MEDINFO 2004
M. Fieschi et al. (Eds)
Amsterdam: IOS Press

IRMA – Content-Based Image Retrieval in Medical Applications

Thomas M. Lehmann[a], Mark O. Güld[a], Christian Thies[a], Bartosz Plodowski[a], Daniel Keysers[b], Bastian Ott[c], Henning Schubert[c]

[a] *Department of Medical Informatics, Aachen University of Technology (RWTH), Aachen, Germany*
[b] *Chair of Computer Science VI, and* [c] *Department of Diagnostic Radiology, RWTH Aachen, Germany*

Abstract

The impact of content-based access to medical images is frequently reported but existing systems are designed for only a particular modality or context of diagnosis. Contrarily, our concept of image retrieval in medical applications (IRMA) aims at a general structure for semantic content analysis that is suitable for numerous applications in case-based reasoning or evidence-based medicine. Within IRMA, stepwise processing results in six layers of information modeling (raw data layer, registered data layer, feature layer, scheme layer, object layer, knowledge layer) incorporating medical expert knowledge. At the scheme layer, medical images are represented by a hierarchical structure of ellipses (blobs) describing image regions. Hence, image retrieval transforms to graph matching. The multilayer processing is implemented using a distributed system designed with only three core elements. The central database holds program sources, processing schemes, images, features, and blob trees; the scheduler balances distributed computing by addressing daemons running on all connected workstations; and the web server provides graphical user interfaces for data entry and retrieval.

Keywords

Information Storage and Retrieval; Information Management; Pattern Recognition; Image Processing, Computer-Assisted.

Introduction

Content-based image retrieval (CBIR) aims at describing the complex object information of digital images by non-textual features, which are applicable for efficient query processing. In recent reports, some approaches for content-based retrieval have been published, which are specially designed to support medical tasks. Korn et al. describe a system for fast and effective retrieval of tumor shapes in mammogram x-rays [1]. This approach has certain restrictions on both the images (mammography only) and the features (tumor shapes only) which are supported by the system. Likewise, the automatic search and selection engine with retrieval tools (ASSERT) operates only on high resolution computed tomography of the lung [2]. A physician delineates the region bearing a pathology and marks a set of anatomical landmarks when the image is entered into the database. Hence, ASSERT has extremely high data entry costs, which prohibit its application for clinical routine. Long et al. access a large collection of 17,000 spine radiographs by means of shape analysis, where biomedical categories such as "anterior osteophytes present/not present" are distinguished automatically [3]. The data entry costs are low, but queries are limited to the pre-defined categories. Chu et al. present a knowledge-based image retrieval system with spatial and temporal constructs [4]. Brain lesions are extracted automatically within three-dimensional data sets from computed tomography and magnetic resonance imaging. Their representation models knowledge-based layer that provides a mechanism for accessing and processing spatial, evolutionary, and temporal queries. Nonetheless, those concepts for medical image retrieval are task-specific, i.e. limited to a distinct modality, organ, or diagnostic study and, hence, usually not directly transferable to other medical applications.

In this paper, we present an approach for content-based image retrieval in medical applications (IRMA) and sample applications to show the general applicability of the concept.

Materials and Methods

Semantic layers of information modeling

IRMA splits the retrieval process into seven consecutive steps of processing (Fig. 1). Each step represents a higher level of image abstraction and content understanding [5].

The *categorization* step aims at determining for each image entry the imaging modality and its orientation as well as the examined body region and functional system. For that, a detailed hierarchical coding scheme was developed [6], which exceeds the complexity of existing tags of the digital imaging and communications in medicine (DICOM) standard, such as (0018/0015) "body part examined" or (0018/5100) "patient position", but could be consistently integrated to supplement the standard. Automatic categorization is based on a reference database of images selected arbitrarily from clinical routine and classified by experienced radiologists.

Registration in geometry (rotation, translation, scaling) and contrast generates a set of transformation parameters that is stored for the corresponding image in each of its likely categories and utilized at higher layers of abstraction. In consent with Tagare et al. [7], registration is based on prototypes which are manually defined for each category, and further incorporate medical expert knowledge into the IRMA system.

The *feature extraction* step derives local image descriptions, i.e. a feature value (or a set of values) is obtained for each pixel. These can be category-free (e.g. resulting from edge detection or

regional texture analysis) or category-specific, such as the application of an active shape model that explicitly uses a-priori knowledge derived from the respective category.

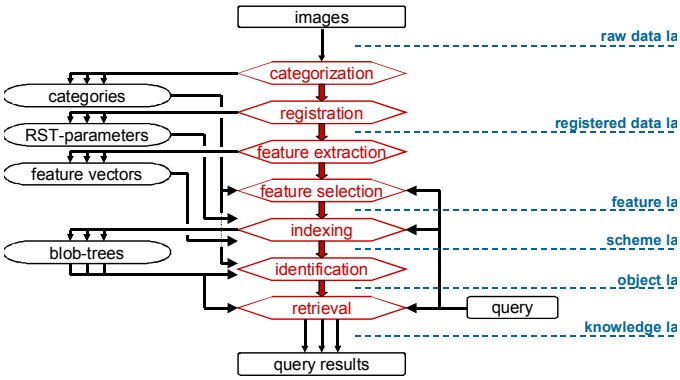

Figure 1 - Processing schemes and semantic layers

Decoupling *feature selection* from feature extraction allows to integrate both image category and query context into the abstraction process. For instance, the same radiograph might be subject to fracture or cancer examination, resulting in a contour-based or texture-based combination of features (feature sets) such as the contour set or texture set, respectively. In order to avoid exhaustive computation during query processing, these feature sets are pre-computed for each image in each likely category.

Indexing provides an abstraction of the previously generated and selected image features, resulting in a compact image description. According to the selected feature set, this is done via clustering of similar image parts into regions represented by their second area moment description as ellipses ("blobs"). In contrast to the Blobworld approach [8], this is done at multiple resolutions yielding a multi-scale blob-representation of the image ("blob tree"). Note that hierarchical indexing enables the processing of regions of interest (ROIs), which are marked by the user when issuing a query.

The *identification* step provides linking of medical a-priori knowledge to certain blobs generated during the indexing step. It relies on the prototypes defined for each category, which are labeled locally by medical experts, and the corresponding parameters for geometry and contrast registration. Thus, identification is the fundamental basis to introduce high-level image understanding by analyzing regional or temporal relationships between the blobs.

In IRMA, the *retrieval* itself is processed either on the abstract blob level or referring to identified objects. Note that only the retrieval step requires online computations while all other steps can be performed automatically in batch mode at entry time of an image into the database. This, of course, requires offline computation of all paths generated by the categorization and the feature selection step.

Feature representation and distance computation

Image categorization is performed by means of *global features*, i.e., a single value or a vector combining a few values is assigned to the entire image. A large number of global features have been proposed in the literature for content-based image retrieval. Besides major components from color and gray scale histograms or the moments of dominant regions, we focus on global measures obtained from frequency, texture, and structure analysis. The feature values obtained for each image are combined to a feature vector which is then used for k-nearest-neighbor classification based on e.g. the Euclidian or Simard's tangent distance [9]. The latter is able to cope with local geometry and contrast differences.

It is important to realize that we do not aim at clustering our feature space in order to find suitable categories but that we apply as many global features as required to distinguish the categories, which are given a-priori. Furthermore, the most likely categories are tracked through the following steps of processing.

Local features are assigned to each image pixel. Category-free local features are extracted uniformly for all images. In other words, all methods for local feature extraction are applied to each image. With respect to image category and query content, suitable feature sets are selected to enable query-adaptive processing without additional computation at the time of query processing.

Following the Blobworld approach [8], dominant image regions are approximated by their best fitting ellipses to which the mean feature vector of the entire region is assigned. While Blobworld applies an expectation maximization clustering technique within the feature space, the partitioning in IRMA is computed in the image domain by means of an edge-preserving region growing algorithm [10]. This ensures connected segments and a complete image partitioning, and permits a hierarchical image decomposition. On the lowest level, each pixel builds its own blob while on the highest level, the entire image is represented by a single blob. In between, a tree structure of blobs is obtained and computing the similarity of images transforms into graph matching.

Figure 2 exemplifies the multi-scale abstraction that is modeled by the IRMA system. Each image is partitioned into representations with decreasing number of regions. The regions are represented by their best fitting ellipsoids. These ellipsoids form the nodes of the resulting graph. Different edge types code the hierarchical tree structure of the graph, adjacencies within a level of the multi-scale decomposition, and adjacencies of nodes crossing the levels of decomposition.

Data management and query processing

Regardless of their semantic layer, all feature extraction and evaluation steps are modeled as methods transforming features. The method encapsulates a program and its parameterization. Only three types of methods are sufficient to model arbitrary steps of retrieval algorithms: a T:1-method transforms a feature set into a single feature (e.g. generating a matrix for statistical feature reduction by principal component analysis), a 1:1-method transforms one feature into another (e.g. computing a spatial convolution of an image), and a 1:T-method transforms one feature into a set of features (e.g. generating multiple representations of one image via small transformations). Queries are modeled as a network of interconnected user-implemented methods with a flexible feature input/output interface. Similar to dataflow process networks, we use directed acyclic graphs composed of methods and control elements [11]. Parameters and in-

formation on reference images and prototypes are stored in an experiment description. Since experiments are composed of methods, a structured generation history for each feature allows the automatic identification and re-use of already computed features.

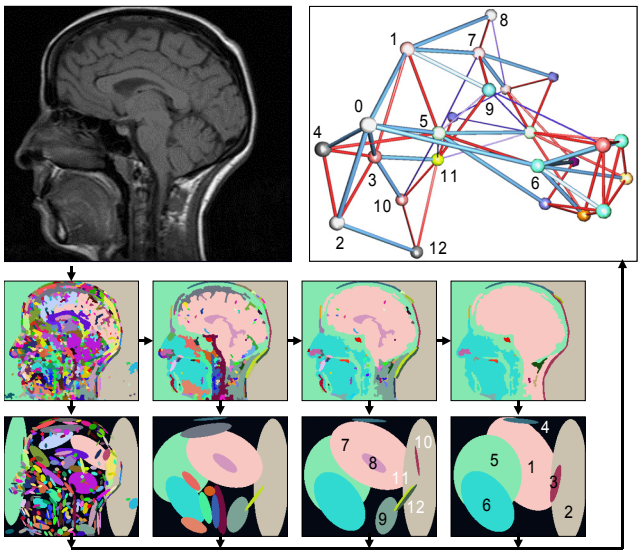

Figure 2 - Image, partitioning, blobs and resulting graphs

Hence, the manifold structure of information abstraction is reduced to a small number of mechanisms that can be handled within a distributed client-server architecture (Fig. 3). The central components (database, scheduler, web server) run on a server. The client processes (daemons, programs, applications) are used for distributed computing in a cluster [12].

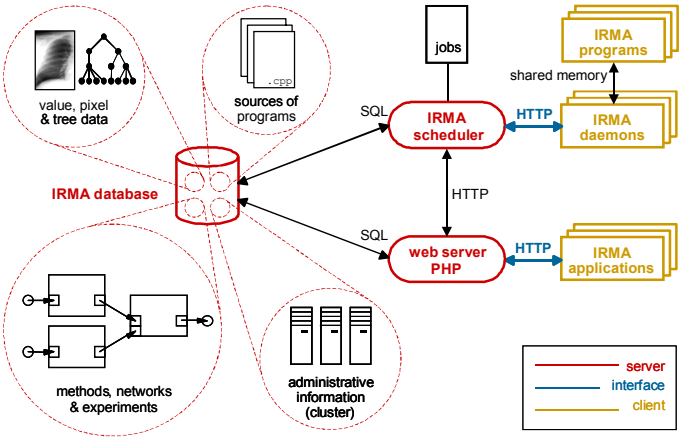

Figure 3 - System architecture

The central relational *database* is used to store administrative information about physical entities, i.e. single value, pixel and tree data as well as sources of programs, and logical entities, i.e. methods, networks, and experiment definitions characterizing the algorithms for image processing and retrieval, their parameters, and the feature sets in use. The physical entities are stored as files outside the database and can be hosted by any computer within the distributed system. Using the information about the cluster infrastructure, transparent access to and automatic replication of all physical entities are implemented.

The *scheduler* is a central service that manages the execution of all queries or feature extraction tasks. It has two functional parts. For each invoked query, the process control sub-part creates a data structure to log the progress during the execution of the corresponding experiment. For each node of the network ready to be executed, a job is generated, which includes the method identifier (ID), the IDs of the input features and their locations, and allocated IDs for the output features. The communication subpart assigns jobs to programs running in the cluster. If the program needed by a job is not running, the scheduler selects an appropriate host and issues the IRMA daemon on this machine to start the program. Additionally, the scheduler can order idle programs to terminate in case this program will not be required within a look-ahead interval. Upon the completion of a job, the scheduler receives a notification from the program and updates the feature information in the database.

Each computer in the system runs an IRMA *daemon*. This background process automatically installs new programs on its host and starts them on demand. The daemon service is also used to inform the scheduler about the current load of its host. Furthermore, it can detect possible abnormal terminations of programs and report them to the scheduler for fault handling. The daemon also performs the automated program transfer.

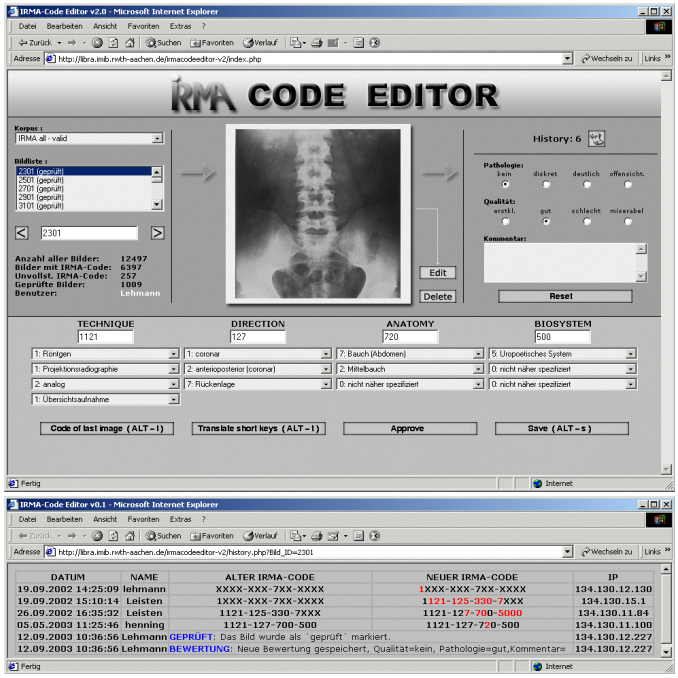

Figure 4 - Code editor for manual reference catégorization

Modular interfaces to support queries within the IRMA system are generated by the *web server* using the hypertext preprocessor PHP. Besides the query by example strategy, two basic mechanisms are provided for medical applications: relevance facts explain to the user why a certain picture has been presented as a query result, and relevance feedback allows the user to correct, adapt, or modify his query for query refinement.

A standard web browser is used as graphical user interface (GUI) for all IRMA *applications*. This has numerous advantages. IRMA applications are running platform independent on any computer connected to the Internet. In addition, physicians are used to handle web interfaces and, therefore, they do not need a special instruction to use IRMA applications. For instance, the IRMA code editor is used for manual labeling of reference images (Fig. 4a). The IRMA code can be edited either directly by typing in the code or by selecting the entries from selection boxes. Based on prior selections, the sub-codes offered are adopted properly. Based on extended query refinement loops, all changes are recorded in a history protocol stored in the central database for easy error recovery (Fig. 4b).

Extended query refinement in IRMA applications

Especially in medical imaging, a retrieval interface must offer functionality for initialization of a query, output of the answer, resubmitting a query for query refinement, accessing previous results, and for merging results. These functions are grouped into the following classes of modules (Fig. 5).

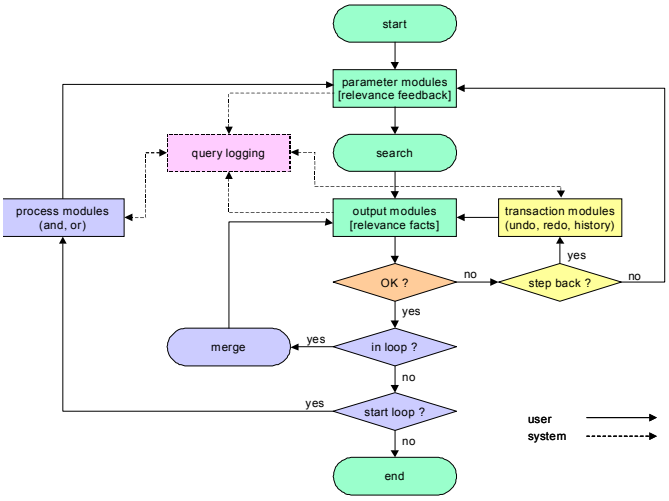

Figure 5 - Extended query refinement

Output modules contain all functionality regarding the visualization of information such as images, descriptions, parameters, etc. These modules are used for initialization of the query and to display the query result. In particular, output modules display relevance facts. Parameter modules allow the user to interact with the system. Input fields, radio buttons, sliders, or scroll bars are some prominent examples. Parameter modules are used whenever a query is initialized or (re-)submitted. In particular, they support relevance feedback. Transaction modules allow to step back and forward (UNDO/REDO-functions) and support a direct restoration of any steady state the system has been before within the current session (HISTORY-functions). Process modules provide the ability to union or cut intermediate sets of query results, which already have been looped for query refinement. Boolean AND and/or OR operations offer a variety of potentials for advanced query refinement. In addition, query logging is used to track the user's interaction with the system. Every action that is performed by the user is stored with a corresponding session identifier (ID), a unique timestamp and the IDs of all images answered within their current order and respective relevance facts. This query logging is fundamental for any process or transaction module.

Based on this modules, the flow chart of a retrieval session is straightforward (Fig. 5). At first, parameter modules are used for the initialization of the query. After performing the search, which compares the query with all images within the system by means of the selected abstract features, resulting images are displayed and relevance facts are given by means of the output modules. If the system offers relevance feedback, a loop from output to parameter modules is added. The transaction modules add a second loop to the flow chart intra-connecting the output models. Note that this loop is supported by the system's query logging, which also reads from the parameter and the output modules. Finally, a third loop provides extended query refinement by combining successive but independent queries based on the same database.

Results

All technical details of the IRMA system are kept as transparent as possible for all participants. There is no need for the programmer to take care of platform-related communications regarding the query context, feature access, or feature storage. In particular, the storage location is transparent. Furthermore, the programmer is disburdened from data-flow synchronization, i.e. concurrency transparency of processing steps. For the physician, the execution of a query does not require additional technical knowledge about the underlying implementation of the retrieval process, i.e. all steps of the process run fully automatically without further user interaction.

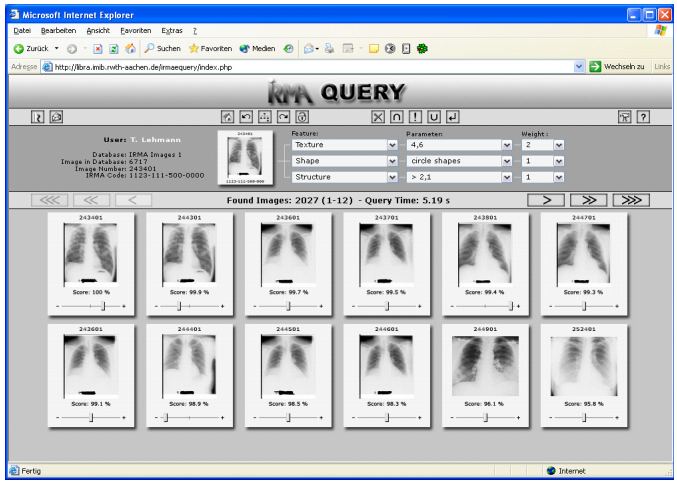

Figure 6 - IRMA query interface

Currently, the IRMA reference database holds 12,479 images. Already 6,397 images have been labeled according to the IRMA code. Figure 6 shows a query interface that encapsulates all loops of interaction within a single screen. It is used to retrieve medical images which are similar with respect to global image features. Since the processing scheme of IRMA models only

methods, these experiments prove the validity and applicability of the IRMA concept in general.

Discussion

In addition to the requirements originating from content-based retrieval in general [13], the design of a medical image retrieval system requires attention to several other aspects and domain specific properties [7]. In particular, the IRMA system supports modular design of arbitrary retrieval algorithms. Modularity easily enables the verification of isolated processing steps and allows the re-use of programs for various experiments and applications. All kinds of features (global, local, blob trees) are uniformly accessible. The system supports the automatic transfer of new and updated processing components into the pool of retrieval algorithms available to the physicians. In addition, new algorithms can quickly access the image database shortening the cycles between development and testing.

The changeability of a medical image database is the single most important aspect of building content-based image retrieval systems in medical applications [7]. Based on the transparent mechanisms for distributed computing, data replication, and program installation, IRMA relies on a generic scheme for image content abstraction. It enables the image and feature database to evolve considerably over the lifetime of the system. Furthermore, image features used in IRMA are computed automatically and are not influenced by knowledge bias arising from a gestalt-driven diagnostic interpretive process. Therefore, IRMA database activity is decoupled from interpretation activity [7].

Conclusions

In contrast to specific applications, IRMA presents a general CBIR approach for medical images. It combines a central database with a distributed system architecture and, therefore, is suitable for large image databases such as within picture archiving and communication systems. IRMA supports rapid prototyping and quick integration of novel image analysis methods. Therefore, it narrows the gap between the semantic imprint of an image and any alphanumerical description that is always incomplete.

Acknowledgments

The IRMA project has been funded by the German Research Community (DFG, grants Le 1108/4 and Le 1108/6). Further information on IRMA is available at http://irma-project.org. We are grateful to Tim Dwyer, Department of Computer Science, University of Melbourne, Australia, for providing the WilmaScope 3D graph visualization system used to display the IRMA blob trees http://www.wilmascope.org.

References

[1] Korn P, Sidiropoulos N, Faloutsos C, Siegel E, Protopapas Z: Fast and effective retrieval of medical tumor shapes. *IEEE Trans* KDE 1998; 10(6): 889–904.

[2] Shyu CR, Brodley CE, Kak AC, Kosaka A, Aisen AM, Broderick LS: ASSERT – A physician-in-the-loop content-based retrieval system for HRCT image databases. *Computer Vision and Image Understanding* 1999; 75(1/2): 111–132.

[3] Long LR, Antania S, Leeb DJ, Krainakc DM, Thomaa GR: Biomedical information from a national collection of spine x-rays – Film to content-based retrieval. *Procs SPIE* 2003; 5033: 70–80.

[4] Chu WW, Hsu CC, Cárdenas AF, Tiara RK: Knowledge-based image retrieval with spatial and temporal constructs. **IEEE Trans KDE** 1998; 10(6): 872–888.

[5] Lehmann TM, Wein B, Dahmen J, Bredno J, Vogelsang F, Kohnen M: Content-based image retrieval in medical applications: A novel multi-step approach. *Procs SPIE* 2000; 3972: 312–320.

[6] Lehmann TM, Schubert H, Keysers D, Kohnen M, Wein BB: The IRMA code for unique classification of medical images. *Procs SPIE 2*003; 5033: 440–451.

[7] Tagare HD, Jaffe CC, Dungan J: Medical image databases: A content-based retrieval approach. *JAMIA* 1997; 4: 184–198.

[8] Carson C, Belongie S, Greenspan H, Malik J: Blobworld – Image segmentation using expectation-maximization and its application to image querying. *IEEE Trans PAMI* 2002; 24(8): 1026–1038.

[9] Simard P, Le Cun Y, Denker J: Efficient pattern recognition using a new transformation distance, in: Hanson S, Cowan J, Giles C (eds): *Advances in Neural Information Processing Systems 5*, Morgan Kaufmann, San Mateo, CA, 1993; 50–58.

[10] Thies C, Malik A, Keysers D, Kohnen M, Fischer B, Lehmann TM: Content-based retrieval in medical image databases by hierarchical feature clustering. *Procs SPIE* 2003; 5032: 598–608.

[11] Lee EA, Parks TM: Dataflow process networks, *Procs IEEE* 1995; 83(5): 773–799.

[12] Güld MO, Thies C, Fischer B, Keysers D, Wein BB, Lehmann TM: A platform for distributed image processing and image retrieval. *Procs SPIE* 2003; 5150: 1109–1120.

[13] Smeulders AWM, Worring M, Santint S, Gupta A, Jain R: Content-based image retrieval at the end of the early years. *IEEE Trans PAMI* 2000; 22(12): 1349–1380.

Address for correspondence

Thomas M. Lehmann, Institut für Medizinische Informatik, D-52027 Aachen, Germany. Tel: +49 241 80-88793,

lehmann@computer.org.