Shape Sensitive Salient-object based Image Query

By
Estifanos Gashaw Kidane

Thesis

Submitted to the School of Graduate Studies of Addis Ababa University in partial fulfillment of the requirements for the Degree of Master of Science in Computer Science

June 2004

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

FACULTY OF INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

Shape Sensitive Salient-object based Image Query

By

Estifanos Gashaw Kidane

Name and Signature of Members of the Examining Board

1. **Dr. Solomon Atnafu, Advisor** _______________________
2. _______________________
3. _______________________

Acknowledgements

First, I would like to forward my deepest appreciation and thanks to my advisor Dr. Solomon Atnafu for his valuable suggestions, comments and guidance. Without his help, this work wouldn’t have been accomplished.

I also would like to thank all my instructors in the Department of Computer Science at Addis Ababa University, who work very hard to make the graduate program in computer science a success.

Finally, I would like to thank my mother Tsehay Bezabeh , my father Gashaw Kidane and the rest of my family for their support and who in one way or another have contributed to the success in my career.

Table Of Contents

List Of Tables

List of Figures

Abstract

Shape Sensitive Salient-object based Image Query

Estifanos Gashaw
Advisor: Solomon Atnafu (PhD)
June 2005

The availability of images in electronic format has resulted in large number of digital images being used for different applications. The growth in the usage and availability of digital images calls for an efficient means of image data management. This has led to the development of content based image retrieval (CBIR) systems, which uses visual contents of an image such as color, texture or shape for querying image databases. CBIR is well researched and there are a large number of image retrieval systems based on this technology.

Image retrieval matching can be performed using global features of the whole image (as in CBIR), but a comparison that considers part(s) of images for similarity-based retrieval is sometimes a more natural approach. Regions of an image that are of particular interest to humans are termed salient-objects of an image. Salient –object based image retrieval is not well researched, but there some developed system that model the salient object of the image using a Minimum Bounding Rectangle. In this work a new approach is proposed which considers the natural shape of the salient objects contained within an image. An Image segmentation tool is used to automatically identify salient objects of an image and store them in an object relational DBMS for multi-criteria image query (query based on both image content, positional predicates and metadata). Furthermore, some techniques have been proposed in this work to make the image retrieval system accessible within a pervasive computing environment.

Keywords: Salient-Object Based Image Retrieval, Image Database, Image Retrieval in Pervasive Environment, Image Segmentation, Multi-Criteria Image Query.

Chapter 1
Introduction

1.1 Image Retrieval

One of the reason for the massive use of Images or pictures nowadays is because images are a good means of conveying information compared to text hence the expression '*an image is worth of a thousand words*'. The availability of image in electronic format has resulted in large number of images being used for different applications. Images now play an important role in different fields including medicine, engineering, advertising, design, security, education, journalism and entertainment. The growth in the usage and availability of digital images calls for an efficient means of image data management.

Images can be stored on a disk file system or on an image database. Storing image in a database has many advantage including easy of manageability, security, backup and restore facilities and flexibility [29]. Retrieval of images of interest from such a varied collection of images could be frustrating. To alleviate such problems, the traditional approach was to annotate images with textual descriptions or keywords. The problem with the traditional approach is that manual annotation could be time consuming and tedious for large collection of images, it is subjective in that people may view different images differently and it is even difficult or even impossible to describe images fully or adequately with the help of metadata [5]. This has led to the development of image retrieval based on visual contents of the image- a technology referred in the literature as Content Based Image Retrieval (CBIR) [30, 5, 4, 3, 2, 1].

Content Based Image Retrieval (CBIR) uses visual contents of an image such as color, texture or shape for querying image databases. With such methods it is possible to query image archives by providing, for instance, an example image for similarity-based comparison. In some CBIR systems sketching can be used in image retrieval. IBM's QBIC (Query by Image Content) CBIR system allows users to search by sketching the image that's going to be retrieved or searched from the image archive [30, 2]. In some

CBIR systems it is also possible to retrieve images by composing some colors and the image retrieval system will return image that have colors close to the composed image [30, 1]. Basically CBIR system will return image that are close or similar to the image query.

In CBIR similarity-based matching is performed by considering the whole image (i.e. global features of the whole image and query image are used for similarity comparison between the two images). Similarity-based image retrieval is an area that is researched very well and many full-fledged content-based image retrieval system exits nowadays. A detailed exploration of existing CBIR systems, both commercial and experimental, can be found in [30].

Database Management Systems such as oracle nowadays can be used to store multimedia data like images, video, audio etc. Using existing DBMS instead of full-fledged CBIR system can be advantageous for storing and managing images because users are familiar with such databases systems and it is possible to reduce extra cost for purchasing a full-fledged CBIR system for image data management. Some CBIR system can be integrated into existing DBMS as a module. For instance Virage, a well know commercial CBIR application, is available as an add-on to database management systems such as Oracle or Informix [2, 5]. Although researches were done on systems for similarly-based image retrieval in the image processing domain, managing image data using an integrated similarity-based image retrieval system within existing DBMS is not researched very well. A work which lays a foundation for image data management within existing DBMS the one done by Solomon Atnafu [5]. The research domain for this thesis will also be on image data management in an existing object-relational DBMS using an integrated similarity-based image retrieval system.

Image retrieval matching can be performed using global features of the whole image, but a comparison that considers part(s) of images for similarity-based retrieval is sometimes a more natural approach [6]. Regions of an image that are of particular interest to humans are termed salient-objects of an image. In such system it could be possible for example to query images of human face by color of the eye, shape of the nose etc. This kind of image retrieval system is more effective in application domains where only part of the image is of interest. The research done by Dawit Bulcha in [6] treats modeling of salient-object based image retrieval in an object relational environment.

The theme of this thesis is also on salient-object based image retrieval. In this research a different approach will be used for salient-object based image retrieval. The work done in [6] uses an object approximation technique called Minimum Bounding Rectangle (MBR) for modeling the salient-objects of an image. The MBR in this work were determined manually through user interaction. Since the MBR does not consider the natural shape of the salient-object and manual extraction of MBR is subjective, there might be false hits and hence inaccuracy in the image retrieval. In this research the shape of the salient-objects that are extracted automatically will be considered using an image segmentation tool.

Users of image retrieval could use mobile and pervasive computing devices. Making an image retrieval system usable in a pervasive environment is an important issue that needs to be considered. The user of such system should be able to query the image by using keywords, by providing an image as an example or by providing a salient-object as an example to retrieve image that contains similar salient-objects. Using salient-objects as a convenient means of image retrieval in a pervasive computing environment is a domain of research that is not yet much explored. This thesis will also propose techniques for image retrieval system appropriate in a pervasive computing environment.

1.2 Problem Statement

Unlike other methods the salient objects to be identified and modeled in to a database in this thesis will be shape-based. The assumption is that by considering shape of salient-objects , errors can be avoided that will be incurred into the similarity-based image retrieval technique due to the distortions that will happen because of the existence of the background in the salient object. Identifying an appropriate image segmentation algorithm that considers the shape of the salient object and incorporating it as a module for identifying shape-sensitive salient objects, modeling these salient objects in an object-relational DBMS for an image query is the main problem of this research work.

The image retrieval system that's to be developed should be multi-criteria. . Such queries could be for example:

- Searching by keyword or textual description. For example, 'Give me a brain image from the image archive'.
- Searching by visual content. 'Give me a brain image(s) which look like this image'
- Searching based on salient-objects. For example, 'Give me a brain image which have a tumor which looks like this' or 'Give me the tumor from a brain image looking similar to the following tumor'.
- Searching based on topological relations. For example queries like 'Give me a brain image which has a tumor at the top left corner'.

With the emerging and new technologies like pervasive computing growing very fast, developing applications and services for pervasive environment is becoming important. The other problem is therefore to propose a technique for image retrieval system in a pervasive environment.

The rest of this thesis is outlined as follows: chapter 2 describes related works on content and salient-object based image retrieval, image data model, image query algebra, image segmentation and pervasive computing systems. In chapter 3 assessments is made on different image segmentation techniques and the selected image segmentation algorithm and proposed image segmentation approaches are presented. Selected repository models and image query algebra are described in chapter 4 and in Chapter 5 a technique for accessing image under a pervasive environment is provided. The prototype developed to demonstrate the viability of the shape sensitive salient object based image retrieval is given in chapter 6. Finally conclusions will be given in chapter 7.

Chapter 2
Literature Review

2.1 Content-Based Image Retrieval

2.1.1 Content-Based Image Retrieval Systems

CBIR is an exciting field of research, but has delivered few operational systems capable of meeting user needs [2]. Fuhui Long et.al [1] stresses that image content may include both visual content and semantic content. Visual content can be very general or domain specific. General visual content includes color, texture, shape, spatial relationship etc while domain specific visual content is application dependent and may involve domain knowledge. The semantic content is obtained by either textual annotation or general visual content descriptors [1].

Content-based image retrieval uses visual contents of an image such as color, texture and spatial layout to represent and index the image [5, 2, 1]. Images, unlike alphanumeric text, are complex structures and exact matching will not be possible. Since exact matching is not possible in images, content-based image retrieval calculates the visual similarity between a query image and images in a database [2, 1]. Accordingly the retrieval result is not a single image, but a list of images ranked by their similarities with the query image. Many similarity measures have been developed for image retrieval based on numerical estimates of the distribution of features in recent years [1, 5]. The most common methods used for CBIR are: the k-NN (k-Nearest Neighbor) search and the Range Query Search. The k-NN search method searches for k closely similar images, where k is a positive integer. The Range Query Search method searches the images that are within a given radius ε in the feature space. This method returns an identified number of images (from zero to some number n) that are similar to the query image.

Many image retrieval systems, both commercial and of research nature have been built. Most image retrieval systems support one or more of the following options [4].

- Random browsing
- Search by example
- Search by sketch
- Search by text (including keyword or speech)
- Navigation with customized image categories

Many applications are available. The most well known commercial systems are, *QBIC* (IBM's Query By Image), *Virage* (VIR image engine from Virage Inc.) and *Excalibur*. Experimental systems, available as a demonstration on the web, include *Photobook* , *Chabot*, *VisualSeek*, *Mars*, *SurfImage* etc. These content-based image retrieval systems are discussed in detail in many literatures [5, 1, 6, 4, 2].

2.1.2 Image Retrieval in an object relational database environment

The relational database management is geared towards storage and retrieval of alphanumeric data. Part of the reason for the success and wide usage of the relational model is because it's ease of use and its good mathematical foundation for formulation of user queries. Database Management Systems such as oracle can be used to store multimedia data like images, video, audio etc. DBMS like oracle and Informix contain CBIR systems as an add-on for similarity-based matching.

Resent researches suggest that the relational DBMS is not designed for and is not able to cope up effectively with types of data such as image, video and audio as well as user defined types [5, 12]. The new model OODBMS could not also penetrate the market and is not widely accepted like the RDBMS. Object Relational (OR) model becomes important here because it allows organizations to continue using their existing relational systems, without having to make major changes, and allow them to start using the object oriented in parallel [5,12].

Modeling the techniques of CBIR in a way that image data management can be integrated into the existing popular database management systems for the purpose of efficiency is a preferable approach. Unfortunately there hasn't been much work on this regard. Solomon Atnafu [5] has addressed the issues regarding image retrieval in a database environment in his work titled 'Modeling and Processing of Complex Image Queries'. The research addresses issues like design and implementation of a data repository model used for storing images in an object-relational DBMS, development of a similarity-based algebra for image query and integration of this algebra with the existing relational operators and methods for similarity based Query optimization.

Image retrieval in a database environment requires a model for representing and storing of images and a mechanism for easy formulation of queries for image retrieval. These problems are well addressed in [5] and it lays a foundation which can be taken as a framework for further research on image retrieval in an object relational database environment. The work uses global feature of an image for similarity-based image retrieval within an object-relational DBMS (i.e. objects contained within an image are not treated in this work).

2.2 Salient-object based image retrieval

Humans tend to use high-level feature in every day life. However what current computer vision techniques can automatically extract from images are mostly low-level features [4]. In some literature assessment of the effectiveness of CBIR systems have been made [2, 4, 7, 8]. According to the assessments made it can be seen that CBIR can't successfully give good result in all cases and in some cases the resulting image could be totally different to the image query. According to this assessments CBIR give better result on retrievals based on primitive features such as color, texture and shape. In constrained application such as fingerprints and human face low-level feature based retrieval can give good results.

The performance of CBIR systems often remains quite modest especially on broad image domains [8]. In some case it may be interesting to query images in a database not based on their global features rather on objects contained by the images-called in literature as *salient objects*. In providing an effective and robust image database, objects in an image need to be extracted [9]. Salient-object based image retrieval is the theme of this thesis.

There are some researches done on image querying by considering salient objects of the image. A good example of such work could be the one done by Dawit Bulcha [6]. In this research salient objects of an image are extracted by manually fixing the Minimum Bounding Rectangle of an object. Then the salient objects are stored in relational database management system using an integrated Virage VIR search engine which is used for computing the similarity score between images. Some of the important features of this research are:

- Images and their salient object are modeled in an ORDBMS and metadata about the objects and their salient objects are also captured.
- It uses a similarity based algebra developed earlier in[5] for salient-object based image queries.
- Querying images are possible using the salient objects set within MBR.
- The user can also specify topological relations when querying the image database using the salient objects.

The research uses a MBR for approximating and modeling salient objects of an image in the database, which makes possible salient-object based retrieval, and considering topological or spatial relations when formulating queries against the image database.

The work made in [6] treats very well the modeling of salient-objects of an image into a database and performing query considering these salient-objects. The salient-objects contained in an image are usually irregular in shape. Such objects are approximated with some regular geometric objects in order to facilitate querying and establishing spatial relations between salient-objects. Several approximations exist in the literature to approximate such irregular shaped objects. These techniques include Minimum Bounding Rectangle (MBR), Minimum Bounding Circle (MBC), Minimum Bounding Ellipse (MBE) etc. Dawit bulcha in [6] uses MBR as an object approximation technique.

Although the work in [6] treats modeling of salient objects, it has the following limitations:

- It uses MBR for identifying salient objects. Since MBR doesn't consider the natural shape of salient-objects, it will have an impact on the precision of similarity-based image retrieval. This is due to the inclusions of background, which sometimes have no relation to the salient-object.

- Identification of the salient-objects of an image is done manually by the user. Such procedure could be tiresome, especially if there is a large collection of images and there are several salient objects within the image. These problem could be alleviated if it is possible to use automatic or semi-automatic procedures for identifying salient-objects

The other major work with regard to salient-object based retrieval is the DISIMA (Distributed Image Database Management Systems) project [10]. In this research salient-objects are classified into two: physical salient objects that store syntactic feature (color, texture and shape) and logical salient objects that give the semantics. The important features include:

- Image and other related data are stored and managed by an object DBMS.
- A high-level declarative query language allows user to query over image content and syntactic features using Multimedia Object Query Language (MOQL).
- The query engine returns a collection of related images.

The limitation of the DISIMA project is that like the work in [6] salient objects are identified by a user that uses some software to manually mark the MBR of the salient object. The difference with the work in [6] is that, DISIMA is a full-fledged image database management system.

2.3 Image Segmentation

In the analysis of objects in an image, it is essential to distinguish between objects of interest and the background. The objects of interest here are the salient objects contained within the image. The techniques that are used to identify the objects of interest (salient objects of the image in our case) are usually referred to as segmentation techniques.

Segmentation is an extremely important operation in several applications of image processing. The goal of the image segmentation process is to decompose an image to its sub parts that will be meaningful for some applications. The application of segmentation of images is used in a wide range of tasks, including image retrieval by its visual content, multimedia libraries, medical image analysis, object recognition, etc [11].

In medical applications, for instance, image segmentation can be used to identify areas of interest from Magnetic Resonance Images (MRI Imaging). Manual segmentation can be performed by specialists of the application domain, such as a radiologist in a medical image analysis while automatic segmentation requires software that uses image segmentation algorithms to decompose the image in to meaningful sub components.

In digital libraries large collections of images and videos need to be catalogued, ordered, and stored in order to efficiently browse and retrieve visual information. Color and texture are the two most important low-level attributes used for content-based retrieval of information in images and videos. Because of the complexity of the problem, segmentation with respect to both color and texture is often used for indexing and managing the data [14].

In wireless communication systems, which allow the transmission of speech, text and images, hand-held wireless devices are now available which may also display images with a limited resolution. In these applications we can deal with storage, processing and bandwidth limitations of the handheld devices by using image segmentation techniques.

Each segmentation technique is usually based on some mathematical model or theory and/or algorithmic approach. Most papers on segmentation stress that there is no universally applicable segmentation technique that will work for all images, and, no segmentation technique will work successfully on all image types [9, 13, 15, 14].

The desirable characteristics that good image segmentation should exhibit have been clearly stated by Haralick and Shapiro in [16] with reference to gray-level images. These characteristics are

- Regions of image segmentation should be uniform and homogeneous with respect to some characteristics such as texture and color.
- Adjacent regions of segmentation should have significantly disjoint values with respect to the characteristic on which they are uniform.
- Boundaries of each segment should be simple, not ragged, and must be spatially accurate.

A variety of image segmentation techniques like thresholding, clustering, split-and-merge algorithms, edge-detection algorithms, and physics-based segmentation [14, 13, 15] exist in the literature. A brief discussion of these techniques is in order.

Thresholding as described in [13, 15] is based upon a simple concept. A parameter θ called the *brightness threshold* is chosen and applied to the each pixel of the image a[m,n] as follows:

$$\begin{array}{ll} \text{If } a[m,n] \geq \theta & a[m,n] = object \\ \text{Else} & a[m,n] = background \end{array}$$

This way the objects and the background within an image are distinguished. The test condition could be based upon some other property than simple brightness. An alternative to this is to find those pixels that belong to the borders of the objects within an image. Techniques that are directed to this goal are termed edge finding techniques. But, to get the complete object a region-filling technique is needed since edge detection process identify only the pixels of the edges in the objects of interest [15].

In clustering, each pixel is mapped to a certain color space. The problem of clustering can be analytically stated as follows as described in [13]. Let us suppose that we have M patterns x_1, … ,x_M for m = 1, … , M within a certain pattern space S; in our case, the space S is a pre selected color space while the patterns x_m are the representations of the image pixels within S. The process of clustering consists in determining the regions S_1, …, S_K such that every x_m, m = 1, … , M, belongs to one of these regions and no x_m belongs to two regions at the same time.

Split–and-merge techniques start with an initial inhomogeneous partition of the image (usually the initial segment is the image itself) and they keep splitting until homogeneous partitions are obtained [14, 13]. There usually exist many small and fragmented regions after splitting. The merging phase accomplishes this task by associating neighboring regions and guaranteeing that homogeneity requirements are met until maximally connected segments can be produced [13].

In this thesis, an appropriate image segmentation system needs to be identified for the purpose of extracting the salient objects from the image.

2.4 Image Data Repository Models

A wide variety of digital image are becoming available in abundant forms nowadays. These images could be from different domain like medical systems, geographic information systems, education, art etc. Unlike text images are rich in content and they can be interpreted differently depending on the individual's perception. To capture such meaningful information from image and to conduct efficient processing on such collection of images, we need to design efficient data models.

An image data model is a set of concepts that can be used to describe the content of an image [5]. There are a variety of image data models, most of them depend on the application domain. Image data model is one of the main issues in the design and development of image database management system and the data model should be extensible and should have the expressive power to present the structure and contents of the image, their objects and the relationships among them [17].

The Image Data model proposed by R. Chbeir et. al [23] provides a mechanism for modeling both low-level feature and semantic features of an image. Such a model describes image related information very well and facilitates the construction of multi-criteria query that considers both the visual content and the metadata of an image. The model also considers relation between the salient objects. The model has two main spaces: the external space and the content space (Figure 2.1).

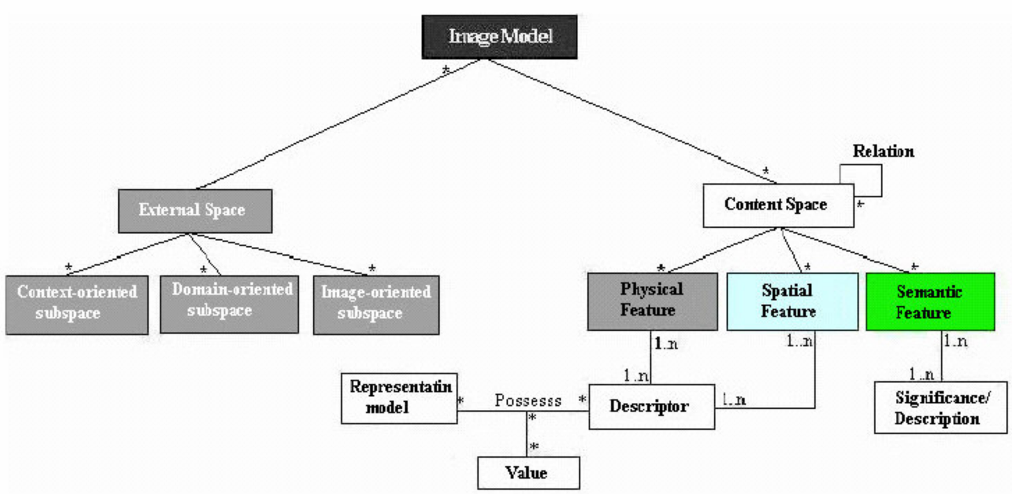


Figure 2.1: An image data model in UML by R. Chbeir et. al. [23]

The external space

The external space captures information that is external or not directly related to the content of the image. It consists of three sub spaces: the context-oriented subspace, the domain-oriented subspace and the image-oriented subspace. The context-oriented subspace contains application-oriented data that are completely independent of the image content and the domain-oriented subspace consists of data that are directly or indirectly related to the image. Information that is directly associated to the image creation, storage, and type are captured in the ***image-oriented subspace.***

The Content Space

The content space as its name suggests describes the content of the image: using both visual content and semantic content (with metadata). It consists of the physical, the spatial and the semantic features. *The Physical Feature* describes the image (or the salient object) using its low-level features such as color, texture, etc. several descriptors such as color distribution, histograms, dominant color, etc can be used to describe color feature.

The Spatial Feature is an intermediate (middle-level) feature that concerns geometric aspects of images (or salient objects) such as shape and position. *The Semantic Feature* integrates high-level descriptions of image (or salient-objects) with the use of an application domain oriented keywords.

The model proposed by R.Cheiber et.al [23] provides a generic view of an image and can be used in modeling images independent of application domains. Figure 2.2 below indicates the image model of R.Cheiber et.al [23] elaborating the placement of salient objects in the content space of the image as described in [6]. This presentation of the image model shows us that the content space of an image can be categorized into two sub-spaces as the features of the image as a whole (global features) and the features of each of the salient objects of interest.

Once we have a good image model, it is necessary to have a repository model that can effectively be used to capture the information related data components of the image into a database environment. Such a repository model describes the way image related data is capture in such a way that queries can be formulated and a query result be presented in an

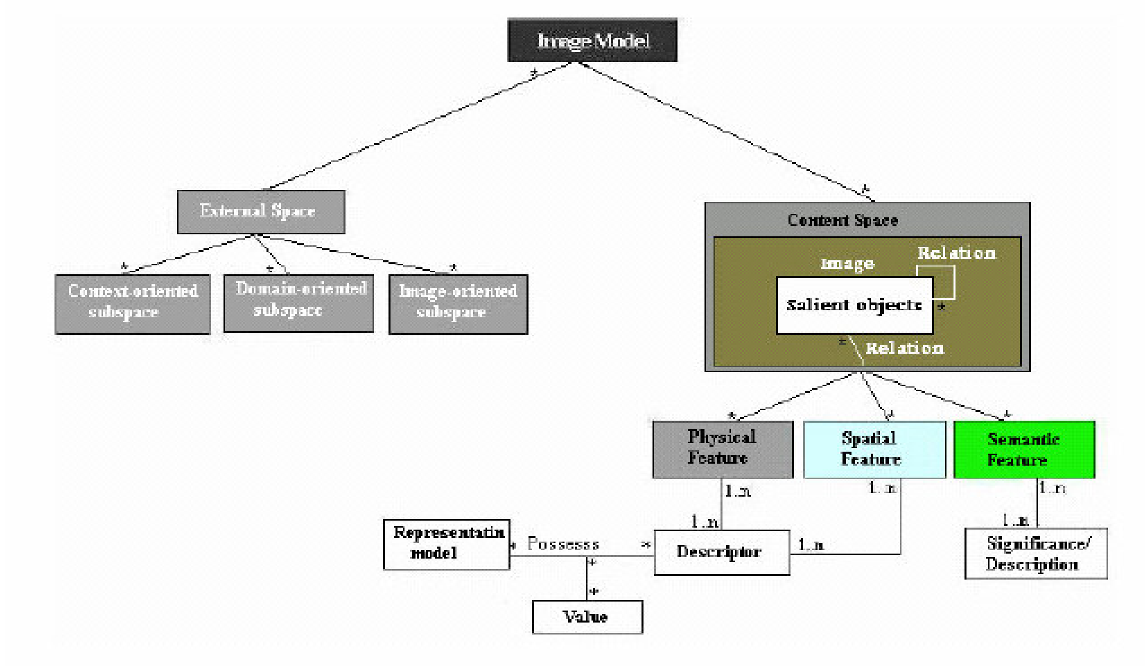


Figure 2.2: Elaboration of the placement of salient objects within the R. Cheiber data Model by Dawit Bulcha [6]

appropriate manner. In this regard, the image repository model presented in [5] provides a convenient method of storing data in an ORDBMS where multi-criteria queries can be formulated. The proposed model is described as follows:

An ***image data repository model*** is a schema of five components **M(id,O,F,A,P)** , under an object-relational model , where:

- **id** is a unique identifier of an instance of M,
- **O** is a reference to the image object itself that can be stored as a BLOB internally in the table or which can be referenced as an external BFILE (binary file),
- **F** is a feature vector representation of the object O,
- **A** is an attribute component that may be used to describe the object O using textual data or keyword like annotations.
- **P** is a data structure that is used to capture pointer links to instance of other image tables as a result of a binary operation.

This work also proposes a model that can capture salient objects of an image, which is important for this thesis. The *salient object repository model* is described as follows.

A ***salient object repository model*** is described as a schema of three components **$S(id_s, F_s, A_s)$**, where:

- **id_s** is an identifier of a salient object.
- **F_s** is the feature vector extracted to represent the low-level features of the salient object, and
- **A_s** is an attribute component that is used to capture all semantic descriptions of all the salient object using textual data or keyword like annotations.

2.5 Similarity-Based Image Query Algebra

One of the reasons for the wide usage of database over flat files is with databases required information can easily be obtained or queried. Standards like SQL have facilitated textual information retrieval from relational databases. The popularity and wide usage of relational database is because of their strong mathematical foundation and existence of a formal algebra for querying data stored within the database.

Unlike textual data Image retrieval from image databases is not based on exact matching rather it is based on similarity based matching. Similarity-based image retrieval from image database can only be facilitated if there is a formal algebra which can be used to formulate queries based on the visual features of an image. In this regard there are effors of formalizaing content-based image queries[5,27]. MOQL (Multimedia Object Query Language) is a text-based multimedia query language, which is an extension of the standard object query language(OQL) [27]. The query lanaguage is used in the DISIMA project illustrated in section 2.2 and permits feature-based queries in addition to image content semantics based search which is common in image databases. It is a fully object-oriented image query lanaguage and it allows DISIMA to support more sophisticated queries. This queyr language is based on fully object oriented database approach. But, it is not possible to use it in an object-relational databases with integrated image similarity matching engine by combining with existing relational query languages like SQL and PL-SQL.

A Major work with regard to similarity-based algebra is the work done in [5] for retrieval of image in an object-relational DBMS environment. This work has proposed five operators:

- The similarity based selection operators
- The similarity based join operator
- The symmetric similarity based join operator
- Operator associated with similarity based join (the Extract operator and the Mine operator).

In the sections that follow we describe the similarity-based selection and similarity-based join operator as discussed in [5].

2.5.1 Similarity-Based Selection Operators

The Similarity-Based Selection Operator

The similarity based selection operator is a unary operator on an image table M(id,O,F,A,P) performed on the component F as defined below.Given a query image ***o*** with its feature vector representation, an image table M(id , O, F, A,P) , and a positive real number ε; the similarity-based selection operator ,denoted by $\delta^{\varepsilon}{}_{o}(M)$ **,** selects all the instances of M whose image objects are similar to the query image ***o*** based on the range query method. Formally it is given as:

$$\delta^{\varepsilon}{}_{o}(M) = \{(id, o', f, a, p) \in M \,/\, o' \in R^{\varepsilon}(M, o)\}$$

Where $R^{\varepsilon}(M,o)$ denotes the range query with respect to ε for the query image o and the set of image in the image table M.

This operator is similar to the relational selection operator except that the operation is similarity-based (non-exact matching) and it operates only on one single component, *F*, of the image table. The result of a similarity-based selection operator is an image table. The similarity based selection operator can be combined with the relational operators on an image table [5]. The similarity based selection operator first uses the range query search method to select the image objects that are most similar to ***o*** from the objects in M (which is also expressed by the notation R^{ε} (M,o)). Then, it identifies instances of M whose image objects are similar to ***o***.

The similarity-based join operator

A similarity-based join is a binary operator on image tables performed on the feature vector components. To perform a similarity-based binary operation on two image tables, we assume that their feature vector components F_i are extracted identically in such a way that it permits a meaningful computation of range query.

Let $M_1(id_1,O_1,F_1,A_1,P_1)$ and $M_2(id_2,O_2,F_2,A_2,P_2)$ be two image tables and let ε be a positive real number. The similarity-based join operator on M_1 and M_2, denoted by $M_1 \otimes^{\varepsilon} M_2$, associates each object O_1 of M_1 to a set of similar object in M_2 with respect to the F components of M_1 and M_2. The resulting table consists of the referring instances of M_1 (the table at the left) where P is modified by inserting a pointer pointing to the id's of the associated instances of M_2 (the table at the right side of the operation) with its corresponding similarity score. More formally it is given as:

$$M_1 \otimes^{\varepsilon} M_2 = \{((id_1, o_1, f_1, a_1, p_1')/(id_1, o_1, f_1, a_1, p_1) \in M_1 \text{ and } p'_1 = p_1 \cup (M_2,,\{(id_2, o_1 - o_2)\} \text{ and } p'_1 \neq Null$$

- $(id_2, o_2, f_2, a_2, p'_1) \in \delta^{\varepsilon}{}_{o_1}(M_2)$ (i.e. instances of M_2 associated by the similarity-based selection $\delta^{\varepsilon}{}_{o_1}(M_2)$, and
- $\|o_2 - o_1\|$ is distance between o1 and o2 in the feature space, also called the similarity score of the o2 and o1.

Salient-Object-based Similarity Selection

Given the definition of the similarity-based selection operator and the range query discussed above, the work in [6] defines the Salient-Object-based similarity selection operator as follows: Given a query image ***o*** and its salient object o_s with its feature vector representation, an image table M (id, O, F, A, P), a salient Objects table S(id_s, F_s, A_s), and a positive real number ε ; a salient-object-based similarity selection operator $\delta^{\varepsilon}{}_{o_s}(M)$ selects all instances of M whose image objects have salient objects similar to the salient object o_s of the query image ***o*** based on range query method.

Formally,

$$\delta^{\varepsilon}{}_{o_s}(M) = \{(id, o', f, a, p) \in M \,/\, o' \in \prod_{M.o}(\delta_{M.id \in I}(M))\}$$

Where,

$$I = \prod_{S.A_s.id}(\delta_{o_s}(S)) \text{ and } (\delta_{o_s}(S) = \{(id_s, f'_s, a_s) \in S \,/\, f' \in R^{\varepsilon}(S, f_s)\}$$

$\mathbf{R}^{\varepsilon}(s,f_s)$ denotes the range query with respect to ε for the salient object O_s whose feature vector is f_s and set of salient objects in the table S. Here, the feature vector f_s represent the salient object. $\delta^{\varepsilon}{}_{o_s}$ is a similarity-based selection operator applied to the salient objects table.

2.5.2 Spatial Query Operators

In some queries involving salient objects, it might be necessary to consider the position of salient object within the main image. For example, in medical application a physician might be interested to retrieve brain images from the image database with a tumor at the top position of the brain image. In such kind of queries positional predicate like top, bottom, left, right etc has to be defined.

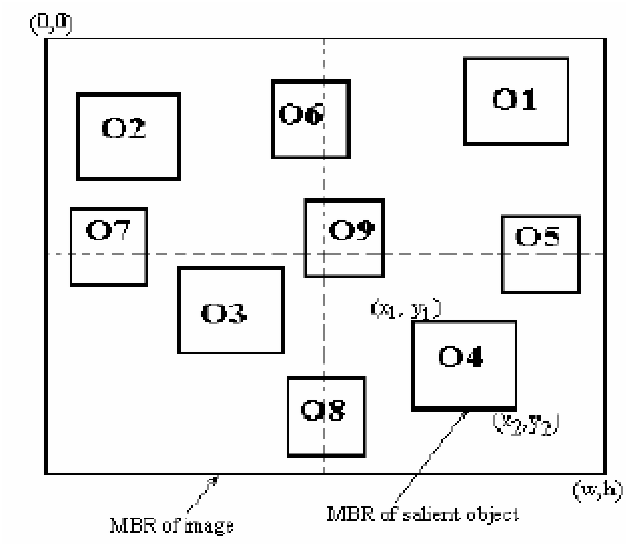


Figure 2.3 Salient objects positions within the main image

The work in [6] has proposed a scheme for describing the position of a salient object within the main image by partitioning the image into four parts of equal size as indicated in the Figure 2.3 above. In this work the salient objects contained within the image are approximated using the MBR. As shown in the Table 1 the position of the salient object within the main image is classified using nine positional descriptors.

Salient Object	**Positional description**
O1	*top right*
O2	*top left*
O3	*bottom left*
O4	*bottom right*
O5	*center right*
O6	*top center*
O7	*center left*
O8	*Bottom center*
O9	*center center*

Table 2.1: The nine positional description of a salient object within the main image

2.6 Image retrieval with pervasive computing devices

Pervasive computing systems has brought a new paradigm in computing through the use of hand-held and mobile computing device, high-speed networks and is expected to change the way we live, work and communicate. The availability of small and inexpensive hand-held devices has made computing easily available anywhere at any time for everyone. Regardless of the location wireless networks have enabled users to easily exchange information and communicate.

Computing is distributed through these different device, which can be from different vendors with different system and application software's installed on them and such heterogeneous systems are seamlessly connected [18]. Furthermore, complexity is reduced within the pervasive computing environment because the computing devices are easy to use as compared to personal computers.

2.6.1 The Pervasive Infrastructure

Pervasive computing uses web technology, mobile devices, and wireless communications. Millions of pervasive devices, public switched telephone networks (PSTN), gateways, servers and applications are connected and pervasive portals provide gateways to adapt the pervasive devices to the standard internet protocols in pervasive computing systems [18].

The devices include PDAs (Personal Digital Assistants), mobile and smart phones, notebook computers etc. Navigation key pads as in mobile phones, on-screen keyboard, handwritten recognition systems, speech recognition systems and Biometrics are used for human-machine interaction [18, 20]. Pervasive devices can't be used to their full potential unless they are connected to services and applications. Such device to device and device-to-server interaction are made possible through technologies like Wireless protocols (WAP, WML), mobile phone technologies (GSM, CDMA), Mobile database tools, Bluetooth, Mobile IP etc [18, 20]. The component of typical wireless networks is depicted in Figure 2.4 as discussed in [21].

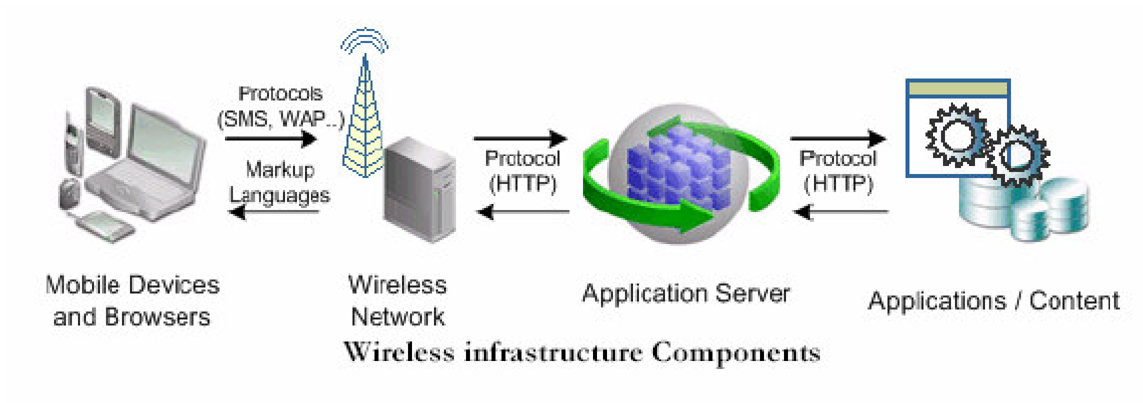


Figure 2.4: Components of Typical Wireless networks

2.6.2 Images Retrieval from Pervasive Devices

In pervasive computing systems mobile and stationary devices can be coordinated or synchronized to help users in accomplishing their tasks. For this vision to become a reality, developers must build applications that constantly adapt to these dynamic computing environments. Although the hardware and networking to realize this vision are available and increasingly becoming a reality only few applications run in this pervasive infrastructure [21].

Accessing image related data using pervasive devices is important form many application domain such as medicine. For instance, a physician or a medical doctor may want to access the medical image archive in the hospital database using a mobile device like PDA while being at home or while traveling. With the advent development of pervasive computing systems and with wide availability of mobile devices, it is an important issue of interest to design image retrieval system that will be accessible from these pervasive devices. A.E.Caroll et.al [19] discusses that the ability to link PDA to a central database allows unlimited potential in developing patient data management. The paper further stresses that although many stand-alone applications were designed on a PDA, none of them are designed to work in an integrated client/server environment. However, the author only considered textual data access. Accessing medical images with a PADA from a server poses additional constraints. Theses constrains are mainly due to the large size of images, the limited bandwidth of the communication medium , the smaller storage capacity of mobile devices and the unavailability of adequate image query techniques.

An implementation of image query system has been proposed and tested on the Nokia 9210 Communicator using Java-based client server architecture as discussed in [22]. The system can retrieve image from the image archive using QBE (Query By Example) approach. Different types of queries were made to the image database, namely, color histogram, shape and texture queries and the demo shows that implementation of image retrieval using pervasive devices is feasible.

To increase the usability of the image retrieval system it is necessary to incorporate designs in the image retrieval system that will make it accessible in a pervasive computing environment. In section 5, we will propose effective techniques to make image retrieval accessible in a pervasive environment.

Chapter 3
Identification of salient objects

Salient-object based image retrieval requires identification of objects of interest from the image and modeling them in the target image data repository environment for storage and later retrieval. Identifying regions of inertest (salient objects) from the image can be done either using manual extraction of the object of interest from the image or automatic image segmentation tools.

Manual segmentation can be performed by human's specialists of the domain application, such as a radiologist in a medical image domain while automatic segmentation requires software that uses image segmentation algorithms. The difficulty with manual segmentation is the problem of accurately locating the objects of interest. Besides manual segmentation can sometimes take too much time and can be a laborious if large amounts of images have to be processed. As discussed in [6], for better results automatic or semi-automatic segmentation of an image into perceptually meaningful regions is crucial in salient-object based image retrieval.

Image segmentation is partitioning of an image into regions that have similar characteristics such as color and texture. Segmentation results in semantically meaningful sub-regions that are perceptually close to segmentation by humans. It is an important technique for efficient image retrieval. Some researchers have developed algorithms for the elimination of the background so that only objects of interest are left out. This is important for most queries that are based on salient objects, since the background of the images is of no use [6]. This increases the efficiency query whose purpose is searching images containing a specific object of interest by avoiding irrelevant results that might be obtained due to the inadvertent similarity contribution of the background.

As discussed in section 1.2, this work considers shape of salient-objects and eliminates the distortion that can be caused by background. Thus, the salient objects are captured into the image repository for efficient retrieval purpose. This requires identifying or designing an efficient image segmentation algorithm that breaks down an image into its salient-objects.

3.1 Assessment of Publicly available Image Segmentation Systems

Many image segmentation algorithms were developed by researchers in the domain of computer vision. Thus, instead of trying to develop a new algorithm assessment of existing image segmentation algorithms has been made. The image segmentation systems were assessed using the criteria's:

- Supported Image Formats/types for both input and output image,
- Quality of the image segmentation results,
- Easiness of the resulting segmented image for further processing,
- Availability of the source code
- Programming language used to develop the system and
- Supported image types (gray scale , color image or both)

Most of the systems accept an input image, some segmentation parameters and outputs a segmented image with region boundaries superimposed. Some of the identified systems were edge detection techniques which mark the boundaries of the salient objects and hence requires further processing to extract the salient objects.

3.1.1 JSEG Image Segmentation Program

JSEG is an image segmentation program developed as a "console application" using visual C. It is compiled on SGI IRIX 6.3, Sun Solaris 2.4 and Windows 95 operating systems. The program is developed in the Department of Electrical and Computer Engineering, image processing and vision research labs, The Regents University of California. Experiments have shown good segmentation results on a variety of images. The supported image format includes YUV, RAW GRAY, RGB, PGM, JPG and GIF both for input and output images. It requires specification of parameters before running the program. The parameters include:

- Input media filename and Input media type (JPG, RGB, GIF etc).
- Output region map file name
- Image height and width (for some image types)
- Color quantization threshold (0-600)
- Region merge threshold (0-1.0)
- Other options: output image format, output region map file name etc.

More precise segmentation can be obtained by varying the segmentation parameters. It outputs two image files after the segmentation process is completed.

- A segmented image with the region boundaries superimposed ,and
- Output region map file

The output region map file is used to identify regions from the segmented image. This is very important for applications that need to process the segmented image.

The value of the color quantization has a great impact on the result of the segmentation, especially for color images. Figure 3.1 shows three types of input image and the resulting output image after the segmentation process using different threshold values.

With JSEG the user has more control, and can use different threshold value until satisfactory segmentation result has been obtained. The quality of the segmentation is good, besides the process of obtaining regions or salient objects from the image have been made simpler because the system also generates region map image file in addition to the segmented image with region boundaries superimposed. Source code is available for download at http://vision.ece.ucsb.edu/segmentation/jseg . The JSEG image segmentation program is described in detail in section 3.3

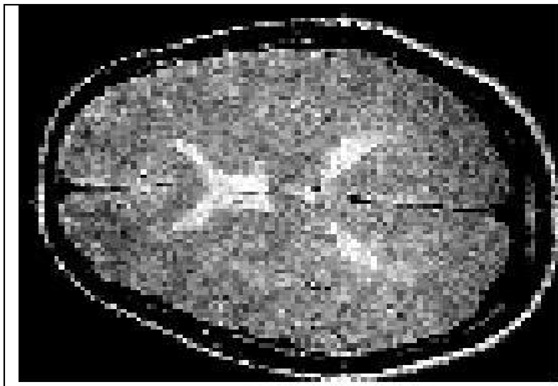
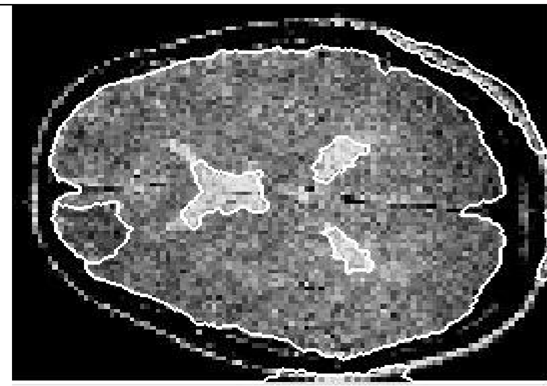
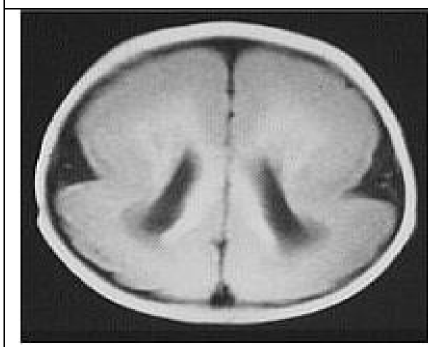
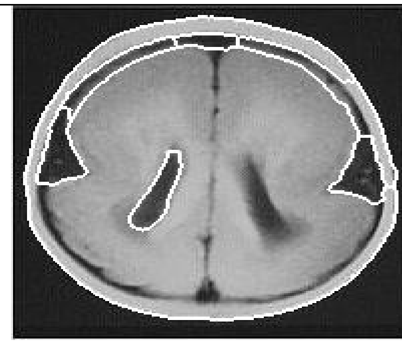
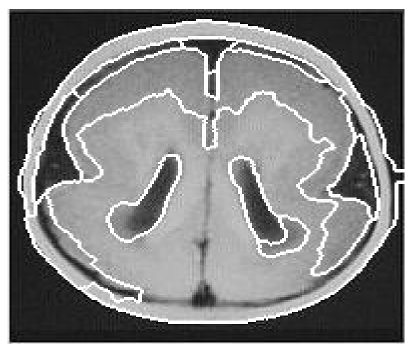
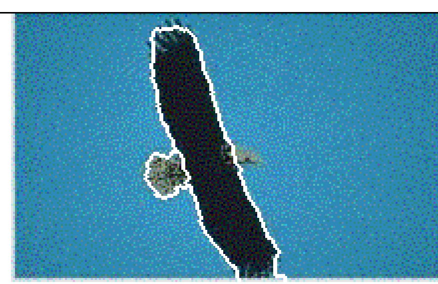
 Input image	 Segmented Image: Threshold 150
 Input Image	 Segmented Image: Threshold 250  Segmented Image: Threshold 150
 Input Image	 Segmented Image: Threshold 150

Figure 3.1: Segmentation result from the JSEG program.

3.1.2 Sobel edge detector from ImageJ

ImageJ is an image-processing program developed at the National Institute of Mental Health, Maryland. It's publicly available in both source code and executable program. It is developed for the java platform and can display, edit, analyze, process, save and print 8-bit, 16-bit and 32-bit images. It can also perform edge detection for image in 8-bit, 16-bit or 32 bit RGB formats. It uses a sobel edge detection algorithm for detecting the edges of regions of an image. This algorithms also has implementation in C++ and C. the sobel edge detection from ImageJ is implemented in java It accepts an image file and outputs a segmented image file with region boundaries superimposed.

The edge detection algorithm is implemented as a separate module in ImageJ and it supports only JPEG images. Edge detection module of ImageJ requires no parameter for detecting the edges of regions found within the image. Souce code is available for download at http://bij.isi.uu.nl.

Figure 3.2 show the results of sobel edge detector from ImageJ applied on the sample images. As can be observed from the image, the first two images are poorly segmented. Some of the contours are not closed which will make the process for obtaining salient objects from such segmented image difficult. Furthermore, since the program doesn't accept parameters, the user has no control on the edge detection process. The algorithm only detects the edges of object found within the image, hence it requires further processing to identifying the salient objects.

3.1.3 Color Image Segmentation Program by D.Comaniciu and P.Meer

Color Image Segmentation Program by D.Commaniciu and P.Meer is an implementation of an algorithm described in [31]. It is developed at the department of Electrical and Computer Engineering, Rutgers University, USA using the c++ programming language (Source code available at http://www.caip.rutgers.edu/~comanici/segm_images.html/).

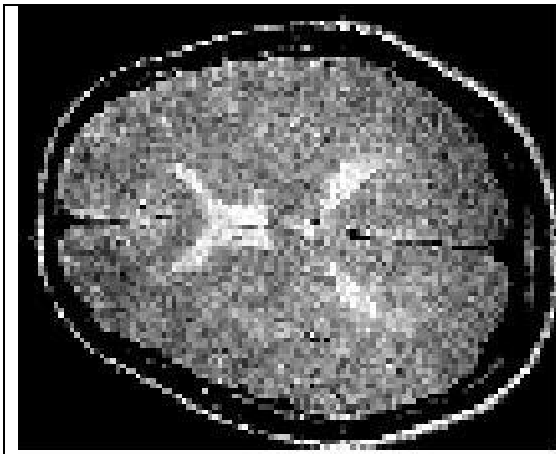
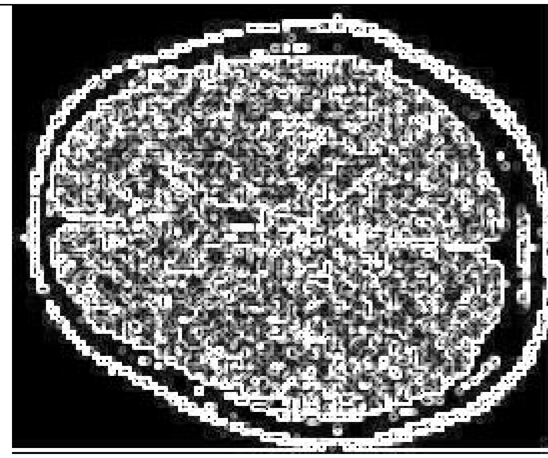
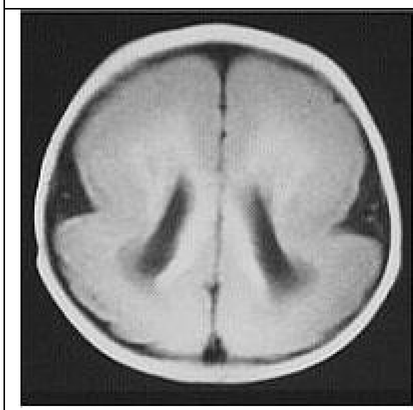
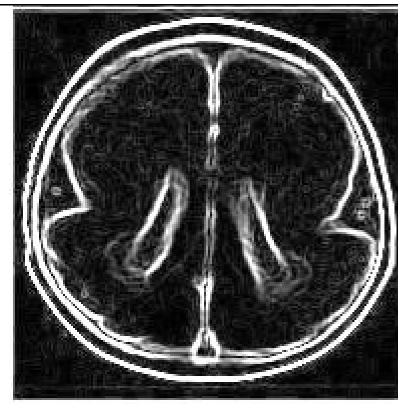
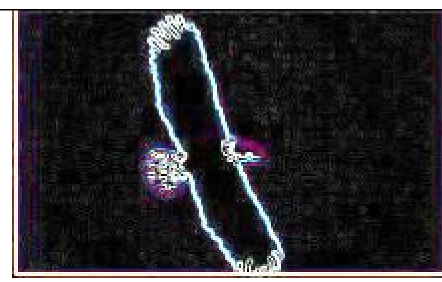
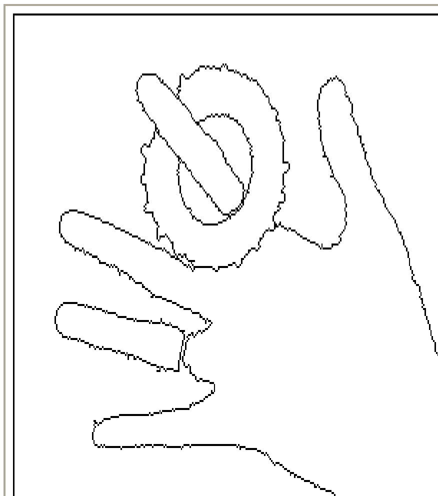
 Input image	 Segmented Image
 Input Image	 Segmented Image
 Input Image	 Segmented Image

Figure 3.2: Segmentation result from the sobel edge detector from ImageJ

The author has tested the functionality on IRIX Release 5.3 IP22 (Silicon Graphics), SunOS 5.5 Generic sun4u, and Digital UNIX V4.0 (Rev. 386) platforms. It supports PPM images only. The program outputs two images after processing the input image for segmentation.

These images are

- result.ppm - the segmented version of the input image in PPM format;
- result.pgm - the associated contour image in PGM format.

Original "hand"	Segmented	Contours
		

Source: http://www.caip.rutgers.edu/~comanici/segm_images.html

Figure 3.3: Segmentation result from the Color Image Segmentation Program by D.Comaniciu and P.Meer

Figure 3.3 above shows the segmentation result from the color image segmentation program by D.Comaniciu and P. Meer. This segmentation technique supports only PPM images as input,. Futheremore, obtaining salient objects from the detected contours requires further processing.

3.1.4 SkinSeg

SkinSeg is an image segmentation program for segmenting skin cancer images [24]. It supports large number of image formats such as RGB, PPM, JPG, GIF, PCX, TIF and BMP for both input and output. With this segmentation program it is possible to perform segmentation by indicating region of interest with a mouse. The segmentation result is not attractive when applied to the sample image used in this thesis. This could be because the skinseg program is designed to be used for segmentation of skin cancer images. The other problem with SkinSeg is that it available only as an executable program; source code is not available for the purpose of modification.

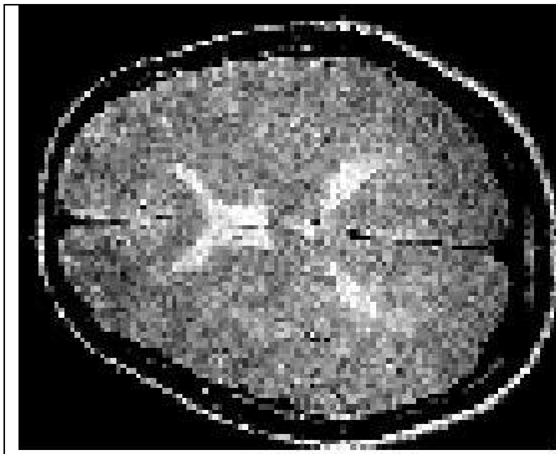
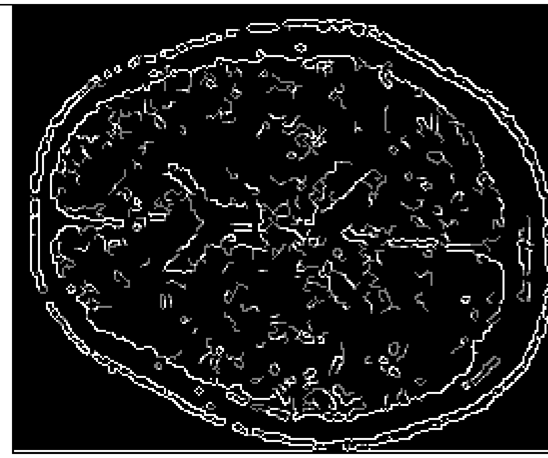
3.1.5 Canny edge detector by Di.Zhong

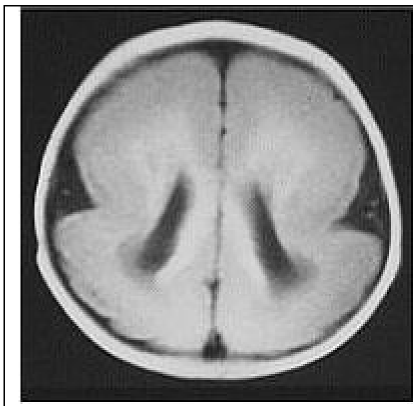
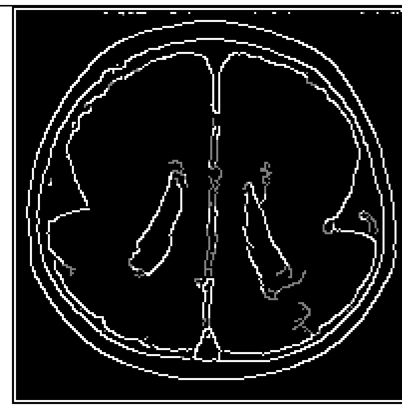
This is an implementation for the canny edge detection algorithms using the java programming language. It is developed by Di Zhong at the Department of Electrical and Computer Engineering, Columbia University, 1999. The Canny edge detectory by Di.Zhong supports GIF, JPEG, PPM and PGM image formats for input and PPM and PGM for output. After the segmentation is completed it displays a segmented image with the region boundaries superimposed. The program fails to segment image in JPEG format. This could probably be because of existence of a bug in the java code.

Figure 3.4 shows the results of the edge detection on the sample images used in this thesis. Segmentation result is similar to sobel edge detector discussed in section 3.2.2 above and some of the contours are not clo***sed.*** Bedsides, since it is an edge detection algorithm it requires further process to identify the salient objects from the resulting contour image .source code is available at
http://www.ctr.columbia.edu/~dzhong/JIM/docs/Package-image.html

3.1.6 EDISON (Edge detection and Image Segmentation) system

EDISON is an edge detection and image segmentation system implemented in c++ at Rutgers University. This version is the Java Implementation of “mean shift” edge detection algorithm [33] from Rutgers University. Its processing is limited to JPEG image format. It accepts JPEG image as input and outputs a PNG image with the region boundaries superimposed.

 Original image	 Segmented Image

 Input Image	 Segmented Image

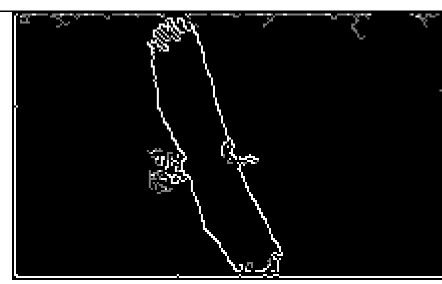
 Input Image	 Segmented Image

Figure 3.4: Segmentation result from the canny edge detection program by Di.Zhong.

Samples segmented images using EDISON are shown on Figure 3.5. Contours from the segmented images are not closed which will make identifying the regions a difficult process. Source code is available at http://www.fxpal.com/people/helfman/edge.html.

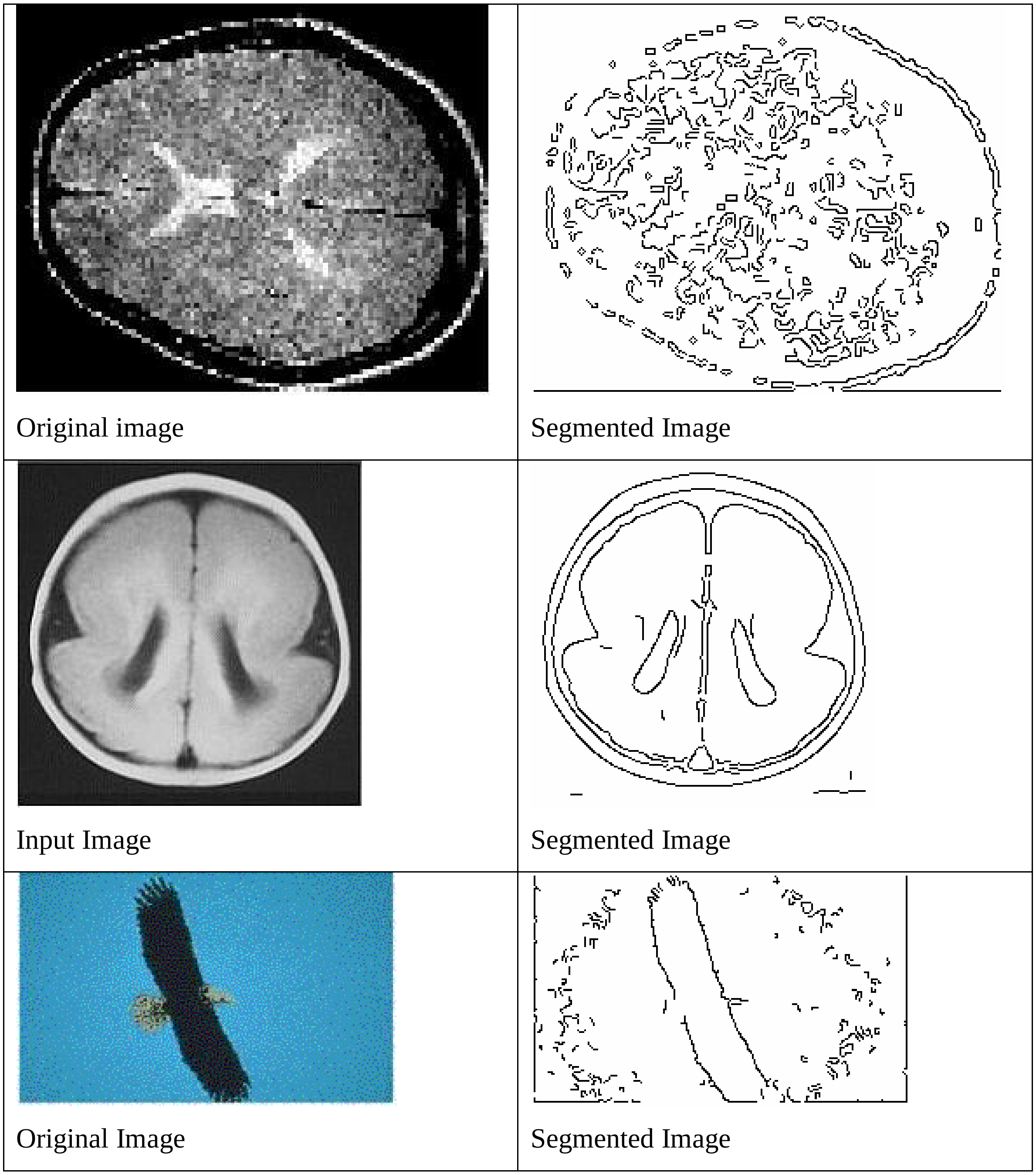


Figure 3.5: Segmentation result from the EDISON system.

3.2 Comparison of the Image segmentation programs

Summary of the assessed image segmentation programs is given in Table 3.1 based on four criteria's, the supported image formats both for input and output, the availability of the source code, the programming language used and the quality of the segmentation. The quality of the segmentation is subjective and evaluation is made on the results of the tests made using the sample images.

From the assessments made on the different segmentation system we have chosen JSEG because of the reason specified below.

- The result of the tests on sample images shows JSEG is better at segmenting medical or non-medical images and it give the user more control on the segmentation process.
- It supports large number of formats for both input and output and the algorithm is stable (no bugs identified).
- The source code is free and publicly available for research purpose.
- JSEG makes the process of identifying regions from the segmented image easier because it labels the pixel with region numbers.
- It supports both gray-level and color images.

The challenge in using JSEG with regard to this thesis is that the source code is written using the C programming language. Since the prototype of the image retrieval system is developed using java there need to be a means of solving the interoperability problem.

The interoperability problem can be solved using the JNI (Java Native Interface) which allows Java code that runs within a Java Virtual Machine (VM) to operate with applications and libraries written in other languages, such as C, C++, and assembly. So here the challenge will be identifying a C compiler that can used to compile JSEG source code and build a DLL (dynamic link library). Then the JNI can be used to generate a Wrapper class that can be used to interface with JSEG program from the java code.

No	Image Segmentation Program	Supported Image Formats	Source Code Available	Language	Quality
1	JSEG	Input/Output: YUV, RAW GRAY, RGB, PGM, JPG and GIF	Yes	C	Very Good
2	Sobel edge detector from ImageJ	Input: JPEG, PNG, PPM, PGM) Output: TIF, GIF, JPEG	Yes	Java	Fair
3	Color Image Segmentation by D. Comaniciu, P. Meer	PPM only both for input and output.	Yes	C++	Good
4	SkinSeg	Input/Output: RGB, PPM, JPG, GIF, PCX, TIF and BMP	No	-	Fair
5	Canny Edge detector by Di.Zhong	Input: GIF, JPEG, PPM, PGM Output: PPM and PGM	Yes	Java	Fair
6	EDISON edge detector	Input: JPEG Output: PNG	Yes	Java	Fair

Table 3.1: comparison of segmentation programs assessed.

3.3 JSEG Image Segmentation Program

Based from the assessment made from the section above, JSEG have been selected as an image segmentation program for the objective of this thesis. As a system of interest, brief description of the techniques implemented in JSEG is given below. Detailed information about JSEG can be found in the paper by Yinning Deng et.al [25].

Like many image segmentation algorithms, the JSEG algorithm assumes that the colors between two neighboring regions are distinguishable. Moreover, the JSEG approach assumes that:

- Each region in the image contains a uniformly distributed color-texture , and
- The color information in each image region can be represented by a few quantized colors which is true for most color images on natural scenes.

With JSEG first the colors are quantized to extract a few representing colors that can be used to differentiate neighboring regions in the images. Then, the quantized colors are assigned labels. The image pixels are replaced by their corresponding color class labels. Color class in the set of images quantized to the same color. This results in a class-map. A region growing and merging technique is finally used to get better segmentation results.

3.3.1 JSEG Parameters

The JSEG program requires specification of a number of parameters that can be used to control the results of the segmentation. In fact, this is why JSEG gives better result on a wide variety of images. The parameters includes the input image file and its type, the output image file, height and width of the source image, the type of the output image, color quantization threshold etc.

The JSEG algorithm has three basic parameters that can be used to by the user to fine tune the segmentation results. These parameters are optional but are necessary to get better segmentation results. The first one is a threshold for the color quantization process which determines the minimum distance between two quantized colors. The second one is the number of scales desired for the image. The last one is a threshold for region merging. These parameters are necessary because of the varying image characteristics in different applications. The JSEG image segmentation program parameters are describe in Table 3.2.

No	Parameter	Description
1	Input media filename	Source image filepath
2	Input media type	The type of the source image 1=imag yuv, 5=imgg ppm 2=imag raw rgb 6=image jpg 3=imag raw gray 9=image gif 4:imag pgm
3	Output image	Output image (region boundary superimposed) file name
4	Output type	Output image type 3=image raw gray 9=image raw gif
5	Height width	Source image height and width
6	Q thresh	color quantization threshold, 0-600, default automatic
7	r thresh	Region merge threshold, 0-1.0, default 0.4
8	Scale	number of scales, default automatic

Table 3.2: JSEG segmentation program parameters

The color quantization threshold takes values in the range of 0-600, but it can be left blank for automatic segmentation. The higher the value, the less number of quantized colors in the image. For color images, 250 can give better results. If results are unsatisfactory because two neighboring regions with similar colors are not getting separated, using small values like 150 can give better results.

The other optional parameter is the number of scales. The algorithm automatically determines the starting scale based on the image size and reduces the scale to refine the segmentation results. If one wants to segment a small object in a large-sized image, using more number of scales is necessary. To have a coarse segmentation, use a 1 as scale will do better.

The region merge threshold takes values in the range 0.0-0.7, but like color quantization threshold it can be left blank which defaults to 0.4. If there are two neighboring regions having identical color, trying smaller values can be used to avoid the merging.

3.3.2 JSEG Output images

The JSEG image segmentation program generates two images:

- Segmented image with region boundary superimposed.
- The region map file

The first output (image with region boundaries superimposed) shows or highlights the boundaries of the regions in the image. In this image the region boundaries are highlighted and can be easily identified. As can be observed in Figure 3.6 the regions are highlighted and extracting these salient objects needs further processing.

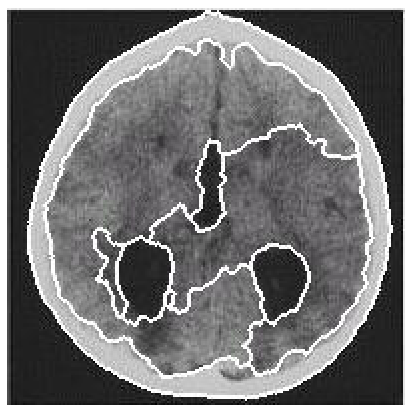


Figure 3.6: Segmented image with region boundary superimposed.

The second output is the region map file which is a gray-scale image. It labels the image pixels. If pixel (0,0) belongs to region 1, its value is 1. The label starts at 1 and ends at the total number of regions. Figure 3.7 below shows the region map file of the medical image shown in Figure 3.6. The region map file is dark image, as can be observed from Figure 3.7, when viewed using imaging software. The region map is very useful for applications that need to process the pixels to identify a particular region of interest.

Figure 3.7: Region Map File

The region map file is dark when viewed because most values are too small. Patches of regions can be seen if equalization is used .But the equalization changes the original pixel values.

3.4 Proposed Image segmentation techniques

3.4.1 Algorithms for salient object identification

The segmentation process will result in two images: segmented image with region boundaries superimposed and region map file which is very important to identify the objects contained in the image. One of the main reasons why JSEG is selected from the assessed segmentation programs is because JSEG results in the region map file after the segmentation process. The region map image file, which labels the image pixels, is very important to identify the salient objects. If pixel (0,0) belongs to region 1, its value is 1. The label starts at 1 and ends at the total number of regions. To identify shape sensitive salient objects once the segmentation process is completed, two algorithms described below, were developed.

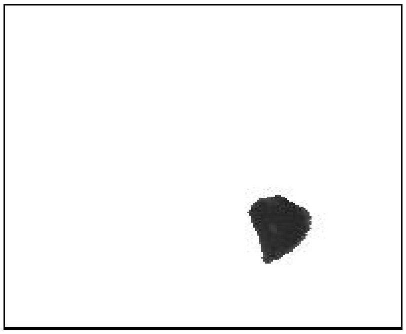
3.4.1.1 An algorithm that suppresses the background of source image

An algorithm is designed to extract the salient object by suppressing the background of the source image so that the region of interest will be left out. The algorithm uses both the source image and the region map file in order to identify the salient object. Assume source[m] is the source image pixel array, regionmap[m] is the region map pixel array, salient[m] is the new salient object image pixel array and let m be the size of the arrays. The following algorithm, shown in Figure 3.8, is used to suppress the background and identify the salient object.

```
For  i=0 to m-1
      If ( regionmap[i]==regionlablel_no)
             Salient[i]=source[i];
      Else
           Salient[i]=background;
      End if
 End For
```

Figure 3.8: Algorithm that suppresses the background of source image for salient object extraction

In Figure 3.8 above regionlabel_no is the pixel value of region to be extracted in the region map file and background refers to the pixel value of the background (white). So the algorithm set all the pixel values, except those that belong to the region, to the background color (white). After applying this algorithm, a new salient object image will be created with the background suppressed as shown in Figure 3.9 below.



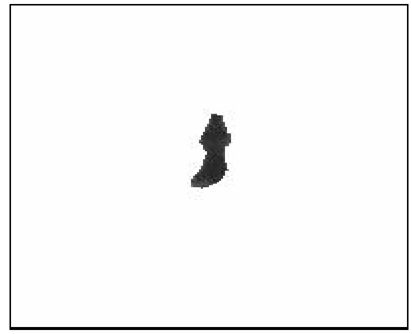


Figure 3.9: Identified Salient object images with their background suppressed.

3.4.1.2 MBR Identifier algorithm

Once the background of the salient object is suppressed, we need to extract the salient object image alone. The MBR of the salient object will be used to approximate the salient object image. The MBR Identifier algorithm finds two coordinate (x1, y1) and (x2, y2), shown in Fig 3.10, by inspecting the value of the pixels of the region map file.

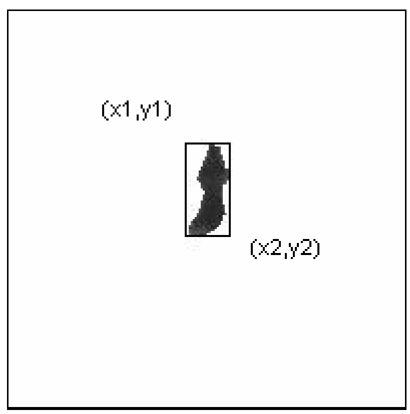


Figure 3.10: MBR coordinate of a salient object with its background suppressed.

The 2D image $a(x,y)$ is divided into *N rows* and *M columns*. The intersection of a row and a column is termed a *pixel*. The value assigned to the integer coordinates $[m,n]$ with $\{m=0,1,2,...,M-1\}$ and $\{n=0,1,2,...,N-1\}$ is $a[m,n]$.

The algorithm reads the value the two-dimensional pixel array of the region map image and determine whether the pixel corresponding to the background or region. If the pixel corresponds to the region, the inspection will stop and the coordinate will be recorded. This process is done from all sides (up, bottom,. Left and right) on the 2D pixel array to identify the two coordinate (x1,y1) and (x2,y2).

Assume regionmap [w, h] is a two dimensional pixel array of the region map file, w is the width of the image and h is the height of the image.

The algorithm in Figure 3.11 finds the x1 (minimum x value) starting from the top left position and reading each pixel horizontally from left to right until the region label pixel is found. When it finds the region label number, the x coordinate value will be recorded as the value of x1.

```
found=0
For i=0 to w-1
     For j=0 to h-1
              If (regionmap[i,j]==regionlabel_no)
                  x1=i
                  found=1
               End If
     End for
     If (found==1) exit loop
End For
```

Figure 3.11: Algorithm that finds the minimum X value

The algorithm shown in Figure3.12 below finds the y1 (minimum y value) starting from the top left position and reading each pixel vertically down until the region label pixel is found. When it finds the region label number, the y coordinate value will be recorded as the value of y1.

```
found=0
For j=0 to h-1
     For i=0 to w-1
              If  (regionmap [i,j]== regionlabel_no)
                  y1=j
                  found=1
              End If
     End for
     If (found==1) exit loop
End For
```

Figure 3.12: Algorithm that finds the minimum Y value

```
found=0
For i=w-1  to 0
     For j=h-1 to 0
              If (salient[i,j]== regionlabel_no)
                  x2=i
                  found=1
              End If
     End for
     If (found==1) exit loop
End For
```

Figure 3.13: Algorithm that finds the Maximum X value

The algorithm, shown in Figure 3.13 above, finds the x2 (maximum x value) starting from the bottom right position and reading each pixel horizontally from right to left until the region label pixel is found. When it finds the region label number, the x coordinate value will be recorded as the value of x1.

Finally, the algorithm shown in Figure 3.14, finds the y2 (maximum y value) starting from the right bottom position and reading each pixel vertically from bottom up until the region label pixel is found. When it finds the region label number, the y coordinate value will be recorded as the value of y2.

```
found=0
For j=h-1  to 0
     For i=w-1 to 0
              If (salient[i,j]== regionlabel_no)
                  y2=j
                  found=1
              End If
     End for
     If (found==1) exit loop
End For
```

Figure 3.14: Algorithm that find the Maximum Y value

The MBR Identifier algorithms will be used to automatically identify the MBR of the salient objects shown in Figure 3.15. The identified MBR coordinate of the salient object will be stored in the appropriate image table associated with the MBR of the containing image so that it will be possible to compute the location of the salient object within the containing image.

Figure 3.15: Salient object with their MBR identified which will be stored in the image database

3.4.2 Design of the Salient object identifier module

The salient object identifier module, which is responsible for segmenting an image into its salient objects, is sub component of the image retrieval system (EMIMS-3S). The high-level architecture of the image retrieval system developed is this work is shown in Figure 3.16. EMIMS-3S architecture is described in detailed in section 6.

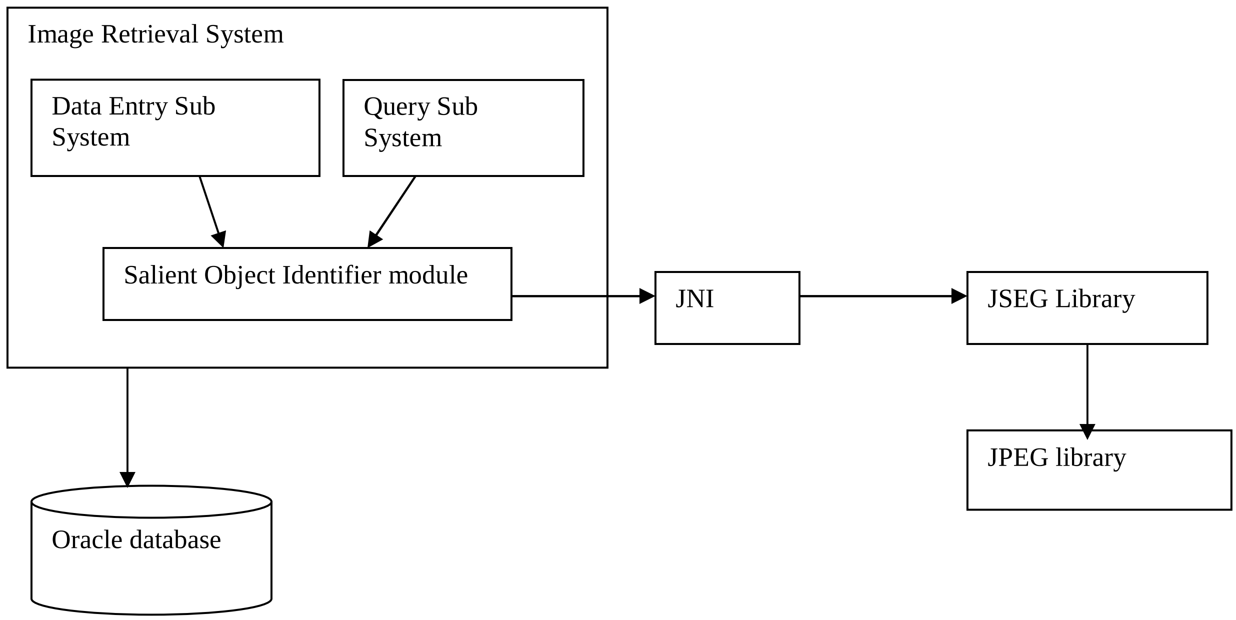


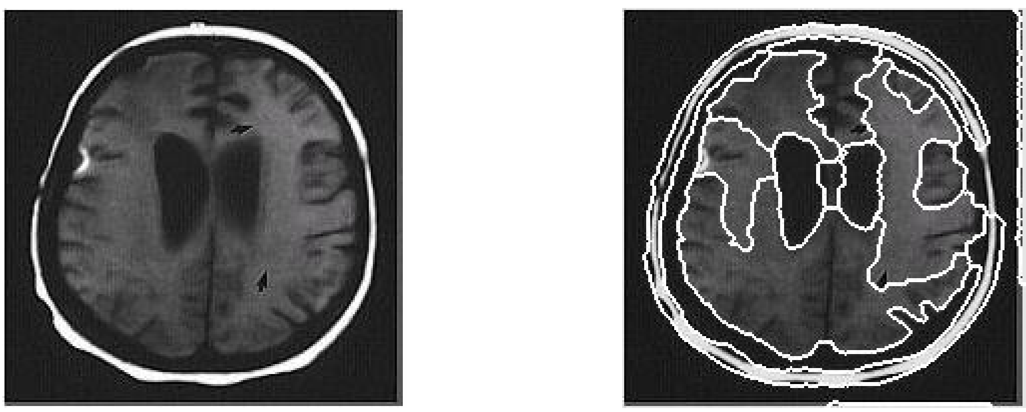
Figure 3.16: High-level architecture of the EMIMS-3S

As stated in section 3.2.1 the JSEG image segmentation program is implemented in ANSI C as a console application. To make the communication between EMIMS-3S and JSEG program, modifications are made on the JSEG program. The user interface of the JSEG is modified so that the program can accept segmentation parameters. JSEG requires another library, the JPEG image library, to work properly. The modified JSEG program and the JPEG image library are built into a DLL (Dynamic Link Library) using Visual C++ 6.0. Since EMIMS-3S is developed using java, the JNI (Java Native Interface) is used to interface the Salient object identifier module with the JSEG Library.

If the regions within an image are homogeneous, segmentation result may sometime be unsatisfactory. Taking this into consideration two techniques were proposed for segmenting an image into salient objects. These two segmentation approaches are *automatic segmentation and user-assisted segmentation.*

3.4.3 Automatic segmentation

In automatic image segmentation, an image will be segmented into its salient objects with little user interaction. The user will provide, if necessary, parameters to fine-tune the segmentation result. Automatic segmentation can be an efficient mechanism for identifying salient objects, especially if there are a large number of images to be stored in the database.



(a) Source Image (b) Segmented Image

Parameters: Color Quantization=150, Region Merge =0.4 and Scale=1

Figure 3.17: Automatic Image Segmentation

Figure 3.17 above shows the source image and the resulting image with region boundaries superimposed after segmentation. The user can change the parameter until satisfactory result is obtained. If the user is satisfied with the segmentation result, the developed technique will result in salient objects with their MBR's detected automatically as shown in Figure 3.18 below.

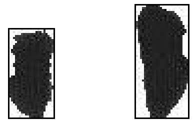
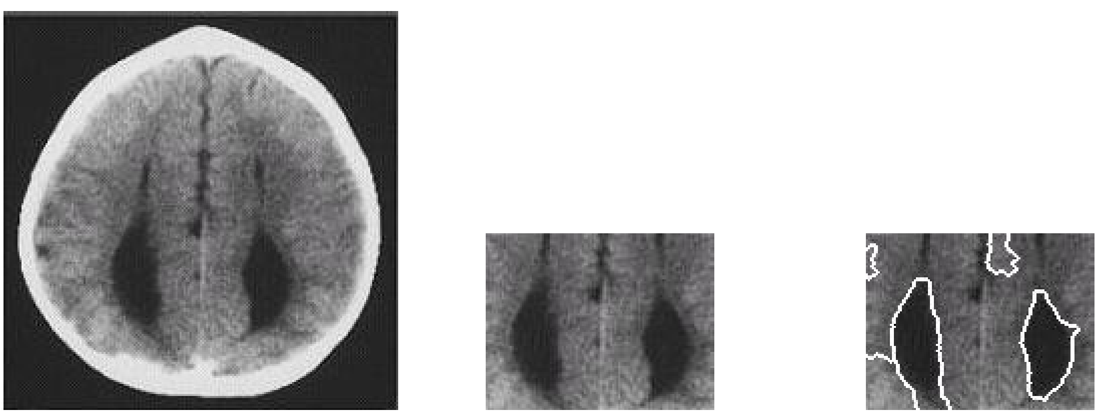


Figure 3.18: Identified Salient objects from Automatic Segmentation with automatically detected MBRs.

3.4.4 User-Assisted Segmentation

Sometimes the segmentation result can be better if the user provides sub region of the source image to the segmentation program. In this approach the user first selects sub region of an image before the segmentation starts. The salient object identifier module will consider only the selected sub region for segmentation and will result in salient objects only from the sub region selected. Figure 3.19 below shows whole image, the selected sub region of the whole image and the resulting image with region boundaries superimposed after sub region is segmented with User-Assisted segmentation.



(a) Whole Image (b) Sub Region Selected (c) Segmented Sub Region

Figure 3.19: An image, its sub region and segmented sub region in user-assisted segmentation.

Chapter 4

Image Data Repository Models and Image Query Algebra

Images are rich in content and in order to capture all the relevant information of an image an efficient data model is required. Storing relevant information about the image alone is not sufficient; the image data should be easily retrievable. Standards like SQL that enables users to easily retrieve and manipulate data within a relational database have made working with database easy. In image database, where image retrieval by visual content is required, similarity based operators are used for querying image data.

An ideal image DBMS that supports the management of salient objects of images would provide [28]:

- a data model to describe the image
- a data repository model to capture the necessary information related to the image and its salient objects
- a general technique to segment the image into meaningful units
- a query language that supports the necessary operators

In the following section, an extended image data repository model used to capture the necessary information about the image and its salient objects is described. An extension is also made on similarity-based algebra identified for querying the image database by visual content. This similarity-algebra will also be described in detail in the following section.

4.1 Image Data Repository Model supporting Salient-Objects

In this work both the main image and identified salient objects need to be stored in the object-relational DBMS. The data repository model proposed in [5] provides a method of storing both the salient object and the whole image in a database. As presented in section 2.4, an ***image data repository model*** is a schema of five components **M(id,O,F,A,P)** . In this work the A component of the image data reposition model will include $\mathbf{MBR_m}$, where $\mathbf{MBR_m}$ is the minimum-bounding rectangle of the image. Including this is in the data repository model was important for supporting multi-criteria queries based on the position of the salient objects.

A data repository model for storing salient objects and their associated information is also proposed in [5] as a schema of three components

$\mathbf{S\ (id_s, F_s, A_s)}$

The A component of this model is later extended in [6] to include a component that's used to used to describe the salient object (including the MBR of the salient object).

$\mathbf{A(MBR_{s}, \ldots)}$

where MBR_s is the minimum bounding rectangle of the salient object. The MBR in this model is necessary for supporting multi-criteria query that takes into consideration the position of the salient objects.

The salient object repository model presented in [6] does not include a reference to the salient image object. In order to support query that retrieves the salient object image, an extension is made on the salient object repository model presented in [6] to include a reference to the salient object that can be stored as a BLOB internally in the table or be referenced as an external BFILE (binary file). Hence the data repository model for storing salient objects and their associated information is extended and proposed as a schema of four components by incorporating the salient object O_s in the associated repository model.

The ***extended* salient *object repository model*** will have a schema of four components as described below

$\mathbf{S\ (id_s, O_s, F_s, A_s)}$

Where,

id_s is an identifier of a salient object.

O_s a reference to the salient object that can be stored as a BLOB internally in the table or be referenced as an external BFILE (binary file)

F_s is the feature vector extracted to represent the low-level features of the salient object, and

A_s is an attribute component that is used to capture semantic descriptions of the salient object using textual data or keyword like annotations. The A component also includes **MBR_s** , where **MBR_s** is the minimum bounding rectangle of the salient object.

The salient object repository model proposed in [6] does not capture a reference to the salient object. Hence, image queries using the salient object returns image which contains similar salient objects. But, some users may need to access or retrieve the salient object image alone, not the whole image that contains similar salient object. This could be particularly important for pervasive devices, which have limited display and storage capacity. Hence, including O_s and extending the salient object repository model was important in this work since it will increase the usability of the image database in pervasive computing environment.

4.2 Similarity Based Image Query Algebra with Salient-Object Support

Similarity-based image query algebra is developed based on the image data repository described above. Since this work is also based on the operators defined in this algebra, similarity–based selection operator and salient-object based similarity selection operators are presented below.

4.2.1 Similarity-Based Selection Operators

4.2.1.1 Similarity-based selection operator

As illustrated in [5], the similarity based selection operator is a unary operator on image table M (id, O, F, A, P) performed on the component F as defined below. Given a query image ***o*** with its feature vector representation, an image table M(id , O, F, A,P) , and a positive real number ε; the similarity-based selection operator ,denoted by $\delta^{\varepsilon}{}_{\mathrm{o}}(M)$, selects all the instances of M whose image objects are similar to the query image ***o*** based on the range query method. Formally it is given as:

$$\delta^{\varepsilon}{}_{\mathrm{o}}(M) = \{(id, o', f, a, p) \in M \,/\, o' \in R^{\varepsilon}(M, \mathrm{o})\}$$

Where $R^{\varepsilon}(M, o)$ denotes the range query with respect to ε for the query image ***o*** and the set of image in the image table M.

4.2.1.2 Salient-object based similarity selection operator

The Salient-object based similarity selection operator proposed [6] and described in Section 2.6 selects instances of M whose image objects have salient objects similar to the salient object o_s of the query image ***o*** based on range query method. It assumes that the whole image contains more detailed information rather that the salient object, hence returns the whole image. But sometimes it might be necessary to select instances S having similar salient object o_s of the query image o based on the range query method.

Pervasive computing devices have limited storage and processing capabilities and the wireless communication medium has also bandwidth limitations. Furthermore, the screen size of the mobile devices should be considered before downloading an image. In such situations, it may be adequate to download only the relevant salient object instead of whole images. In the previous section, we have extended the salient object repository model so that it will be possible to store reference to the salient object in the image database. To query the salient objects in the image database a salient–object based similarity selection operator is necessary.

In this work, two salient-object based similarity selection operators are defined: the first operator returns salient objects similar to the query image while the second operator return main images having similar salient object to the query image.

Salient-object operator that returns similar salient objects

This operator retrieves salient objects that are similar to the query image and is defined as follows. Given a query image ***o*** and its salient object o_s with its feature vector representation, a salient Objects table S (id_s, O_s ,F_s, A_s), and a positive real number ε; a salient-object operator that retrieves similar salient objects $\delta^{\varepsilon}{}_{o_s}(S)$ selects all instances of S that are similar to the salient object o_s of the query image ***o*** based on the range query method. Formally it is given as:

$$\delta^{\varepsilon}{}_{o_s}(s) = \{(id_s, o'_s, f'_s, a_s) \in S / o'_s \in R^{\varepsilon}(S, o_s)\}$$

Where $R^{\varepsilon}(S, o_s)$ denotes the range query with respect to ε for the query image o_s and the set of image in the image table S. The use of this operator can be described using an example query. For instance, the operator can be used to respond to queries like:

> *Find all salient object images in table S that are similar to the salient object Os of the query image O.*

Salient-object operator that returns images containing similar salient objects

This operator retrieves main image containing similar salient object to the query images and is defined as follows. Given a query image ***o*** and its salient object o_s with their feature vector representations, an image table M (id, O, F, A, P), a salient objects table

S (id_s, O_s, F_s, A_s), and a positive real number ε; a salient-object operator that retrieves images containing similar salient objects $\delta^{\varepsilon}{}_{o_s}(M)$ selects all instances of M whose image objects have salient objects similar to the salient object o_s of the query image ***o*** based on range query method. Formally,

$$\delta^{\varepsilon}{}_{o_s}(M) = \{(id, o', f, a, p) \in M \;/\; o' \in \prod_{M.o}(\delta_{M.id \in I}(M))\}$$

Where,

$I = \prod_{S.id}(\delta^{\varepsilon}{}_{o_s}(s))$ and

$\delta^{\varepsilon}{}_{o_s}(s) = \{(id_s, o'_s, f'_s, a_s) \in S \;/\; o'_s \in R^{\varepsilon}(S, o_s)\}$

Where $R^{\varepsilon}(S, o_s)$ denotes the range query with respect to ε for the query image o_s and the set of image in the image table S. The use of this operator can best be described using an example query. For instance, the operator can be used to respond to queries like:

> *Find all images in table M, that have similar salient object to the salient object Os of the query image O.*

4.2.2 Spatial Query Operators

A query that considers locations of salient objects is sometimes important. For instance, the image retrieval system may have to respond queries like

> *Find all images that have similar salient object to the salient object Os of the query image O at the top left position.*

The work in [6] has proposed a technique for identifying the position of the salient objects with respect to the main image. Assuming that {(0, 0), (w, h)} are the coordinate of the MBR of the main image and {(x1, y1), (x2,y2)} are the coordinates of the MBR of an arbitrary salient object within the image, the nine positions can be expressed mathematically as shown in Table 4-1.

Position description	Operator symbol	Mathematical description
Top right	top_right	$w/2 \leq x_1 \wedge y_2 \leq h/2$
Top left	top_left	$X_2 \leq w/2 \wedge y_2 \leq h/2$
Bottom left	bottom_left	$X_2 \leq w/2 \wedge y_1 \geq h/2$
Bottom right	bottom_right	$X_1 \geq w/2 \wedge y_1 \geq h/2$
Right	right	$X_1 \geq w/2 \wedge (y_1 < h/2 \wedge y_2 > h/2)$
Top	top	$Y_2 \leq h/2 \wedge (x_1 < w/2 \wedge x_2 > w/2)$
Left	left	$X_2 \leq w/2 \wedge (y_1 < h/2 \wedge y_2 > h/2)$
Bottom	bottom	$Y_1 \geq h/2 \wedge (x_1 < w/2 \wedge x_2 > w/2)$
Center	center	$(x_1 < w/2 \wedge x_2 > w/2) \wedge (y_1 < h/2 \wedge y_2 > h/2)$

Table 4.1: Implementation of salient object main image relations [6]

MBR of the salient object is marked manually by the users in [6]. In this work, identification of the MBR of the salient objects is done automatically, without user interaction using the MBR Identifier algorithms described in section 3.4.1. This is very important because it relieves the user from marking the MBR of the salient object manually, which is prone to error and time consuming.

In order to use these nine operators for formulation multi-criteria queries, both the MBR of the main image and the salient object has to be stored into the image database. The coordinates {(0, 0), (w, h)} can be used as the MBR of the main image, where w and h are the width and height of the image respectively. In addition the MBR of the salient object has to be identified like the main image so that comparison can be done to identify the location of the salient object. For responding to queries that considers position of salient objects, the spatial operators in Table 4.1 and defined in [6], which are implemented as function within the object-relational DBMS will be used for this work.

Chapter 5
Image Retrieval in a Pervasive Environment

Pervasive computing devices like PDAs and smart phones are designed to access data anywhere and anytime thanks to the advancement of technologies in hardware and software. Due to its mobile nature and its convenience, users are being more accustomed to using these devices. Hence, designing applications and services for pervasive devices is becoming an important issue.

Linking these devices to a central database server will enable users of pervasive devices to access and retrieve textual and image. Accessing data from database servers via pervasive device requires applications or programs that can synchronize the two computing environments. However, as discussed in section 2.6, only a few applications exist in this computing environment.

Lack of application or services that can be easily accessed in a pervasive computing environment could be attributed to many characteristics. The heterogeneity of pervasive devices, which ranges from handheld devices like PDAs and WAP enabled mobile phones to personal computers, and the bandwidth limitations are some of the reasons [18, 32]. These devices have different user interfaces; markup languages, and communication protocols.

The challenge here is to understand the requirements and use design methods that are best suited for those requirements. So, application developers and system designers should incorporate designs and techniques that make their services accessible across pervasive computing environment.

5.1 Hardware limitations

A major challenge in designing a system for pervasive access is to understand the characteristics of such devices and their hardware and software limitations [19]. Devices like PDAs and Smart Phone store their operating system codes and applications data in a non-volatile flash memory and battery backed random-access memory. The typical capacity of built-in memory in mobile device ranges from 2 to 16 MB [18]. These devices are also characterized by smaller screen sizes.

The newly introduced pervasive devices are having faster processors, larger storage capabilities but it is far behind compared to personal computers. The proposed image retrieval engine deals with images and thus consumes a large amount of memory. To be usable across these devices the hardware limitations must be taken into consideration.

5.2 Technique for image content and metadata retrieval

The image database designed in this work contains an image, its salient objects and metadata (both about the main image and its salient objects). Those devices which have limited processing, storage and display capabilities may retrieve only textual (metadata) about stored images or salient objects. We propose that a pervasive device can be made to retrieve a salient objects, since they require smaller storage and processing resource compared to the whole image.

Figure 5.1 below shows the main image which can be stored in the associated table in the image database. This image contains salient objects which can be seen as dark objects in the bottom and central position of the image. The image is segmented and the identified salient objects can easily be observed in the image with region boundaries superimposed in Figure 5.1- (b).

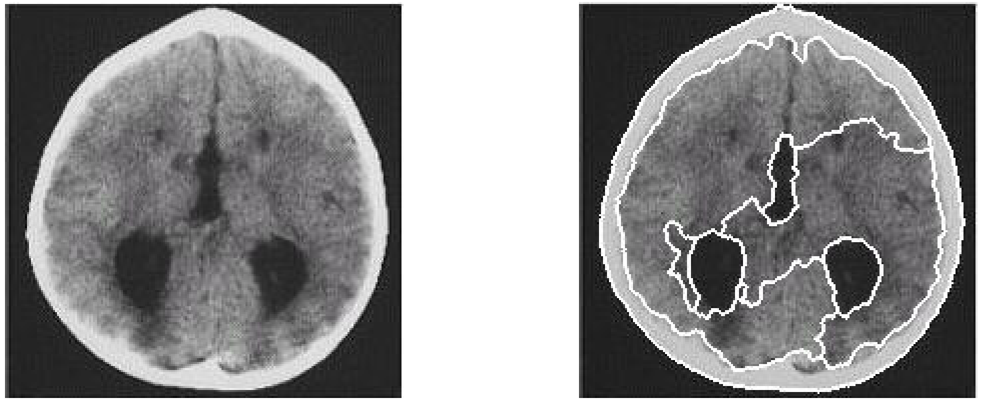


Figure 5.1: Medical Image and it associated segmented image with region boundaries superimposed

The proposed image segmentation module of the image retrieval system will result in salient objects which can be observed in Figure 5.2 below.

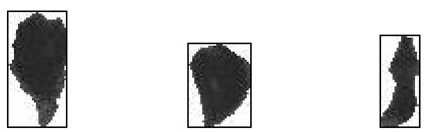


Figure 5.2: The three Salient objects identified from the medical Image in Figure 5.1.

The size of the salient objects is smaller in size compared to the main image and can easily be displayed in mobile computing devices. A user may retrieve the whole image which contains much more information, store and analyze the object contained on his personal computer. But, this may not be feasible for a user with a PDA or other handheld computing device. A user from such devices may retrieve metadata about the main image or the salient object contained within the main image or may retrieve the salient object only. These salient objects are smaller in size and need a limited screen area to be displayed and analyzed and also do not consume bandwidth when transmitted over the network infrastructure.

The image retrieval system designed in this thesis will enable the user to retrieve:

- The entire image (main image)
- The salient object contained in the main image
- Metadata about the main image or contained salient object.

Depending on the users computing devices capabilities, this image retrieval system can be made give its users an option for retrieving different information about the image. This will increase the accessibility of the image database in a pervasive computing environment.

However, if the query result returns too many images or salient objects then it may not be possible to store and process all these images or salient objects on the mobile device. An appropriate technique can be proposed to deal with such types of problems. For example, it is possible to design a query interface that will return results step by step in their order of similarity. Because this problem is being dealt with another project, we didn't go to the details of designing the mobile client interface other than designing the requirement on the server side.

5.3 Providing the Image Content to End-User

Generally, two models can be used for providing application and content to end-user devices: applet and servlet models [20]. The applet model executes the application on the client devices. A disadvantage of this model is that it needs applications to be deployed and administered in the client devices. With the applet model the user can work offline and periodically synchronize the local data with remote server database.

But the servlet model processes the entire application on the server side. Servlets are Java programs that extend the capabilities of the server [21]. The client sends the query to the servlet; which processes the request and passes it to the appropriate query handler. The client queries and displays the results using a browser. Unlike the applet model, this model requires online connection.

The image retrieval system can be made accessible using internet portals. Portals aggregate information from diverse sources to make it available through the web interface [20]. Providing architecture for accessing the image database is beyond the scope of this work but there are architectures discussed in the literature for providing such content is illustrated in [18, 20, 21].

Chapter 6

EMIMS-3S (Extended Medical Image Management System that considers Shape Sensitive Salient Objects)

To show the validity of the proposals in this thesis, a prototype named EMIMS-3S is developed. EMIMS-3S (Extended Medical Image Management System that considers Shape Sensitive Salient Objects) is an extension of EMIMS-S [6]. EMIMS-S (Extended Medical Image Management System with salient object support) is presented in [6] as a prototype to demonstrate salient object-based image data modeling and retrieval. EMIMS-3S demonstrates image data management in an object-relational DBMS by considering the natural shape of the salient objects.

EMIMS-3S is implemented under oracle 9i with an application to medical images. Medical image obtained from American College of Radiology are used for testing the functionality of the EMIMS-3S. Content based image retrieval is a meaningful and important means of accessing image in medicine. For Instance, a clinician could retrieve image of similar cases from an image archive for consultations before making diagnosis.

EMIMS-3S is designed to run either as an applet accessible on the Web or as a standalone application. J2SE (Java 2 Platform, Standard Edition, v 1.4.2) with JBuilder8 is used to develop the query interface, the data entry interface and the required libraries for image segmentation. Oracle 9i enterprise edition is used for managing the image data at the server side.

6.1 Structure of EMIMS-3S

EMIMS-3S is developed on top of the Oracle 9i DBMS. Figure 6.1 below presents a general architecture of EMIMS-3S under an existing object-relational DBMS based on general architecture presented in [5]. EMIMS-3S interacts with an Oracle 9i database through JDBC. It has three main components: the *visual interface* and the *query manager* and *salient object identifier module*. The visual interface includes the *data-entry* and the *query* interfaces.

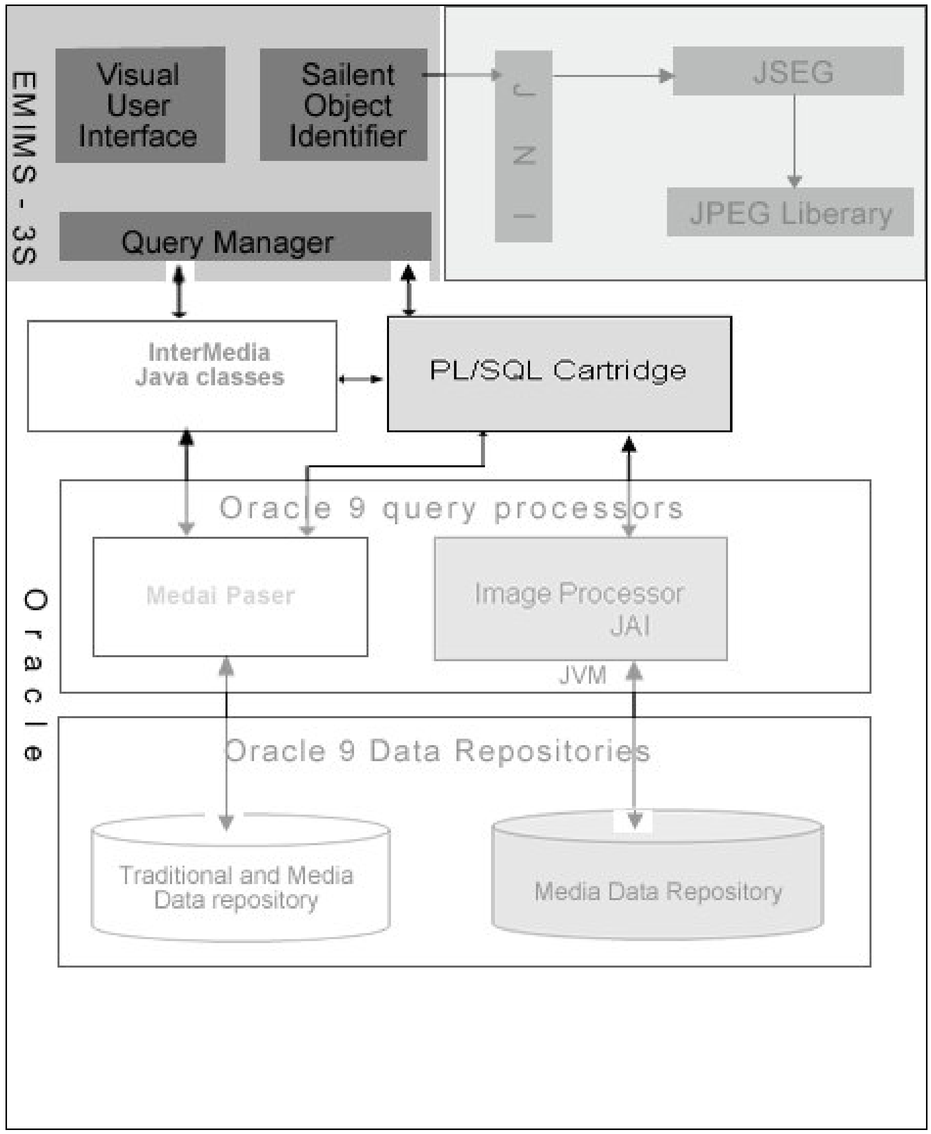


Figure 6.1: General Architecture of EMIMS-3S under a DBMS

Oracle 9i stores rich content in tables along with traditional data through *inter*Media. The *inter*Media Java classes make it possible for JDBC result sets to include both traditional relational data and *inter*Media media objects [26]. A server-side media parser and image processor is supported through the Oracle 9i Java Virtual Machine (JVM). The media parser has object-oriented and relational interfaces, supports format and application metadata parsing. The image processor includes Java Advanced Imaging (JAI) classes and provides image-processing tools for converting, matching, and indexing images [5]. The salient object identifier module uses the Java Native Interface (JNI) to communicate with the JSEG image segmentation program. Since EMIMS-3S is an extension of EMIMS-S, some of the classes are migrated from the prototypes developed in [5] and [6]. The detailed structure of EMIMS-3S is depicted in Figure 6.2.

6.1.1 The User Interfaces of EMIMS-3S

EMIMS-3S has mainly two interfaces: **Data Entry Interface** and **Query Interface**. Data Entry is extended from the implementation in [6] to support identification of salient objects. This interface will enable the user to extract salient object from an image and store them in an oracle database. The Query Interface is also extended to support identification of salient objects and use of these salient objects for similarity-based retrieval of images. Screen shots for the user interfaces of EMIMS-3S are presented in detain in section 6.3.

6.1.2 The classes of the EMIMS-3S Modules

The Connection class

The connection class is migrated from EMIMS [5]. It establishes client connection to the Oracle database using JDBC data access library.

The QueryManager class

The query manager class implements the similarity-based selection operator, the join operator, and others described in [5]. These include: the similarity join (SimJoin), the query by example (QBE), Insert, Mine and other operators.

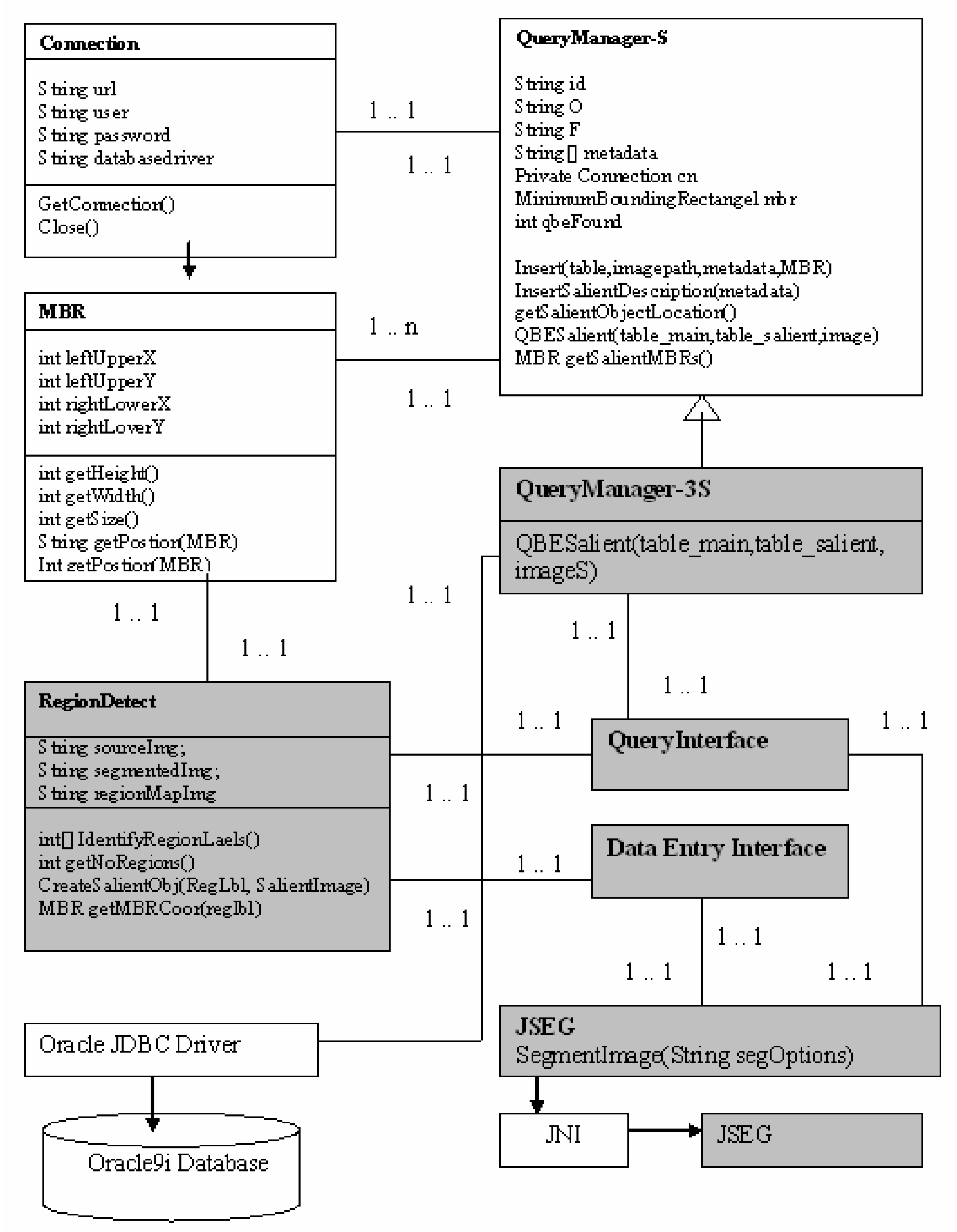


Figure 6.2: Structure of EMIMS-3S

The QueryManager-S class

The QueryManager-s migrated from [6], is an extension of QueryManger class implemented in [5]. The overloaded Insert method supports insertion of main image together with its MBR and insertion of metadata of the salient object. The **QBESalient** method implements the salient-object-based similarity selection. It takes the salient object as input and retrieves the images which have similar salient objects based on a given threshold. It also takes the position of the salient as an additional optional parameter and performs retrieval considering the position.

The QueryManager-3S class

The QueryManger-3S QBESalient method can make similarity based selection using the salient object. But unlike the implementation in [6] it returns the salient objects instead of the whole image.

The MBR Class

The MBR is migrated from [6] and implements the minimum bounding rectangle entity required both for the main image and the salient objects to process MBR related functionalities. It provides method to access the height, width, and size of the MBR respectively. The Method *getPosition* returns the position of an arbitrary MBR of the salient object with reference to the main image. The result will be one of the nine positions: *top*, *bottom*, *left*, *right*, *top right*, *top left*, *bottom right*, *bottom left*, and *center*.

The JSEG Class

The JSEG class is a wrapper of the JSEG image segmentation library. It implements the proposed segmentation techniques. It uses the Java Native Interface to communicate with the JSEG dynamic link library implemented in C. The SegmentImage method has one argument named segOptions. The method accept list of arguments and options discussed in section 3.4 as string with the arguments and options separated using tabs and spaces and segments the image.

The RegionDetect class

The segmentation process results a segmented image with region boundaries superimposed and a region map image. The RegionDetect class processes these two images to automatically identify the salient objects. RegionDetect class has the following methods:

- **Int[] IdentifyRegionLabels():** this method identifies all the region labels from the region map file and store them in an array.
- **getNoRegions():** Returns the total number of regions found in the source image after segmentation operation.
- **CreateSalientObject(int regionLabel,String SalientImgPath):** creates a salient object image with background suppressed . Accepts the region label and the salient image path.
- **MBR getMBRCoord(int regLabel):** accepts the region label and return the MBR coordinates of the salient object.

6.2 The sample database

EMIMS-3S uses the database is implemented in [5] as a prototype for whole image retrieval and later extended by [6] for salient-object based retrieval. In this thesis, the same database is used for demonstrating shape-sensitive salient-object based retrieval. The sample database allows the storage of both the feature vector and spatial information of the main image and the constituent salient objects. Metadata information related to the whole image and its constituent salient-objects is also captured in the database.

EMIMS-3S tables

The sample database is designed based on the model by R.Cheiber et.al [23] and contains data both about the content of the image and external to the image. Database tables and their fields enclosed in braces are specified below. The primary key fields are underlined.

DOCTOR (*DSN*, Name, Specialization, P_History) : Basic Information on the medical Doctor.

HOSPITAL (H_CODE, NAME, ADDRESS, SECTIONS): Basic information on the hospital.

MED_EXAM (SSN, DSN, H_Code, ME_Code, DateOfExam C_Presenation, Case, M_History, Findings, Diagnosis, M_Image) : Detail Information on patient medical examination.

M (ID, O, F, Rect, ME_Code, Image_Path, P): main image table. ID is the unique identifier.

S (ID, O, F): Salient objects table, stores each salient object and its feature vector. ID is the unique identifier.

S_A (SALIENT_ID, IDMAIN, Rect, Anomaly_Type, Case, Diagnosis, Findings, Remark) : Metadata description of the salient objects. This table stores semantic textual description of the salient objects and their MBRs.

PATIENT (SSN, Name, DateOfBirth, R_Address, R_History, M_History) : Basic patient information, Uniquely identified by patient social security number (SSN)

6.3 EMIMS-3S User interface Screen Shots

The user interface of EMIMS-3S consist of

- Whole image data entry interface migrated from [5] ,
- Interface for identification of salient-object using automatic or user assisted image segmentation,
- Salient object data entry interface,
- An extended Query By Example interface either whole image and salient-object based image queries.

6.3.1 Main Image Data Entry Interface

The EMIMS-3S main data entry interface is migrated from EMIMS developed in [5] and is used to insert main image and it’s MBR into the associated image table of the oracle database. The user also has the option to specify metadata about the image that's going to be inserted in to the associated table in the oracle database.

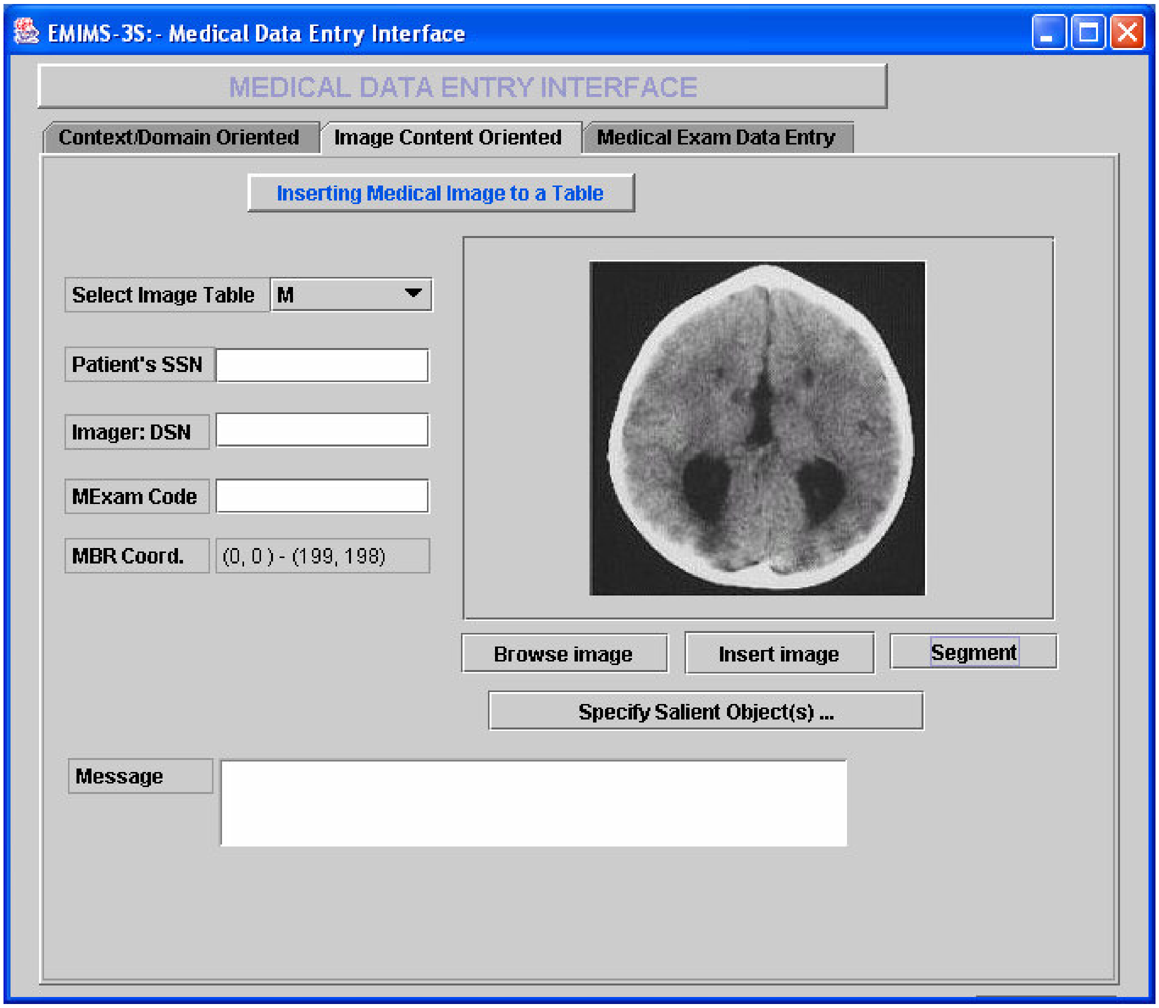


Figure 6.3 EMIMS-3S: Medical Data Entry Interface

The medical data entry interface is shown in Figure 6.3. Once the main image is inserted, the user will have the option to insert the salient objects by drawing MBR (implementation of [6]) or using Salient Object Identification module developed in this work. The **Segment** button is added to this interface to enable the user to segment an image into its salient objects once the main image is inserted into the associated table in the database.

6.3.2 Salient-Object Identification and Insertion Interface

The salient-object specification interface shown in Figure 6.4 below is used to implement automatic and user-assisted segmentation proposed in section 3.4. It's used to identify salient objects from an image. This interface is displayed when the user clicks on **Segment** button from the main medical image entry interface shown in Figure 6.3.

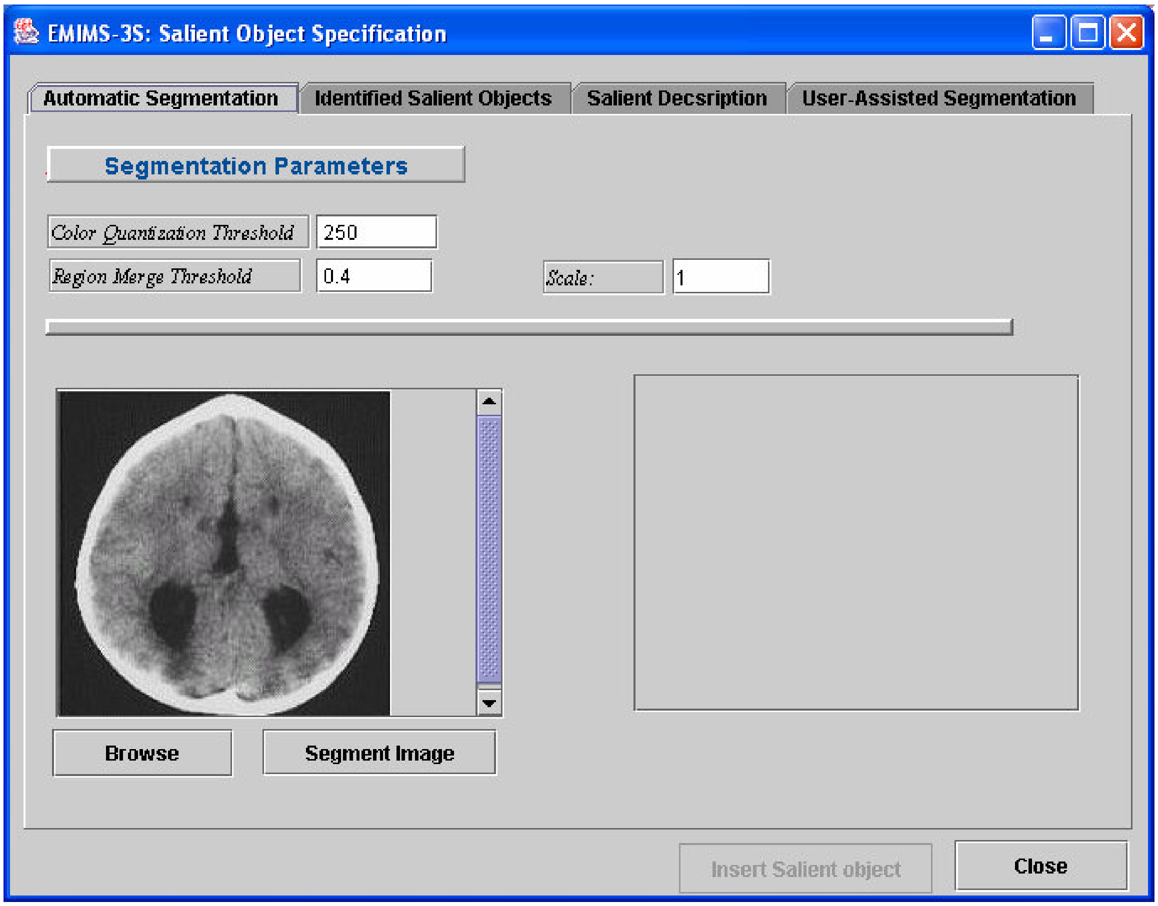


Figure 6.4: Salient-Object specification interface

The **Automatic Segmentation** tab will be active by default. To perform automatic segmentation, the user has to specify the required parameters discussed in Section 3.4 and click on the **Segment** button. This display an image with region boundaries superimposed. Figure 6.5 below shows the source image (at the left side) and the segmented image (at the right side) with region boundaries superimposed.

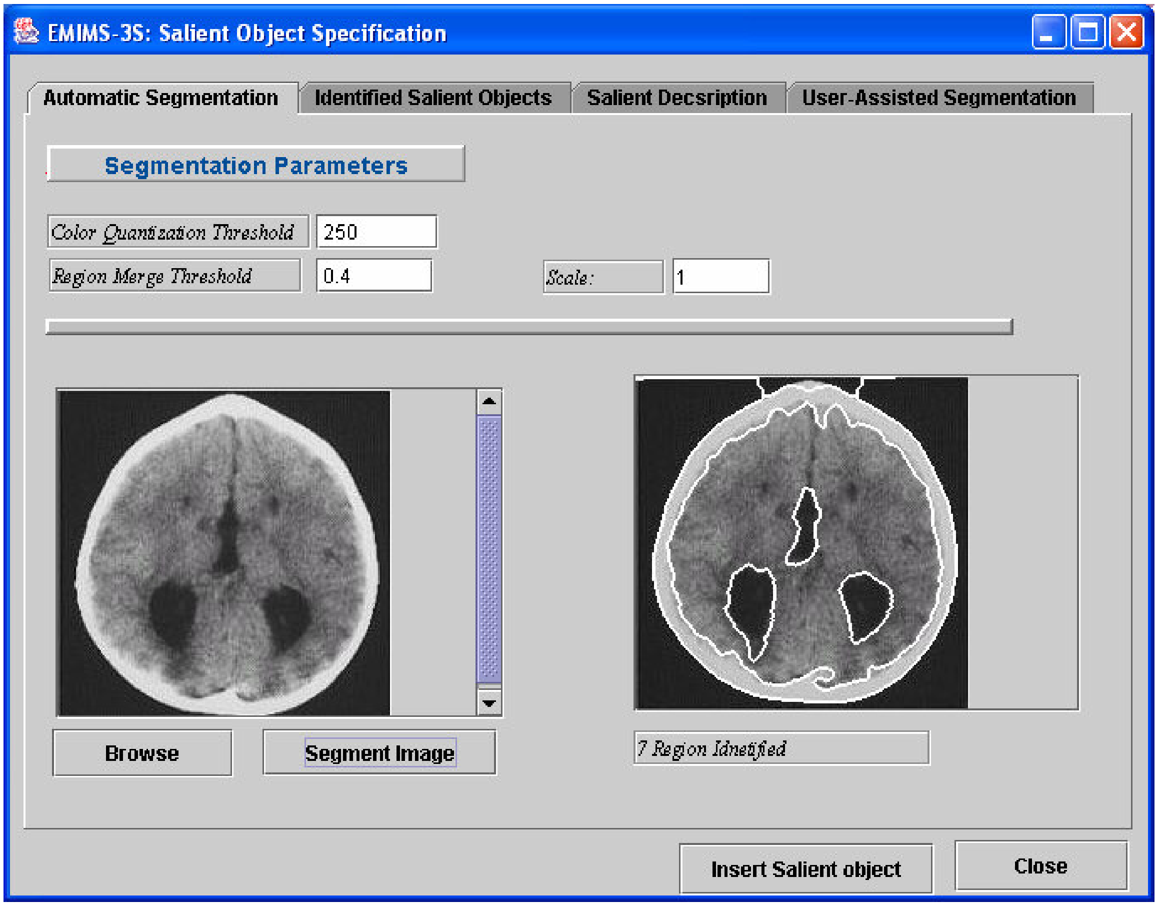


Figure 6.5: Segmented image with region boundaries superimposed.

If the user is not satisfied it is possible to change the parameters and segment the image until satisfactory result is obtained. Furthermore, the segmentation process creates a list of identified salient objects as shown in Figure 6.6 which is displayed when the user selects the **Identified Salient Objects** tab from the Salient Object Specification interface. When the user selects a image file path from the list shown in Figure 6.6, the salient object will be automatically displayed on the right side. The coordinates of the MBR is also automatically identified, which is important for spatial queries.

The **Insert Salient Object** button is used to store the selected salient object, and its associated metadata to the oracle salient object table of the database. The coordinates of the MBR of the selected salient object will also be stored into the salient-object table for responding to queries involving salient-object locations.

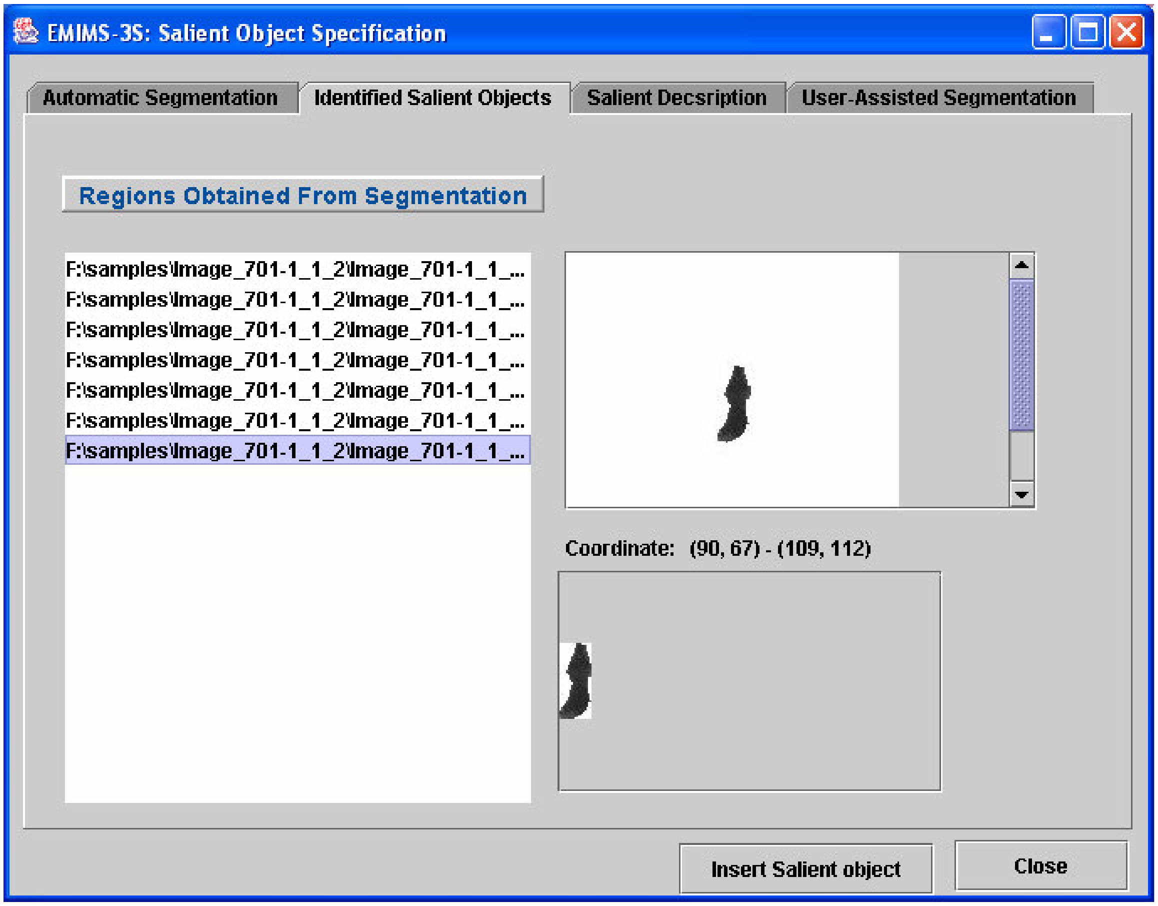


Figure 6.6: Identified salient-object from the image segmentation.

The **Salient Description** panel is used to enter metadata about the selected salient object image. Screen shot of the salient description panel is shown in Fig 6.7 below. The screen shot in Figure 6.8 shows the user selecting sub regions of the main image for user-assisted segmentation. Once the sub region is selected it is possible to segment the sub image and identify the salient objects by using the **Segment Sub Image** button.

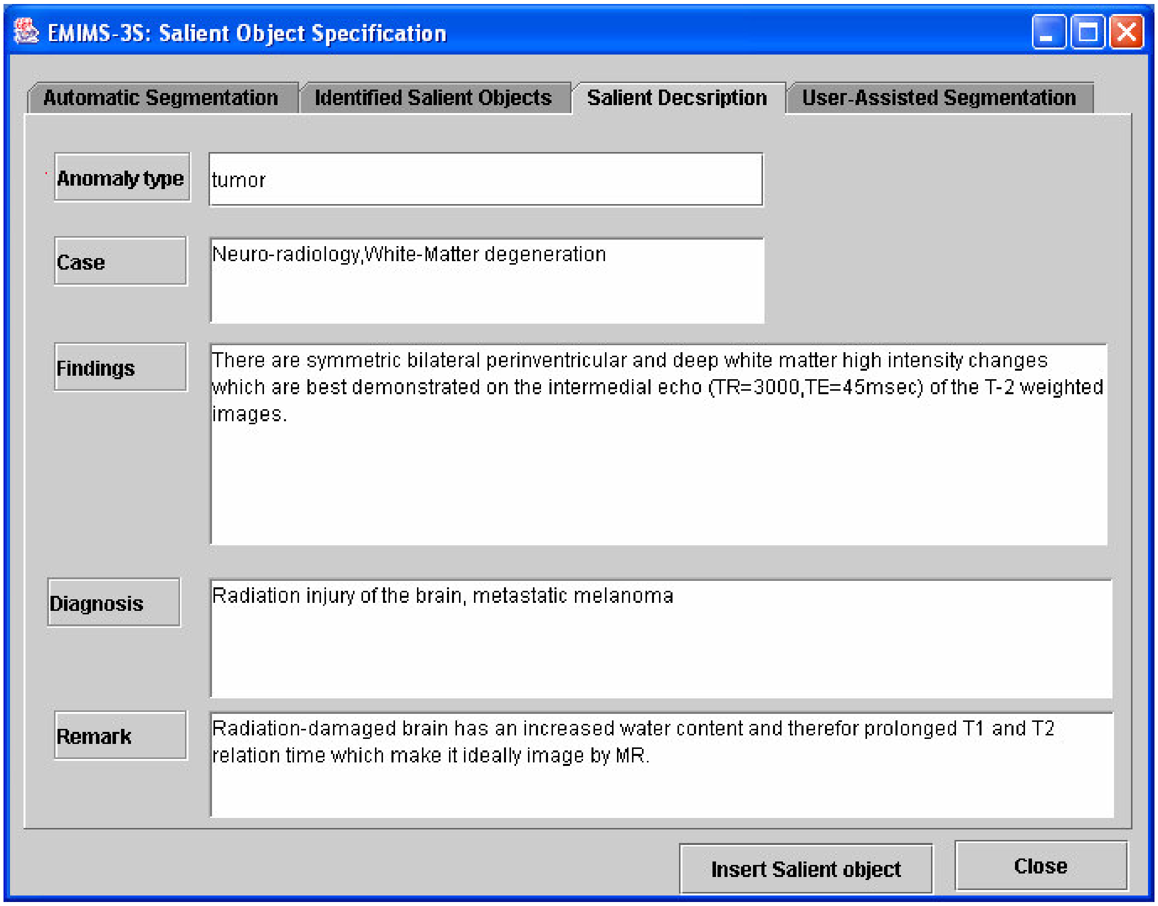


Figure 6.7: The salient metadata specification panel.

6.3.3 The Querying Interface of EMIMS-3S

The querying interface of EMIMS-3S is extended from the previous implementations. It supports segmenting an image into its salient objects, similarity-based matching on salient-object resulting from image segmentation and retrieval of whole images or the salient objects in the query result.

Using the query interface the user may query the image database using the whole image, the salient object, or metadata description. The EMIMS-3S querying interface is shown in Figure 6.9 illustrates query by example using a salient object obtained from image segmentation. Segmentation of an image into its salient objects is included in the interface to support QBE (Query By Example) using salient objects. The Querying Interface is also extended to display salient objects in the query result.

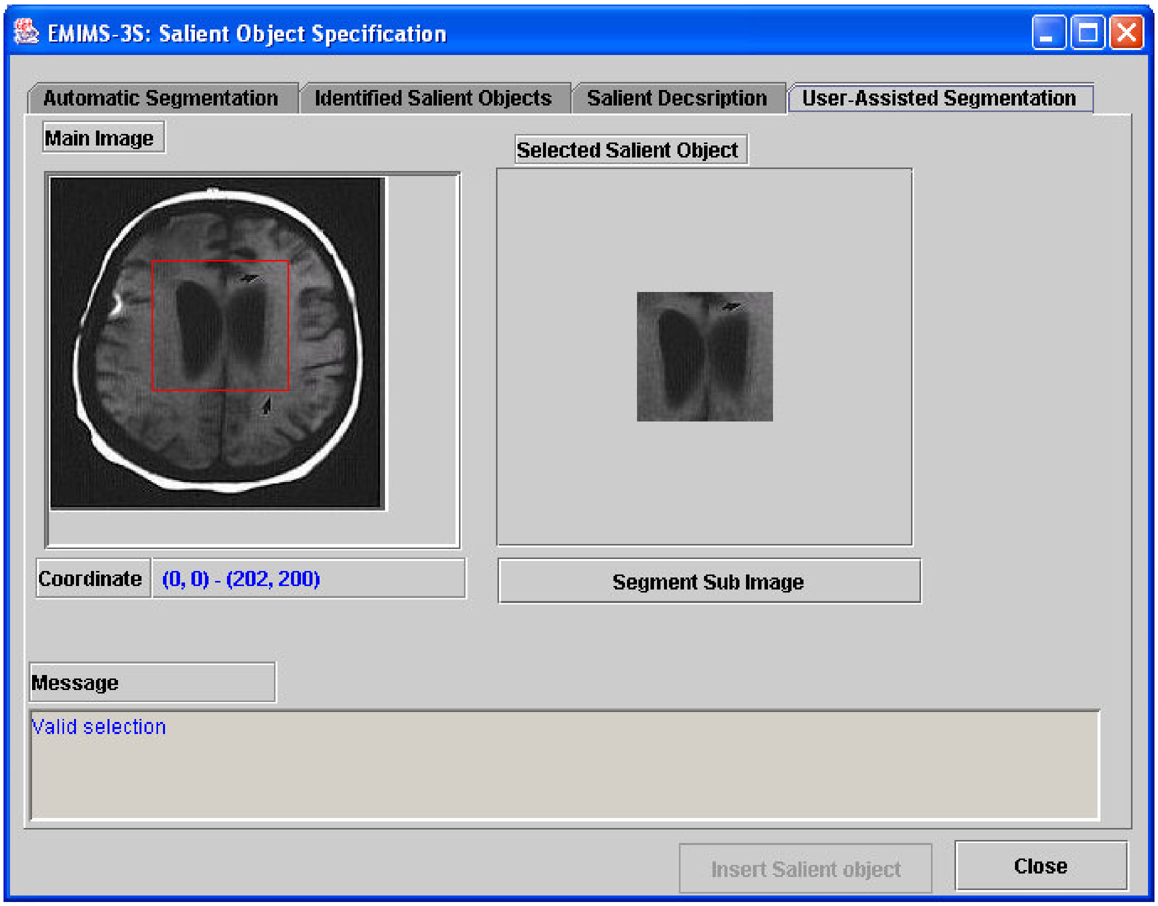


Figure 6.8: Sub Region Selected in user -assisted Segmentation

The **Browse** button is used to browse and insert an image. To segmented an image into its salient objects the user has to specify the required parameters and click on the **Segment** button. This will display the segmented image with region boundaries superimposed on the left pane (in place of the original image) and the path of the identified salient objects will be inserted in the combo box. The user can select the path of a salient object from this combo box to display the salient object with its MBR detected automatically on the middle scroll pane. The scroll pane located on the right side will display the images of the query result. Using the similarity parameters frame the user can specify the following options/settings on the query interface.

- **Use Main Image** and **Use Salient Object** are used to determine weather to use the salient object or the main image for similarity based matching respectively.
- The **Show Main Image In Query Result** Check Box is used to determine weather to retrieve and display main image or the salient object in the query results pane.
- The **Consider Salient Object Position** Check Box is used weather to consider the position of salient objects in salient object based retrieval or not.
- The **Threshold** Text Box is used to specify threshold value for similarity comparison.

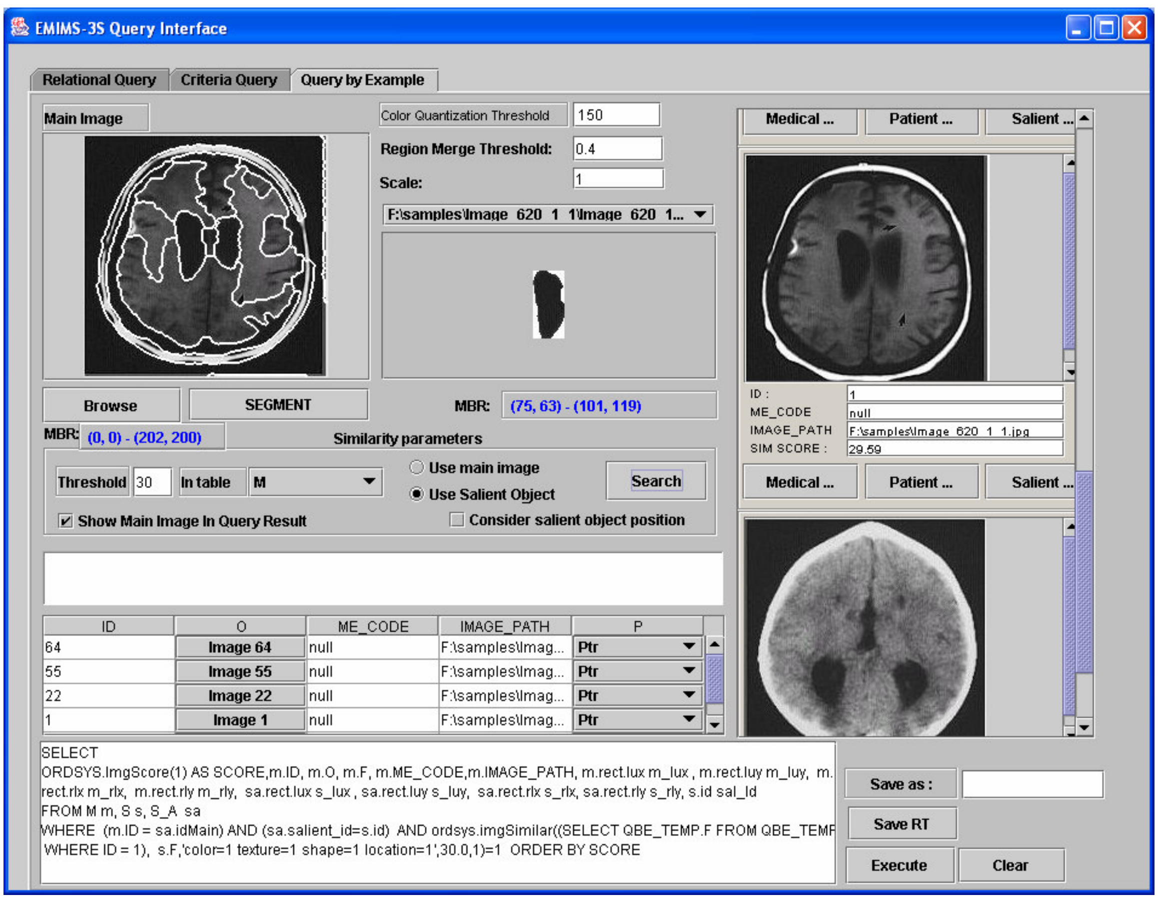


Figure 6.9: EMIMS-3S QBE (Query By Example) with whole images in query result

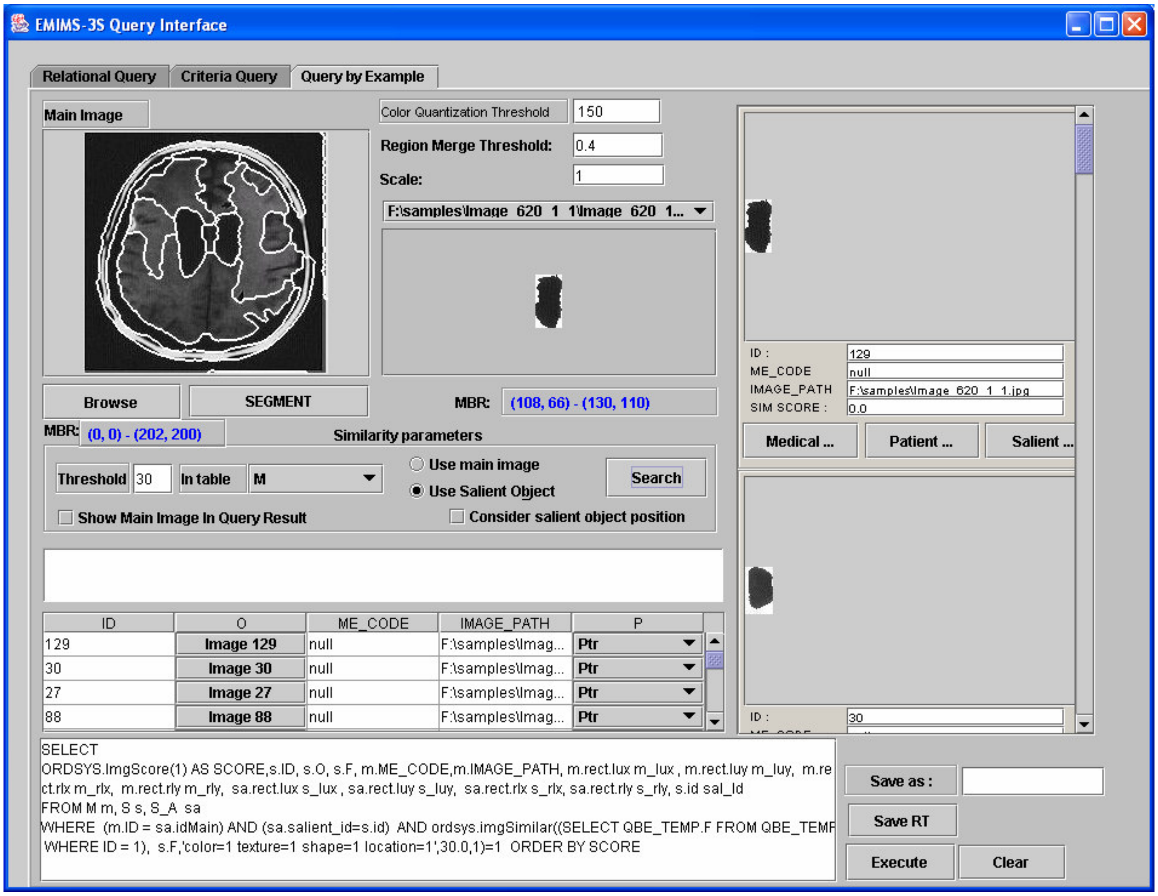


Figure 6.10: EMIMS-3S QBE Query Interface with Salient Objects in Query Result

The Query Interface screen shot shown in Figure 6.9 returns images contain image which have similar salient objects to the query salient object. But, with EMIMS-3S it is also possible to retrieve only salient object image similar to the query image as illustrated in Figure 6.10 above.

6.4 Experimental Results

The objective of this experiment is to find out the retrieval efficiency attained by the newly proposed technique that considers the shape of salient objects. The performance of the new proposal is compared with image retrieval system developed in [6] that uses MBR for salient object approximation. To compare these two techniques precision and recall measurements are used.

For the judgments on the similarity, subjective measures are used. Generally size, color and texture of the retrieved salient objects are used for making the judgments on the similarity. Precision and recall measures are used in this experiment to measure the retrieval efficiency. Precision and recall are the basic measures used in evaluating search strategies [34]. Although these concepts are frequently used in text searches, it is also possible to apply these concepts in image queries [6]. As described in [34], these measures assume:

1. There is a set of records in the database which is relevant to the search topic
2. Records are assumed to be either relevant or irrelevant
3. The actual retrieval set may not perfectly match the set of relevant records

Records (image retrieved in our case) must be labeled as relevant and irrelevant to calculate the recall and precision capabilities. Recall is the ratio of the number of relevant records retrieved to the total number of relevant records in the database. Precision is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved. Recall and precision are usually expressed in a percentage.

The image and their salient objects are stored in oracle 9i database under the windows platform. A personal computer with Pentium IV processor, 348 MB of RAM is used in conducting the experiment. Medical Images obtained from the American College of Radiology are used for making the comparison. The following steps are used to conduct the experiment:

1. Two databases are created in the oracle 9i database.
2. In Database 1 60 different brain images are stored in the main images table, M and 130 salient objects were extracted from the brain images and stored in the salient objects table, S using EMIMS-S developed in [6].

3. Similarly the same 60 different brain images are stored in the main image table, M and 130 salient objects were extracted from the brain images and stored in the salient object table, S on Database 2 using approach proposed in this work (the same image and their salient objects are used in both cases).
4. Ten salient objects are used as a query images. For each of these query images, different salient objects are manually (visually) identified as relevant.
5. The same salient objects were used as query images in the two database but the salient objects are extracted using the two approaches (MBR approximated salient objects were used on Database 1 and shape-sensitive salient objects were used on Database 2). A total of 20 queries are run. The results shown in Table 6.1 are obtained. A fixed threshold value of (ε) is used for each of the queries performed.
6. For each of the resulting images of each query, relevant retrieval and total retrieval are identified. Returned images are counted as relevant when they are found to be in the set of initially identified relevant images. These numbers are used to compute the precision and recall of the retrieval (Table 6.2).

Query Image	# of relevant Images (in M)	Image Query based on EMIMS-3S		Image query based on EMIMS-S	
		Total Retrieved	Relevant Retrieved	Total Retrieved	Relevant Retrieved
A	12	14	8	16	5
B	6	26	5	15	4
C	3	1	1	10	2
D	7	17	6	12	4
E	10	26	9	10	5
F	6	56	6	20	3
G	12	28	7	7	1
H	5	20	4	27	2
I	8	23	7	6	4
J	8	22	6	15	3

Table 6.1: relevant images of the ten query images and results of retrieval run on the two databases.

Query Image	Image Query based on EMIMS-3S		Image Query based on EMIMS-S	
	Precision	Recall	Precision	Recall
A	57.14	66.77	31.25	41.67
B	19.23	83.33	26.77	66.67
C	100	33.33	20	66.67
D	35.29	85.71	33.33	57.14
E	34.61	90	50	50
F	10.71	100	15	50
G	25	58.33	14.28	8.33
H	20	80	7.41	40
I	30.43	87.5	66.77	50
J	27.27	75	20	37.5

Table 6.2: Precision and recall measures for the ten query image run on the two databases.

The following general observations made:

- EMIMIS-3S returns large number of images
- The results returned by EMIMS-3S are close in color and texture to the query image.
- Most of the images returned by EMIMS-S are not similar to the image query

Figure 6.11 and Figure 6.12 below shows the comparisons of total retrieval and relevant retrieval for image queries based on the two approaches. Total retrieval refers to the total number of images retrieved as a result of the query run while the relevant retrieval refers to the number of relevant image retrieved. As can be observed the approached proposed in his work gives larger number of relevant and total retrieval in the query result in most of the queries run.

Total retrieval and relevant retrieval by EMIMS-3S on query C less than EMIMS-3S as can be observed from the experiments made. The total number of relevant image for query C was three images. EMIMS-3S return only a single image, which was relevant, while EMIMS-S returns 2 relevant images and eight irrelevant images, a total of ten images.

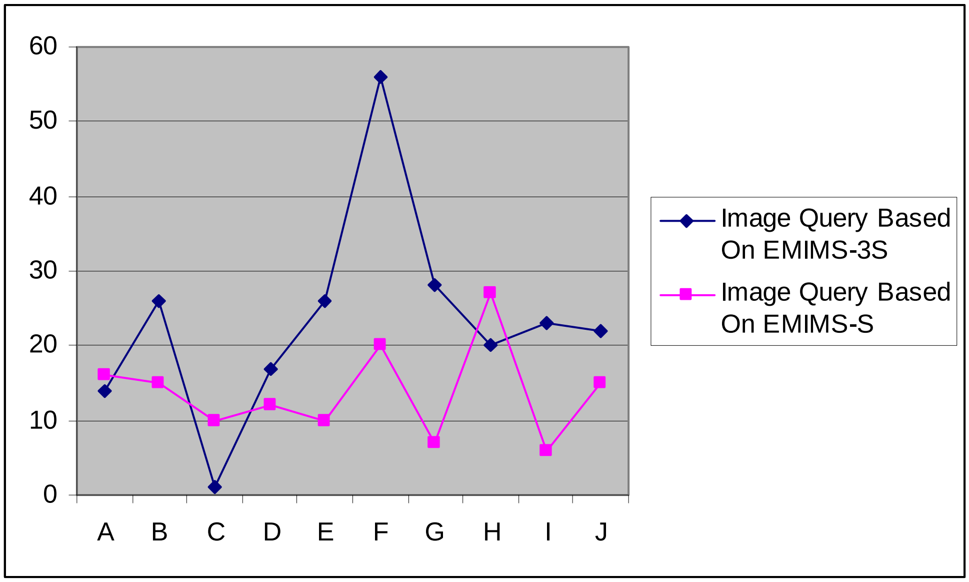


Figure 6.11: Total Retrieval

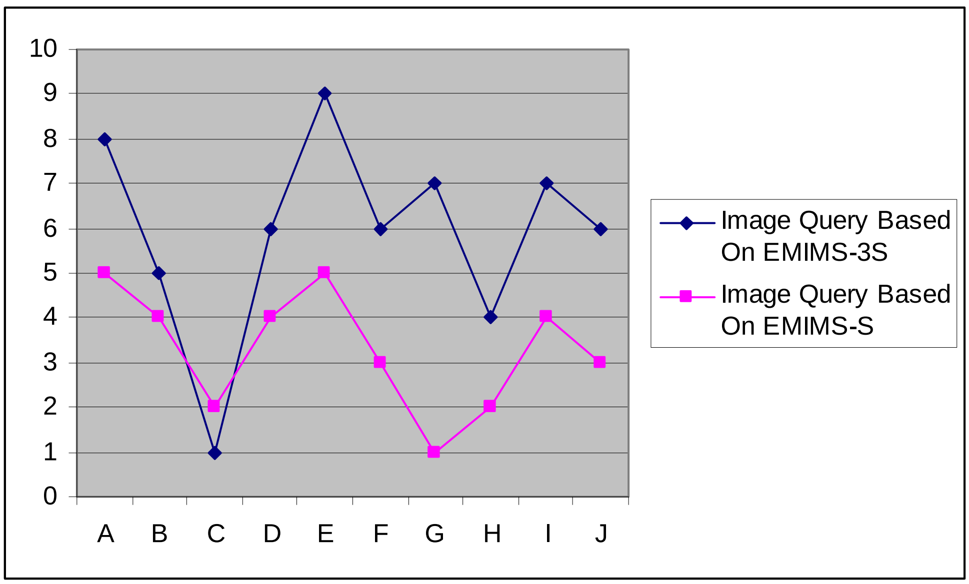


Figure 6.12: Relevant Retrieval

Comparisons on the precision and recall measure of the two approaches are illustrated in Figure 6.13 and 6.14 below. The recall measure of the proposed approach in this thesis is much greater than one proposed in [5]. But, the precision measures of the two approaches are close although the approach proposed is this work is better on the average. Although the precision is better, it is not exaggerated. This is attributed to the large number of retrieval, which decreases the precision.

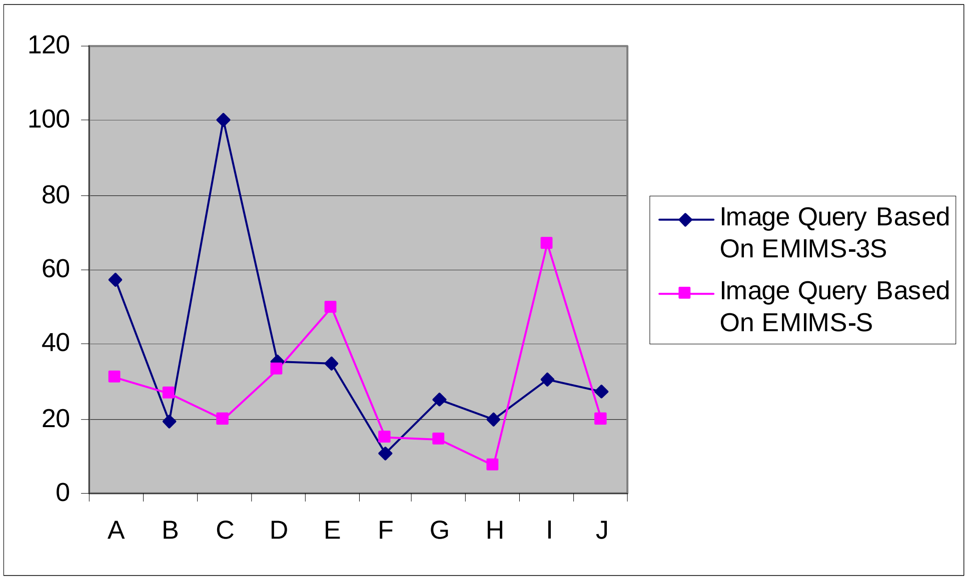


Figure 6.13: Precision measures of Image Queries based on the two approaches

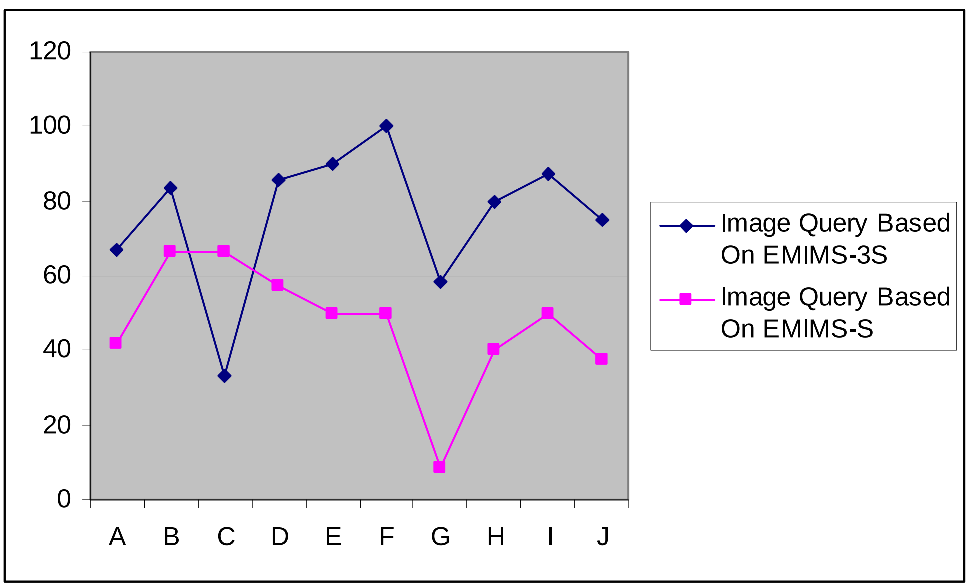


Figure 6.14: Recall measures of Image Queries based on the two approaches

Table 6.3 below summarizes the average recall, precision, total retrieval and relevant retrievals.

Method/Approach	Average total Retrieval (# of images)	Average Relevant Retrieval (# of images)	Average Precision (%)	Average Recall (%)
EMIMS-3S	23.3	5.9	36	76
EMIMS-S	13.8	3.3	28.5	46.8

Table 6.3: Average precision, recall, total retrieval and relevant retrieval

The precision measures are more or less similar as can be observed from table 7, although the shape-sensitive salient object based approach is better. But as the recall and relevant retrieval measures indicate the approach used in this work is better. This shows that the approach discussed in this work is better at retrieving image similar to the query image.

6.5 Summary

EMIMS-3S demonstrates image data management in an object-relational DBMS by considering the natural shape of the salient objects. It uses the extended data repository model to store semantic, spatial and content of salient objects which are identified from using image segmentation. The prototype demonstrates the following issues discussed in this thesis.

- Extraction of salient objects of interest from the image using either automatic or user-assisted segmentation techniques.
- Modeling of the image and its salient objects using the extended repository model
- Similarity-Based image retrieval with shape-sensitive salient objects.
- Option of retrieving either similar salient objects or image containing similar salient objects for similarity-based image queries.
- Automatic detection of the MBR of salient objects.

- Giving the user the option of selecting salient objects and storing the selected salient objects in a salient object table after image segmentation process.
- The effect on performance of queries caused by suppressing the background within the MBR of the salient object.

The prototype also demonstrates implementation of the two image segmentation approaches discussed in section 3.5 that are be used to identify salient object from an image. Through the extended data entry interface it is possible to store image and its constituent salient objects. It also demonstrates how to query the image using the salient objects identified from image segmentation process.

From the experiments done, it can be observed that the approach proposed in this thesis give a 30% increase in recall measure and 7.5 % increase in precision compared to the approach proposed in [6]. This could be attributed to the consideration of the natural shape of the salient objects and the method used to suppress the distortion caused by the background of the salient which reduces the inaccuracies that might occur in image queries.

Chapter 7
Conclusion and Future Works

Content Based Image Retrieval (CBIR) uses visual features of an image such as color, texture or shape for querying image databases. CBIR is nowadays widely adopted to tackle problem and complexity of image data management. There are a large number of image retrieval systems developed based on this technology, although CBIR can't fully satisfy all requirements with regard to image data management. In image retrieval, matching can be performed using global features of the whole image (as in CBIR), but a comparison that considers part(s) of images for similarity-based retrieval is sometimes a more natural approach. . Regions of an image that are of particular interest to humans are termed salient-objects of an image.

There were some researches conducted with regard to salient-object based image retrieval by as presented in [6]. The salient objects were approximated using MBR marked manually. The drawback of the above approach in [6] is that it doesn't consider the natural shape of salient object. This will have an impact on the precision of similarity-based image retrieval due to the inclusions of background, which sometimes have no relation to the salient-object. The other drawback is that identification of the salient objects an image is done manually by the user. Such procedure could be prone to errors and laborious, especially if there is a large collection of images and there are several salient objects within the image.

The theme of this thesis was also on salient-object based image retrieval. . In this work, a new approach was proposed which considers the natural shape of the salient objects contained within an image. An image segmentation technique is used to automatically identify salient objects and store them in an object relational DBMS for multi-criteria image query (query based on both image content, positional predicates and metadata).

Related works done on Content–Based Image Retrieval (CBIR), image retrieval in a database environment, salient object based image retrieval and image segmentation were reviewed in this thesis. The drawbacks of the previous works were also identified. Image data models, which can be used to capture relevant information about the image and image data repository model for storing image and its constituent salient objects, were assessed. Works in similarity-based image query algebra that can be used to retrieve image and their salient objects from an image database are also reviewed. Furthermore, spatial query operators, which are important for responding to queries that consider position of salient objects, were also assessed in detail. Finally, literatures on image retrieval with pervasive devices were reviewed.

Publicly available image segmentation systems were assessed in order to come up with image segmentation tool for identifying salient objects from an image. Sample image were used to find out how well the image segmentation segment the image. These system were compared using criteria's like the supported image type both for input and output, the efficiency of the segmentation, the availability of the source code, the programming language used, easiness of the segmented image for further processing etc. The JSEG image segmentation program is selected from the publicly available segmentation systems after making comparison using the criteria's described above.

A salient object identifier module is developed which is used for identification of salient objects using either automatic or user-assisted image segmentation techniques. In order to identify each salient object from the image two algorithms were developed: an algorithm that suppresses the background and an algorithm that automatically detect the MBR of the salient object. Modifications were made to the selected JSEG image segmentation program and it's compiled into a Dynamic Link Library (DLL) to produce a library that can be used from the image retrieval system developed using java .The Java Native Interface (JNI) is used to solve the interoperability problems since the JSEG library is implemented in C and the image retrieval system is developed using Java.

Once the shape sensitive salient objects were identified, the whole image and the identified salient objects have to be stored in to the image database. An appropriate image data repository model was identified and an extension was made to this model so that it can be used to store the main image, the salient object and metadata in an object-relational DBMS. Furthermore, to easily retrieve whole image or their salient object from the image database, two similarity-based selection operators were defined based on the similarity-based image query algebra developed in [5]. These defined operators are very import for responding to queries based on salient objects.

Furthermore, this work also proposed some techniques that can be incorporate into mobile image retrieval system to make images and image related data accessible on pervasive devices. It was proposed that pervasive devices could retrieve metadata or the salient object image, which can easily be processed and stored in mobile devices.

Finally, based on the techniques and systems proposed above, a prototype was developed using J2SE on top of Oracle9i, which is used for storing the medical images. To compare the retrieval effectiveness of the proposed approach in this thesis with the one proposed in [6], an experiment is conducted. Performs of the system were measure using recall and precision measures. From the experiment conducted it was observed that there is 7.5% increase in the precision measure and 36% increase in the recall measures, which shows that the proposed approach was better at retrieving relevant images from the image database.

The major contributions of this research work are the following:

- A salient object identification tool is designed for automatic identification of salient objects from an image based on the JSEG approach of image segmentation.
- An extension is made to the data repository model in [5] so that it will be possible to include the salient objects in image query results.

- Selection operators were defined based on previously developed similarity-based image query algebra for salient object based image retrieval.
- Algorithms were developed to automatically detect MBR of a salient objects and to suppress distortions caused by the background of salient objects within their MBR
- Server side design techniques were proposed to make the image, salient objects and their related data accessible on pervasive devices.
- A prototype is developed that demonstrate shape-sensitive identification of salient objects and using them form a more effective image retrieval of an image and multi-criteria image query.

Future works in this domain include designing of an image retrieval system for pervasive access. Although we have proposed some server side design technique to make the system accessible in a pervasive environment, much work is left like designing of an application that considers the bandwidth limitation of the communication medium, storage and processing limitation of the pervasive devices. The approach of image management in this work, supported by the JSEG image segmentation program can be used to identify region of interest in an image for many applications. Design of a medical image interpretation supports system, for example, for radiologists, using an appropriate image segmentation tool also is left as a future work.

References:

[1] Fuhui Long, Hongjiang Zhang, David D. Feng: Fundamentals of Content-based Image retrieval, in Multimedia Information Retrieval and Management - Technological Fundamentals and Applications, D. Feng, W.C. Siu, and H.J.Zhang. (ed.), Springer, 2002

[2] John P. Eakins : Retrieval of Still Images by Content. In proceedings of European Summer School in Information Retrieval (ESSR 2000),111-138, Villa Monsterio, Varena, Italy, September 2000.

[3] M. Emre Celebi, Y. Alp Aslandogan : Content Based Image Retrieval Incorporating Models of Human Perception, in *IEEE International Conference on Information Technology, Coding and Computing,* Las Vegas, NV, April 2004

[4] Yong Rui, Thomas S. Huang, and Shih-Fu Chang : Image Retrieval: Current Techniques, Promising Directions and Open Issues, *Journal of Visual Communication and Image Representation* , Vol. 10, 39-62, March, 1999

[5] Solomon Atnafu : Modelling and Processing of Complex Image Queries, Ph.D Thesis. Laboratories d'Ingenieres des system, d'Inforamtion INSA de lyon ,July 2003

[6] Dawit Bulcha.: Salient- object based image query by visual content, Master's thesis, Department of computer science, Addis ababa University , May 2004.

[7] Rong Zhao and William I. Grosky: Bridging the Semantic Gap in Image Retrieval , *Distributed Multimedia Databases: Techniques and Applications* , T. K. Shih (Ed.), Idea Group Publishing, Hershey, Pennsylvania, 2001.

[8] Markus Koskela, Jorma Laaksonen, Erkki Oja: Use of Image Subset Features in Image Retrieval with Self-Organizing Maps. International Conference on Image and Video Retrieval (CIVR 2004:508-516), Dublin City University, Ireland July 21-23, 2004

[9] A.Wardhani : Image Searching Tool Using Category-Based Indexing, Proceedings of APRS Workship on Digital Image Computing, APRS WORKSHOP ON DIGITAL IMAGE COMPUTING (WDIC), University of Queensland, St Lucia, Brisbane, Australia , 7 FEBRUARY 2003 .

[10] Vincent Oria, M. Tamer Özsu, Paul Iglinski, Bing Xu, L. Irene Cheng: DISIMA: An Object-Oriented Approach to Developing an Image Database System. 16**th** International Conference on Data Engineering (ICDE 2000), San Diego, CA, February 2000.

[11] P. A. Bromiley, N.A.Thacker and P. Courtney: Colour Image Segmentation by Non-Parametric Density Estimation in Colour Space ,Imaging Science and Biomedical Engineering Division Medical School, University of Manchester Manchester, M13 9PT, UK.

[12] M.Stonebraker and P.Brown: Object-Relational DBMSs : tracking the next greate wave, San.Francisco : Mogan Kaufmann Pub. Inc, 1999 ,287 p, ISBN 1-55860-452-9

[13] Skarbek W. and Koschan A : Colour Image Segmentation - A Survey, Technisher Bericht 94-32, Technical University of Berlin, October 1994;

[14] L. Lucchese and S.K. Mitra . Color Image Segmentation: A State-of-the-Art Survey. Dept. of Electrical and Computer Eng., University of California, Santa Barbara. Dept. of Electronics and Informatics, University of Padua, Italy

[15] T. Young , J.J. Gerbrands , L.J. van Vliet : Image Processing Fundamentals. Available online: http://www.ph.tn.tudelft.nl/Courses/FIP/noframes/fip-contents.htm. Quantitative Imaging group, Delft university of technology, Department of Imaging science and Technology , Faculty of applied sciences , (Consulted on Feb 2005.)

[16] R.M. Haralick, L.G. Shapiro: Image Segmentation Techniques," Computer Vision, Graphics, and Image Process ing, Vol. 29 , No. 1, Jan. 1985. CA, 10-11 Feb. 1997, pp. 143-154.

[17] William I. Grosky, Peter L. Stanchev: An Image Data Model. International Conference on Visual Information and Information Systems (VISUAL 2000), Lyon, France, November 2-4, 2000

[18] Jocjhen Burrkhardt , Dr.Horst Henn, Stefan Hepper, Llaus Rintdorff , Thomas Shack : Pervasive computing-Technology and architecture of mobile internet applications. Addison-Wesley, 2002.

[19] Aaron E.Caroll , MD, Sunil Saluja , Peter-Hornoch , Peter Tarczyh-Hornch.: Development of apersonal Digital assistant (PDA) Basied Client/Server NICU Patient Data and Charting Systems. Proc AMIA Symp, 2001:

[20]Uwe Hansmann, Lothar Merk, Martin S.Niklous, Thomas Stober: Pervasive Computing, Second Edition. Springer. 2003

[21] Kalle Radage: Oracle9*i*AS Wireless: Creating a Mobilized Business. *An Oracle White Paper. March 2002*

[22] Ahmad Iftikhar, Faouzi Alaya Cheikh, Bogdan Cramariuc and Moncef Gabbouj: Query by Image Content using NOKIA 9210 Communicator.source www.iva.cs.tut.fi/COST211/publications/ag_01_27.pdf (Consulted on June 2005)

[23] R. Chbeir, S. Atnafu, L. Brunie: Image Data Model for an Efficient Multi-Criteria Query: A Case in Medical Databases. 14th International Conference on Scientific and Statistical Database Management (SSDBM'02), Edinburgh, Scotland, July 24 - 26, 2002.

[24] Lang Xu, Marcel Jackowski, A. Ardeshir Goshtasby, D. Roseman, S. Bines, Clement T. Yu, A. Dhawan, A. Huntley: Segmentation of skin cancer images. Image Vision Comput. 17(1): 65-74 (1999)

[25] Yining Deng, B. S. Manjunath, Hyundoo Shin: Color Image Segmentation. CVPR 1999: 2446-2451

[26] Oracle® interMedia User's Guide and Reference Release 9.0.1. June 2001.

[27] Vincent Oria: Similarity Queries in the DISIMA Image DBMS. Avialable online : http://woodworm.cs.uml.edu/~rprice/ep/oria/.(Consulted on May 2005)

[28] Niels Nes - Martin Kersten: The Acoi Algebra: A Query Algebra for Image Retrieval Systems .Available online: http://homepages.cwi.nl/~niels/Pub/bncod/paper. , (Consulted on May 2005)

[29] Marcel Kratochvil, Piction*:* The move to store images in the database. Available online: http://www.piction.com., (Consulted on March 2005)

[30] Remco C. Veltkamp, Mirela Tanase : Content-Based Image Retrieval Systems: A Survey .Technical Report UU-CS-2000-34, October 2000.

[31] D. Comaniciu, P. Meer: *Robust Analysis of Feature Spaces: Color Image Segmentation*, IEEE Conf. Computer Vision and Pattern Recognition (CVPR'97), San Juan, Puerto Rico, 750-755, 1997.

[32] Robert Grimm, Thomas E. Anderson, Brian N. Bershad, David Wetherall: A system architecture for pervasive computing. Proceedings of the ACM SIGOPS European Workshop, Kolding, Denmark, September 17-20, 2000

[33] Dorin Comaniciu , Peter Meer. Mean Shift: A robust approach toward feature space analysis. IEE Transaction on pattern analysis and machine intelligence , vol. 24 , No .5 , May 2002.

[34] Richard Jizba: Measuring Search Effectiveness. Available online: http://www.hsl.creighton.edu/hsl/Searching/Recall-Precision.html (Consulted on June 2004).

Declaration

I, the undersigned, declare that this thesis is my original work and has not been presented for a degree in any other university, and that all source of materials used for the thesis have been duly acknowledged.

Declared by:

Name: ________________________________

Signature: ________________________________

Date: ________________________________

Confirmed by advisor:

Name: ________________________________

Signature: ________________________________

Date: ________________________________

Place and date of submission: Addis Ababa, June 2005.