Proceedings of the 2004 IEEE
Conference on Cybernetics and Intelligent Systems
Singapore, 1-3 December, 2004

A Parallel Architecture for Feature Extraction in Content-Based Image Retrieval System

Kien-Ping Chung*, Jia Bin Li*, Chun Che Fung*, and Kok Wai Wong**

* School of Information Technology, Murdoch University, Western Australia
** School of Computing Engineering, Nanyang Technology University, Singapore
EMAIL: K.Chung@murdoch.edu.au; J.Li@murdoch.edu.au; L.Fung@murdoch.edu.au; askkwong@ntu.edu.sg

***Abstract*—Although it is possible to retrieve images from database using a unique identification defined by a human operator as an index to images, it is more convenient and natural to search images based on their contents. The principle of Content-Based Image Retrieval (CBIR) system is to retrieve images based on the content of the images. One of the important components in CBIR system is to extract the visual features of the images for performing more abstract analysis. However, some of these features are computationally expensive. To solve this issue, a flexible parallel architecture has been proposed to improve the extraction time for the system. This architecture will also provide the software system with the flexibility of adding and removing any visual features from the system. Thus, a system becomes more intelligent and so it is able to adapt changes caused by the replacement of more appropriate visual features for representing the images.**

I. Introduction

CONTENT-BASED image retrieval (CBIR) is systems that retrieve images from databases based on the content of the input query. The content can be in the form of keywords describing the image or the visual features such as colour, texture and shape describing the dominant object of the image. This is a relatively new technology, and it is an active field in the research community. Technology such as digital camera, World-Wide-Web (WWW) and cheap storage are the main driving vehicles for such systems. Internet nowadays becomes an extremely huge database, containing millions of website, where virtually every site contains images. To organise and classify such huge amount of images is problematic and time consuming. Therefore, it is necessary to provide an automated mechanism to assist the organisation and classification of images in a database system.

The other reason for gaining such attention is because the essential technologies for such systems are still relatively immature for commercial purpose. The like of user query refinement, multi-dimension indexing structures, image processing and image feature analysis are all equally essential and yet immature to be used for any commercial purpose. Many research efforts have been focused on each of the above technology and a good introduction can be found in [1, 2]. From these survey studies, it can be seen that most of the research attentions have been focused on the software side of the system. Through out the study, only on several very rare occasions [3-5] one would report on the improvement made to the system through hardware configuration. While software packages provide better flexibility for a CBIR system, efficient hardware structure is able to improve performance of the system significantly. Study of software and hardware aspects of a CBIR system is thereby equally important.

This paper will first provide a brief description on content-based image retrieval systems, and follow by the background of the CBIR framework the authors are currently investigating. The paper will then look at the traditional approach in implementing the parallel architecture in these systems. Lastly, a parallel architecture for the CBIR system is proposed.

II. Content-Based Image Retrieval Systems

A. Background

CBIR is a complex system that comprise of many different components. Figure 1 is a CBIR framework proposed by Su, Li and Zhang [9]. This framework is commonly agreed among the CBIR community and it is also the framework that the authors are based on. This framework consists of four tiers of system interaction, and they are: user interface, query processor, indexing structure and the actual image database. The arrow in the figure shows the interaction and the relationship between each tier of the system. The framework is specifically designed to achieve maximum modularity between each tier. The following will provide a brief overview of each tier of the system.

Traditionally, there are two ways to query the systems. Users can either query the system via keywords or image example(s). In addition, some systems also provide feedback and ranking strategies for refining the search query. All these strategies are designed with one goal in mind. That is to provide a mechanism for assisting the system to have a better estimation on the interpretation of the user input.

Query processing, in any content-based retrieval systems, is a module between the user interface and the indexing structure. It acts as a module to bridge the semantic gap between the user's input and the actual query applied to the database. In shorts, it converts the user input into features to be applied as searching key to the database. For keyword query, language tools such as dictionary and thesaurus may be used to further process and possibly also expand on the keyword(s) for later indexing and retrieval purpose. As for image example(s) query, the common approach is to extract low level features such as the colour and texture information of the input image(s) as the key for retrieval. Most of the current researches have been focused on finding a combination of features that can be used to best capture the semantic interpretation of the images. The semantic interpretation of the images is rather subjective. Hence, another related research trend is the development of frameworks for capturing user's interpretation of the images.

In order to make any CBIR systems truly scalable for large size image collection, the images are required to be indexed in a systematic manner. In a traditional database system, the data is indexed by a search key or combination of keys that uniquely identify an individual record. Often, a simple one dimensional data structure is adequate for indexing the data in such systems. However, images are more complex. Images are better represented using image content. Some systems have used keywords for indexing the images, while others have used the low level visual to do so. However, the more resent systems [3] use the combination of both. Attempts to reflect any of the image content usually results in images being represented by a set of values or attributes, commonly known as the feature vector. When images are represented in this manner, each value in the set becomes a point in an n-dimensional space, implying a multi-dimensional structure is required. So far, the research efforts for indexing structures applied to CBIR systems have been mostly revolved around three issues, and they are:

1. How is the data indexed?
2. How is the data organized?
3. How to reduce the dimensionality of the indexing feature without loosing crucial information?

B. *Current Systems*

In the early 90's, the idea of retrieving images based on the image's primitive characteristics have generated a lot of interest from the research community. Attentions have been specially focused on several areas. These areas include: the indexing structure of the database, techniques apply in analysing the image features and ways of approximating the actual intention of the user query.

Over this period of time, researchers have developed different ways of improving both the accuracy and efficiency of the retrieval result. System such as MARS [4] and MindReader [5] have introduced query feedback approach in fine tuning the query for the better approximation of user's true intention. Su, Li and Zhang [6] have taken a step to incorporate Principle Component Analysis (PCA) along with relevance feedback to filter out the less relevant images, hence, reducing the dimensionality of the indexing structure. In addition, the likes of Lu, Zhu, Zhang and Yang [3] and Cox, Miller, Minka, Papathomas and Yianilos [7] have integrated the keywords along with the low level visual features into the indexing structure for a better representation of the image semantics. Figure show the basic framework for a CBIR system with relevance feedback.

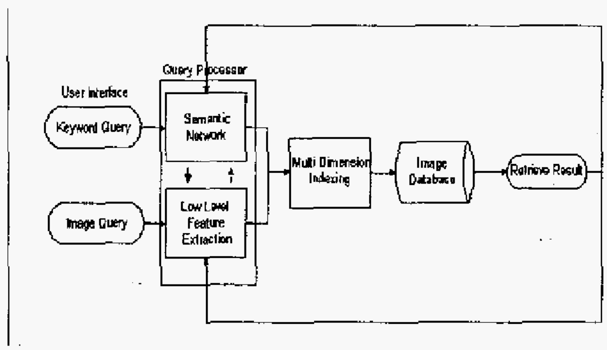

Figure 2 A CBIR framework with relevance feedback

C. *Issues*

To date, most of the research efforts on CBIR can be simplify to two issues: the retrieval accuracy and the feedback time of the results. This should not come as a surprise as this is essentially what most of the users is concerned with. The following sections will provide a brief background on the software framework the authors are currently investigating, follow by discussion of a possible parallel architecture solution for the improvement of the system retrieval time.

III. BACKGROUND OF THE PROPOSED SOFTWARE FRAMEWORK

Most CBIR systems may be grouped into two main categories, namely, *generic* and *domain-specific* systems. Domain-specific retrieval systems only contain images that are closely related to a specific application area. The domain knowledge of the specific application often provides extra information which may assist the analysis of the visual and semantic content of the image with remarkable results. Automatic diagnostic system or decision support for medical applications are areas where researchers have great success in integrating the domain knowledge with the image retrieval techniques. However, the lack of robustness is often a problem exists among the domain specific applications. Very rarely, one can find a generic image processing or analysis algorithm that performs equally well in multiple domain specific applications.

Generic domain retrieval systems contain images created or taken from different sources. The theme and contents of the images may also cover diverse topics. Typical examples of generic domain applications are systems like Yahoo Image Surfer and Lycos image library. Generic domain applications generally do not suffer the problem of lack of robustness in the models applied to the processing and analysis of images. However, the trade off for using the more generic applications is that the image models used to represent the images are often very generic. In these systems, only a few selected common low-level features are usually used for representing the content of the images. Quite often, these features only include colours, texture and sometime shapes. Thus, one of the biggest issues in generic CBIR systems is their inability to automatically capture the semantic content of the images and retrieval performance, in the terms of precision and recall rate, are often not as good as the domain specific systems. For instance, the medical retrieval system medGIFT [8] is an adaptation of the generic image retrieval system GIFT [9]. The adaptation of the system is mainly done in the colour quantization. By knowing medGIFT is primary for handling greyscale images, the system designers modified the visual features used in GIFT by increasing the number of grey levels while reducing the number of colour level used in analysing image. The system designers claimed such change lead to much better result in retrieval accuracy.

It is obvious that generic and domain specific applications in CBIR are at opposite ends. Both types of application possess their own unique characteristics which have presented different challenges to the researchers. The proposed software framework for CBIR system will be to use the generic approach to classify and group images into different categories, and then apply a more application specific approach within each group to achieve a better analysis result for these images.

IV. Issues in Distributed CBIR Systems

Often, in the context of CBIR, the term architecture merely means the software framework of such system. For instance, two of the most well known CBIR literature survey studies [1, 2] used the term architecture as the grouping and the layout of different software components and their relationship within the system. Most of these authors assume such system will only run on a single machine. As limitations of the Moore's Law and the success of the cluster computing systems, the use of distributed systems interconnected throughout networks becomes increasingly common. First, a cluster computing system is implemented using common-off-the-shelf (COTS) technologies. Hence, such system is inexpensive yet reliable to use. Second, such system is scalable and easily maintained. As a result, it is suggested that there is a significant advance to study a CBIR system based on the cluster computing structure.

There is a very important property of most CBIR systems, which it encourages the use of distributed systems. That is, most user attempt to read images only, while the consistency problem does not exist in most CBIR systems and so it is ignored. Such assumption is essential that it helps to simplify the design of a CBIR system, while the consistency problem caused by the replication operation adds complexity to the design. Therefore, the key issues in the design of parallel architecture of a CBIR system are the followings.

1. **Extensibility in feature selections.**
   CBIR is a rapidly changing research area. Therefore, any image search and retrieval system using content-based retrieval method must be extensible with respect to new feature extraction algorithms and indexing methods. This is to say that the architecture should be flexible enough to allow system operator to add, remove and modify feature processor with ease. Therefore, it is essential to have a framework that defines a set of interfaces so that modification of the system becomes much easier.
2. **Collaborative computing.**
   As mentioned previously, the replication technique is well-suited to CBIR systems where images are duplicated for efficient computation. Within such system, a number of processing nodes work collaboratively to generate primitive features for an image. To illustrate, some of the visual features such as colour histogram and standard deviation of the pixel's neighbourhood intensity can be very easily divided into number of smaller tasks and calculate in parallel. That is, the computational model of the system is organised in a hierarchical structure, where the root node spreads out multiple copies of an image to its precedent nodes that performs specific analysis to the image. Furthermore, sub-layers of computation nodes can be attached to the system to share the workload. Hence, the feature extraction process can be improved.
3. **Maximized utilisation of resources.**
   Utilisation of resources is the most critical problem for any distributed systems. Inappropriate resource uses can lead to serious deficiency in performance. However, completely balancing workloads across the system is an extremely difficult assignment, if not impossible. This is largely due to the complex relationship associated with all performance measures, such as, the network latency, the communication overhead, and the CPU utilisation and so forth. It is thereby required to design a simplified cost function to satisfy key performance measures closely related to the CBIR system without sacrificing other factors too much. Such cost function is largely ignored by most researchers as they primarily focus on study of efficient algorithms, while these constraint models are extreme important in the design of a parallel architecture. Consequently, design of proper cost functions is one of the key areas focused in the project.
4. **Scalability.**
   Scalability is required for both software and hardware aspects of a CBIR system. From the software perspective, as mentioned previously, the proposed parallel CBIR system has the capability to add/remove agents to/from the system. So the system may be expanded or shrunk during execution. However, it is not unusual that the number of agents is excess to the number of available computational nodes. In such case, it is crucial to have a mechanism to

map these agents to the existing network, especially, the interconnection among agents should be affected regardless the actual network topology. On the hardware perspective, the system is continuously evolving due to system maintenance or node failure, where a node may be attached to or removed from the existing system. Moreover, the procedure involved in addition or removal of any nodes or agents should be transparent to users.

V. Feature-Domain Agents

The proper definition of agents varies in different contexts of researches. However, the basic concept of agents is more or less similar in all areas, where an agent is an automated, intelligent, software entity that has limited knowledge in a specific subject. As for the image feature agent system, every agent is only responsible for executing one task at any given time. In the context of CBIR, agents will be used for extracting visual image features, indexing and retrieving images and possibly other tasks. As for the integrated-features systems, the feature extraction task is commonly share by multiple computers. This type of systems has a distinct advantage that it is very flexible in term of adding, deleting and modifying features used for describing the images.

As images are replicated to improve the feature extraction process, a feature-domain agent may also be replicated to facilitate the analysis process, a replicated agent has the exact same behaviours and attributes as the original one. That is, either an image is shared by a group of similar feature-domain agents or the image is segmented so an agent is responsible for analysis of that segment. For the first case, the feature extraction process is partitioned into a number of sub-tasks and feature-domain agents with the same functionality perform the sub-tasks with interaction to other agents. Within such collaborative environment, partitioning of a task and scheduling of the sub-tasks are both critical to the performance of the system since improper operations will result in deficiency in performance. Again, study of efficient parallel execution of feature extraction processes is largely ignored in the image processing research community, while the problem is closely related to that of distributed database and distributed operating systems.

On the other hand, each feature-domain agents from the same group work on a small portion of the image only in the second case. While such computational model is not a new concept, most studies have focused on the interaction between the neighbour nodes for analysis of joint edges of each sub-image. However, since the problem of network latency is critical to the performance of the system, so both static and dynamic organisation of feature-domain agents is important. While static partitioning and structuring is relative easy to achieve, dynamic approach is problematic as a right cost function is extremely difficult to find to allow load balance throughout the system.

VI. Proposal of a Distributed Structure for a Parallel CBIR System

A. *Distributed Computational Model*

Figure 2 illustrates the abstract computation model of the proposed parallel CBIR system. First, the analysed image is replicated for analysis of different features, e.g. the wavelet analysis, the colour analysis, and the texture analysis and so on. For each feature analysis, either the process of the feature extraction or the copy of the image is portioned into a series of sub-tasks or sub-images. Based on the procedure described in the previous section, feature-domain agents are created to participate in a sub-task or work with a sub-image. Further, the system has the ability of adding extra agents during performance, as illustrated in Figure 2. An agent (A.4) with capability of wavelet analysis arrives to the system to improve the speed of the process.

While such computation model may be implemented using any programming language, however, Java is considered in use of development for the parallel CBIR system. Features of Java include the following items.

- good portability and interoperability, that is, the program needs virtually no or minimal modification to run across various platforms;
- facilitate reusability;
- support of a certain degree of parallelism; and,
- support of synchronisation mechanism.

In addition, a number of efficient image processing and matrix manipulation APIs have been developed for Java, such as the JavaNumerics working group which studies efficient matrix manipulation APIs for Java. These researches enhance the strengths of Java in use of image processing.

B. *Distributed Shared-Memory Structure*

Two basic forms of parallel computer structures are shared-memory structure and message-passing structure, respectively. While the shared-memory structure implies that a global memory space is used in the system and processors operate and interact with others upon the global memory space. In the message-passing structure, on the other hand, each node has private memory space and these nodes are interconnected with some kinds of networks. However, if a node allows some nodes to directly access to some locations in the private memory, such computer structure is termed distributed shared memory (DSM) structure. The DSM structure provides efficient accesses to share objects by caching these objects in the shared memory space in each node.

Based on different organisation, various DSM systems may be created. For the sake of simplicity, the authors will only focus on one form of organisation for parallel CBIR systems. The proposed structure is illustrated in Figure 3. When there is a request of features extraction for the agent server, the agent server will *publish* the image to a common pool – the notice board and so feature-domain agents may *subscribe* the task according to their abilities and requirements of the task. Once an agent successfully signed up the task, a copy of the image is sent to that agent. That is, the image is shared by a number of agents with different roles. So the first level of the partitioning process is completed. We assume that each feature extraction process is independent to each other and so the consistency problem does not happen in this level.

To improve each feature extraction process, a feature-domain agent is cloned and each clone or a group of clones is allocated to a processor. For both cases, the image is shared by a group of agents and agents perform collaboratively upon either a logical shared memory space formed by a number of nodes or a central shared memory space allocated in the agent node.

VII. CONCLUSION

This paper gives an overview on CBIR and the reasons for requiring parallel system. In addition, the paper has also discussed the weaknesses of some of the existing parallel approaches in feature extraction, and, a new parallel system has been proposed to resolve these weaknesses. The proposed system utilises the agent concept, where an agent has knowledge in a specific area. In addition, the proposed system is based on a hierarchical DSM structure where two-level feature extraction process had been described in the paper. However, such process may be further extended with adding another level such that a sub-task may be subdivided into several smaller tasks thanks to the scalability of the system.

Readers should also note that this paper has only focused on the design of the feature extraction system. One should really utilise the parallel resources by investigating the potential of parallel retrieval. This will be one of future research directions for the authors.

REFERENCE

[1] Y. Rui, T. S. Huang, and S.-F. Chang, "Image Retrieval: Current Techniques, Promising Directions, and Open Issues," *Journal of Visual Communication and Image Representation*, vol. 10, pp. 39-62, 1999.

[2] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jai, "Content-Based Image Retrieval at the End of the Early Years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22(12), pp. 1349-1380, 2000.

[3] Y. Lu, X. Zhu, H. J. Zhang, and Q. Yang, "A Unified Framework for Semantics and Feature Based Relevance Feedback in Image Retrieval Systems," presented at Proceedings of the 8th ACM International Conference on Multimedia, Marina del Rey, California, United State, 2000.

[4] Y. Rui, T. S. Huang, and S. Mehrotra, "Content-Based Image Retrieval with Relevance Feedback in MARS," presented at International Conference on Image Processing, Washington, DC, 1997.

[5] Y. Ishikawa, R. Subramanya, and C. Faloutsos, "MindReader: Querying Database through Multiple Examples," presented at 24th International Conference on Very Large Databases, New York City, NY, 1998.

[6] Z. Su, S. Li, and H. J. Zhang, "Extraction of Feature Subspaces for Content-Based Retrieval using Relevance Feedback," presented at Proceedings of 9th ACM International Multimedia Conference, Ottawa, Canada, 2001.

[7] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos, "The Bayesian Image Retrieval System, PicHunter: Theory, Implementation, and Psychophysical Experiments," *IEEE Transactions on Image Processing*, vol. 9, pp. 20-37, 2000.

[8] H. Muller, A. Rosset, J.-P. Vallee, and A. Geissbuhler, "Comparing Feature Sets for Content-based Image Retrieval in a Medical Case Database," presented at SPIE Medical Imaging, San Diego, CA, USA, 2004.

[9] H. Muller, W. Muller, S. Marchand-Maillet, and T. Pun, "Strategies for Positive and Negative Relevance Feedback in Image Retrieval," presented at Proceedings of IEEE Internet Conference on Pattern Recognition, 2000.

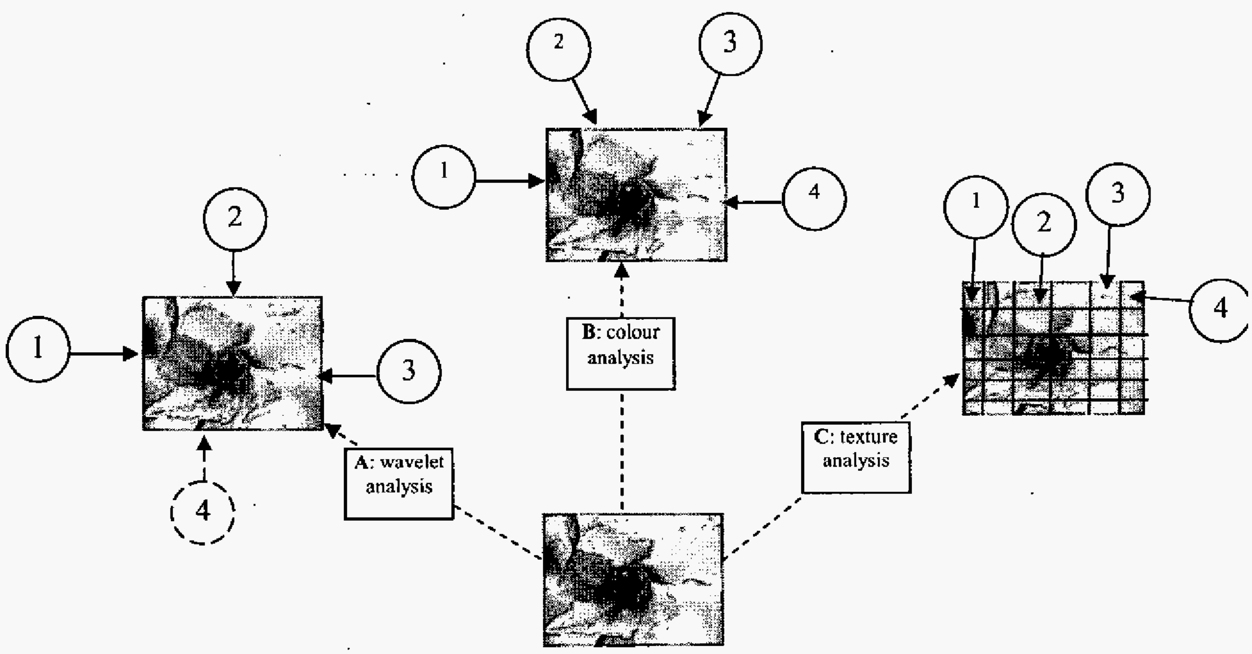

Figure 3: Distributed Computation Model

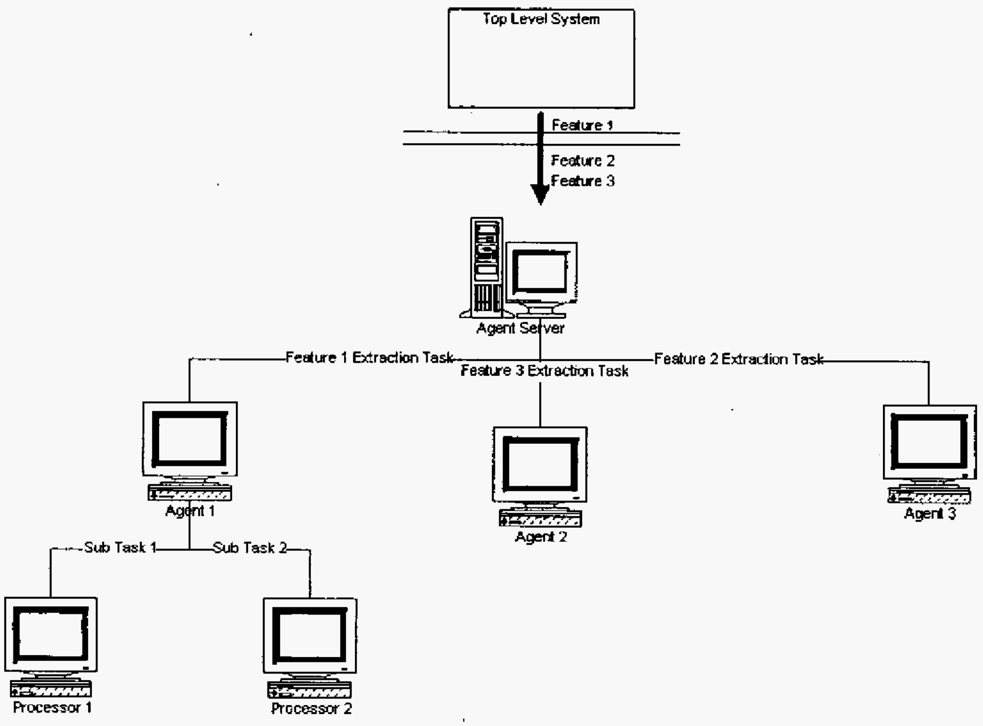

Figure 4: Propose Parallel System for CBIR