Recuperación de Imágenes Médicas por Contenido usando Indexamiento Métrico

Metric Indexing for Content-Based Medical Image Retrieval

Edwin A. Niño, Ing.[1], Juan C. Caicedo, MSc.[2], Fabio A. González, PhD.[2]
[1]Intelligent Systems Research Lab, National University of Colombia - Branch Bogota
[2]Bioingenium Research Lab, National University of Colombia - Branch Bogota
{eaninov, jccaicedoru, fagonzalezo}@unal.edu.co

Recibido para revisión 28 de Noviembre de 2007, aceptado 14 de Febrero de 2008, versión final 28 de Febrero de 2008

Resumen—Este artículo propone un método de indexamiento para hacer recuperación basada por contenido en un repositorio de imágenes médicas de manera eficiente. El método está basado en una estrategia de indexamiento métrico que evita calcular la distancia de la imagen de consulta a todas las imágenes en el repositorio. El método de indexamiento propuesto se prueba en una colección de imágenes de patología comparando su desempeño contra el de una búsqueda secuencial. Los resultados muestran que un método de indexamiento métrico mejora el tiempo de acceso en un factor de 10, sin pérdida significativa de precisión.

Palabras Clave—Sistemas y Organización de Información, Indexamiento métrico, LAESA, Recuperación de imágenes basada en contenido, Recuperación de información, Imágenes médicas.

Abstract—The paper proposes an indexing method for fast content-based retrieval in an image repository. The method is based on a metric-indexing strategy that avoids calculating the distance from the query to all the images in the repository. The proposed indexing method is tested on a pathology image collection comparing its performance against sequential scanning indexing. The results show that the metric indexing method improves the access time by a factor of 10, without a signi?cant sacri?ce on precision.

Keywords—Systems and Information Organization, Content-based Image Retrieval, Information Retrieval, LAESA, Medical Images, Metric Indexing

I. INTRODUCTION

Content-Based image retrieval (CBIR) systems allow to solve queries such as "return the most similar images to image x". A CBIR system is able to solve this kind of query by comparing the content of the image query to the content of the images stored in the system. The design and development of effective and efficient CBIR systems involve two well known problems: the semantic gap and the computational load to manage large file collections.

The semantic gap, as was defined by Smeulders et al. [13], deals with the lack of coincidence between the information that one can extract form the visual data and the interpretation that the same data have for a user in a given situation. It has linguistic and contextual consequences, and mainly depends on the domain knowledge to represent images.

On the other hand, indexing and searching are the tasks that contribute the most to the computation load in a CBIR system. The main problem is that traditional efficient indexing and searching strategies used in database systems are not helpful in this context, since it is not obvious how to de?ne, for instance, a primary key for an image.

This paper tackles the problem of doing efficient indexing and searching in a medical image repository. Specifically, a metric indexing algorithm is proposed, which allows calculating the k-nearest neighbors of the query image without calculating the distance to all the elements in the repository.

A CBIR system for storing medical images is a valuable asset for the medical practice. A large volume of medical images constitutes a knowledge base that collects the experience of many physicians, which have attended a wide diversity of patients. Consequently, effective access to this knowledge base will be a valuable tool to support the decision making process.

The paper is organized as follows: Section II discusses the utility of a CBIR system in the medical context; Section III describes a CBIR model; Section IV introduces the indexing problem and two algorithms to solve it, including a modi?cation of the LAESA algorithm for finding k-nearest neighbors; Section

V compares both algorithms and shows experimental results for time and precision analysis; Conclusions are found in section VI.

II. CONTENT-BASED MEDICAL IMAGE RETRIEVAL

Medical images have been widely used to support clinical decisions in health centers for many years. The medical practice requires evaluating patient's health with reliable evidence to recommend effective treatments, and medical images provide a good amount of information about disease state. Physicians evaluate medical images looking for patterns that reveal diseases and then make the pertinent recommendations. Once images have been used to diagnose, they are archived and rarely used again. This practice is applied in hospitals many times per day, producing large volumes of medical images to be stored and managed [11].

The management of clinical registries and digital images has been delegated to medical information systems. These systems support management processes, historical storage and patient follow up. A large number of medical images available in health centers, hospitals and universities constitute a knowledge base about diseases and diagnosis, useful for academic activities [12], research evaluations, and the decision making process [3]. However, current hospital information systems only provide functionalities to query for data using simple clinical attributes, such as patient's name or some dates, which generally are forgotten and do not allow to identify valuable information for the decision making process.

Medical image databases and Picture Archiving and Communication Systems (PACS) are traditionally based on a search-by-keyword approach. Although medical registries are often composed of alphanumeric data and images, keywords are not enough to find relevant images in large databases, mainly because keywords are not always available and images cannot be fully described by textual annotations. Medical images have specific visual patterns which indicate the presence or absence of a disease, with different shapes, intensities, colors and distributions.

Medical image databases may provide capabilities to search by image content, taking advantage of visual properties. Research in that direction has evolved to Content Based Image Retrieval (CBIR) in different scenarios such as arts, advertising and film industry, among others. Comprehensive surveys on general purpose CBIR systems can be found in [5], [13], [14]. In the medical field a lot of work has also been done, mainly to model the semantics of clinical image analysis [6].

In general, a CBIR system is composed of three main modules: feature extraction, similarity functions and a retrieval algorithm. The feature extraction module is able to quantify some properties of image content, using image processing techniques. Based on the extracted features a similarity function is modeled to evaluate how close two images are. Given an example pattern, a retrieval algorithm is used to identify the set of most similar images using the similarity function. The example pattern is often an image that the user has at hand to query the system, a frequent scenario in medicine, when a physician is attending a new patient. This approach to query the system is known as query by example.

In our previous work, a CBIR system for storing histopathology images was implemented [2]. This system uses both low-level and high-level feature extraction for describing the images, which are retrieved based on tailored similarity measures. The histopathology images were acquired to diagnose a special skin cancer called basal-cell carcinoma. Slides were obtained from biopsy samples which were fixed in paraphin, cut to a 5 mm thickness, deposited onto the glass slides and finally colored with Hematoxilin-Eosin. The whole collection contains about 6.000 images associated with clinical cases. A subset of the collection consisting of 1,502 images were annotated and organized in semantic groups by a pathologist. The groups are representative of the semantic categories that are relevant in the scenario of a content-based image retrieval system, according to the expert. The identified groups are not disjoint because of the nature of the images content.

III. CONTENT-BASED MEDICAL IMAGE RETRIEVAL MODEL

The first goal here is to define how images have to be compared. A medical image is a two, three or four dimensional discrete signal with information about colors, luminance, volume or time in one scene. Therefore, images are difficult to compare because each one is an object, provided with complex properties and different structure. Other technical aspects make images difficult to compare, such as different widths or heights, color representations and formats. A very convenient manner to face these difficulties consists of using a statistical frame: images are modeled as random variables because they are the product of a stochastic process. Then, many statistical measurements can be obtained from images and used as characteristic features. On the other hand, image analysis is required for structuring visual data information. Common features computed for such a task comprise a broad range of possibilities [1], but the very basic ones are color distribution, edges and textures.

Fig. 1 shows the general model used in this system: the image database I contains all histopathology samples and those images are mapped to different feature spaces. Each feature space requires a metric that allows the evaluation of similarity according to the criterion of such space.

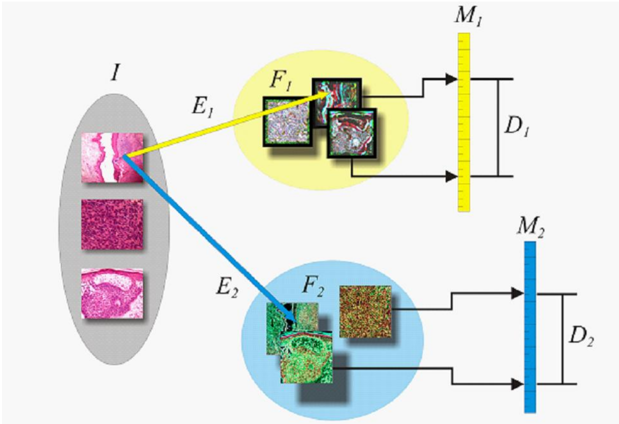

Fig. 1. Image retrieval general model. Elements in the original images space I are mapped to a feature space F_k trough a feature extraction process E_k. In those spaces, images are represented by particular features that can be compared by the use of a distance function D_k.

Let I be the image collection into the database. Let F_k be the space defined for a feature k. The extraction process for a feature k is defined as a function between I and F_k:

$$E_k : I \longrightarrow F_k \tag{1}$$

There exists a feature space onto which images are mapped when a specific feature is extracted so that all images are now represented by their corresponding features in that space. In addition, many feature spaces have to be supported by the image database system and different measurement strategies must be defined for each one. If we assume that those feature spaces are metric spaces, distance functions can be devised for determining the similarity degree of images in each of such metric spaces. A metric space is a tuple (F_k, M_k), where F_k is a set and M_k a metric on F_k as follows.

Let $F_k \times F_k$ be the Cartesian product between features of the same space. Let M be a metric that calculates the similarity between a pair of given features,

$$M_k : F_k \times F_k \longrightarrow \mathbb{R} \tag{2}$$

such that:

1. $M_k(x, y) \geq 0$. Non-negativity
2. $M_k(x, y) = 0$, if and only if x = y. Identity
3. $M_k(x, y) = M_k(y, x)$. Symmetry
4. $M_k(x, z) \leq M(x, y) + M_k(y, z)$. Triangle inequality

Definition 2 permits to introduce an order relationship between images using a feature k and a metric M_k. Previous definitions allow performing image comparisons using one feature and one metric. However, better comparisons may be achieved using many features and a linear combination of different metrics, as follows:

Let $x, y \in I$ be images. Let E_k be the feature extraction function of a feature k and M_k be a metric in the feature space F_k. A similarity function for different features is defined as the linear combination of metrics M_k with importance factors w_k:

$$d(x, y) = \sum_k w_k M_k \left(E_k(x), E_k(y) \right) \tag{3}$$

A. Features

A statistical frame is used to analyze basic image features. Images are modeled as random variables to estimate their probability distribution in different feature spaces: luminance, color, textures, edges and invariant features. The following low-level features have been selected to analyze image contents in a visual perceptual level:

- *Gray scale and color histogram*
- *Local Binary Partition histogram*
- *Tamura texture histogram*
- *Sobel histogram*
- *Invariant feature histogram*

B. Similarity measures

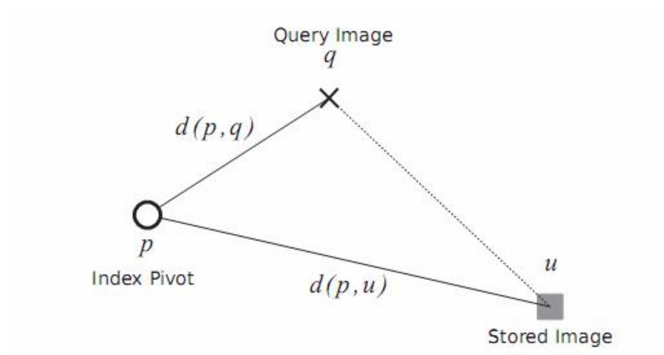

Fig. 2. The triangle inequality

Since low-level features are histograms, they require metrics evaluating differences between probability distributions. The selected features are histograms ranging between 256 and 512 bins, that is high dimensional vector spaces. Similarity measures can be evaluated using distance functions, and it is important to determine the most suitable function for each type of histogram. The following metrics were experimentally evaluated. Let H, H' be histograms, both with M bins, and H_m be the m-th individual bin.

- Euclidean distance: used to evaluate distances in n-dimensional vector spaces

$$L_2(H, H') = \left(\sum_{m=1}^{M} (H_m - H'_m)^2 \right)^{\frac{1}{2}}$$

- Histogram intersection: evaluate which histogram bins share the same frequency values

$$D_\cap(H, H') = \sum_{m=1}^{M} min(H_m, H'_m)$$

- Jensen-Shannon divergence: a metric based on information theory

$$D_{jsd}(H,H') = \sum_{m=1}^{M} H_m log\frac{2H_m}{H_m + H'_m} + H'_m log\frac{2H'_m}{H'_m + H_m}$$

- Relative-Bin Deviation: based on the statistical variation coefficient

$$D_{rbd}(H,H') = \sum_{m=1}^{M} \frac{\sqrt{(H_m - H'_m)^2}}{\frac{1}{2}\left(\sqrt{H_m} + \sqrt{H'_m}\right)}$$

- Chi-square distance: adapted from the chi-square test

$$D_{\chi^2}(H,H') = \sum_{m=1}^{M} \frac{H_m - H'_m}{H_m + H'_m}$$

For each feature, we experimentally found the most appropriate metric that best discriminate images in the collection. This extensive evaluation is widely detailed in [2]. Many features can be evaluated in an individual metric using a linear combination approach of the feature-metric pairs, according to the equation 3. Each w_k is a factor that controls the relative importance of each feature-metric pair. A set of good values for all w_k, in the same data set, were found in [2].

IV. EFFICIENT INDEXING

When a user makes a query to the system, the actual problem is to identify the subset of most similar images. Therefore, we need to consider the problem of accessing a set of images using a similarity measure. Let $\mathcal{M} : \mathcal{I}x\mathcal{I} \rightarrow \mathbb{R}$ be a similarity function between two image features. The problem of retrieving the most similar images from the database is reduced to the problem of identifying the nearest neighbors according to $\mathcal{M}$ [8].

Several algorithms have been developed to perform the indexing task on vector spaces, which take advantage of geometric and coordinate information (e.g. kd-trees, R-trees). These methods work under the assumption that the Euclidean distance is enough to represent the similarities between objects [4]. However, in this work image features are histograms and it was experimentally found that other metrics such as Jensen-Shannon Divergence and Relative Bin Deviation give a better similarity notion [2]. Therefore, an indexing method based on similarity functions was chosen and evaluated.

This section presents two approaches to find the most similar images to a query image in a database. The aim of these methods is to decide which results have to be presented to the user as fast as possible.

A Sequential Scan

The sequential scan is a simple algorithm that compares all images in the database with the example image. The general definition of the sequential scan is shown in Algorithm 1, which receives as input a query image $x \in \mathcal{I}$, the similarity measure and the image database . This algorithm returns as output the distance measure $\mathcal{M}$ between the image x and all images into the database $\mathcal{D}$, sorted by closeness. This algorithm is regarded by two runtime characteristics that make it computationally expensive: the number of required similarity computations and the sorting cost, both depending on the size of the database.

The sequential scan is widely applied in experimental settings, to test mainly the precision of similarity measures more than the computational performance of the system. It is actually not usable in real scenarios, because large a number of images in the database will generate prohibitive response times.

Algorithm 1 Simple image retrieval algorithm

```
Inputs: QueryImage x
        SimilarityMeasure sm
        ImageDatabase d
Output: SimilarityResults sr
Begin
    foreach y in d
        score = sm.evaluate(x,y)
        sr.add( score )
    end
    sr.sort()
End
```

B. Metric indexing

The purpose of an indexing method is to avoid the evaluation of distances to many non-relevant images. The algorithm used in this system is a modi?cation of the Linear Approximating and Eliminating Algorithm (LAESA) [7] for ?nding k-nearest neighbors. LAESA is a similarity search method working in metric spaces, where no vector representation of the data is required. This algorithm is based on the triangle-inequality property to eliminate irrelevant data as shown in Fig. 2.

The LAESA algorithm has two main procedures: the preprocessing procedure and the search procedure. The preprocessing procedure selects a subset of base prototypes (index pivots) $\mathcal{B}$ from the database $\mathcal{D}$, to build an index structure that contains distances between images in both sets. The search procedure uses a *branch and bound* strategy to discard non-relevant images with two main steps: approximation and elimination. The approximation step selects a new image closer to the query while the elimination step decides whether one image is near enough to be relevant or not. Both steps use the information of a lower bound for the distance between each image into the database and the query image, which is calculated as:

$$g_q(u) = max_{p \in \mathcal{B}} \{|d(q,p) - d(p,u)|\} \quad (4)$$

Although LAESA was originally designed to search the nearest neighbor, it is used to solve k-nearest neighbor queries by being stopped when approximately *k* images have not been discarded yet [9]. This heuristic has some problems:

- It is highly dependant on the order in which the images are evaluated.
- When the algorithm is stopped the images that have not been evaluated yet are taken as relevant though their distances to the query are unknown.
- The error rate increases with *k*.

We propose an algorithm which does not discard elements based on the distance to the current nearest neighbor but based on an estimation of the distance to the k-nearest neighbor. It can be proven that if we have a perfect estimation of this value the retrieved elements will be exactly the k-nearest neighbors. Let *e* be the value for the estimated k-nearest neighbor distance. At each iteration of the algorithm lower bounds are computed for all the elements in the database, choosing randomly a pivot from $\mathcal{B}$, and those with a lower bound greater than *e* are pruned. If the number of retrieved elements is far from *k*, *e* is adjusted by an approximation factor *f* and the process is repeated. Finally, the results are sort and retrieved. Algorithm 2 presents pseudocode for this procedure.

Algorithm 2 The modified LAESA algorithm scheme

```
Inputs: QueryImage x
        SimilarityMeasure sm
        ImageDatabase d
        BasePrototypes b
        RequestedResults k
        ApproximationFactor f
        KNNDistanceEstimation e
        k-Tolerance eps
Output: SimilarityResults sr
BEGIN
  REPEAT
     s = b.getArbitraryPrototype()
     n_results = 0
     FOREACH p IN d
        lb = estimateLowerBound(p,x)
        if lb > e eliminate(p)
        else n_results = n_results + 1
     END
     e.adjust(k-n_results, f);
  UNTIL (|k-n_results|<=eps)
END
```

One important aspect of LAESA is the index structure, that is, the table with previously calculated distances between the base prototypes and all images in the database. The main idea with the selection of these base prototypes is to find representative points in the search space. Moreno-Seco et al. [10] suggest selecting a set of base prototypes that are maximally separated because it has been shown that the number of distance computations is reduced in the following steps. In this work, the base prototypes are selected by the calculation of the medoid in each of the 18 semantic groups identi?ed by pathologists.

In order to compute the medoid of each semantic group, the distance between images is used to determine which image has the minimum distance sum with respect to the others in the same group.

For estimating the distance to the k-nearest neighbor *e* a random sample **S** of images from the database is taken. Using sequential scan the distances to the k-nearest neighbor for each image in **S** is obtained and the 90th percentile of these values is chosen. The approximation factor *f* is determined by exhaustive evaluation of values in (0,1), such that the precision of the system is maximized while keeping the querying time low.

V. EXPERIMENTATION

A Experimental design

This evaluation aims to analyze the general system performance comparing the sequential scanning algorithm vs. the metric indexing. The measurement of the system efficiency is based on the response time taken to retrieve a set of results. This system was implemented in the Java programming language and all experiments were run under GNU Linux, in a standard PC with an Intel Pentium D processor and 1 GB of RAM memory.

The feature extraction phase is not included in this performance evaluation, since these computations are made only when images are first added to the database. Once images are archived, features are also stored to be used later in a retrieval algorithm, and the time spent in the feature extraction phase does not impact the response time in the retrieval phase.

Reported times are in milliseconds and correspond only to the time required to execute the retrieval algorithm. An efficiency experiment is comprised of 30 random queries, in each query the run time is measured and then averaged with others to get a mean value. First, experiments to evaluate the sequential scanning algorithm alone were made, in which every available similarity measures were tested to identify their general computational load. Then the metric indexing was evaluated using the semantic metric and the low-level feature combination to compare its performance against the sequential scan.

Since the metric indexing retrieves a subset of images in the database, the precision is also measured to analyze which is the error introduced by an approximate search. The precision in the first-retrieved image is averaged in each experiment, e.g. when a subset of 100 images is required, 30 queries are randomly performed and the average precision in the ?rst-retrieved image is reported.

B. Experimental results and discussion

The required time to identify a set of similar images using sequential scanning is O(n), with n the number of images in the database. This is corroborated by plots in Fig. 3: while the number of images in the database increases, the response time also increases linearly.

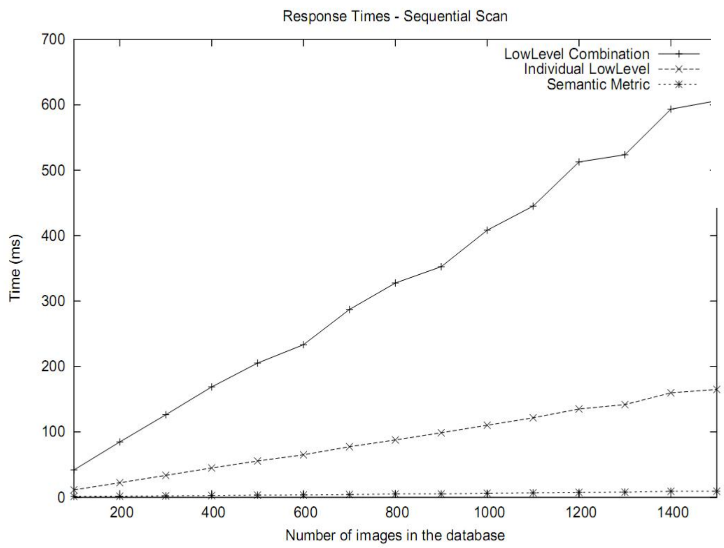

Fig. 3. Response times for sequential scan for different database sizes.

An important fact to take into account is that the retrieval algorithm has a computational load that depends on the cost of the underlying metric. The LAESA algorithm has also a linear complexity both in time and space. The main idea in this algorithm is to avoid the explicit calculation of the distance, whenever is possible, to reduce the response time, using the metric computation only when it is really needed.

Using the metric indexing, the system is able to identify a set of relevant images with a predetermined size. Thus, if the user needs to find *k* relevant images, the system does not need to calculate the similarity between all images into the database. In this experimentation, different *k* values are evaluated, starting with $k = 50$ to $k = 1000$, with steps of 50. Indexing response times are analyzed in the following subsections.

Fig. 4 shows the response time trend of the sequential scan and the metric indexing, using as underlying metric the low-level feature combination. The sequential scan tends to remain constant near to 600 ms. Results shows an important performance increase when using the metric indexing. In particular, when only 50 images are needed, the LAESA algorithm takes 57 ms demonstrating a performance improvement of 90%. When k takes a value of 100, the 80% of time is saved. To retrieve datasets under 250 images, this improvement still remains up to 80%.

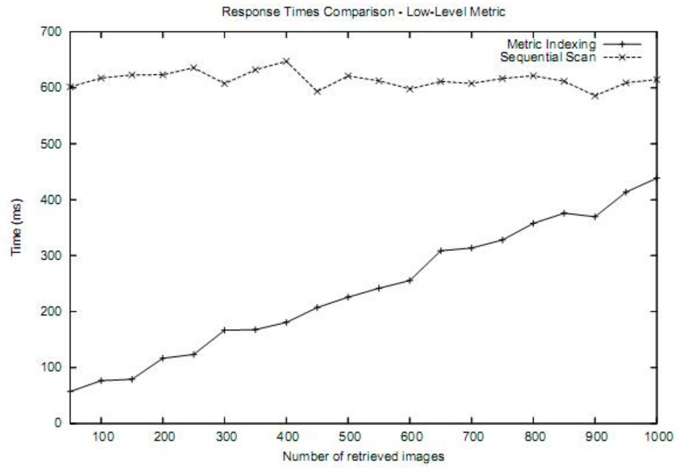

Fig. 4. Response times for sequential scan and metric indexing using the low-level metric

According to the plot, the metric indexing outperforms the sequential scan in average with a 60% of improvement. It is important to point that this average improvement was calculated through different *k* values, and for fixed values, times are as previously commented. Results on the low-level metric shows that the metric indexing effectively avoids many distance computations, and this difference is better shown by the fact that the low-level metric is expensive.

The metric indexing helps to save time in identifying the *k* most similar images, but another important question is if those images are really the *k*-nearest-neighbor images. Fig. 5 shows that in general, the metric-indexing precision trends to remain near to the sequential-scan precision. Taking in account that the sequential scan retrieves exactly the *k* most similar images, it makes sense to compare the metric-indexing precision with the sequential-scan precision. The plot shows many differences between algorithm precisions, with a maximum-precision decrease of 30% and in average 12% trough all the experimentation. This precision decrease may be explained by a non-metric behavior of the low-level features combination, which violates the triangular inequality under some situations. In general, the precision of both algorithms oscillates around the 65% as was expected according to the effectiveness evaluation.

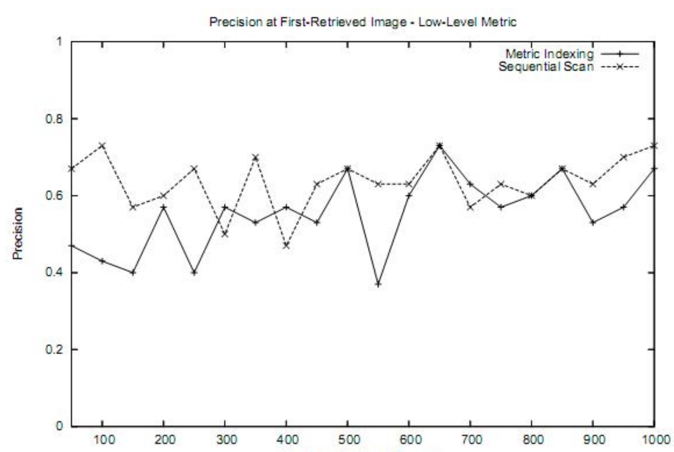

Fig. 5. Precision rates using both metric indexing and sequential scan.

VI. CONCLUSION

A method for efficient image retrieval was implemented. The LAESA algorithm, mainly used in classification tasks, was adapted to find the k-nearest neighbors of a query image. This algorithm takes advantage of properties of metric spaces in which images are represented. The experimental results show that this indexing scheme can make important improvements in terms of system efficiency. Since interactive response is a requirement for image retrieval, defined as response times lower than one second, the evaluation shows that this system is in essence interactive.

The algorithm shows important improvements in time responses with regard to the sequential scan, when the metric calculation is expensive. However, when the metric computation is also fast, the algorithm still shows better performance than the sequential scan, because it effectively prunes irrelevant images. An important factor for the success of the metric-indexing strategy implemented by the algorithm is a good choosing of pivots. In this work, pivots were chosen based on semantic information, i.e. conceptual image groups were identified by an expert. In both situations, low-level metric and semantic metric, the medoids of each conceptual group were identified and used as base prototypes to calculate lower bounds in order to prune non-relevant images.

REFERENCES

[1] Alberto S. Aguado, Mark S. Nikson. *Feature Extraction and Image Processing*. Elsevier, 2002.

[2] Juan C. Caicedo., Fabio A. Gonzalez and Eduardo Romero. Evaluación de características de bajo nivel para recuperación de imágenes de histopatología por contenido. *II Congreso Colombiano de Computación*. Pontificia Universidad Javeriana, 2007.

[3] Davide Caramella. Is pacs research and development still necessary? International Congress Series, 1281:11-14, 2005.

[4] E. Chavez, G. Navarro, R. Baeza-Yates, and Jose L. Marroquin. Searching in metric spaces. *ACM Computing Surveys* (CSUR), Volume 33. Issue 3, 273-321, 2001.

[5] Bjorn Johansson. A survey on: Contents based search in image databases. 2000.

[6] Thomas M. Lehmann, Mark O. Güld, Christian Thies, Benedikt Fischer, Klaus Spitzer, Daniel Keysersa, Hermann Neya, Michael Kohnenb, Henning Schubertb, and Berthold B. Weinb. The irma project: A state of the art report on content-based image retrieval in medical applications. *In Korea-Germany Workshop on Advanced Medical Image*, page 161-171, 2003.

[7] Luisa Micó, Jose Oncina, and Rafael C. Carrasco. A fast branch and bound nearest neighbour. *Pattern Recognition Letters*, 17:731-739, 1996.

[8] Nicolas Moënne-Loccoz. High-dimensional access methods for efficient similarity queries. Technical report, CentreUniversitaire D'Informatique. Computer Vision and Multimedia Laboratory, 2005.

[9] Francisco Moreno-Seco, Luisa Micó, and Jose Oncina. A modification of the Laesa algorithm for approximated k-nn classification. *Pattern Recognition Letters* 24, pages 47-53, 2003.

[10] Francisco Moreno-Seco, Jose Oncina, and Luisa Micó. Improving the Laesa algorithm error rates. *Proceedings of the VIII Symposium Nacional de Reconocimiento de Formas y Análisis de Imágenes*, pages 413-419,1999.

[11] Henning Müller, Nicolas Michoux, David Bandon, and Antoine Geiss-buhler. A review of content based image retrieval systems in medical applications clinical benefits and future directions. *International Journal of Medical Informatics*, 73:1-23, 2004.7

[12] Robert Sigal. Pacs as an e-academic tool. *International Congress Series*, 1281:900-904, 2005.

[13] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content based image retrieval at the end of the early years. Technical report, Intelligent Sensory Information Systems, University of Amsterdam, 2000.

[14] Remco C. Veltkamp and Mirela Tanase. Content-based image retrieval systems: A survey. *Technical Report UU-CS-2000-34, Dept. of Computing Science*, *Utrecht University, 2000.*

Edwin A. Niño. Earned his bachelor's degree in computer science and engineering from the National University of Colombia at Bogota, and is currently a master student in computer science and engineering at the same university.

Juan C. Caicedo. Earned his bachelor's degree and master's degree in computer science and engineering from the National University of Colombia at Bogota, and is currently a PhD student in computer science and engineering at the same university.

Fabio A. González. Earned his bachelor's degree in computer science and engineering, and his master's degree in mathematics from the National University of Colombia at Bogota. Then, he earned a master's degree in computer science and his PhD in computer science from the University of Memphis, USA. He is currently an Associate Professor of the Systems and Computer Engineering Department at the National University of Colombia.

Universidad Nacional de Colombia Sede Medellín
Facultad de Minas

120 años
TRABAJO Y RECTITUD

Reseña Histórica

La Escuela Nacional de Minas fue fundada el 11 de abril de 1887, bajo la dirección del general Pedro Nel Ospina como rector y como Vice-rector Luís Tisnés, aunque el general Pedro Nel Ospina no se posesiono, elaboro con ayuda de su hermano Tulio los estatutos y reglamentos de la escuela, los cuales fueron una adaptación de los estatutos y reglamentos de la Escuela de Minas de California (Berkeley)los cuales fueron cambiando de acuerdo a las necesidades de cada década, en ellos se fomento una filosofía con valores cívicos, éticos y de orden por medio del estimulo y el ejemplo que comprometían el comportamiento del estudiante no solo dentro de la escuela sino fuera de ella, a demás se introdujeron hábitos de sobriedad, de economía y principios morales de honradez, honestidad y respeto.

En sus inicios contó con 22 alumnos matriculados, y luego de tres meses fue cerrada por la poca cantidad de estudiantes, fue reabierta un año después, el 2 de enero de 1888, bajo la rectoría de Tulio Ospina V, esta vez contó con 27 alumnos matriculados y con un plan de estudios de 4 años de un mejor control de los programas curriculares y adaptarlos a nuevas condiciones adelantándose a las necesidades futuras de la educación y asegurando así un buen desempeño de los futuros profesionales.

En 1906 la Escuela Nacional de Minas se anexo a la universidad de Antioquia, a la que perteneció durante cinco años más, en 1911 paso a ser de nuevo una entidad independiente.

En 1940 la institución fue incorporada a la Universidad Nacional y continúo con el nombre de Escuela Nacional de Minas, ese mismo año comenzó la construcción de la actual sede, la cual fue inaugurada el 19 de diciembre de 1944, en el marco del primer Congreso Nacional de Ingenieros.

Entre 1941 y 1950 se crean las carreras de ingeniería geológica y petróleos y arquitectura, la cual se separo de la facultad de Minas en 1954, en1960 se crea la carrera de ingeniería administrativa, luego se crearon los programas de ingeniería industrial, ingeniería mecánica e ingeniería química y se separaron los programas de ingeniería geológica y petróleos en dos programas diferentes, actualmente la Facultad de Minas Administra 11 programas de pregrado en ingeniería, 17 de posgrado y cuatro doctorados.

La Facultad a lo largo de su existencia ha sido motora del desarrollo de la ciudad, del departamento y del país, a través de sus 12.000 egresados quienes han constituido la mayor parte del personal dirigente y técnico en las explotaciones mineras, las construcciones de distinto tipo, la infraestructura vial, los desarrollos hidroeléctricos, las obras de abastecimiento de agua, las obras sanitarias y la industria, así como en los planes de desarrollo físico, económico y social.