

Compression and network effect on content-based image retrieval on Java enabled mobile devices

Iftikhar Ahmad¹ and Moncef Gabbouj²

¹Nokia Corporation

P.O.Box 88, FIN 33721, Tampere
FINLAND
Tel. +358-503480631
E-mail: Iftikhar.ahmad@nokia.com

²Signal Processing Laboratory
Tampere University of Technology
P.O.Box 553, FIN-33101, Tampere
Finland
Tel: + 358 (0) 3 3115 3967
moncef.gabbouj@tut.fi
<http://muvis.cs.tut.fi>

ABSTRACT

In this paper we present compression and network effect on content-based image retrieval on Java enabled mobile devices. We use client-server framework where client run on Java enabled mobile devices and servers on personal computer. Servers are designed to provide faster query results and compressed image items to the clients on request. Framework is tested on different networks by using different mobile devices.

1. INTRODUCTION

The way people communicate is changing very fast. Few years ago, mobile phones were lucrative items restricted to a very small community of rich businessman and government agents. Moreover, they were used exclusively for voice calls. Today the mobile terminal penetration is growing steadily and continuously. And their use is no longer restricted to voice communication only. A phenomenal change in the use of mobile phones occurred by the introduction of the multimedia phones. In Finland, it is widely accepted among youngsters to use a GSM phone for sending multimedia messages to share their images, audios or videos. The third generation, or 3G [4] services are already in use and create new opportunities for content providers, by providing a way of transmitting text, voice, images, and streamed videos. Moreover, their ability to stay connected to the Internet all the time will provide users with an overwhelming access to a huge amount of information. Users will then face the problem of how to retrieve the information of interest to them in an efficient manner. The goal is to allow searching and navigation in this wealth of data without text-based queries in an efficient way.

Content-based image retrieval (CBIR) addresses the problem of accessing the images that bears some certain content and usually relies on the characterization of low-level features such as color, shape and texture, all of which can be extracted from the images. CBIR area possesses a tremendous potential for exploration and utilization equally for researchers and people in the industry due to its promising results. It has been an active

area of research for the past decade. The content-based retrieval of desired multimedia items requires indexing of the content by the extraction of low-level visual features based on shape, color and texture.

Systems such as “Multimedia Video Indexing and Retrieval System” (MUVIS) [4], [11], VisualSEEk [7], Photobook [6] and Virage [12] have a framework designed for indexing and retrieving images and/or audio-video files. The contemporary MUVIS has been developed as a system for content-based multimedia retrieval on a PC-based environment. It basically provides a unified and global framework that consists robust set of applications for capturing, recording, indexing and retrieval combined with browsing and various other visual and semantic capabilities.

With the encouraging results of content-based information retrieval [1],[2],[3] favorable mobile platform support and limitations imposed by text-based queries researchers and scientists have undertaken challenge of meeting content management of users using mobile devices. In our study a content-based image retrieval system that uses a combination of low-level features for image retrieval as compared to previous retrieval schemes, which only work over a single feature for content-based retrieval [1],[2].

Our research work targets to bring the MUVIS framework beyond the desktop environment into the realm of wireless devices such as mobile phones, Personal Digital Assistants (PDAs), communicators etc., where user can perform query operations in large multimedia databases and the query results can be retrieved within a reasonable time. Therefore, our main goal is to design and develop a content-based image retrieval system that would enable any client supporting Java platform to retrieve images from an image database similar to query image. We have developed a system called Mobile MUVIS (*M-MUVIS*) based on contemporary MUVIS. *M-MUVIS* has client-server architecture. The *M-MUVIS* server comprises of two Java servlets [9] running inside a Tomcat [8] web server, which in effect transforms the standalone MUVIS into a web application. The MUVIS Query Server (MQS) has native libraries for efficient image query related operations. The second servlet so called MUVIS Media Retrieval Server (MMRS) is used for the media retrieval.

In order to take the advantage of flexibility and portability of Java, a *M-MUVIS* client application has been developed by using Java 2 Platform, Micro Edition (J2ME) [10]. Such a system can find its application in sharing or reuse of digital media, content management, networked photo album, shopping and travel.

This paper is organized as follows: Section 2 gives an overview of *MUVIS*, which is the basis of the proposed *M-MUVIS*. Section 3 describes the basic architecture and several functionalities of *M-MUVIS*. Section 4 contains experiments carried out over *M-MUVIS*. Finally in the last section we draw conclusions.

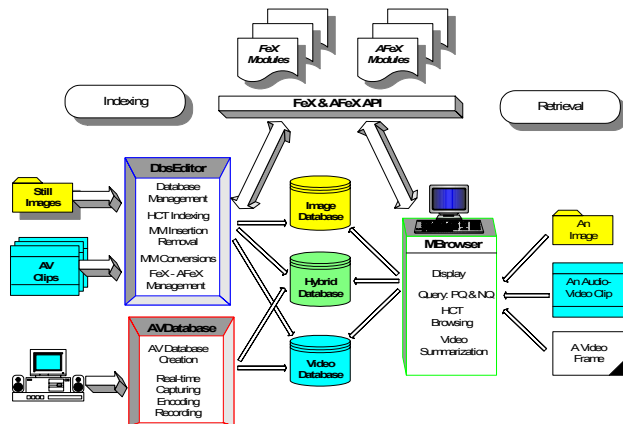


Fig. 1. MUVIS framework

2. MUVIS SYSTEM OVERVIEW

MUVIS is a system, which aims to bring a unified and global approach on indexing, browsing and querying of various digital multimedia types such as audio/video clips and digital images. In order to achieve such a global objective *MUVIS* consists of a set of applications that are illustrated in Fig. 1. *DBsEditor* and *AVDatabase* are used for database creation and modification. *MBrowser* application is used for query operation and query result display, which also has the functionalities of an advanced multimedia player.

Table 1: MUVIS supported image formats

MUVIS Image Types						
Convertible Formats						
JPEG	JPEG 2K	BMP	TIFF	PNG		
Non-convertible Formats						
PCX	GIF	PCT	TGA	PCX	EPS	WMF PGM

The underlying design of *M-MUVIS* uses the querying capability of *MBrowser* and the entire *DBsEditor* application for offline indexing and feature extraction operations performed over multimedia databases. *MUVIS* supports a wide range of multimedia formats; list of supported image formats is given in Table 1.

3. M-MUVIS SYSTEM ARCHITECTURE

As stated above *M-MUVIS* is a client-server application, where servers are running on a PC and client application is running on mobile devices (Nokia's 6630, 9500 [14] etc). *M-MUVIS* client sends the query request to *MQS* and *MQS* then respond by sending the query results to the client. For the retrieval of the Query Resultant Image (QRI), *M-MUVIS* client sends the request to the *MMRS*. The proposed M-MUVIS framework is shown in Fig. 2.

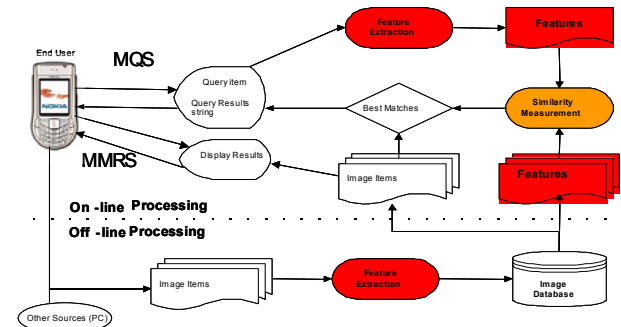


Fig. 2. M-MUVIS Framework

3.1 M-MUVIS Client

M-MUVIS client consists of three parts: User Interface (UI), *Engine* and *Utility*. The user can initiate several operations such as query, change of *M-MUVIS* client settings, etc. and engine use the *UI* to display the query resultant image or images.



Fig. 3. M-MUVIS Client UI

UI display capability can support different screens and image sizes. During the startup phase of *M-MUVIS* client, *UI* is adapted according to device screen size. The same *M-MUVIS* client application can thus be used on different devices with different screen sizes. Main view of *M-MUVIS* client and a query resultant image is shown in Fig. 3.

The *Engine* part is responsible for the activities behind the scene. It determines the thumbnail size according to the screen size of the device. The user may then change such default setting of the thumbnail size from the settings dialog. When the user initiates a query, it performs the query operation by contacting the *MQS* and retrieves the query results as an encoded string from the *MQS*. After retrieving the query results from *MQS*, it sends a request to *MMRS*, to retrieve the QRI. The *Engine*

is also responsible to maintain the list of thumbnails. It can later be used to retrieve the image or to initiate another query using the menu.

Utility the third part consists of commonly used classes for string handling used in engine and UI.

The query and media retrieval wait dialogs are shown in Fig. 4.

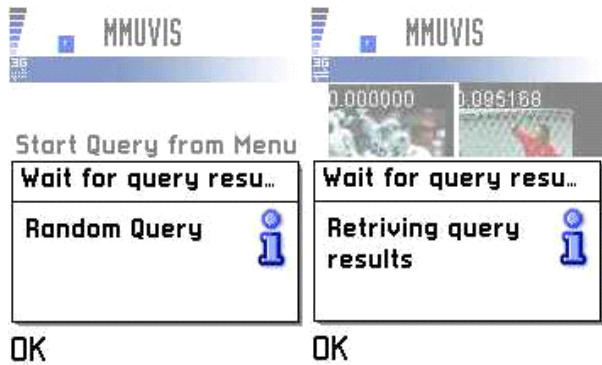


Fig. 4. M-MUVIS query and query resultant image wait dialogs

3.2 M-MUVIS Query Server

MQS is responsible for performing the query operation on the server side. It uses native libraries (C/C++ code) to perform a query operation. When *MQS* receives the query request from an *M-MUVIS* client, it first parses the query string and then passes the query request to the native side. Upon completion of the query operation, the native side sends the query results to the *MQS*. Then *MQS* sends the query results as encoded string to the *M-MUVIS* client. *MQS* is shown in Fig. 2.

3.3 M-MUVIS Media retrieval server

MMRS creates the *QRI* where the first twelve similar images are drawn as thumbnails.



Fig. 5. Query resultant image for 9500

Similarity score of each image is drawn on top of it as shown in Fig. 5. Later *MMRS* compresses the *QRI* and send it to the *M-MUVIS* client. *MMRS* can also provide

images to *M-MUVIS* client on its request.

3.4 Protocol between client and servers

In order to specify the internal settings of a particular query operation, the *M-MUVIS* client, *MQS* and *MMRS* use stream messages in an encoded string format.

The encoded string between client and server uses HTTP [13] protocol underneath. HTTP is a stateless protocol, so a session is created in *MQS* when the query request is received from an *M-MUVIS* client as shown in Fig. 2. Session tracking allows the *M-MUVIS* client to retrieve the query result.

4. EXPERIMENTAL RESULTS

Image database that we used in our experiments is consisted of 1867 images in Portable Network Graphics (PNG) format. The specifications used in our image database are tabulated in Table 2. During our experiments the *MQS* and *MMRS* were running on a PC equipped with Intel Pentium 4 processor 2.13GHz, 1.50GB RAM (Random Access Memory) and running Microsoft Windows XP operating system.

Table 2: Database features

Database Type	Features
Image	<ul style="list-style-type: none"> • HSV and YUV color histogram • Gray Level Co-occurrence Matrix for texture

Client Query Time (CQT) is the waiting period for the query results measured on an *M-MUVIS* client.

Table 3: Average and variance of CQT of 120 values

Connection Type	Average CQT	Variance
6620 (GPRS)	5343 ms	17332
6630 (3G)	3808 ms	85982
9500 (wireless LAN)	1684 ms	41520
9500 (Edge)	4933 ms	19605

An experimental *CQT* in millisecond is measured on Nokia 9500, 6620 and 6630 is shown in Table 3. We measure the *CQT* by using different networks: the fastest query time is achieved in Wireless Local Area Network (WLAN) whereas EDGE performs the query operation lasting in between 3G and GPRS.

Client Media Retrieval Time (CMRT) is the time on the *M-MUVIS* client for the *QRI* retrieval. An experimental *CMRT* is measured on Nokia 9500, 6620 and 6630 as plotted in Fig. 6 where *CMRT* in milliseconds is shown along y-axis and resultant image quality factor of JPG is shown along x-axis. As shown in the Table 3 and, Fig. 6 WLAN has minimum network latencies so the fastest query and media retrieval operations can be achieved over WLAN.

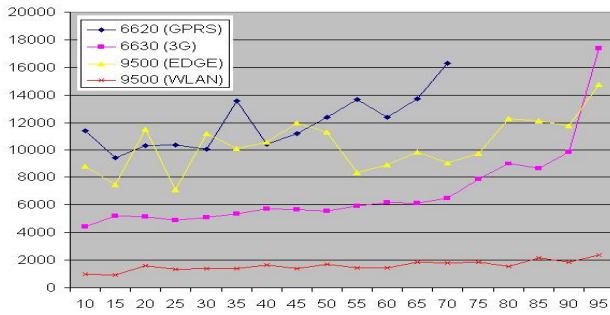


Fig. 6. Client media retrieval time (CMRT)

Fig. 7 shows the Media Retrieval Time (MRT) on *MMRS* is inversely proportional to the *JPG* quality in *QRI*. Lower quality value reduces the *QRI* size that reduces the network traffic and thus in effect reduces the *QRI* retrieval time. Reducing the network traffic in effect makes this service feasible for the practical usages.

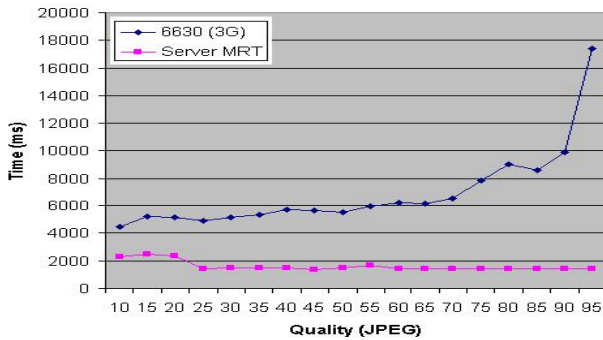


Fig. 7. M-MUVIS client *QRI* and *MMRS* time

Fig 8 shows the *QRI* size versus the quality of *QRI*. During our experiments we observed that quality 20 to 25 is enough to display the *QRI* with out any noticeable degradation of user experience. Since the mobile devices do not support high-resolution display so high compression rate can be used conveniently.

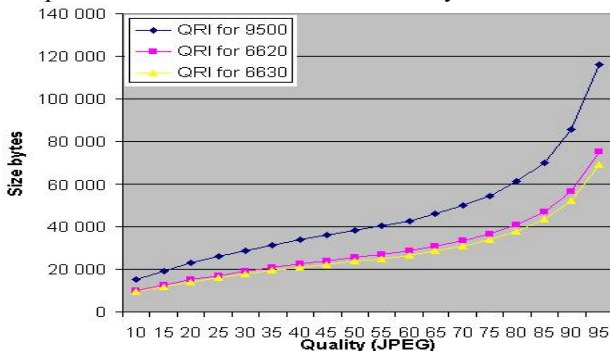


Fig. 8. Query resultant image quality and size

Nokia 6630 mobile phone can use 3G networks during a query operation whereas the other models, 6620 and 9500, can only use GPRS [14] and EDGE [14] technologies respectively.

5. CONCLUSIONS

In this paper, compressions of *QRI* and network effects are studied, in order to achieve the following innovative properties:

- Efficient protocol between client and server is implemented that reduce the *CQT* and client *MRT*.
- As a result of using a joint (submerged) resultant image to retrieve several images in reduced (thumbnail) dimensions, the media retrieval time is significantly reduced.
- *QRI* is compressed that reduce the network traffic and thus further improve the media retrieval time.

We foresee that protocol between client and server can further be optimized for faster retrieval and the same client server architecture can be used for audio and video queries.

REFERENCES

- [1] Ahmad Iftikhar, Faouzi Alaya Cheikh, Bogdan Cramariuc and Moncef Gabbouj, Query by Image Content using Nokia 9210 Communicator, WIAMIS 2001 Workshop on Image Analysis for Multimedia Services 16 – 17 May 2001 Tampere, Finland.
- [2] Ahmad Iftikhar, Faouzi Alaya Cheikh, Bogdan Cramariuc and Moncef Gabbouj Query by image content using Mobile Information Device profile (MIDP), Finsig' 03, Tampere International Center for Signal Processing, Tampere, Finland, May 19, 2003
- [3] I. Ahmad, S. Abdullah, S.Kiranyaz, M.Gabbouj, "Content-Based Image Retrieval on Mobile Devices", Proc. SPIE (Multimedia on Mobile Devices) 5684, Electronic Imaging Symposium 2005, San Jose, California (USA), 16-20 Jan. 2005.
- [4] S. Kiranyaz, K. Caglar, E. Guldogan, O. Guldogan, and M. Gabbouj. "MUVIS: a content-based multimedia indexing and retrieval framework", Proceedings of the Seventh International Symposium on Signal Processing and its Applications, ISSPA 2003, Paris, France, pp. 1-8, July 2003.
- [5] J. Lempiäinen, M. Manninen, Radio Interface System Planning for GSM/GPRS/UMTS, Kluwer Academic Publishers, 2001.
- [6] Pentland, R.W. Picard, S. Sclaroff, "Photobook: tools for content based manipulation of image databases", Proc SPIE (Storage and Retrieval for Image and Video Databases II) 2185:34-37, 1994.
- [7] J.R. Smith and Chang, VisualSEEK: A fully automated content-based image query system, ACM Multimedia, Boston, Nov. 1996.
- [8] Professional Apache Tomcat 5 by Vivek Chopra, Amit Bakore, Jon Eaves, Ben Galbraith, Sing Li, Chanoch Wiggers.
- [9] Java Servlet Programming (O'Reilly) by Jason Hunter with Willaim Crawford.
- [10] The Complete Reference J2ME by James Keogh, McGrawHill OSBORNE Edition.
- [11] "MUVIS", <http://muvis.cs.tut.fi>
- [12] "Virage," <http://www.virage.com>
- [13] HTTP Pocket Reference (Pocket Reference (O'Reilly)) by Clinton Wong.
- [14] <http://www.nokia.com>