Improved CBIR through Combined Indexing Technique

Sihem Mostefai and Sunitha Jeyasekhar

***Abstract*—Multimedia databases get larger and larger in our days. There are various aspects that affect the demand for efficient database techniques to manage the flood of multimedia data, namely the increasing number of objects, the increasing complexity of objects. Whereas traditional indexing structures cope with large numbers of simple objects, complex multimedia objects require more sophisticated indexing techniques. In this paper we discuss the current state-of-the-art of various indexing technique as seen from content-based image retrieval point of view and proposed a novel approach named Hierarchical Hash Grid Indexing structure which has excellent performance and capable of handling huge number of images with less response time.**

***Index Terms*—Indexing Technique, Content Based Image Retrieval, Hierarchical Hash Grid indexing.**

I. Introduction

The immense growth of multimedia elements in the web and digital media led to a greater advancement in research communities. Since there is a vast collection of multimedia information stored in databases, efficient techniques are needed to extract features, to search and to retrieve information in an acceptable time period. Searching and retrieving information efficiently and fast is a key challenge for academic research. When collections of images are stored in a large databases, some of the following issues will be met:1) Semantic Gap 2) Computational Load 3) Time Consumption. The huge amount of images and their sizes and the use of storage devices make it necessary to use effective indexing techniques to facilitate access and thereby reduce the retrieval time[1].

Indexing is a process of collecting, parsing and storing data in order to facilitate fast and accurate information retrieval [7]. Indexing plays a vital role in retrieving relevant images from a large database and is used to accelerate the query performance in the search process.

Efficiency and scalability require a good index structure for retrieving results. Retrieval and Indexing in multimedia databases have been active topic in Information Retrieval. There are various indexing techniques used in CBIR(Content-Based image Retrieval) systems. Among others, there are : R-tree, R*-tree, KDB tree, B-tree, Bx-tree, TB-tree Quad-tree, packed tree, CR-tree, etc. In that R-tree is the very first access methods created by Guttman in 1994.The most critical task of advanced indexing methods is to handle high-level image retrieval in efficient way.

This paper is organized as follows: Section 2 shed some light about the structure of CBIR systems; Section 3 presents a state-of the art on approaches of Indexing techniques. In Section 4, we present a comparative analysis of various indexing techniques; Section 5 describes a novel approach to indexing that we propose to improve the efficiency and scalability of CBIR systems. We conclude the paper by giving important remarks about HHG structure in section 6.

Manuscript received February 19, 2012; revised March 28, 2012.

Authors are with Information Technology, Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia(smostefai@ksu.edu.sa, sjeyasekhar@ksu.edu.sa)

II. Strucure of CBIR Systems

Content Based Image Retrieval systems are used to search digital images in large databases and retrieve relevant ones based on the actual content of the image. Content can be in the form of low-level features or any other information from the images.

Content-Based Retrieval has been used by different communities for various applications like Medical Diagnosis [10], Intellectual Property [10], Broadcasting Archives [10], and Information Searching on Internet [10], Biomedicine [11], Crime Prevention [11], and in Personal photo albums such as Picasa and Flickr [8].

The retrieval process involves the following steps, as shown in fig. 1:

1) Receiving queries from the user in the form of image or sketch.
2) Extracting features of the query images and storing them in the feature database as feature vector/space.
3) Doing similarity matching with the features that are already stored in the feature database one by one.
4) Indexing the vectors for efficient retrieval.
5) Then sending back the retrieved images to the user.

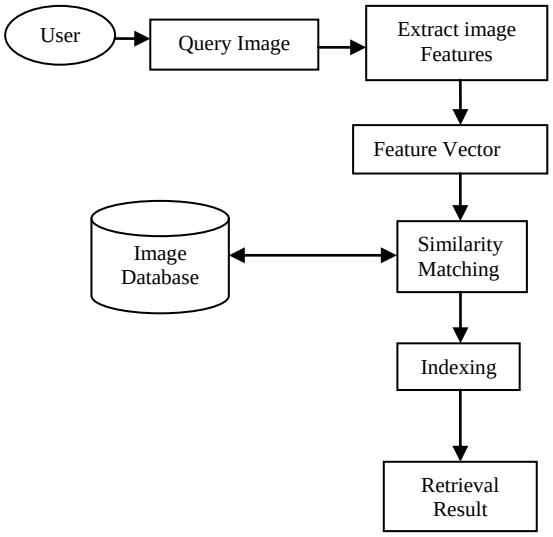

Fig. 1. Diagram for CBIR system

In the following, each step is explained in more detail.

A. Queries

Queries given by the user can be in any form like Query-

by-example, Query-by-keyword, and Query-by-sketch.

- *Query-by-example*: User provides a Query image for search. E.g.: in CANDID (Compression Algorithm for Navigating Digital image Databases), Circus, Compass, FIDS, PicHunter [12].etc.
- *Query-by-sketch*: User draws the query for search. E.g.: in DrawSearch, ImageScape [12] etc.
- *Query-by-Keyword*: User provides text for search. E.g. in AltaVista Photofinder, Diogenes [12] etc.

B. Feature Extraction

The feature extraction is used to measure some components of image contents using image processing. In most of the systems feature extraction is used as a preprocessing step [2]. Typically, there are two types of visual features in CBIR: primitive features which include color, shape and texture, domain specific features [3]. Feature extraction is concerned with capturing visual content of images for search and retrieval.

C. Feature Vector

Extracted features from images are stored as a feature vector or feature space in a feature database. The Feature database is a place where the extracted feature vector will be stored. Each feature vector is sorted in a specific order for the purpose of similarity matching.

D. Similarity Matching

This step consists of matching the query image to the most similar images in the database according to some image to image similarity measure. Images are said to be similar when the value of similarity measure is minimal for the feature vector [2]. Most commonly used similarity functions are: Euclidean function, Hausdorff Distance, Earth Mover's Distance, Weighted Correlation Distance and Signature Quadratic Form Distance [2].

E. Indexing and Retrieval

Indexing is a kind of sorting based on the value given to the image after finding the similarity of each images. It is used to accelerate the query performance in the search process and plays a main role in supporting effective retrieval of sequences of images [2]. Proper indexing makes the search easy and efficient. In this paper we are discussing various indexing techniques.

III. Indexing: State-of-the-Art in CBIR Systems

Nowadays CBIR systems are the focus point of researches because of the tremendous growth in digital repositories. Storing image feature space is the most arduous work in CBIR systems. Once the images are indexed in an efficient way, then it is easy to retrieve the relevant images for the user.

Indexing is used to arrange data in a certain order for efficient retrieval. Here we are going to discuss about three different indexing techniques in Content-Based Image Retrieval. Namely: a) Affinity Hybrid Tree b) Hierarchical Grid Based Indexing c) Tertiary Hash Tree Indexing.

A. Affinity Hybrid Tree (AH-Tree): Indexing Technique for CBIR Systems

This approach combines both Space Based and Distance Based indexing technique to form a hybrid structure which is efficient in term of computational overhead and result accuracy which is close to human perception [5]. Distance Based indexing structures are built based on the distances or similarity between two data objects. Some of the indexing structures are: SS-tree, M-Tree, vp-Tree [5], etc.

Feature Based indexing structure project an image as a feature vector stored in vector space and index the space. Some of the structures are: KDB –Tree, R-Tree, etc. AH-Tree is an indexing technique which maps the low-level features of images with high level concepts.

AH-Tree supports both range queries and k-NN queries [5] it also uses Branch and Bound technique to group the images and Depth First Search for traversal. This approach is a promising approach to bridge the gap between low level and high level features.

B. Indexing Using Hierarchical Grid Based Technique

In this approach the image is converted into the YUV color space and the feature vector derived from the discrete cosine transform coefficients is used to generate a grid code through vector quantization, to group the database images into grids each of which corresponds to a grid code and visually similar patterns. It is called Hierarchical grid Based Indexing (HGBI) [4]. In the classification stage, a reduced set of candidate with the grid code the same as that of the query image can be obtained through the HGBI.

In the matching stage, only the remaining images with similar features are tested for similarity comparison [4]. Through this approach both the search space and dimensionality of feature space are thus reduced [4]. And thus use to retrieve images in an efficient way.

C. Indexing Using Tertiary Hash Tree

This approach works like binary search tree since the destination can be found by repeatedly branching '0' or '1' according to its hash address. To minimize the space usage and computational overhead, two performances like internal conflict [6] and external conflicts [6] are adjusted locally. For the internal conflict, we need to find more bits for hash and location in the conflict node for the conflicting objects.

Conflict nodes are placed at the rightmost nodes in the tree and indicate all the prior conflicts. Then trace down the conflict nodes and check if the expanded hash address by the conflict node. If it does, add them under the node according to their bit pattern. We need to find more bits in the feature to say hash addresses are different [6].

As like internal conflict, external conflict can be handled locally. In this, trace the tree until the bit invoking the conflict is located. As compared to internal conflict, external conflict is very simple. Just create a new node at the position invoking the external conflict and add existing image at the position to '0' branch and the input image to the '1' branch [6]. This approach shows that it provides less resource usage and good search performances than the traditional extendible hash structure. And also compared to R-tree it took less time to search and same accuracy even when the dimensionality is getting higher [6].

IV. Analysis of Various Indexing Techniques

In this section we present a comparative analysis of the various indexing techniques described in section 3. All these techniques were used for the purpose of better search and relevance retrieval improvement. Table 1 lists the indexing techniques in the first column, the algorithms used for these techniques in the second column and the systems that have been used to evaluate these algorithms in the third column.

TABLE I: Comparison of Various Indexing Approaches

Indexing Techniques	Algorithms Used	Implemented in
Affinity Hybrid –Tree Indexing	Range queries, k-NN queries, Branch and Bound technique, Ordered DFS	H-Tree,M-Tree packages as framework
Hierarchical Grid Based Indexing	Discrete cosine Transform, K-NN, Euclidean Distance, SSD	Experimental CBIR system
Tertiary Hash Tree Indexing	L2-norm metric, Range Search	Prototype system on the Linfield i5 CPU, Range search, Center-Contour distance curve.
Novel Approach	HHG Indexing, Euclidean Distance , RBF NN	Personal Photo Album in Process

In Table 2, three various techniques are compared with regards to key performance measure. These include accuracy, Computational load and Execution time.

In comparing the techniques with these measures, we have found that they vary in execution time and computational load.

All the three indexing techniques are same in accuracy. The Affinity Hybrid Indexing technique and Tertiary Hash tree had better performance than Hierarchical Grid based Indexing. In term of execution time, Tertiary Hash Tree performs better than the other two techniques. Computational load and reduction of high dimensional feature values are the main issues for efficient indexing technique. To overcome this problem, we have proposed a new indexing technique with the combination of Hierarchical Hash Grid Structure.

TABLE II: Performances of various Indexing Techniques

Performance	AH-Tree	HGBI	Tertiary Hash-Tree	Novel Approach
Accuracy	Good	Good	Good	Good
Computational Load	Less	-	-	Less
Execution Time	-	-	Low	Low

V. Novel Approach for Efficient Indexing in CBIR Systems

For an efficient search and retrieval of images from a large digital database, we need a powerful indexing technique which can able to take less time for searching and retrieving exact images. For that we have proposed an improved indexing technique with the combination of Hierarchical Hash Grid Structure [9].This HHG indexing technique has an excellent performance and scaling. Only limited storage space is needed and it is capable of handling huge number of objects. By using HHG indexing technique, the systems enable with less response time, make the search easy and good in accuracy.

VI. Conclusion

In this paper we have discussed about the basic structure of CBIR system. Also we have shown the key performance measures of various indexing techniques by comparing with each technique. From that we have concluded that all the three indexing techniques are increasing improving the efficiency of image retrieval. But lack in computational load and execution time is less when high dimensional feature values are in image databases. To solve this issue, we have proposed a novel approach called Hierarchical Hash Grid indexing structure which we are currently working on by implementing it in the context of personal photo image databases.

References

[1] M. L Kerfi, D. Ziou, and A. Bernardi, "What is behind image retrieval from the world wide web," In *Proceedings of the International Conference on Web-Based Communities, Lisbon, Portugal*, March 2004.

[2] Sunitha Jeyasekhar,Sihem Mostefai, "Towards Effective Relevance FeedBack Methods in Content-Based Image Retrieval Systems," In *Proceedings of the 2nd International Conference on Information and Multimedia Technology*.

[3] Gaurav Jaswal, Amit Kaul, "Content-Based Image Retrieval-A literature Review," *National Conference on Computing Communication and Control*(ICCC-09)

[4] Te-Wei Chiang, Tienwei Tsai, and Mann-Jung Hsaio, "A Hierarchical Grid-Based Indexing Method for Content-Based Image Retrieval," *Intelligent InformationHiding and Multimedia signal processing* 2007,IIHMSP 2007

[5] Kasturi Chatterjee and Shu-Ching Chen, "Affinity Hybrid Tree: An Indexing Technique for Content-Based image retrieval in Multimedia databases," *Proceedings of the Eight IEEE international Symposium on Multimedia*(ISM'06)IEEE,2006

[6] Yoon-Sik Tak and Eenjum Hwang, "Tertiary Hash Tree: Indexing Structure for Content Based Image Retrieval," *International Conference on Pattern Recognition*, IEEE, 2010.

[7] Deepti Gupta, Komal Kumar Bhatia, and A. K. Sharma, "A Novel Indexing Technique for Web Documents using Hierarchical Clustering," *International Journal of Computer Science and Network Security*, vol. 9, no. 9, September 2009.

[8] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang, "Image Retrieval: Ideas, Influences and Trends of the New Age," *ACM Computing Surveys*, vol. 40, no. 2, Article 5, April 2008.

[9] Prof .Dr.Ulrich Rude, "Hierarchical Hash Grids for Coarse Collision Detection."

[10] Chia-Hung Wei and Chang-Tsun Li, "Content–Based Multimedia Retrieval - Introduction, Applications, Design Ofcontent-Based Retrieval Systems, Feature Extraction And Representation," http://encyclopedia.jrank.org/articles/pages/6567/Content-BasedzzzMultimedia-Retrieval.html#ixzz13DZO776I

[11] Ryszard S. Choras, "Image Feature Extraction Techniques and their Applications for CBIR and Biometrics Systems," *International Journal of Biology and Biomedical Engineering*, Issue 1, vol. 1, March 2007.

[12] Remco. C. Veltkamp and Mirela Tanase, "Content-Based Image Rettrieval Systems: A Survey," *Technical Report* UU-CS-200-34, October 2000.

Sihem Mostefai, Associate Professor is from King Saud University, Kingdom of Saudi Arabia. She holds a PhD in Computer Science from Mentouri University of Constantine (Algeria). She is actively involved in research in various theoretical and practical areas of Computational Intelligence and Information Technology namely Industrial Engineering and PLM (Product Lifecycle Management), Application Integration and reference models for PLM, Knowledge Engineering and Artificial Intelligence, Image Retrieval, Ontologies and Software Engineering.

She has many research papers to her credit. She is the member of several boards of examiners or PhD and M.sc theses.

Sunitha Jeyasekhar, Researcher in the Department of Information Technology, is from King Saud University, Kingdom of Saudi Arabia. She holds a Master degree in Information Technology from Madras University, India. She is actively involved in various researches like Image Retrieval, Optimization Techniques and Reading Patterns using Eye Tracking.