A Prototype Content-Based Retrieval System that Uses Virtual Images to Save Space*

Leonard Brown
The University of Oklahoma
School of Computer Science
Norman, OK, 73019
lbrown@cs.ou.edu

Le Gruenwald
The University of Oklahoma
School of Computer Science
Norman, OK, 73019
gruenwal@cs.ou.edu

Abstract

Previous research has demonstrated that space can be saved in a MultiMedia DataBase Management System (MMDBMS) by storing some of the data items virtually, meaning they are stored as sequences of editing operations. The existing approaches for performing Content-Based Retrieval (CBR) in an MMDBMS, however, typically assume that the data items are stored as large, binary objects. The result is that an MMDBMS cannot use the existing approaches without losing the space savings gained by storing the data items virtually. In our demo, we present a prototype CBR system for virtual images that avoids this problem by using the semantic information in the editing operations during retrieval.

1. Motivation

Because images and other types of multimedia data objects are different than traditional alphanumeric data, a MultiMedia DataBase Management System (MMDBMS) has different requirements from those of a traditional database management system. For example, multimedia objects are typically larger than conventional data, so an MMDBMS needs techniques to store them efficiently. For this requirement, [1] proposed that instead of the usual approach of storing images as large, binary objects, an MMDBMS can save space by storing some images virtually, meaning to store them as sequences of editing operations that can be used to recreate them. To illustrate, consider an example application that stores photos obtained from satellites. A user may wish to create a new image by enlarging a portion of another photo, such as in Figure 1 where a new image was created by enlarging a cropped photo of a storm. [1] proposed to save space by storing the new image as a reference to the first photo, called the *base*, along with the select, crop, and enlarge editing operations. By applying these operations to the base, the image of the enlarged storm can be recreated.

Another requirement of an MMDBMS is that it should facilitate *Content-Based Retrieval (CBR)* of its data. A common technique used to perform CBR is to extract and store a set of features from each object as it is inserted into the database, and then search the features in response to a user's query. For example, many CBR systems retrieve images by creating color histograms for each image [2, 3], and then use them to process retrieval queries based on color. This and other feature extraction techniques [4] are inefficient when used with virtual images for two reasons. First, the techniques typically access images in their binary format. So, in order to extract its features, a virtual image must first be converted to its binary format, which can be a slow process. Second, it may be redundant to store the features of a virtual image and its base. To illustrate, consider the colors extracted from a virtual image that was created by rotating its base. These colors will be exactly the same as the ones extracted from the base, so it would waste space to store both sets of extracted colors.

We have developed a new technique that makes use of the semantic information in the virtual images to perform CBR and implemented it in a prototype MMDBMS. In our prototype, we can submit two types of queries, "*Retrieve all images that are between PCT_{min} and PCT_{max} percent of color C_Q*" and "*Retrieve the k images that most resemble Q based on feature F*", where PCT_{min} and PCT_{max} represent percentages, C_Q represents a color in the RGB model, k represents a number, Q represents a query image, and F is a subset of {Color, Shape, Texture}.

* This research is partially supported by NASA Grant #NCC5-398

**Proceedings of the 27th VLDB Conference,
Roma, Italy, 2001**

2. Our CBR Prototype

The goal of our proposed CBR algorithm is to identify all images that satisfy a given query irrespective of their storage format. The algorithm consists of two major steps: 1) Identify the binary images that satisfy the given query using conventional feature extraction techniques, and 2) Identify the virtual images that satisfy the query when they are instantiated. To determine if a virtual image satisfies a given query, we first access the extracted, relevant features of its base, and then determine how the associated sequence of editing operations affects those features. The effects are determined using a set of rules such as *"cropping an image does not add any new colors to it"*. We have defined a set of rules for all editing operations that may appear in the syntax of a virtual image for each query type. For range queries, each rule adjusts the minimum and maximum bounds on the number of pixels that may be of color C_Q in the virtual image. For nearest neighbor queries, once the similarity between the query image Q and the base of the virtual image is measured, each rule adjusts the measurement by an amount dependent on the associated editing operation.

We implemented our CBR algorithm in a web-enabled prototype written in the Perl language that is accessible from the URL http://www.cs.ou.edu/~lbrown. The prototype has several different databases, each consisting of images obtained from various sites on the Internet [5, 6, 7]. Each database also has a set of virtual images created by editing some of the original files. Figure 2 displays the interface for users to specify retrieval queries.

In response to a user's query, our prototype searches the desired database and returns the file names of each found image along with its thumbnail as displayed in Figure 2. When the user clicks a thumbnail for an image stored in the conventional binary format, our prototype displays the full size of the image. When the user clicks a thumbnail for an image stored virtually, the prototype converts the image to a binary format, and then displays it on the web as shown in Figure 3. In addition, a user may view the syntax of a virtual image by clicking on its file name.

References:

[1] Speegle, Greg, Xiaojun Wang, and Le Gruenwald, *"A Meta-Structure for Supporting Multimedia Editing in Object-Oriented Databases"*, 16th British National Conference on Databases, July 1998, Lecture Notes in Computer Science, Vol. 1405, Springer, pp. 89-102.

[2] Ortega, Michael et al., *"Supporting Ranked Boolean Similarity Queries in MARS"*, IEEE Transactions on Knowledge and Data Engineering, Vol. 10, No. 6, Nov/Dec 1998, pp. 905-925.

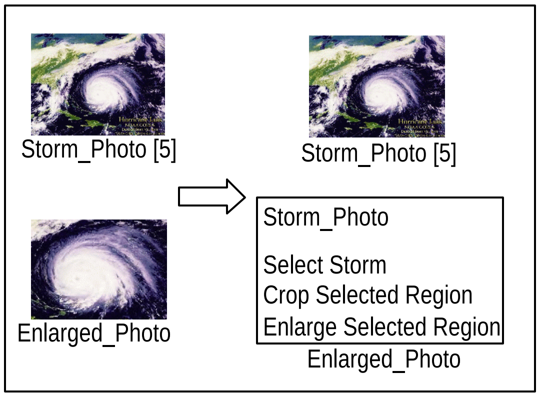
[3] Park, Du-sik et al., *"Image Indexing using Weighted Color Histogram"*, 10th International Conference on Image Analysis and Processing, 1999.

[4] Gonzales, Rafael C. and Richard E. Woods, Digital Image Processing, Addison-Wesley, Reading, MA, 1993.

[5] Images obtained from http://nix.nasa.gov/browser.html.

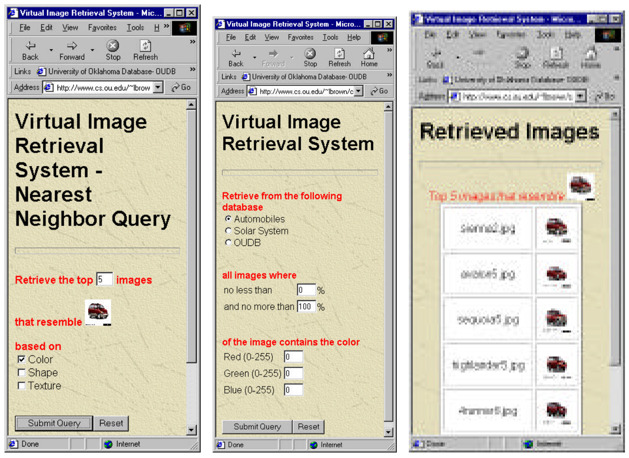
[6] Images obtained from http://www.toyota.com.

[7] Images obtained from http://www.flags.net



Conventional Storage *Use of Virtual Images*

Figure 1 – Conventional and Virtual Image Storage


Virtual Image Retrieval System - Mic...
File Edit View Favorites Tools H
Back Forward Stop Refresh
Links University of Oklahoma Database- OUDB
Address http://www.cs.ou.edu/~lbrow
Go
Submit Query
Reset
Done
Internet
Virtual Image Retrieval System - Micro...
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home
Links University of Oklahoma Database- OUDB
Address http://www.cs.ou.edu/~lbrown/c
Go
Submit Query
Reset
Done
Internet

Virtual Image Retrieval System - Nearest Neighbor Query

Retrieve the top 5 images

that resemble

based on

[✓ Color]
Shape
Texture

Virtual Image Retrieval System

Retrieve from the following database

[✓ Automobiles]
Solar System
OUDB

all images where
no less than 0 %
and no more than 100 %

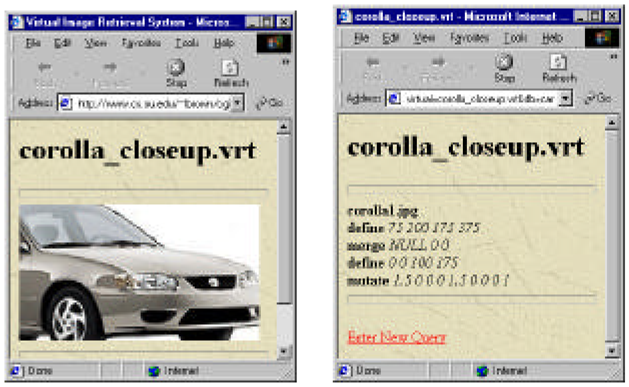
of the image contains the color
Red (0-255) 0
Green (0-255) 0
Blue (0-255) 0

Retrieved Images

Top 5 images that resemble

sienna2.jpg
avalon5.jpg
sequoia5.jpg
highlander5.jpg
4runner5.jpg

Figure 2 – User Interface for Prototype [6]


Virtual Image Retrieval System - Micros...
corolla_closeup.vrt - Microsoft Internet ...
File Edit View Favorites Tools Help
Stop Refresh
Done
Internet

corolla_closeup.vrt

corolla_closeup.vrt

corolla1.jpg
define *75 200 175 375*
merge *NULL 0 0*
define *0 0 100 175*
mutate *1.5 0 0 0 1.5 0 0 0 1*

Enter New Query

Figure 3 – Instantiated Virtual Image and Syntax [6]