Interactive Image Retrieval over the Internet

Jozsef Vass Jia Yao Anupam Joshi* Kannappan Palaniappan Xinhua Zhuang
Multimedia Communications and Visualization Laboratory
Department of Computer Engineering & Computer Science
University of Missouri-Columbia
Columbia, MO 65211
E-mail: {vass, jiayao, joshi, palani, zhuang}@cecs.missouri.edu

Abstract

In the paper, an efficient image database system is developed. The most important features of the proposed system include compressed domain indexing, searching by using scalable features, and progressive image transmission. User interaction is involved at both the search refinement stage and display of the query results. The most important query types include query by color layout and query by wavelet coefficients clustering information. The indexing and searching algorithms are tightly coupled with the underlying image compression algorithm by means the images are stored in the database reducing both the complexity and the storage requirements of the database management system. In this research, we utilize our previously developed very high performance wavelet image coding algorithm termed significance-linked connected component analysis, which not only renders very high compression performance when compared to other top-ranked wavelet image coding algorithms and the JPEG standard, but also inherently supports scalable features and progressive transmission. Computer experiments demonstrate the efficiency of the developed system.

1. Introduction

With the decreasing cost and increasing performance of digital image and video capture devices, computing power, and storage capabilities, more and more visual information will be available on-line in large image and video repositories. However, image and video data is much more voluminous than textual data, and visual information cannot be indexed by traditional methods well suited for the indexing of textual information. This makes *effective* and *efficient* visual information search a very difficult task. Here, "effective" means that the user can find exactly what he or she is looking for, and "efficient" stands for the prompt response of the system. Thus visual information retrieval has of late attracted significant research interest from the academia as well as industry [8].

The paper describes our initial work in designing a flexible image database system which uses the highly efficient wavelet-based image compression algorithm termed significance-linked connected component analysis (SLCCA) [3, 4, 5] for both data storage and indexing, and supports user interactivity for both query by using scalable features, and display by using progressive image transmission.

The organization of the paper is as follows. We describe the three main requirements of any visual database system in Section 2. Section 3 reviews the state-of-the-art, and Section 4 presents the proposed image database system. Performance evaluation is addressed in Section 5 and the last section gives conclusions and further research directions.

2. Image Database Requirements

It can be argued that a visual database system should address three key issues [9, 7] to be effective and efficient, namely, efficient search algorithms, progressive image transmission, and user interactivity during both query and transmission of results.

2.1. Efficient Search Algorithms

In an image database system, searching is the most computationally expensive operation due to the large number of images available in the database. Thus efficient search algorithms are prerequisite of any visual

*Dr. Joshi's work was supported by IBM Faculty Development Award.

database system. In most systems, image features are extracted off-line from the original images, and stored as metadata in the database. In query, the same features are extracted from the query image, and the features of the query image are compared with the features of each target image in the database. To further speed up the search, prefiltering operations can be applied to reduce the number of candidates, and the actual feature matching is only carried out on a subset of target images [14]. In the proposed search engine, *scalable* features are used. Scalable feature means that the higher layer feature is enhancement or refinement of the lower layer feature. Scalable features not only allow interactive query but also accelerate the search. We use both *object-scalable* and *resolution-scalable* features defined in Section 4. Thus the search is done on several *layers* by using scalable features, i.e., at the initial layer only coarse image features are searched which can be done very efficiently. As the search progresses to higher layers, features are refined, or finer features are introduced, and the number of candidate images is gradually decreased. In the proposed algorithm, indexing is tightly coupled with the underlying compression algorithm used to store images in the database, and features are directly extracted from the compressed bitstream thereby substantially reducing the storage requirements and complexity of the database management system.

2.2. Progressive Transmission

The rapidly increasing number of users on the Internet makes bandwidth more and more valuable, thus progressive transmission and display of the search results is a critical issue in the design of a visual database system. In the developed system, the search results are returned in the coarsest resolution. Then, the user can interactively refine the results. Quite frequently, based on the coarse representation of the image, the user can decide whether or not examine this image further, i.e., only images with interest are refined for further evaluation. It is also possible, that the user can examine part of the results while other parts are still downloading and decoding. Most of the search engines lack these capabilities. Usually, results are represented by so-called *thumbnail* images [14, 19, 1], which represent the downsampled version of the original images. If the user selects a thumbnail image, then the original resolution image is downloaded. In the VideoQ system [6], besides thumbnails, hyperlinks to the resulting video files at different resolution and quality are also provided. Thus, based on the available bandwidth the proper resolution and quality video sequence can be downloaded. However, this requires the user's knowledge of the available bandwidth which is usually not the case. Another drawback of this approach is the increased server side storage requirements. In the proposed approach, we use SLCCA as the compression scheme which naturally supports progressive transmission, i.e., a higher resolution image makes full use of the previous resolution which allows maximal utilization of the available bandwidth, user's time, and server storage space.

2.3. User Interactivity

Interactive image query is important due to several reasons such as the large number of images available in the image database and the inherently ambiguous nature of image query, i.e., the same description might match several images with no or little interest to the user. Furthermore, a large percent of users may not know what exactly they are looking for, or how to describe the scene. So it is preferable that after a coarse and fast query, the user can preview and narrow down the search by providing feedback based on images returned by the search engine. Interactive search is also closely related to scalable features. Based on a low layer fast and coarse query, the user can refine the search at higher layers. As the user progresses with the search, more and more search algorithms are provided. Higher layer search techniques usually give more freedom to the user along with more accurate results at the price of higher computational complexity. The main motivation for user interactivity during transmitting and displaying search results is the limited available bandwidth.

3. Review of Previous Work

There have been several image and video search algorithms developed in both academia and industry. Histogram-based techniques are far the most popular in image indexing and searching [21, 2, 15]. Their advantages include compact representation of the image and computational efficiency. In [20], by using several moments of the histogram, Sticker and Orengo presented improved results over traditional histogram comparison with reduced storage and computational complexity. Translation and scale invariant moments are proposed in [13]. Jacobs *et al.* [10] introduced a wavelet-based indexing and searching algorithm, where the similarity between two images are measured by comparing the sign of the most significant wavelet coefficients. In Liang *et al.* [11], a wavelet-based image retrieval system is developed, where the translation variance of wavelet transform is combatted by adopting the number of significant coefficients in each subband as the primary feature. In [12], histogram comparison of highpass subbands is proposed. To increase the efficiency, instead of direct histogram comparison, the histogram of each highpass subband is modeled by generalized Gaussian distribution and the variance and shape parameters of the distribution are used as features. Substantial improvement over the traditional histogram approach is reported.

In all the above approaches, indexing and searching is done based on global features, i.e., the entire fullband image or subbands are used to extract features. In VisualSEEk [19], local features are also used by allowing the user to specify rectangular regions represented by color, size, and location. Rectangular image regions are also automatically extracted from target images by color segmentation algorithm and stored as metadata. Then searching amounts to the determination of the similarity between the regions of the target and query images.

As a summary, we deduce that only some of the proposed algorithms use scalable features [11], progressive display [11], or user interactivity [19, 10], but to our best knowledge none of the existing image database engines integrate scalable features and progressive display with user interactivity by using a unified compression algorithm for both indexing and data storage.

4. Proposed Image Database System

4.1. Image Database Architecture

The block diagram of the implemented system is shown in Fig. 1. The client is implemented as a Java interface. Requests received from the client are passed to the search engine which executes the search based on the metadata directly extracted from the compressed SLCCA bitstream and stored in the database. The system is made extremely flexible by executing individual search algorithms through the common gateway interface (CGI). The state of each client is maintained at the server which is inevitable as a higher layer search is based on the result of a lower one.

4.2. SLCCA Coding Algorithm

Significance-linked connected component analysis (SLCCA) is a very high performance wavelet image coding algorithm. As opposed to zerotree-based algorithms such as embedded zerotree wavelet (EZW) [18] and set partitioning in hierarchical trees (SPIHT)

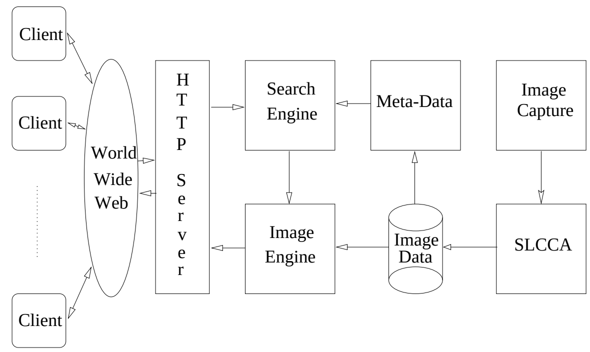

Figure 1. Architecture of the developed image database system.

[16], it organizes significant wavelet data by decomposing them into irregularly shaped connected components or clusters in each subband, a technique called morphological representation of wavelet data (MRWD) [17], and by linking two clusters across subband if one subband contains *parent* and another contains at least one *child*. This exploitation of *cross-scale* dependency among clusters results in the highest peak signal-to-noise ratio (PSNR) in comparison with the other three aforementioned wavelet image coders. SLCCA consistently outperforms JPEG by 3-5 dB in PSNR. The subjective performance comparison of SLCCA and JPEG at compression ratio 48:1 is shown in Fig. 2. As seen in Fig. 2b, SLCCA introduces small visual distortion by smoothing fine image details which appears quite appealing to the human visual system. On the other hand, JPEG introduces quite noticeable blocking artifacts.

4.3. Features and Implemented Query Types

Most of the image and video search algorithms use the red, green, and blue (RGB) color space for searching [14, 2]. The main drawback of RGB color space lies in that the brightness (luminance) and color (chrominance) information are not treated separately, and all three color components need to be represented at the same resolution, which results in large storage requirements and computational overhead. However, it is well known that most of the image information (about 80%) such as shape and texture are conveyed in the luminance component. By using YUV color space, the luminance component is separated from the chrominance components and the later can be downsampled, reducing the storage requirement and accelerating the indexing and searching procedures. Since this is the way

Figure 2. **(a) The original standard "Lena" test image. Decoded images by (b) SLCCA and (c) JPEG at compression ratio 48:1.**

the image is stored and transmitted, the YUV components are available immediately from the compressed domain. Thus the proposed system exclusively uses the YUV color space. However, colors are easily specified by using the GUI color hexagon as shown in Fig. 3.

The simplest query is *query by average color*. The average color is extracted from each *original* image and stored as metadata in the database. Note, that average color is a global, non-scalable feature.

Query by significant coefficients count [11] and *query by significance map* [10] have also been implemented for performance comparison with the respective algorithms. Both the number of significant coefficients and the significance map are global, resolution-scalable features.

In *query by color layout*, users can specify the location, extent, and color of individual objects in the image by using a Java drawing tool. Color layout is a local, object-scalable feature. Object-scalability means that at a higher layer, the user can increase the number of objects in the sketch which increases the computational complexity. However, as the searching progresses, the number of candidates is reduced at every layer, thereby keeping the approximate response time constant.

Query by clustering is based on the observation that clusters of significant coefficients provide an invaluable information about the structure of the image, i.e., clusters in high frequency subbands correspond to texture and edge regions. This information is readily available from the SLCCA bitstream, and is used for indexing and searching. In SLCCA, the relationship between clusters at different scales is represented by the significance-linkage, thus the *number* and *spatial distribution* of significance-links within each subband are used as global, resolution-scalable features. In addition to location, the extent of each cluster provides further insight about the structure of the image, i.e., texture regions from edge regions are likely distinguishable due to the apparent line structure of edges. Thus we use the clustering information, where the sign of coefficients within clusters are compared. Clustering is a local, both resolution-scalable and object-scalable feature. Clusters of the query image are compared with clusters of the target images in decreasing size order, i.e., as the search progresses, clusters with smaller sizes are also involved in the comparison with the number of candidates being reduced. Note, that when all the clusters at the full resolution are considered, the algorithm is identical to [10], where the sign of all the wavelet coefficients are used for matching.

In *query by example*, the weights of the all the above features can be specified by the user.

5. Performance Evaluation

The lack of accepted performance metrics of visual retrieval systems makes performance evaluation and comparison of different search engines fairly complicated. This stems from the subjectivity of visual similarity and lack of standard image database that everybody can use. Currently, the developed system contains about 500 images scanned by the authors. The developed image database system is freely accessible on the WWW with client-side Java interface, and can be evaluated at http://meru.cecs.missouri.edu/vdb. The screen capture of the query interface and the progressive display are shown in Figs. 3 and 4, respectively.

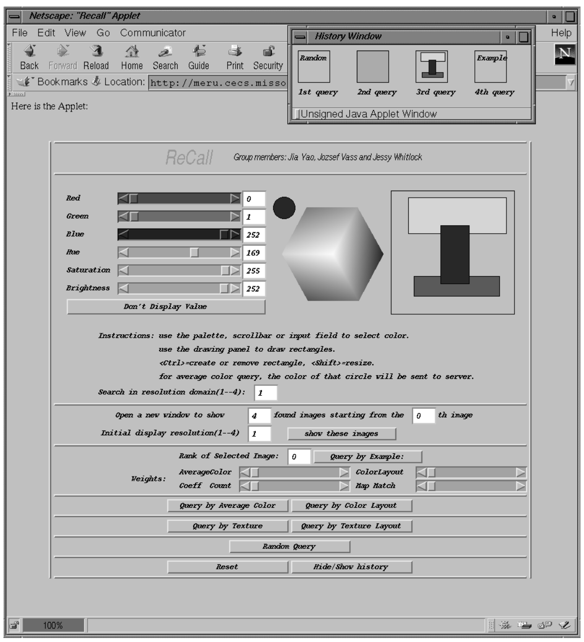

Figure 3. Image query interface.

Currently we are moving the image database into Oracle 8.03, in order to compare the performance with the integrated Virage image engine [1].

6. Conclusions and Further Research Directions

In the paper, our preliminary work in creating a novel image retrieval system is presented. The proposed image database is tightly coupled with the very high performance SLCCA image compression algorithm that naturally supports scalable features used for both indexing and searching, and progressive transmission. The user is involved for both query refinement and transmission of search results in an interactive manner.

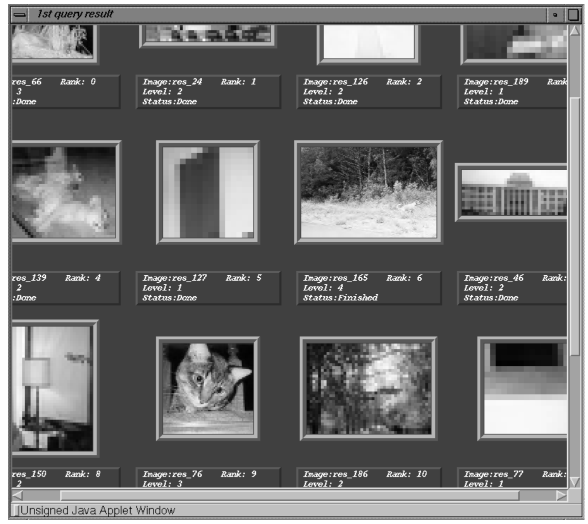

Figure 4. Progressive image display interface.

Further research directions include (1) the derivation of effective objective performance evaluation and comparison algorithms, (2) the development and refinement of different type of queries such as query by structure and texture, (3) the incorporation of efficient video indexing based on our high performance video algorithms [23, 22], and (4) the extension of our database for medical and remote sensing applications, which may require different type of features, indexing, and queries. Finally, (5) by using our automatic spatiotemporal segmentation algorithm [24], the system will be extended to support content-based image and video retrieval as well.

References

[1] J. Bach, C. Fuller, A. Gupata, A. Hampapur, B. Horowitz, R. Humphrey, and R. J. C. Shu. Virage image search engine: An open framework for image management. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, volume 2670, pages 76–87, Feb. 1996.

[2] A. Berman and L. Shapiro. Efficient image retrieval with multiple distance measures. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, volume 3022, pages 12–21, 1997.

[3] B.-B. Chai, J. Vass, and X. Zhuang. Highly efficient codec based on significance-linked connected compo-

nent analysis of wavelet coefficients. In *Proceedings of SPIE AeroSense*, 1997.

[4] B.-B. Chai, J. Vass, and X. Zhuang. Significance-linked connected component analysis for wavelet image coding. *IEEE Transactions on Image Processing, accepted*, 1997.

[5] B.-B. Chai, J. Vass, and X. Zhuang. Statistically adaptive wavelet image coding. In C.-W. Cheng and Y.-Q. Zhang, editors, *Visual Communications and Image Processing*. Marcel Dekker, New York, NY, 1998.

[6] S.-F. Chang, W. Chen, H. Meng, H. Sundaram, and D. Zhong. VideoQ: An automated content-based video search system using visual cues. In *Proceedings of ACM Multimedia*, Seattle, WA, Nov. 1997.

[7] S.-F. Chang, J. Smith, M. Beigi, and A. Benitez. Visual information retrieval from large distributed online repositories. *Communications of ACM*, 40:63–71, Dec. 1997.

[8] M. R. Group. MPEG-7: Context and objectives, July 1997. Doc. ISO/MPEG N1722, MPEG Stockholm Meeting.

[9] M. R. Group. Third draft MPEG-7 requirements, Oct. 1997. Doc. ISO/MPEG N1921, MPEG Fribourg Meeting.

[10] C. Jacobs, A. Finkelstein, and D. Salesin. Fast multiresolution image querying. In *Proceedings of ACM SIGGRAPH*, pages 277–286, Aug. 1995.

[11] K.-C. Liang, X. Wan, and C.-C. Kuo. Indexing, retrieval and browsing of wavelet compressed imagery data. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, volume 3022, pages 506–517, 1997.

[12] M. Mandal, T. Aboulnasr, and S. Panchanathan. Image indexing using moments and wavelets. *IEEE Transactions on Consumer Electronics*, 42(3):557–565, 1996.

[13] M. Mandal, S. Panchanathan, and T. Aboulnasr. Image indexing using translation and scale-invariant moments and wavelets. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, volume 3022, pages 380–388, 1997.

[14] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, and C. Faloutsos. The QBIC project: Querying images by content using color, texture, and shape. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, volume 1908, Feb. 1993.

[15] T. Raymond and D. Tam. An analysis of multilevel color histograms. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, volume 3022, pages 22–32, 1997.

[16] A. Said and W. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(3):243–250, June 1996.

[17] S. Servetto, K. Ramchandran, and M. Orchard. Wavelet based image coding via morphological prediction of significance. In *Proceedings of IEEE International Conference on Image Processing*, pages 530–533, Oct. 1995.

[18] J. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(12):3445–3462, Dec. 1993.

[19] J. Smith and S.-F. Chang. VisualSEEk: A fully automated content-based image query system. In *Proceedings of ACM Multimedia*, Nov. 1996.

[20] M. Stricker and M. Orengo. Similarity of color images. In *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, volume 2420, pages 381–392, Feb. 1995.

[21] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[22] J. Vass, B.-B. Chai, and X. Zhuang. 3DSLCCA – A highly scalable very low bit rate software-only wavelet video codec. In *Proceedings of IEEE Workshop on Multimedia Signal Processing*, Dec. 1998.

[23] J. Vass, B.-B. Chai, and X. Zhuang. Significance-linked wavelet video coder. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 2829–2832, May 1998.

[24] J. Vass, K. Palaniappan, and X. Zhuang. Automatic spatio-temporal video sequence segmentation. In *Proceedings of IEEE International Conference on Image Processing*, Oct. 1998.