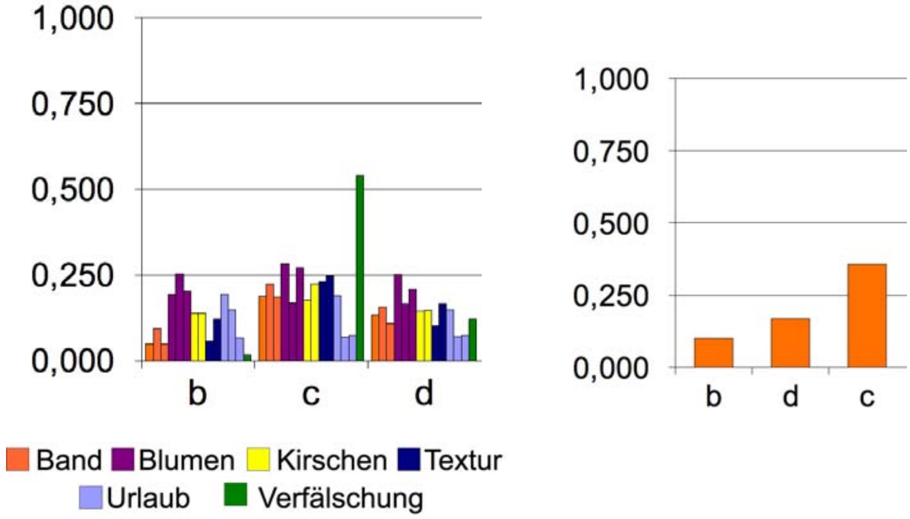


Figure 5: $\widetilde{Rank}_{WRN}$ and Scoring for all Caliph&Emir psets on all queries. On the left, queries are grouped by pset and color coded by domain.

4.4 Caliph&Emir

Caliph&Emir is a combination of two applications for managing image data. Caliph can be used to annotate the images, and Emir is the corresponding search engine. A strong feature of Caliph is its representation of annotations as semantic graphs, which greatly enhances the search. For our benchmark however, we focussed on Emir's visual similarity search. To use Emir as CBIR system, relevant features in form of MPEG-7 descriptors have to be extracted and stored as MPEG-7 documents by Caliph.

Caliph&Emir offer the three descriptors *ColorLayout*, *EdgeHistogram* and *ScalableColor* to choose from when performing a search. Unfortunately, *ScalableColor* didn't seem to work (the search didn't seem to start with this descriptor), so only *ColorLayout* and *EdgeHistogram* were used. The parameter settings were as follows: the default setting b was used for *ColorLayout*, pset c was used only for *EdgeHistogram* and pset d combines the two descriptors for the search.

The Caliph&Emir CBIRS is the second best system in this test behind VIPER/GIFT, as shown by the ranking and scoring (see figure 2). There is one outlier with pset c in the *Verfälschung* domain caused by missing images, but otherwise the results are rather balanced.

It turned out that *ColorLayout* in general is clearly better than *EdgeHistogram* or the combination of the two descriptors, see figure 5. There are however a few queries where *EdgeHistogram* beats *ColorLayout*, for example the query *Blumen-10*, shown in figure 6.

4.5 VIPER/GIFT

The design of the VIPER search mechanism, which is now part of the GIFT architecture as a plugin, borrows much from text retrieval. It uses an inverted index of up to 80 000 features, which are like terms in documents. For each image roughly 1000 features are extracted.

The numerous features can be grouped as color features and texture features. Each group in turn consists of histogram features and block or local features. The color features are based on the HSV color space, which is quantized into bins. From these, the color histogram features are extracted. Unlike other

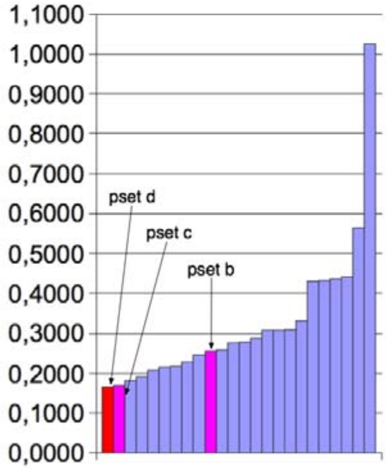


Figure 6: Query *Blumen-10* for all systems.
Caliph&Emir pset d is marked red (darker), c and b are marked light magenta (lighter).

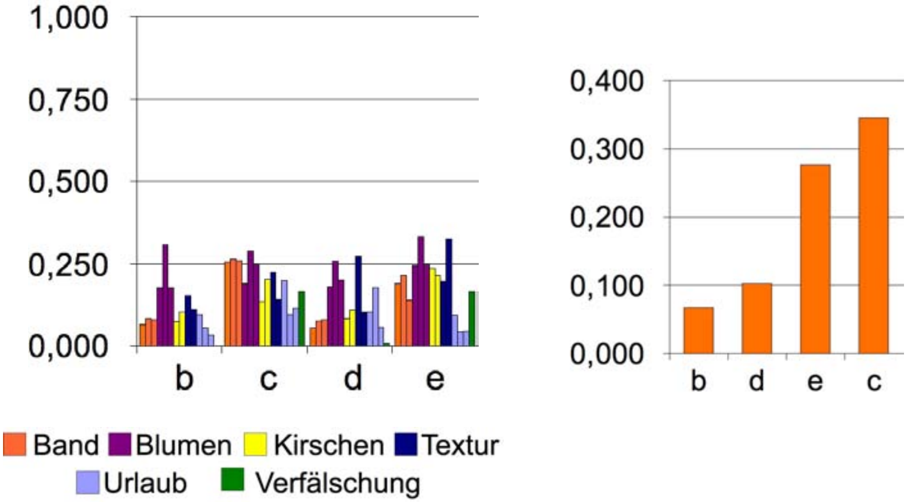


Figure 7: $\widetilde{Rank}_{WRN}$ and Scoring for all psets of VIPER/GIFT.

CBIRSs, in VIPER each bin of a histogram counts as feature. As local color features the *mode color*s[17] of small blocks are used the images are split into. The texture features are based on *Gabor filters*[18]. The filters yield local texture features which are also based on small blocks of the image. Additionally, histograms of the average filter outputs are stored as a measure for an image's overall texture qualities.

The similarity measure in VIPER is similar to methods used in text retrieval. Two methods are used for feature weighting and the computation of a similarity score: (1) classical IDF, the classical method from text retrieval. It weights and normalizes all features together. (2) *Separate Normalisation*, which weights and normalizes the different feature groups (color histogram, color blocks, Gabor histogram, Gabor blocks) separately and merges the results.

VIPER/GIFT ranks and scores best of all systems. Furthermore it ranks best in 4 of the 6 separate domain rankings. This clearly marks VIPER/GIFT as the best CBIRS in this benchmark. Note that except for the *Blumen* domain it's always VIPER/GIFT's default setting (pset b) which achieves the best results.

The different parameter sets are the following: pset b uses separate normalization and all feature sets. Pset c is like b, but with classical IDF instead of separate normalization. Pset d uses separate normalization and color blocks and histogram only (color focussed). Pset e uses separate normalization and Gabor blocks and histogram only (texture focussed).

Unfortunately this CBIRS is also affected by the missing-images-problem. The outlier with pset d in the domain *Textur* is caused by this, see figure 7 (6th bar from the right). But apart from this, the results are not influenced by this problem.

Another outlier can be seen in the texture domain *Textur* with pset e (only texture features). The reason could be that the texture images of this particular query also have strong overall color features, which are ignored in pset e. Interestingly, for both queries in domain *Textur,* the *color centered* pset d seems to outperform the *texture centered* pset e[19]. This trend is even stronger in the overall result: color features and the combination of all features are better than texture features only, see figure 7.

Our benchmark confirms a result shown in [Mül02], namely that the weighting method separate normalisation (psets b) is better than classical IDF (pset c). Classical IDF gave the texture feature groups (i.e. Gabor histogram and blocks) too much weight due to much higher numbers of features (according to [Mül02, page 50]). We conclude that VIPER/GIFT performs best employing separate normalization and color features (psets b and d). Emphasizing texture too much (directly in pset e, indirectly through classical IDF in pset c) degrades its performance.

4.6 Oracle Intermedia

Intermedia is an extension to *Oracle DB*, providing services and data types for handling media data (sounds, images, videos etc.). Specifically images and according signatures can be stored as objects in a database. A score function is used to compare images by calculating distances between them based on their signatures. This can be used to write SQL-queries with example images which produce a ranked list of similar images. Four parameters govern the image search: *color*, *texture*, *shape* and *location*. The score is computed using the *Manhattan distance*: a weighted sum of single distances for *color*, *texture* and *shape.* It remained unclear to us how *location* integrates in the score. The weights are normalized so that their sum equals 1.

The Oracle Intermedia CBIRS shows very good results with the exception of one big outlier. The parameter sets b - e are based on the four parameters and are defined as follows: pset b: (color = 0.6, texture = 0.2, shape = 0.1, location = 0.1); pset c: (color = 0.5, texture = 0.5, shape = 0, location = 0); pset d: (shape = 1.0, all others set to 0); pset e: (color = 0.33, texture = 0.33, shape = 0.34, location = 0).

[17] The most frequent color within the block.

[18] These are often used to describe the neuron layers in the visual cortex which react to orientation and frequency.

[19] Assuming that the outlier result for pset d caused by missing images would be significantly better without the missing-images-problem.

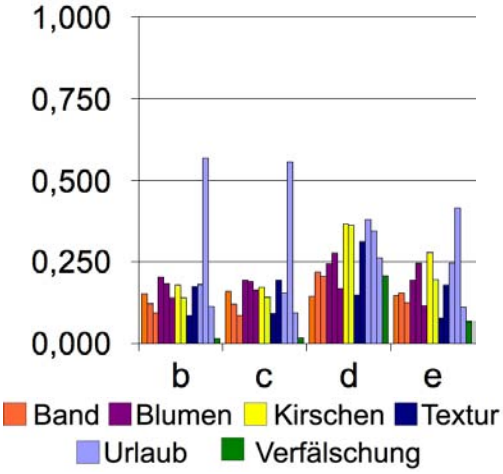


Figure 8: $\widetilde{Rank}_{WRN}$ for all psets of Oracle Intermedia

The mentioned outlier can be seen easily in the *Urlaub* domain. The causes remained unclear to us. The query (*urlaub-Image15*) doesn't seem special, none of the other CBIR systems show a similar behaviour. The outlier is less dominant in psets d and e, but still clearly visible. These parameter sets put more weight on shape search, which seems to improve performance on this query but degrade most of the others.

As mentioned the results are otherwise very good, especially in the *Blumen* domain, where Oracle Intermedia (pset b) ranks second behind SIMPLIcity. It would be interesting to see how the outlier could be improved and in how this would influence Oracle Intermedia's ranking and scoring.

Focussing on shape in pset d degrades performance practically for all queries. Also, equally using color, texture and shape is not as good as concentrating on color (pset b) or color and texture (pset c), see the ranking and scoring in figure 2. The shape-only pset d is actually much worse then the other three sets, which could mean that the shape parameter is only useful in special cases.

5 Summary and outlook

We have benchmarked seven CBIR Systems: SIMBA, SIMPLIcity, PictureFinder, ImageFinder, Caliph &Emir, VIPER/GIFT and Oracle Intermedia. Different parameter settings of the CBIRSs have been created by modifying interesting parameters.

The ground truth for the benchmarks was determined through an online survey. The image database was built from freely usable images taken from various sources. The images have been grouped roughly by similarity and forming image domains.

The systems were benchmarked by running a number of defined queries using the query by example approach. The systems were then compared to each other based on their visual retrieval accuracy. The results from the runs were measured using the custom accuracy measure $\widetilde{Rank}_{WRN}$, which we derived from *Normalized Average Rank*[MMS$^+$01b]. Based on these measurements the systems have been ranked and scored.

Results of the Benchmarks

VIPER/GIFT (with its default parameter setting) was found to be the best system in our tests. It ranks and scores best in the overall comparison, and also is first in many of the domain specific rankings. Caliph&Emir and SIMBA are second and third, respectively. We found that some systems have weak spots, like for example Oracle Intermedia.

Color features seem to work best for most systems, other features seem to be more domain specific. Different parameter settings produce significant differences, but mostly do not result in big jumps in the overall performance. From the very good performance of Caliph&Emir it can be seen that MPEG-7 descriptors are a good choice for CBIR, especially the descriptor *ColorLayout*. Changes to the default parameters tend to worsen the result, most likely because the default parameters are already optimized.

Two significant problems were encountered: First, ImageFinder's parameters obviously need to be optimized to specific domains. Second, SIMPLIcity kept crashing during the benchmark runs resulting in missing images, which distorted the results. Therefore, we decided to not include SIMPLIcity in the overall ranking and the scoring.

Outlook

Clearly the trends go towards semantic search, for example by using automatic annotation techniques, and towards Web engine capabilities (addressing scaling problems). It is also agreed that the reliance on standards for the interoperable usage is important. It seems that these are one of the most important challenges in content-based image retrieval right now: closing the semantic gap (at least a little) and agreeing on standards, for benchmarks as well as for the CBIR systems themselves. Interesting future directions would be to combine our benchmark with text retrieval benchmarks and to measure the system performance in terms of memory and CPU usage.

6 Acknowledgements

We would like to thank the authors and developers of SIMBA, SIMPLIcity, PictureFinder, ImageFinder, Caliph&Emir, VIPER/GIFT, Oracle Intermedia, for their support and many helpful thoughts and suggestions. Specifically, we would like to thank Gerd Brunner and Hans Burkhardt from the University of Freiburg for granting access to a server running SIMBA, Prof. James Z. Wang for providing SIMLIcity, Thorsten Hermes from TZI in Bremen for providing PictureFinder, the staff at Attrasoft for providing a copy of ImageFinder and Wolfgang Müller from the University of Bamberg for his helpful suggestions regarding VIPER/GIFT. Furthermore we would like to thank Markus Koskela and Jorma Laaksonen for their help and support with PicSOM, Gianluigi Ciocca and Raimondo Schettini for their efforts and help with Quicklook^2.

References

[att07] Attrasoft, company web page. `www.attrasoft.com`, May 22 2007.

[ben] Benchathlon website. `http://www.benchathlon.net/`.

[CGPF07] M. Cord, P.-H. Gosselin, and S. Philipp-Foliguet. Stochastic exploration and active learning for image retrieval. *Image and Vision Computing*, 25:14–23, 2007.

[CGS01] Gianluigi Ciocca, Isabella Gagliardi, and Raimondo Schettini. Quicklook2: An integrated multimedia system. *Journal of Visual Languages & Computing*, 12(1):81–103, 2001.

[CTB^{+}99] Chad Carson, Megan Thomas, Serge Belongie, Joseph M. Hellerstein, and Jitendra Malik. Blobworld: A system for region-based image indexing and retrieval. In *Third International Conference on Visual Information Systems*. Springer, 1999.

[DJLW08] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 40(2):60, 2008.

[DK08] Mario Döller and Harald Kosch. The MPEG-7 multimedia database system (MPEG-7 MMDB). *Journal of Systems and Software*, 81(9):1559–1580, 2008.

[DKN04] Thomas Deselaers, Daniel Keysers, and Hermann Ney. Fire – flexible image retrieval engine: Imageclef 2004 evaluation. In *Working Notes of the CLEF Workshop*, pages 535–544, Bath, UK, September 2004.

[FCPF01] J. Fournier, M. Cord, and S. Philipp-Foliguet. Retin: A content-based image indexing and retrieval system. *Pattern Analysis and Applications Journal, Special issue on image indexation*, 4(2/3):153–173, 2001.

[FSN^{+}01] Myron Flickner, Harpreet Sawhney, Wayne Niblack, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, David Steele, and Peter Yanker. Query by image and video content: the QBIC system. pages 255–264, 2001.

[GK04] M. Gabbouj and S. Kiranyaz. Audio-visual content-based multimedia indexing and retrieval - the MUVIS framework. In *Proceedings of the 6th International Conference on Digital Signal Processing and its Applications, DSPA*, pages 300–306, Moscow, Russia, March 31 - April 2 2004.

[HMH05] Th. Hermes, A. Miene, and O. Herzog. Graphical search for images by PictureFinder. *Journal of Multimedia Tools Applications*, 27(2):229–250, 2005.

[ima05] Imageclef. `http://ir.shef.ac.uk/imageclef/`, November 2005.

[KLLO00] Markus Koskela, Jorma Laaksonen, Sami Laakso, and Erkki Oja. The picsom retrieval system: Description and evaluations. In Proceedings of Challenge of Image Retrieval (CIR 2000). Brighton, UK, P.O. BOX 5400, Fin-02015 HUT, Finland, May 2000. Laboratory of Computer and Information Science, Helsinki University of Technology.

[LBK03] Mathias Lux, Jutta Becker, and Harald Krottmaier. Caliph & emir: Semantic annotation and retrieval in personal digital photo libraries. In *Proceedings of CAiSE '03 Forum at 15th Conference on Advanced Information Systems Engineering*, pages 85–89, Velden, Austria, June 16th-20th 2003.

[LTU07] Ltu technologies. `www.ltutech.com`, August 2007.

[LWW00] Jia Li, James Z. Wang, and Gio Wiederhold. IRM: Integrated region matching for image retrieval. In *Proc. ACM Multimedia*, pages 147–156, Los Angeles, CA, October 2000. ACM, ACM.

[MMS01a] Henning Müller, Wolfgang Müller, and David McG. Squire. Automated benchmarking in content-based image retrieval. *IEEE International Conference on Multimedia & Expo*, page 290, 2001.

[MMS^{+}01b] Henning Müller, Wolfgang Müller, David McG. Squire, Stéphane Marchand-Maillet, and Thierry Pun. Performance evaluation in content-based image retrieval: overview and proposals. *Pattern Recogn. Lett.*, 22(5):593–601, 2001.

[MMW06] Stéphane Marchand-Maillet and Marcel Worring. Benchmarking image and video retrieval: an overview. In *MIR '06: Proceedings of the 8th ACM international workshop on Multimedia Information Retrieval*, pages 297–300, New York, NY, USA, 2006. ACM Press.

[Mül02] Henning Müller. *User Interaction and Performance Evaluation in Content-Based Visual Information Retrieval.* PhD thesis, l'Université de Genève, 2002.

[oim07] Oracle Intermedia. `http://download.oracle.com/docs/html/B14302_01/toc.htm`, August 2007.

[QMTM04] Till Quack, Ullrich Mönich, Lars Thiele, and B. S. Manjunath. Cortina: a system for large-scale, content-based web image retrieval. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 508–511, New York, NY, USA, 2004. ACM Press.

[Sig02] Sven Siggelkow. *Feature Histograms for Content-Based Image Retrieval.* PhD thesis, Albert-Ludwigs-Universität Freiburg im Breisgau, 2002.

[VT02] Remco C. Veltkamp and Mirla Tanase. Content-based image retrieval systems: A survey. Technical report, Department of Computing Science, Utrecht University, 2002.

[VV99] Jules Vleugels and Remco C. Veltkamp. Efficient image retrieval through vantage objects. In *Visual Information and Information Systems*, pages 575–584, 1999.

[WLW01] James Z. Wang, Jia Li, and Gio Wiederhold. Simplicity: Semantics-sensitive integrated matching for picture libraries. In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, volume 23, pages 947–963, 2001.

[YHR06] A. Yavlinsky, D. Heesch, and S. Rüger. A large scale system for searching and browsing images from the world wide web. In *Proceedings of the International Conference on Image and Video Retrieval*, Lecture Notes in Computer Science 3789, pages 530–534. Springer, 2006.

Delivery Context Descriptions – A Comparison and Mapping Model

Christian Timmerer, Johannes Jabornig, and Hermann Hellwagner

Department of Information Technology (ITEC)
Klagenfurt University
Universitätsstrasse 65-67
A-9020 Klagenfurt
{christian.timmerer, hermann.hellwagner}@itec.uni-klu.ac.at
j.jabornig@jabjo.net

Abstract: Nowadays, mobile devices have implemented several transmission technologies which enable access to the Internet and increase the bit rate for data exchange. Despite modern mobile processors and high-resolution displays, mobile devices will never reach the stage of a powerful notebook or desktop system (for example, due to the fact of battery powered CPUs or just concerning the small-sized displays). Due to these limitations, the deliverable content for these devices should be adapted based on their capabilities including a variety of aspects (e.g., from terminal to network characteristics). These capabilities should be described in an interoperable way. In practice, however, there are many standards available and a common mapping model between these standards is not in place. Therefore, in this paper we describe such a mapping model and its implementation aspects. In particular, we focus on the whole delivery context (i.e., terminal capabilities, network characteristics, user preferences, etc.) and investigated the two most prominent state-of-the-art description schemes, namely User Agent Profile (UAProf) and Usage Environment Description (UED).

1 Introduction and Motivation

Today, the access to the Internet via mobile phones and other devices which are limited in power capacity and/or rendering functionality increases. Additionally, manufacturers equip their devices with technologies to access various networks, and mobile providers offer services for connecting these devices to the Internet. Thus, using small, mobile devices enables the access to the Internet but often the available content of Web pages is not suitable according to their capabilities. In order to solve this issue, some projects and standards have been released which deal with the description of capabilities and characteristics of all kind of devices. Terminal capabilities and network conditions as well as user characteristics may allow the adaptation of the content for certain purposes.

Amongst others, there are two standards which were designed to meet the requirements of device and user descriptions and are often compared to each other. The first one was released by the WAP Forum (now the Open Mobile Alliance) and is named User Agent Profile (UAProf) [OM06] and the second one is the Usage Environment Description (UED) standard which was standardized within MPEG-21 Digital Item Adaptation (DIA) [VT05]. The aim of this paper is to build a mapping model for these description formats enabling context-aware content delivery independent of the actual description format used. Recently, W3C has started a new work item with the aim to define a delivery context ontology [LF08] but, still, the mapping issue remains.

While the User Agent Profile standard is very popular and has been implemented in a wide range of mobile devices the Usage Environment Descriptions are only limited available and tools which would ease the creation of such descriptions rarely exist. However, a high availability of Usage Environment Descriptions is desired by research projects [Da06][Ax08][En08] and, thus, obtaining information about terminals, networks and users from projects with similar aims to that of Usage Environment Descriptions should be enabled. For example, an implementation compliant to a standardized delivery context description format A requires a mapping module if it receives descriptions compliant to another standard B and vice versa. In order to keep the mapping effort minimal and scalable the proposed method in this paper enable the implementation of a service that performs such kind of mapping.

This paper is organized as follows. Section 2 highlights the main requirements on standards for delivery context descriptions by classifying terminals and their properties. An analysis and comparison of delivery context description formats is presented in Section 3 while Section 4 describes the actual mapping model. Finally, the implementation details are described in Section 5 and the paper is concluded with Section 6 which contains also future work items.

2 Requirements on Standards for Context Descriptions

In this section we have identified different classes of terminals including their hardware and software capabilities. This kind of information should provide us a rough estimation on the requirements for delivery context description standards from the terminal’s point of view which also includes the access networks.

When the Internet became popular, the only way to access the Web was through a personal computer (PC) or a workstation. In general, these computers had large color displays with full graphic capabilities, sufficient computational power without battery issues, and a decent network connection [GLS06]. Nowadays, people tend to access the Web using smaller and mobile devices with various constraints on display capabilities, user input/output facilities, computational power, electric power, and access networks ranging from high-speed Wireless Local Area Network (WLAN) to low-speed General Packet Radio Service (GPRS). In the following we will provide a classification of the various terminals based on their hardware (HW) and software (SW) characteristics.

Table 2 in the Appendix provides an overview of HW/SW characteristics of different end user devices (i.e., desktop PC/workstation, notebook/tablet PC, sub-notebook/netbook, handheld, smart phone, and mobile phone) with respect to performance (i.e., CPU), display, permanent storage, memory, network connectivity, electric power, user input/output, extensibility, operating system support, and software in general.

A summary and comparison of terminal's display and memory properties is depicted in Figure 1. As one can see there is still a huge gap between classical mobile devices (i.e., phones) and devices that may have full power supply. Thus, a comprehensive delivery context standard needs to accommodate all these HW/SW properties in an easy-to-understand/use, extensible, and manageable way.

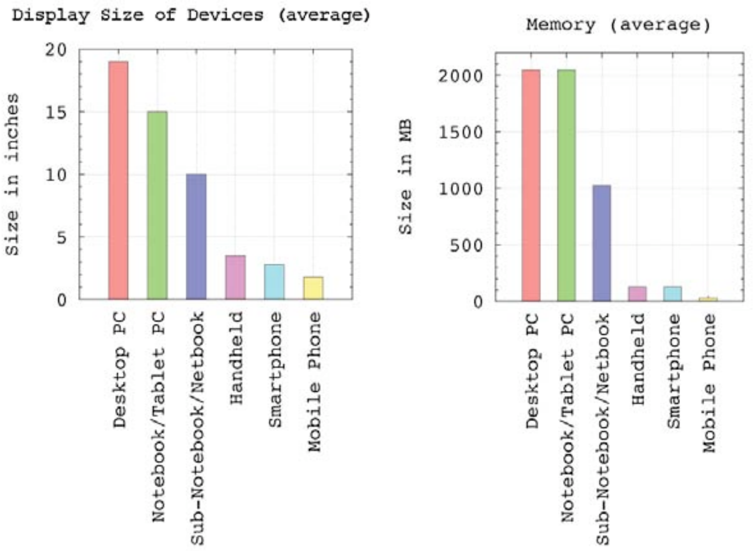


Figure 1. Summary and comparison of terminal's display and memory properties.

However, not only HW properties like screen size, color capabilities, or user input/output facilities are important, also SW properties like supported operating systems, audio/video/image codecs, etc. become more and more important as diversity of devices increases. In particular, the number of different coding formats a terminal is capable to support – both for encoding and decoding – is of interest for delivery context description formats. As there are so many coding formats available, some may have certain profile/level definitions, and even a class of terminals may define its own constraints, there exists a strong requirement to describe these properties effectively. A key functionality is the possibility to add new coding formats – e.g., through a registration authority – in a convenient and relatively unbureaucratic way as they arise rather rapidly on the market.

3 Analysis and Comparison of Context Description Formats

3.1 Composite Capabilities/Preference Profiles (CC/PP)

The Composite Capabilities/Preference Profiles (CC/PP) [Kl04] comprises descriptions based on the Resource Description Framework (RDF) which cover device capabilities and user preferences by introducing a two-level hierarchy consisting of *components* and *attributes*. Components are groups of attributes with related meaning such as the software or the hardware properties of a terminal. A CC/PP document shall contain at least one component each identified by an `rdf:type` attribute which indicates the type of the component. Attribute values may be *simple*, i.e., string and integer or rational numbers, or *complex*, i.e., set (`rdf:bag`) or sequence (`rdf:seq`) of simple values. However, CC/PP does not define a vocabulary of terms but provides a common structure for holding any arbitrary vocabulary. Thus, another description format is required which specifies the actual vocabulary, e.g., the User Agent Profile as described next.

3.2 User Agent Profile (UAProf)

The Open Mobile Alliance (OMA) defines the User Agent Profile (UAProf) [OM06] which is based on CC/PP and defines a vocabulary for describing characteristics and capabilities of mainly WAP-enabled mobile devices. The components can be clustered into

— `HardwarePlatform` defines manufacture, CPU type, display/audio output capabilities, device interaction possibilities, and keyboard layout;
— `SoftwarePlatform` comprises supported media types, preferred language, operating system, audio/video codecs, etc.;
— `BrowserUA` includes whether the browser supports certain (X)HTML features, frames, tables, and its JavaScript capabilities;
— `NetworkCharacteristics` holds information about supported and current bearers, security options, and some details about Bluetooth support; and
— various `WapCharacteristics` and `PushCharacteristics`.

3.3 Usage Environment Description (UED)

The Usage Environment Description (UED) is defined in Part 7 of MPEG-21, i.e., Digital Item Adaptation (DIA) [VT05]. The UED is a very comprehensive vocabulary organized in so-called properties. It is based on XML Schema and its properties can be divided into four categories:

— *User characteristics* provide information pertaining to the user plus his/her usage preferences/history, presentation preferences, accessibility characteristics, and location information including the user’s movement.
— *Terminal capabilities* comprise codec capabilities, input/output characteristics including display/audio output capabilities, and device properties such as device class, power/storage characteristics, data input/output facilities, and CPU capabilities.

- *Network characteristics* include static and dynamic properties pertaining to the capabilities (e.g., max. bandwidth) and conditions (e.g., available bandwidth) of a network.
- *Natural environment characteristics* provide means for describing lightning conditions, noise level, time, and location.

3.4 Delivery Context Ontology (DCO)

The Delivery Context Ontology (DCO) [LF08] is based the Web Ontology Language (OWL) [Mv04] and is divided into *four entities*, namely:

- *Environment* including information about the location and network.
- *Hardware* provides information about various hardware capabilities including display, input, memory, camera, Bluetooth, CPU, etc.
- *Measure* defines terms related to units with respect to physical electric charges and length as well as unit conversions.
- *Software* describes whether the delivery context supports certain APIs, formats (i.e., audio, video, image, text, binary), operating systems, application-layer protocols, Java/Web browser specifics, etc.

3.5 Analysis/Comparison

The previous sections provided an overview of existing standards for delivery context description formats. In this section we will analyse differences and commonalities of these formats. First of all, and most importantly, all standards make use of XML that provides extensibility but UED is based on XML Schema whereas CC/PP and, consequently, UAProf are based on RDF. The most recent standard in this series is DCO which adopted already OWL which is based on RDF. Hence, we observe an incompatibility at the level of technology used for these description formats, mainly between XML Schema and RDF. Although it is possible to provide a high-level mapping between these two technologies, the mapping of concrete schemas/instances itself is a difficult and cumbersome task [HL01].

The second observation we made was that there are only a few characteristics or capabilities that are common across all delivery context description formats in question, e.g., display capabilities and file/coding formats. However, there is sometimes a huge difference in the actual syntax used. For example, display resolution described as `horizontal=1024` and `vertical=768` versus `1024x768` or using MIME types for file/coding formats versus classification schemes (i.e., URNs).

Finally, CC/PP defines only a basic structure (i.e., components and attributes) without specifying a particular vocabulary of terms. UAProf adopts CC/PP and provides a concrete vocabulary mainly targeting WAP-enabled mobile devices. A repository of some specific device profiles is available [W308]. Other industry adoptions of CC/PP are not known but some are envisaged and documented in Annex E of [Ki07]. The UED defines both the structure (i.e., properties) and a comprehensive vocabulary while DCO defines an ontology including not only a vocabulary of delivery context terms but also basic measure units.

In conclusion, there is a need to describe the relationship between commonalities of the different delivery context description formats, i.e., a mapping model which will be described in the next section. To the best of our knowledge, such a mapping model has not been published yet.

4 Mapping Model

UAProf and UED are based on different data models as the former is based on RDF whereas the latter is based on XML Schema with having their pros and cons [HL01]. That is, RDF provides support for rich semantic descriptions but provides limited support for the specification of local usage constraints, e.g., cardinality and datatype constraints. On the other hand, XML Schema provides support for explicit structural, cardinality and datatype constraints but provides little support for the semantic knowledge necessary to enable a flexible mapping between metadata domains.

The main issue is to find a suitable technology for the mapping process which includes the advantages of both standards. Basically, the mapping can be performed by two approaches as discussed in the following and depicted in Figure 2:

— **Direct mapping model**: creating mapping functions that perform direct mapping from one standard to another.

— **Integration model**: integrating both models into a new one with functions to convert between this new model and the initial model.

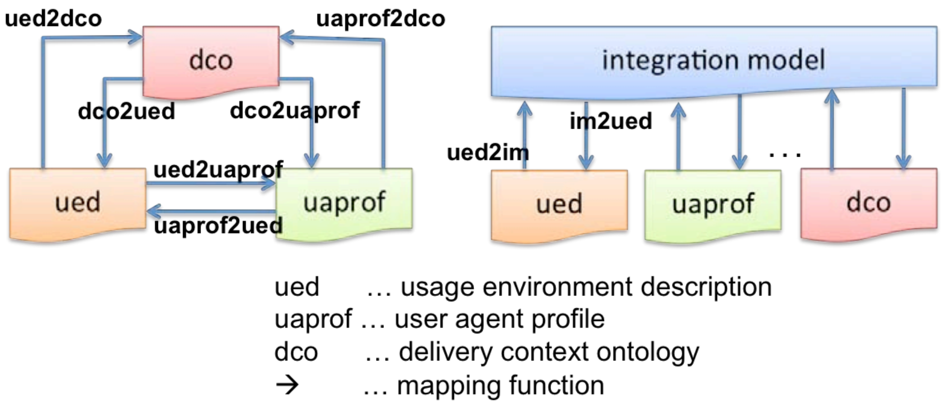


Figure 2. Direct Mapping Model vs. Integration Model.

As the *direct mapping model* provides an explicit mapping from one format to another format it lacks of flexibility with respect to the integration of other formats. Thus, it can be only applied for specific solutions whereby the number of explicit mappings increases exponentially with the number of formats between which mappings should be provided.

The *integration model* defines a common interface that allows the provisioning of the individual description formats. For new formats to be added, only the methods for converting to/from this model needs to be implemented without taking into account the existence of the other formats. Thus, the number of mappings increases linearly with the number of formats. However, the integration model needs to be implemented with a certain technology and those in question are XML Schema, RDF, or even OWL:

- One could define an **XML Schema** that covers all components of each standard and well-established XPath/XML processors could be used to extract the required data for a certain standard. Unfortunately, datatype or value range incompatibilities (e.g., UED: `colorCapable={true,false}` vs. UAProf: `ColorCapable={Yes,No}`) cannot be represented with XML Schema which requires external knowledge to be provided. Thus, it would be better to use tools which are able to express the relations between, e.g., datatypes or attribute values.
- **OWL** is based on RDF and provides means to describe the relationship between classes and properties, e.g., classes can be declared distinct or equal, restrictions on properties can be defined as transitive or functional, or the use of cardinality restricts the number of values which can be associated to properties.

4.1 Mapping Levels

The relationships between different delivery context description formats can be found at different levels within the entities of their respective schemas (i.e., XML Schema of OWL). In this paper we introduce four different mapping levels, namely *component*, *datatype*, *element*, and *value*. An example thereof is shown in Table 1.

Table 1. Example of Mapping Levels for Network Characteristics.

Level	UAProf Example	UED Example
Component	`prf:NetworkCharacteristics`	`dia:NetworkType`
Element	`prf:InputCharSet`	`dia:CharacterSetCode`
Datatype	`prf-dt:Boolean`	`xsd:Boolean`
Value	`Yes`	`true`

The *component* mapping level tries to map a predefined group of elements/attributes (e.g. `prf:NetworkCharacteristics`) to similar group of the other description format (e.g. `dia:NetworkType`). Difficulties may arise in case the structure is different, e.g., one has only attributes defined whereas the other includes also elements within a nested structure.

Thus, one needs to dig a level deeper and the *element* mapping level tries to map attributes/elements with equal semantics but possibly different syntax, i.e., different tag names. Note that a mapping at the component level is not always sufficient as indicated above which requires describing relationships between individual elements/attributes or even beyond.

The *datatype* mapping level tries to map datatypes with equal domains but different syntax whereas the *value* mapping level tries to map datatypes with different domains but equal semantics.

An example of mapping attributes with the equal semantics and Boolean values is described in Listing 1 which maps the `prf:ColorCapable` to the `map:colorCapable` (assuming `map:colorCapable` is the RDF/XML representation of the `colorCapable` attribute of the `dia:DisplayCapabilityType`).

Listing 1. Mapping `prf:ColorCapable` to `map:colorCapable`.

```
...
<owl:FunctionalProperty rdf:about ="&prf;ColorCapable">
  <owl:equivalentProperty rdf:resource="&map;colorCapable"/>
</owl:FunctionalProperty>
...
```

A problem arises as both datatypes are Boolean types but with different syntax: while `xsd:boolean` (as used within UED) accepts `true` and `false`, `prf-dt:Boolean` (as used within UAProf) accepts only `Yes` and `No` which requires an appropriate mapping. Listing 2 shows one possibility for such a mapping of these Boolean datatypes. Lines 2 to 6 and lines 8 to 12 describe properties to create a relation between a new defined resource for a Boolean value (prefixed by `btd`) and the Boolean values used by UAProf and UED. The rest provides the mapping between the values and the `bdt` Boolean datatypes.

Listing 2. Mapping the values of `prf-dt:Boolean` and `xsd:Boolean`.

```
...
<owl:DatatypeProperty rdf:about="&bdt;hasXsdBoolean">
 <rdf:type rdf:resource="&owl;FunctionalProperty"/>
 <rdfs:domain rdf:resource="&prf-dt;Boolean"/>
 <rdfs:range rdf:resource="&xsd;boolean"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="&bdt;hasOmaBoolean">
 <rdf:type rdf:resource="&owl;FunctionalProperty"/>
 <rdfs:domain rdf:resource="&xsd;boolean"/>
 <rdfs:range rdf:resource="&prf-dt;Boolean"/>
</owl:DatatypeProperty>

<bdt:Boolean rdf:about="&bdt;true">
 <bdt:hasXsdBoolean>true</bdt:hasXsdBoolean>
 <bdt:hasOmaBoolean >Yes</bdt:hasOmaBoolean>
</bdt:Boolean>

<bdt:Boolean rdf:about="&bdt;false">
 <bdt:hasXsdBoolean>false</bdt:hasXsdBoolean>
 <bdt:hasOmaBoolean>No</bdt:hasOmaBoolean>
</bdt:Boolean>
...
```

Another example is shown in Listing 3 which maps a Uniform Resource Name (URN) identifying a certain key input type to the equivalent string representation of UAProf. The usage of URNs to uniquely identify predefined terms is heavily used within UED.

Listing 3. Mapping of key input types: `DIA-KeyInputCS-NS:1` and `Querty`.

```
...
<owl:DatatypeProperty rdf:about="&key;hasMpegUrnRepresentation">
 <rdf:type rdf:resource="&owl;FunctionalProperty"/>
 <rdfs:domain rdf:resource="&key;Keyboard"/>
 <rdfs:range rdf:resource="&xsd;string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:about="&key;hasOmaKeyboard">
 <rdf:type rdf:resource="&owl;FunctionalProperty"/>
 <rdfs:domain rdf:resource="&key;Keyboard"/>
 <rdfs:range rdf:resource="&xsd;string"/>
</owl:DatatypeProperty>

<key:Keyboard rdf:about="urn:mpeg:mpeg21:2003:01-DIA-KeyInputCS-NS:1">
 <key:hasMpegUrnRepresentation>
   urn:mpeg:mpeg21:2003:01-DIA-KeyInputCS-NS:1
 </key:hasMpegUrnRepresentation>
 <key:hasOmaKeyboard>Querty</key:hasOmaKeyboard>
</key:Keyboard>
...
```

Datatypes such as `prf-dt:Number` and `xsd:nonNegativeInteger` can be mapped directly to each other because both cover the same range of values. However, integer values also raise problems when a mapping from `dia:bitsPerPixel` to `prf:BitsPerPixel` is provided because the UED standard defines an `xsd:integer` datatype and the UAProf standard uses the `prf-dt:Number` datatype which is equal to `xsd:nonNegativeInteger`. Of course, it is unlikely to describe the number of bits per pixel or the horizontal and vertical resolution with a negative number but there is the possibility to do that. OWL lacks to offer appropriate tools to describe what to do if such problems arise.

Another issue is raised with the datatype `prf-dt:Dimension` which is used to describe the resolution of a terminal's display within one string of the form "`HorizontalResolution`" x "`VerticalResolution`" (e.g., `1024x768`). This means that two values of the UED (i.e., `horizontal` and `vertical` attributes of the `Resolution` element) are combined to represent one value within UAProf. Thus, external tools are needed to enable such mappings.

4.2 Mapping Classes

In practice, the tag names and datatypes of different description formats can be clustered into four classes which are described in the following.

Direct. Elements falling into this class have equal semantics and compatible datatypes with equal domains but may differ in their syntax (i.e., tag name).

For example, `dia:bitsPerPixel` of type `xsd:integer` and `prf:BitsPerPixel` of type `prf-dt:Number` where these datatypes are compatible. Another example is the `dia:CharacterSetCode` of type `mpeg7:characterSetCode` (equal to `xsd:string`) and `prf:CcppAccept-Charset` of type `prf-dt:Literal` (equal to `xsd:anySimpleType`) which are used to store string representations of the supported character sets.

Advance. The class advance comprises elements describing the same concept (i.e., equal semantics) but with different, non-compatible datatypes and/or domains. Thus, the actual format is that much different and requires major changes if mapped from one format to the other. For example, the `dia:Resolution` includes two attributes (`horizontal` and `vertical`) for describing the resolution of a screen whereas `prf:SreenSize` uses only one value (e.g., `480x320`). Another example is the usage of classification schemes versus MIME types as detailed in Section 4.3. Thus, an advanced mapping mechanism is required.

Derive. This class includes mappings where element values can be derived from one or more elements of the respective other description format. The difference to the advance class is that for the derive class the semantic equality is not necessarily a requirement. For example, `prf:SoundOutputCapable` indicates whether a terminal is able to output sound which could be derived from only the presence of a `dia:AudioOutputCapability` element.

Extend. Elements that cannot be mapped directly, in an advanced way through additional mapping rules, or derived from other elements require proprietary extensions of the respective other description format. For example, properties defined within UAProf but not defined in UED require an extension of the UED schema by adding additional elements and datatypes representing these UAProf properties.

In our example, the UAProf standard defines six components and 77 elements which have been mapped – with respect to UED – to the classes described above (quantities in brackets): direct (4), advance (7), derive (4), and extend (62). The specific support for mobile phones within UAProf causes the high number within the class extend whereas UED does not provide means for describing WAP or push characteristics. One could now argue that such a mapping is not required. Please note that for most of the application scenarios – in particular, multimedia content adaptation – the required elements/attributes/tags fall into direct, advance, and derive classes, e.g., adaptation to screen size, codec, bitrate which are covered in all delivery context description formats. Therefore, the class 'extend' can be ignored for this kind of applications.

4.3 Additional Mapping Rules for Coding Formats

This section specifically discusses means for describing supported coding formats as this seem to be an inherent part of each delivery context description standard. Unfortunately, the standards in question adopt different technologies for describing this property. In particular, the CC/PP and, thus, UAProf adopts an approach which is based on MIME media types [FB96] whereas MPEG-21 UED relies on classification schemes introduced within MPEG-7 [MSS02].

MIME media types are well known within the Internet – thanks to its adoption for HTTP, etc. – and comprises five discrete top-level media types, i.e., text, image, audio, video, and application, as well as two composite top-level media types, i.e., multipart and message. These top-level media types are referred to as content types and the actual coding format is identified through the content sub-type (e.g, `video/mp4`). It is also possible to associate an arbitrary number of parameters in form of key-value pairs to media types which could be used to describe specifics usually defined within profiles/levels. However, most of the audio/video/image MIME type definition does not make use of this possibility. Thus, it is up to the application to identify the exact data format by other means. For example, `video/mp4` may contain a bitstream compliant to MPEG-4 Part 2 (Visual) or MPEG-4 Part 10 (Advanced Video Coding), not mentioning all the available profile/level combinations.

An MPEG classification scheme is an XML document that may contain terms – identifiable by URN – and corresponding definitions of arbitrary semantics in a hierarchically fashion. Thus, it is also possible to include profile/levels of a certain coding format as shown in Listing 4. Although classification schemes are extensible as they are based on XML there is a lack of an approved registration authority to accommodate future coding formats. However, the European Broadcasting Union (EBU) maintains a set of classification schemes used within their specifications (including TV-Anytime) [EBU09].

Listing 4. Excerpt of Visual Coding Format Classification Scheme MPEG-4 Visual Simple Profile.

```
<?xml version="1.0" encoding="UTF-8"?>
<ClassificationScheme
  uri="urn:mpeg:mpeg7:cs:VisualCodingFormatCS:2001">
  <!-- further terms -->
  <Term termID="3">
    <Name xml:lang="en">MPEG-4 Visual</Name>
    <Definition xml:lang="en">MPEG-4 Visual Coding Format</Definition>
    <Term termID="3.1">
      <Name xml:lang="en">MPEG-4 Visual Simple Profile</Name>
      <Term termID="3.1.1">
      <Name xml:lang="en">MPEG-4 Visual Simple Profile @ Level 0</Name>
    </Term>
    <Term termID="3.1.2">
      <Name xml:lang="en">MPEG-4 Visual Simple Profile @ Level 1</Name>
    </Term>
    <Term termID="3.1.3">
      <Name xml:lang="en">MPEG-4 Visual Simple Profile @ Level 2</Name>
    </Term>
    <Term termID="3.1.4">
      <Name xml:lang="en">MPEG-4 Visual Simple Profile @ Level 3</Name>
    </Term>
    <!-- further terms -->
  </Term>
  <!-- further terms -->
</ClassificationScheme>
```

Listing 5 describes the mapping from the MIME type image/jpeg to an equivalent URN representation which is used in UED. Line 1 of Listing 5 defines a resource for the MIME type `image/jpeg`. The other prefixes `csm`, `mit` and `owl` are shortcuts for the resources where the used vocabularies are defined (e.g., `owl` as shortcut for `http://www.w3.org/2002/07/owl#`). Line 2 defines the mapping from the resource `&uic;jpeg` to the resource `urn:mpeg:mpeg7:cs:VisualCodingFormatCS:2001:4` which represents JPEG as a reference to a classification scheme term. Line 3 defines which string representation for JPEG should be used in UAProf descriptionss. Lines 4 and 5 define all representations for the `image/jpeg` MIME media type. Line 6 uses standard OWL syntax to define that the resource `&uic;jpeg` is different from the resource `&uic;bitmap`.

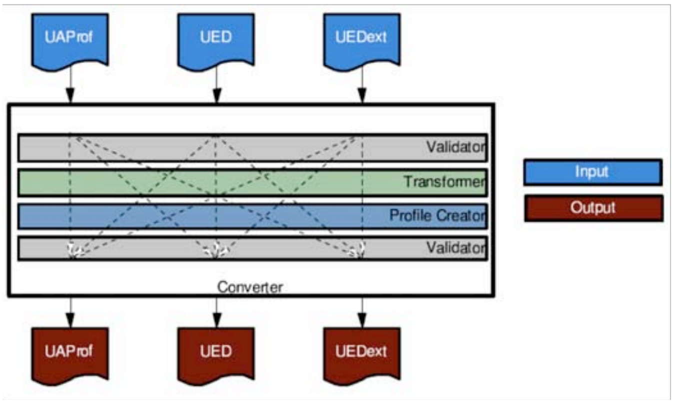
Listing 5. Mapping MIME media type `image/jpeg` to `urn:mpeg:mpeg7:cs:VisualCodingFormatCS:2001:4`.

```
<uic:KluItecImage rdf:about="&uic;jpeg">

  <csm:mapsToMpegResource
    rdf:resource="urn:mpeg:mpeg7:cs:VisualCodingFormatCS:2001:4"/>

  <mit:hasMimeType
    rdf:datatype="&xsd;string">image/jpeg</mit:hasMimeType >
  <mit:hasMimeRepresentation
    rdf:datatype="&xsd;string">image/jpeg</mit:hasMimeRepresentation >
  <mit:hasMimeRepresentation
    rdf:datatype="&xsd;string ">image/jpg</mit:hasMimeRepresentation >

  <owl:differentFrom rdf:resource="&uic;bitmap "/>
  <!-- ... -->
</ uic:KluItecImage >
```

5 Implementation Details

The aim of this section is to provide an overview of our implementation that currently performs a mapping between UAProf descriptions and UEDs (and vice-versa). The high-level architecture is depicted in Figure 3 and comprises three components:

- — Validator.
- — Transformer.
- — Profile Creator.

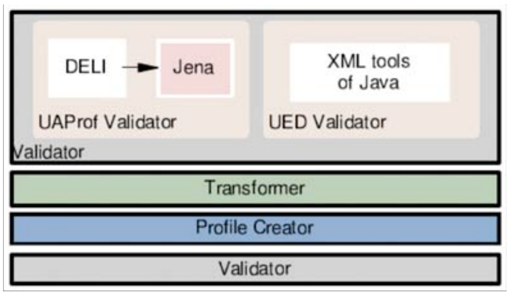


UAProf	…	User Agent Profile
UED	…	Usage Environment Description
UEDext	…	Usage Environment Description extensions

Figure 3. High-Level Architecture of the UAProf/UED Mapping Implementation.

The *Validator* is responsible for validating incoming and outgoing UAProfs and UEDs. If the received profile is a UED profile, a transformation to an RDF/XML document is needed for further processing which the *Transformer* accomplishes. UAProfs need not be translated because they are already written in RDF/XML syntax which is a requirement of the profile creator. The *Profile Creator* queries data from the profile data which is available in a consistent syntax and creates the desired profile as output which is again checked by the validator before it is delivered. These three components are further detailed in the following.

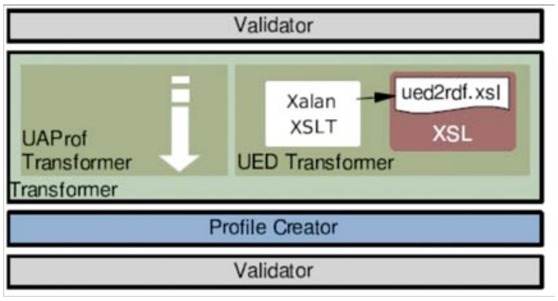
Validator (cf. Figure 4). The purpose of this component is to validate instances against its specification. This is performed both for inputs and outputs of our implementation. For UAProf we have integrated the DElivery context LIbrary (DELI) [Bu08] which is one of some rarely available tools that are able to validate UAProfs and to extract data from these documents. As the UED schema is an XML schema we have used standard XML schema validation tools such as the Java built-in XML validation Application Programming Interface (API).



DELI	… A Delivery Context Library For CC/PP and UAProf
Jena	… A Semantic Web Framework for Java
UAProf	… User Agent Profile
UED	… Usage Environment Description

Figure 4. Architecture of the Validator.

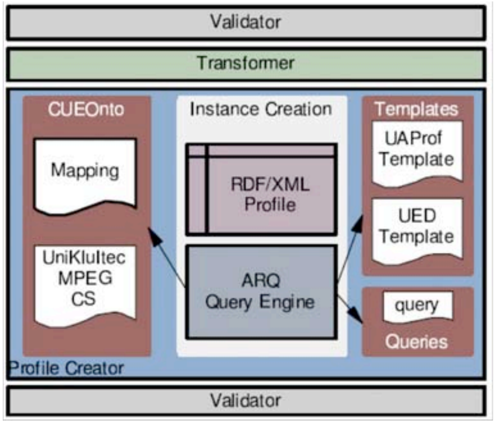
Transformer (cf. Figure 5). The transformer is responsible for translating the input instances into an integration model based on RDF as already introduced in Section 4. Therefore, we have implemented style sheets based on the Extensible Stylesheet Language Transformation (XSLT) [Cl99], i.e., only one style sheet is required for each delivery context description language keeping the overall approach scalable.



UAProf	…	User Agent Profile
XSLT	…	Extensible Stylesheet Language for Transformations
XSL	…	Extensible Stylesheet Language
UED	…	Usage Environment Description

Figure 5. Architecture of the Transformer.

Profile Creator (cf. Figure 6). Finally, this component generates the designated target delivery context description based on the integration model. In order to query the RDF-based integration model we have used SPARQL Protocol And RDF Query Language (SPARQL) [PS08] and A SPARQL Processor for Jena (ARQ) [HP08] as the actual query engine. The implementation adopts predefined templates and queries to generate desired output format based on the integration model.



UAProf	…	User Agent Profile
UED	…	Usage Environment Description
RDF	…	Resource Description Framework
ARQ	…	A SPARQL Processor for Jena

Figure 6. Architecture of the Profile Creator.

6 Conclusions and Future Work

In this paper we have presented a model that allows one to map context delivery descriptions between different formats (e.g., OMA UAProf and MPEG-21 UED) that are based on different technologies (i.e., XML Schema and RDF/OWL). For this model we have investigated state-of-the-art terminals in terms hardware and software capabilities as well as analyzed and compared existing delivery context description formats. Based on this analysis and comparison we concluded that there is a need for describing the commonalities and relationships between these description formats using a common model, i.e., following the integration model approach introduced earlier. The mapping model clusters the properties of the individual description formats based on their levels into four classes, namely *direct*, *advance*, *derive*, and *extend*. Based on these classes we have defined the integration model and formulated templates (i.e., using SPARQL and OWL) to query information from the integration model in order to generate the target context delivery format. The feasibility of the approach has been validated through a prototype and implementation details have been described in this paper.

The major findings can be summarized as follows:

- The overlap between different context delivery description formats is not that huge as expected but is clustered around those properties which are considered by the majority of applications areas (e.g., screen size, coding formats, etc.).
- Hence, the classes *direct*, *advance*, and *derive* are sufficient for most of the application areas.
- The relationship between different delivery context description formats needs to be described manually with respect to an integration model (i.e., the mapping function) and requires a thorough analysis of these formats which is sometimes cumbersome (cf. also [HL01]). For each format the mapping functions need to be defined only once with respect to an integration model.
- However, in this paper we have demonstrated that it is feasible – in principle – but requires the integration of many XML-based technologies ranging from XML Schema and RDF to SPARQL and OWL.

The following items are to be considered for future work. The integration of an OWL reasoner may be used to automatically recognize related data and extract specific information by using inference (e.g., mapping between different versions or slight syntax variations). Another future work item is a more detailed investigation of W3C's Delivery Context Ontology (DCO) and whether it can be used as the basis for the integration model for both UED and UAProf. Finally, as the newest description format under development (i.e, DCO) is based on OWL it confirms our decision to use OWL as underlying technology for the integration model.

7 References

[Ax08] IST-FP6: AXMEDIS (Automating Production of Cross Media Content for Multi-channel Distribution), http://www.axmedis.org/ (last accessed: December 2008).

[Bu08] Butler, M.: DELI: A Delivery Context Library For CC/PP and UAProf, October 2006. http://delicon.sourceforge.net/ (last accessed: December 2008).

[Cl99] Clark, J.; (ed.): XSL Transformations (XSLT) Version 1.0, W3C Recommendation, November 1999. http://www.w3.org/TR/1999/REC-xslt-19991116 (last accessed: December 2008).

[Da06] IST-FP6: DANAE (Dynamic and Distributed Adaptation of scalable multimedia coNtent in a context-Aware Environment), http://danae.rd.francetelecom.com/ (last accessed: December 2008).

[EBU09] EBU Metadata Classification Scheme Mainenance Form, http://www.ebu.ch/metadata/maintenanceforms/index_cs.html (last accessed 2009).

[En08] IST-FP6: ENTHRONE (End-to-End QoS through Integrated Management of Content, Networks and Terminals), http://www.ist-enthrone.org/ (last accessed 2008).

[FB96] Freed, N.; Borenstein, N; (eds.): Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types, RFC 2064, November 1996.

[GLS06] Gimson, R.; Lewis, R.; Sathish, S.; (eds.): Delivery Context Overview for Device Independence, W3C Working Group Note, March 2006. http://www.w3.org/TR/di-dco/ (Last accessed: December 2008).

[H01] Hunter, J.: Adding Multimedia to the Semantic Web - Building an MPEG-7 Ontology, Proceedings of the International Semantic Web Working Symposium (SWWS), Stanford, August 2001; pp. 261-281.

[HL01] Hunter, J.; Lagoze, C.: Combining RDF and XML Schemas to Enhance Interoperability Between Metadata Application Profiles, Proceedings of the 10th World Wide Web Conference, Hong Kong, May 2001.

[HP08] HP Laboratories: ARQ - A SPARQL Processor for Jena, http://jena.sourceforge.net/ARQ/ (last accessed: December 2008).

[Ki07] Kiss, C.; (ed.): Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 2.0, W3C Working Draft, April 2007. http://www.w3.org/TR/CCPP-struct-vocab2/ (last accessed: December 2008).

[Kl04] Klyne G.; et.al. (eds.): Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0, W3C Recommendation, January 2004. http://www.w3.org/TR/CCPP-struct-vocab/ (last accessed: December 2008).

[LF08] Lewis, R.; Fonseca, J.M.C.; (eds.): Delivery Context Ontology, W3C Working Draft, April 2008. http://www.w3.org/TR/dcontology/ (last accessed: December 2008).

[MSS02] Manjunath, B.S.; Salembier, P.; Sikora, T.; (eds.): Introduction to MPEG-7: Multimedia Content Description Interface, Wiley & Sons, April 2002.

[Mv04] McGuinness, D.L.; van Harmelen, F.; (eds.): OWL Web Ontology Language Overview, W3C Recommendation, February 2004. http://www.w3.org/TR/owl-features/ (last accessed: December 2008).

[OM06] Open Mobile Alliance Ltd: User Agent Profile, Approved Version 2.0. Technical report, OMA, February 2006.

[PS08] Prud'hommeaux, E.; Seaborne, A.: SPARQL Query Language for RDF, W3C Recommendation, January 2008. http://www.w3.org/TR/rdf-sparql-query/ (last accessed: December 2008).

[VT05] Vetro, A.; Timmerer, C.: Digital Item Adaptation: Overview of Standardization and Research Activities, IEEE Transactions on Multimedia, vol. 7, no. 3, June 2005; pp. 418-426.

[W308] w3development.de: UAProf profile repository. http://w3development.de/rdf/uaprof repository/ (last accessed: December 2008).

8 Appendix

Table 2. HW/SW Characteristics of Desktop PC/Workstation, Notebook/Tablet PC, Sub-Notebook/Netbook, Handheld, Smartphone, and Mobile Phone

	Desktop PC/Workstation	**Notebook/Tablet PC**	**Sub-Notebook/Netbook**	**Handheld**	**Smart Phone**	**Mobile Phone**
Performance	High performance with dual, triple or quad core processors	Power saving dual core processor	Medium performance special power saving processors	*Medium and low performance with various processors*		*Low performance with various processors*
Display	17" - 30", multiple displays	12" - 20"	7" - 12"	*4" (or smaller), color and monochrome*	*1,8"-3,5", color (and monochrome)*	*1,8"-3,5", color and monochrome*
Storage	Up to 2 TB	Up to 1 TB	2GB to 160GB	*8MB-256MB (also 16GB)*	*Up to 16GB (expandable, e.g., microSD-Cards)*	*Up to 64MB*
Memory	Up to 8 GB	Up to 4 GB	Up to 1 GB	*Similar to permanent storage*	*Often 128MB*	*Similar to permanent storage*
Network	Gigabit Ethernet, WLAN, cable/xDSL, etc.	Gigabit Ethernet, WLAN, Modem, Bluetooth, (UMTS, HSDPA)		*IrDA, Bluetooth, WLAN, GSM, GPRS, UMTS, HSDPA, GPS*	*IrDA, Bluetooth, USB, WLAN, GSM, GPRS, EDGE, UMTS, HSDPA, HSUPA, GPS, DVB-T, etc.*	*IrDA, Bluetooth, GSM, GPRS, UMTS*
Power	AC	AC, DC, battery 2-6h (typically 2,5h)		*DC, battery (3-15h)*	*Battery 100h with normal use (up to 400h Standby)*	
User I/O	Keyboard, Mouse, Monitor, Loudspeaker, Microphone, Webcam, etc.	Keyboard, Touchpad, LCD, Touchscreen and Stylus, Loudspeaker, Microphone, Webcam		*Touchscreen and Stylus, Loudspeaker, Microphone*	*Touchscreen, QUERTY-Keyboards (often very small), limited Keyboards, Loudspeaker, Microphone, Cameras*	*Phone Keyboard, (joystick or navigation buttons)*
Extensibility	USB, Firewire, PCIe, PCI, etc.	USB, Firewire, eSATA, ExpressCard, PCMCIA	USB	*CF and SD card slot for devices and memory, USB*	*USB, TV-out*	*Usually not supported*
OS	Windows Vista, Windows XP, Mac OS, Linux, etc.		Windows XP, special Linux versions, Windows Mobile	*Windows Mobile, Palm OS*	*Symbian OS, Windows Mobile, RIM Blackberrry, Apple OS X (special edition for iPhone), Google Android (Linux and Java based)*	*Symbian OS, other proprietary systems*
Software	Almost unlimited		Limited to Operating System and hardware capabilities	*Limited to Operating System and hardware capabilities*		*Preinstalled, (sometimes expandable through Java-Midlets)*

A Novel Tool for Quick Video Summarization using Keyframe Extraction Techniques

Mathias Lux*, Klaus Schöffmann*, Oge Marques+, Laszlo Böszörmenyi*

*Institute for Information Technology
Klagenfurt University
Universitätsstrasse 65-67
9020 Klagenfurt, Austria
{mlux, ks, laszlo}@itec.uni-klu.ac.at

+ Department of Computer Science and Engineering
Florida Atlantic University
777 Glades Road
Boca Raton, FL 33431 - USA
omarques@fau.edu

Abstract: The increasing availability of short, unstructured video clips on the Web has generated an unprecedented need to organize, index, annotate and retrieve video contents to make them useful to potential viewers. This paper presents a novel, simple, and easy-to-use tool to benchmark different low level features for video summarization based on keyframe extraction. Moreover, it shows the usefulness of the benchmarking tool by developing hypothesis for a chosen domain through an exploratory study. It discusses the results of exploratory studies involving users and their judgment of what makes the summary generated by the tool a good one.

1 Introduction

The explosion of video information available on the Web has generated an unprecedented need to organize, index, annotate and retrieve video contents to make them useful to potential viewers.

During the past few years, video has been promoted to a first-class data object in web-based applications. From a video consumer’s perspective it has become increasingly common to watch streaming video online, or download video programs for future playback on a wide variety of devices, from desktops to cell phones. It has never been easier to create video contents, encode it with a variety of standardized codecs, upload it, embed it into existing web pages or blogs and share it with the world at large.

The popularity of YouTube[1] (with an estimated number of 200,000 videos published every day) and its many competitors and copycats has exacerbated the need for effective ways to present the essence of a video clip without requiring that the user actually watch (part of) the video to know what it is about. After all, despite the fact that the average length of a video clip available on YouTube is only 2 minutes and 46.17 seconds, the time it would take to view all of the material on YouTube (as of March 17th 2008) is 412.3 years!

In summary, video is an inherently unwieldy medium whose contents need to be summarized in order to be truly useful to its potential viewers. Professional video summarization tools (e.g., Virage VideoLogger[2]) have been around for more than a decade and focus on large video footage repositories (e.g., from major TV news networks) and often benefit from the structure found in those programs. In this paper we present a novel, simple, and easy-to-use tool for the benchmarking of low level features for video summarization based on keyframe extraction. We focus on the process for the generation of a number of still images from video frames, which describe the video in an optimal way, while leaving out frames with low relevance for a summary.

The remainder of the paper describes the details of the algorithms and features used in the current version of the tool. Moreover, we discuss the results of exploratory studies involving users and their judgment of what makes a summary a good one to show the applicability of our tool for hypothesis development and benchmarking.

The tool described in this paper is publicly available to video processing researchers willing to experiment with several parameters and features used in the keyframe extraction algorithm. The open source nature of the tool makes it possible to modify and expand it to one's needs or desires. The simple and intuitive summaries provided by the tool make it possible to obtain a quick assessment of video summarization algorithms and parameters, in a way that is comparable to a basic content-based image retrieval (CBIR) system with query-by-example (QBE) functionality for testing image features and dissimilarity metrics (among other things) while fine-tuning an image retrieval solution.

[1] Statistics from http://ksudigg.wetpaint.com/page/YouTube+Statistics
[2] URI: http://publications.autonomy.com/pdfs/Virage/Datasheets/Virage%20VideoLogger.pdf

2 Related work

A *video summary* – or *video abstract* – is generally described as a series of still or moving images that represent the content of the video in such a way as to provide concise information about the video to the viewer [Pf96]. Many studies, surveys, and research papers on video summarization have been published during the last decade (e.g. [YL97], [KM98], [HZ99], [PC00], [ZC02], [XMX+03], [CS06], [MP08]). A recent comprehensive survey and review [TV07] alone contains more than 160 references! In spite of the significant amount of work in this field, the consensus is that "video abstraction is still largely in the research phase" [TV07].

Two basic forms of video summaries have been identified by Truong and Venkatesh [TV07]: *keyframes* and *video skims*. A *keyframe* is a *representative frame* for a video, also known as *R-frame*, *still-image abstract*, or *static storyboard*. A *video skim* is a *dynamic summary*, consisting of representative segments of the video, also known as *moving-image abstract* or *moving storyboard*. A very popular example of a video skim is a video trailer. The main advantage of video skims over keyframes is that the former can also communicate audio/speech and motion information while the latter are limited to static visual contents only. However, keyframe-based summarizations have the advantage that a user can immediately see the content of the summary, which can be a significant time-saving advantage in some situations, e.g., when browsing through a large video archive (e.g., YouTube). In such cases, a static preview showing keyframes of the video content can help the user to quickly identify a video of interest, while video skims would require the user to sequentially watch them. Although early studies [KM98] have suggested that users prefer static keyframes to dynamic summaries, the issue is far from settled, and more recent studies [TV07] have concluded that the optimal visualization of the summarized content remains an open question and that research must put more emphasis on "viewer-pleasant" summaries. This need for simpler and more effective summaries has also been corroborated by Money and Agius [MA08] who complained about the lack of personalized video summaries, such as the ones presented in [MP08], and proposed that future research should concentrate on user-based sources of information in such a way that the effort for the user is kept minimal.

Given the current situation with this broad range of methods, tools for assessment of techniques for different domains and scenarios are needed. To the best of our knowledge a tool for benchmarking the effects of different low level features in keyframe selection has not been developed or discussed.

3 Keyframe selection benchmarking tool

The selection of keyframes for a video summary relies on information about the individual frames. In most cases, low-level features such as color histograms or texture features are used to compare keyframes to one another. Our benchmarking tool was motivated by the idea that one could assess the appropriateness and quality of video summaries with different combinations of low-level features and dissimilarity measures (used for pair wise comparison of frames based on the selected low-level features). To try and assess different combinations of low-level features and dissimilarities, an algorithm that works satisfactorily in many different feature spaces (spanned by the different features) is needed.

In our benchmarking tool keyframes are selected by using a clustering algorithm. All available frames are clustered with a fixed number of clusters (*n)*, whereas the number of clusters is equal to the number of selected keyframes. Since clustering is performed based on low-level features extracted from each frame, for appropriately chosen values of *n*, all frames within a cluster tend to be visually similar (and one of them can be selected as a representative keyframe for that cluster). For the sake of video summarization, we assume that one image per cluster describes a whole set of visually similar images. The actual size of a cluster can further be used to assess how much of a videos duration is actually covered by a cluster. Finally, the selected keyframes are visualized as a video summary. Just as with the low-level features, the actual composition of the video summary is defined in a modular way and can be easily changed and adjusted to the specific requirements of a domain, user group or evaluation strategy.

The video summarization approach implemented in the tool described in this paper consists of the following steps:

1. Extraction of global features and calculation of appropriate dissimilarity metrics (Section 2.1);
2. Clustering of frames (Section 2.2);
3. Composition of the summary image (Section 2.3).

The benchmarking tool is written in Java, has been tested on Windows and Linux and is available online[3]. It features a graphical user interface as well as a command line interface for batch processing.

3.1 Low-level feature extraction and dissimilarity calculations

For the sake of keyframe selection, an uncompressed input video is interpreted as a sequence of still images. For each of the images (frames) within a video stream, we extract selected low-level features. The algorithms for low-level feature extraction included in this study were originally made available in an open source Java-based CBIR framework, LIRe [LC08]. Additional feature extraction methods can be easily integrated by implementing a simple Java interface. In the current implementation of the video summarization tool, we employed five different combinations of features and dissimilarity functions, namely those already existent in the underlying framework:

1. 64-bin RGB color histograms with L1 distance.
2. Tamura global texture features [TMY78].
3. Color and edge directivity descriptor [CB08a] with the Tanimoto coefficient.
4. Fuzzy color and texture histogram [CB08b] with the Tanimoto coefficient.
5. Auto color correlograms [Hu97] with L1 distance.

[3] URI: http://www.semanticmetadata.net

3.2 Clustering and keyframe selection

After indexing all frames we employ a clustering algorithm to assign each frame to one of *n* clusters (where *n* is a fixed number). The choice of a clustering algorithm is limited to those that rely on a distance (dissimilarity) measure without imposing (additional) requirements on the feature space. For the current implementation, the *k*-medoid clustering algorithm has been chosen, which is a very common partitioning clustering algorithm similar to *k*-means [JMF99]. The *k*-medoid approach is applicable to keyframe selection as it has been shown for instance in [HET06]. This approach has two main advantages for our application: (i) The cluster centre is always represented by a real data point and not an "artificial cluster centre" (which is the case with the *k*-means algorithm); and (ii) the clustering only depends on the dissimilarity function applied to the image feature vectors, and not the feature itself or the feature space.

The resulting *n* clusters group frames that are visually similar according to the chosen image feature. The clusters' medoids M_1, M_2, ... M_n minimize the distance to all elements of a clusters and are therefore interpreted as most descriptive elements for the respective groups. Furthermore, to allow a ranking of chosen keyframes relative to their ability to describe the content of the video, we introduce a relevance function for medoids M_k. The relevance $r(M_k)$ of the medoids M_k depends on the number of frames in cluster C_k. Consequently, the bigger a cluster, the more keyframes are in it and more of the video's duration is covered by the cluster. Therefore the medoid of the biggest cluster summarizes the largest part of the video. Also the medoid of the smallest clusters summarizes the smallest part of the video.

$$r(M_k) = \|C_k\|$$

3.3 Summary image composition

The final step of the video summarization method implemented in our tool is the visualization of the medoids, which are actual frames of the video. Our tool presents the video frames selected for the summarization as single still images. Additional data generated in the process, such as the size of the cluster and the actual frame number, are encoded in the file name. In addition to this basic output option we also added two different exemplary visualizations. A simple storyboard summary presents all found keyframes from left to right in sequence as shown in Fig. 1. The order of the sequence depends on the size of the medoid's respective clusters. The leftmost frame in the summary represents the biggest cluster.

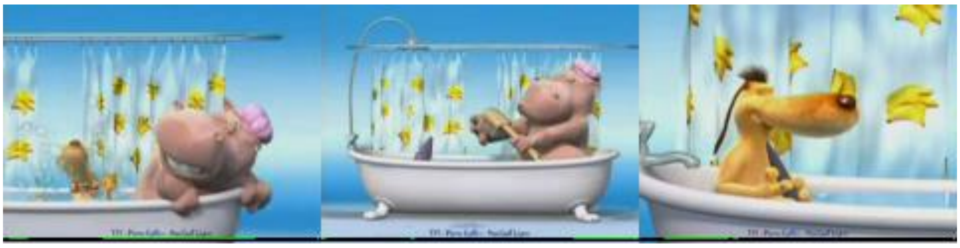


Figure 1 - Simple visualization where keyframes are shown in cluster-size order (medoid of the biggest clusters is the first frame to the left).

A second exemplary visualization shows the medoid frame of the largest cluster in full size and the other keyframes in smaller size (see Fig. 2). This visualization reduces the overall width of the summary. Since this visualization was used in the evaluation of the tool, it is explained in more detail in Section 3.

4 Exploratory study

In this section we describe an exploratory evaluation performed on a small user group, whose goal was to gain insight on the impact of different low-level descriptors and dissimilarity metrics on the keyframe selection algorithm. We surveyed seven users on three different videos. To underline the applicability of the benchmarking tool for new domains the videos were taken from YouTube and selected from the overall most viewed animations (Table 1).

Title	**Length**	**Views[4] (~)**
Hippo bathing	30 s	360,000
The Room - Vancouver Film School (VFS)	194 s	350,000
Dinosaurs vault	49 s	493,000

Table 1 - Videos employed for exploratory study

For this study we presented summaries based on different descriptors and using different numbers of clusters. For visualization of the selected frames we chose the following image composition (Fig. 2): the medoid frame of the biggest cluster is visualized in full size (on the left of the figure), while the remaining frames (two in our example) are resampled to a quarter of their size (half in width and height) and displayed on the right-hand side of the screen.

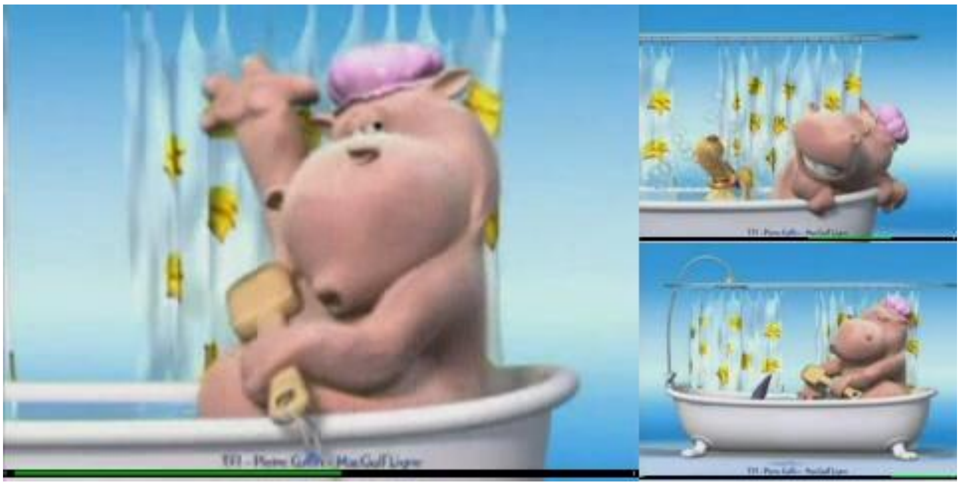


Figure 2 – Sample visualization of a summarization of the “Hippo bathing” video with the frame of the biggest cluster in full size to the left and the other two to the right.

[4] As of December 1, 2008

Another (subtle) feature of the proposed summary image composition scheme is the ability to visualize the distribution of cluster members over the video timeline. The last row in each keyframe represents the occurrence of frames in the respective cluster – marked with green pixels – along the time axis. A sample cluster distribution can be seen in a zoomed view on the part marked with the dotted line in Fig. 3. Assuming that the whole width of the larger frame in Fig. 3 represents the timeline of the entire video, we can see that the majority of cluster members in this case concentrate in the first half of the video.

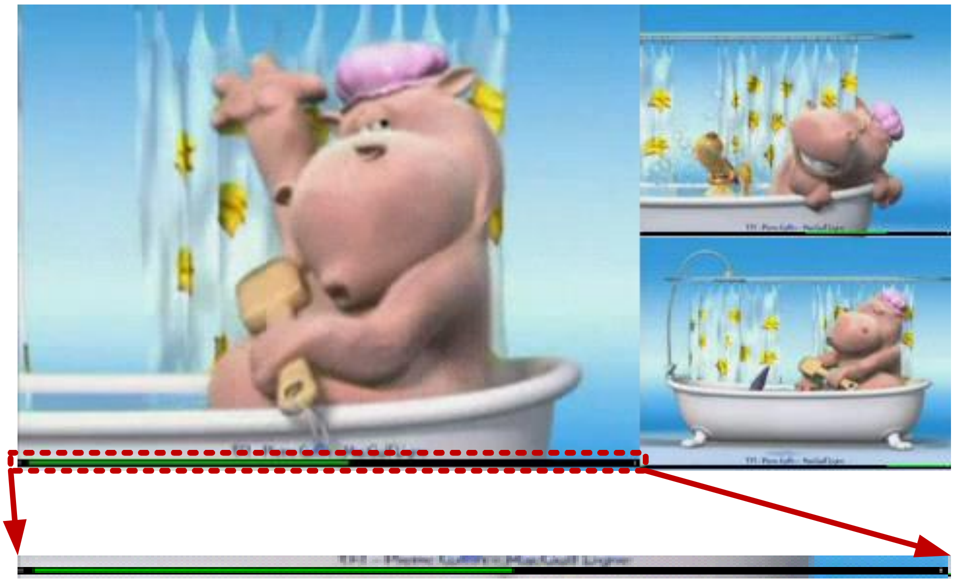


Figure 3 – Distribution of cluster members within the visualization of the selected frame. Green dots (lines) of the black bar (zoomed in from the area indicated with the dotted rectangle) show where cluster members are located in the video timeline.

Two main parameters have been varied for the study: number of clusters (*n*) and feature/dissimilarity metrics combinations. The case where *n*=1 has been omitted due to its triviality and the case where *n*=2 has been omitted due to disappointing results in a first exploratory investigation. Based on the selected visualization metaphor we wanted to study if users preferred 3 still images (one big and 2 small) or 5 still images (one big 4 small). Also we wanted to find out whether a visualization with 3 still images should be generated based on 3 or 4 clusters. We investigated:

- *n*=3 with a visualization displaying all three medoids,

- *n*=4 displaying only the three most relevant medoids and
- *n*=5 displaying all five medoids.

Note that the selected visualization metaphor features an odd number of images, so we did not test with 4 clusters showing all 4 keyframes. Furthermore for each *n* and video under consideration we created five different video summaries with different feature and dissimilarity combinations as mentioned in Section 2.2. This results in a set of 15 video summaries to assess per video.

The participants were experienced computer users, who use YouTube on a regular basis (at least once a week) and the computer on a daily basis. The survey group consisted of three female and four male participants, with ages ranging from 15 to 30 years old. For each participant the survey took place in a single session, where only the participant and the moderator (the same for each test) were present. For each video the moderator showed the actual video first. Then three groups of summaries were presented: (i) the group of summaries generated with *n*=3, (ii) the group of summaries generated with *n*=4 and (iii) the group of summaries with *n*=5. Each of the groups consisted of five different summaries generated based on the five before mentioned low-level features. The participant had to choose the best summary out of each group and had to rank the three chosen summaries according to their descriptiveness for the video. In addition to selection and ranking the moderator further asked the participant *why* the specific summary was chosen and *which criteria* were used to assess the ranking.

4.1 Results

Out of the 63 chosen images (three images per video with three videos per participant) there is no clear winner in terms of low-level features although one of the features (namely, color histogram) has been chosen the most times in absolute terms, as it can be seen in Figure 4. The visualization based on the color histogram feature has been chosen 19 times as most appropriate video summary followed by the auto color correlogram (ACC, 12 times), the fuzzy color and texture histogram (FCTH, 12 times), the color and edge directivity descriptor (CEDD, 11 times) and the Tamura global texture descriptor (9 times). Table 2 shows how often participants have picked a specific feature for different values of *n*.

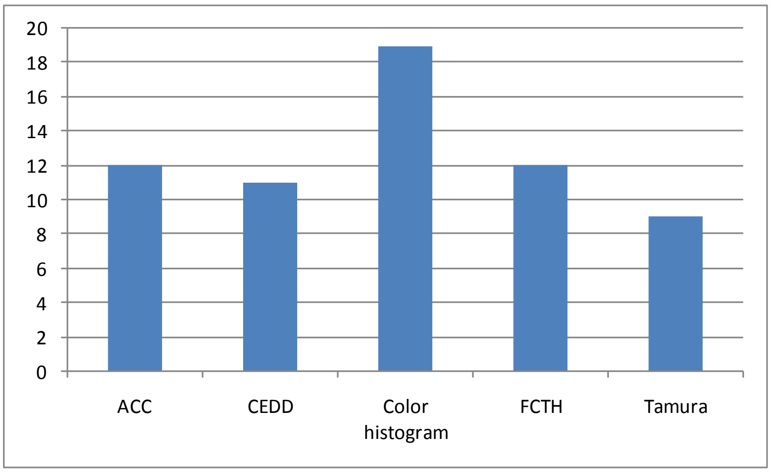


Figure 4 – Low-level features used for keyframe selection and a visualization of how often they have been selected.

	n=3	***n=4***	***n=5***
ACC	5	4	3
CEDD	8	0	3
Color Histogram	5	7	7
FCTH	0	8	4
Tamura	3	2	4

Table 2 - Selected features for different values of *n*.

From Table 2 one can see that the type of chosen features heavily depends on the chosen *n*. An example is the CEDD feature, which performs well on *n*=3 but has not been chosen at all for *n*=4. Table 3 however also indicates that the preference for low-level features changes with different videos. CEDD was mostly selected for the *Dinosaurs* video while FCTH was mainly used for the other two.

	Hippo	**Room**	**Dino**
ACC	7	2	3
CEDD	2	2	7
Color Histogram	3	8	8
FCTH	6	5	1
Tamura	3	4	2

Table 3 - Selected features for specific videos.

When asked to rank the three selected video summaries, the users ranked first the *n*=5 video summary (13 times), followed by the *n*=4 video summary (6 times) and the *n*=3 video summary (2 times). Most users voted for the 5-cluster-based summary because more of the video was captured in the more extensive summary (5 frames compared to 3 in the other two approaches).

4.2 Identified hypotheses

Based on the results of the exploratory evaluation we state three different hypotheses. Note that these hypothesis are based on the observations of the exploratory study and are intended for a detailed future study. Note also that these hypotheses are highly domain-dependent and may not hold for more general use cases. This shouldn’t come as a surprise, though, since it is widely acknowledged in the multimedia research community that state-of-the-art solutions for common problems (among them, summarization and content-based retrieval) are limited to narrow domains. Moreover, by allowing users to quickly and effectively experiment with different algorithms and fine-tune their parameters, our tool makes it easier to pursue further work within a domain of choice.

The first and main hypothesis H1 is: *There is a combination of low-level feature and dissimilarity metric that performs best for the sake of keyframe selection.* Once such combination is found for a certain dataset, it may lead to subsequent improvements and optimizations. Note that the selected features and dissimilarity metrics can be quite different from the ones listed in Section 2.1 and might include specialized features and metrics that are more suitable to the chosen domain.

Moreover, due to the users' preference for the *n*=4 approach (where three images are shown and the smallest cluster is discarded in the visualization) over the *n*=3 approach, an interesting hypothesis H2 is: *Users prefer summaries where the medoid of the smallest cluster is not shown.* This hypothesis would support the idea of a "junk cluster", where unimportant or low quality frames are grouped together.

Finally, based on the qualitative feedback of the participants we can also postulate hypothesis H3: *There is an optimal number X of frames to be displayed within a video summary which is enough to cover the content of the video but still not too many to be investigated by the user in a short time.* The proposed tool allows the experimental determination of the best value of *X* for a certain domain in an easy way.

5 Conclusions

We have presented a tool for benchmarking different combinations of low-level features and dissimilarity metrics for video summaries based on keyframe selection. In an exploratory study we have shown the applicability of our tool and we further found that varying the number of clusters and the choice of low-level features and dissimilarity metrics used for analysis provides frame selection results that are different enough to be used as input to user satisfaction studies. The feedback received from this exploratory evaluation led us to identify three promising hypotheses to be investigated in future domain-specific evaluations. These hypotheses suggest additional research on the issues of low-level feature and dissimilarity combinations, the optimal number of images displayed within the video summary, and the relationship between the number of clusters and the number of images displayed in the video summary.

References

[CB08a] Chatzichristofis, S. A. & Boutalis, Y. S.: (CEDD: Color and Edge Directivity Descriptor. A Compact Descriptor for Image Indexing and Retrieval, in A. Gasteratos; M. Vincze & J.K. Tsotsos, ed.,'Proceedings of the 6th International Conference on Computer Vision Systems, ICVS 2008', Springer, Santorini, Greece, pp. 312-322, 2008

[CB08b] Chatzichristofis, S. A. & Boutalis, Y. S.: FCTH: Fuzzy Color And Texture Histogram A Low Level Feature For Accurate Image Retrieval, in 'Proceedings of the 9th International Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS 2008', IEEE, Klagenfurt, Austria, pp. 191-196.

[CS06] Ciocca, G. & Schettini, S.: An innovative algorithm for key frame extraction in video summarization. Journal of Real-Time Image Processing, 1(1) pp. 69-88, 2006

[HET06] Hadi, Y.; Essannouni, F. & Thami, R. O. H.: Video summarization by k-medoid clustering, in 'SAC '06: Proceedings of the 2006 ACM symposium on Applied computing', ACM, New York, NY, USA, pp. 1400-1401.

[HZ99] Hanjalic, A. & Zhang, H.: An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis, IEEE Transactions on Circuits and Systems for Video Technology, 1999 9(8), 1280--1289.

[Hu97] Huang, J.; Kumar, S. R.; Mitra, M.; Zhu, W.-J. & Zabih, R.:: Image Indexing Using Color Correlograms, in Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition, CVPR 1997, IEEE, San Juan, Puerto Rico, pp. 762-768.

[JMF99] Jain, A. K.; Murty, M. N. & Flynn, P. J. (1999), 'Data clustering: a review', ACM Comput. Surv. 31(3), 264--323.

[KM98] Komlodi, A. & Marchionini, G. (1998), Key frame preview techniques for video browsing. In DL: Proceeding of the 3rd ACM Conference on Digital Libraries. ACM Press, New York. 118-125

[LC08] Lux, M. & Chatzichristofis, S. A. (2008), Lire: lucene image retrieval: an extensible java CBIR library, in 'MM '08: Proceeding of the 16th ACM international conference on Multimedia', ACM, New York, NY, USA, pp. 1085--1088.

[MA08] Money, A.G. & Agius, H.: Video summarisation: A conceptual framework and survey of the state of the art, 2008

[MP08] Matos, N. & Pereira, F.:. Using MPEG-7 for Generic Audiovisual Content Automatic Summarization. In Image Analysis for Multimedia Interactive Services, 2008. WIAMIS'08. Ninth International Workshop on, pp. 41-45

[PC00] Parshin & Chen, L.: Implementation and analysis of several keyframe-based browsing interfaces to digital video. Lecture Notes on Computer Science, 2000, vol. 1923, pp. 206

[Pf96] Pfeiffer, S.; Lienhart, R.; Fischer, S. & Effelsberg, W.: Abstracting Digital Movies Automatically, University of Mannheim, 1996

[TMY78] Tamura, H.; Mori, S. & Yamawaki, T.: Textural Features Corresponding to Visual Perception', IEEE Transactions on Systems, Man, and Cybernetics 8(6), 1978, pp. 460-472.

[TV07] Truong, B. T. & Venkatesh, S.: Video Abstraction: A Systematic Review and Classification, in ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP) , 2007, Vol. 3, No. 1, Article 3

[XMX+03] Xu, M. & Maddage, N.C. & Xu, C. & Kankanhalli, M. & Tian, Q.:. Creating audio keywords for event detection in soccer video. In Multimedia and Expo, 2003. ICME'03. Proceedings. 2003 International Conference on, vol. 2

[YL97] Yeung, M. M. & Leo, B. L.: Video visualization for compact representation and fast browsing of pictorial content. IEEE Trans. Circ. Syst. Video Technol.1997, 7, 5

[ZC02] Zhang, D. & Chang, S. F.: Event detection in baseball video using superimposed caption recognition. In Proceedings of the tenth ACM international conference on Multimedia 2002. ACM New York, NY, USA, pp. 315-318

[Zh98] Zhuang, Y.; Rui, Y.; Huang, T. & Mehrotra, S.: Adaptive key frame extraction using unsupervised clustering, in 'Proceedings of the 1998 International Conference on Image Processing. ICIP 98.', pp. 866--870.

Authoring Interactive Mobile Services Using MPEG-7 and MPEG-4 LASeR

Albert Hofmann, Andreas Kriechbaum, Werner Bailer

Institute of Information Systems & Information Management
JOANNEUM RESEARCH Forschungsgesellschaft mbH
Steyrergasse 17, 8010 Graz, Austria
{firstName.lastName}@joanneum.at

Abstract: Mobile TV is becoming increasingly popular. As hand held devices are personal and used in a "lean-forward" mode, mobile devices are well suited for interactive TV. The porTiVity system provides authoring tools for mobile interactive content that allow to attach interactivity to moving objects in the video. We describe automatic and semi-automatic annotation tools that create content descriptions represented as MPEG-7 documents. The descriptions of moving objects are then transformed into MPEG-4 LASeR, which is a suitable lightweight format for visualisation and interactivity on the mobile device. We show the use of these authoring components in porTiVity in two example scenarios.

1 Introduction

1.1 Motivation

With the introduction of the mobile broadcast standard DVB-H [DVB04] mobile TV is becoming increasingly popular. The number of mobile TV users is increasing at an enormous rate and consumers are showing their interest to enjoy innovative mobile applications. Market analysis predicts growth rates of 100% and more[1] for the number of users in the coming years and within the last year the average daily time a user spent watching mobile TV has nearly doubled[2]. Mobile interactive TV can be one of these emerging applications. Results from previous interactive TV projects such as GMF4iTV [Car05] show that hand held devices are better suited for interactivity than classic TV sets as they are personal and used in "lean-forward" mode.

This paper presents results from the porTiVity project [Por] which aims to provide interaction with objects in video shown on mobile devices. The interactive experience for the end-user is achieved by simply clicking on highlighted objects overlaid on the video on the touch screen of the mobile device. This interaction triggers e.g. downloading of additional content from the web, participating in a quiz, in mobile network based games

[1] http://www.3gnewsroom.com/3g_news/apr_05/news_5811.shtml
[2] http://futurezone.orf.at/stories/1500388/

or purchasing a chosen product. In a live scenario such as a football game interaction on the soccer players or via menu buttons allows to retrieve content from a player's website, game statistics, watching replays of interesting events etc.

The efficient semantic annotation of audiovisual content is an important part of this workflow. MPEG-4 LASeR annotation tools supporting moving objects and interaction with those objects were not available when the project started. Relevant annotation tools for audiovisual content are among others M-OntoMat-Annotizer[3] developed by the aceMedia project or the IBM VideoAnnEx Tool[4]. However, these tools lack object redetection feature and automatic object tracking functionality.

We describe in this paper an authoring system for creating interactive TV content with interactivity based on moving objects. Section 2 describes the automatic and semi-automatic annotation of content described using MPEG-7 [MPE01]. In Section 3 we discuss how this description is transformed into a suitable representation for the mobile device, namely MPEG-4 LASeR [LAS06]. The integration of these authoring steps into the porTiVity system and the end-user experience in two scenarios is described in Section 4. Section 5 concludes this paper.

1.2 Authoring for mobile interactive TV

The architecture of the end-to-end platform for providing rich media interactive TV services for portable and mobile devices developed in the porTiVity project is shown in Figure 1 and described in Section 4 in more detail. In the authoring phase the main video content is analysed and annotated to link it to additional content (images, web pages, etc.) and the possible user interactions are defined. The tools support workflows for both offline and online scenarios. The result of the authoring phase is a scene description in MPEG-4 LASeR which is multiplexed into an MXF [MXF04] stream together with the main and additional content. MPEG-4 LASeR is a standard for rich media scene description on mobile and embedded devices. In our work it is used to link additional content to the main video stream which can be simple text, HTML pages, pictures, audio and video clips. In order to reduce the bandwidth and processing requirements it has been used as a more lightweight alternative to MHP [MHP07]. MXF is a container format defined by SMPTE that can hold various types of audiovisual essence as well as metadata, with capabilities for multiplexing.

In Figure 2 the workflow between the tools of the proposed Semantic Video Annotation Suite in a typical authoring chain and the metadata formats used on the interfaces are shown. Media-Analyze is an automatic video preprocessing tool (cf. Section 2.1). In the Semantic Video Annotation Tool (SVAT) the results from the automatic preprocessing are used to support the user in annotating the video by providing navigation aids, redetection and tracking of objects. The Interactive Application Tool (IAT) uses the MPEG-7 de-

[3] http://www.acemedia.org/aceMedia/results/software/m-ontomat-annotizer.html

[4] http://www.research.ibm.com/VideoAnnEx

scription of the annotated objects generated by the SVAT, transforms them into MPEG-4 LASeR objects, performs the LASeR segmentation and packs all the required data into an MXF file suitable for the playout system.

2 Content annotation

The content annotation is divided into two steps: an automatic preprocessing step done with the Media-Analyze tool and a semi-automatic annotation step performed with the Semantic Video Annotation Tool (SVAT). Both tools are bundled together in the Semantic Video Annotation Suite [SVA] and are described here in the context of authoring interactive mobile TV services. The output of both steps is described using MPEG-7, a comprehensive standard for multimedia content description covering both low-level and high-level description. In particular, the Detailed Audiovisual Profile (DAVP) [BS06], an MPEG-7 subset aimed at fine-grained description of audiovisual content in media production and archiving, is used.

2.1 Automatic preprocessing

In a first step a video file is automatically analysed by the Media-Analyze tool. We extract metadata for efficient navigation in videos and structuring the content. These are shot boundaries, key-frames, stripe images and camera motion.

In order to aid the user in efficient object annotation we extract SIFT descriptors [Low99] of interest points of key-frames and build a visual vocabulary [SZ03]. By doing this it is possible to redetect similar objects and shots in the same video in a very efficient way [NTM07, RBN^{+}07].

After identifying single occurrences by the user these occurrences need to be tracked within the same shot. Since this step is interactive it requires a very efficient tracking algorithm. For this purpose feature points and descriptors are extracted within this automatic preprocessing step [TM07, NTM07].

Both the visual vocabulary of SIFT descriptors and the descriptors for object tracking are stored in a proprietary binary format for efficiency reasons.

2.2 Semi-automatic object annotation

After this automatic annotation step objects of interest can be annotated with interaction. This is done with the Semantic Video Annotation Tool and its support for spatio-temporal object annotation.

As operators often wish to assign the same additional content to similar objects we provide

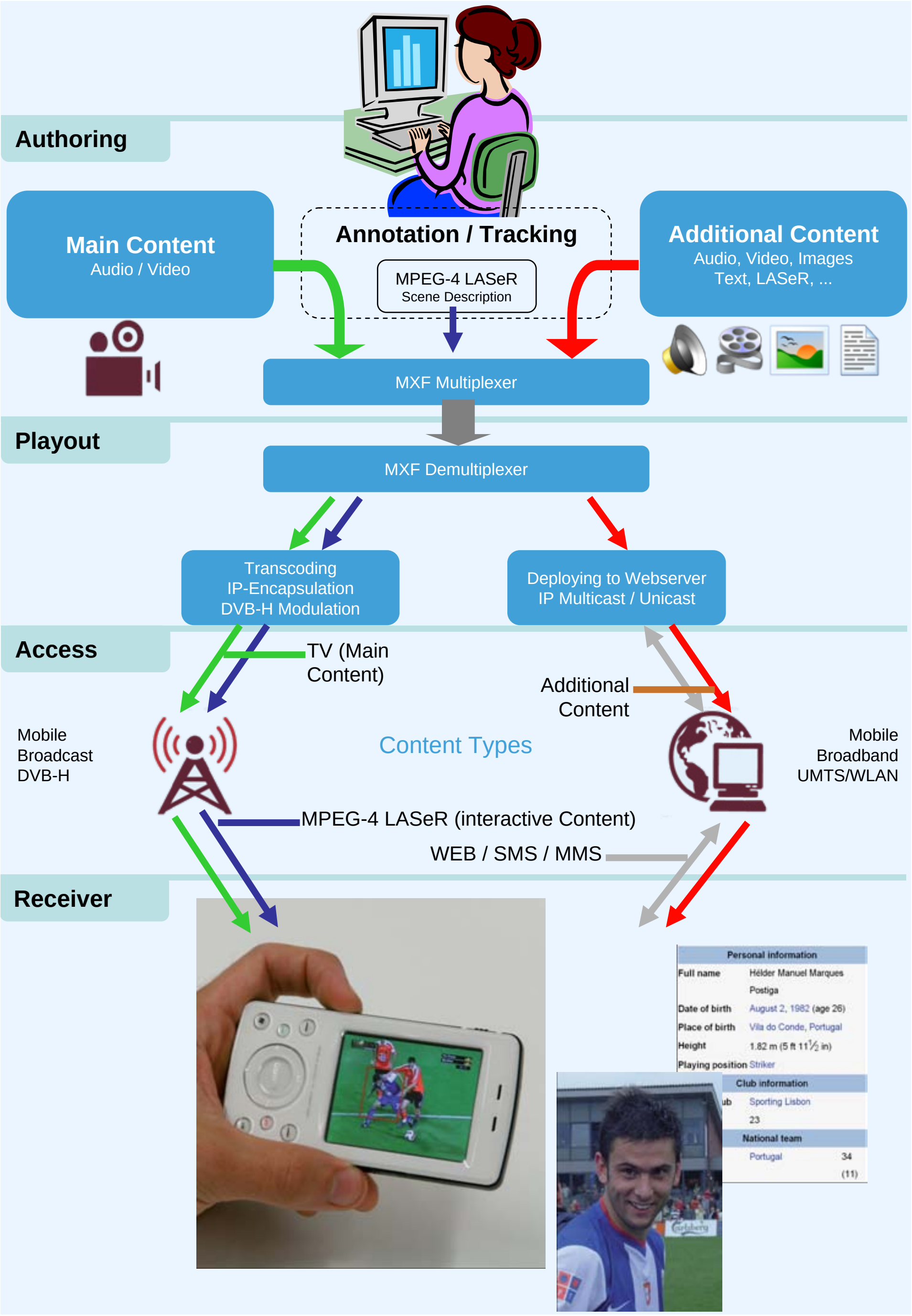


Figure 1: porTiVity system overview.

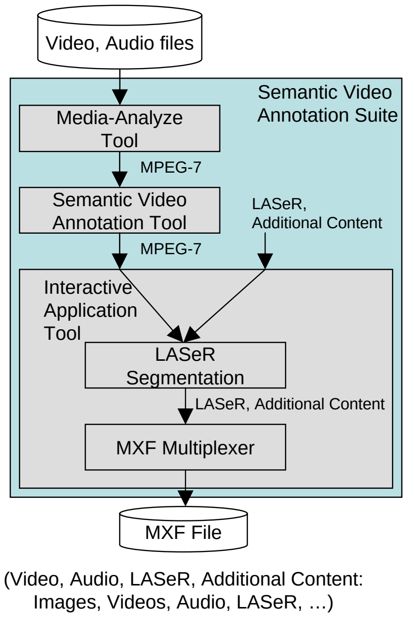

Video, Audio files
Semantic Video Annotation Suite
Media-Analyze Tool
MPEG-7
Semantic Video Annotation Tool
LASeR, Additional Content
MPEG-7
Interactive Application Tool
LASeR Segmentation
LASeR, Additional Content
MXF Multiplexer
MXF File
(Video, Audio, LASeR, Additional Content: Images, Videos, Audio, LASeR, ...)

Figure 2: Annotation chain in the Semantic Video Annotation Suite including exchanged metadata formats.

an object redetection feature. The input for the object redetection is a region of interest identified in a video player component. This can be done by simply marking a bounding box around the region, or by using a color segmentation functionality in order to get the exact border of the object. By using the visual vocabulary from the preprocessing step we can search for appearances of similar SIFT descriptors as extracted from the query region. For this application scenario higher recall is preferred since false positives in the result list can be easily deleted.

In the next step these single occurrences can be tracked over time within the boundaries of the shot. The input for the tracking algorithm can be a bounding box as well as a complex polygon around an object. The resulting moving region is always described as bounding box. The tool also supports manual tracking in case the automatic tracking fails. In this case the operator specifies key-regions over time that will be connected with a spline interpolation algorithm.

The output of the Semantic Video Annotation Tool is an MPEG-7 description which includes the metadata from the automatic analysis step enriched with the annotation of moving objects. An example of such a moving region is shown in Listing 1. The moving region shows an object with name *bottle* (*SemanticRef* element) and a bounding box that changes over time (*ParameterTrajectory* elements as defined by [MPE01, part 3]). The values inside the *Coords* element define the three points that build a closed bounding box.

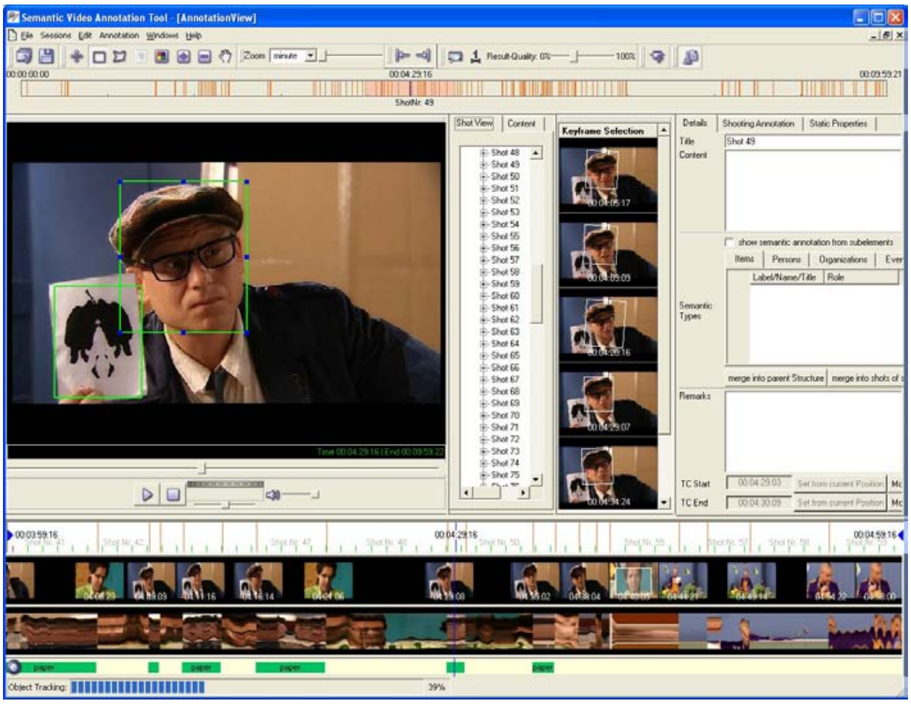

Semantic Video Annotation Tool - [AnnotationView]
File Sessions Edit Annotation Windows Help
Zoom minute
Result-Quality: 0% 100%
ShotNr: 49
Shot View Content
Keyframe Selection
Details Shooting Annotation Static Properties
Title Shot 49
Content
show semantic annotation from subelements
Items Persons Organizations
Label/Name/Title Role
Semantic Types
merge into parent Structure
Remarks
TC Start 00:04:29:03 Set from current Position
TC End 00:04:30:09 Set from current Position
paper
Object Tracking: 39%

Figure 3: Semantic Video Annotation Tool for object annotation.

3 MPEG-7 to MPEG-4 LASeR transformation

3.1 Transforming moving object descriptions

In this step the representation of moving regions described in MPEG-7 format are transformed to MPEG-4 LASeR, a suitable delivery format for the mobile device. MPEG-7 is intended for content description, but does not provide visualisation and interaction possibilities. In addition, a large part of the MPEG-7 description is not needed at the mobile device. We present an approach for transforming shape descriptions of moving objects (like polygons and rectangles) to MPEG-4 LASeR.

MPEG-4 LASeR is based on SVG[5] Tiny 1.2 and is thus very powerful regarding shape representation. We have focused on transforming polygons and simple bounding boxes to SVG taking different personalisation options for the end user in consideration. Currently three options are supported which are also reflected in the MPEG-4 LASeR snippet in Listing 2 and the images in Figure 4: a rectangle with solid or dashed line style, a contact point in the middle of the area and rectangle with only corner indicators.

[5] Scalable Vector Graphics (SVG) is a W3C recommendation for the XML based representation of 2D vector graphics (`http://www.w3.org/Graphics/SVG/`).

Listing 1: Moving object described in MPEG-7.

```
<MovingRegion>
  <SemanticRef idref="bottle" />
  <SpatioTemporalLocator>
    <ParameterTrajectory motionModel="still">
      <MediaTime>
        <MediaTimePoint>T00:01:23:1F25</MediaTimePoint>
        <MediaDuration>P0DT0H0M0S2N25F</MediaDuration>
      </MediaTime>
      <InitialRegion>
        <Polygon>
          <Coords dim="2␣4">421 96 0 -96 300 0 151 0</Coords>
        </Polygon>
      </InitialRegion>
    </ParameterTrajectory>
    <ParameterTrajectory motionModel="still">
      <MediaTime>
        <MediaTimePoint>T00:01:23:3F25</MediaTimePoint>
        <MediaDuration>P0DT0H0M0S1N25F</MediaDuration>
      </MediaTime>
      <InitialRegion>
        <Polygon>
          <Coords dim="2␣4">422 95 0 -95 301 0 151 0</Coords>
        </Polygon>
      </InitialRegion>
    </ParameterTrajectory>
    ...
  </SpatioTemporalLocator>
</MovingRegion>
```

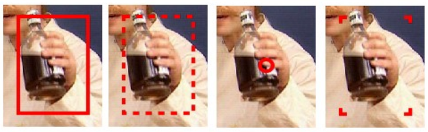


Figure 4: Styles for highlighting moving objects.

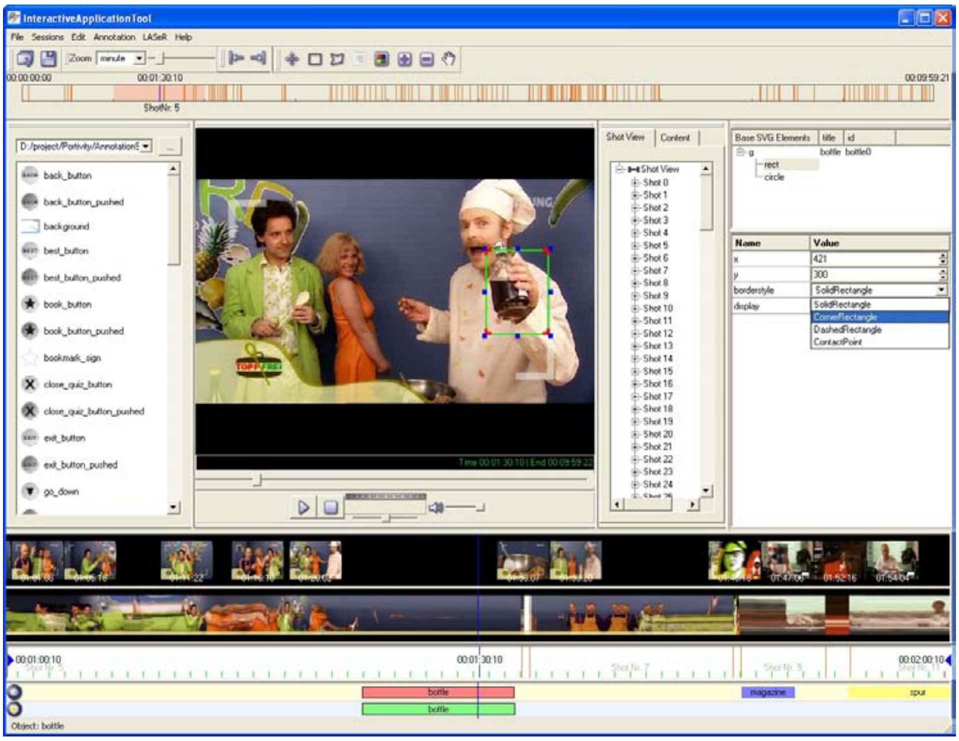
For the SVG description of the animation of the moving object over time we considered two options: specifying the coordinates in a fixed interval for the whole time range, or using a variable interval depending on the movement of the object and specifying those time points in the *keyTimes* attribute. Using a variable interval has advantages in terms of description size for moving objects that hardly move or keep their position for a longer period, but increases the complexity of the synchronised broadcast (see LASeR Segmentation in Section 3.2). So we have chosen the approach using a fixed interval (e.g. 0.2 sec) that can be configured by the user.

The result of the conversion from the moving region described in the MPEG-7 snippet in Listing 1 to MPEG-4 LASeR is shown in Listing 2. The id from the MPEG-7 description is reflected in the *title* element (unchanged) and also in the *id* attribute, taking into account that XML requires unique ids. Also when accessing specific elements later during MPEG-4 LASeR command processing we require unique ids when adding, modify or removing objects.

The graphic object is embedded inside the SVG document of the main MPEG-4 LASeR scene. In order to reduce the complexity of MPEG-4 LASeR handling within the authoring tools, no LASeR commands are used in this authoring step for time related updates of the scene. This is done in a separate post processing step – the LASeR segmentation – which is described in Section 3.2. The animation of objects over time is specified for the whole object time range. Also the visibility of the object is modeled by setting the SVG *display* attribute to 'none' and 'block' for the respective time points. So up to this step the produced MPEG-4 LASeR document just contains one scene with one SVG document containing the whole description for the whole video time range. Also no interaction possibilities are added up to this point.

Listing 2: Moving object described in MPEG-4 LASeR.

```
<g pointer-events="all" fill="none" stroke="red" id="bottle0" stroke-width="2">
  <title>bottle</title>
  <desc>Moving Object</desc>
  <rect width="96" x="421" y="300" height="151">
    <animate values="421;426;432;439;447;..."
      fill="freeze" begin="83.040s" dur="9.680s" attributeName="x" />
    <animate values="300;305;302;277;229;192;..."
      fill="freeze" begin="83.040s" dur="9.680s" attributeName="y" />
    <animate values="96;94;93;95;99;..."
      fill="freeze"begin="83.040s" dur="9.680s" attributeName="width" />
    <animate values="151;149;148;149;153;..."
      fill="freeze" begin="83.040s" dur="9.680s" attributeName="height" />
  </rect>
  <circle cx="469" cy="375" display="none" r="10">
    <animate values="375;379;376;351;305;..."
      fill="freeze" begin="83.040s" dur="9.680s" attributeName="cy" />
    <animate values="469;473;478;486;496;..."
      fill="freeze" begin="83.040s" dur="9.680s" attributeName="cx" />
  </circle>
  <g display="none">
    <desc>corner rectangle</desc>
    <polyline points="421,310␣421,300␣431,300">
      <animate values="421,310␣421,300␣431,300;426,315␣426,305␣436,305;..."
        fill="freeze" begin="83.040s" dur="9.680s" attributeName="points" />
    </polyline>
    <polyline points="507,300␣517,300␣517,310">
      <animate values="507,300␣517,300␣517,310;510,305␣520,305␣520,315;..."
        fill="freeze" begin="83.040s" dur="9.680s" attributeName="points" />
    </polyline>
    <polyline points="517,441␣517,451␣507,451">
      <animate values="517,441␣517,451␣507,451;520,444␣520,454␣510,454;..."
        fill="freeze" begin="83.040s" dur="9.680s" attributeName="points" />
    </polyline>
    <polyline points="431,451␣421,451␣421,441">
      <animate values="431,451␣421,451␣421,441;436,454␣426,454␣426,444;..."
        fill="freeze" begin="83.040s" dur="9.680s" attributeName="points" />
    </polyline>
  </g>
  <set begin="0s" attributeType="XML" to="none" attributeName="display" />
  <set begin="83.040s" attributeType="XML" to="block" attributeName="display" />
  <set begin="92.720s" attributeType="XML" to="none" attributeName="display" />
</g>
```



Base SVG Elements	title	id
g	bottle	bottle0
rect		
circle		

Name	Value
x	421
y	300
borderstyle	SolidRectangle
display	

SolidRectangle
CornerRectangle
DashedRectangle
ContactPoint

Figure 5: Interactive Application Tool for creating MPEG-4 LASeR scenes.

3.2 Adding interactivity

Next to the powerful shape representation capabilities of MPEG-4 LASeR, it provides support for interactivity to the end user. In our scenario we link additional content to moving objects like statistics and photos of soccer players, background information on actors etc. If the user clicks on a moving region this additional content should be displayed. Moreover, the additional content can be personalised, which means that different additional content items can be linked for different target groups (e.g. fans of the different teams in sports broadcasts, male and female viewers, . . .). We can edit these actions in the Interactive Application Tool (IAT, see Figure 5). The resulting MPEG-4 LASeR snippet after adding an additional content is shown in listing 3. For non personalised additional content we place an *a* element around the shapes for highlighting the moving object. For personalised additional content (not shown in the listing) we use a script function that evaluates the additional content link based on the local user profile and makes the respective HTTP request.

Before sending the MPEG-4 LASeR description to the playout system, a post processing step is needed to make the service ready for transmitting over a broadcast channel. Users can join the TV channel any time, so it is necessary to transmit the LASeR content period-

Listing 3: Interactivity described in MPEG-4 LASeR.

```
<g pointer-events="all" fill="none" stroke="red" id="bottle0" stroke-width="2">
  <title>bottle</title>
  <desc>Moving Object</desc>
  <a xlink:href="http://server.portivity.org/PoE1_a.xsr">
    <rect width="96" x="421" y="300" height="151">
      <!-- animation of rect -->
    </rect>
    <circle cx="469" cy="375" display="none" r="10">
      <!-- animation of circle -->
    </circle>
    <g display="none">
      <desc>corner rectangle</desc>
      <!-- polylines of corner rectangle -->
    </g>
        </a>
</g>
```

ically. This is required for the general scene description containing the static objects (like menus), while moving objects just need to be transmitted during the time they are visible. This is done by the LASeR segmentation module. It takes the whole LASeR scene from the Interactive Application Tool as input, and splits the objects into different access units. An example of how the MPEG-4 LASeR snippet presented in Listing 2 is segmented is shown in Listing 4:

Listing 4: LASeR Segmentation into Access Units.

```
<!-- Initial object description in the main SVG document: -->

<g pointer-events="all" fill="none" stroke="red" id="bottle0" stroke-width="2">
  <!-- shape representation cut here -->
  <set begin="0s" attributeType="XML" to="none" attributeName="display" />
  <set begin="83.040s" attributeType="XML" to="block" attributeName="display" />
  <set begin="92.720s" attributeType="XML" to="none" attributeName="display" />
</g>

<!-- After the segmentation into access units for insert and delete: -->

<saf:sceneUnit time="83040">
  <lsr:Insert ref="parent_id">
    <g id="bottle0" />
  </lsr:Insert>
</saf:sceneUnit>

<saf:sceneUnit time="92720">
  <lsr:Delete ref="bottle0"/>
</saf:sceneUnit>
```

Another task of this module is the periodic generation of RefreshScenes, which are access units containing a refresh of the scene as it should be at that precise moment. These access units are generated at a given interval – usually every two seconds – and they are ignored by the terminal if it already has this information (either from a NewScene or from a previous RefreshScene). These access units will contain the information on static and dynamic objects that become visible or invisible, and will also update the position of visible dynamic objects. The additional binary MPEG-4 LASeR stream only costs a small amount of bandwidth compared to the TV stream. An alternative system design to periodically poll a web server would lead to a traffic bottleneck due to the huge amount of requests to be handled by the web server.

The final step of the authoring is the MXF multiplexing. The video and audio content as well as all additional content files are packed into an MXF stream to ensure time synchronisation.

4 porTiVity system

A detailed description of the porTiVity system is given in [DFL$^+$08]. This section summarises the steps after the authoring as shown in Figure 1.

4.1 Playout system

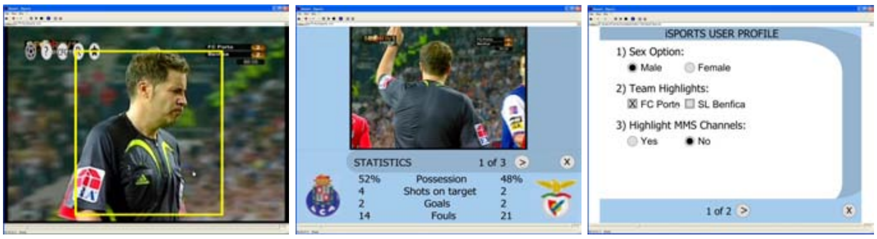
The playout system consists of several modules making the video stream and MPEG-4 LASeR content ready for transmission over DVB-H and the additional content accessible on a web server.

The MXF demultiplexer takes the produced MXF file from the authoring system as input, extracts all tracks and forwards them to the respective modules in the chain. The transcoder module converts the high-quality video and audio streams used in the production environment to the target formats on the mobile device: MPEG-4 AVC (H.264) and MPEG-4 AAC. Additional content files are also transcoded to the target format if necessary. The Metadata Processor Module receives LASeR access units from the MXF demultiplexer, encapsulates them over RTP (Real-Time Transport Protocol), and sends them to the DVB-H module responsible for broadcasting.

4.2 Receiver

The mobile device receives the DVB-H stream consisting of an audio, video and LASeR MPEG-4 elementary stream encapsulated in RTP. These streams are synchronised on the terminal by RTCP.

The software responsible for rendering the porTiVity service is based on the Osmo player of the open source multimedia framework GPAC [FCM07]. An implementation with the full support of the porTiVity features is available for Windows Mobile version 5 and 6, Windows XP and an experimental version for Symbian OS. The developed scenarios work on normal off-the-shelf mobile phones with DVB-H receivers and touch screens. Furthermore any kind of web access is required for downloading the additional content files requested by the user clicking on the annotated objects. For various demonstrations we used terminals with limited processing power like a Gigabyte GSmart T600, a QTec 9000 and an LG KS20. As a high-end terminal an HTC Shift and a Samsung Q1 Ultra were used.



STATISTICS 1 of 3

52%	Possession	48%
4	Shots on target	2
2	Goals	2
14	Fouls	21

iSPORTS USER PROFILE

1) Sex Option:
● Male ○ Female

2) Team Highlights:
☒ FC Porto □ SL Benfica

3) Highlight MMS Channels:
○ Yes ● No

1 of 2

Figure 6: Interactive sports scenario.

4.3 End-user experience

iSports – interactive sports scenario

The iSports scenario (see Figure 6) demonstrates how interactivity can be applied to live sport events. For this case a separate annotation tool – the Live Annotation Tool [DFK+09] – has been developed in the porTiVity project. With this tool it is possible to define objects of interest and link additional content, track the objects in realtime and directly send the results to the playout system – with an overall delay of 5 seconds to the input stream.

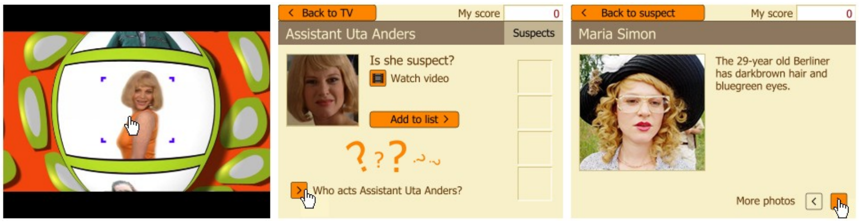
Next to the interaction possibility with the highlighted objects, a menu with buttons is shown in the top left corner. The menu, which can be collapsed in order not to consume valuable display space, allows to access replays of interesting events like goals, to access a help page that explains the available features of the service and to configure personalisation settings.

Spur & Partner – interactive crime series for children

The Spur & Partner scenario (see Figure 7) is based on a successful interactive TV crime series for children produced by the public German broadcaster ARD. Users can interactively collect hints by clicking on the objects annotated by the producer. They also can answer questions referring to those pieces of evidence and receive points for correct answers, which gives the TV program a little game characteristic.

The end-users can also access background information on actors and learn from an interactive detective magazine more about the approach that the detective is using to identify the culprit. And finally at the end of the series the users have the chance to name the culprit by themselves and directly transmit their results.

Spur & Partner was evaluated by Rundfunk Berlin Brandenburg (RBB) with 6 children at the age of 9 to 11 years. They had normal computer skills and were familiar with the original TV series and the interactivity by its MHP application. Evaluation results have shown that recognizing the bounding box of the moving objects and further interaction with the additional content improve with the experience and are no problem after a little trying.



Is she suspect?

Who acts Assistant Uta Anders?

The 29-year old Berliner has darkbrown hair and bluegreen eyes.

Figure 7: Interactive detective story Spur & Partner.

4.4 Evaluation of the authoring tool

The authoring system was evaluated by RBB with professional users from various departments. An evaluation of the tools has shown that the time required to annotate objects are realistic and acceptable in a TV production environment. The system was tested on the interactive detective story with a video length of 8:40 minutes and consisting of 16 interactive elements. Not considering the automatic preprocessing step, the whole annotation process took 1:15 hours with the additional content already prepared. Missing shortcuts in the evaluation version for common task could further increase the annotation performance.

5 Conclusion

This work has shown that MPEG-7 is suitable as description format for video including the description of moving objects. We presented tools that aid an operator in efficient object annotation by using automatically extracted metadata in MPEG-7 format and enrich it with annotations for moving objects. On the delivery side, MPEG-4 LASeR has proved to be a powerful standard for enabling rich multimedia interactive services on mobile devices with limited processing power. Due to the fact that MPEG-4 LASeR is based on SVG it allows scalability for a wide range of target devices. The chosen approach to use existing annotation tools based on MPEG-7 and adding MPEG-4 LASeR functionality combined the advantages of both standards and resulted in a full featured MPEG-4 LASeR authoring suite.

Acknowledgements

The authors would like to thank their colleagues at the Institute of Information Systems & Information Management at JOANNEUM RESEARCH, especially Helmut Neuschmied, Georg Thallinger and Werner Haas, as well as the other partners of the porTiVity project for the good collaboration.

The research leading to this paper has been partially supported by the European Commission under the contracts FP6-027098, "porTiVity – Portable Interactivity" (`http://www.portivity.org`), FP6-045032, "SEMEDIA - Search Environments for Media" (`http://www.semedia.org`) and FP7-231161, "PrestoPRIME".

References

[BS06] Werner Bailer and Peter Schallauer. The Detailed Audiovisual Profile: Enabling Interoperability between MPEG-7 Based Systems. In Huamin Feng, Shiqiang Yang, and Yueting Zhuang, editors, *Proceedings of 12th International Multi-Media Modeling Conference*, pages 217–224, Beijing, CN, Jan. 2006.

[Car05] B. Cardoso. Hyperlinked Video with Moving Objects in Digital Television. In *International Conference on Multimedia and Expo*, pages 486–489, 2005.

[DFK$^+$09] J. Deigmöller, G. Fernandez, A. Kriechbaum, A. Lopez, B. Mérialdo, H. Neuschmied, F. Pinyol Margalef, R. Trichet, P. Wolf, R. Salgado, and F. Milagaia. Active Objects in Interactive Mobile Television. In *15th International Conference on MultiMedia Modeling*, Sophia-Antipolis, FR, Jan. 2009.

[DFL$^+$08] J. Deigmöller, G. Fernandez, A. Lopez, B. Mérialdo, H. Neuschmied, F. Pinyol, R. Trichet, P. Wolf, R. Salgado, F. Milagaia, S. Glaser, and A. Duffy. Portivity : Object Based Interactive Mobile TV System Networked and Electronic Media. In *NEM SUMMIT 2008, Networked and Electronic Media, October 13-15, 2008, Saint-Malo, France*, Oct 2008.

[DVB04] Transmission System for Handheld Terminals (DVB-H). ETSI EN 302 304, Jun. 2004.

[FCM07] Jean Le Feuvre, Cyril Concolato, and Jean-Claude Moissinac. GPAC: open source multimedia framework. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 1009–1012, New York, NY, USA, 2007. ACM.

[LAS06] MPEG-4 LASeR. ISO/IEC 14496-20, 2006.

[Low99] David G. Lowe. Object Recognition from Local Scale-Invariant Features. In *Proceedings of the International Conference on Computer Vision*, pages 1150–1157, 1999.

[MHP07] MHP 1.2. ETSI 102 590, 2007.

[MPE01] Information Technology—Multimedia Content Description Interface. ISO/IEC standard 15938, 2001.

[MXF04] Material Exchange Format (MXF) – File Format Specification (Standard). SMPTE 377M, 2004.

[NTM07] Helmut Neuschmied, Rémi Trichet, and Bernard Mérialdo. Fast annotation of video objects for interactive TV. In *MM 2007, 15th international ACM conference on multimedia, September 24-29, 2007, Augsburg, Germany*, Sep 2007.

[Por] porTiVity web page. http://www.portivity.org.

[RBN$^+$07] Herwig Rehatschek, Werner Bailer, Helmut Neuschmied, Sandra Ober, and Horst Bischof. A Tool Supporting Annotation and Analysis of Videos. In Stefanie Knauss and Alexander D. Ornella, editors, *Reconfigurations. Interdisciplinary Perspectives on Religion in a Post-Secular Society*, pages 253–268. LIT, Vienna, 2007.

[SVA] Semantic Video Annotation Suite. http://www.joanneum.at/en/fb2/iis/products-solutions-services/semantic-video-annotation.html.

[SZ03] J. Sivic and A. Zisserman. Video Google: A Text Retrieval Approach to Object Matching in Videos. In *Proceedings of the International Conference on Computer Vision*, volume 2, pages 1470–1477, October 2003.

[TM07] Rémi Trichet and Bernard Mérialdo. Generic object tracking for fast video annotation. In *VISAPP 2007, 2nd International Conference on Computer Vision Theory and Applications, 8 - 11 March, 2007 Barcelona, Spain*, Mar 2007.

A User Study on Rich Media Mobile Guide Applications

Omar Choudary, Benoit Baccot, Romulus Grigoras, Vincent Charvillat

omar.choudary@enseeiht.fr, bbaccot@sopragroup.com

romulus.grigoras@enseeiht.fr, vincent.charvillat@enseeiht.fr

IRIT, University of Toulouse

Abstract: As smartphones' capabilities are more and more similar with those of computers, the demand for high-end multimedia content is increasing. In this paper we present two new visually rich mobile guide applications and then we compare them with the popular Google Maps and Nokia Maps applications. We present and analyze the results of an empirical study made with over 20 users in which we have compared these applications in the context of a game event at a big stadium. We have obtained direct feedback from participants as well as implicit feedback using a tracking system to analyze the interaction with the applications and infer activity metadata.

1 Introduction

Over the past years we have seen an incredible growth of mobile applications, especially caused by the advances in technology from both sides: mobile manufacturers and mobile operators. Devices have many capabilities, many smartphones have become much more powerful than two-year old PCs.

A very important functionality brought by the advances in technology is the location capability of these devices. Today, a smartphone is not just able to show your agenda, or to run a small application, but it is also able to give your position (more or less accurate, depending if you are using GPS, Wi-Fi or GSM-based positioning [DR07, CK02]). It can analyze and react on your movements (using accelerometer), store the position of your car and show you the path to follow to return to it, share your photos/videos and other personal items with your family or friends and much more. The smartphone has become the digital personification of ourselves.

Applications and services that use the location capabilities of mobile devices are commonly called Location Based Services (LBS)[BKH08] and applications that use the context of the user (position can be included here together with profile, mood, orientation, social networks, etc...) are so called context-aware applications [MZ07].

A particular type of LBS and context-aware applications are the mobile guide applications [AAH^{+}97]. These applications use context information (position, environment, user information, social network, personal interests, etc...) to present the user a map centered on his current position as well as activities (fan, cultural events, concerts) or places of interest (bars, restaurants, theaters, metro station) that are near the user. Popular examples are Google Maps [Mapa] and Nokia Maps [Mapb].

The advances in mobile technology (both network and devices) lead to a greater demand for better and richer content. Current mobile data-speeds exceed 10Mb/s (using High-Speed Downlink Packet Access) and will continue to grow in the future as LTE (Long Term Evolution) [3GP] and WiMax [16006] become fully deployed. As for the hardware, we can already see GPU, GPS, Wi-Fi, dual core and 5 mega-pixel camera in a single device (e.g. Nokia N95 [N95]). It is clear that we can deliver great rich multimedia content to mobile users such as interactive 2D images and videos, interactive 3D and Augmented Reality [BC05, Nur06, HMB07].

We think the content on mobile guide applications can be greatly improved and we have created two applications to prove this. In order to check what is the user preference and what is the most popular visual content for mobile guiding applications we have made a user study. In this study we have compared our applications with two of the most popular mobile guide applications: Google Maps and Nokia Maps. We have also created a test framework in order to record the user interaction with these applications. We used this framework to analyze the activity of the user while using these applications. As it will be presented in the following sections, the recorded information is very important in order to get an implicit feedback on the usefulness and simplicity to use of the applications.

The paper is organized as follows. Section 2 describes the motivation of our work. In Section 3 we present our framework and the applications tested. In Section 4 we detail the user study and we show and interpret the results. In Section 5 we summarize our work and discuss about future research.

Figure 1: *Stadium Activities Before the Match*

2 Motivation

On big events like football games, before and after the match there are plenty of attractions and interesting activities around the main venue (the stadium in our case). Example of such activities are small football contests, car auctions or rotating cups (see Figure 1). These attractions can be very hard to observe and locate, especially due to the large number of people that participate to such events. Very often people don't know how to locate these activities or they don't know of their existence. In this paper we analyze the particular case of a football game at the TFC Stadium in Toulouse.

Mobile devices can be of great help in the task of locating and showing the interesting activities around a stadium.

Currently used mobile guide applications lack rich-media support to help users in visualizing these activities or locating them. Existing applications such as Google Maps or Nokia Maps don't show at all these small activities around the stadium, and for those places of interest that are marked (like bars or restaurants near the stadium) there is generally a great level of interaction needed from the user side. Moreover, these commercial applications don't offer enough visual content to the user and they make very little use of multimedia capabilities in current devices. However, future and beta versions of these applications show great evolution in terms of multimedia content on mobile devices.

We created two different prototype applications in order to expose the interesting activities around the stadium to the user. We show the position of these activities and we also use a rich visual content in order to let the user know better what are these activities about. The first prototype uses fake 3D images, simulating a 3D model because the user can move around the stadium and zoom in and out as if he/she were inside the stadium area. This prototype is based on our recent work regarding fake 3D images to guide users in stadiums [CCG08]. The second prototype uses video sequences recorded directly on site. Each video sequence represents an activity that is going on at the stadium on the occasion of a big game.

We want to test the four different applications (our two prototypes, Nokia Maps and Google Maps) in the context of our stadium. As it is not fair to compare the four applications using only one criteria (for example our prototypes have clearly more attractive POI around the stadium because we have created them), we have investigated multiple characteristics of these mobile guide applications, including the level of interaction, visual content, user input and application reaction. We have conducted an empirical study to test the applications, presented in Section 4.

In order to save the results from the test and analyze them further, we have created a test framework which logs all interactions between the user and the tested applications. Besides the implicit feedback obtained from the logged events we have also created a questionnaire (explicit feedback) that was given to each test user. This questionnaire contains multiple questions that are relevant to the usability of the application as well as to the background of the test user. We describe the test framework and the feedback questionnaires more in detail in the following sections. Using the results from both implicit and explicit feedback we are able to derive some interesting conclusions.

3 System Overview

In this section we will present the mobile device, the four applications and the test framework used in our user evaluation.

3.1 Test Device

For our research we are using a Nokia N95 device. This smartphone has multiple features, including A-GPS, GPU (graphic processing unit), dual processor (ARM11), Wi-Fi, 5 Mega-pixel camera, 240x320 screen resolution and the possibility to play MPEG4 video files. The N95 runs on the Symbian OS, version 9.2 and it provides a rich sets of APIs to access most of the phone's features. We have used the native Symbian C++ to develop our prototypes.

Figure 2: *Nokia N95 8G device*

3.2 Test Framework

In order to get interesting results from the empirical evaluation we have created a test framework to log and analyze the interactions between the user and the tested applications.

All tested applications contain the same points of interest around the TFC stadium (the common area tested in our evaluation). For each user we start a new session and then for each application tested we start a new application session. We send all input events (see Figure 3) from the mobile applications to our server in order to log the interaction and extract useful information on a per user and per application bases. We define an `event` as one simple interaction between the user and the mobile application (e.g. pressing a key).

Normally the user tracking service works only with HTML pages : our platform includes

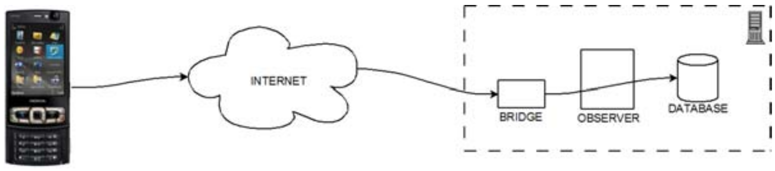


Figure 3: *Test Framework Overview*

a proxy that modifies the HTML page returned to the user in order to add some JavaScript listeners intended for tracking user interactions (mouse-click, mouse-move, key-input, etc...). In order to make our mobile applications work with the platform, we have created a bridge (`Bridge`) which listens to HTTP requests from the mobile application and then forwards them to the actual service (`Observer` and `Database`, see Figure 3).

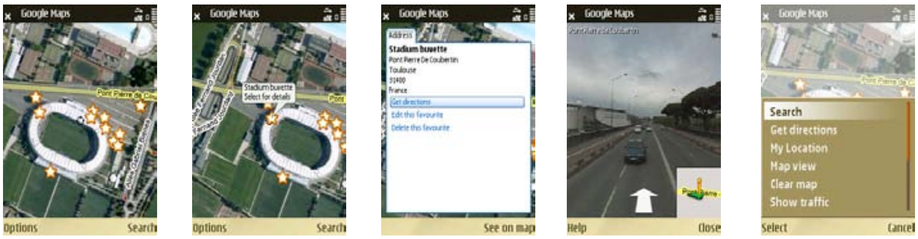
3.3 Tested Applications

The four applications that we have used for our tests are the two popular applications Google Maps and Nokia Maps and our two prototype applications using fake 3D images and video sequences.

3.3.1 Google Maps

Since its release in February 2005, Google Maps has been one of the most popular web guiding applications. Besides the web application, Google has created Google Maps for mobile devices. One of the most popular features of the mobile Google Maps is called `My Location` which can give the approximate location of the user by using only the information from the mobile network without any constraint on the GPS capabilities of the phone. The last version of the Google Maps for Symbian (2.03) also offers the Street View feature, allowing users to actually see real images from the places they choose. Although, this last feature is still limited to major cities in a few countries.

The Google Maps application interface can be seen in the Figure 4. The stars represent the points of interest -POI- (the activities around the stadium in our case). The user can move through the map using the navigation keys (up, down, left, right), zoom in and out using the `1` and `3` keys, select a POI using the middle key in order to see more information about it, see a list of the POI, center the map on the selected POI and switch between a simple 2D map or a satellite view. The position of the user (if available) is marked in the interface by a small blue dot. It is also possible to use the Street View feature to see images from the *Point Pierre de Coubertin* which is close to the stadium.

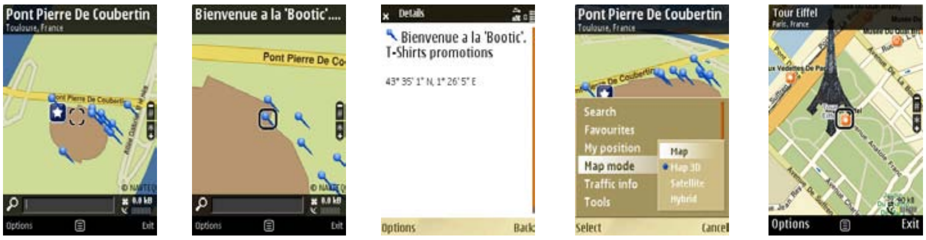


(top view) (target on POI) (POI selected) (Street View) (Options Menu)

Figure 4: *Different snapshots of the Google Maps application*

3.3.2 Nokia Maps

Nokia Maps is developed by Nokia and it is one of the most popular mobile guide applications [Res]. Its success is mainly due to the fact that it is targeted only to mobile phones and its features and user interface have been thought especially for these devices. This application is only available for Nokia phones, running the Symbian OS and using the S60 or S40 user interface. The current version (2.0) only has 2D and satellite images, but the 3rd version [3.0] which is already available for some models running Symbian 9.3 or later (soon available on Symbian 9.2 and thus on the N95) allows a 3D navigation of the maps, including 3D landmarks (the TFC stadium has a coarse 3D model in these maps).



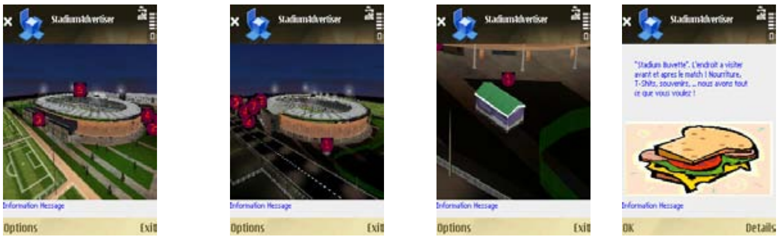
(top view) (target on POI) (POI selected) (Options Menu) (Version 3.0)

Figure 5: *Different snapshots of the Nokia Maps application*

In the Figure 5 we can see the user interface of Nokia Maps for both versions 2.0 and 3.0. The blue balloon-like icons represent our POI. The navigation through the map and selection of POI is done in a similar manner as in the Google Maps application. The main differences are in the map and POI interface, the way to display the information of the POI and the options of the application. The position of the user is presented by the rounded dashed rectangle. When the user is navigating through the map and the rectangle position is close to a POI its border line transforms to a continuous line.

3.3.3 Prototype 1 - Fake 3D

Our first prototype uses a fake 3D model based on images rendered from a 3D model of the stadium. One of the main differences of this approach when comparing it to the 3D landmarks from Nokia Maps is that we use a fine and high quality model of the stadium. The application allows the user to navigate around the stadium using simple images and it gives the sensation of a realistic navigation in the stadium.



"Stadium Buvette". L'endroit a visiter avant et apres le match ! Nourriture, T-Shits, souvenirs, ... nous avons tout ce que vous voulez !

(perspective view) (perspective view) (zoom in) (POI selected)

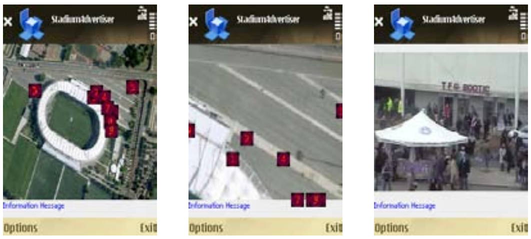
Figure 6: *Different snapshots of the fake 3D application*

The interface of the application can be seen in Figure 6. The application presents a perspective view of the stadium around which the user can move using the left and right keys. On top of the image we have overlaid numbered icons representing the POI. When the user presses the key corresponding with the number on the icon a new image is presented to the user, containing a visual representation of the POI as well as a text containing more information regarding the activity or place. Using the $*$ and $\#$ keys of the phone the user can zoom in and out in the images in order to see better where each activity is exactly located. A similar approach of using images and text as advertisements can be seen on the Smart Rotuaari project [TJM+03].

3.3.4 Prototype 2 - Video Sequences

In the second prototype we use video sequences recorded from the stadium to show exactly what interesting activities or places exist around the stadium just before the actual game.

As it can be seen in Figure 7 we have used a top view of the stadium taken from Google Earth to display the stadium area to the user. On top of this image we have added overlay icons representing the interesting activities and places. The user can zoom in or out and move around the top image to locate the position of the icons. When he/she presses the key corresponding to the icon a video sequence is played in the application, showing the real activity. The user has the possibility to adapt the volume using the volume keys of the N95 and he can stop playing the video using the middle key.



(top view) (zoom in) (POI selected)

Figure 7: *Different snapshots of the video application*

3.3.5 Advertisements

The points of interest (POI) used in the four applications represent different activities that exist around the stadium with the occasion of a game (see Figure 1). We call these POIs **advertisements** interchangeable throughout the text. The same POIs are used in all the four applications, even if they are represented in different ways: default markers on Nokia Maps and Google Maps, images and videos in our prototypes.

3.4 User Tracking Server

We use a tracking web service in order to get and analyze user interactions with the mobile applications. As shown in Figure 3, we were compelled to make a bridge in order to use the actual service. Nevertheless, whereas the user session identification is made automatically using the JavaScript listeners and the web service, it has to be handcrafted in the presented case : the generic listener on the mobile phone manages sessions and adds their ID to each request done to the server.

4 Empirical Evaluation

In this section we explain the organization and the results of our empirical evaluation. We did this user test to compare the four applications presented in order to learn what is the preferred multimedia content, what is the best user interface, what is the most time consuming application and to learn what is the level of interaction required by each application.

4.1 Test Organization

After preparing the applications (basic sanity checks) we made two kinds of tests (see Figure 8). Firstly, we made half of the tests with unknown people in a crowded area in down town Toulouse. A few days later we made another test with colleagues from our lab (IRIT) and students from our university (ENSEEIHT).

(Lab Test) (Student Test)

Figure 8: *Images taken while testing the applications*

We needed an Internet connection to connect with our user tracking system. For the tests made in the lab we used our wireless LAN and for the tests done in the city and in the university we used the 3G connection of a mobile network.

For each test user we followed these steps:

- brief the user with an overview of the test, explain why we are doing it and present the four applications
- start a new test session so we know that all the following recorded events belong to this test user (we note the session number so we can compare the implicit feedback with the explicit one)
- present shortly each individual application and let the user test them. The order was: Google Maps, Nokia Maps, First Prototype and Second Prototype. This order was chosen arbitrary but it was kept for all the participants.
- give the questionnaire to the user to be completed
- collect the questionnaire and reward the user with a pen that has the institute label on it

Each user tested each application for a maximum of two minutes. The scope of all the tests was to visit all the activities around the stadium within the two minutes by interacting with the application: moving the display screen around the stadium, zooming in and out, selecting POI, watching a video or using some special feature like *StreetView* on Google Maps. We have chosen the limit of two minutes based on preliminary experiments.

We had a total of 21 users, 11 were women and 10 were men. 5 of them were young (15 to 20 years old), 11 adults (between 20 and 30 years) and 5 mature people (30 to 50 years old).

The questionnaire comprised 27 questions, organized in three categories: user reactions using the tested applications, technical problems and user background. The questionnaire usually took about five minutes to complete.

4.2 Qualitative data analysis

Using the information from the completed questionnaires we compiled multiple statistics. The charts of these statistics can be seen in Figure 9.

From chart (a) we can observe that the `Fake 3D` application had the best visual content, while `Nokia Maps` had the worst content. In chart (b) we see that people liked the interaction with the `Google Maps` application. Most of the people decided that these type of applications will not definitely make them visit the presented places but at least the information was useful and they liked the applications as they would recommend the application to others (see charts (c) and (e)). From chart (d) we see that men preferred the `Google Maps` application while women voted for the `Fake 3D` application. Chart (f) is also representative for the difference of perception between men and women as men remembered nine activities while women only five.

4.3 Quantitative data analysis

We made many statistics based on the recorded information. We had over 7000 user events recorded in our database and we created different scripts in order to obtain the different statistics. In Figure 10 we present the results of our implicit analysis.

Based on chart (a) we can observe that most of the users used the 2 minutes to test all the applications except the `Prototype 2`. On this application users only had to press a key to watch a video which was about 8 seconds long. This fast interaction allowed users to test the application in less than one minute. Chart (b) reveals one of the most interesting results from our research, the number of events divided by the number of visited advertisements in each application. This result tell us, in average, the number of events (key pressed) done by the user before actually seeing an advertisement (video, image, text, etc...). The `Fake 3D` application required the highest number of events before visiting a POI but this can be explained also because the application offered a simulated 3D view of the stadium and most of the users where interacting extensively with the application in order to see the exact location of the advertisements as well as the stadium itself. Chart (c) shows the average number of events done by users in each application while chart (d) shows the average time spent looking at the information from the different advertisements. As expected the `Video` application required the least amount of interaction while the time visiting the advertisements is the largest because each advertisement was represented by a video about 8 seconds long. Chart (f) confirms that users explored the visual content in the `Fake 3D` application as they have been zooming in and out extensively in order to see better the stadium and the position of the advertisements.

4.4 Overall Analysis

In this section we analyze the results from both implicit and explicit feedback and we try to observe the best and the worst in each application based on the user opinion.

The fastest access to information was provided by Google Maps, as it can be seen in Figure 9g. It was also one of the preferred applications and the most easy to use application, despite the small problems in use caused by our logging system. Google Maps has provided a simple but attractive user interface and a fast access to POIs. Based on Figure 10f we can see that users didn't have to zoom in or out to visit a POI and it was the application with the most visited advertisements (see Figure 10e). The *StreetView* feature of the application had a very positive impact as most of the users were amazed by this feature.

Nokia maps was not able to perform in any category based on our results. Even if most of the functionality is similar to Google Maps the users preferred the later or the two prototypes in all the aspects presented. The main reasons for this were the lack of a visual image of the stadium and the latency in selecting POIs. Nokia Maps features a satellite view as well as Google Maps but its resolution is much lower and our stadium was not well visible. This is why we have used the 2D map which contained only a brown shape of the stadium. To select a POI users had to wait for a few seconds for the cursor to detect the POI under it and change its state. This was clearly a bad influence on the application usage. We are confident that the next version of Nokia Maps (3.0) which is already available for some devices will bring a much better user experience as it provides 3D landmarks and what we hope a better navigation.

The *Fake 3D* application (Prototype 1) has clearly the best visual content based on Figure 9a. This success is caused by the use of multiple images rendered from a nice 3D model of the stadium. Looking at Figures 10b and 10f we could think that this application involved too much interaction caused by the nice but possibly hard to use model. Contrary to this assumption, users actually liked this kind of content, fact proved by the result in Figure 9d. This means that mobile guide applications could benefit a lot from nice visually attractive models as users will enjoy them.

In the *Video* application (Prototype 2) we've tried to use video sequences as advertisements for two reasons: to reduce the interaction needed and to show real content directly from the stadium. Based on Figures 10a, 10b and 10c we can see that the degree of interaction is the lowest between the tested applications. Even if we expected that users will prefer this type of application, based on Figure 9 it seems that the *Video* application was not the best in any category, even if some users preferred this kind of application. Some people argued that the video sequences presented scenes with many people and this made the presented place hard or impossible to observe. Other comments refer to the impossibility to locate yourself or the interesting activity within the stadium while watching the video scenes. Based on comments and results we think that video and other type of rich multimedia content could greatly improve mobile guide applications but this content has to be made in a professional manner and it should present the POIs as clear as possible.

In Table 1 we compare the answers to some of the questionnaire items between the lab and the public tests. Based on the results the only common opinion is the preference of a nice visual model to represent the environment and clear images and descriptions to represent the POIs.

Table 1: Difference between lab and public test results

	Favorite	Most intuitive	Most visually appealing	Most easy to use
Lab	Google Maps	Prototype 2	Prototype 1	Prototype 1
Public	Prototype 1	Google Maps	Prototype 1	Google Maps

5 Conclusion

In this paper we have analyzed four different mobile guide applications: Google Maps, Nokia Maps and two prototypes created in our lab. We added some missing features of the two commercial applications into our prototypes: fake 3D images rendered from a 3D model and real video sequences.

We made an empirical evaluation, testing the applications within the lab but also in a public place with people not related with the lab. We monitored the user activity using a tracking system and we recorded all the actions of the users. After the test we have asked the users to complete a questionnaire.

Based on the results from both the questionnaires and the recorded data we were able to identify the good characteristics of each application as well as their weaknesses.

Both Nokia and Google have already been making improvements in the multimedia content of their applications. The last version of Nokia Maps (3.0) contains 3D landmarks, including the TFC stadium that was used in our test and the last version of Google Maps (2.03) has included the `Street View` feature which allows the user to see real images from the city. The main drawback of these applications is the limited number of 3D landmarks in the case of Nokia Maps and the limited number of real images in the case of Google Maps. It is clear that users want richer multimedia content and commercial applications are already making the steps necessary to improve this very important part of mobile guide applications.

Next, we would like to concentrate on the context of the user, in order to show more relevant content according with the user preference. We want to take in consideration all the available context information, including position, time, profile and mood as well as personally selected preferences. Also we would like to research the mobile services that adapt themselves based on the environment conditions: noise, light, etc. On a close direction we also want to investigate the usefulness of augmented reality for mobile applications like mobile guides or museum applications.

References

[16006] IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems. *IEEE Std 802.16e and IEEE Std 802.16 Cor 1-2005*, pages 1–822, 2006.

[3.0] Nokia Maps Beta 3.0. **http://www.nokia.com/betalabs/maps**.

[3GP] Long Term Evolution System Architecture 3GPP. **http://www.3gpp.org/article/lte**.

[AAH+97] Gregory D. Abowd, Christopher G. Atkeson, Jason Hong, Sue Long, Rob Kooper, and Mike Pinkerton. Cyberguide: a mobile context-aware tour guide. *Wirel. Netw.*, 3(5):421–433, 1997.

[BC05] Stefano Burigat and Luca Chittaro. Location-aware visualization of VRML models in GPS-based mobile guides. In *Web3D '05: Proceedings of the tenth international conference on 3D Web technology*, pages 57–64, New York, NY, USA, 2005. ACM.

[BKH08] P. Bellavista, A. Kupper, and S. Helal. Location-Based Services: Back to the Future. *Pervasive Computing, IEEE*, 7(2):85–89, April-June 2008.

[CCG08] Omar Choudary, Vincent Charvillat, and Romulus Grigoras. Mobile guide applications using representative visualizations. pages 901–904, 2008.

[CK02] Yongguang Chen and H. Kobayashi. Signal strength based indoor geolocation. *Communications, 2002. ICC 2002. IEEE International Conference on*, 1:436–439, 2002.

[DR07] N. Deblauwe and P. Ruppel. Combining GPS and GSM Cell-ID positioning for Proactive Location-based Services. *Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on*, pages 1–7, Aug. 2007.

[HMB07] Anders Henrysson, Joe Marshall, and Mark Billinghurst. Experiments in 3D interaction for mobile phone AR. pages 187–194, 2007.

[Mapa] Google Maps. **http://www.google.com/gmm**.

[Mapb] Nokia Maps. **http://europe.nokia.com/maps**.

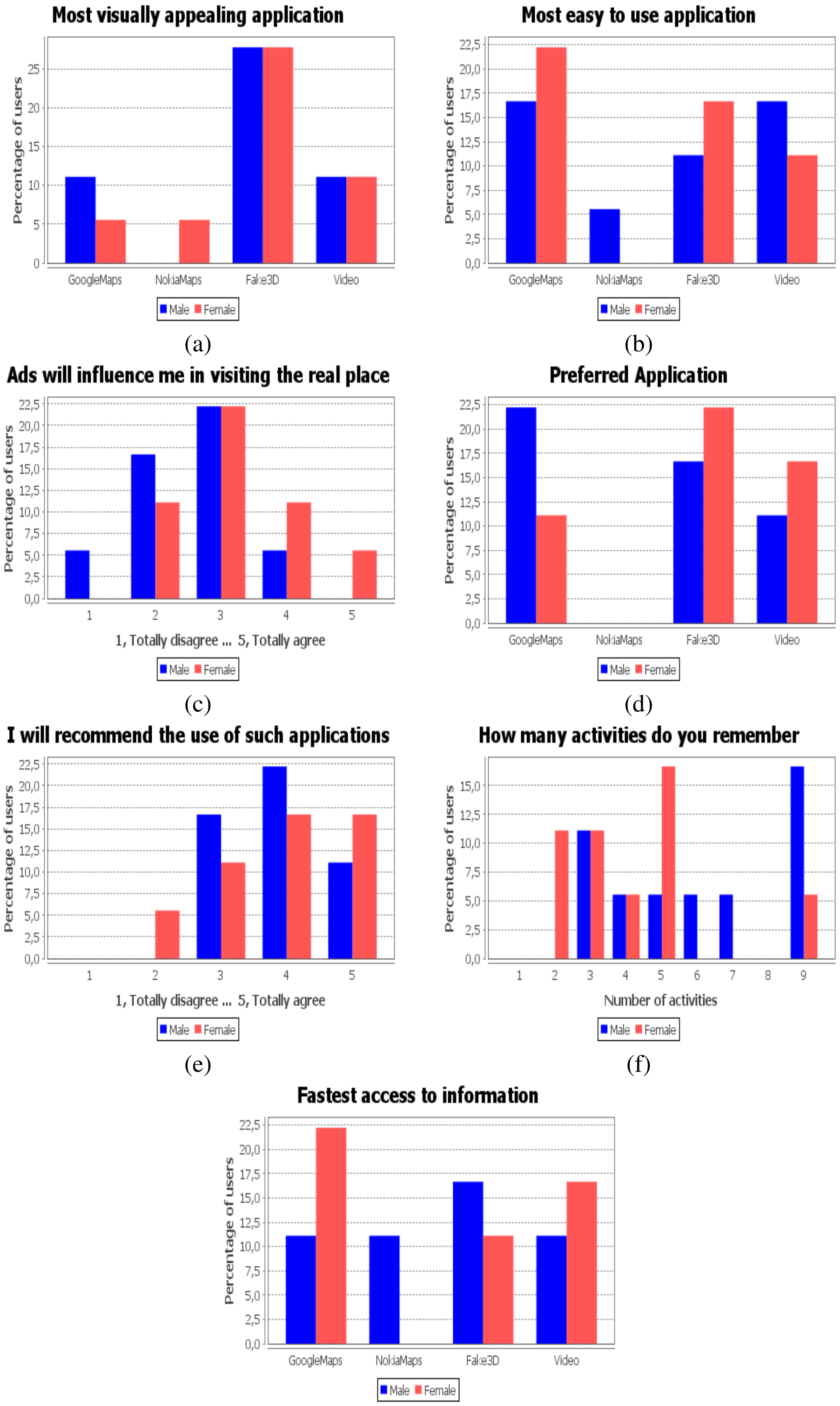
[MZ07] Y. Mowafi and Dongsong Zhang. A User-centered Approach to Context-awareness in Mobile Computing. *Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on*, pages 1–3, Aug. 2007.

[N95] Nokia N95. **http://www.forum.nokia.com/devices**.

[Nur06] Antti Nurminen. m-LOMA - a mobile 3D city map. In *Web3D '06: Proceedings of the eleventh international conference on 3D web technology*, pages 7–18, New York, NY, USA, 2006. ACM.

[Res] ABI Research. **http://www.abiresearch.com**.

[TJM+03] Ojala T, Korhonen J, Aittola M, Ollila M, Koivumäki T, and Tähtinen J & Karjaluoto H. SmartRotuaari - Context-aware mobile multimedia services. In *Proc. 2nd International Conference on Mobile and Ubiquitous Multimedia, Norrköping, Sweden, 9 - 18*, 2003.



(a)

(b)

(c)

(d)

(e)

(f)

(g)

Figure 9: *Statistics based on feedback questionnaires*

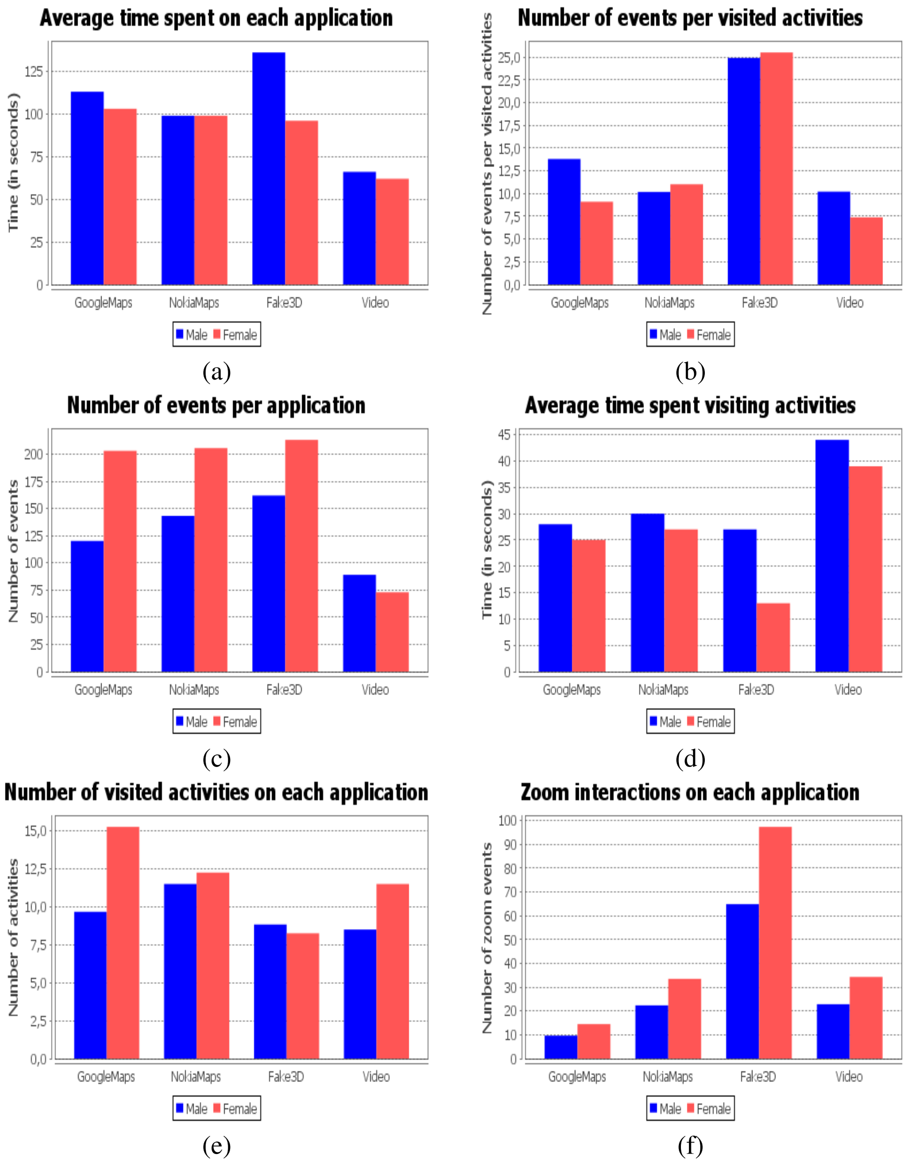


Figure 10: *Statistics based on events logged using the user tracking system*

What Algorithms for Urban Routing on Mobile Devices?

Tristram Gräbener, Alain Berro, Yves Duthen
IRIT-UT1, UMR 5505, CNRS
Université de Toulouse
2 rue du doyen Gabriel Marty 31042 Toulouse Cedex 9, France
{tristram.grabener,alain.berro,yves.duthen}@irit.fr

Abstract: This article presents a model for the multimodal shortest path problem. That model allows to use existing shortest path algorithm implementation and thus is very efficient and optimal. The performances are well suited for a mobile device like a smartphone.

We also present how multiobjective optimization improves the user interface by presenting similar solutions without any user interaction.

Finally we present the architecture of our prototype and give some experimental results.

Introduction

Today many mobile devices offer a GPS and a navigation system. However, those navigation systems are mostly oriented for car drivers: when a pedestrian mode exists, it is limited to the road network. It would be much more interesting to provide a complete itinerary to the destination including public transportation and possibly others means of transports like rental bikes. This problem is called multimodal shortest path.

However, such a system would be difficult to use as the user preferences are difficult to model and annoying to input on a mobile device: the best itinerary will depend on the user's personality, the weather, the luggage he might carry etc. Therefore presenting multiple solutions will allow the user to choose the most suited path with no need to enter the preferences.

We present in the first section a model for multimodal path calculation and the situations where good performances can be expected. The second sections presents multiobjective optimization and how it can greatly simplify the interaction on mobile devices. Then we present the architecture we developed for our prototype and some implementation details.

1 Multimodal transportation

Multimodal transportation consists in combining during a single itinerary multiple modes of transportation. An example would be to take the bike until the subway station, the subway and finally walk until the destination. Even if it becomes more and more popular among users and transportation network planing, only little research has been carried out about calculating optimal itineraries.

1.1 Graphs and paths

A valuated and oriented *graph* is a tuple $\mathcal{G}(\mathcal{N}, \mathcal{E}, \mathcal{F})$ where $\mathcal{N} = \{1, 2, \ldots, n\}$ is the set of n nodes, $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ the set of $m = |\mathcal{E}|$ *edges* and $\mathcal{F} = \{F_{ij} \mid (i,j) \in \mathcal{E}\}$ the m cost functions associated to each edge.

A *path* $p = (p_i)_{i \in 1,\ldots,l}$ is an ordered sequences of $l = |p|$ nodes such that two successive nodes are connected by an edge $\forall i \in 1, \ldots, l-1, \quad (p_i, p_{i+1}) \in \mathcal{E}$. The cost of a path is calculated by applying the cost function of each arc $F(p) = F_{p_1 p_2} \circ F_{p_2 p_3} \circ \ldots \circ F_{p_{l-1} p_l}(t_0)$. Traditionally, an edge (i,j) has a constant weight w_{ij}. Hence the cost function is $F_{ij}(t) = w_{ij} + t$.

Modeling a road network as a graph is straightforward: the intersections are the nodes, and street sections are the edges. The weight on the edges can be length of the link or the travel time through that link.

Between a source s and a dwell t, out of all possible paths, the one that has smallest cost is called *shortest path*. Many algorithms were developed to find the shortest path. The most known is probably Dijkstra's that runs in $\mathcal{O}(n \cdot m)$ [Dij59]. It is proven that it finds the shortest path as long as cost functions are increasing. The formal proof and more general conditions of optimality can be found in [GM]. By selecting carefully the datastructures, it is possible to have the algorithm that runs in $\mathcal{O}(n \cdot \log n)$ [FT87].

1.2 Shortest path for public transport

Public transport are also modeled as a graph: every station is represented by a node and an edge exists if there is a line directly connecting two station. In the literature, to handle the waiting time at a station, it is usual to add a waiting time on every node to define it when searching for the shortest path. This requires to use specific algorithms as it can be more interesting to wait longer on a node (for example to catch a direct train instead of an omnibus). A graph is said to be *FIFO* (first in, first out) if there is no such situation: leaving earlier from a node guaranties to arrive earlier.

The figure1 depicts a non FIFO graph: to go the node 4, the fastest way is to use only the car, while to reach the node 3, the fastest way is to use the subway. Therefore it an algorithm like Dijkstra's cannot be used.

An other approach is to split the day into q time intervals and to expand the graph to a *time-space* graph. For every node of the original graph, q nodes a created. The edges in the time-space graph exist if it is possible to go from a node to an other if there is a connection that matches the time interval of both nodes. The *Chrono-SPT* algorithm presented in [PS98] calculates the shortest path in $\mathcal{O}(q \cdot m)$. The strongest aspect of this algorithm is

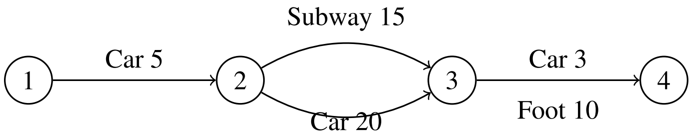


Figure 1: Non FIFO situation

to be able to implicitly handle the time-space graph and is quite memory efficient.

Both [ZW00] used that approach to solve the multimodal shortest path problem. This approach allowed them to solve the problem with 100 time intervals and 1000 nodes in a few seconds. A city like Toulouse has about 10000 nodes, and 100 time intervals represent 15 minutes.

As it is pointed out in [AOPS02], there is a significant difference between computing the *fastest* and the *minimum cost* shortest path in a dynamic network. The fastest path can be computed in polynomial time, while minimum cost is generally NP-hard. The intuition behind this property is that, as the cost depends on the time, it is necessary to explore every node at any time.

1.3 A model for multimodal shortest path

The model we present in this section allows to build a graph such that the FIFO constraint holds for the multimodal fastest path. With this transformation and by including the waiting time in the cost functions, our model allows the use of traditional algorithms and existing implementations.

One graph per transport mode Each transport mode is modeled by a different graph. The pedestrian graph and the bicycle graph will be very similar but the cost functions will be different. Also two trains using the same tracks, will be modeled as two different graphs if they don't mark the same stops. The nodes of two transport modes are connected if it is physically possible to switch from one mode to an other. The cost function on that connexion edge reflects the time needed to switch. There will be an edge in both directions if it is possible to switch, for example, from foot to public transportation and vice versa. By adding edges from the bike layer to the foot layer, but the not opposite, we model the fact that it is possible to start the itinerary by bike and once it is left somewhere, it won't be possible to use it again. The figure2 represents such a graph.

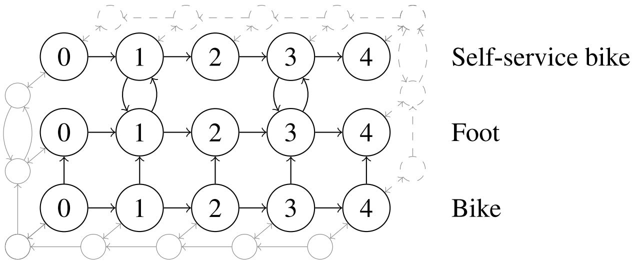


Figure 2: Multiple layer model

Including waiting time in cost function Instead of using separate waiting cost, we integrate the waiting time in the cost function. Given an edge $e = (i, j)$, the cost function would be $F_e(t) = w_e(t) + d_e$ where d_e is the duration on that edge (duration between two bus stops) and $w_e(t)$ the waiting time at the instant t until the next bus on the stop i. Considering that between two consecutive stops a bus won't overtake the previous one, this graph is FIFO. As the cost functions are increasing, it is possible to use the Dijkstra algorithm to compute the shortest path.

If the execution of the cost function is in constant time, then the global complexity of Dijkstra's algorithm will remain $\mathcal{O}(n \cdot \log n)$.

Discussion This model allows a very versatile and precise modeling of any transport mode. It allows to use a standard algorithms and thus use existing optimized implementations. First benchmarks prove the excellent performances of that approach.

However, this models requires more memory than a compact representation. The road map of the urban region of Toulouse (about one million inhabitants) requires 100Mb of memory. Therefore this isn't a real problem. The most restrictive condition is that only travel time can be optimized. Indeed any other objective depends on time (is a bus available? What will traffic be?) and thus breaking the FIFO constraint.

As it is impossible to ignore time, in order to handle other objectives at least two objectives have to be taken in account, hence multiobjective optimization cannot be avoided.

2 Multiobjective optimization

2.1 Benefits for a better user interface

Someone on the street with a mobile phone that is looking for the fastest itinerary will not want to spend time setting his preferences. Furthermore setting the preferences on a mobile device can be easily annoying. Last not least, it is not possible to save the preferences as they will be different on every use: if it is raining or if the user has heavy luggages, he would prefer avoiding multiple changes and long walks; on the contrary, if the weather is nice and he has some free time, he could enjoy walking even if it isn't the fastest way.

The objectives that are the most likely to be optimized are: time, cost, pollution (for example CO_2 emissions), number of changes and comfort. It is not possible to optimize each objective separately: the cheapest and less polluting will always be walking, but probably not the preference of the user. Aggregating the objectives into a single one is not desirable as it requires a precise configuration of the user.

Multiobjective optimization tries to optimize simultaneously all objective. The immediate consequence is that there is not one optimal solution, but a set of equivalent solutions: no solution beats any other solution on *all* objectives. This set is called the *Pareto front*.

Once the Pareto front is calculated, all solutions can be suggested to the user. According to his preferences, the user will choose the ideal itinerary.

Multiobjective optimization is a great tool to make interaction easier between a navigation system and the user as only the destination has to be entered. However calculating the Pareto front is not trivial.

2.2 Algorithms

Because most multiobjective optimization problems are NP-difficult and because there is not one, but multiples optima, most algorithms are based on genetic algorithms. More generally a population based algorithm is well suited to model the multiplicity of the solutions. For a general presentation of this topic, the reader can refer to [CVVL02].

However genetic algorithm aren't well suited for the point-to-point shortest path problem. Indeed, the usual genetic algorithms operators (mutation and crossing) aren't very natural to adapt to shortest path. Most research on multiobjective shortest path has been carried on exact algorithms. Several exact approaches are compared in [GM01].

Some others metaheuristics have been adapted to solve multiobjective problems like the particle swarm optimization or the ant colony optimization ([CVVL02] and [ASG07].

Ant colony optimization has been used successfully on some multiobjective problems, but not yet (to our knowledge) to the multiobjective shortest path problem. We believe that it is an promising approach, as this metaheuristic is inspired by ants searching for shortest paths.

3 Implementation and architecture

We created a prototype to demonstrate that our approach allows very fast itinerary calculations that scales well on bigger networks. The prototype is a web page where the user only has to select the start, destination and start time. The calculated itinerary is shown on a map detailing the transport modes to use.

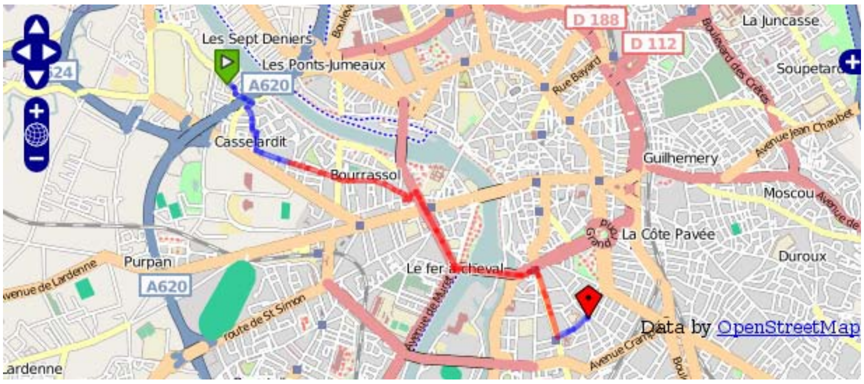


Figure 3: Screen shot of the prototype

3.1 Data

In our prototype we support pedestrian and self-service bikes in urban area of Toulouse. This bike rental system is available in most big cities in France and some other European cities: with a subscription it is possible to take and deposit a bike on any station. Toulouse has about 250 stations, and Paris about 1500.

The road network come from *OpenStreetMaps*[1] while the bike stations are available as XML files on the official website [2].

3.2 Implementation

On the client side (web-browser) we use OpenLayers, a javascript application, that allows to display the pre-rendered cartography and arbitrary paths.

On the server side, we use the implementation of Dijkstra's algorithm of the Boost Graph Library in C++. A Ruby on Rails application communicates with the client through Ajax. This allows an easy to use and reactive interface while minimizing the amount of data exchanged.

An example of the interface can be seen on figure3: a calculated itinerary using two different transport modes can be seen.

[1] http://www.openstreetmap.org/

[2] http://www.velo.toulouse.fr/

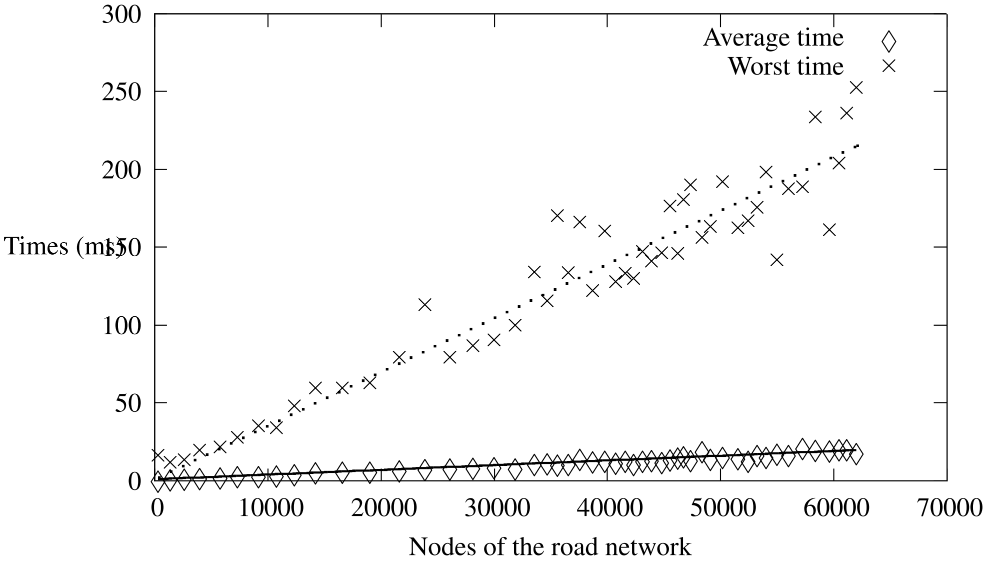


Figure 4: Multimodal shortest path computation time

3.3 Benchmarks

The benchmark were made by running 1000 itinerary calculations by choosing random start and destination nodes on 50 different graph sizes. This allows to test the average performance and how it scales. The tests were run on a 1.66GHz laptop.

The graphs are squares centered on Toulouse having a width between 2 a 100km, representing from 361 nodes up to 62020. The results are displayed on figure4. We can see that both the average and the worst running time out of 1000 calculations increase linearly with the number of nodes.

Those results are significantly better than in previous works. Therefor there is good hope to have an responsive software once we port our algorithm on a mobile device.

4 Future developments

The current models lacks of public transportations, and we hope to be able to include them very soon. We also believe that the model could easily be expanded to greater regions, like whole France. However this will require a bit of work in order to add heuristics that will limit the exploration space. Going on a even larger scale (the world for example) will be more problematic as it will not be possible to load all the data in memory.

While the current prototype works perfectly on a desktop computer, it has issues on mobile devices as the user interface is not designed for devices without pointing device or with touchscreen. We however don't think that creating a mobile version of our interface will be too difficult if the itinerary calculation is done on a remote server (approach used on the iPhone or on Android). Running the calculation on the mobile device will be more challenging as we get again a limited memory. Furthermore, or prototype requires about 15 seconds to build the graph. While it is negligible on a server (it is built only once and then kept in memory), it will be annoying for a mobile user.

Last but not least, the implementation of a multiobjective shortest path algorithm and its integration in our prototype would make difference with existing systems. As it also the most interesting part from a research point of view, we currently focus on that part.

Conclusions

We presented in this article a representation model for the multimodal shortest path problem. It allows to match closely the reality and to use existing implementation of shortest path algorithms. Benchmarks shows its excellent performances on realistic situations and scales well.

We made a small introduction to multiobjective optimization and how it would help to make simple user interfaces that match precisely user preferences without any interaction. It is therefore a great tool for mobile devices where input possibilities are limited.

Acknowledgement

We would like to thank the company SODIT (`http://www.sodit.eu`) and the region Midi-Pyrénées for their financial participation on this research topic. We also thank the OpenStreetMap community for high quality of the maps they produce.

References

[AOPS02] R.K. Ahuja, J.B. Orlin, S. Pallottino, and M.G. Scutella. Minimum Time and Minimum Cost-Path Problems in Street Networks with Periodic Traffic Lights. *Transportation Science*, 36(3):326–336, 2002.

[ASG07] Ines Alaya, Christine Solnon, and Khaled Ghedira. Ant Colony Optimization for Multiobjective Optimization Problems. In *19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 450–457. IEEE Computer Society, October 2007.

[CVVL02] C.A.C. Coello, D.A. Van Veldhuizen, and G.B. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer Academic Publishers, 2002.

[Dij59] E.W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, 1959.

[FT87] M.L. Fredman and R.E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM (JACM)*, 34(3):596–615, 1987.

[GM] Michel. Gondran and Michel. Minoux. *Graphs, dioïds and semirings: new models and algorithms*. Springer.

[GM01] F. Guerriero and R. Musmanno. Label Correcting Methods to Solve Multicriteria Shortest Path Problems. *Journal of Optimization Theory and Applications*, 111(3):589–613, 2001.

[PS98] S. Pallottino and M.G. Scutella. *Shortest path algorithms in transportation models: classical and innovative aspects*. Kluwer Academic Publishers, 1998.

[ZW00] A. Ziliaskopoulos and W. Wardell. An intermodal optimum path algorithm for multimodal networks with dynamic arc travel times and switching delays. *European Journal of Operational Research*, 125(3):486–502, 2000.

Towards the use of multimedia contents to represent events in vehicular ad hoc networks

Nicolas Cenerario #1, Thierry Delot #2, Sergio Ilarri *3

#*LAMIH Laboratory, University of Valenciennes*
Le Mont Houy, 59313 Valenciennes - FRANCE
1Nicolas.Cenerario@univ-valenciennes.fr
2Thierry.Delot@univ-valenciennes.fr

**IIS Department, University of Zaragoza*
Maria de Luna 1, Zaragoza, 50018 - SPAIN
3silarri@unizar.es

Abstract:

VESPA (Vehicular Event Sharing with a mobile P2P Architecture)[1] is a system designed for vehicles to share information in inter-vehicle ad-hoc networks. The originality of VESPA is to support any type of event (e.g., available parking spaces, accidents, emergency brakings, obstacles in the road, information relative to the coordination of vehicles in emergency situations, etc.) in the network.

In this paper, we discuss the use of multimedia content to describe events and the impact of exchanging such data on the dissemination protocol used to diffuse the events to the potentially interested vehicles.

1 Introduction

Today, the car is indisputably the most heavily used mode of transportation. Unfortunately, its popularity has been accompanied by numerous problems, for example, in the areas of safety and the environment. In spite of significant efforts to reduce the number of persons dying on the road, this number remains quite high, mainly due to the human factor (e.g., accident-prone behavior or low response time). To reduce the number of accidents, a variety of programs, generally involving "Intelligent Transport Systems", have been initiated.

Thus, many works have focused on information exchange in vehicular ad hoc networks (VANETs). These wireless networks rely on the use of short-range networks (about a hundred meters), like IEEE 802.11 or Ultra Wide Band (UWB) standards for vehicles to communicate [LH05] and provide bandwidth in the range of Mbps. Using such communication networks, the driver of a car can receive information – for example, about accidents, traffic congestion or available parking spaces – from its neighbors.

These last years, different systems have been designed ([XOW04, MHD$^+$03, NDLI04,

[1] For more information, see: http://www.univ-valenciennes.fr/ROI/SID/tdelot/vespa/

FFH$^+$02, NDK04a]). They aim at assisting the drivers by providing them information about accidents, emergency brakings, or available parking spaces. VESPA follows a quite different approach. Contrary to other systems, dedicated to the dissemination of one particular type of information, the originality of VESPA is to support any type of event occurring on the roads, even mobile events. Indeed, numerous types of events –both mobile and stationary– are possible, since there is a lot of information that drivers may find relevant. For example, about accidents, traffic congestion, emergency braking situations, fuel prices, available parking spaces, emergency vehicles such as ambulances, obstacles in the road, or the behavior of drivers (e.g., strange manoeuvres due to intoxication or lack of vigilance), to name but a few possibilities. Therefore, VESPA relies on the concept of encounter probability to estimate the relevance of an event for a vehicle [DCI08]. VESPA also includes a dissemination protocol [CDI08]. This protocol ensures an adaptive broadcast of the events in the vehicular network according to their type.

In this paper, we present the basic functionalities of VESPA and focus on the event representation. We particularly highlight the interest of using multimedia data to enhance the description of events and facilitate the communication of relevant and useful information to drivers. The rest of this paper is organized as follows. Section 2 describes how the relevance of events received on a vehicle is estimated. In Section 3, we explain how the events occurring on the roads are represented to be communicated to potentially interested drivers. Section 4 focuses on our dissemination protocol in charge of that communication task. In Section 5, we discuss how to improve the basic description of events. We notably introduce the use of multimedia data to complete that description. Section 6 presents the architecture and the main functionalities of our VESPA prototype. Finally, Section 8 offers our conclusions.

2 Relevance estimation

Many pieces of information may be exchanged in the context of inter-vehicle communications, for instance to warn drivers when a potentially dangerous event arises (an accident, an emergency braking, an obstacle in the road, etc.) or to try to assist them (with information about available parking spaces, traffic congestions, real-time traffic conditions on a road, etc.). Those different events may be detected by a car and lead to the generation of a message transmitted to the other potentially interested vehicles, either directly or using multi-hop relaying techniques. Once received by a vehicle, the relevance of a message has to be evaluated, according to spatial and temporal criteria to determine whether the driver should be warned or the message should be further broadcast.

To estimate the relevance of an event (i.e., to determine whether a vehicle will encounter an event or not), it is necessary to have an estimation of the vehicle's trajectory. Therefore, VESPA exploits mobility and direction vectors to characterize the vehicle's displacement and so estimate a future position of the vehicle. These vectors are computed thanks to GPS position statements (including 3-dimensional coordinates as well as a statement of the GPS time) obtained regularly.

The current version of VESPA does not rely on the use of digital maps. Obviously, the information provided by such maps would provide very interesting information about the road network. This would clearly help in efficiently configuring the dissemination protocol. Thanks to such maps it would be possible to determine precisely the destination area of a message. For example, the last exit before reaching a traffic congestion could be determined as the minimum objective to reach while disseminating such an event. However, such maps are not always available or accessible on users' devices. So, our goal was initially to show the feasibility of the approach without using maps, even if we are now considering their use to improve our solution. Moreover, actual maps do not provide all the information needed to evaluate the relevance of the events received by a vehicle. For instance, such maps do not provide information about the entrances of parking lots whereas that information is crucial to determine the closest parking space for a vehicle about to reach its destination.

By using vectors, the estimated future position is highly dependent on the t_n and t_{n-i} time interval selected between the position statements used to compute the vectors. Thus, if t_n and t_{n-i} are far away, the estimation of the future position is not precise but provides an overall impression of the object's direction. If the time interval is shorter, then the estimation is much more precise on the short term but no global view of the displacement can be observed. As an example, see arrows A and B in Figure 1.

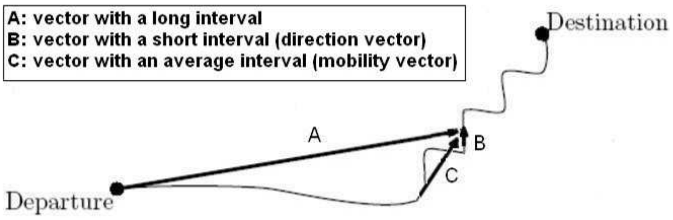

A: vector with a long interval
B: vector with a short interval (direction vector)
C: vector with an average interval (mobility vector)
Destination
A
B
C
Departure

Figure 1: Mobility and direction vectors

Depending on the way we select the time interval $[t_{n-i}, t_n]$, we distinguish:

- The *direction vector*, which is computed with a short interval. It provides a quite precise estimated future position but only in the very short term (see arrow B in Figure 1).
- The *mobility vector*, whose role is to provide an overall impression of the object's

movement in addition to a good estimated future position. To achieve a good compromise between the previous two cases (arrows A and B in Figure 1), an "average" interval must be used to compute it (see arrow C in Figure 1).

Using the mobility and direction vectors and the positions of the vehicle and the event, we can deduce four elements which have an influence on the encounter probability:

- The minimal geographical distance between the vehicle and the event over time (Δd).
- The difference between the current time and the time when the vehicle will be closest to the event (Δt).
- The difference between the event's generation time (stored in *CurrentPosition*) and the moment when the vehicle will be closest to it (Δg, *expected age of the event*).
- The angle between the direction vectors of the vehicle and the event (denoted by a colinearity coefficient c).

As an example, Figure 2 shows the geometrical representation of Δd and Δt. In the figure, B represents the vehicle's position, C the position of a stationary event, and $\overrightarrow{AB}$ is the mobility vector of the vehicle. Point D can then be determined, which allows a right-angled triangle to be constructed in D with $[BC]$ as hypotenuse. D is the closest point to C on the straight line between A and B. $|\overrightarrow{DC}|$ (= Δd) represents the minimal geographical distance between the vehicle and the event over time. $|\overrightarrow{BD}|$ is the distance between the vehicle and the point D. Since the mobility vector $\overrightarrow{AB}$ has a temporal dimension, $|\overrightarrow{BD}|$ can be converted into time to obtain Δt.

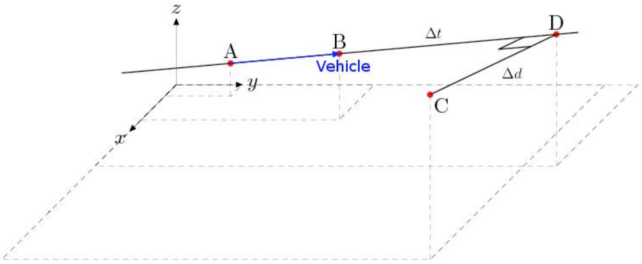


Figure 2: Representation of Δd and Δt

As explained previously, the vehicle estimates its direction vector and the event's direction vector. From these two direction vectors, a *colinearity coefficient* (c) is obtained, which is a measure of the angle formed by the vectors. For direction-dependent events that are not relevant for all nearby vehicles, but only for the vehicles travelling in a particular direction (e.g., an emergency braking, an accident, etc.), this allows us to determine whether the directions of the vehicle and the event match. For non-direction-dependent events, c is set to 0.

Once these Δd, Δt, Δg, and c values have been calculated, they are used to estimate an "encounter probability" between a vehicle and an event. The encounter probability (EP) is a value between 0% and 100%. It is computed, based on the previous values, using the following function:

$$\mathbf{EP} = \frac{\mathbf{100}}{\alpha \times \mathbf{\Delta d} + \beta \times \mathbf{\Delta t} + \gamma \times \mathbf{\Delta g} + \zeta \times \mathbf{c} + \mathbf{1}}$$

where α, β, γ and ζ are penalty coefficients with values ≥ 0. They are used to balance the relative importance of the Δd, Δt, Δg, and c values. The bigger the coefficient is, the more penalized the associated valued is when computing the EP. For example, the greater the α value, the shorter the spatial range where the event is relevant. β and γ are used so that only the information about events that will be encountered very rapidly and the most recent information is considered. Finally, ζ is used to weigh the importance of the colinearity coefficient. Notice that if the vehicle is moving away from the event, then Δt is 0 and Δd is the current distance to the event. Therefore, the computation of the EP makes sense even in cases where an interesting event (e.g., a parking space) is behind us. The EP is used to determine the relevance of an event. The greater its value, the more likely the vehicle is going to meet the event.

3 Representation of events

Thus far, existing V2V solutions have considered only a small subset of the possible types of events, primarily focusing on stationary events. However, numerous types of events – both mobile and stationary– are possible, since there is a lot of information that drivers may find relevant. For example, about accidents, traffic congestion, emergency braking situations, fuel prices, available parking spaces, emergency vehicles such as ambulances, obstacles on the road, or the behavior of drivers (e.g., strange maneuvers due to intoxication or lack of vigilance[2]), to name but a few possibilities. In order to determine the relevance of events, it is first necessary to classify the different types of events. In the rest of this section, we propose a system of event classification and describe how these events are represented in VESPA. For simplicity, not only all kind of events but also road hazards and available resources are called *events* in the following.

3.1 Event classification

The solution that we propose not only supports stationary events, such as the presence of gas stations, but also mobile events, such as an emergency vehicle asking preceding

[2] Lack of vigilance, or hypovigilance, can be detected today with oculometers using techniques that essentially count the driver's number of eye blinks.

vehicles to yield the right of way. When supporting such mobile events, the set of vehicles for which the event information is relevant evolves according to both the movements of the mobile event (in the example, the emergency vehicle) and the other vehicles involved (in the example, the preceding vehicles). Besides, the direction of traffic is also of major importance in establishing the relevance of shared information, even for non-mobile events (e.g., consider a traffic congestion affecting only the vehicles moving in one direction).

So, we classify inter-vehicle network events in four different categories:

1. *stationary, non-direction-dependent events*;
2. *stationary, direction-dependent events*;
3. *mobile, non-direction-dependent events*;
4. *mobile, direction-dependent events*.

By *direction-dependent events* we mean events that are not relevant for all nearby vehicles, but only for the vehicles traveling in a particular direction. On the other hand, *mobile events* are (as explained before) events whose locations change along time.

Let us illustrate our classification system by giving some examples. Available parking spaces correspond to stationary, non-direction-dependent events since they are static and may interest all vehicles close to that resource, regardless of the direction of movement. A warning about an accident is a stationary, direction-dependent event because its location is fixed and only those vehicles that are expected to encounter the accident will find the message relevant. The vehicles close to the accident but moving in the opposite traffic stream should ignore the message so as not to distract the driver and cause a second accident. Messages warning vehicles of the lack of vigilance of a person driving on a two-way road is a mobile, non-direction-dependent event because it concerns all vehicles likely to meet such driver, regardless of their direction of movement. Finally, an emergency vehicle broadcasting a message for other vehicles to yield the right of way is a mobile, direction-dependent event. Our goal in proposing such a classification of events is to support, in the same solution, all the types of events which can occur on the roads.

3.2 Basic Event Representation

In our solution, the four types of events identified in the previous section are used to represent all events occurring on the roads. In the following, we describe how these different events are represented when created[3] in order for them to be exchanged between vehicles (a summary of the attributes considered is shown in Table 1):

Each event is characterized by:

[3] We will not consider Human Machine Interface (HMI) aspects in this article. We rather focus on the representation and relevance estimation of events. The creation of those events may be initiated by devices embedded in the vehicles (for example by coupling the airbag system with the creation of an event representing an accident).

Attribute Name	**Type**
Key	string
Version	int
Importance	int
CurrentPosition	PositionAndTime
DirectionRefPosition	PositionAndTime
MobilityRefPosition	PositionAndTime
LastDiffuserPosition	PositionAndTime
HopNumber	int
Type	EventType

Table 1: Basic Event Representation

- A unique *Key*.
- A *Version* number to distinguish between different updates of the same event. Once generated, an event is disseminated among a set of potentially interested vehicles. To update the information transmitted to other vehicles, for example because a mobile event has moved, the vehicle which created the event may produce a new version of the same event.
- The *Importance* attribute, to determine whether the information should be presented to the driver or not. Unless the event is a very important one (e.g., an emergency braking), the driver is informed only if s/he is interested in that type of event.
- The *CurrentPosition* attribute indicating the generation time and place of the event.
- Two different preceding reference positions and their timestamps (*DirectionRefPosition* and *MobilityRefPosition*) for each vehicle to receive information to evaluate the mobility and direction of an event (see Section 2), which is necessary to estimate the event's relevance.
- The *LastDiffuserPosition* used by the dissemination protocol and containing the position of the last vehicle which relayed the message.
- The *HopNumber* attribute indicating the number of broadcasts of the message.
- The *Type* field describing more precisely the event considered (e.g., an accident, an emergency braking, etc.). This field is used to transmit concrete information to drivers when they need to be warned as we will see in Section 6.

4 Dissemination of events

Our objective as concerns the dissemination protocol is to disseminate different types of events (an accident, an emergency braking, an available parking slot, etc.) in the vehicular network. VESPA relies on vehicle-to-vehicle communication (V2V) (i.e., on spontaneous information exchanges between vehicles) and do not use mobile telephony networks (e.g., 3G), providing a worlwide access but increasing response time what is very penalizing in some situations (e.g., dealing with an emergency braking). Therefore, we have to support different dissemination modes according to the type of event considered. Indeed, an emergency braking has to be diffused to the vehicles driving in a particular direction whereas an available parking slot has to be transmitted to all close vehicles, whatever their direction, as it may interest them. VESPA uses a dissemination protocol[4] relying on the EP to determine the vehicles which have to broadcast some information they received. More precisely, when a message about an event is received by a vehicle, the vehicle will relay the message if and only if the value computed for the EP is greater than a predefined diffusion threshold. We indeed consider that a message relevant to a vehicle may also be relevant to its neighbors. Thus, our dissemination protocol allows diffusing the messages in the right direction, that is, towards the vehicles for which these messages may be relevant according to the type of event considered. Anyway, since this may happen at the same time on different vehicles, the same event may potentially be diffused numerous times by different vehicles. Therefore, to avoid flooding and so network congestion, our solution aims at desynchronizing the diffusions performed by the different vehicles. Thus, each vehicle waits for a period *t* before broadcasting the message. The size of that period depends on the distance between the receiving vehicle and the one which sent the message. The intuition behind this is to choose, among the neighboring vehicles which received the message, the farthest neighbor from the sender to relay the message. Indeed, this farthest neighbor may have the greatest number of neighboring vehicles not yet informed about the event being transmitted. It is so the best candidate to try to broadcast the message to all concerned vehicles as quickly as possible[5]. The value of *t* is determined by each vehicle as follows:

$$t = D \times (1 - \frac{d}{r})$$

where D is the maximum time to wait before broadcasting, r is the communication range of the wireless network used by the vehicles to communicate, and d corresponds to the distance between the receiving and the diffusing vehicle[6]. Since *d* may vary from 0 to *r*, *t* is between 0 and D.

5 Improving the description of events using multimedia content

Thanks to the basic attributes presented in Section 3, a driver can be informed or warned of events observed in its vicinity. Obviously, it can be interesting to add optional information

[4] See [CDI08] for more details.

[5] This removes the need of real-time monitoring the positions of the vehicles. Such a monitoring is indeed unrealistic in such dynamic environments.

[6] The value of d is computed using the position of the last vehicle that has relayed the event stored in the corresponding message.

to complete the description of each event, for example to indicate to the drivers the price of a parking or the length of a traffic congestion. Clearly, the use of multimedia content added to these event descriptions can also be interesting, either to give more details to the driver (e.g., a picture of an available parking space for the driver to know if s/he is able to park there) or to ease the delivery of the information to the driver (e.g., an audio message explaining that the warning transmitted to the driver corresponds to a dog walking on the road).

Multimedia content can help to communicate details about events to the drivers. Nevertheless, exchanging multimedia content in inter-vehicle networks is a challenging task. Indeed, whereas messages exchanged to described "classical" events (i.e., those without attachment of multimedia files) can be represented and disseminated over the network using a single packet, the size severely increases when multimedia files (e.g., pictures or audio files) are added to this message. Since vehicular networks are highly dynamic due to both the movements of the vehicles and the short range of the wireless communications, the exchange of multimedia contents between vehicles may fail frequently. This may be due to the fact that the available interaction time between two vehicles is not big enough (e.g., the vehicles are driving in the opposite direction) or due to a high probability of losing a packet.

So, we chose to use multimedia content only as optional attributes. Thus, even if the corresponding packets cannot be correctly received, the driver can be warned or informed of an event. Besides, when the workload is high in the vehicular network, the events can be relayed without their optional content in order to minimize the quantity of data transmitted and so to avoid losing packets.

We present in Table 2 the representation of an event including the optional attributes. An optional attribute is described by its name (represented as a string), a type and a value (several triples are allowed, as we will show later in Table 4). If the type of the optional attribute is different from "text" (e.g., "audio", "picture", etc.), then the associated value corresponds to an identifier referencing another entity where the raw data is stored (see Table 5).

Attribute Name	**Type**
Key Version	string int
Importance	int
CurrentPosition DirectionRefPosition MobilityRefPosition	PositionAndTime PositionAndTime PositionAndTime
LastDiffuserPosition HopNumber	PositionAndTime int
EventType	string
OptionalContent	$<$ string , type , string $>$

Table 2: Event representation with optional content

Attribute Name	Type
IdContent	string
Data	byte []

Table 3: Multimedia content storage

To illustrate the usage of this structure, an example is proposed in Tables 4 and 5.

Attribute Name	Example
Key	v1e1
Version	1
Importance	1
CurrentPosition	50° 19’15.91 N 3°30’51.11 E 10h25m17s
DirectionRefPosition	null
MobilityRefPosition	null
LastDiffuserPosition	50° 19’15.91 N 3°30’51.11 E 10h25m17s
HopNumber	1
EventType	parking
OptionalContent	< “Description” , text , “Available parking space” “For disabled” , text , “No” “Cost” , text , ”2 Euros / h” “Picture” , picture, “idpics01” >

Table 4: Parking example

Attribute Name	Example
IdContent	idpics01
Data	{0001 ... 0110}

Table 5: Parking multimedia content example

6 VESPA Prototype

In this section, we describe our prototype. We first present basic aspects of the prototype and then describe its software architecture.

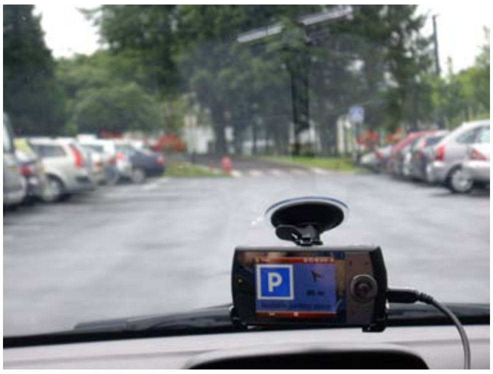


Figure 3: Testing VESPA in a real environment

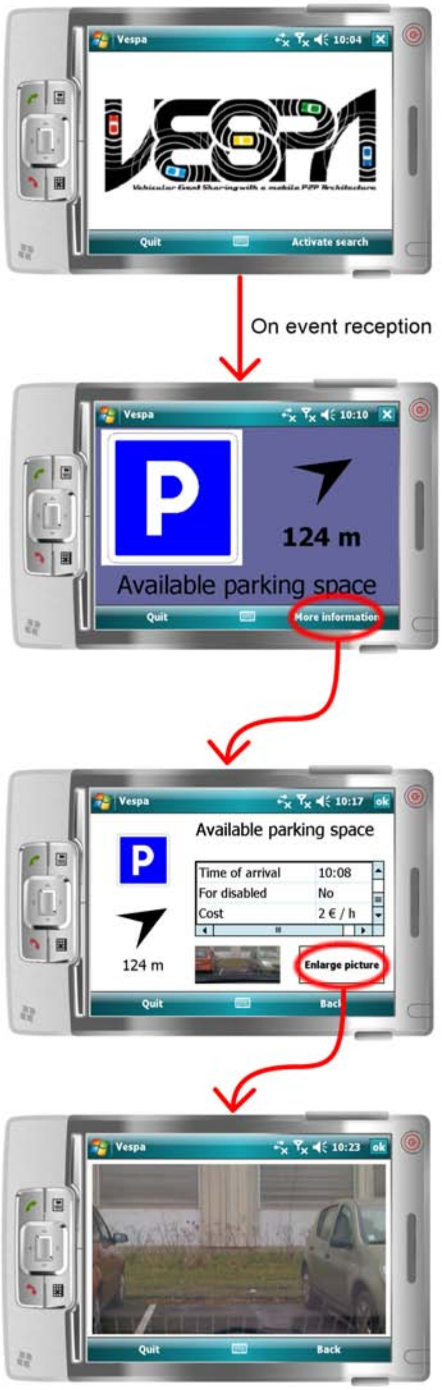
6.1 General Presentation

For obvious scalability reasons, our dissemination and relevance estimation techniques were evaluated on a simulator. Anyway, a prototype of VESPA has been implemented using Microsoft .Net/C# to observe its behavior in "real conditions"[7]. Our prototype was used to validate the dissemination of the events according to their Encounter Probability. It supports different events such as a vehicle leaving a parking space (i.e., an event relevant for all the vehicles that are close to that space during a given period of time), and an emergency braking (only relevant for the vehicles following the vehicles generating that event).

Our VESPA prototype runs on PDAs equipped with embedded GPS receivers. The dissemination protocol presented in Section 4 relies on Wi-fi communications to support the exchanges between the vehicles. Using our prototype, a driver can receive on her/his PDA information about events transmitted by neighboring vehicles. As described in Figure 3, s/he can basically observe the type of event (e.g., an available parking space, an accident, etc.), the distance between his/her car and the event, and an arrow indicating the direction to follow to reach the event.

In Figure 4, we present an example describing how a driver can access the optional information attached to an event in the case of a parking space event. The first screen presents our prototype interface when waiting for potential events. The second screen shows the basic information printed on reception of an available parking space event. The driver can also watch the optional attributes, which may correspond to multimedia data such as a picture in our example. As concerns the exchange of multimedia data through the vehicular ad hoc networks, we only manage to exchange small files (i.e., a few KBytes). Indeed, we do not use any antenna with the smartphones yet and the communication range is about 100 m, which limits the duration of the exchanges between the vehicles, in particular when they move in different directions.

[7] The number of vehicles used during our field tests remain limited for the moment.



Time of arrival	10:08
For disabled	No
Cost	2 € / h

Figure 4: Information about an available parking space event

Let us note that in real conditions the generation of many events could be initiated using the numerous sensors embedded in our cars (for example, by coupling the airbag system with the creation of an event representing an accident). Since the smartphones used are not yet connected to these sensors (i.e., to the CAN bus of the vehicle), the generation of the events is managed by the device (using the GPS signal) in the current version of our prototype.

6.2 Architecture

The architecture of VESPA, which is deployed on every equipped vehicle, is presented in Figure 5, where the following main elements can be distinguished:

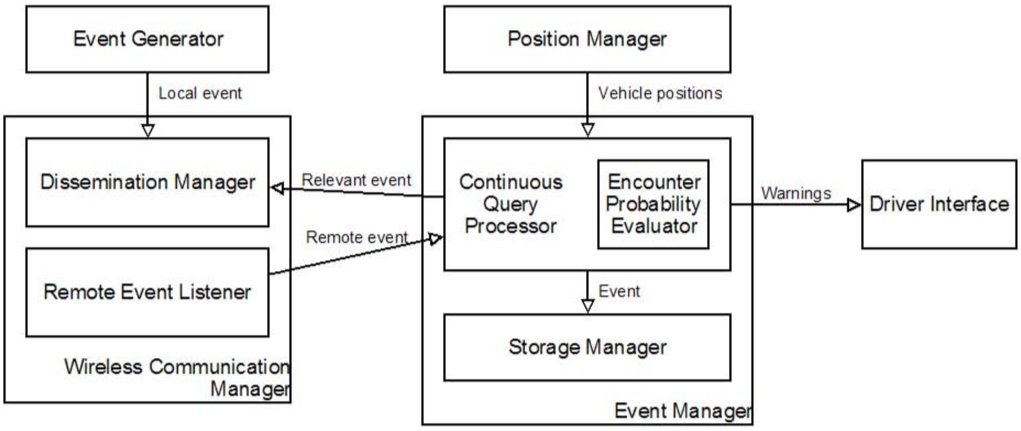


Figure 5: VESPA architecture

- The *Wireless Communication Manager* is in charge of the reception and transmission of events. This module is composed by the *Dissemination Manager*, which allows the vehicle to broadcast events, and the *Remote Event Listener*, in charge of the reception of the events transmitted by the neighboring vehicles.

- The *Event Manager* handles the events received by the vehicle. It is composed of the *Continuous Query Processor*, which processes the active continuous queries to determine not only whether the vehicle is going to encounter the event or not (by using an *Encounter Probability Evaluator*) but also if the driver is interested or not in that event. Finally, the *Storage Manager* is in charge of deciding about the storage and removal of events.

- The *Driver Interface* is the graphical user interface used to interact with the driver (e.g., showing information about events detected).

- The *Position Manager* interacts with the GPS receiver of the vehicle to retrieve information regarding the location of the vehicle.

- Finally, the *Event Generator* releases events detected by the vehicle. The generation of many events could be initiated using the numerous sensors embedded in modern cars (for example, by coupling the airbag system with the creation of an event representing an accident) or via other static information sources (e.g., sensors on a road). Specific Human Machine Interface (HMI) aspects are not considered in this article.

In the following, we briefly explain the way the different modules interact:

1. An event is first received by the Continuous Query Processor. The Encounter Probability Evaluator computes the encounter probability of the event using the information provided by the Position Manager. The relevance of an event may change continuously due to the different dynamic factors affecting the computation of the encounter probability, such as the distance to the event (as explained in Section 2). Therefore, the Continuous Query Processor, using the Encounter Probability Evaluator, evaluates periodically the active continuous queries to verify which events must be reported to the driver through the Driver Interface. Moreover, a new event received could also be processed immediately if its *Importance* field has a high value (e.g., for accidents or emergency braking situations). Each event for which the encounter probability is higher than a *relevance threshold* must be checked against the set of active continuous queries.

2. The Storage Manager is informed by the Encounter Probability Evaluator about the encounter probabilities it computes. If the encounter probability of a previously stored event is smaller than a *storage threshold*, then the Storage Manager removes the event from the storage area. On the contrary, if the encounter probability of a new event is greater than the storage threshold, the event is stored.

3. For a new event received, in case its encounter probability is higher than the diffusion threshold, the Dissemination Manager is contacted by the Continuous Query Processor to broadcast the event and inform other vehicles.

7 Related Works

Numerous works have addressed communication protocols in vehicular networks, either in the context of geocasting protocols (for example, [NI97, KV99]), whose goal is to transmit data to all the targets within an area, or in the context of dissemination protocols [NSI06, FMH$^+$02, LSCM07, XOW04, NDK04b].

However, those works consider very small messages exchanged between vehicles not compatible with the use of multimedia data. Up to our knowledge, the only work that considers multimedia data in vehicular networks is [GAZ05], where an architecture to provide live video streaming to vehicles is presented (*V3*). Vehicles within an interesting region are assumed to be able to capture video data from that region. In this approach, two areas are considered: the *trigger message forwarding zone* (*TMFZone*, which is the area within which a query must be forwarded) and the *data forwarding zone* (*DFZ*, which is the area from where video data must be forwarded to the interested vehicle). Both the case where there is a single receiving vehicle and the case with multiple receivers (where some optimizations are possible by using multicast, in order to reduce sending duplicate packets) are considered. A store-carry-and-forward approach is proposed, and several algorithms are analyzed by considering the tradeoff between transmission delay and bandwidth overhead. For example, one of the algorithms proposed to select a data forwarder is the *optimal selection* where, assuming that each vehicle knows the mobility function of other vehicles within its communication range, the vehicle that could reach the receiver (or the next network partition) sooner is selected.

8 Conclusion & Perspectives

In this paper, we have presented the basic functionalities of VESPA and discussed improvements in the representation of events. Particularly, we have highlighted the interest of using multimedia data. VESPA can be seen as a system complementary to existing navigation systems. Indeed, whereas navigation systems can be used to guide drivers and show them the location of different points of interest (e.g., petrol stations, railway stations, airports, etc.), VESPA can provide them ephemeral information about the road hazards they may encounter along their displacement (e.g., information about an emergency braking, an available parking space, etc.).

Our current work is to evaluate VESPA in real conditions using our prototype. We are also studying how to improve it. Therefore, we are working on the aggregation of the events received by a vehicle. Our goal is to extract additional knowledge to be used by the drivers. For example, using the summaries generated with the available parking spaces [DDM$^+$08], it becomes possible to determine the areas where the probability to find an available parking space is high.

Acknowledgements

The present research work has been supported by the International Campus on Safety and Intermodality in Transportation the Nord-Pas-de-Calais Region, the European Community, the Regional Delegation for Research and Technology, the Ministry of Higher Education and Research, and the National Center for Scientific Research. The authors gratefully acknowledge the support of these institutions.

References

[CDI08] N. Cenerario, T. Delot und S. Ilarri. Dissemination of information in inter-vehicle ad hoc networks. In *Intelligent Vehicles Symposium (IV'08)*, 2008.

[DCI08] T. Delot, N. Cenerario und S. Ilarri. Estimating the relevance of information in inter-vehicle ad hoc networks. In *IEEE International Conference on Mobile Data Management (MDM'08) - Workshops*, 2008.

[DDM$^+$08] B. Defude, T. Delot, J.L. Zechinelli Martini, N. Cenerario und S. Ilarri. Extraction de connaissances dans les réseaux ad hoc inter-véhicules. In *ACM conf. on Mobilité & Ubiquité (Ubimob'08)*, May 2008.

[FFH$^+$02] A. Festag, H. Füßler, H. Hartenstein, A. Sarma und R. Schmitz. Fleetnet: Bringing car-to-car communication into the realworld. In *World Congress on Intelligent Transport Systems (ITS)*, 2002.

[FMH$^+$02] H. Füβler, M. Mauve, H. Hartenstein, M. Käsemann und D. Vollmer. A Comparison of Routing Strategies for Vehicular Ad Hoc Networks. Bericht TR-02-003, Department of Computer Science, University of Mannheim, July 2002.

[GAZ05] Meng Guo, Mostafa H. Ammar und Ellen W. Zegura. V3: A vehicle-to-vehicle live video streaming architecture. *Pervasive and Mobile Computing*, 1(4):404–424, December 2005.

[KV99] Y.-B. Ko und N. H. Vaidya. Geocasting in Mobile Ad Hoc Networks: Location-Based Multicast Algorithms. In *Second IEEE Workshop on Mobile Computer Systems and Applications (WMCSA'99)*, Seite 101. IEEE Computer Society, October 1999.

[LH05] J. Luo und J-P. Hubaux. A Survey of Research in Inter-Vehicle Communications. In *Embedded Security in Cars - Securing Current and Future Automotive IT Applications*. Springer-Verlag, 2005.

[LSCM07] C. Lochert, B. Scheuermann, M. Caliskan und M. Mauve. The Feasibility of Information Dissemination in Vehicular Ad-Hoc Networks. In *Conf. on Wireless On demand Network Systems and Services (WONS'07)*, 2007.

[MHD$^+$03] P. Morsink, R. Hallouzi, I. Dagli, C. Cseh, L. Schafers, M. Nelisse und D. De Bruin. CARTALK 2000: Development of a cooperative ADAS based on vehicle-to-vehicle communication. In *Intelligent Transport Systems and Services*, 2003.

[NDK04a] S. Nittel, M. Duckham und L. Kulik. Information Dissemination in Mobile Ad-hoc Geosensor Networks. In *3rd Int. Conf. on Geographic Information Science*, 2004.

[NDK04b] S. Nittel, M. Duckham und L. Kulik. Information Dissemination in Mobile Ad-Hoc Geosensor Networks. In *3rd International Conference on Geographic Information Science (GIScience'04)*, Seiten 206–222, 2004.

[NDLI04] T. Nadeem, S. Dashtinezhad, C. Liao und L. Iftode. TrafficView: Traffic Data Dissemination Using Car-to-Car Communication. *ACM Sigmobile Mobile Computing and Communications Review, Special Issue on Mobile Data Management*, 8(3):6–19, 2004.

[NI97] J. C. Navas und T. Imielinski. GeoCast – geographic addressing and routing. In *Third Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'97)*, Seiten 66–76. ACM, September 1997.

[NSI06] T. Nadeem, P. Shankar und L. Iftode. A Comparative Study of Data Dissemination Models for VANETs. In *3rd Int. Conf. on Mobile and Ubiquitous Systems (MOBIQUITOUS'06) - Workshops*, 2006.

[XOW04] B. Xu, A. M. Ouksel und O. Wolfson. Opportunistic Resource Exchange in Inter-Vehicle Ad-Hoc Networks. In *5th Int. Conf. on Mobile Data Management*, 2004.

A Multimedia Service with MPEG-7 Metadata and Context Semantics

Yiwei Cao, Ralf Klamma, and Maziar Khodaei

Lehrstuhl für Informatik 5
RWTH Aachen University
Ahornstr. 55
52056 Aachen Germany
{cao|klamma|khodaei}@dbis.rwth-aachen.de

Abstract: With the emergent and rapid development of mobile technologies, more and more multimedia applications run on handheld devices in mobile networks. This raises new challenges to mobile multimedia information systems. In this paper three technical aspects are considered as tightly intertwining and crucial for developing mobile multimedia applications: context awareness, multimedia adaptation, and mobility. An approach to integrating multimedia metadata standards such as MPEG-7 and MPEG-21 into context-aware ontologies effectively and seamlessly is proposed and related algorithms are extended. A context-aware mobile multimedia application (CA3M) is implemented and evaluated as a proof of concept. The evaluation results of the prototypes show the soundness of the concept. Open issues and development potentials for further research in the area of context-aware mobile multimedia information system development are addressed.

1 Introduction

Mobile applications have been developed with advancement and rapidly in recent years. Mobile devices like iPhones make a revolution of mobile application development. The conventional barriers of mobile application development like limited capacities have made great progress. It is quite normal to have an 8 GB, 16 GB hard disk or even larger. The display sizes have also been maximized within the device size available. All that innovative progress makes mobile multimedia applications more and more attractive and important for users.

Nowadays mobile devices and applications are featured with new focuses such as integrity, multimodality, “one device fitting all” or “only carrying one gadget at a time” etc. Multimodality is realized both in information input and output channels. Google supports speech-based keyword search, while iPhone has three built-in sensors, the accelerometer, the proximity sensor, and the ambient light sensor, to detect movements even movement intentions of the mobile device, besides GPS and camera etc. Making a call is becoming the side function of cell phones, because cell phones are also MP3-players, video players, cameras, recorders, gaming-consoles, digital books, personal digital assistants to organize contacts and appointments, and have much other functionality.

A recent report shows that the variety of functionality available on cell phones raises usage complexity as well. And many users still only stick to the phone call function of their cell phones[1] . Thus, the intuitiveness of the mobile user interfaces is still limited. Mobile multimedia information systems with enhanced context awareness are the potential approaches to reduce the complicated user device interaction.

In addition, fault tolerance and speed are identified as the most critical aspects for general multimedia applications, because both audiovisual media and metadata including control information are processed simultaneously [24]. Hence, we consider the context uncertainty problems as an important aspect at mobile multimedia application development. Furthermore, three technical aspects are considered as tightly intertwining and crucial for developing mobile multimedia applications: context awareness, multimedia adaptation, and mobility. Uncertainty reasoning is based on context reasoning which is enabled both by metadata based multimedia adaptation and context-aware multimedia adaptation (cf. Figure 1).

The rest of the paper is organized as follows. The state-of-the-art technologies are discussed in Section 2. An approach to integrating multimedia metadata standards such as MPEG-7 and MPEG-21 into context-aware ontologies effectively and seamlessly is proposed in Section 3. Section 4 provides an insight into implementation and evaluation of a context-aware mobile multimedia application (CA3M). CA3M delivers a service of context-aware multimedia search and multimedia adaptation which can be accessed from mobile devices. Section 5 concludes this paper with open issues and development potentials for further research in the area of context-aware mobile multimedia information system development.

2 Related work

Much research work has been done in the area of context modeling, context-aware information systems, mobile computing, and multimedia systems, multimedia metadata standards, and ontologies. Research progresses in these areas contribute to the ubiquitous computing as well as pervasive computing paradigms. The context information might be uncertain which needs to be reduced via an appropriate context model.

[1] Rhizomik Initiative, http://rhizomik.net.

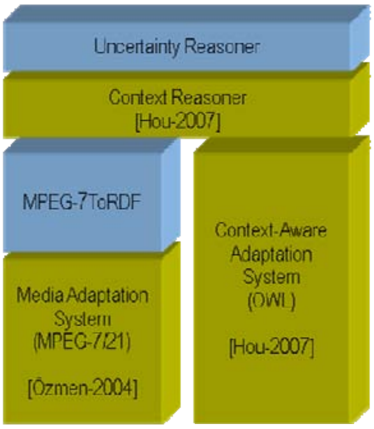


Figure 1: Conceptual approach

2.1 Context awareness, mobility and multimedia adaptation

First of all, these three main concepts are reviewed from related literature. Any piece of information for interaction between user communities and applications can be observed as a context. Dey defines context as piece of information characterizing certain entity including person, location, or a physically assessable object [7]. Correspondingly, context-aware systems refer to any information system that deals with and makes use of context information. Representation of context information is an important part of research in pervasive computing and considered as a sub field of knowledge engineering, within knowledge management and artificial intelligence. Context-aware mobile systems can be applied in such fields as presenting context information to users, services adaptation to mobile users, as well as handing context information at real time [25].

Mobility assures mobile device users of reaching information according to certain context such as user communities, location, temporal and device capacities anywhere at any time [19]. Kakihara and Sorensen associated mobility with three dimensions which act as a whole: spatial, temporal, and context-dependent [11]. Among them, multimedia content is of high interest. Multimedia refers to content and data from more than one digital resource such as text, photographs, graphics, animation, audio, and video etc. [24]. Multimedia annotation, adaptation and different interactions are main aspects for multimedia applications. In order to deliver right information to right person at right time and right location, multimedia search results are adapted to users' preferences and up-to-data context information in certain environments.

2.2 Context uncertainty

It is difficult to avoid the problems with data uncertainty especially in context-aware systems. Inconsistency occurs between models and the real world, as well as among different local environment models. Four main factors are stated to lead to that inconsistency in [9]:

- *unknown* when no information about the property is available;
- *ambiguous* when several different reports about the property are available (for example, when two distinct location readings for a given person are supplied by separate positioning devices);
- *imprecise* when the reported state is a correct yet inexact approximation of the true state (for example, when a person's location is known to be within a limited region, but the position within this region cannot be pinpointed to the required (application-determined) degree of precision); or
- *erroneous* when there is a mismatch between the actual and reported states of the property.

Correspondingly, Quality of Context (QoC) or Quality of Information is often discussed in these cases and quality of context information depends greatly on the change of context sources, by which context is provided [14].

2.3 Context modeling

In order to deal with the context information as well as context uncertainty mentioned before, specific frameworks and data structures are needed to capture, manage, process and retrieve context information. In the field of knowledge engineering and artificial intelligence, context information must be at first collected and presented to the application in order to enable efficient context-aware adaptation. Therefore, a common representation format for the context information is required [17]. A well-defined context model is needed to define and store context data in machine readable forms in order to enhance interoperability. In [23] Strang and Linnhoff-Popien made an in-depth survey across several context-aware systems and compared the most relevant context modeling approaches including Key-Value, Markup Scheme, Graphical, OO, Logic-based and Ontology-based models. These approaches are based on different data structures and represent context information for machine processing and reasoning.

2.4 MPEG-7, MPEG-21 and RDF

With regard to various multimedia metadata, MPEG-7 provides a large set of pre-defined elements to describe multimedia contents. In particular, these elements are composed of two different types: Description Schemes (DS) and Descriptors (D). Depending on the fields of applications, a specific description scheme can be defined by freely combinable descriptors (i.e. tags). Each descriptor itself refers to a specific feature or attribute of multimedia content [12, 15]. In addition, the Description Definition Language (DDL) of MPEG-7 makes this standard more powerful than other metadata standards, since it allows the creation of new descriptors and description schemes within the standard. Hence, it makes the vocabulary of MPEG-7 to be extensible by employing the XMLSchema.

Together with MPEG-7, MPEG-21 provides a comprehensive framework for multimedia adaptation. In general, MPEG-21 consists of twelve parts [2, 12]. These parts are independent of each other. So the excerpts of the standard might also be applied. Alone for the purpose of managing multimedia contents in mobile end devices we confine the system to the MPEG-21 Digital Item Adaptation (DIA) and MPEG-21 Digital Item Declaration Language (DIDL). The MPEG-21 DIDL contains six top level descriptors (tags), of which the first four are particularly important for mobile data management.

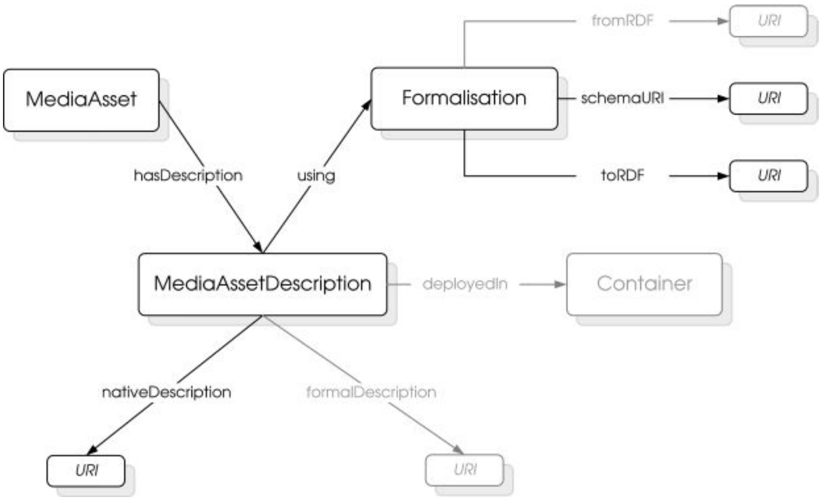


Figure 2: The ramm.x vocabulary [16]

However, neither MPEG-7 nor MPEG-21 multimedia metadata standard defines ontology which can be used for context modeling. Several ontology-based MPEG binding specifications are surveyed in the related research fields.

RDFa-deployed Multimedia Metadata (ramm.x) offers mapping from multimedia standards such as MPEG-7 to the Semantic Web whose documents are often described in RDF (Resource Description Framework). The ramm.x can be used for reuse of existing multimedia metadata, converting, validation and exchange of metadata via an easy-to-use vocabulary set (cf. Figure 2). Main requirements of ramm.x include enabling the resulting description to be consumable by a Semantic Web agent through being encoded in RDF; embedding reference to description in existing multimedia metadata format in (X)HTML; providing reference to services capable of mapping between a specific multimedia metadata format and RDF; and multiple descriptions available for one media asset (e.g. in different formats, covering different aspects), etc. Based on the ramm.x, an MPEG-7 ontology demonstrates the first successful practical realization [18] with MPEG-7 schema version of 2001. The MPEG-7 ontology specified by Rhizomik [18] shows its advantage in comparison to those approaches like MPEG-7 Ontology [10] Core Ontology for Multimedia (COMM) [4, 5] and aceMedia Visual Descriptor Ontology (aceMedia VDO) [1] (cf. Table 1).

Framework	Supporting format	Mapping	Comments
MPEG-7 Ontology [10]	OWL-full	-	-
COMM [4]	OWL-DL	-	-
aceMedia [1]	RDFS-DS	-	-
Rhizomik MPEG-7 Ontology [18]	OWL-full	XSD2OWL, XML2RDF	Confidence value can be set for uncertainty reasoning

Table 1 A comparison of MPEG-7 ontologies

3 System design of a mobile context-aware multimedia service

A service is designed and implemented to meet the following requirements. How can preferences of user communities, spatial and temporal context information contribute to the context-aware multimedia search? How can Semantic Web and multimedia metadata standards enhance multimedia search together?

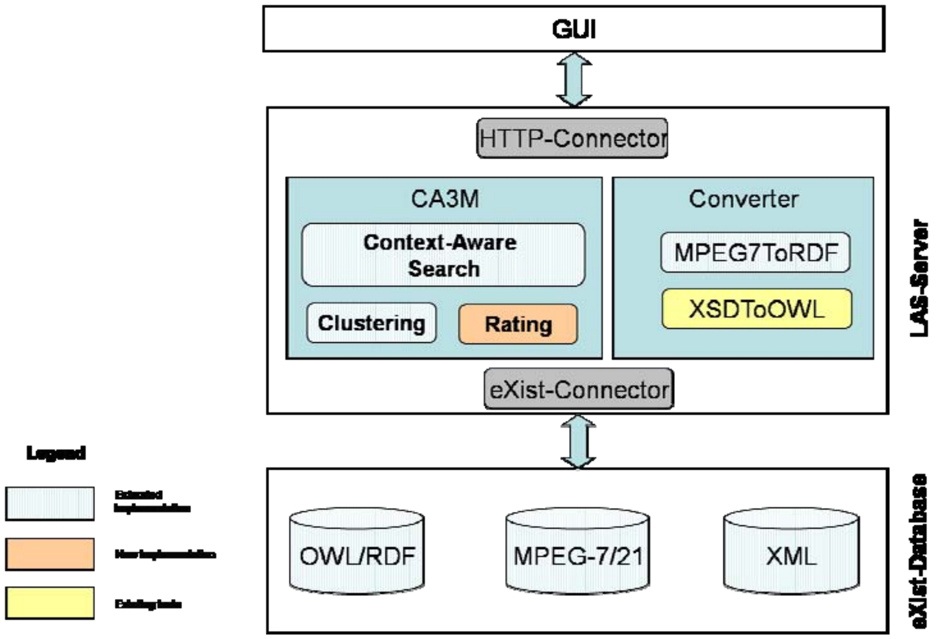


Figure 3: CA3M system architecture

3.1 Conceptual approach of CA3M

The CA3M (Context-Aware Mobile Multimedia) service enables multimedia search based on keywords specified by users and users' context. CA3M uses the Web Service technologies in order to provide easy access to mobile applications on the client side. On the server side, a set of services provide functionality such as context acquirement, context reasoning, and context querying. On the client side, mobile user interfaces can process users' requests and retrieve demanding multimedia results. For rapid prototyping we employ our previous research result, the Lightweight Application Server (LAS) [20] as the basic framework. LAS provides HTTP and SOAP connectors, which makes the implementation of client-server communication easier. In addition, the LAS components provide functionality such as the connector to a database. Ontologies using OWL, RDF and a media repository with metadata are maintained by a native XML database, e.g. eXist [8]. Moreover, the main benefit of LAS is that the services on it can be flexibly extended for specific applications by using the LAS Java APIs (cf. Figure 3).

Thus, CA3M is designed and implemented as LAS services within the LAS framework. Figure 4 illustrates the information flow in CA3M. Besides Context reasoning and multimedia search functionality, rating and clustering are provided. As the first step, user communities are grouped into different clusters according to their own interests or preferences. Users' rating to multimedia search results is carried out based on different user clusters and individual users. The rating mechanism by user communities can reduce context uncertainty via users' feedback to certain multimedia search results.

3.2 The MPEG-7 to RDF converter

The binding of MPEG-7 and Semantic Web encoded in RDF is one of the basic components beside the CA3M service. The contribution of this research work is mainly in two aspects. One is the adapted converting service which enables MPEG-7 and MPEG-21 multimedia standards to be understandable in order to realize the Semantic Web. This is an extension of the existing research by the Rhizomik Initiative[1]. The converter consists of transformation from RDF to HTML, XML Schema to OWL, and XML to RDF. An ontology schema based on a mapping of MPEG-7 schema to RDF is additionally specified.

The other contribution is that CA3M supports both XQuery (XML Query Language) and SPARQL (Simple Protocol and RDF Query Language). SPARQL is a tree-structured query language for RDF documents. The semantics of SPARQL is similar to SQL, while the query processing mechanisms of both are different [21, 22].

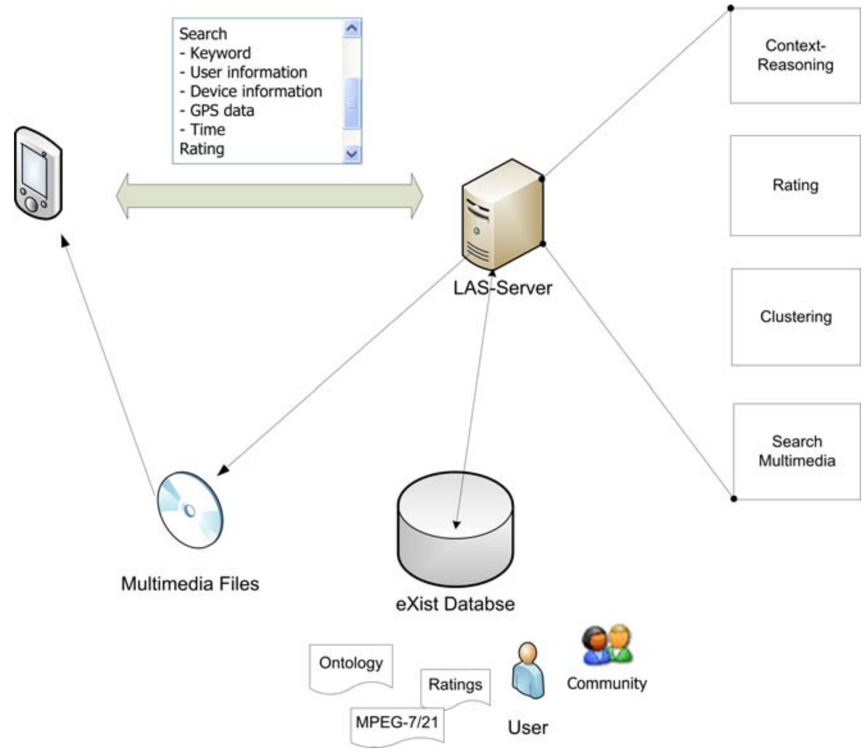


Figure 4: CA3M information flow

4 Implementation and evaluation

MPEG-7 Schema is mapped to ontology before MPEG-7 documents are transformed into RDF. Our converter is adapted and realized for the latest MPEG-7 Schema of 2004. For the context-aware query processing, two steps are carried out sequentially. Firstly, all queries related to context information are expressed and evaluated in XQuery. The to-be-searched multimedia metadata at this phase is in MPEG-21 DIDL format. Second, SPARQL queries are prepared for executing on RDF documents, after the converter has processed the metadata documents.

The MPEG-7 to RDF converter works within two packages, *XSD2OWL* and *XML2RDF*. XSD2OWL facilitates the transformation of MPEG-7 Schema into MPEG-7 Ontology, while XML2RDF transforms XML based MPEG-7 documents into RDF formats. A LAS service called MPEG7ToRDF Service has been implemented.

A set of evaluation work is done on the CA3M service prototype. In our project Virtual Campfire [6], there are a great amount of multimedia files with metadata stored on streaming servers, FTP servers, HTTP servers, and several multimedia or XML databases in a distributed computing environment. Over 300 main multimedia metadata files are converted from MPEG-7 into RDF. Then SPARQL queries are executed to obtain context-aware multimedia search results across the 300 MPEG-7 documents and their related MPEG-7 documents which e.g. save information about location using MPEG-7 *SemanticPlaceType*. The first evaluation result shows that more efforts could be made to optimize the service execution, because around 30 seconds are needed to handle the SPARQL queries at the current development stage.

5 Conclusions and outlook

Context awareness, mobility and multimedia adaptation are discussed as three main factors to enable context-aware multimedia adaptation and search on mobile platforms. This work is based on two accomplished prototypes of multimedia information systems with metadata standards based multimedia adaptation and context modeling and reasoning in the previous work [13, 3]. To make good use of and to advance the existing tools, a conceptual approach is proposed to enhance multimedia adaptation and search combining comprehensive multimedia standards such as MPEG-7/21 and context modeling and context awareness. Both technologies intertwine and are able to support context reasoning in a better way. Furthermore, they make it possible and promising to shape and perform measurements in order to reduce context uncertainty.

In our ongoing research, we will focus on mobility issues and context uncertain problems related to multimedia and context awareness. Within the UMIC (Ultra High-speed Mobile Information and Communication) research cluster of the German Excellence Initiative, a lot of research work for mobile context-aware multimedia services is challenging. P2P data management with regard to MPEG-7/21 metadata standards will be addressed. Data uncertainty and context uncertainty problems need to be in-depth identified, analyzed and handled. Mobile (Web) services will be developed to bridge the mobile social software on the higher application layer and the mobile wireless network technologies on the lower network layer. The performance of these services should also be improved.

Acknowledgment

This work has been supported by the UMIC Research Centre, RWTH Aachen University. We would like to thank our colleagues for the fruitful discussions.

Reference

[1] Ace Media Project, http://www.acemedia.org/aceMedia, {December 2008}.

[2] Burnett, I.; Van de Walle, R.; Hill, K.; Bormans, J.; Pereira, F.: MPEG-21: goals and achievements. IEEE Multimedia, 10(4): 60–70, 2003.

[3] Cao, Y.; Klamma, R.; Hou, M.; Jarke, M.: Follow Me, Follow You - Spatiotemporal Community Context Modeling and Adaptation for Mobile Information Systems, In: Proc. of the 9th International Conference on Mobile Data Management, April 27-30, 2008, Beijing, China, pp. 108-115.

[4] COMM: Core Ontology for Multimedia, http://comm.semanticweb.org/, {December, 2008}.

[5] Bloehdorn, S.; Petridis, K.; Saathoff, C.; Simou, N.; Tzouvaras, V.; Avrithis, Y.; Handschuh, S.; Kompatsiaris, Y.; Staab, S.; Strintzis, M. G.: Semantic Annotation of Images and Videos for Multimedia Analysis. In: Proceedings of the Second European Semantic Web Conference (ESWC 2005), Springer, 2005, pp. 592-607.

[6] Cao, Y.; Spaniol, M.; Klamma, R.; Renzel, D.: Virtual Campfire - A Mobile Social Software for Cross-Media Communities, K. Tochtermann, H. Maurer, F. Kappe, A. Scharl (Eds.): Proceedings of I-Media'07, International Conference on New Media Technology and Semantic Systems, Graz, Austria, September 5 - 7, 2007, J.UCS (Journal of Universal Computer Science) Proceedings, 2007, pp. 192-195.

[7] Dey, A. K.; Abowd, G. D.: Towards a Better Understanding of Context and Context-Awareness. In: HUC '99: Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing Bd. 1707. London, UK: Springer, 1999; pp. 304-307.

[8] eXist, Open Source Native XML Database, http://exist.sourceforge.net/, {December 2008}.

[9] Henricksen, K.; Indulska, J.: Modelling and Using Imperfect Context Information, In: PERCOMW 04: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops. Washington, DC, USA: IEEE Computer Society, 2004, pp. 33-37.

[10] Hunter, J.: Adding Multimedia to the Semantic Web - Building an MPEG-7 Ontology. In: Proceedings of the First Semantic Web Working Symposium (SWWS), Stanford, USA, 2001, pp. 261-281.

[11] Kakihara, M.; Sorensen, C.: Mobility: an extended perspective. In: Proceedings of the Hawaii International Conference on System Sciences, New York, NY, USA: IEEE Computer Society, January 2002; pp. 1756-1766.

[12] Kosch, H.: Distributed Multimedia Database Technologies Supported by MPEG-7 and MPEG-21, CRC Press, 2003.

[13] Klamma, R.; Spaniol, M.; Cao, Y.: Community Aware Content Adaptation for Mobile Technology Enhanced Learning, In: W. Nejdl, K. Tochtermann (Eds.): Innovative Approaches to Learning and Knowledge Sharing, Proceedings of the 1st European Conference on Technology Enhanced Learning (EC-TEL 2006), Hersonissou, Greece, October 1-3, LNCS 4227, Springer-Verlag, 2006, pp. 227-241.

[14] Lei, H.; Sow, D. M.; Davis, J. S.; Banavar, G.; Ebling, M. R.: The design and applications of a context service. In: SIGMOBILE Mobile Computing Communication, Rev. 6 (2002), No. 4, pp. 45-55.

[15] Martinez, J. M.; Gonzalez, C.; Fernandez, O.; Garcia, C.; de Ramon, J.: Towards universal access to content using MPEG-7, In: Proceedings of the 10th ACM International Conference on Multimedia, ACM Press, 2002, pp. 199–202.

[16] ramm.x: RDFa-deployed Multimedia Metadata. http://sw.joanneum.at/rammx/, {December 2008}.

[17] Rothermel, K.; Bauer, M.; Becker, C.: Sonderforschungsbereich 627: Nexus - Umgebungsmodelle für mobile kontextbezogene Systeme. it - Information Technology 45(5): 293-; 2003.

[18] Roberto Garcia Gonzalez @ Rhizomik. http://rhizomik.net/~roberto/, {December 2008}.

[19] Roy, N. L. S.; Scheepers, H.; Kendall, E.; Saliba, A.: A comprehensive model incorporating mobile context to design for mobile use. In: Proceedings of the 5th Conference on Human Computer Interaction in Southern Africa, January, 2006; pp. 22-30.

[20] Spaniol, M.; Klamma, R.; Janßen, H.; Renzel, D.: LAS: A Lightweight Application Server for MPEG-7 Services in Community Engines, In: Proceedings of I-KNOW ’06, 6th International Conference on Knowledge Management, Graz, Austria, September 6 - 8, ser. J.UCS (Journal of Universal Computer Science) Proceedings, K. Tochtermann and H. Maurer, Eds. Springer-Verlag, 2006, pp. 592–599.

[21] SPARQL Query Language for RDF. http://www.w3.org/TR/rdf-sparql-query/, {December 2008}.

[22] SPARQL Tutorial. http://jena.sourceforge.net/ARQ/Tutorial/, {December 2008}.

[23] Strang T.; Linnhoff-Popien, C.: A Context Modeling Survey, In: First International Workshop on Advanced Context Modelling, Reasoning and Management at UbiComp, Nottingham, UK, September 2004.

[24] Steinmetz, R.; Nahrstedt, K.: Multimedia Systems, Springer-Verlag, 2004.

[25] Zhang, D.; Gu, T.; Pung, H.: A Middleware for Building Context-Aware Mobile Services. In: Proceedings of IEEE Vehicular Technology Conference, 2004.

A Standards-Based Generic Approach for Complex Multimedia Management

Anna Carreras[1], Ruben Tous[1], Eva Rodríguez[1], Jaime Delgado[1], Giovanni Cordara[2], Gianluca Francini[2], Diego Gibellino[2]

[1] Universitat Politècnica de Catalunya, Departament d'Arquitectura de Computadors,
Campus Nord, Mòdul D6, Jordi Girona 1-3, E-08034 Barcelona, Spain
{annac, rtous, evar, jaime.delgado}@ac.upc.edu
[2] Telecom Italia Lab, Via Reiss Romoli, 274 - 10148 Torino, Italy
{giovanni.cordara, gianluca.francini, diego.gibellino}@telecomitalia.it

Abstract. This paper presents a standards-based architecture for a complex and generic distributed multimedia scenario, which combines content search and retrieval, DRM, and context-based content adaptation together. It is an innovative and totally generic approach trying to narrow the semantic-gap by integrating a flexible language for multimedia search based on MPEG Query Format (MPQF) standard with the application of video analysis algorithms for the automatic extraction of low-level features and with the use of contextual information.

Keywords: Context-based adaptation, Digital Rights Management-Rights Expression Language, Features extraction, MPEG Query Format.

1 Introduction

A substantial amount of work has been done in the area of Universal Multimedia Access (UMA) and the most recent works focus on the maximisation of user experience. Nevertheless, these approaches are usually application-specific and it is easy to identify serious limitations in terms of interoperability and extensibility. The majority of the research activities reported in this area focus mainly on content adaptation [1], where the use of contextual information and metadata is essential to achieve efficient and useful adaptations that enrich the user experience. Furthermore, the rising tide of available content has created the need of new tools for guiding users to be able to search and find what is of their interest. In both (content adaptation and search) applications, similar problems need to be addressed:

- The lack of context and metadata textual descriptors.
- The semantic gap, i.e. the gap between the low-level features (LLFs) that can be automatically extracted from digital contents and the interpretation that a user would have of the same content.
- The use of standards.

The affordability of consumer electronic devices such as MP3 and recordable players, digital cameras, etc., allow users to become content producers as well as consumers. This evolution creates new and interesting challenges: the abovementioned issues are tightly related to the industry exploitation of new search and retrieval solutions, able to address the specific requirements of the emerging social networks featuring tons of user generated content (UGC). The combination of techniques taking advantage of automatically extracted features, textual metadata and context information can represent the key for the success of such services, offering simple and seamless management of personal digital media repositories in the home network or on the big Internet, as well as Premium content catalogue browsing and search features.

Furthermore, a multipurpose framework dealing with heterogeneous contents needs to take into account the enforcement of Digital Rights Management (DRM) technologies during content access and consumption. Such a feature represents a key issue for business models design, ensuring a transparent and correct usage of the content throughout each stage of the value chain (from the content creator, through the service provider, and to the end user).

Following the background information about the different lines of research activities integrated in the theme of the proposed work (Section 2), this paper will address all the identified challenges by presenting a standards-based architecture for a complex and generic distributed multimedia scenario, which combines search and retrieval, DRM and context-based content adaptation (Section 3). Finally, before the conclusions and the future work, an application scenario based on Social Networks is described in order to better evaluate our proposal (Section 4).

2 Background

2.1 Multimedia Search and Retrieval

In this section we will first analyse the requirements of today's multimedia search and retrieval services, and present the MPEG Query Format (MPQF) as the most suitable solution as it satisfies those requirements. Finally, we will also refer to different search and retrievals algorithms based on video processing techniques.

Unified Querying Languages and Interfaces. The first thing to take into account when defining a search and retrieval service is that user information needs can be expressed in many different ways. On the one hand, when search preferences can be expressed in terms of precise conditions as those in a relational algebra expression, clearly determining which objects of collection to select, it is known as data retrieval (DR). In this case, a single erroneous object among a thousand retrieved objects means a total failure. In the context of multimedia search and retrieval, DR refers to queries expressed in terms of metadata and also in terms of low-level features. On the other hand, there are user information needs which cannot be easily formalized. Information retrieval (IR) aims to retrieve information which might be relevant to the user, given a query written from the user's point of view. In the context of multimedia search and retrieval, IR refers to text keywords and query-by-example (QBE) for instance.

Querying today's digital contents can imply the combination of data retrieval-like conditions referred to a well-defined data model and also information retrieval-like conditions.

Many modern multimedia databases (MMDBs) and various providers of multimedia search and retrieval services already offer advanced indexing and retrieval techniques for multimedia contents. However, their databases and service interfaces are proprietary, and therefore the solutions differ and do not interoperate.

Our proposed search and retrieval service is based on MPEG Query Format (MPQF) standard in order to guarantee the interoperability needed to ease the access to repositories by users and applications, and to allow the deployment of distributed search and aggregation services.

MPEG Query Format Overview. The MPEG Query Format (MPQF) is Part 12 of ISO/IEC 15938-12, "Information Technology - Multimedia Content Description Interface", better known as MPEG-7. The standardization process in this area started in 2006, and MPQF became an ISO/IEC final standard after the 85th MPEG meeting in July 2008.

MPQF is an XML-based query language that defines the format of queries and replies to be interchanged between clients and servers in a distributed multimedia information search-and-retrieval context. The two main benefits of standardising such kind of a language are 1) interoperability between parties (e.g., content providers, aggregators and user agents) and 2) platform independence (developers can write their applications involving multimedia queries independently of the database used, which fosters software reusability and maintainability). The major advantage of having MPEG rather than industry forums leading this initiative is that MPEG specifies international, open standards targeting all possible application domains, which are not conditioned by partial interests or restrictions.

Multimedia Search and Retrieval Techniques. Video data can be indexed based on its audiovisual content (such as colour, speech, motion, shape, and intensity), and semantic content in the form of text annotations. Because machine understanding of the video data is still an unsolved research problem, text annotations are often used to describe the content of video data according to the annotator's understanding and the purpose of that video data.

As far as indexing and retrieval techniques for the visual content are concerned, content-based solutions propose a set of methods based on low-level features such, for example, colours and textures. Several frameworks dealing with the automatic extraction of low level features have been proposed [3]; their main disadvantage, however, relies in the impossibility for such systems to process complex queries to express high level semantic concepts, like, for example "find a video of my sister on a beach at the sunset". Some recent technologies allow the indexing of content based on high level concepts through specific algorithms, but the categorization is limited to few concepts due to the implicit constraints imposed by those algorithms. In TRECVID 2007, the search task consisted of finding shots in a test collection satisfying queries expressed by *topics* – a kind of complex high level features. Examples of such topics are: "waterfront with water and buildings" and "street protest or parade”. This TRECVID contest regarded 24 topics. The recall and precision values are still quite modest, as compared to other information retrieval scenarios.

Some opportunities to improve those systems can be offered by the combination of signal and symbolic characterizations in order to diminish the semantic gap and support more general queries: this approach allows to take into account low level and high level concepts and to enable different query paradigms (search by similarity, search by analogy, etc.).

2.2 Concepts and Models for Context and Metadata

Even if the study of metadata and context has been carried out for many decades, nowadays there is still some confusion on defining and modelling context and metadata.

For example, when dealing with Information Services, the dynamic behaviour of some metadata descriptors is sometimes interpreted as context, as in [4]. In other works usually focused on mobile applications, such as [5], context metadata is defined as “information that describes the context in which a certain content item was created”. And finally, advanced works trying to integrate context and content like [6] decided that “the term *context* refers to whatever is common among a set of elements”.

Context and metadata are clearly associated to knowledge; “meta-data” is information about data, but what is context doesn’t seem to be so clear. We agree with the probably most generic definition found in the literature provided by A. Dey [7]:

"Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves."

Furthermore, while metadata is supported by really mature standardised schemas such as the one defined by MPEG-7, there is a clear need of defining a common schema of contextual information for generic multimedia scenarios. If not, it will be difficult to take the maximum advantage of it use because the advanced works in the area (such as the previously identified) won’t be able to be extended and interoperable with the incoming complex multimedia scenarios of the future.

2.3 Context-Aware Content Adaptation

Context-aware content adaptation has become an important line of research, however it has always lacked of standardised models to represent and manage contextual information. Three main initiatives should be identified: the CC/PP (Composite Capability/Preference Profile) created by the W3C which defines an RDF-based framework for describing device capabilities and user preferences, the UAProf (User Agent Profile) of the Mobile Alliance Forum which provides an open vocabulary for WAP (Wireless Access Protocol) clients to communicate their capabilities to servers, and the Usage Environment Description (UED) tool included in MPEG-21 Digital Item Adaptation (DIA) which consists of a complete set of context descriptors for a multimedia adaptation scenario. It includes user information, network and terminal capabilities as well as natural environment descriptors.

The first two are limited to specific applications, and represent a small subset of contextual information only. Without doubt, the most complete initiative trying to identify and represent the context for generic multimedia applications has been carried out by the MPEG community by means of the 7th part of its MPEG-21 standard. It is called MPEG-21 DIA (Digital Item Adaptation) and includes all kind of descriptors to facilitate context-based content adaptation.

2.4 Digital Rights Management (DRM) Initiatives

A DRM system provides intellectual property protection so that only authorized users can access and use protected digital assets according to the rights expressions, which govern these assets.

Nowadays, there are several commercial initiatives that specify a complete DRM system. Moreover, there are standard initiatives that specify the elements that form a part of a DRM system and the relationships between them. Among the standard initiatives, the most relevant is the MPEG-21 standard that defines a framework for dealing with different aspects of multimedia information management. This standard has normatively specified the different elements and formats needed to support the multimedia delivery chain. In the different parts of this standard, these elements are standardised by defining the syntax and semantics of their characteristics, such as the interfaces to these elements.

In a DRM system rights expressions are defined to be the terms that govern the use of the digital assets. They are presented to the different actors of the digital value chain in the form of licenses expressed according to a Rights Expression Language (REL). RELs specify the syntax and semantics of a language that will be used to express the permissions and restrictions of use of a digital content. Licenses created according to a specific REL are associated to digital assets and can be interpreted and enforced by a DRM system.

Adaptation Authorisation. Adaptation operations should only be performed if they do not violate any condition expressed in the licenses. MPEG-21 DIA specifies description formats for *permissions and conditions for multimedia conversions* that are useful to determine which changes (adaptations) are permitted on content in view and under what kind of conditions.

3 Architecture Description

The proposed search and retrieval architecture for a complex and generic distributed multimedia scenario is depicted in Fig. 1. The core of the architecture is a multimedia content search module based on MPQF, which is elaborated in Section 3.1. Furthermore, the content retrieval service is augmented with a context-based content adaptation service based on MPEG-21 DIA and a DRM service based on MPEG-21 REL, which are discussed in detail in Section 3.2 and Section 3.3 respectively.

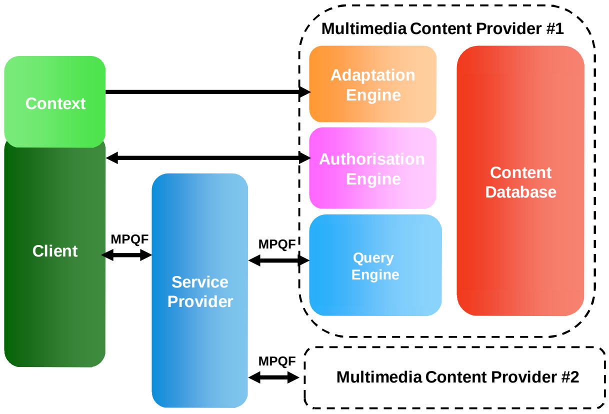


Fig. 1. Proposed search and retrieval architecture for a complex and generic distributed multimedia scenario.

The proposed architecture is flexible and extensible not only because standards have been used, but also because of its modular structure. It allows many different types of multimedia content providers ranging from a simple metadata repository to an advanced provider such as Multimedia Content Provider #1 in Fig. 1. Furthermore, the application of video analysis algorithms for the automatic extraction of low level features could also be integrated in order to address the lack of metadata textual descriptors.

The users can transparently use the DRM engine and the adaptation engine to enrich the content retrieved from the content databases and the query engine. Furthermore these services can also be called by the user application, directly. Finally, the management of contextual information based on profiles (presented in Section 3.2) enriches content adaptation services, as well as content search and retrieval, whenever necessary; and therefore, will also be addressed. .

3.1 Multimedia Content Search and Retrieval Service

MPEG Query Format-Required search functionalities amongst the modules in the proposed architecture vary depending on their roles. On the one hand, service providers (e.g. content aggregators) need collecting metadata descriptions from content providers, and this is usually performed through a harvesting mechanism. Metadata harvesting consists on collecting the metadata descriptions of digital items (usually in XML format) from a set of digital content providers and storing them in a central server. Metadata is lighter than content, and therefore, it is feasible to store the necessary amount of metadata with the service provider, so that real-time access to information about distributed digital content becomes possible without the burden of performing a parallel real-time querying on the underlying target content databases. The search functionalities required for harvesting are very simple, because “harvesters” usually request information on updated records using a datestamp range.

On the other hand, content "retailers", which include service providers and also some content providers (generally medium or large scale providers), should be able to deploy value-added services offering fine-grained access to digital items, and advanced search and retrieval capabilities.

We have chosen the MPEG Query Format as the interface between parties in either of the two different situations described below. Although there exist mechanisms for metadata harvesting (e.g., Open Archives Initiative), MPQF can offer not only a similar functionality, but also a broad range of advanced multimedia search and retrieval capabilities. One of the key features of MPQF is that it is designed for expressing queries combining the expressive style of IR systems with the expressive style of XML DR systems (e.g., XQuery), embracing a broad range of ways of expressing user information needs. Regarding IR-like criteria, MPQF offers a broad range of possibilities that include, but are not limited to query-by-example-description, query-by-keywords, query-by-example-media, query-by-feature-range, query-by-spatial-relationships, query-by-temporal-relationships and query-by-relevance-feedback. Regarding DR-like criteria, MPQF offers its own XML query algebra for expressing conditions over the multimedia related XML metadata (e.g., Dublin Core, MPEG-7 or any other XML-based metadata format) while also offering the possibility to embed XQuery expressions (see Fig. 2).

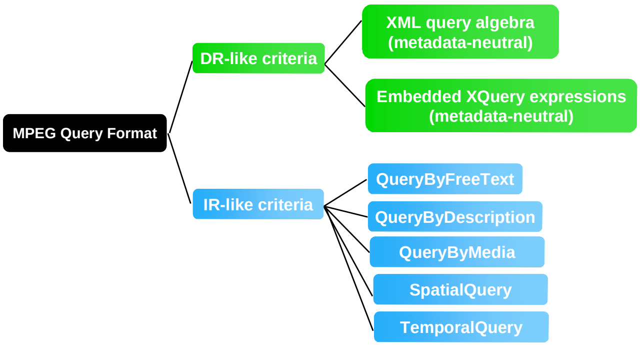
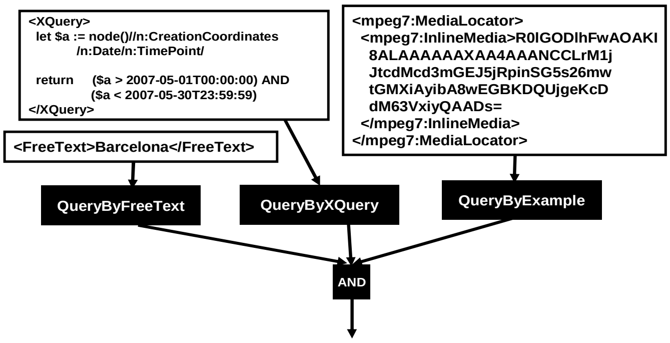


Fig. 2. MPEG Query Format Outline

A valid MPQF document (according to the MPQF XML schema) always includes the Mpeg-Query element as the root element. Below the root element, an MPQF document includes the Input element or the Output element, depending on the fact that if the document is a client query or a server reply (there is only one schema and one root element). The part of the language describing the contents of the Input element is usually named as the Input Query Format (IQF), which it mainly allows specifying the search condition tree (Fig. 3) and also the structure and desired contents of the server output. The part of the language describing the Output element is usually named as the Output Query Format (OQF), and it specifies what the valid outputs are from the server to the client. IQF and OQF are used to facilitate understanding only, but do not have representation in the schema.



```
<XQuery>
  let $a := node()//n:CreationCoordinates
          /n:Date/n:TimePoint/

  return     ($a > 2007-05-01T00:00:00) AND
             ($a < 2007-05-30T23:59:59)
</XQuery>
```

```
<FreeText>Barcelona</FreeText>
```

```
<mpeg7:MediaLocator>
  <mpeg7:InlineMedia>R0lGODlhFwAOAKI
    8ALAAAAAAXAA4AAANCCLrM1j
    JtcdMcd3mGEJ5jRpinSG5s26mw
    tGMXiAyibA8wEGBKDQUjgeKcD
    dM63VxiyQAADs=
  </mpeg7:InlineMedia>
</mpeg7:MediaLocator>
```

Fig. 3. Example Condition Tree

Multimedia Content Search and Retrieval Algorithms. A search and retrieval service able to respond to generic queries expressed with MPQF needs to be integrated with effective search and retrieval algorithms.

As stated above, technology that has proven to guarantee good performances in different use cases is the analysis of textual metadata (keywords, textual descriptions, plots, actors, user comments, etc.): there are several standards (like MPEG-7) describing information related to multimedia contents in a textual and interoperable format. In a real scenario, however, video content is not always accompanied by other corresponding information. There is the need, therefore, to provide innovative ways to allow users to search for content exploiting all the available information. A solution can be identified in the automatic analysis of visual information and MPEG-7 represents a standard way to describe a set of low level features in an interoperable XML format. Nevertheless, one has to get over the abovementioned semantic gap. A way to address the problem and improve the overall performances, still maintaining the standard compatibility is to analyse MPEG-7 descriptors related to low level features jointly with textual metadata, whenever available.

The technical work conducted in this activity can be described as a sequence of different operations:

- Automatic extraction of low level features: for each video, a set of MPEG-7 descriptors is extracted; the obtained low level features are then processed in order to extract temporal and spatial features related to the whole content. The latter are represented by the MPEG-7 descriptors themselves, providing information about visual aspects. Each frame is associated to an element of a codebook made clustering the MPEG-7 descriptors extracted from a training set of videos, using the Generalized Lloyd Algorithm (GLA). A probability distribution related to the whole video is then computed. This distribution describes the visual aspects of the video.
- Starting from the analysis of MPEG-7 descriptors it is also possible to obtain information about the temporal evolution of the videos, revealing, therefore, aspects of storytelling style. After having compressed the total amount of data through the Singular Value Decomposition (SVD), we have chosen to develop new low level features that reflect the complexity of the temporal evolution of

the principal components extracted by the data set: the spectral flatness and the fractal dimension. This is because the amount of change among frames can be inferred directly from the variation of the first few coefficients.

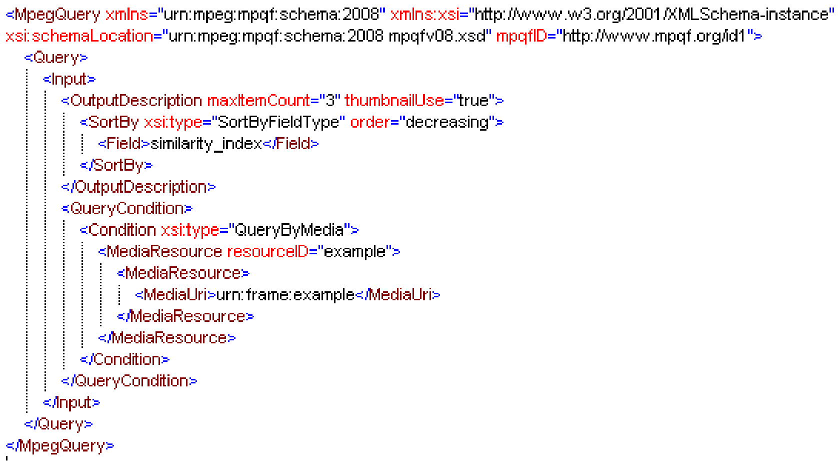
- The visual and temporal features undergo a fusion process to compute the overall similarity among contents;
- Analysis of textual information: Latent Semantic Indexing (LSI) technique has been used, a vector space technique that exploit co-occurrences between terms. Using LSI it is possible to discover similarities between texts even if they share few or no words;
- Construction of searchable indexes: The data extracted with textual and visual analysis are used jointly for creating tables of distances between contents in the repository. Such tables can be used in real time to provide answers to different kind of queries.

For further information about these algorithms one can refer to [8].

Once the searchable indexes are constructed, we can consider that this Content Provider (CP) includes not only the Content database (Fig. 1), but also these processing techniques and the associated database with the similarity measures between contents. When contents lack of metadata, or even for a more accurate result based on a user's Relevance Feedback (RFB), or combined with metadata, QueryByExample (QBE) is an interesting retrieval approach that needs to be addressed.

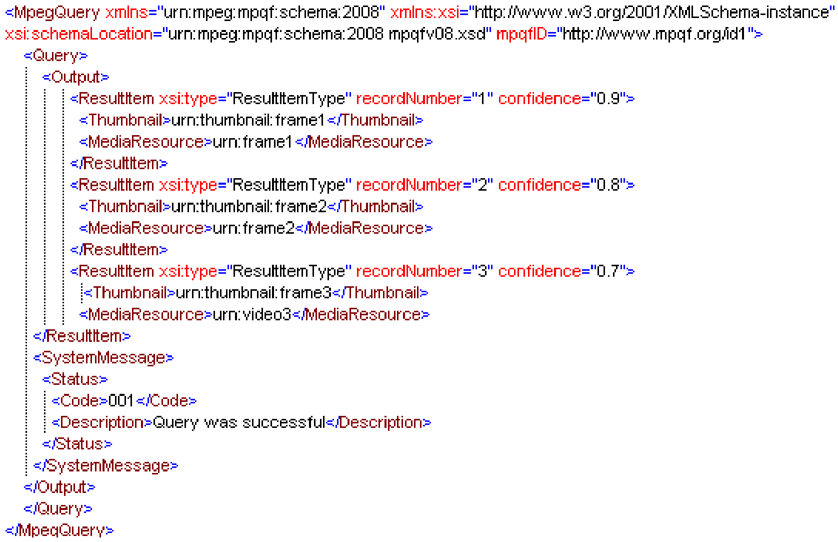
We already stated, at the beginning of this section, that MPQF is capable of expressing different types of queries for IR systems, as for example, query-by-example-media In Fig. 4 and Fig.5 we can see an example of the InputQuery and the OutputQuery, respectively, that could be used between the Service Provider (SP) and the CP previously identified. On the one hand, the request includes the sample of content that is used to express user's interest and the description of the desired output (number of results, etc.). On the other hand, the response includes the list of the "most similar" items that have been retrieved.

Moreover, we could consider these messages as the ones exchanged between the user and the SP in a more specific application scenario.



```
<MpegQuery xmlns="urn:mpeg:mpqf:schema:2008" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:mpeg:mpqf:schema:2008 mpqfv08.xsd" mpqfID="http://www.mpqf.org/id1">
  <Query>
    <Input>
      <OutputDescription maxItemCount="3" thumbnailUse="true">
        <SortBy xsi:type="SortByFieldType" order="decreasing">
          <Field>similarity_index</Field>
        </SortBy>
      </OutputDescription>
      <QueryCondition>
        <Condition xsi:type="QueryByMedia">
          <MediaResource resourceID="example">
            <MediaResource>
              <MediaUri>urn:frame:example</MediaUri>
            </MediaResource>
          </MediaResource>
        </Condition>
      </QueryCondition>
    </Input>
  </Query>
</MpegQuery>
```

Fig. 4. Example of an MPQF QueryByMedia Input Query



```
<MpegQuery xmlns="urn:mpeg:mpqf:schema:2008" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:mpeg:mpqf:schema:2008 mpqfv08.xsd" mpqfID="http://www.mpqf.org/id1">
  <Query>
    <Output>
        <ResultItem xsi:type="ResultItemType" recordNumber="1" confidence="0.9">
          <Thumbnail>urn:thumbnail:frame1</Thumbnail>
          <MediaResource>urn:frame1</MediaResource>
        </ResultItem>
        <ResultItem xsi:type="ResultItemType" recordNumber="2" confidence="0.8">
          <Thumbnail>urn:thumbnail:frame2</Thumbnail>
          <MediaResource>urn:frame2</MediaResource>
        </ResultItem>
        <ResultItem xsi:type="ResultItemType" recordNumber="3" confidence="0.7">
          <Thumbnail>urn:thumbnail:frame3</Thumbnail>
          <MediaResource>urn:video3</MediaResource>
     </ResultItem>
     <SystemMessage>
       <Status>
         <Code>001</Code>
         <Description>Query was successful</Description>
       </Status>
     </SystemMessage>
  </Output>
  </Query>
</MpegQuery>
```

Fig. 5. Example of an MPQF QueryByMedia Output Query

3.2 Context-based Content Adaptation Service

In a complex multimedia scenario, many different adaptation operations may be performed on contents. As mentioned in Section 2, MPEG-21 DIA is a complete standard that specifies the syntax and semantics of tools that assist in the adaptation of multimedia content. It is used to satisfy transmission, storage and consumption constraints as well as Quality of Service management. The proposed context-based adaptation service for search and retrieval application is based on MPEG-21 DIA; a more detailed specification of a possible modular architecture of this service can be found in [9].

Due to the inherent complexity of the standards, their practicality in video search and retrieval domain is slim. In order to address this issue, a number of context profiles are defined. These profiles include User Profile, Network Profile, Terminal Profile, and Natural Environment Profile. They contain all the associated descriptors of MPEG-21 UED, and thus, cover a complete set of contextual information. A detailed description of them can be found in [9]. Furthermore, they could be extended to new types of context that could be identified by new sensors.

The use of profiles eases the introduction of standards while providing more flexibility and scalability to any architecture. In our proposal, their use would definitely enrich the search and retrieval service while guaranteeing interoperability. Not only the user can identify the content he/she wants, but also will receive it in the most optimized way thanks to the Context-based Content Adaptation Service

3.3 Digital Rights Management Service

The DRM Management service will ensure that multimedia copyrighted content is used according to the terms stated by content creators or rights holders. This service will inform to the user the operations that he/she can perform with the videos found by the audiovisual search and retrieval service. It provides functionalities to obtain the licenses governing a digital resource, in this case a video, and provides information content usage information, according to licenses governing the selected video, to the user. Then, the user will select the operation he/she wants to perform, and if necessary will purchase the appropriate license.

This service provides two operations: The first one obtains the licenses associated to the video selected by the user and the second one determines the user's permissions and constraints of content usage. Next are described the two operations in detail:

- *getLicenses*: It receives as parameters a video label and returns an XML file containing the set of MPEG-21 REL licenses governing the image. These licenses will specify the rights that the user can exercise and the conditions that he previously has to fulfil. Moreover, it also returns the rights that the use could exercise if he previously purchases the appropriate license.
- *verifiyRights*: It receives as parameters the user's licenses governing the video and an XML file containing information about the usage that the user has previously done with this concrete video, and determines if the user can exercise the requested operation. This operation implements a license verification algorithm, based on the MPEG-21 REL Authorization Model [x], which verifies if an entity was authorized to perform the requested operation over a video.

Adaptation Authorization: In order to govern content adaptations for protected contents, licenses integrating MPEG-21 REL and MPEG-21 DIA should also be used. The main reason for integrating both standards is that, due to the increasing complexity of adaptations, we require more detailed descriptions about content adaptation in order to govern them. A detailed work on the adaptation authorisation can be found in [10].

4 Social Networks application scenario

Recently, online social networking sites are experimenting a dramatic growth in their use. Users of these sites form social networks and share contents (photos, videos, etc.) and personal contacts. In certain occasions users can wish to protect their personal information and the contents they share for privacy issues. Online social networking sites can be accessed from a broad diversity of devices (PDAs, mobile phones, PCs, laptops, etc.) and in different network conditions (fixed, mobile, local area, wide area, etc), then an efficient adaptation of the contents is required. Due to the huge amount and diversity of contents shared by users, online social networking sites also require efficient search and retrieval solutions. Furthermore, these solutions also can take advantage of automatically extracted features and textual metadata.

In online social networking sites, users will benefit of the proposed solution, since service providers can collect metadata descriptions from content providers and content retailers should be able to deploy value-added services offering fine-grained access to digital items, and advanced search and retrieval capabilities. Moreover, access to these sites can be done from a broad range of consumer electronic devices, since the framework provides efficient and useful adaptations. Finally, users will be able to protect their personal information, contacts as well as the contents they provide to other users.

5 Conclusions and Future Work

This paper has presented a standards-based generic approach for complex multimedia management.

First of all, several weaknesses when dealing with context and metadata for multimedia management have been identified. The novelty of our proposed solutions comes from the fact that these problems are addressed in the most generic way, as the authors consider it is the best approach to exploit the maximum potential of both types of descriptors.

On the one hand, a flexible and extensible way of representing context is used to enrich content adaptation, content search, and Digital Rights Management. On the other hand, the use of a flexible language for multimedia search and retrieval based on MPEG Query Format is the key point trying to narrow the semantic gap, as it gives all the required functionalities. Furthermore, the lack of metadata textual descriptors has been addressed by integrating video analysis algorithms for the automatic extraction of low-level features with this flexible language for multimedia search and retrieval.

The use of standards, such as MPEG-21 DIA, MPEG-21 REL, MPEG-7 and MPQF, is also mandatory to guarantee interoperability with similar systems.

We will continue working on the instantiation of the approach presented in this paper in an ongoing project, the XAC2 project (sequel of XAC, Xarxa IP Audiovisual de Catalunya, Audiovisual IP Network of Catalonia); XAC2 is a network for digital assets interchange among TV channels and content producers. It is worth mention that it is expected that from this work it will emerge the first known implementation of an MPEG Query Format processor. Currently, parts of the ongoing implementation are being contributed to the MPEG standardisation process in the form of Reference Software modules.

Acknowledgments. This work has been jointly supported by the European Commission IST FP6 programme (VISNET II Network of Excellence, IST-2005.2.41.5) and the Spanish government (DRM-MM project, TSI 2005-05277).

References

1. Wang, J.-G. Kim, S.-F. Chang, and H.-M. Kim, "Utility-Based Video Adaptation for Universal Multimedia Access (UMA) and Content-Based Utility Function Prediction for Real-Time Video Transcoding," IEEE Trans. Multimedia, vol. 9, no. 2, pp. 213-220, February 2007.
2. ISO/IEC/SC29/WG11/N9341. *"ISO/IEC 15938-12 FCD MPEG Query Format"*, October 2007.
3. L. Xu, L. and Li, Y. 2003. Video classification using spatial-temporal features and PCA. In *Proceedings of the 2003 international Conference on Multimedia and Expo - Volume 3 (ICME '03) - Volume 03* (July 06 - 09, 2003). ICME. IEEE Computer Society, Washington, DC, 485-488.
4. A. Sorvari, J. Jalkanen, R. Jokela, A. Black, K. Koli, M. Moberg and T. Keinonen, "Usability issues in utilizing context metadata in content management of mobile devices", in Proc. of the third Nordic conference on Human-computer interaction (NordiCHI'04), Tampere, Finland, 2004.
5. M. S. Aktas, G. C. Fox and M. Pierce, "Managing Dynamic Metadata as Context", in Proc. Of the 2005 Istanbul Internacional Computational Science and Engineering Conference (ICCSE2005), Istanbul,,Turkey, 27-30 June 2005.
6. M. Wallace, G. Akrivas, Ph. Mylonas, Y. Avrithis and S. Kollias, "Using Context and Fuzzy Relations to Interpret Multimedia Content", in Proc. of the Third International Workshop on Content-Based Multimedia Indexing (CBMI), Rennes, France, Sep. 2003.
7. A. K. Dey, "Providing Architectural Support for Building Context-Aware Applications," Ph.D. Thesis, College of Computing, Georgia Institute of Technology, Atlanta, Georgia, 2000.
8. IST-1-038398 - Networked Audiovisual Media Technologies - VISNET II, "Deliverable D2.2.5: First set of developments and evaluation for search systems for distributed and large audiovisual databases". November 2007.
9. M. T. Andrade, H. Kodikara Arachchi, S. Nasir, S. Dogan, H. Uzuner, A. M. Kondoz, J. Delgado, E. Rodríguez, A. Carreras, T. Masterton, and R. Craddock, "Using context to assist the adaptation of protected multimedia content in virtual collaboration applications", in *Proc. 3rd IEEE Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2007),* New York, USA, 12-15 Nov. 2007.
10. A. Carreras and J. Delgado, "A new type of contextual information based on the adaptation authorisation", *accepted to the 9th Int. Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2008)*, Klagenfurt, Austria, 7-9 May 2008.

Contribution to the modelling of multimedia metadata in a distributed architecture

Ana-Maria Manzat, Florence Sedes, Romulus Grigoras

Institute de Recherche en Informatique de Toulouse
Ana-Maria.Manzat@irit.fr
Florence.Sedes@irit.fr
Romulus.Grigoras@enseeiht.fr

Nowadays almost any human activities create electronic documents. These documents are more or less complex. Not only that they are generated, but their content is needed sooner or later and the retrieval of relevant documents has become an important problem in the present. The retrieval of multimedia documents is a problem because of the important size of the content's collection and also because of the distributed aspect of the collection and the heterogeneity of the multimedia content.

A fair amount of research has been conducted in the field of information retrieval. There are approaches that are focused on the management of multimedia documents in a distributed architecture (e.g., CAIM project[1], CANDELA project[2]). In an Information Retrieval system, the most important phase is the indexation process. This process generates the metadata (data about data, data that describes the document, its content and also its context). These metadata are used in the retrieval process of documents that respond to a user query.

Now there are many people who show interest in the metadata field, researchers and industrials as well. They work together for establishing standards for meta-data and for creating new ones and improving the existing ones. In the present there are many meta-data standards defined for audio files, video files, text and image like Dublin Core, EXIF, STMPE, MPEG-7,etc. These standards are used for specifying certain information about the multimedia document. Some of the existing formats contain only a limited number of elements (e.g., Dublin Core) and others that are too exhaustive (e.g., Mpeg-7). Thus, the information systems must handle many standards and they become very complex.

The thesis is accomplished under the ITEA2 project LINDO[3] (Large scale distributed INDexation of multimedia Objects). This project is focused on managing the multimedia indexation process inside a distributed environment. An important issue of LINDO is the integration of different indexation engines in the system and their deployment in real time, while the system is running.

[1] http://caim.uib.no/
[2] http://www.hitech-projects.com/euprojects/candela
[3] http://www.lindo-itea.eu

In this context, the core problem of my thesis concerns the development of a generic modelling of the multimedia metadata in a distributed architecture.

Our main idea is that the format to be proposed should contain the existing metadata standards with limited changes in these standards. The integration of the standards must be done with the elimination of redundancy. In the different existent formats there are elements with the same meaning but different names. Our format will take into consideration the synonymy between the elements to be integrated. Some mapping rules must be proposed in order to resolve the synonymy and the equivalence between the standards.

The format we will propose should have an extensible hierarchical structure (e.g., new elements can be added anywhere in the structure). The levels of the structure are not a priori fixed. Such a structure can integrate easily the notion of granularity of the metadata (i.e., we can have metadata related to the hole document, to its content, to a part of the document, and so on).

A generic metadata format has several advantages. It solves the interoperability problem in the Information Retrieval Systems, where the indexing engines have as output different formats of metadata. If these outputs can be integrated in the generic format the system can use this format in the retrieval process and in the management of the multimedia documents. Another advantage is that once the mapping is done between the standards, it can be used to integrate these standards in the generic format and also the same mapping rules can be employed to obtain certain standard elements from the generic format.

Our generic format will be used in the LINDO project in order to integrate any indexation engine in the system without changing the system itself, the output of the new indexation engine will be transformed in the generic format that the system can handle.

Media Center oriented Linux Operating System

Tudor MIU, Olivia STANESCU, Ana CONSTANTIN, Sorin LACRIŢEANU,
Roxana GRIGORE, Domnina BURCA, Tudor CONSTANTINESCU,
Alexandru RADOVICI

University "Politehnica" of Bucharest
Faculty of Engineering in Foreign Languages

tudor.miu@gmail.com, olivia_uk@hotmail.com,
constantin.ana89@gmail.com, sorin_lacriteanu@yahoo.fr,
roxana_grigore89@yahoo.com, domnina_burca@yahoo.com,
tudormihai_constantinescu@yahoo.com, , msg4alex@gmail.com

Abstract: Nowadays there is a high demand of computer controller multimedia home systems. A great variety of computer software media center systems is available on the market, software which transforms an ordinary computer into a home media system. This means it adds some functionality to the normal computer. Our goal is to develop such a user-friendly intuitive system, dedicated for home media centers. In contrast with other proprietary approaches (Windows Media Center, Apple TV), we are building an entire operating system specialized on this. It is based on the Linux kernel, thus providing high portability and flexibility at a very low cost. The system is designed to work out of the box (*plug it in and use it*), needing zero configurations (no human configuration as much as possible) and no installation (Linux-live system, works from a CD, DVD or USB device).

The user interface is not more complicated than a generic TV user interface. In this aim, the file system is hidden from the user, being replaced with an intuitive media library, the driver configurations is done automatically, network configuration is also handled without user actions (as much as possible).

Key-words: OS, media center, Linux, multimedia, portable, intuitive, free, open-source

1. Introduction

The first Linux system was created within an academic environment. With a goal of obtaining a free alternative for commercial UNIX systems at that time, members of an enthusiast community gathered around Linus Torvalds in order to help improving and extending *the kernel*, kernel that he initially created just as a university project. Subsequently, plenty of software projects based on this kernel have been founded, thus giving birth to Linux distributions. Their importance grew more and more each year, achieving nowadays 2% of the market.

Although Linux distributions are free (and some aren't subject to copyright law), they require advanced operating systems knowledge and skills due to the simple reason of lack of graphical interface tools. The existing ones are quite difficult to be handled by those who don't have a programming background. Just imagine an unskilled user trying to set up the sound modules for the *kernel*. The user doesn't know exactly what the sound card is; nevertheless would he know what a *kernel module* means.

Even advanced user interfaces like KDE and GNOME (which by the way occupy a lot of disk space) cannot be considered intuitive. If for us, it takes several minutes just to find the configuration tools, imagine a user which has just bought a computer and wants to watch some movies. This is exactly like the comparison between a TV and a computer. The user goes to the store, buys a TV, plugs it in and it works. On the other hand, if he buys a computer, he will plug it in, will have to wait … and wait … and most surely reboot it (*what does that mean ?*) and maybe than it will display something. In order to actually watch some movies or listen to some music, downloading application is necessary. The result: the user sticks to the TV. [TW06, Ta04]. This means that serious improvement in computer user interfaces is needed.

When it comes to software for playing multimedia, the sky is the limit. Players like Winamp, Windows Media Player, MPlayer [MT08], Amarok, XMMS, GeeXboX, Domotixi, MMS are just a small list of examples. The problem: they lack an attractive user interface, they are not intuitive to use and usually don't meet all of today's users' most common needs. For sure, an ordinary hardware DVD player will be the best choice for a *computer unskilled* user.

Further on, we will present some related work, meaning other Linux approaches to multimedia [CM08].

GeeXboX is a free embedded Linux distribution which aims at turning the computer into a so called HTPC (Home Theater PC) or Media Center. Being a standalone LiveCD-based distribution, it's a ready to boot operating system than works on any Pentium-class x86 computer or PowerPC Macintosh, implying no software requirement. It supports playback of nearly any kind of audio/video and image files and all known codecs and containers are shipped in, allowing playing them from various physical supports, either being CD, DVD, HDD, LAN or Internet.

Domotixi is a free Media Center application for Windows that gathers under the same interface multimedia, communication functions and access to information. It enables the user to watch TV, listen to CDs, Mp3s, watch movies and access information about it (actors, reviews etc), listen to radio, consult the weather forecast, and communicate with MSN contacts. Its interface is adjusted so it can be used on a TV and controlled with the help of a remote controller.

MMS (My Media System) is an application that manages, displays and plays media content such as videos, music, pictures, and more. MMS runs perfectly on anything from a Set-Top-Box connected to your TV-Set, to your specially tailored multimedia PC and HD display. As the name implies, it is a media system *with the user in control*. It lets other applications such as MPlayer, VDR, or Xine take care of what they respectively do best, and integrates them into one system, that is easy to understand and operate.

The goal of this project was to create a simple, intuitive and easy to use Linux based media center. This has been achieved by designing a system which starts from a CD, DVD or USB stick (thus does not require any installation), tries to automatically configure the hardware devices (video, network, etc.), hides the file system from the user by means of a media library and provides an intuitive graphical user interface.

In order to test our goal achievement, we have provided the system to people that do not have any knowledge of computers, so that we could observe their reaction. We will also provide some public beta testing in the future in order to improve usability. Moreover, we will try to detect the most used configuration patterns, so that we can apply them to the new versions of the system.

2. Description

Multimedia oriented system

The aim of this project was to create media center Linux distribution. Unlike other commercial products that exist on the market, which are basically only some software upon a normal general purpose operating system, this project's sole purpose is being a media center. This means that no other software tools are installed (browsers, test editors, etc.). The user interface was specifically design for multimedia (no *Start Menu* or *Applications Menu*, no icons files managers, etc.).

Moreover the system's size has been dramatically reduced, so that it can fit on a very small space. All the other available space on de the device (CD, DVD or USB stick) will be used for multimedia material (music, videos, pictures, etc.).

Several notable differences from a general purpose system are worth mentioning. First of all, the system tries it's best to auto configure the device drivers. This means it will try applying several *known good configurations*. The file system is hidden from the user. Usually file systems can be tricky and have no sense on a multimedia system. Organizing multimedia content into a media library is a much better approach. While searching songs or videos, users prefer searching through a library than through folders and files. For this, several *intelligent* multimedia scanners are used.

The user interface is unique. No Linux standard user interface (such as KDE, Gnome, Xfce, etc.) is used. The main idea of the design is *three-clicks-away* - *if some action takes more than three clicks from the user, it must be redesigned*.

Minimal Live Operating System

In order to use Linux, a minimal live[1] system had to be setup. We have used the *Linux From Scratch* [Be08] (further referred as LFS) approach. This means compiling a minimal Linux system, which should be able to run any program without a problem. This is good, but not enough. The system compiled like this does not make any configuration automatically. Several tools for auto configuring had to be developed. Let's discuss them a little.

Distribution

The aim of the project is to create a mobile drive (CD, DVD or USB stick) with a very small Linux system and a lot of space for multimedia material. Once plugged in, this mobile drive should be able to boot and transform the computer into a dedicated media center. It should be able to combine the multimedia material located on the drive with the multimedia material that if finds on the host computer. Thus, one actually obtains a *portable media center*.

The software for this distribution is minimal: the *kernel*, *startup and auto configuration tools*, *graphical user interface* (the media center), *media library* and a *media assistant*.

Kernel

One of the most important parts is the Linux kernel. After compilation, any operating system kernel does not do very much. It needs device drivers (*modules*) in order to be able to communicate with hardware devices. This is somehow tricky in the case of Linux systems, as default kernel modules are mostly generic. Companies usually don't write hardware code for Linux (e.g. Wi Fi card drivers) and/or release it under licenses different from GPL (e.g. NVIDIA Graphics Card driver). Our approach is to compile a non GPL kernel, a kernel which contains proprietary drivers.

The drivers that we have tried to include are *NVIDIA graphics card drivers*, *ATI Graphic card drivers*, *Intel graphics card drivers* and *Intel Wi Fi network drivers*. These are still undergoing tests, with some success for now. Even without these drivers, the system is still able to run graphics applications, using generic drivers. Performance is lower though.

Another important part of a media center is the sound subsystem. This usually works with the *kernel* generic drivers, but with lower performance. Sadly, there is not much that we can do, as most of the soundcards found on the systems are *on-board*, which means *software cards*[2]. The main problem with software cards is that the 5.1 and/or 7.1 sound systems are specific for every card, so it is not working with generic drivers.

Network is another key point for our media center. Even if the system is not a

[1] A live Linux system is a distribution which boots from a mobile device (CD, DVD, USB stick etc.) and runs without installing anything on the host computer. Examples of live distributions are: *Knoppix*, *ROSLiMS*, *SuSE Live*, *Mandriva One*, *Ubuntu installation*.

[2] Software sound cards are completely software controlled, and without hardware specification, no drivers can be written.

general purpose one, network and Internet connections are important. People might want to share files or download multimedia content. As simple as it might sound, configuration of network devices is a real problem. For now, we are focusing on wired connections (connection which the kernel automatically recognizes). Wi Fi connections are supported only for *Intel Centrino* systems, which are more or less easily recognized by the kernel.

Startup and auto configuration tools

These are key software tools for our media center distribution. The major problem of Linux systems is configuration. As powerful as command line configuration tools are [GA06, So05], as hard to use they get. Most of the times it is rather difficult for experienced users to configure Linux systems. Imagine how it is for inexperienced ones. We are getting back at our *TV problem*[3] [TW06, Ta04].

We have designed special startup software which will do it's best to configure the system. First of all we start with the hardware devices. We try to detect which hardware devices are on the system and subsequently load the respective kernel modules. If no suitable module is found, the generic modules will be loaded as a fallback solution. The errors will be logged so that they can be presented to the user[4].

Once the device drivers are loaded, the next step is to start the graphical subsystem. This poses some problems, as the configuration of the graphical environment has to be done dynamically. Remember that our goal is to start the system on every configuration, meaning different kind of hardware. Our auto configuration tool will do it's best to test the hardware and configure the environment. Upon success, the startup tool will launch the media center graphical user interface.

As a fallback solution in case of graphical system failure, the media center provides a text mode user interface. This is not as shiny as the graphical one, but still provides access to the multimedia content[5].

With the modules and user interface loaded, the auto configuration tool can begin configuring other parameters which can be done while the user already uses the system. We are referring here to *network configuration*, *multimedia content scanning* and other non key configuration points.

Network configuration is somehow tricky. The simplest way is if the user has access to a DHCP network. In this case, the system will be provided automatically with the configuration needed. The only problem is in setting necessary routing parameters, if more than one connection exists (e.g. wired connection together with a Wi Fi one). This is done fairly easy, with no difficulties. Our tool also tries to test the network connection for connectivity.

[3] The comparison between a user buying a TV and a computer, the TV works out of the box, the computer doesn't.

[4] The user will be informed about problems by means of a *media assistant*.

[5] Video multimedia content might not be available, depending on the graphics driver problem.

If no DHCP service exists, the configuration tool will try to set up some *well-known* configuration. This means setting up *classical* IP, DNS and routing schemes (e.g. the most used IP address for the gateway is 192.168.0.1 and for the stations 192.168.0.x The DNS is usually the same as the gateway). This is a brute force trial and error method, which in most cases works perfectly. If none of these works, the error will be logged and the user will be asked for input.

The multimedia content of the host computer will be also scanned. It is somehow awkward to ask the user to find all of his media files on the system. A special tool will scan all the hard drives of the computer and progressively add the found content to the library. This is done completely transparent for the user, user which will only notice that more and more media files appear in the library.

Special heuristics is used to classify media content. This is needed as not all the files on a computer a *real media files*. Imagine for instance that if *Microsoft Windows* is installed on the host computer, several *wav* files will be found on the hard drives. These files contain sounds used by the *Windows* graphical interface. Obviously, they must be filtered out. Moreover, images that are used for creating the interfaces of several programs are also on the hard drive (e.g. messengers' emoticons).

Various algorithms are still under testing, so no conclusions can be drawn yet regarding the efficiency of the algorithm.

Last but no least, an important function is the *plug and play device detection*. This tool detects automatically the insertion of portable devices and adds their contents to the media library. This is a main function for the user which simply wants to insert a camera card and watch his photos or a music player and listen to the music from it[6].

Graphical User Interface

Standard Linux graphical interfaces (GUI) are not suitable for the media center. They are general purpose GUIs, which have much more functions than needed. Moreover, they occupy a lot of space and, sadly, are not very intuitive.

We have designed a completely new user interface, as stated before, guided by the *three-clicks-away* principle. The GUI should be as simple as possible, but as useful as possible. The user must be able to reach any function as fast and as intuitive as possible.

The GUI focuses on three main functions: playback of music, videos and display of photos. We will discuss about all of them.

Photos

When using a media center, a user wants to be able to view his photos. This

[6] Due to propietary formats and digital rights protections, compatibility with Apple iPod is not provided.

function of the GUI allows the user to browse through his photos, view slideshows, and organize them. Future work hopefully will also provide image modification tools.

Music

The main component of a media center is the music system. Mainly media centers are used to playback music. Functions like play lists, next songs[7], repeat, and shuffle are standard for any player. In additions to these, we provide intelligent play lists, based on user ratings, play rates and other parameters. We are still testing several algorithms which will be described in the full paper.

Lyrics are another important component. The system will provide a way for the user to be able to retrieve[8], view and edit the lyrics of a song.

Videos

Movies are an important part of each person's life. Everyone wants to have fast access to his or her movies. Linux provides several powerful *command line* programs for video playback. The purpose of our media center is to provide a GUI for these programs.

Media Assistant

Even if we did our best to automatically configure the system, some issues might still require user intervention. This a bit of a problem for users that have no computer knowledge. To overcome this, we have built a *media assistant*[9] function. This will display a cute character which will retrieve the messages provided by the auto configuration tool and will try to ask the user *simple* questions. This is doable, as mostly the questions that the user has to answer are rather simple. For instance, there might be a problem on detecting the video card type (NVIDIA or ATI). A user will usually know what card he has (he has paid a lot of money on it). With this simple information provided by the user, the system will be able to easily configure the graphic card. Another good example is the network configuration. If none of the *well-known* solutions work, a simple question like *Do you have a small paper with some pairs of four numbers separated by dots given to you by you Internet provider? Maybe a manual will help.* will solve the problem of the addresses.

Media Library

The file system is a real challenger for users. It is somehow hard to imagine folders into folders, different types of files that contain the same thing[10]. This is way our system tries to hide the file system from the user. This is done by means of a media library. A media scanner scans in background the whole

[7] The user is able to tell the system which songs to play next.

[8] Several online databases will be used (e.g. http://www.leoslyrics.com/)

[9] A similar approach was used by *Microsoft* in creating *Office Assistant*.

[10] Images for instance can be in several formats like JPEG, PNG, DIB etc. Music can be MP3, OGG, MOD etc. and videos have even more formats like AVI, MP4, DIVX etc.

hard drive and creates a database with multimedia content. In this way, the user will see a nice organized media library. It will not matter in which format a file is, the important information being the metadata of the file and the content.

This approach poses several challenges. First of all, files are located on the mobile media, on the host computer hard drives and on other portable devices. The necessity to move files around is a problem. Transferring files is still a problem that we are looking into. We have tested several solutions for this matter. The main issue was where to place the newly transferred file. For now, we use a special folder on each device where we place the files or a Windows Media Player similar solution[11].

Another problem posed by the media library is unifying the multiple metadata information. For instance, some music files contain some kind of metadata (sometime doubled – ID3 v1 and ID3 v2), others other type. We have tried to make the database as comprehensive as possible. Several results will be presented in the full paper.

Another important feature of the media library is the transfer of files to mobile media players. The user will be able to transfer files and synchronize to a mobile media player with just a click.

3. Technical Details

We will discuss now a few words about the technical details of the media center Linux.

Kernel

The kernel used for this system is 2.6. We have chosen this kernel due to its support for most of the generic multimedia devices. Most of the new hardware devices, like USB sticks, media players, sound cards, network cards and Intel WiFi cards, are directly supported by this kernel.

User Interface

The user interface is currently based on GTK+ 2 [Pe99], the same libraries as widely used GUIs[12] are based on. The design of the interface though is not done using entirely GTK Widgets (buttons, labels etc.), but also by using several custom graphics. The chosen platform for programming is C/C++ [Dr05].

All these libraries are stacked on top of the graphics Xorg server.

[11] Windows Media Player creates a folder structure similar to the media library one (Album \ Artist \ Song File).

[12] Gnome, Xfce

Multimedia Content

We have used several libraries in order to playback the media files. Music is mostly handled by FMOD Sound System, which delivers high performance. The only disadvantage is that its license is not free for commercial use. Still, it provides high performance as other free tools.

Videos are played back using MPlayer. This very flexible command line video player integrates itself very well in any GUI. It is perfect for our media center.

Media Library

The media library uses a database backend. For this, we have chosen two platforms, MySQL and Sqlite. We have tested which one suits better the system and came to the conclusion that both have ups and downs. Eventually, we have chosen the first option.

MySQL

MySQL has the advantage of being a fully featured transactional database system. The complexity, flexibility and speed of MySQL make it perfect for our system. As it works as a client-server application, the system might be extended to work over a network. This is useful if more media center systems where to be interconnected in a distributed system [TS07], thus a good starting point for improvement.

The drawback of this system is that it occupies a considerable space (more than 50 MB).

Sqlite

This is a very simple SQL library, which is linked into a program. It will store the database into a file (the system is similar to *FoxPro*). The great advantage is the size and complexity. It does not need more than a few KB for the library.

The drawback of this solution is that it cannot be used over a network. Moreover transactions and more complex queries are not supported. Still, it is a solution for our system's library.

Administration Tools

Besides the GUI, the system provides a professional administration tool for debugging and service reasons. These tools include software for experienced users, which will be able to fix a system which won't start with standard configuration.

Conclusions

The system that we have created is perfect suited for a home media center. Moreover, due to the fact that it is Linux based, it is very flexible and portable.

We have to underline that unlike other media center systems, which are in fact just some programs that run over a general purpose computer system, our solution is a stand alone system which is intended to be used only for the purpose of multimedia.

Most important, our system provides almost a complete auto configuration tool, meaning that the user does not have to know anything about computers. The system works exactly *like TV*, *out-of-the-box*. Moreover, it runs like a live system, requiring no installation. All that the user needs a CD, DVD or USB stick.

The user interface is designed by obeying the *three-clicks-away* rule, which states that anything should be done in no more than three clicks.

The file system is hidden from the user by means of a media library. This makes the usage of the system even simpler.

Besides the importance for the users, the features of this media center can be further used in normal Linux distribution. Auto configuration tools are features that are not present in almost any Linux system. This makes them hard to use for users with no computer knowledge.

Bibliography

[Be08] Beekmans, G; Linux From Scratch 6.4, http://www.linuxfromscratch.org/lfs/

[CM08] Centre Multimedia, http://fr.wikipedia.org/wiki/Media_center

[Dr05] Dragoi, G.; Programmation en C++ avec Visual C++ 6.0, Printech, Bucharest 2005

[FM08] Firelight Technologies, FMOD Wiki Documentation Page, http://www.fmod.org/wiki/index.php5?title=Main_Page

[GA06] Glass, G.; Ables, K.; Linux for Programmers and Users, Prentice Hall, 2006

[MT08] MPlayer Team, Mplayer – The Movie Player Documentation, http://www.mplayerhq.hu/DOCS/HTML/en/index.html

[Pe99] Pennington H.; GTK+/Gnome Application Development, New Riders Publishing, 1999

[So05] Sobell, M.; A Practical Guide to Linux Commands, Editors, and Shell Programming, Prentice Hall, 2005

[Ta04] Tannenbaum, A.; Sistemew de operare moderne, Editia a II-a (Modern Operating Systems, Second Edition), Byblos, Buchraest, 2004

[TS07] Tannenbaum, A; Steen M.; Distributed Systems – Principles and Paradigms, Pearson, 2007

[TW06] Tannebaum, A.; Woodhull, A, Operating Systems: Design and Implementation, Prentice Hall, 2006

A Multi-level Access Control Scheme For Multimedia Database

Vanessa EL-KHOURY

Lyon University, CNRS
INSA-Lyon, LIRIS,
UMR5205, F-69621, France
vanessa.el-khoury@insa-lyon.fr

1 Introduction

Security of multimedia database systems becomes a critical problem, especially with the proliferation of multimedia data and applications. One of the most challenging issues is to provide a content-based multimedia database access control that efficiently handles different user's access with possible fine-grained restrictions at a specific level of the multimedia data. However, the realization of such a model depends on other related research issues: (a) Efficient multimedia data analysis for supporting semantic visual concept representation; (b) Practical representation of the multimedia database; (c) Effective multimedia database indexing structure for content-based retrieval; (d) Development of a suitable access control.

2 Related Works

Several efforts have been reported in the literature to support a multi-level multimedia access control. Some of these works [CSZ04], [ZH08] aim to extend existing database access control models by providing new access modes beyond the conventional ones (i.e., read, write, execute); when others [EB03], [TH06] propose their own access control model. In both cases, we noticed that the same steps are followed: First, the entire video is segmented in multi-level access unit and stored in a hierarchy structure. Then, an indexing method is applied over the structure. Finally, a video access control model is presented to tell what kind of visual concepts should be detected and which access control rules should be applied on these visual concepts.

However, these approaches face many problems such as: (1) *Space inefficiency;* instead of adapting the user request on the multimedia data, multiple versions are stored in the multimedia database, where each is customized to meet user-based restrictions. (2) *Complexity;* the extraction and the segmentation techniques are done over the entire multimedia data regardless of the constraints in the user profile. The result is a multi-level hierarchical structure, complex for re-assembling the data when requested. Thus, the time request for a video element becomes so expensive at the run-time due to the time processing of the authorization rules over this structure.

3 Proposed approach

In this work, we intend to present a new scheme that supports a multi-level security protection for multimedia data while considering temporal, spatial and contextual constraints based on restriction extracted from the user profile. The main idea of our work is to avoid an entire segmentation of the video and instead, to gradually treat it by means of a novel and efficient indexing method, as constraints are introduced. Then, these indexes are classified in an efficient hierarchy that maintains segments summaries as defined in MPEG-7. Thereby, when a new constraint is applied over a video, a data mining technique is executed to update the existing indexes by locating the segments containing salient objects. Only segments containing sensitive objects in the frames are stored independently in a blurred form. Then, they are indexed hierarchically so they can replace the corresponding originals one when reading the video. Finally, the access control is done by identifying segment summaries that should be applied to read the document.

From this viewpoint, this scheme remedies to the space efficiency, respectively to the request-time efficiency problem, since only one version of the video is stored in the database and the access to the data is straightforward thanks to the summaries.

Bibliography

[EB03] Elisa Bertino, Jianping Fan, Elena Ferrari, Mohand-Said Hacid, Ahmed K. Elmagarmid, Xingquan Zhu: A hierarchical access control model for video database systems. ACM Trans. Inf. Syst. 21(2): 155-191, (2003).

[TH06] Thuraisingham, B., Lavee, G., Bertino, E., Fan, J., and Khan, L.. Access control, confidentiality and privacy for video surveillance databases. In *Proceedins of the 11 th ACM Symposium on Access Control Models and Technologies,* (2006).

[CSZ04] S.-C.Chen, M.-L. Shyu, and N. Zhao, “SMARXO: Towards Secured Multimedia Applications by Adopting RBAC, XML and Object-Relational Database,” In *Proceeding of the 12th Annual ACM International Conference on Multimedia (ACM-MM),* (2004)

[ZH08] Na Zhao, Min Chen, Shu-Ching Chen, Mei-Ling Shyu, "MRBAC: Hierarchical Role Management and Security Access Control for Distributed Multimedia Systems, In *Proceeding of the 11th IEEE Symposium on Object Oriented Real-Time Distributed computing (ISORC),* (2008)