[UCI]I804:24011-200000239322

2010 February

Master's Degree Thesis

Content Based Image Retrieval Using Median Filtering, Edge Detection, and Overlay Blocks

Graduate School of Chosun University

Department of Information and Communication Engineering

Zhao Hui

Content Based Image Retrieval Using Median Filtering, Edge Detection, and Overlay Blocks

25 February 2010

Graduate School of Chosun University

Department of Information and Communication Engineering

Zhao Hui

Content Based Image Retrieval Using Median Filtering, Edge Detection, and Overlay Blocks

Advisor: Jong-An Park

This thesis is submitted to Chosun University for the partial fulfillment of Master's Degree

2009 October

Graduate School of Chosun University

Department of Information and Communication Engineering

Zhao Hui

Zhao hui 의 석사학위 논문을 인준함

위원장 조선대학교 교수 Seung-Jo Han (인)

위 원 조선대학교 교수 Jae-Young Pyun(인)

위 원 조선대학교 교수 Jong-An Park (인)

2009 년 11월

조선대학교 대학원

저작물 이용 허락서

학 과	정보통신공학과	학 번	20087739	과 정	석사
성 명	영문 : Zhao Hui				
주 소	광주광역시 동구 서석동 375 조선대학교 전자정보공과대학 821호				
연락처	E-MAIL : hhhhh3023@163.com				
논문제목	영문 : Content Based Image Retrieval Using MedianFiltering, Edge Detection, and Overlay Blocks				

본인이 저작한 위의 저작물에 대하여 다음과 같은 조건 아래 조선대학교가 저작물을 이용할 수 있도록 허락하고 동의합니다.

\- 다 음 -

1. 저작물의 DB구축 및 인터넷을 포함한 정보통신망에의 공개를 위한 저작물의 복제, 기억장치에의 저장, 전송 등을 허락함.
2. 위의 목적을 위하여 필요한 범위 내에서의 편집과 형식상의 변경을 허락함. 다만, 저작물의 내용변경은 금지함.
3. 배포・전송된 저작물의 영리적 목적을 위한 복제, 저장, 전송 등은 금지함.
4. 저작물에 대한 이용기간은 5년으로 하고, 기간종료 3개월 이내에 별도의 의사 표시가 없을 경우에는 저작물의 이용기간을 계속 연장함.
5. 해당 저작물의 저작권을 타인에게 양도하거나 출판을 허락을 하였을 경우에는 1개월 이내에 대학에 이를 통보함.
6. 조선대학교는 저작물 이용의 허락 이후 해당 저작물로 인하여 발생하는 타인에 의한 권리 침해에 대하여 일체의 법적 책임을 지지 않음.
7. 소속 대학의 협정기관에 저작물의 제공 및 인터넷 등 정보통신망을 이용한 저작물의 전송・출력을 허락함.

동의여부 : 동의(O)

2010년 2월

저작자:Zhao Hui (인)

조선대학교 총장 귀하

ABSTRACT

Content Based Image Retrieval Using Median Filtering, Edge Detection, and Overlay Blocks

Zhao Hui

Advisor: Prof. Jongan Park

Dep. of Information and Communication Engineering

Graduate School of Chosun University

With the rapid development of technology of multimedia, content - based image retrieval (CBIR) has been an active research topic. Two CBIR algorithms will be introduced and several simulation results will be shown. One is a new CBIR method using the feature analysis of median filtering and edge extraction. Because median filter could reduce the image resolution especially on edges, edge detection is applied in order to get the edge values. Then the values of edge position of median filtered image are

replaced with detected edge values. It is aim at noise reduction while edge detail preservation. After feature extraction, histogram quantization and comparison are applied for feature vector calculation and similarity measurement. Experiment results show that this algorithm could preserve the image border details and has a good retrieval ability of image. Also in order to organize, manage, browse, and retrieve image database effectively, the second algorithm introduces a novel image retrieval algorithm using overlay blocks and maximal eigenvalues. The algorithm generates feature vectors using overlay parts between adjacent blocks, and generates the binary map using maximal eigenvalues and block truncation technique for image retrieval. It could noticed that compare with the traditional block method, the binary map generated using the proposed method could reflect the target object information more efficiently. The experimental results show that the indexes have good ability of characterizing the image contents.

The results generated through simulations produced through MATLAB software are also added where needed.

요약

메디안 필터링, 에지 검출 그리고 오버레이 블럭을 이용한 콘텐츠 기반 영상 검색

조혜

지도교수 박종안

정보통신공학과

조선대학교 대학원

멀티미디어 기술의 급속한 발전과 더불어 내용기반의 영상검색은 활발한 연구주제가 되어왔다. 본 논문은 두 개의 CBIR 알고리즘을 소개하고 몇몇의 시뮬레이션 결과를 보일 것이다. 첫 번째 알고리즘은 중간값 필터링의 특징 분석과 에지 추출을 이용한 새로운 CBIR 방법이다. 특히 에지에서 영상의 해상도를 줄이는 중간값 필터의 약점때문에 에지값을 얻기위해 에지 검출이 적용되고, 그 다음에 중간값으로 필터링된 영상의

에지 위치값이 검출된 에지값으로 대체된다. 에지의 세부내용을 저장하는 것과 동시에 잡음 제거를 목적으로 한다. 특징 추출후에 히스토그램 양자화와 히스토그램 비교는 특징 벡터 계산과 유사도 측정에 적용된다. 실험결과는 이 알고리즘이 영상경계의 디테일을 저장할 수 있고 좋은 검색 능력을 가지고 있음을 보여준다. 두번째로 소개할 알고리즘은 효율적으로 영상 데이터베이스를 구성, 관리, 탐색, 검색하기 위하여 오버레이 블록과 고유값의 최대치를 사용한 참신한 영상 검색 알고리즘이다. 이 알고리즘은 인접한 블록 사이의 오버레이 부분을 이용하여 특징 벡터를 생성하고, 영상 검색을 위해 최대 고유값과 블록 모서리를 잘라내는 기술을 사용한 이진화 맵을 생성한다. 기존의 블록 방법과 비교했을 때, 제안된 방법을 사용하여 생성된 이진화 맵이 더 효율적으로 target object 정보를 반영한다는 것을 알 수 있었다. 실험의 결과는 인덱스가 영상 컨텐츠의 특성을 나타내는데 좋은 능력이 있음을 보여준다.

실험결과는 필요한 부분에 추가된 MATLAB 에서 생성된 시뮬레이션을 통해 수행되었다.

Table of Contents

List of Figures

List of Tables

I. Introduction

A. Overview

In the first chapter, image retrieval and CBIR system will be introduced. Then two CBIR algorithms utilize median filtering, edge detection, and overlay blocks will be introduced in chapter 2 and chapter 3. The results generated through simulations produced through MATLAB software are also added where needed.

B. Image retrieval

The emergence of multimedia technology and the rapid growth in the number and type of multimedia assets controlled by public and private entities, as well as the expanding range of image and video documents appearing on the web, have attracted significant research efforts in providing tools for effective retrieval and management of visual data. Image retrieval is based on the availability of a representation scheme of image content. Image content descriptors may be visual features such as color, texture, shape, and spatial relationships, or semantic primitives.

Image retrieval techniques integrate both low-level visual features, addressing the more detailed perceptual aspects, and high-level semantic features underlying the more general conceptual aspects of visual data [1][2] shown as Figure 1.1.

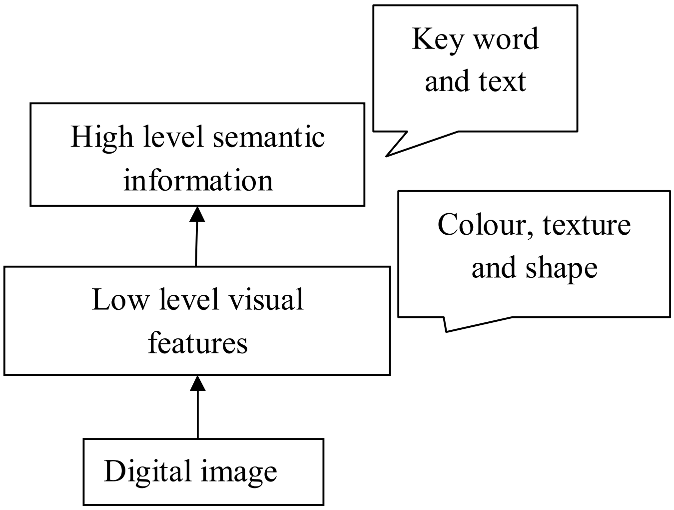


Figure 1.1: Image retrieval system

Conventional information retrieval is based solely on text, and these approaches to textual information retrieval have been transplanted into image retrieval in a variety of ways, including the representation of an image as a vector of feature values. However, "a picture is worth a thousand words." Image contents are much more versatile compared with text, and the amount of visual data is already enormous and still expanding very rapidly. Hoping to cope with these special characteristics of visual data, CBIR methods have been introduced. It has been widely recognized that the family of image retrieval techniques should become an integration of both low-level visual features, addressing the more detailed perceptual aspects, and high-level semantic features underlying the more general conceptual aspects of visual data. Neither of these two types of features is sufficient to retrieve or manage visual data in an effective or efficient way. Although efforts have been devoted to combining these two aspects of visual data, the gap between them is still a huge barrier in front of researchers. Intuitive and heuristic approaches do not provide us with satisfactory performance. Therefore, there is an urgent need of finding and managing the latent correlation between low-level features and high-level concepts. How to bridge this gap between visual features and semantic features has been a major challenge in this research field.

The different types of information that are normally associated with images are:

- Content-independent metadata: data that is not directly concerned with image content, but related to it. Examples are image format, author's name, date, and location.
- Content-based metadata:
    - Non-information-bearing metadata: data referring to low-level or intermediate-level features, such as color, texture, shape, spatial relationships, and their various

combinations. This information can easily be computed from the raw data.

- Information-bearing metadata: data referring to content semantics, concerned with relationships of image entities to real-world entities. This type of information, such as that a particular building appearing in an image is the *Empire State Building* , cannot usually be derived from the raw data, and must then be supplied by other means, perhaps by inheriting this semantic label from another image, where a similar-appearing building has already been identified.

Low-level visual features such as color, texture, shape and spatial relationships are directly related to perceptual aspects of image content. Since it is usually easy to extract and represent these features and fairly convenient to design similarity measures by using the statistical properties of these features, a variety of content-based image retrieval techniques have been proposed [3]. High-level concepts, however, are not extracted directly from visual contents, but they represent the relatively more important meanings of objects and scenes in the images that are perceived by human beings. These conceptual aspects are more closely related to users' preferences and subjectivity. Concepts may vary significantly in different circumstances. Subtle changes in the semantics may lead to dramatic conceptual differences. Needless to say, it is a very challenging task to extract and manage meaningful semantics and to make use of them to achieve more intelligent and user−friendly retrieval.

High−level conceptual information is normally represented by using text descriptors. Traditional indexing for image retrieval is text−based. In certain content−based retrieval techniques, text descriptors are also used to model perceptual aspects [4]. However, the inadequacy of text description is very obvious:

- It is difficult for text to capture the perceptual saliency of visual features.
- It is rather difficult to characterize certain entities, attributes, roles or events by means of text only.
- Text is not well suited for modeling the correlation between perceptual and conceptual features.
- Text descriptions reflect the subjectivity of the annotator and the annotation process is prone to be inconsistent, incomplete, ambiguous, and very difficult to be automated.

Although it is an obvious fact that image contents are much more complicated than textual data stored in traditional databases, there is an even greater demand for retrieval and management tools for visual data, since visual information is a more capable medium of conveying ideas and is more closely related to human perception of the real world. Image retrieval techniques should provide support for user queries in an effective and efficient way, just as conventional information retrieval does for textual retrieval. In general, image retrieval can be categorized into the following types:

1. Exact Matching – This category is applicable only to static environments or environments in which features of the images do not evolve over an extended period of time. Databases containing industrial and architectural drawings or electronics schematics are examples of such environments.
2. Low-Level Similarity-Based Searching – In most cases, it is difficult to determine which images best satisfy the query. Different users may have different needs and wants. Even the same user may have different preferences under different circumstances. Thus, it is desirable to return the top several similar images based on the similarity measure, so as to give users a good sampling. The similarity measure is generally based on simple feature matching and it is quite common for the user to interact with the system so as to indicate to it the quality of each of the returned matches, which helps the system adapt to the users' preferences. In general, this problem has been well-studied for many years.
3. High-Level Semantic-Based Searching – In this case, the notion of similarity is not based on simple feature matching and usually results

from extended user interaction with the system. Research in this area is quite active, yet still in its infancy. Many important breakthroughs are yet to be made.

For either type of retrieval, the dynamic and versatile characteristics of image content require expensive computations and sophisticated methodologies in the areas of computer vision, image processing, data visualization, indexing, and similarity measurement. In order to manage image data effectively and efficiently, many schemes for data modeling and image representation have been proposed. Typically, each of these schemes builds a symbolic image for each given physical image to provide logical and physical data independence. Symbolic images are then used in conjunction with various index structures as proxies for image comparisons to reduce the searching scope. The high-dimensional visual data is usually reduced into a lower-dimensional subspace so that it is easier to index and manage the visual contents. Once the similarity measure has been determined, indexes of corresponding images are located in the image space and those images are retrieved from the database. Due to the lack of any unified framework for image representation and retrieval, certain methods may perform better than others under differing query situations. Therefore, these schemes and retrieval techniques have to be somehow integrated and adjusted on the fly to facilitate effective and efficient image data management.

C. Existing Techniques of image retrieval

Visual feature extraction is the basis of any content-based image retrieval technique. Widely used features include color, texture, shape and spatial relationships. Because of the subjectivity of perception and the complex composition of visual data, there does not exist a single best representation for any given visual feature. Multiple approaches have been introduced for each of these visual features and each of them characterizes the feature from a different perspective.

1. Color

Color is one of the most widely used visual features in content-based image retrieval. It is relatively robust and simple to represent. Various studies of color perception and color spaces have been proposed, in order to find color-based techniques that are more closely aligned with the ways that humans perceive color.

The color histogram [5] has been the most commonly used representation technique, statistically describing combined probabilistic properties of the various color channels (such as the Red, Green, and Blue channels), by capturing the number of pixels having particular properties. For example, a color histogram might describe the number of pixels of each red channel value in the range [0, 255]. Figure 1.2 shows an image and three of its derived color histograms, where the particular channel values are shown along the x-axis, the numbers of pixels are shown along the y-axis, and the particular color channel used is indicated in each histogram.

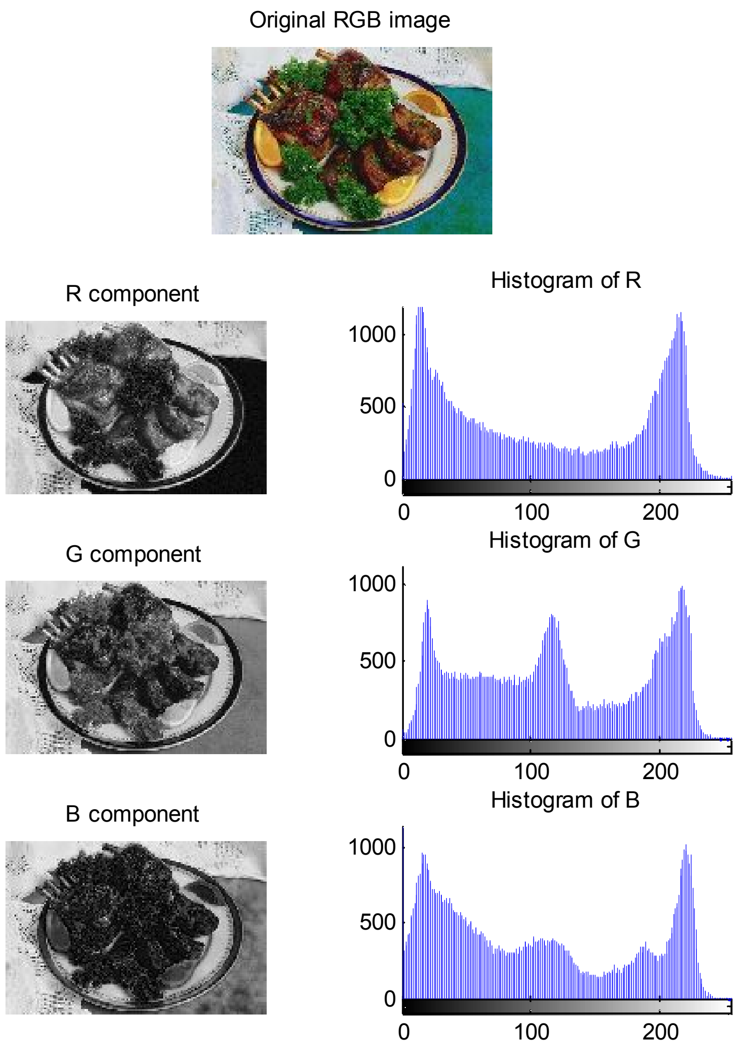


Figure 1.2: RGB image and three of its derived color histograms

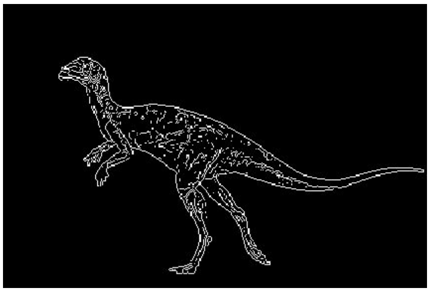
It is well known that histograms lose information related to the spatial distribution of colors and that two very different images can have very similar histograms. There has been much work done in extending histograms to capture such spatial information. Two of the well-known approaches for this are correlograms [6] and anglograms [7]. Correlograms capture the distribution of colors of pixels in particular areas around pixels of particular colors, while anglograms capture a particular signature of the spatial arrangement of areas (single pixels or blocks of pixels) having common properties, such as similar colors. We note that anglograms also can be used for texture and shape features.

2. Texture

Texture refers to the patterns in an image that present the properties of homogeneity that do not result from the presence of a single color or intensity value. It is a powerful discriminating feature, present almost everywhere in nature [8]. However, it is almost impossible to describe texture in words, because it is virtually a statistical and structural property. There are three major categories of texture-based techniques, namely, probabilistic/statistical, spectral, and structural approaches. Probabilistic methods treat texture patterns as samples of certain random fields and extract texture features from these properties. Spectral approaches involve the sub−band decomposition of images into different channels, and the analysis of spatial frequency content in each of these sub−bands in order to extract texture features. Structural techniques model texture features based on heuristic rules of spatial placements of primitive image elements that attempt to mimic human perception of textural patterns.

3. Shape

Shape representation is normally required to be invariant to *translation, rotation,* and *scaling*. Shape information of objects in images is also a very important image visual feature. Searching image databases using shape−based techniques is very common in CBIR systems. Usually in such a CBIR system, shape features of all images in the database are extracted and indexed, as well as query images. The system then searches the database to find images with a similar shape features to the query image. Segmentation is the most important step during image shape feature extraction and includes many procedures such as noise removing and edge detection. Figure 1.3 is an example of shape feature extraction. As we can see, since objects in this image have clear and clean boundaries, the shape information is extracted accurately.



(a) (b)

Figure 1.3: Shape extraction for an image with clear edges
(a) Original grayscale image
(b) Edge detected image using 'canny' operator

Shape-based Image Retrieval techniques can be categorized into boundary-based and region-based, according to shape feature representations. Boundary-based shape feature representation uses the outer edges of objects in an image, while region-based feature representation uses the entire shape region. Fourier Descriptor and Moment Invariants are the most famous techniques for these two categories, respectively. Fourier Descriptor [9] uses Fourier transformed edges as shape features. Moment Invariants [10] uses region-based moments as shape features which are invariant to transformations [11].

As the ultimate goal of image retrieval is to serve the needs and wants of users who may not even know what they are looking for but can recognize it when they see it, there has been much work done in trying to discover what is in the mind of the user. A very common technique for this is *relevance feedback*. Originally advanced in the information retrieval community, it has become a standard in most existing image retrieval systems, although some researchers believe that more involved user interactions are necessary to discover user semantics. This technique helps the system refine its search by asking the user to rank the returned results as to relevance. Based on these results, the system learns how to retrieve results more in line with what the user wants. There have been many new approaches developed in recent years, but the classical techniques are *query refinement* or *feature reweighting*. Query refinement transforms the query so that more of the positive and less of the negative examples will be retrieved. Feature reweighting puts more weight on features which help to retrieve positive examples and less weight on features which aid in retrieving negative examples. This process continues for as many rounds as is necessary to produce results acceptable to the user.

Needless to say, human beings are much better than computers at extracting and making use of semantic information from images. Many researchers believe that complete image understanding should start from interpreting image objects and their relationships. The process of grouping low-level image features into meaningful image objects and then automatically attaching correlated semantic descriptions to image objects is still a challenging problem in image retrieval. One of the earliest examples of such an approach is that used in the ImageMiner [12] system. Their method is structural in nature, using graph grammars, and generates scene descriptions with region labels. Current techniques in this area use Baysian [13] approaches which integrate textual annotations and image features.

D. Content-Based Image Retrieval (CBIR) Systems

CBIR, also known as query by image content (QBIC) [15] and content-based visual information retrieval (CBVIR) is the application of computer vision to the image retrieval problem, that is, the problem of searching for digital images in large databases [14].

"Content-based" means that the search will analyze the actual contents of the image. The term 'content' in this context might refer to visual features such as color, texture, shape, and spatial relationships, or any other information that can be derived from the image itself. Without the ability to examine image content, searches must rely on metadata such as captions or keywords, which may be laborious or expensive to produce.

There are several excellent surveys of content-based image retrieval systems. We mention here some of the more notable systems. The first, QBIC, was one of the first prototype systems. It was developed at the IBM Almaden Research Center and is currently folded into DB2. It allows queries by color, texture, and shape, and introduced a sophisticated similarity function. As this similarity function has a quadratic time-complexity, the notion of dimensional reduction was discussed in order to reduce the computation time. Another notable property of QBIC was its use of multidimensional indexing to speed-up searches. The Chabot system [16], developed at the University of California at Berkeley, brought text and images together into the search task, allowed the user to define concepts, such as that of a sunset, in terms of various feature values, and used the post-relational database management system Postgres [17]. Finally, the MARS system [18], developed at the University of Illinois at Urbana-Champaign, allowed for sophisticated relevance feedback from the user.

II. Feature analysis based on Median Filtering and Edge Extraction for CBIR

A. Overview

With the rapid development of technology of multimedia, the traditional information retrieval techniques based on keywords are not sufficient. CBIR has been an active research topic. A new CBIR method using the feature analysis of median filtering and edge extraction is proposed. Because median filter could reduce the image resolution especially on edges, edge detection is applied in order to get the edge values. Then the values of edge position of median filtered image are replaced with detected edge values. It is aim at noise reduction while edge detail preservation. After feature extraction, histogram quantization and comparison are applied for feature vector calculation and similarity measurement. Experiment results show that this algorithm could preserve the image border details and has a good retrieval ability of image.

B. Introduction

Due to feature extraction is a key step of any image retrieval system, selection of features is an important element for determining the image retrieval performance of the final system. Before feature extraction process it is usually necessary to perform high degree of noise reduction in an image before performing any high−level processing steps. Median filtering is one of the techniques used to reduce noise from the images.

1. Median filtering

Median filtering is a common step in image processing. It is a non−linear digital filtering technique, often used to remove noise from images or other signals, particularly useful to reduce speckle noise and salt and pepper noise [19]. This is performed using a window consisting of an odd number of samples. Figure 2.1 shows an example of 5x5 median filtering.

(a) (b)

Figure 2.1: Median filtering
(a) Original grayscale image
(b) Median filtered image using '5x5 block'

1. The values in the window are sorted into numerical order.
2. The median value, the sample in the center of the window, is selected as the output.
3. The oldest sample is discarded, a new sample acquired, and the calculation repeats.

A common problem with median filter based on all adjacent pixels is how to process the edges of the image. Show as figure 2.1, it courses energy loss on edges. It is also more complex to write a filter that includes a method to specifically deal with the edges. Jongan Park [19] et al. proposed a method using Median Filtering on RGB Segments for Image Retrieval.

2. Edge detection

Edge detection is a terminology in image processing and computer vision, particularly in the areas of feature detection and feature extraction, to refer to algorithms which aim at identifying points in a digital image at which the image brightness changes sharply or more formally has discontinuities. The Canny edge detection operator [20] was developed by John F. Canny in 1986 and the Canny edge detection algorithm uses two thresholds to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. It is more likely to detect true weak edges. Figure 2.2 shows edge detection using different operators.

Figure 2.2: Edge detection using different operators

(a) Original image

(b) Edge detection using ‘sobel’ operator

(c) Edge detection using ‘prewitt’ operator

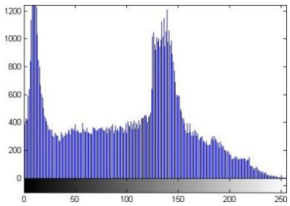
(d) Edge detection using ‘canny’ operator

3. Histogram equalization

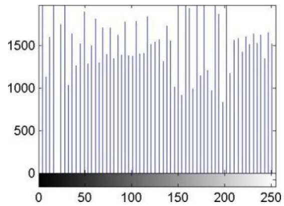
Histogram equalization is a method in image processing of contrast adjustment using the image histogram [21]. This method usually increases the global contrast of many images, especially when the usable data of the image is represented by close contrast values. Through this adjustment, the intensities can be better distributed on the histogram.

This allows for areas of lower local contrast to gain a higher contrast without affecting the global contrast. Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values. The method can lead to better views of structure in images, and to better detail in the case that are over or under−exposed. Figure 2.3 shows an example of histogram equalization.

(a)



(b) (c)



(d) (e)

Figure 2.3: An example of histogram equalization
(a) Original Image
(b) Grayscale image
(c) Histogram of (b)
(d) Histogram equalized image of (b)
(e) Histogram of (d)

C. Method

This section reports a comparative study of two algorithms which cater retrieval of images using median filtering. One algorithm used just pure median filtering and histogram quantization method for feature extraction; the other one is a new image feature analysis technique that is proposed using median filtering and edge value preservation, which could compensate the energy loss of median filtering on edges.

1. Pure median filtering method

The procedure of pure median filtering method through 5 Steps:

(1). The RGB image is acquired and converted to grayscale.
(2). Get the histogram equalized grayscale image.
(3). 3x3 blocks median filtering is applied to the histogram equalized (HE) grayscale image.
Following steps are performed to extract the feature vector for comparison:
(4). Quantized the filtered image using 32 bins.
(5). Calculate average value of pixels for each bin, and these values are used to construct the feature vector.

2. Median filtering and edge compensation method

The image feature analysis technique is proposed using median filtering and edge value preservation. Because median filtering could reduce image resolution especially on edges, edge detection is applied in order to reserve the edge pixel values. All the edge positions are detected from the histogram equalized image. Median filtering is also applied in parallel on the histogram equalized image. Then the pixel values of detected edge position are replaced in median filtered image. It is aim at noise reduction while edge detail preservation.

The following Figure 2.4 shows the details of processing:

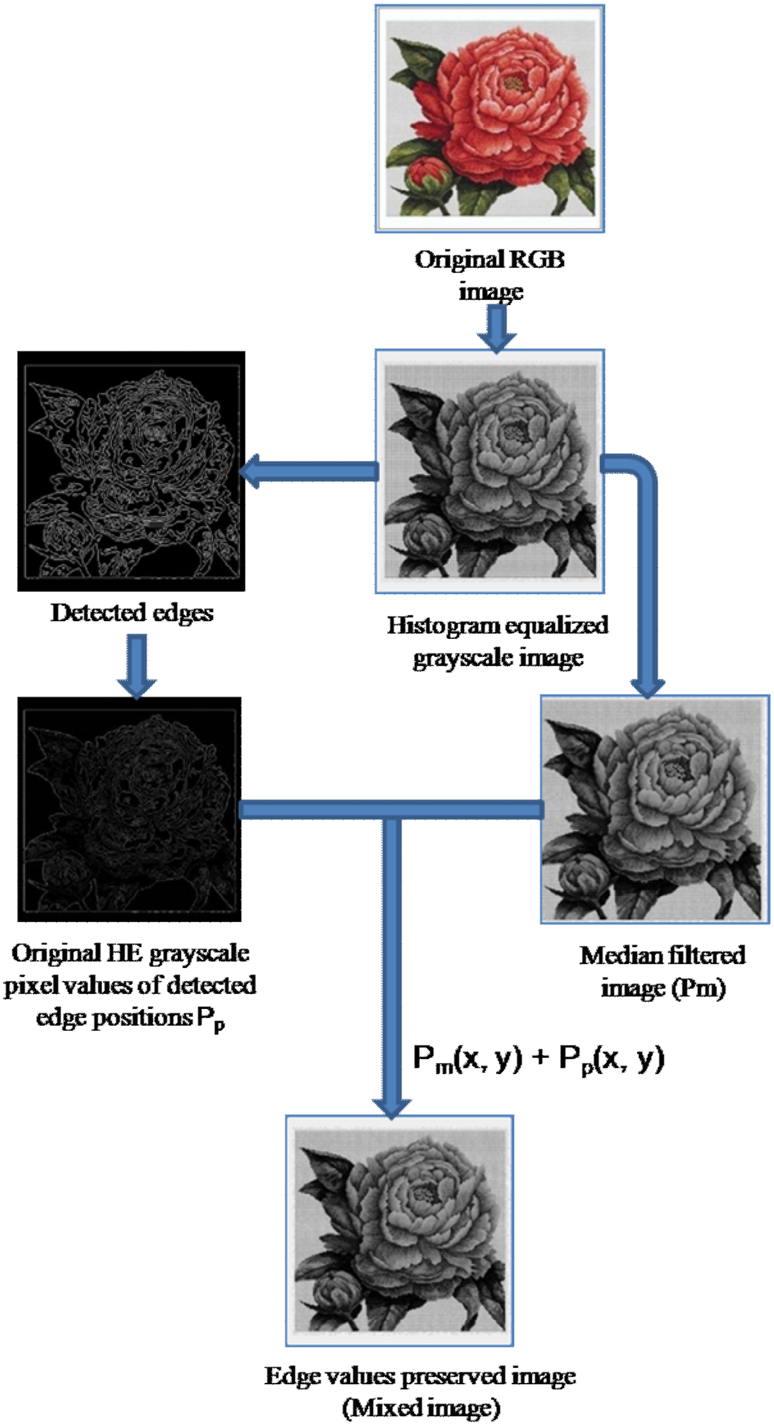
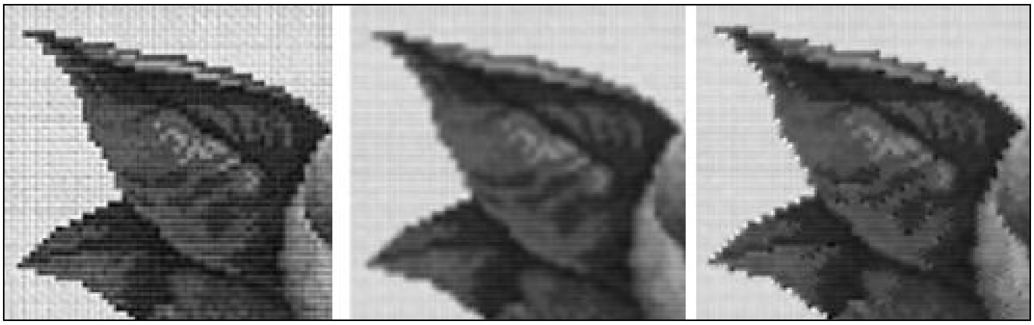


Figure 2.4: Procedure of the proposed method

Steps:

1. The RGB image is acquired and converted to grayscale.
2. Get the histogram equalized grayscale image.
3. Edge detection using 'canny' detector.
4. 3x3 blocks median filtering is applied to the histogram equalized grayscale image.
5. Replace the original pixel values with detected edge positions in the median filtered image.

The following Figure 2.5 shows the difference between images:



(a) (b) (c)

Figure 2.5: Difference between each image
(a) Original HE gray scale image
(b) Median filtered image
(c) Edge values preserved image

The algorithm used histogram quantization to retrieve images. Figure 2.6 shows edge values preserved image and histogram.

1. Quantized edge values preserved image (mixed image) using 32 bins.

2. Calculate average value of pixels for each bin, and these values are used to construct the feature vector.

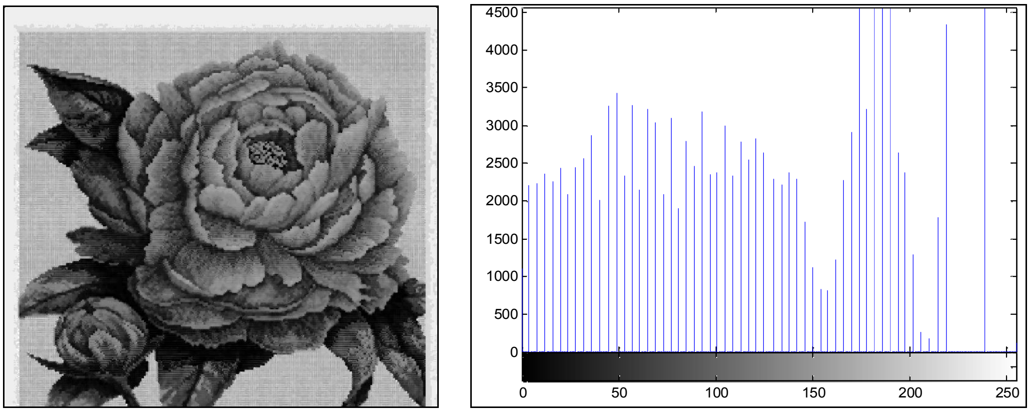


Figure 2.6: Edge values preserved image and histogram

Two images A and B can be compared by using the L1 (Euclidian) distance.

$$d(A,B) = \sum_{j=1}^{n} \left| H_j^A - H_j^B \right| \tag{2.1}$$

The following Figure 2.7 shows procedure of entire retrieval process.

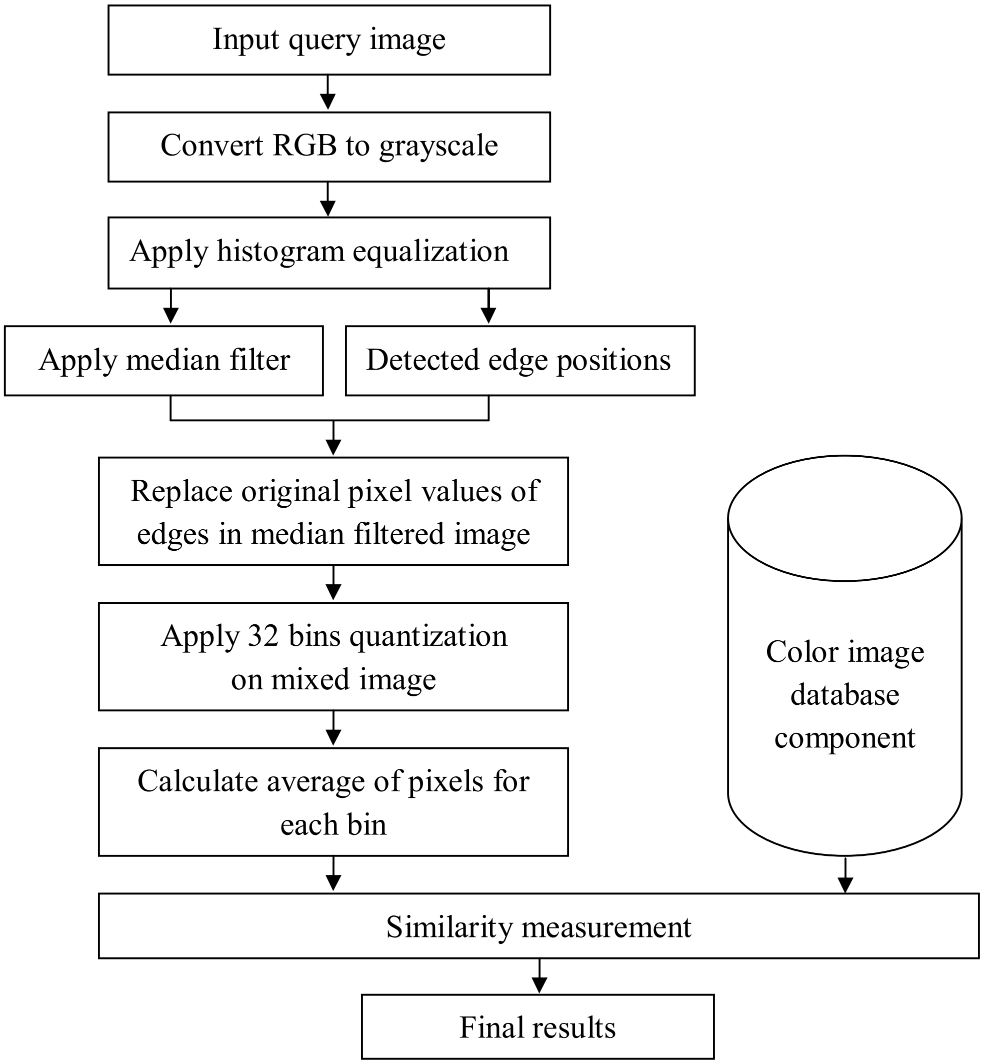


Figure 2.7: Procedure of the proposed algorithm

D. Simulation and results

The simulation engine consists of feature extraction process, batch feature extraction and storage process for a collection of images, and the interactive process. Figure 2.8 shows overview of entire retrieval process.

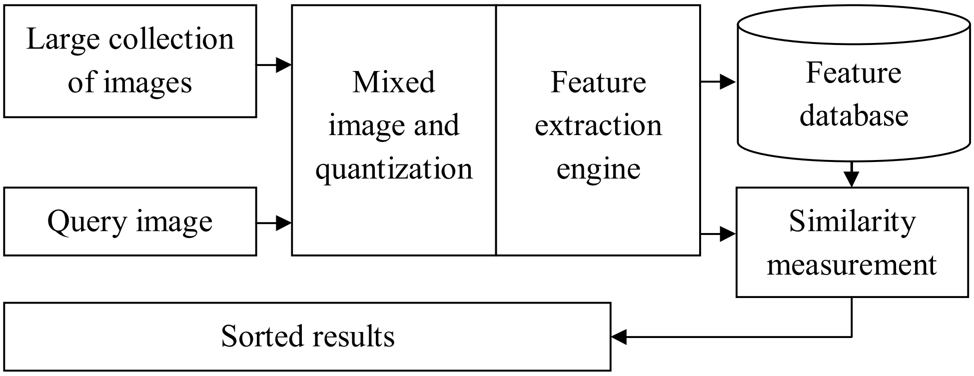


Figure 2.8: Overview of entire retrieval process

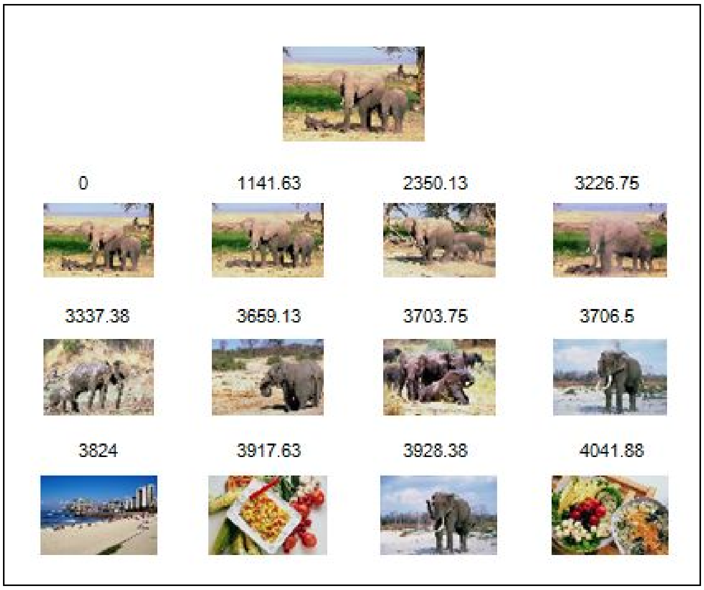
MATLAB 7.0 is used to verification the two algorithms. And the WANG database [22] [23] is used for testing the proposed idea.

Figure 2.9 Categories of the database

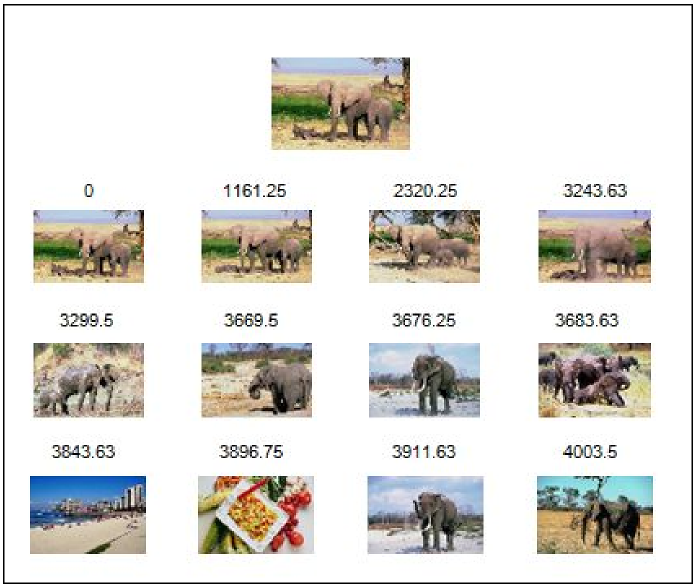
The image database is freely available to researchers and is free from any conditions or restrictions.

1. To minimize the dependence on hardware, the entire image collection is stored in main memory during evaluation.
2. The initial number of images in the database is 1000.
3. Images are in JPEG format.
4. 10 different types of objects are available in the database, including people, sea, building, bus, dinosaur, elephant, flower, horse, mountain, and food. Each type has 100 images shown as Figure 2.9.

The following Figure 2.10, 2.11 and 2.12 show particular example of different types acceding to two algorithms. The first image is query image. The similarity is reduced from the top down and from left to right.



(a)



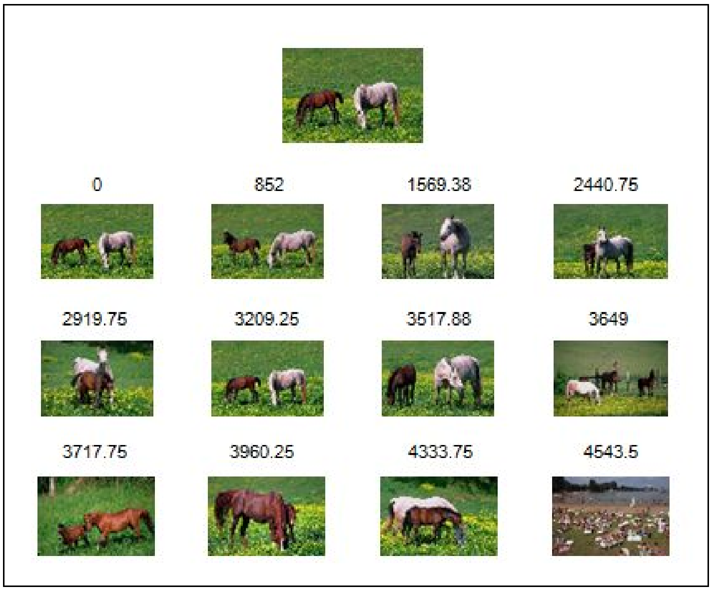
(b)

Figure 2.10: Simulation results comparation of elephant's case
(a)Pure median filtering method
(b)Median filtering and edge compensation method

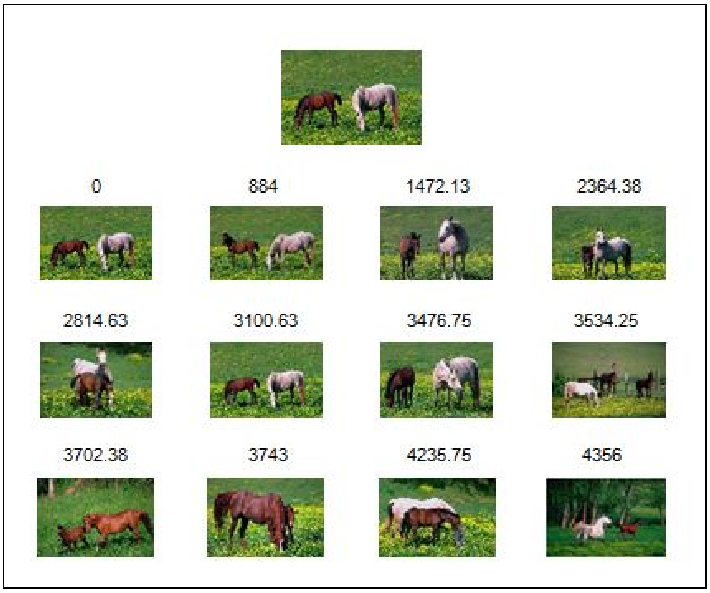
(a)

(b)

Figure 2.11: Simulation results comparation of flower's case
(a)Pure median filtering method
(b)Median filtering and edge compensation method



(a)



(b)

Figure 2.12: Simulation results comparation of horse's case
(a)Pure median filtering method
(b)Median filtering and edge compensation method

Each type of image was tested separately using the two algorithms, and calculate the precision for comparison. Method 1 is pure median filtering method, method 2 is median filtering and edge compensation method. From the precision comparison result we can see:

Table 2.1: Precision comparison of different categories

Top 30 results

	Method 1	Method 2
Dinosaurs	1	1
Flowers	0.90	0.91
Animals	0.76	0.80
Buses	0.70	0.73
Foods	0.56	0.60
Buildings	0.50	0.53
Human beings	0.40	0.46

1. Precision of median filtering and edge preservation algorithm is better than pure median filter based algorithm based on the same recall.
2. The advantage of median filtering and edge preservation algorithm is obviously to those complex types of images.
3. It was also noticed that the benefit of method 2 could verify that compensate the energy loss of median filtering on edges is helpful for improving precision.

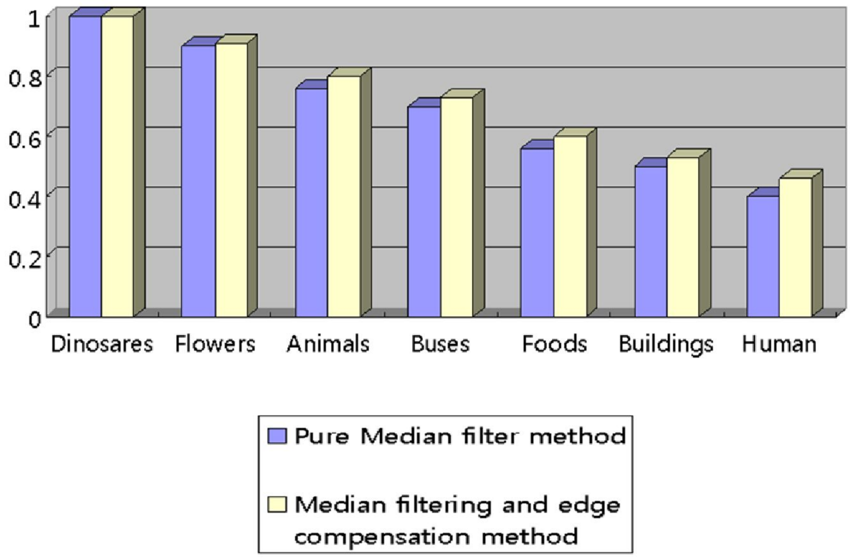


Figure 2.13: Precision comparisons

E. Conclusion

A comparative study of two algorithms which cater retrieval of images using median filtering was introduced. Both of the two algorithm use median filtering and histogram for feature analysis. For the pure median filtering method could reduce image resolution especially on edges, the median filtering and edge compensation method was proposed that compensated the energy loss of median filtering on edges and accomplished the aim of noise reduction while edge detail preservation. A lot of experimental results according to different types of images showed that the recall and precision of median filtering and edge preservation algorithm is better than pure median filter based algorithm. This work would be helpful for selecting and optimizing retrieval algorithms.

III. Feature Extraction Using Overlay Blocks and Maximal Eigenvalues for Image Retrieval

A. Overview

With the rapid development of technology of multimedia, CBIR has been an active research topic. In order to organize, manage, browse, and retrieve image database effectively, a novel technique for CBIR using overlay blocks and maximal eigenvalues will be introduced. Because image pixels have strong relationship with each other, the proposed algorithm generates feature vectors that use overlay parts between adjacent image blocks, and generates the binary map using maximal eigenvalues and block truncation technique for image retrieval. It was noticed that the binary map generated using the proposed method could reflect the target object information more efficiently. The experimental results show that the indexes have good ability of characterizing the image contents.

B. Introduction

CBIR retrieves images based on their visual similarity to a user−supplied query image or user−specified image features. The visual content such as color, shape and image structure is considered for the retrieval of images instead of an annotated text method. One major problem with CBIR is the issue that arises when predicting the relevancy of retrieved images. This retrieval is based on various image features. The objective is the selection of such features which can provide accurate and precise query results.

For any image, the image block can be easier visually distinguished than image pixel. The background mainly consists of smooth blocks, and the target object and details consist of non−smooth blocks. In CBIR system, non−smooth blocks made a major contribution than smooth blocks.

The proposed algorithm highlighted the contribution of non−smooth blocks. And because image pixels have strong relationship with each other, the proposed algorithm generates feature vectors that use overlay parts between adjacent image blocks, and utilize maximal eigenvalues and block truncation technique to generate binary map for image retrieval. The algorithm is detailed explained in chapter C.

C. Method

1. Block Truncation technique

Block Truncation Coding, or BTC [24], is a type of lossy image compression technique for grayscale images. It divides the original images into blocks and then uses a quantiser to reduce the number of grey levels in each block.

A 256x256 pixel image is divided into blocks of typically 4x4 pixels. For example, there is a 4x4 block from an image as shown in Figure 3.1.

245	239	249	239
245	245	239	235
245	245	245	245
245	235	235	239

Figure 3.1: A 4x4 block from an image

For each block the Mean and Standard Deviation are calculated, these values change from block to block. The mean of given block is 241.875. If a pixel value is greater than the mean, it is assigned the value "1", otherwise "0". The binary map can be generated as shown in Figure 3.2.

1	0	1	0
1	1	0	0
1	1	1	1
1	0	0	0

Figure 3.2: Binary map

2. Overlay block method

Before utilizing the idea of block truncation code technique, the original image matrix is overlay blocked. The method is shown in Figure 3.3. The block size depends on the sampling point as shown. There is an overlay part between adjacent blocks. We could notice that every block of the image is related with each other. It improved the relationship among the adjacent pixels.

7	11	14	14	35	29	25	28	30	9	10	2	24	23	19	12
21	15	18	14	36	94	230	208	187	8	10	10	25	20	12	14
17	21	22	5	48	198	201	208	190	208	13	11	27	18	16	14
6	21	26	6	51	196	199	196	186	211	197	187	28	9	15	15
8	17	21	14	6	194	195	175	183	204	188	180	9	8	20	17
21	18	8	19	23	203	174	180	184	163	189	210	164	9	14	21
23	30	18	231	202	166	197	165	205	185	208	212	22	15	13	11
26	32	46	188	191	187	204	164	237	207	201	200	8	6	24	29
43	54	58	213	196	192	182	227	239	230	215	204	14	37	28	26
45	50	63	201	233	221	230	233	206	234	207	182	21	34	34	25
68	44	43	180	227	203	193	237	230	185	199	166	13	36	35	21
65	30	40	47	222	202	200	211	211	188	199	210	5	41	43	25
25	22	28	11	201	212	212	216	224	191	206	172	7	11	50	27
19	11	7	9	130	229	217	232	222	168	208	5	7	8	55	32
16	12	9	13	20	30	35	24	143	142	71	7	7	3	15	35
16	13	14	16	14	6	16	23	13	34	19	6	4	3	13	16

(a)

7	11	14	14	35	29	25	28	30	9	10	2	24	23	19	12
21	15	18	14	36	94	230	208	187	8	10	10	25	20	12	14
17	21	22	5	48	198	201	208	190	208	13	11	27	18	16	14
6	21	26	6	51	196	199	196	186	211	197	187	28	9	15	15
8	17	21	14	6	194	195	175	183	204	188	180	9	8	20	17
21	18	8	19	23	203	174	180	184	163	189	210	164	9	14	21
23	30	18	231	202	166	197	165	205	185	208	212	22	15	13	11
26	32	46	188	191	187	204	164	237	207	201	200	8	6	24	29
43	54	58	213	196	192	182	227	239	230	215	204	14	37	28	26
45	50	63	201	233	221	230	233	206	234	207	182	21	34	34	25
68	44	43	180	227	203	193	237	230	185	199	166	13	36	35	21
65	30	40	47	222	202	200	211	211	188	199	210	5	41	43	25
25	22	28	11	201	212	212	216	224	191	206	172	7	11	50	27
19	11	7	9	130	229	217	232	222	168	208	5	7	8	55	32
16	12	9	13	20	30	35	24	143	142	71	7	7	3	15	35
16	13	14	16	14	6	16	23	13	34	19	6	4	3	13	16

(b)

7	11	14	14	35	29	25	28	30	9	10	2	24	23	19	12
21	15	18	14	36	94	230	208	187	8	10	10	25	20	12	14
17	21	22	5	48	198	201	208	190	208	13	11	27	18	16	14
6	21	26	6	51	196	199	196	186	211	197	187	28	9	15	15
8	17	21	14	6	194	195	175	183	204	188	180	9	8	20	17
21	18	8	19	23	203	174	180	184	163	189	210	164	9	14	21
23	30	18	231	202	166	197	165	205	185	208	212	22	15	13	11
26	32	46	188	191	187	204	164	237	207	201	200	8	6	24	29
43	54	58	213	196	192	182	227	239	230	215	204	14	37	28	26
45	50	63	201	233	221	230	233	206	234	207	182	21	34	34	25
68	44	43	180	227	203	193	237	230	185	199	166	13	36	35	21
65	30	40	47	222	202	200	211	211	188	199	210	5	41	43	25
25	22	28	11	201	212	212	216	224	191	206	172	7	11	50	27
19	11	7	9	130	229	217	232	222	168	208	5	7	8	55	32
16	12	9	13	20	30	35	24	143	142	71	7	7	3	15	35
16	13	14	16	14	6	16	23	13	34	19	6	4	3	13	16

(c)

Figure 3.3: Overlay block method

(a) 4x4 overlay block method

(b) 5x5 overlay block method

(c) 7x7 overlay block method

3. Procedure of the algorithm

The process of the algorithm through 6 steps (using 4x4 overlay block method as an example).

1. Input a query image and obtain R, G, B components.
2. The image is divided into blocks using block method of overlay 4x4 pixels as shown in figure 3.3(a).

3. Calculate the maximal Eigen value of each block.
4. Get the matrix created with every maximal eigenvalue instead of the pixel block as shown in Figure 3.4.

60.466	438.24	550.02	172.43	73.567
235.91	551.96	750.5	641.94	88.459
344.96	808.71	838.23	635.4	89.506
204.44	729.36	849.78	602.49	106.99
62.104	228.23	499.51	430.51	55.786

Mean=401.98

Figure 3.4: Matrix of maximal eigenvalue

After obtaining the matrix created with maximal eigenvalue, block truncation coding technique is utilized to generate the binary map for image retrieval through 2 easy steps.

5. If a pixel value of the obtained matrix is greater than the mean it is assigned the value "1", otherwise "0".
6. Finally generate the binary map for image retrieval as shown in Figure 3.5.

0	1	1	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	1	1	0

Figure 3.5: Binary map

The procedure of the algorithm is shown in Figure 3.6 below.

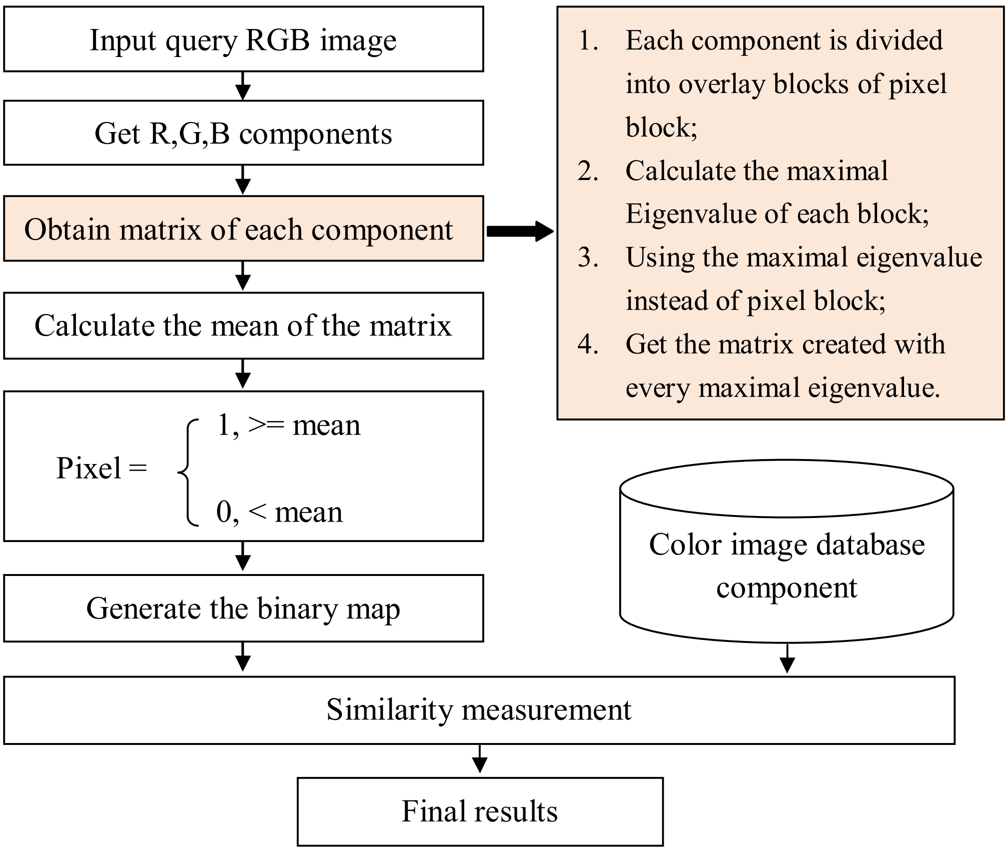


Figure 3.6: Procedure of the proposed algorithm

D. Simulation and results

Content−based image retrieval calculates visual similarities between a query image and images in a database instead of exact matching. Many similarity measures have been developed for image retrieval based on empirical estimates of the distribution of features in recent years. Different similarity/distance measures will affect retrieval performance of an image retrieval system significantly.

In order to evaluate the affect of different similarity measures on the algorithm we used known Euclidean distance similarity measures for histogram matching. Two images A and B can be compared by using the most commonly used Euclidean distance given as:

$$d(A,B) = \sum_{j=1}^{n} \left| H_j^A - H_j^B \right| \tag{3.1}$$

The simulation engine consists of feature extraction process, batch feature extraction and storage process for a collection of images, and the interactive process.

The feature extraction process is based upon the following steps, also shown in Figure 3.7.

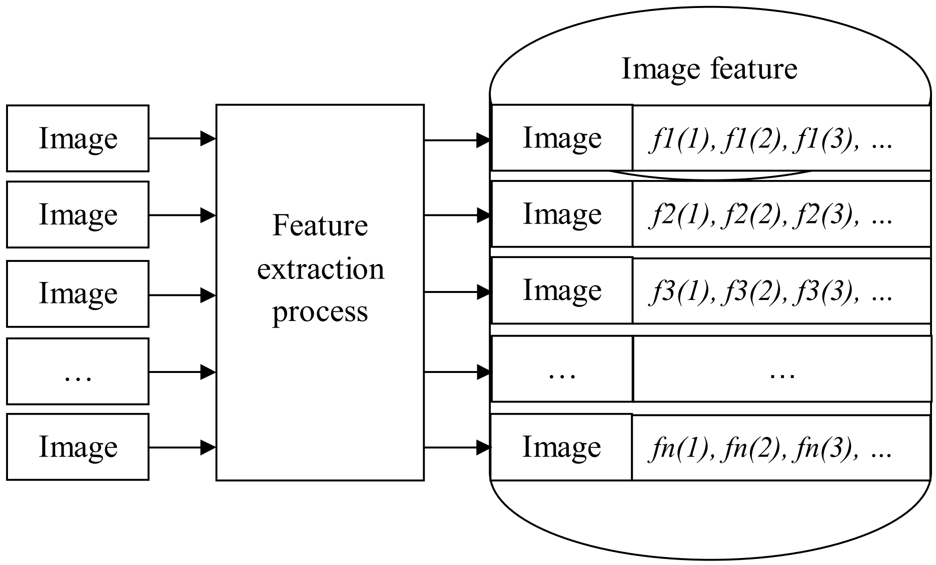


Figure 3.7: Feature extraction and storage process

Figure 3.7 shows the batch feature extraction and storage process as described in the following steps.

a) Images are acquired from a collection one after another.

b) Feature extraction process is applied to them.

c) The resultant vector is saved in a database against the image name under consideration.

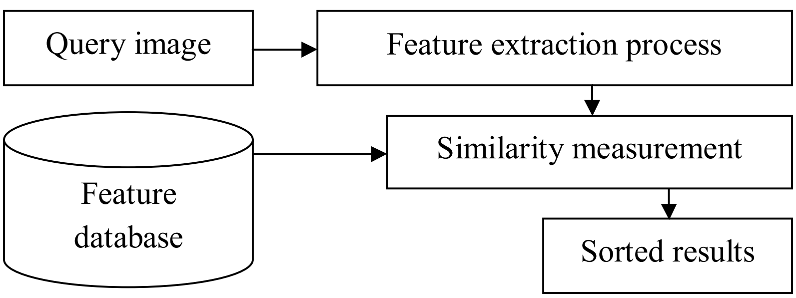


Figure 3.8: Query by image content process

The interactive retrieval process, as shown in Figure 3.8, is carried out in following steps.

a) The program asks the user to select a query image.
b) The feature extraction process runs on the query image and extracts the feature information in the form of a vector.
c) This vector is used to calculate similarity of query image against the feature vectors saved in the feature database of the image collection. Figure 3.9 shows the overview of entire retrieval process.

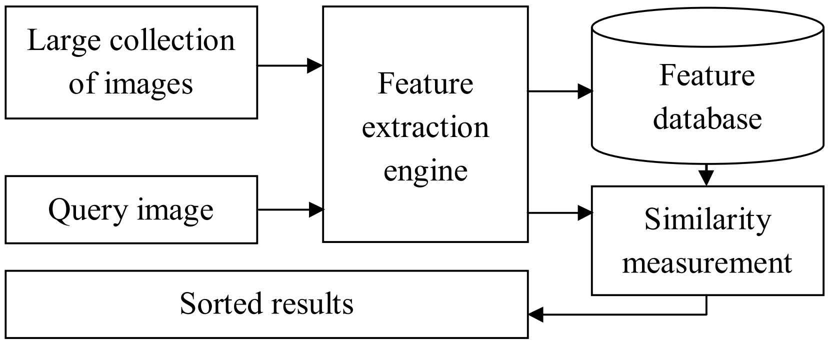


Figure 3.9: Overview of entire retrieval process

The database provided by James S. Wang et al. [22][23] is used for testing the proposed algorithm. The image database is freely available to researchers and is free from any conditions or restrictions.

a) To minimize the dependence on hardware, the entire image collection is stored in main memory during evaluation.
b) The initial number of images in the database is 1000.

c) Images are in JPEG format.

d) 10 different types of objects are available in the database including people, sea, building, bus, dinosaur, elephant, flower, horse, mountain and food as shown in Figure 3.10.

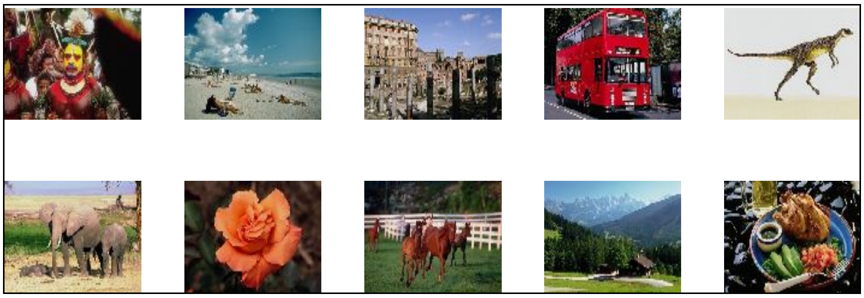
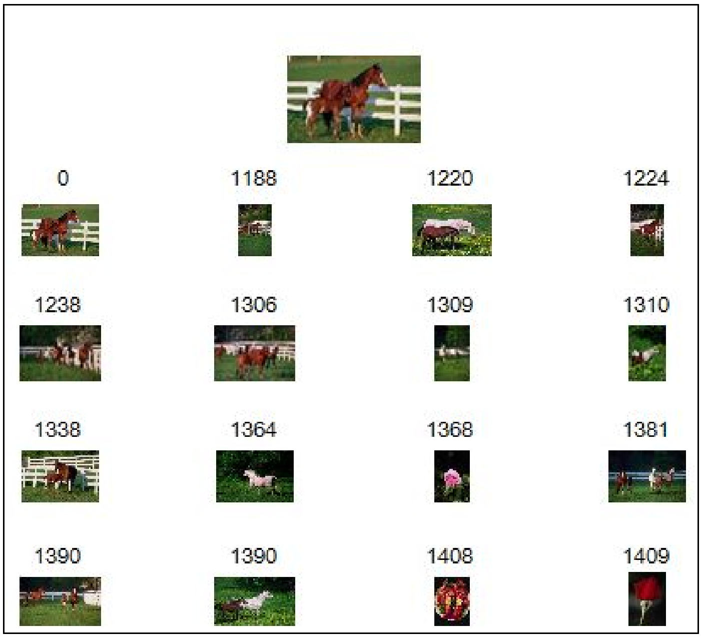


Figure 3.10: Categories of the database

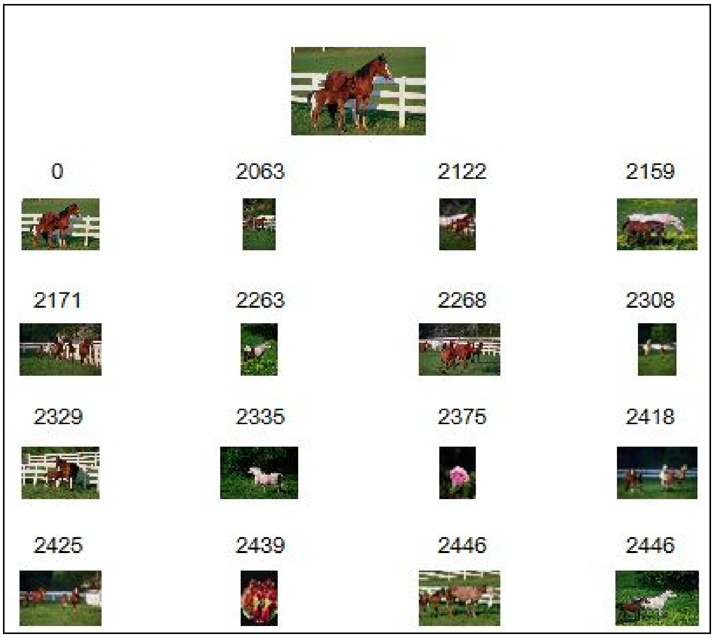
The following figures show some of the simulation results (top 16 results).


0
1188
1220
1224
1238
1306
1309
1310
1338
1364
1368
1381
1390
1390
1408
1409

(a)

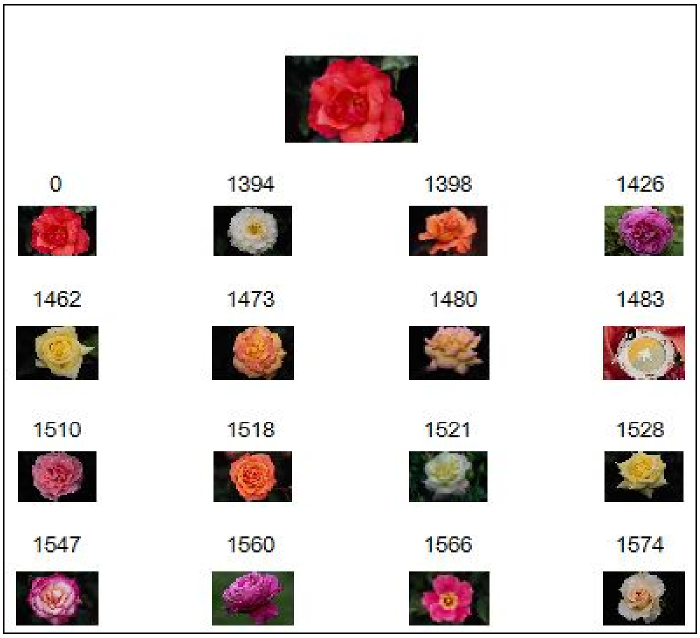

0
1104
1154
1177
1182
1243
1248
1263
1293
1316
1340
1346
1348
1348
1358
1368

(b)



(c)

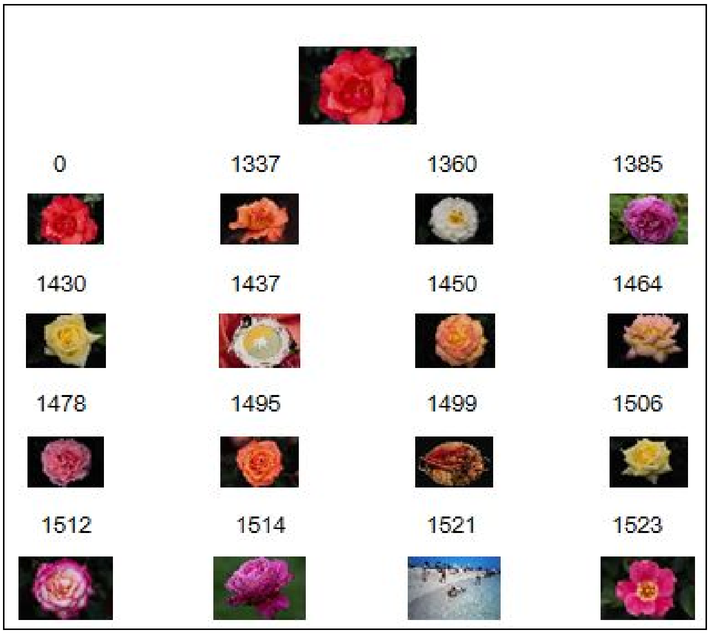
Figure 3.11: Simulation results comparation of animal's case
(a) 4x4 overlay block method
(b) 5x5 overlay block method
(c) 7x7 overlay block method


0
1394
1398
1426
1462
1473
1480
1483
1510
1518
1521
1528
1547
1560
1566
1574

(a)

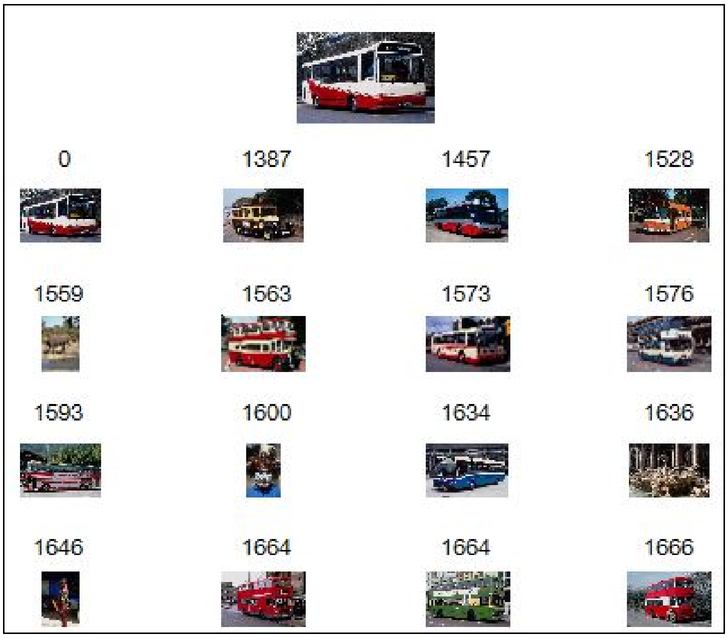

0
2405
2428
2485
2539
2573
2578
2583
2611
2647
2674
2676
2678
2684
2710
2713

(b)



(c)

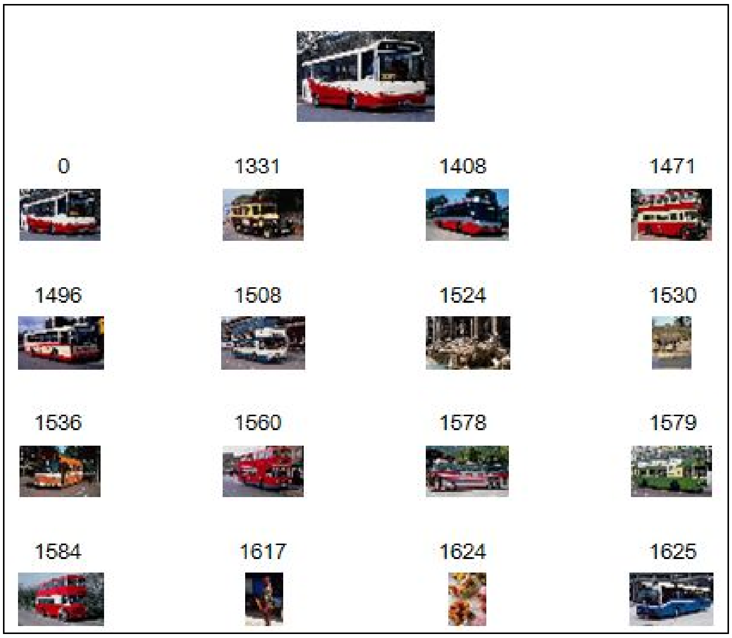
Figure 3.12: Simulation results comparation of flower's case
(a) 4x4 overlay block method
(b) 5x5 overlay block method
(c) 7x7 overlay block method


0
1387
1457
1528
1559
1563
1573
1576
1593
1600
1634
1636
1646
1664
1664
1666

(a)


0
2411
2545
2683
2684
2744
2753
2771
2776
2828
2834
2834
2861
2876
2913
2919

(b)



(c)

Figure 3.13: Simulation results comparation of bus's case
(a) 4x4 overlay block method
(b) 5x5 overlay block method
(c) 7x7 overlay block method

The precision and recall is shown in table below, which carries results of average from different categories of image data.

Table 3.1: Precision comparison of different categories

Top 10 results

	dinosaur	flower	nature	animal	bus
4x4 overlay block	0.99	0.95	0.86	0.80	0.68
5x5 overlay block	1.00	0.98	0.88	0.90	0.74
7x7 overlay block	1.00	0.96	0.87	0.92	0.72

Table 3.2: Precision comparison of different categories

Top 30 results

	dinosaur	flower	nature	animal	bus
4x4 overlay block	0.97	0.91	0.77	0.65	0.60
5x5 overlay block	0.98	0.91	0.80	0.69	0.63
7x7 overlay block	0.98	0.90	0.80	0.71	0.62

From the simulation results it could be noticed that the 5x5 overlay block algorithm has relatively high retrieval accuracy than the other two algorithms. The experimental results also show that the indexes have good ability of characterizing the image contents.

E. Conclusion

This chapter has described a new image feature analysis which caters retrieval of images based on image queries.

The proposed CBIR method using overlay blocks and maximal eigenvalues for feature extraction. First the original image matrix is overlay blocked. And then the algorithm generates feature vectors using overlay parts between adjacent blocks, and generates the binary map using maximal eigenvalues and block truncation technique for image retrieval. It was noticed that the algorithm could reflect the target object information efficiently and confers a precise matching instrument. And 5x5 overlay block is recommended using the proposed algorithm. Because several experiments performed using the 5x5 overlay block method have generate better results compare with 4x4 overlay block method and 7x7 overlay block method. The algorithm is especially suited for searching the images that the objects and background are clearly differentiate. For example: flowers and dinosaurs' case. It was also noticed that the algorithm works well for searching a very large database of images, and the extraction and retrieval process too is very quick.

References

1. Kushki, A., Androutsos, P., Plataniotis, K.N.,and Venetsanopoulos, A.N., “Query feedback for interactive image retrieval”, Circuits and Systems for Video Technology, IEEE Transactions on Volume 14, Issue 5, pp.644 - 655, May. 2004.

2. Wei Jiang, Kap Luk Chan, Mingjing Li, and Hongjiang Zhang, “Mapping low-level features to high-level semantic concepts in region-based image retrieval”, Computer Vision and Pattern Recognition, IEEE Computer Society Conference on Volume 2, pp.244 - 249, Jun. 2005.

3. Ozturk, O., Bulu, H., Alpkocak, A., and Guzelis, C., “Low level feature selection for a content based digital mammography image retrieval system”, Signal Processing and Communications Applications Conference, IEEE 17th, pp.444 - 447, Apr. 2009.

4. Ying Liu, Dengsheng Zhang, Guojun Lu, and Wei-Ying Ma, “Region-Based Image Retrieval with High-Level Semantic Color Names”, Multimedia Modelling Conference, Proceedings of the 11th International, pp.180 - 187, Jan. 2005.

5. Rasheed, W., Youngeun An; Sungbum Pan, Ilhoe Jeong, Jongan Park, and Jinsuk Kang, “Image Retrieval Using Maximum Frequency of Local Histogram Based Color Correlogram”, Modeling & Simulation, AICMS 08, Second Asia International Conference, pp.322 - 326, May. 2008.

6. Jongan Park, Youngeun An, Ilhoe Jeong, Gwangwon Kang, and Pankoo, K., “Image Indexing using Spatial Multi-Resolution Color Correlogram”, Imaging Systems and Techniques, 2007. IST '07, IEEE International Workshop, pp.1 - 4, May. 2007.

7. Rong Zhao and Grosky, W.I., “Narrowing the semantic gap – improved text-based web document retrieval using visual features”, Multimedia, IEEE Transactions on Volume 4, Issue 2, pp.189 - 200, Jun. 2002.

8. Kokare, M., Biswas, P.K., and Chatterji, B.N., “Texture image retrieval using new rotated complex wavelet filters”, Systems, Man, and Cybernetics, Part B, IEEE Transactions on Volume 35, Issue 6, pp.1168 - 1178, Dec. 2005.

9. A. D. Bimbo, “Visual Information Retrieval”, Morgan Kaufmann Publishers, 1999.

10. R. C. Gonzalez and R. E. Woods, “Digital Image Processing”, Addison Wesley Publishing Company, 1993.
11. Y. Rui, A. She, and T. Huang, “Modified fourier descriptors for shape representation – a practical approach”, in Proceedings of the First International Workshop on Image Databases and Multi Media Search, pp.22 - 23, Aug. 1996.

12. Vermilyer, R., “Intelligent User Interface Agents in Content–Based Image Retrieval”, SoutheastCon, Proceedings of the IEEE, pp.136 - 142, Apr. 2005.

13. Sano, A., Tsuji, H., and Nagasawa, K., “Bayesian information theoretic criterion for detection of number of signals in array processing”, Spectrum Estimation and Modeling, Fifth ASSP Workshop, pp.129 - 133, Oct. 1990.

14. R. C. Gonzalez and R. E. Woods, “Digital Image Processing”, Addison–Wesley, Reading, MA, 1992.

15. Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Qian Huang, Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D., and Yanker, P., “Query by image and video content: the QBIC system”, Computer Volume 28, Issue 9, pp.23 - 32, Sep. 1995.

16. Ogle, V.E. and Stonebraker, M., “Chabot: retrieval from a relational database of images”, Computer Volume 28, Issue 9, pp.40 – 48, Sep. 1995.

17. Stonebraker, M., Rowe, L.A., and Hirohama, M., “The implementation of POSTGRES”, Knowledge and Data Engineering, IEEE Transactions on Volume 2, Issue 1, pp.125 – 142, Mar. 1990.

18. Prakash, R., Burkhart, P.D., Chen, A., Comeaux, K.A., Guernsey, C.S., Kipp, D.M., Lorenzoni, L.V., Mendeck, G.F., Powell, R.W., Rivellini, T.P., San Martin, A.M., Sell, S.W., Steltzner, A.D., and Way, D.W., “Mars Science Laboratory Entry, Descent, and Landing System Overview”, Aerospace Conference, 2008 IEEE, pp.1 – 18, Mar. 2008.

19. Gwangwon Kang, Junguk Beak, and Jongan Park, “Features Defined by Median Filtering on RGB Segments for Image Retrieval”, Computer Modeling and Simulation, EMS '08, Second UKSIM European Symposium, pp.436 – 440, Sep. 2008.

20. Canny, J., “A Computational Approach To Edge Detection”, IEEE Trans, Pattern Analysis and Machine Intelligence, pp.679—714, 1986.

21. Yeganeh, H., Ziaei, A., and Rezaie, A., “A novel approach for contrast enhancement based on Histogram Equalization”, Computer and

Communication Engineering, ICCCE 2008, International Conference, pp.256 - 260, May. 2008.

22.J. Li and J. Z. Wang, "Automatic Linguistic Indexing of Pictures by a statistical modeling approach", Pattern Analysis and Machine Intelligence, IEEE Transactions on volume 25, pp.1075-1088, 2003.

23.J. Z. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries", Advances in Visual Information Systems: 4th International Conference, VISUAL 2000, Nov. 2000.

24.Kai-Kuang Ma, Lei Huang, Shan Zhu, and Ho, A.T.S., "SPOT image compression using block truncation coding techniques", Geoscience and Remote Sensing, IGARSS '97, Remote Sensing - A Scientific Vision for Sustainable Development., 1997 IEEE International, volume 4, pp.1996 - 1998, Aug. 1997.