Islamic University – Gaza

Deanery of Higher Studies

Faculty of Engineering

Computer Engineering Department

Improved Image Retrieval with Color and Angle Representation

استرجاع صورة محسّن باستخدام تمثيل لوني وزاوي

Hadi A. J. Alnabriss

Supervisor

Prof. Ibrahim S. I. Abuhaiba

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in Computer Engineering

1434H (2013)

بسم الله الرحمن الرحيم

الجامعة الإسلامية – غزة
The Islamic University - Gaza

مكتب نائب الرئيس للبحث العلمي والدراسات العليا هاتف داخلي 1150

الرقم Ref: ج س غ/35/

التاريخ Date: 2013/10/06

نتيجة الحكم على أطروحة ماجستير

بناءً على موافقة عمادة الدراسات العليا بالجامعة الإسلامية بغزة على تشكيل لجنة الحكم على أطروحة الباحث/ **هادي عبد الكريم جاد الله النبريص** لنيل درجة الماجستير في **كلية *الهندسة*** قسم **هندسة الحاسوب** وموضوعها:

استرجاع صورة محسن بتمثيل لوني وزاوي

Improved Image Retrieval with Color and Angle Representation

وبعد المناقشة التي تمت اليوم الأحد 01 ذو الحجة 1434هـ، الموافق 2013/10/06م الساعة الثانية مساءً، اجتمعت لجنة الحكم على الأطروحة والمكونة من:

أ.د. إبراهيم سليمان أبو هيبة	مشرفاً ورئيساً	....
د. محمد أحمد الحنجوري	مناقشاً داخلياً	....
د. إيهاب صلاح زقوت	مناقشاً خارجيًا	....

وبعد المداولة أوصت اللجنة بمنح الباحث درجة الماجستير في **كلية *الهندسة*** / قسم **هندسة الحاسوب.**

واللجنة إذ تمنحه هذه الدرجة فإنها توصيه بتقوى الله ولزوم طاعته وأن يسخر علمه في خدمة دينه ووطنه.

والله ولي التوفيق ،،،

مساعد نائب الرئيس للبحث العلمي وللدراسات العليا

أ.د. فؤاد علي العاجز

P.O. Box 108, Rimal, Gaza, Palestine fax: +970 (8) 286 0800 فاكس Tel: +970 (8) 286 0700 ص.ب. 108 الرمال، غزة، فلسطين هاتف
public@iugaza.edu.ps www.iugaza.edu.ps

ACKNOWLEDGMENT

At first of all I thank Allah for helping me to fulfill my ambition and to complete this research after three continuous years of study and search. And my special gratitude to my supervisor, Professor Ibrahim Abuhaiba, for his advice, patience, support, motivation and guidance to conduct my research, in addition to his rule in broadening my understanding for his courses during the period of my education in the masters program, which had a profound effect on the selection of this topic for my thesis.

Furthermore I would like to acknowledge with much appreciation the crucial role of the discussion committee and my friends for their help and assistance to complete this research.

Also, I would like to thank my family: my wife and my parents for their endless love and support from the very first day of my education in this program.

Table Of Contents

List of Figures

List of Tables

List of Abbreviations

CAH: Color and Angle Histogram

CBIR: Content Based Image Retrieval

CCH: Conventional Color Histogram

CCV: Color Coherence Vector

FCH: Fuzzy Color Histogram

HSB: Hue Saturation Brightness

HIS: Hue Saturation Intensity

HSL: Hue Saturation Lightness

HSV: Hue, Saturation, Value

IRM: Integrated Region Matching

LBP: Local Binary Pattern

MPEG-7: Moving Picture Experts Group

MRF: Markov Random Field

MSD: Micro-Structure Descriptor

NUCQ: Non-Uniform Color Quantization

QBIC: Query by Image Content

RGB: Red, Green, Blue

RoI: Region of Interest

SOM: Self-Organizing Map

TBIR : Text Based Image Retrieval

UCQ: Uniform Color Quantization

استرجاع صورة محسّن باستخدام تمثيل لوني وزاوي

هادي عبد الكريم النبريص

ملخص

في هذا البحث تم استخدام أفكار جديدة لتحسين عملية البحث عن الصور الملونة من خلال محتواها عن طريق تمثيلها باستخدام اللون والزاوية لكل نقطة في الصورة، بحيث يتم استخدام الرمز التمثيلي الناتج في قياس التشابه بين صورتين.

وتعتبر عملية تكميم اللون أو تقليل عدد الألوان في الصورة من العمليات المهمة في مراحل تحويل الصورة إلى رمز يمثلها، حيث تم عمل مجموعة مقارنات بين تقنيات تقليل الألوان المنتظمة والغير منتظمة، ومن ثم تم استخدام الأنظمة الغير منتظمة والتي أظهرت كفاءة أعلى وفقا للنتائج التي حصلنا عليها.

لقياس التشابه بين صورتين من خلال الرموز التمثيلية لكل منهما تم اقتراح طريقة جديدة لقياس المسافة –أو التشابه- وتم عمل مقارنات مع المسافة الاقليدية وتبين أن الطريقة المقترحة هي طريقة منافسة وقادرة على التفوق على الطرق الأخرى بسبب اعتمادها على القيم في داخل الرمز التمثيلي. كما تم عمل مقارنات بين الطرق المبتكرة في البحث وطرق منشورة في أبحاث أخرى وكانت النتائج أفضل في معظم أنواع الصور، كما تم استخدام تقنيات تسمح باكتشاف التشابه بين الصورة الأصلية ونسخة أخرى منها بعد تدويرها بدرجات مختلفة.

وسوف يمثل كل الصور المتوفرة رمز مكون من 296 قيمة لكل صورة، مما يجعل المساحة المستخدمة لتمثيل الصور صغيرة مقارنة مع حجم الصورة، كما يمكن ضغط الرمز التمثيلي باستخدام التقنيات التقليدية للضغط، كما ستكون سرعة تحويل قاعدة بيانات الصور إلى رموز تمثيلية غير مؤثرة لأنها ستتم في الخلفية ودون شعور المستخدم.

لفحص الطريقة المقترحة تم استخدام قاعدة بيانات Corel-1000 الخاصة بالصور، كما تم عمل كود محاكاة باستخدام برنامج Matlab ، وتم عمل مقارنة مع مجموعة أخرى من الطرق المستخدمة لإرجاع الصور مثل (IRM, Fuzzy Club ,ERBIR, , Geometric Histogram، استرجاع الصورة باستخدام التوقيع أو البصمة)، وكانت النتائج إيجابية في كل الحالات تقريبا.

Improved Image Retrieval with Color and Angle Representation

Hadi A. Alnabriss

ABSTRACT

In this research, new ideas are proposed to enhance content-based image retrieval applications by representing colored images in terms of its colors and angles as a histogram describing the number of pixels with particular color located in specific angle, then similarity is measured between the two represented histograms. The color quantization technique is a crucial stage in the CBIR system process, we made comparisons between the uniform and the non-uniform color quantization techniques, then according to our results we used the non-uniform technique which showed higher efficiency.

To find the similarity between two signatures we made comparisons between the Euclidean distance and a new proposed similarity measurement, our proposed similarity showed better results and higher precision ratios. Also specific techniques has been used for making the represented histogram rotation-tolerant. The results showed promising performance compared to many other techniques.

Every image in the database will be represented as a vector of 296 values, which makes the used space for representing images not significant and able to be compressed using traditional techniques of compression. The speed of converting the whole database does not matter because it will be done in the background.

In our tests we used the Corel-1000 images database in addition to a Matlab code, and we made comparisons with other approaches like Fuzzy Club, IRM, Geometric Histogram, Signature Based CBIR and Modified ERBIR, our proposed technique showed high retrieving precision ratios compared to the other techniques.

Keywords: Content Based Image Retrieval, Image Processing, Color and Angle Representation, Non-Uniform Color Quantization.

Chapter 1

INTRODUCTION

1.1Image Retrieval

The development of information technology, and the increasing number of Internet users in addition to the spreading of social networks and video/image hosting websites, have a great influence on the tremendous growing of the digital media and image collections, a huge number of photos and videos are taken every day using digital cameras and mobiles to be uploaded later on the Internet.

These huge collections of multimedia are almost useless if we do not have the sufficient technique to retrieve specific data from this database depending on our information need, and the strength of retrieving depends on the power of measuring the distance or similarity between two objects (images in our case).

In our rapidly-developing world, image retrieval is involved in many important applications like image searching and image indexing used in search engines for retrieving similar images either from the internet or from local databases in a library or a hotel or for police and intelligence uses.

Also image retrieval is used in medical diagnosis applications, that depends on finding similar images (or parts within the image) to discover certain diseases and vulnerabilities. Photograph archives' applications also depend on retrieving similar images to be able to organize your archives and albums. Duplicate-image detection applications are also very important, these applications are used to discover if an intellectual property is abused, or used by someone else without citation.

Other applications like Architectural and engineering design, geographical information and remote sensing systems, object or scene recognition, solving for 3D structure from multiple images, motion tracking and crime prevention, all these applications depend on the ability of finding similar images within a collection.

The importance of the abovementioned applications has added more demand to develop a new branch of information retrieval called the “image retrieval” to enhance finding similarity among images using some features within the image. The size of the available huge collections of images adds more complexities and more obstacles for the image retrieval approaches, and subsequently many approaches have been emerged to tackle this problem of searching, retrieving and indexing images, but the main two approaches are Text Based Image Retrieval (TBIR), and Content Based Image Retrieval (CBIR).

In TBIR the query is text that describes the image, in Figure 1.1, TBIR is used from the Google's Search Engine (which supports TBIR and CBIR), we can see that the retrieved images for the query “apple” are the well-known Apple Company and the apple fruit, because TBIR systems consider these two things as similar because of their names similarity.

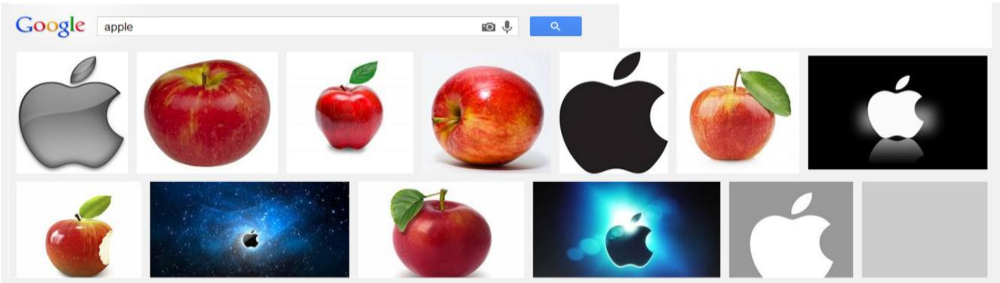


Figure 1.1: TBIR for the query “apple”

In Figure 1.2, a CBIR system has been used, in this example an apple’s image was used, the retrieved images are all apples, here there is no confusion with other logos or signs, and all the retrieved images are for what we want “apple”.

Figure 1.2: CBIR system used for the query image shown in the blue frame

From the results shown in Figures 1.1 and 1.2 we can see that TBIR is useful when we do not know the nature or the figure of our query (if we ignored the annotations problem), for example if we do not know what is the apple fruit, but the CBIR systems are very useful when we have the query image, and it will be very useful for detecting if any new similar images have been added to the database.

1.1.1 Text-Based Image Retrieval

TBIR is the common and traditional approach; it is used in many search engines like Baidu, and Google. In this approach, images are annotated using keywords such as captioning, metadata, or descriptions added to the image database either manually by annotators or automatically by detecting tags and keywords around the image in the web page or the document; the first way is time consuming and needs great staff to annotate each image, while the second way is not guaranteed because it depends on any tag or word within the web page or the document, and so the similarity measurement will not be accurate, and it is highly expected to be irrelevant to the query. For example searching for the name of specific country will retrieve the flag, the map and may be the president as three similar or identical images for our information need.

The TBIR system works on reading documents and web pages to find images, then, tags and annotations and keywords are extracted from the processed document to build a bag-of-words for this image, this process can be done in the offline mode, and the result will be a database with two main columns: the image ID and the bag of words describing the image. Figure 1.3 illustrates the main process of the TBIR system, however, real systems add many other techniques like compression, users feedback modules and other processing steps.

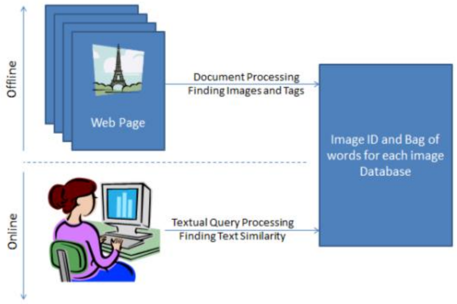


Figure 1.3: Illustration of simple TBIR system process

1.1.2 Content-Based Image Retrieval

The other approach is Content Based Image Retrieval (CBIR), which has been found in 1981 by Moravec [1], and some references consider the beginning of this field in 1992 by Kato [2]. The query in the CBIR is an image which used to retrieve images based on specific features like color, texture and shape, the term has been widely propagated to solve the problem of Text-Based Image Retrieval by retrieving images depending on image features instead of surrounding text or annotations, and because of its nature and power many researchers found that CBIR much sufficient for many applications more than the text based approaches [3-5], and because of its power many available search engines like Google, Yahoo and AltaVista have added CBIR systems to their engines.

The main process in the CBIR (illustrated in Figure 1.4) depends on reading the images' database or the collection, each image is converted from its original form into a signature that describes specific features in the image, this process is applied in the background of the server or offline and it does not affect the speed of the retrieving function, in the end of this process we will generate a signature for every image, this signature describes the image, and it has to be quite enough representative to be used in measuring similarity between two images.

The second part of the CBIR process is applied online, in this stage the client uploads his image to the CBIR system, this image is converted to a signature (in the same matter of the first stage), then this signature has to be compared to each one in our signatures' database, then images are ranked and retrieved according to their similarity ratios.

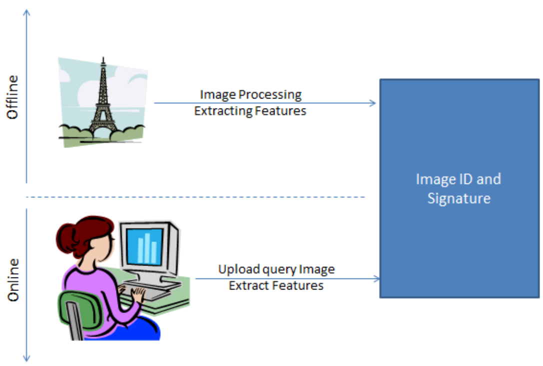


Figure 1.4: CBIR system process illustration

The key challenge in CBIR is to represent the image as a value or series of values (the signature) that describe the image using its features, and then to measure the similarity between two images according to the generated signatures, the similarity measurement and enhancement depends on the type of features extracted from the image in addition to the technique used to measure the similarity. So, the core work of CBIR research is to find the right way to extract specific and important features to represent the image as an object or signature composed of specific number of values, and then to use an adapted algorithm to measure the distance or similarity between two objects.

To measure the similarity many references used the color features, In [6, 7], similarity measurement in these references depends on the difference between the histogram of each image. Other researches divided the image into many regions according to the color of each region, then the color features or histogram is extracted from each region.

In the CBIR algorithm many factors must be taken seriously in our similarity measurement, these factors like the size of the new representation for the image, and the ability for compression, the time used to convert the image from its original form to the measurable data (the conversion of the whole database will be applied in the background as an offline process, but the conversion of the query image will be done online in the foreground), the ability to cluster these new representations (to increase the search speed), the ability to find the similarity between rotated or resized images, all these factors must be highly considered to build a powerful approach for image retrieval system.

1.2 CBIR Problems

The efficiency of a CBIR system depends on many factors like the approach used to represent the image, the color quantization technique, the similarity measurement equations, and other factors. Some CBIR systems depend on the histogram of the image to find the similarity between two images, the problem with histogram approaches is obvious, because many images can be similar in the histogram but for the human visual system they are irrelevant, so the similarity of two histograms does not mean that the two images are similar, however these approaches showed some success in some types of images, but it has failed against many other types.

Figure 1.5 shows two different images, and we can see from the distribution of the colors that the two images have identical histograms, but actually they look totally different.

The shape feature depends on segmenting the image into different regions, and then the main shape of each region is detected [8], these techniques face many problems in the approaches used to detect objects within the image, and the potential huge number of objects within the image is another challenge. Other methods identify shapes in the image using shape filters, but detection will require human intervention because segmentation methods like [8, 9] are very difficult to completely automate [10].

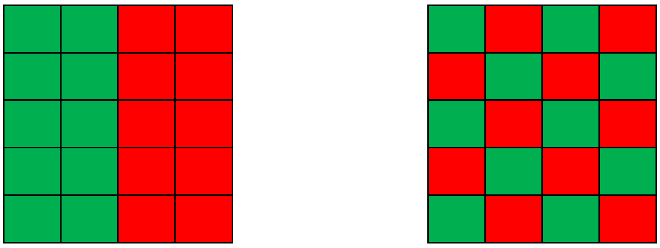


Figure 1.5: Two identical histograms for two different images

The texture is another interesting feature to find similarity, it depends on the visual patterns for the similarity measurement in addition to the spatial distribution of these textures within the image [11], and even though texture depends on many features such as direction, periodicity, and coarseness, but it is very effective for the human visual system. Many researches in CBIR used the texture approach for representing images and measuring similarity. Texture is considered as a prominent and powerful feature for content based image retrieval [12], and many previous researches such as QBIC[13], NETRA [14], Photobook [15], and Visual Seek[16], used texture as the main feature for representing image, but the main problem in the texture feature is its weakness against scale, rotation and translation, because the texture features are potential subjects for change. Other approaches used the combination of the color and texture features to get a much more powerful similarity measurement such as the multi-texton histogram[17], integrative co-occurrence matrix[18], texton co-occurrences matrix [19] and color auto-correlograms [20].

One of the intelligent and powerful approaches was introduced in [21], it depends on the texture elements in the image using specific types of structure elements that can detect the number of specific patterns within the image, then a structure element histogram is used to represent the image, which is a development for another approach[22] that used a novel image representation idea called micro-structure descriptor (MSD) , the MSD approach is built based on the underlying colors in micro-structures with similar edge orientation.

The problem in the approaches presented in [21, 22] is their disability if any change is applied to the image like rotation, scale, or even one-bit translation. To address this problem, the Color and Angle Histogram (CAH) will be used to represent the image in this research.

1.3 Work and Motivation

The image retrieval systems development is still a challenging field of research, because of its nature, and images are considered much more difficult to be evaluated for similarity than textual information. The main challenges that face researchers in this field are:

1. The type and kind of features and information to be extracted from the image.
2. The time consumption used to measure the similarity between two images, this time depends on the speed of finding the signature of an image, and then measuring the distance between two signatures.
3. The mathematical algorithm used to measure the distance between two signatures.
4. The size of the signatures database, because this affects the search speed, in addition to the ability to cluster these signatures.
5. The ability to measure similarity among rotated, translated and resized images.
6. The Color Quantization Problem.

In this research many ideas have been proposed to tackle each problem of the above mentioned problems, the novel idea proposed in this research is to represent an image as a histogram for the pixel's color and angle according to a central reference point.

The proposed idea is expected to be very powerful in detecting specific textures in the image. The image will be quantized in the HSV space into a number of colors, each color will be divided into subdivisions representing the angle, then the number of pixels with specific color and specific angel will be used to represent the Color/Angle Histogram CAH. The proposed solution is expected to overcome the approaches presented in [21, 22] in detecting underlying textures and colors, and to be tolerant for certain rotations with incremental degrees, and to be scale-invariant too. Also, it will be much more powerful than the approach introduced in [23] which considers that each image has only one object or shape and a background.

Depending on the color and the angle, we will represent the image as if we know the length and the angle of all the lines with specific color and in the same direction from the reference point, this will be very powerful because it depends on the color features and the texture features at the same time, and depending on the angle will make the detection of rotation very simple.

1.4 Thesis Contribution

In this research we are going to work on solving each problem of the problems mentioned in section 1.3, of course we have a tradeoff here, we might fix some problems but others will occur, so we have to try to solve the problems without affecting any other factors.

One of the main steps in the CBIR process is to be able to represent a colored image as a collection of values that describe your image, these values in the next step have to be used to measure the similarity between two images, then your CBIR system has to be able to retrieve relevant images from the database according to your calculations.

In this thesis we begin our CBIR system process by quantizing an image using the non-uniform color quantization, which depends on human visual system perceiving fundamentals to quantize the image into 37 colors. In the second step we convert an image into a novel representation called the color and angle histogram CAH, this CAH divides an image into different regions or angles (like the pizza slices), then statistics for each color within each slice or angle are collected.

After building the CAH a new similarity measurement technique is used to measure the similarity between two CAHs, the new technique depends on making a measured value in the range [-∞, +∞], because some of the measured cases between two values (one from each CAH for two images) are considered negative, the similarities are converted to be in the period [0, 1] after retrieving the images.

The advantages of using the CAH is to be able to overcome the problem of depending on the color histogram, which retrieves irrelevant images because of similarity in the histograms. Another advantage of dividing the image into angles will be the ability to find an image even if rotated with specific increments of the division angle.

The huge number of CAHs for our images' database can be also clustered as any data vectors, the ability to cluster the CAH database will increase the speed of the retrieving process.

1.5 Organization of This Thesis

Image retrieval history and different approaches used in this field, particularly approaches that depend on extracting specific features or objects from the image, will be discussed in Chapter 2. In Chapter 3, we will discuss the background used and needed for traditional CBIR systems and the approaches used to extract information from the image, in this chapter we will discuss extracting features like color, texture and shape from the image, each topic will be discussed apart. In Chapter 3 we will also talk about the concept of converting an image into a signature and the approaches used to find the similarity between two images via calculations on their signatures.

In Chapter 3 we will talk about important concepts about the color, texture and shape, like the different types of color systems, different textures types, shapes concepts and different fields that depend on these approaches.

In Chapter 4 we discuss our proposed approach, we will talk about the importance of the color quantization in CBIR systems, and we will illustrate the non-uniform color quantization, then we will discuss our approach to convert an image to a signature using the proposed color and angle histogram. Furthermore we will show our proposed similarity measurement to find the distance or similarity between two signatures. Also in Chapter 4 we will discuss proposed solutions to make image retrieval rotation and scale invariant or tolerant.

In Chapter 5 we will compare the results when we use the uniform and non-uniform color quantization techniques, we will also use our proposed similarity measurements in addition to the Euclidean distance, then we will select the best CBIR approach to be compared with other approaches in other researches. In Chapter 5 we will also test the impact of rotation, translation, resizing and adding more noise, on our system.

In the last chapter we will make a conclusion for our research, we will also suggest the future work ideas that can be done to improve our work.

Chapter 2

RELATED WORK

The growing size of the images collections has motivated the research in the Image Retrieval field since the beginning of the 1990s. The main motivation is to provide an effective access to this huge database of digital images, one of the early projects was the IBM's system for query by image content in 1995 [24], this system was prepared to annotate images according to visual features like color, texture and shape. The use of extracted features showed good power in finding similarity among images, this success attracted many researchers to develop more advanced ideas to extract images' features and to depend on more combinations of these features.

Many works have tested a variety of features for retrieving images in large collections of digital images. Other researchers applied researches on intensity features [25], spectral (color) features [26, 27], shape features [28, 29], structural features [31, 32], texture features [29, 30, 33, 34], and combinations thereof such as multispectral texture features [35].

The following sections divide the previous researches in the CBIR approaches into three fields: the CBIR approach that depends on the image's features to measure similarity, the approach that depends on detecting shapes and objects within the image, the approach that depends on statistical descriptors for the image.

2.1 CBIR via Features Extraction

The development of image retrieval using a set of features, or a group of points of interest, began in 1981, when Moravec [1] used a corner detector on stereo matching, this detector then was improved in [36] by Harris and Stephens to make it more sensitive to edges and points of interest in the image. In 1995, Zhang et al. [37] used the correlation window around the four corners of the image to select matches over larger collections, and then the fundamental-matrix for describing the geometric constraints has been proposed for removing outliers in the same research too.

Schmid and Mohr [38] started for the first time to use invariant local features matching for recognizing images. A feature was checked against a large database, their work was a development for [36], they used a rotationally invariant descriptor of the local region instead of the correlation window. In their research, they could make image comparisons invariant to orientation change between two images, they also proved that multiple feature matches could make better recognition under distortion and clutter.

Harris corners were very sensitive for change in scale. Lowe, in [39], enhanced the Harris corners approach to be scale invariant. Lowe used a new local descriptor also to make the approach less sensitive to local image distortions, such as 3D viewpoint change.

In recent works, the color features started to be the dominant for researches in CBIR. Because the CBIR systems became more interested in the human visual system perception, and not interested in other factors. The color layout descriptor and scalable color descriptor were used in the Moving Picture Experts Group MPEG-7 standard, but the main problem for color histogram and other color features is the disability to characterize image spatial characteristics, or color distribution, so many researches started to use other features like texture and shape in combination with color, because texture describes smoothness, coarseness and regularity of the image, which can be a good descriptor.

Combining many features and factors with the color feature explains two points, the first is the importance of the color feature for the human eye, the second point is the weakness of color histograms when there is no spatial-dependent features are collected.

Because texture features are more descriptive for the real world images, many researches were making more focus on texture, like Markov Random Field (MRF) model [40], Gabor filtering[41], and Local Binary Pattern (LBP) [42].

But the main problem with texture features is its weakness for any modifications applied on the image, like resizing, distortion or any type of noise. Texture based features, also, are not enough to measure the similarity between two images, because the smoothness or coarseness for two images doesnot mean that they are similar.

Another problem in the approaches that depend on the texture features is its complexity and its time-consuming process.

2.2 Objects Extraction

The Gabor filters [41] has been used in [43] to segment images into distinct regions, then features are extracted from each region separately instead of the whole image, then the Self-Organizing Map (SOM) clustering algorithm is used to group similar regions in the same class, in the same manner the query image is clustered and each region is compared with each centroid to find the region's class, the class of each region is used later to find the similarity with other images composed of similar regions.

The problem in this approach is its log increasing function when the number of objects in the image increases, another problem is the similarity measurement approach that can be used to measure the similarity between two images, each with different number of objects.

2.3 Similarity using Descriptors Statistics

The MPEG-7 standard adopted three texture descriptors: homogeneous texture descriptor, texture browsing descriptor and the edge histogram descriptor [44].

In the texture features a new term, texton, was introduced in [45] that depends on the texture features. In [46], the authors proved that the human visual system is sensitive to global topological properties.

In [22], the Micro Structure Descriptors (MSD) were introduced. They depend on the underlying colors with similar edge orientation. The MSD extracts features and integrates colors, texture and shape information to represent an image for the retrieval process, the extraction process depends on structure elements shown in Figure 2.1.

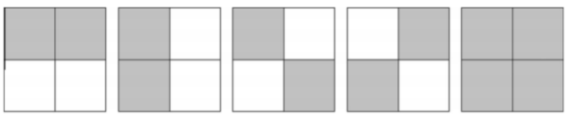


Figure 2.1: Structure elements used to detect orientations in the image

In [22], statistics are gathered for the existence of every structure element in the image, these statistics are used later to build the signature and to measure the similarity between two images.

Two years later after [22], another approach was developed [21]. The new approach replaced the MSDs and used structuring elements histograms; the second approach showed good results compared to the first one with MSD.

The structure element descriptor in [21] is used to find the structuring elements histogram which represents the image's texture. The results in [21] are very promising and it deserves more work and development, but the main problem was its disability to discover similarities if translation, rotation or scale is applied to the image (which is a big problem).

In addition to its disability to find similarity if the image was subject to any modification or noise, this approach is time consuming (the author did not focus on this issue), because the system proposed in [22] depends on applying each structure element for each quantized color on each pixel in the image, so if we have 5 structuring elements, and 20 quantized color and a 200×200 pixels image, then we need to make a 5×20×200×200 operation before finding the structure elements histogram.

Most of the new researches pay more attention to approaches that include the color features to adapt to the human visual system, the color histogram is an old approach but it still powerful, some additions and enhancements (or adaptations) on the color histogram approaches can result in building a powerful CBIR system.

In [47] the researchers used the spatial relationship Fuzzy Color Histogram (FCH), by considering the color similarity of each pixel's color associated to all the histogram bins through fuzzy-set membership function. Instead of the Conventional Color Histogram (CCH), which assigns each pixel into one of the bins only, FCH spreads each pixel's total membership value to all the histogram bins

Another approach in [48] used the Color Coherence Vector (CCV), which partitions pixels based upon their spatial coherence. A coherent pixel is part of some sizable contiguous region, while an incoherent pixel is not. So two images with similar color histograms, will have different CCVs, if they were actually different.

In this research we will focus on fixing the weakness axis in the approaches that depend on the color histogram. So, a novel approach will be used in this research, the main idea is to use a color/angle histogram to represent the image; this histogram is expected to be very descriptive for the image and to be scale invariant and rotation tolerant.

The approach will use a reference point in the center of the image for measuring the angle between the reference point and each point in the image.

Chapter 3

BACKGROUND

Content Based Image retrieval (CBIR), is an active research area in the field of image retrieval, this field has been developed rapidly to adapt to the rapid spreading and increasing in the size of image databases and collections, and to make these available resources of information useful by adding the ability to find similar images for the client's query.

CBIR is the art of extracting the useful information from the image by analyzing its content to be able to find similar and relevant images later, the extracted information must conform to human perception of visual semantics, because the user is interested in finding the most relevant and similar images for his information need or query.

3.1 Features Extraction

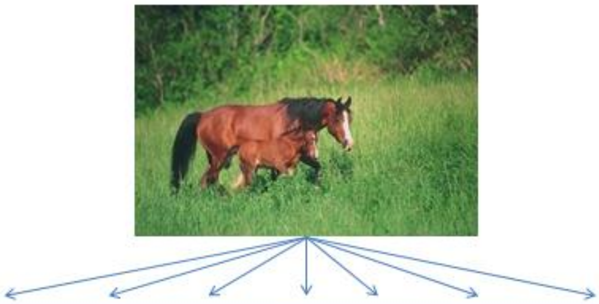
The first stage of a CBIR system is to extract visual features from the images database, each image has to be converted to a signature that describes the image, this stage is usually applied in the offline mode to generate a new database of signatures, each image is connected to one signature at least. The most common features used in this field are the color, texture and shape, CBIR approaches can use separate features or a combination of more than one feature to represent an image.

Figure 3.1 shows an image, the image is converted to a series of descriptor values or a signature that describes the image, so the existence of another image with the highest number of signature matching values means high similarity between the two images.

For example if we found two colors with the same number of main colors, the same texture value, the same percent of background and the same number of objects, this increases the probability of having two similar images.

3.1.1 Color

Color is the most feature employed in representing image, the main idea is the impact of the color on the human visual system perception, which makes the color factor a very important factor in the image. The color is a very powerful descriptor for the image, it is simple and very fast in execution time terms compared with other features like shape and texture which need complex computations for its process.



Main Colors	# of Objects	Background	Texture
4	2	70%	Regular, Smooth

Figure 3.1: Some features are extracted to describe the image

The color feature is a statistical operation for pixels and colors in the image (Figure 3.2 illustrates the color histogram for two similar images), this operation will gather information like the number of specific colors in the image, but the color in the computer vision system is usually expressed as a compound of some variables, these variables depend on the color system used, but however the number of all the colors in any system will count up to millions of colors, this makes the color feature more complex in computations, so color feature CBIR systems use a preprocessing step called color quantization, the rule of this preprocess is to decrease the number of the colors in the image to a limited number, this operation has many advantages, the main advantage is the simplification of computations for the new limited number of colors, the second advantage is to make the CBIR system able to retrieve images with non-identical colors, the second advantage is crucial, it makes the CBIR system unable to detect the similarity between two sky images if there is any difference in the darkness of the blue color.

Figure 3.2 illustrates the similarity between two images according to the similarity between the two histograms, the first image and the second image are not identical, but they are very similar for the human eye perception. To the right of each image the histogram is shown, but here the histogram is taken for the gray scale image. If we look to the two histograms we can see obviously that they are looking very similar to each other, and therefore the images are supposed to be similar too.

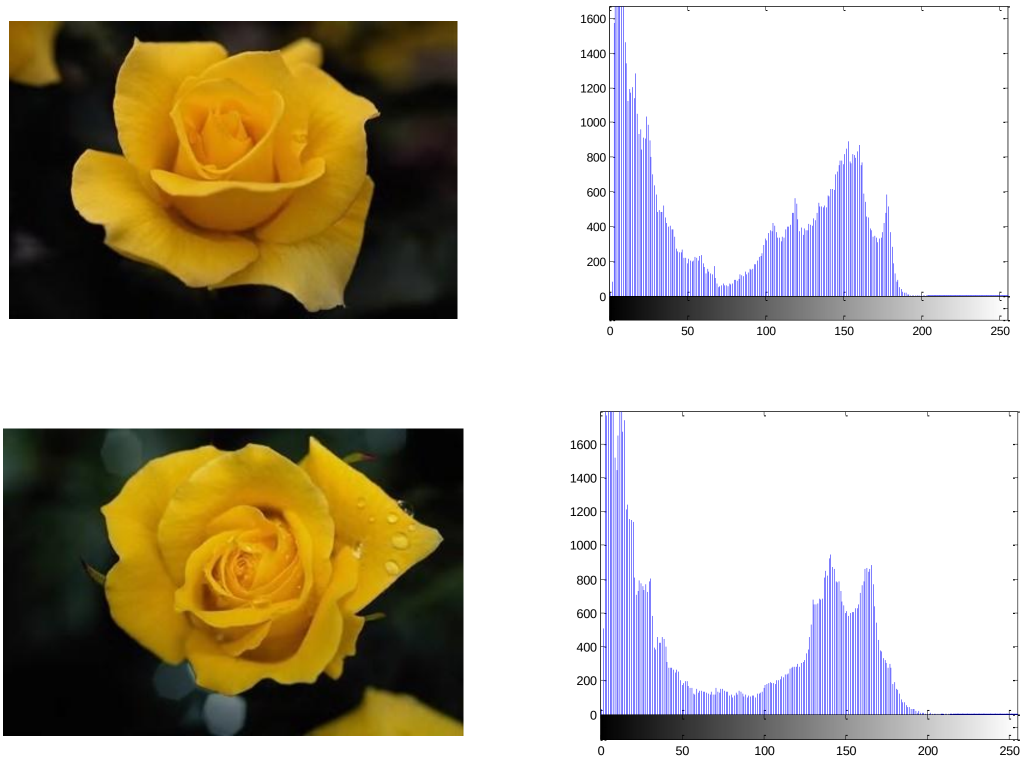


Figure 3.2: Histogramsare nearly similar for two similar images

To make the proper quantization for the color feature we have to understand the color systems used to express a specific color, and to be able to pick up the right color model for our quantization process. There are a variety of common color models like RGB, HSV, CIE, XYZ and many other models[54].

Figure 3.3 illustrates the quantization process, the left image is the original image and the right image is the result of quantizing the image into 4 colors.

Figure 3.3: Image quantized into 4 colors

The RGB (Red, Green and Blue) color model is the most common model used in computer vision systems, it describes the color in terms of three values (RGB: Red, Green and Blue), this model can be imagined as three spot lights for each color (RGB) as shown in Figure 3.4, the absence of the three colors is the black while the full existence of the three colors is the white, and the mixture between the different values of the RGB colors will make the other spectrum of the colors, for example the mixture of two similar values of green and blue will result in the Cyan color, and the mixture of the blue and the red color will result in the Magenta, and we can have the yellow color by mixing the green and the red colors.

In the above mixtures, we were talking about similar values of two colors or three colors, however random numbers of the RGB colors will result in the other true colors that can be generated by this system.

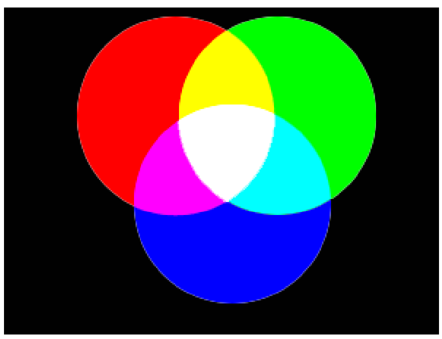


Figure 3.4: RGB color system

The RGB color system is a mixture of three main colors, but other systems depend on other techniques like the characteristics of colors instead of addition of colors. The

HSV (Hue, Saturation and Value) sometimes referred to as HSL or HSB, is the closest system to the human eye perception for colors, the hue represents the chromatic component of the color, the saturation describes the predominance of the color, the value is the intensity of the color, Figure 3.5 illustrates the Hue spectrum of colors, in the middle we can see the Saturation for a particular color (Red), if the color is not saturated it will look like white, but it will look in its true case if it is fully saturated.

To the right in Figure 3.5 we can see the value for the particular Red color, the color looks as black if this value was 0, and it looks in its original state if the value is 100%.

The above notion will be used later in the quantization process to quantize colors with specific values of the Saturation and Value as black or white colors.

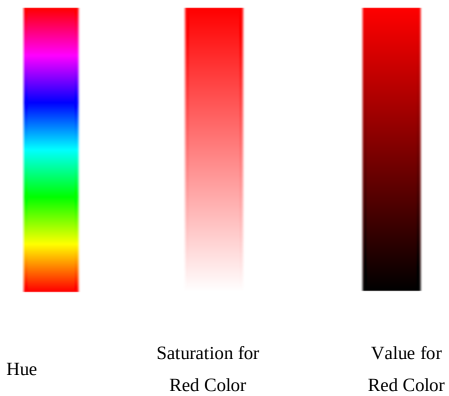


Figure 3.5: HSV color space

The understanding of the abovementioned models is very important to make the color quantization for the image, after the quantization the extraction stage will begin.

The most common approach applied in the color features CBIR systems is the Color Histogram approach, in this approach the image is represented as a vector of values, the index of each value represents the color (i.e. image quantized into 256 colors will be represented in 256 cells vector), each cell in the vector is called bin and it will include the number of pixels with the index's color (i.e. value of X in cell Y means there are X pixels in the image colored with Y). The resulting histogram is used for similarity measurement between two images.

In CBIR systems that depend on color features, the rotation does not make any effect on the similarity measurements, the translation of the image is not a concern also, but the resizing of the image will add more pixels with specific colors, or some pixels may be eliminated, this will change the results of the similarity measurement, however a simple normalization process can be applied to fix this problem.

The most common problem for color features CBIR systems (and particularly histogram based approaches) is its weakness in many types of images because it depends on the number of pixels with specific colors and nothing else, this problem makes this approach unable to find similarity for objects with specific background for example, or with some common color.

3.1.2 Shape

The shape is another important feature used to represent an image, in shape extraction process the system works on finding the Regions of Interest (RoI), this process depends on segmenting the image into many regions, then the objects or shapes have to be extracted from these regions. The good advantage in these approaches that depend on objects' extraction is its ignorance for the low level features used in the color and texture feature. In Figure 3.6 a good example for the shape feature, in the figure we can see the letter "A", the CBIR system in this case has to extract this shape from the whole text, then it has to be able to know this letter according to its shape characteristics.

Shape characterization is divided into two major parts, boundary based characterization (also known contour based shape representation), and region based characterization. Boundary based shape features include geometric shape description (area, perimeter, compactness, eccentricity, elongation and orientation), polygonal approximation, Fourier-based shape descriptors, etc., Region based features include statistical moments and grid-based approaches.

Figure 3.6: Shape detection for the letter “A”

Shape-Feature-dependent methods are suitable for applications where regions with specific objects or shapes are available, but this approach faces many obstacles and problems like the process used to segment the image into regions, and another problem is to make the object’s representation resistant for translation, scale and rotation.

However Shape feature CBIR systems are not suitable for images with many objects, and the number of objects adds up more challenges for the time factor, even though clustering is usually used in these systems, but it is only sufficient for images with limited numbers of objects, or as shown in our previous example for letters.

The complexity of calculations in this approach is another challenge, because every image in the database will be fragmented into a number of objects, every object has its own signature that describes it, then to find the similarity between two images we have to measure the similarity between each object in the first image with each object in the second image, these calculations add a tangible time when we have a huge database of images, and it also makes challenges in ranking the retrieved images according to similarity ratio.

3.1.3 Texture

Texture measures detect visual patterns in the image [53], and how they are spatially defined, these patterns are classified as regions each region is defined by sets, these sets not only define the texture, but also where in the image the texture is located.

The texture is considered as an important feature for describing a specific region, and it provides measures of properties like coarseness, smoothness and regularity[54]. Furthermore, texture can be thought as the regulation of patterns of pixels over a spatial domain. Figure 3.7 divides a random image into a number of different regions, each region has different characteristics for texture features. The texture can be affected with any noise to change from classification to another (i.e. regular or smooth to coarse). However there is no specific mathematical expressions to evaluate the texture, but many different methods are developed for computing texture, but these methods are not considered to be perfect for all the types of textures.

Figure 3.7: Image segmented into regions with specific characteristics

Figure 3.8 illustrates the common texture features used in the CBIR systems, these features are coarseness, contrast, directionality, line likeness, regularity, roughness. The figure illustrates two contradicting states of each feature.

Texture CBIR systems are very sensitive for image scale changes, because increasing or decreasing the size of the image has a great effect on the type of the texture, because it is expected to change from type to another one.

The statistical method is used to analyze local features for each pixel in the image, then a set of statistics is used to represent the image depending on the distribution of the local features, this method is usually applied on gray scale images to compute a co-occurrence matrix according to gray level differences.

The texture and color features are usually combined in the CBIR systems for extracting more representing information from the image.

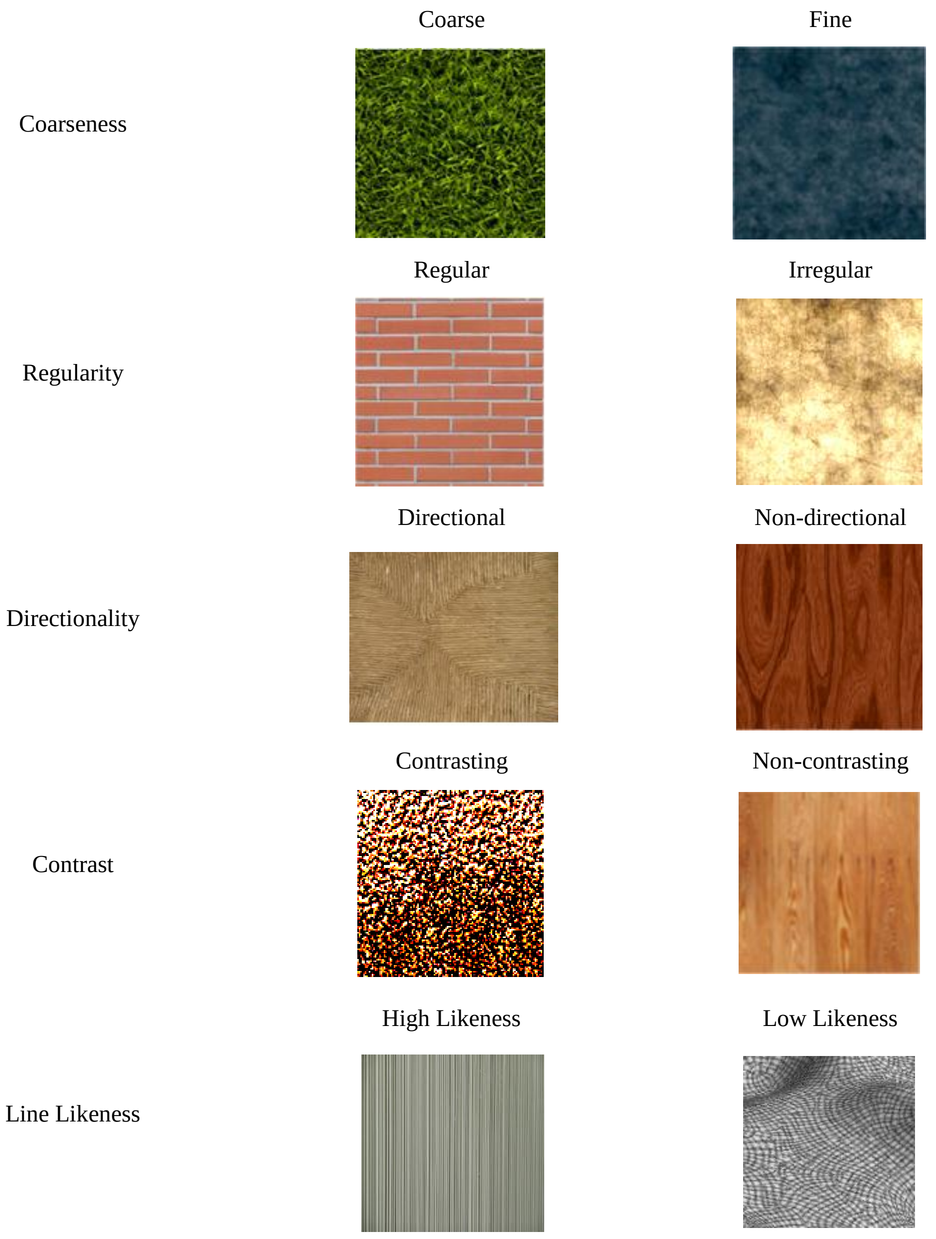


Figure 3.8: Texture features

In addition to the features shown in Figure 3.8, other features are used in some researches as a combination between two features or more, for example the Roughness feature is measured as the Sum of Coarseness and Contrast values.

3.2 Uniform Color Quantization

The quantization stage is one of the crucial stages in any CBIR system, in this research we will discuss the traditional quantization technique (the uniform color quantization), and -in our proposed approach- we will discuss another adapted technique (the non-uniform color quantization).

In this section we are going to summarize the uniform color quantization to find our CAH signature, but in our measurements we will use the technique used in [21], which is modified to be smarter in dividing the H space according to similar color regions (for the human eye), and it divides the SV spaces to make dark colors closer to the black and bright colors closer to the white. The Quantization used in [21] is elaborated as follows:

1. Hue is quantized as follows:

   $0, H \in [0,24] \cup [345, 360]$

   $1, H \in [25,49]$

   $2, H \in [50,79]$

   $3, H \in [80,159]$

   $4, H \in [160,194]$

   $5, H \in [195,264]$

   $6, H \in [265,284]$

   $7, H \in [285,344]$

2. Saturation is quantized as follows:

   $0, S \in [0,0.15]$

   $1, S \in (0.15,0.8]$

   $2, S \in (0.8,1]$

3. Value is Quantized as follows:

   $0, V \in [0,0.15]$

   $1, V \in (0.15,0.8]$

   $2, V \in (0.8,1]$

After quantizing the 3 planes we will get three values H, S and V, H is quantized into 8 values {0, 1 , 2 , 3 , 4 , 5 , 6 , 7}, S and V are quantized into 3 values {0, 1, 2}, now to find the final quantized color *P* which is the mixture of the previous HSV values we have to follow the equation 3.1:

$$P = 9H + 3S + V \quad \text{..........(3.1)}$$

The above formula will result in 72 colors, the first color is 0 and the last is 71, those would be our final quantized colors. The length of our CAH signature will be 576 (72 * 8, where 8 is the number of quantized angles).

In the following section we will discuss the adapted Non-uniform color quantization technique, and then we will make comparisons between the two approaches using different similarity measurements.

3.3 Global Based Features

The direct and traditional CBIR systems depend on computing the signature for the whole image, this approach is very simple and fast in calculating the signature and in measuring the similarity.

But this approach is not accurate in many cases, because it treats the image as a unique object, that makes the result as an average for the regions signatures, for example if we use the texture based CBIR system, and with the existence of an image with two halves the first is coarse and the second is smooth, we will get an average result looking like a regular image, this problem can be imagined for the color and the shape features too. This problem motivated many researchers to divide the image into many regions or segments, then the features are extracted from each region separately, this approach showed more accurate results.

3.4 Region Based Features

In this approach an image is divided into a number of objects or regions, each region is then analyzed to generate its own signature, the division of an image depends on finding similar areas within the image to be unified as one area or object, the similarity here depends on the used features, for example we can consider adjacent pixels with specific color as one region, or we can use the texture to consider an adjacent regular areas as one object.

The Integrated Region Matching (IRM) [56] treats each image as a vector of values, for example let $X = \{x_1, . . . , x_n\}$ and $Y = \{y_1, . . . , y_m\}$ two images, where $x_1, x_2, .., x_n$ are regions inside the image X, the similarity between each combination for regions in each image are calculated to build a many-to-many relationship, to find the final similarity between two images the weighted sum of distances is used.

This approach has the accuracy advantage because it depends on many objects and regions in the image, so the similarity here guarantees that the characteristics of particular region in the first image is similar to another region in the second image, but the problem in this approach is its high cost in time and computation measures.
Some approaches depend on dividing the image into static regions, in these approaches the image is divided into predefined set of regions, then each region is processed as a separate part, this can increase the power of the CBIR system and decrease the time required for the retrieval process, Figure 3.9 shows an example for dividing an image to a predefined number of regions.

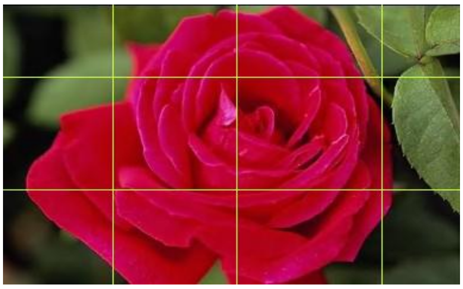


Figure 3.9: Dividing image into a static number of regions.

In the proposed approach in this research the image is divided into regions, but the regions are static, expressed as angles, so the computations in our approach are not a concern.

3.5 Image Segmentation

Image segmentation is a term used to describe the field of image processing used to detect specific objects like lines, curves , etc. , the segmentation of an image marks each pixel with a specific label, the pixels with similar labels share the same segment (i.e. the same line, or curve).The segmented image is usually a simple image used for another analysis or detection of some objects or patterns. In Figure 3.10 a segmented image is used to illustrate the human brain for medical issues.

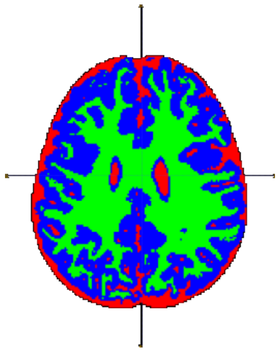


Figure 3.10: Human brain segmented image

3.6 Similarity Measurement

After building the signatures database which includes a signature at least for every image in our database, the next step is to be able to find the distance between the query image signature and each signature in our database. CBIR researchers depend on different approaches to measure this similarity, some approaches depend on the Euclidean Distance (equation 3.2), where X and Y are two vectors, $X=(x_1, x_2, \ldots. x_n)$ and $Y=(y_1, y_2, \ldots\ldots y_n)$.

$$EU(X,Y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots .. + (x_n - y_n)^2} \quad \ldots\ldots\ldots\ldots (3.2)$$

In other researches, the Canberra distance [57] is used, the equation 3.3 shows the distance between two vectors X and Y, where x_i is an attribute belongs to X, and y_i is an attribute belongs to vector Y, and d is the length of the vector X, which equals the length of vector Y.

$$d_c(X\,,Y) = \sum_{i=1}^{d} \frac{|x_i - y_i|}{|x_i| + |y_i|} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.3)$$

In the above equation, the numerator signifies the difference and the denominator normalizes the difference. Thus the distance value will always be < 1 whenever either of the feature components is zero and also reduces the scaling effect.

Other approaches use similarities like the Manhattan distance (equation 3.4), and the vector cosine angle distance (equation 3.5), which considers two vectors $X = (x_1, x_2, .. , x_n)$ and $Y=(y_1, y_2, \ldots, y_n)$, then $cos\Theta$ is the cosine vector angle between X and Y. Figure 3.11 illustrates the similarity measurement in the vector cosine angle distance between two vectors, and there is a variety of other distances that can be used in this field like the histogram intersection distance that can be used when we depend on the color histogram to measure the distance between two images.

In the histogram intersection similarity measurement if the histograms *I* and *Q* are identical, then the similarity *S(I, Q) =1*, and if either of the two histograms is completely contained in the other, then *S(I, Q) =1* also. In [63] there is a good comparison between different approaches for similarity measurements.

$$MH(X,Y) = |x_1 - y_1| + |x_2 - y_2| + \cdots + |x_n - y_n| \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.4)$$

$$VCAD(X,Y) = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2}\sqrt{\sum_i y_i^2}} = \frac{X.Y}{\|X\|\|Y\|} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.5)$$

The abovementioned similarities are very effective approaches, and they are used widely in image retrieval systems, but in this research we will use another similarity that will be more conditional and depends on the compared values within the histogram or the CAH signature used in this research.

In this research another technique is used to measure the distance between two signatures, the used technique has showed better results compared with other techniques used in other references.

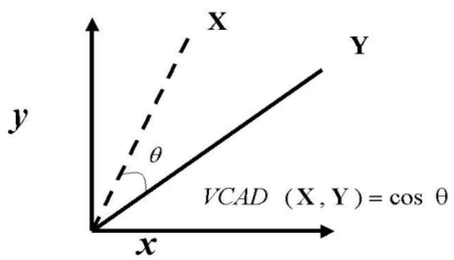


Figure 3.11: Vector cosine angle distance definition

The Proposed technique used in similarity measurement in this research tries to ignore the similarity between the small values, i.e. if two values in two signatures have the value 0 then this is not considered as similarity between the two signatures, but if the two values were high (i.e. 0.9 or 1) this has more impact on the similarity, and makes higher similarity.

Figure 3.12 shows that low values look like noise while the high values express the core of the image and have more impact on the similarity, in the Figure the similarity between the high peaks on the left has more impact than the random values located on the right of the histogram. In the Figure we can see that areas in the red rectangles (dashed lines) have more priority than areas in the green rectangles.

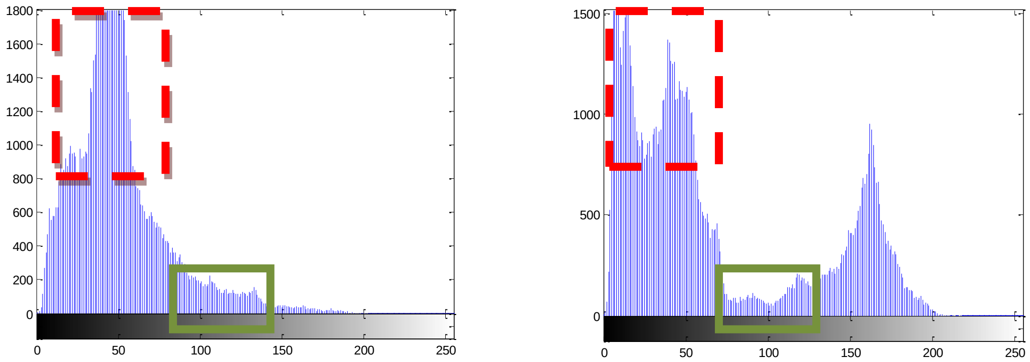


Figure 3.12: Similarity measurement depends on the value

The main similarity approach in this research will make priorities for the values in the signature according to its value (1 has more priority than 0 or 0.1).

Chapter 4

PROPOSED APPROACH

4.1 System Description

Our approach addresses two main problems in the CBIR system workflow, the first problem is the color quantization and the second is the representation of an image into a signature or series of representative values.

Color quantization is a very important process for any CBIR system, and it is the first process to be applied to the image before extracting the required color features, in [54] the authors focused on the quantization problem and its impact on the CBIR performance. The two common approaches used for quantization are the uniform and non-uniform color quantization.

To convert an image to signature, in our proposed approach, we divide an image into a number of angles, then statistics for the number of each quantized color in each angle are gathered to make the required signature (section 4.3). The signature generation is not enough but also we have to choose the right technique to measure the similarity between two signatures, a measurement technique is also proposed in section 4.5.

To enhance the operation of the proposed CBIR system a rotation resistant technique has been added in section 4.4, this technique will make our CBIR system able to detect similarity between images even if they were rotated.

To increase the retrieval speed for the CBIR system a clustering algorithm will be used, after this step a query image will be compared with the centroids of the clusters instead of measuring similarity with each image in the database.

Figure 4.1 illustrates the block model of the proposed CBIR system, the first block in this system is the image quantization module, this module will decrease the number of the colors in the image from millions to 37 colors.

Then The second module is to convert the image into the color and angle histogram, the CAH. The final step is to make normalization and to make the CAH rotation tolerant before saving it in the CAH database, then any new image will be converted in the same manner and the similarity will be measured with each CAH in the database using our proposed similarity measurement.

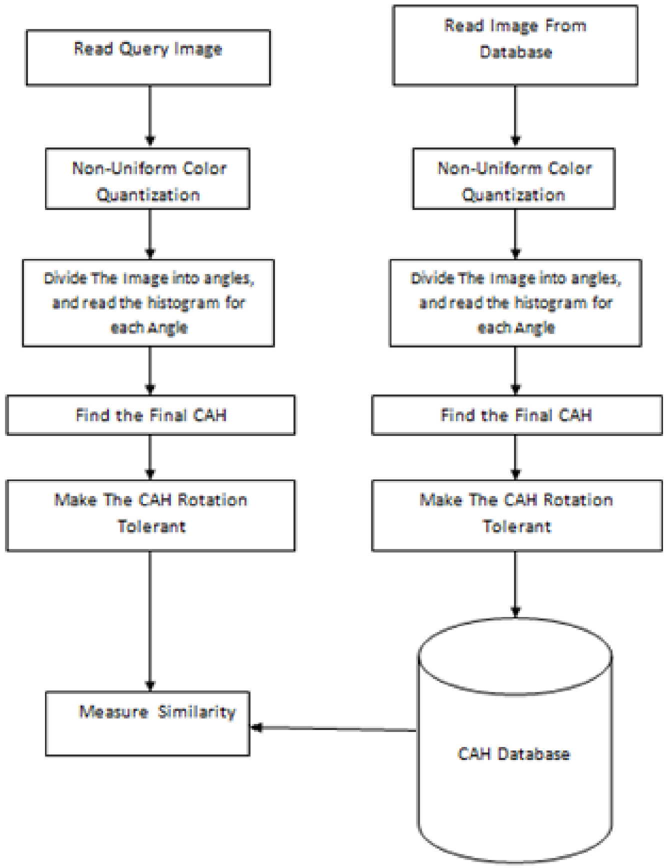


Figure 4.1: Block diagram for the proposed approach

After generating the CAH database we can increase the efficiency of our system by clustering the database into a number of clusters, this will eliminate the number of similarity measurements, which will decrease the image retrieval time.

The following pseudo code illustrates the whole operation of the process of converting a database of images into a database of signatures, it elaborates the modules shown previously in Figure 4.1, the explanation for the whole process will be in the following sections.

```
FOR each image in the collection
        Read RGBimage
        Resize Image (Optional)
        Read ImageWidth
        Read ImageHight
        Convert RGBImage to HSVImage
        Quantize HSVImage as QuantizedImage
        Build the CAH array with length = numberOfQuantizedColors × 8 Angles
        FOR each pixel in the QuantizedImage
                Read quantizedColor
                Read angle
                Read quantizedAngle = angle % 45
                Index in the CAH array = quantizedColor×8 + quantizedAngle
             Increment CAH(Index) by one
         END FOR
        Make CAH Rotation-Tolerant
        Save the CAH in the database
END FOR
```

In the second part of the process we have to find the CAH for the query image and to use our proposed similarity approach to find the distance between each CAH in the database and the CAH of the query image, then images are retrieved and ranked.

4.2 Non-Uniform Color Quantization

The number of colors within any image depends on the image type, color system and the depth of the color per pixel, for example if the RGB system used, with 8-bit depth for each color, then, our image includes 2^{24} colors or 16 million colors. Processing the image with all the possible colors needs high computations, which makes a main problem for image retrieval applications, this problem can be solved by decreasing the number of colors in the image to a small number that does not make high computations in the image processing stage. The second problem with this huge number of possible colors in the image is the difference between two similar colors, for example two colors with two different saturation will look different for the computer vision, but for the human vision they might look as one color. The solution for the second problem is to consider looking-similar colors as one color.

The solution for the two abovementioned problems is to use quantization, which is the process of defining blocks of similar or close colors as one quantized color, this process will decrease the number of colors from millions to dozens, and many colors with different grades will be treated as one color, this process will decrease the computation time, which is our first goal, and it will increase the power of our CBIR system.

To quantize a specific color system we have to define our criteria, for example in the RGB system we can define all the colors with ($R<50$, $G<50$ and $B<50$) as black, we can also define specific value for the white color, for example if (R>200, G>200 and B> 200) the color is considered white, but the RGB color is very good in the computer vision, but it differs very much from the human vision system which depends on perceiving the color and its saturation, the closest color system for this is the HSV system.

From [43] we can use the equations 4.1, 4.2 and 4.3 to convert the RGB color system to an HSV system:

$$H = \arctan \frac{\sqrt{3}(G-B)}{(R-G)+(R-B)} \quad \text{……(4.1)}$$

$$V = \left(\frac{R+G+B}{3}\right) \quad \text{……(4.2)}$$

$$S = 1 - \frac{min\{R,G,B\}}{V} \quad \text{……(4.3)}$$

The human eye color-perception system is very close to the HSV color space distribution, so it is expected to be very powerful in the content based image retrieval applications, and it is used in almost all the CBIR systems to retrieve similar images, The HSV color divides the color into three values, Hue H (Hue is represented as a disk from 0 to 360 degrees), Saturation S (0 to 1) and Value V (0 to 1), these values are illustrated in Figure 4.2.

The Hue elaborates the distribution of the colors, similar colors for the human vision are located as adjacent colors on the Hue disk, for example the similarity between the

Blue and Cyan colors for the human vision system is very high, and the two colors are adjacent in the Hue colors distribution.

Colors with low saturations are light and look like white for the human eye, and colors with low values look like black for the human vision system.

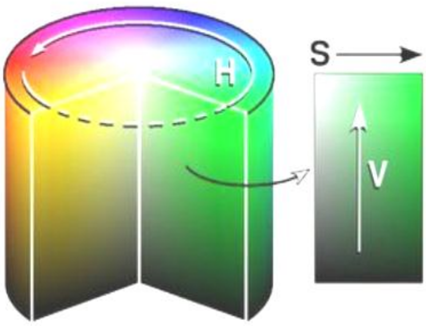


Figure 4.2: HSV color space

Different quantization systems used in CBIR applications, most of these applications depend on the uniform quantization for the HSV space, in these approaches the Hue is quantized into a number of periods, each period is substituted by a new number, for example if we assume the Hue is distributed across a disk like Figure 4.2. Then the disk is divided into uniform slices each is 60 degrees for example, these approaches that depend on the uniform quantization suffers some problems. In our proposed approach we depended on the non-uniform color quantization, which does not depend on similar angles.

Figure 4.3 (a) shows how the distribution of the colors across the HSV disk, the colors appear to be non-uniform, that makes the traditional quantization algorithms that divide the H space into similar pieces inaccurate.

Figure 4.3 (b) and 4.3 (c) illustrates the quantization blocks in the uniform and non-uniform quantization.

Figure 4.3 shows that the main colors of red, green and blue occupies more H-space than the other colors like Yellow, Cyan and Pink do, so we can use i.e. 80 degrees for the blue color and 30 degrees for the pink color to solve this problem, which is called the non-uniform color quantization.

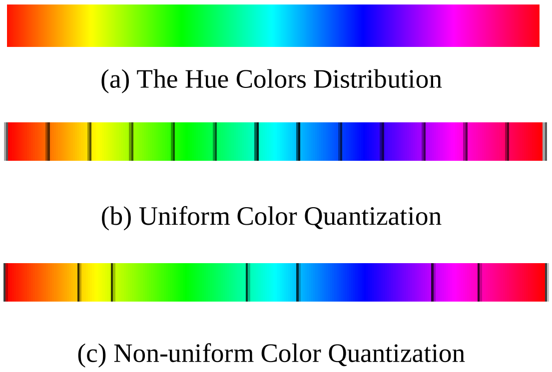


Figure 4.3: Hue color quantization

So, in this step it is not fair to divide the H plane into same-size pieces or blocks. Some ideas excerpted from [51] will be used in this paper for quantizing colors, the first idea is to divide the Hue plane into non-equal blocks according to the spread of each color, the second idea is to depend on the Saturation and the Value for the first step in the quantization process, in the first step the quantized color is classified as white, black or gray. The results showed that the ideas in [51] can make very good results but with some modifications on the Hue plane segmentation and more work in detecting the intervals of the first step quantization.

Another good quantization technique has been used in [21], this approach does not divide the Hue, Saturation and Value into similar blocks, but it depends on the distribution of the color to quantize it, and it depends on the darkness to consider whether the color is black or white.

Another idea is used in the non-uniform color quantization to quantize the Saturation and Value plane, to understand this approach we have to understand Figure 4.4 that shows the SV plane for three colors, Red, Green and Blue, Then Figure 4.5 will make more illustration for the quantization process.

From Figure 4.4 we have to note that low saturated colors look like white, and low valued colors look like black, and other notions will be discussed on Figure 4.5

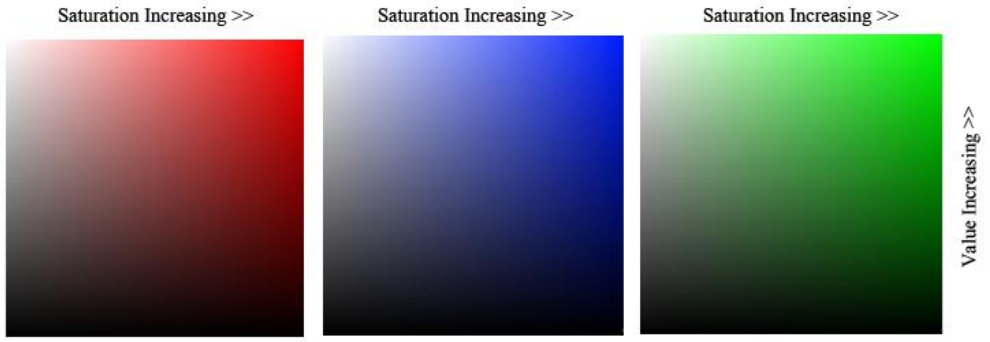


Figure 4.4: Saturation and Value planes

Figure 4.5 describes the SV space for saturation and value (the figure used the Red color, and it is applicable for the other colors), it is obvious from the figure that the low values are perceived by the human eye as black (Area 1), because of its darkness, the main idea here is to quantize all colors with $V<0.2$ as one color, so the darkness of the color will be used to detect black colors with ignoring the main color Hue and Saturation in this step, this makes our work very close to the human vision system, because it is another problem in the uniform color quantization approaches that they depend on giving more importance to the Hue color [62], while the non-uniform color quantization depends on the value of the Hue, Saturation and Value to decide which color has to take the power in the quantization process.

Another observation in the SV space is the gray area when $S<0.2$ (Area 2), in addition to the bright area which can be quantized as one color(white), the intended area is labeled 3.

In Area 2 the saturation $S<0.2$ and the Value $V<0.8$ and $V>0.2$, all the colors with these characteristics (the hue is ignored) will be considered as gray. In Area 3 where $S<0.2$ and $V>0.8$ the color is quantized as white color, if the conditions of Area 1, 2 and 3 are not matched, then the Hue will be used as the main value for our quantization. The remaining region is the real color region, which can be quantized as a combination of the three attributes H,S, and V.

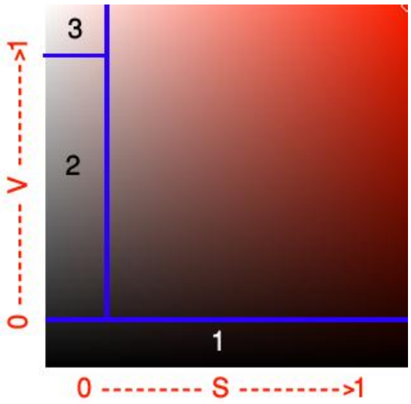


Figure 4.5: SV planequantization

The quantization steps can be described as the following steps, where C is the final value of the quantized color:

1. $C = 0, for v \in [0, 0.2]$

This is area 1 in the bottom of Figure 4.5, from the color of this area it is obvious that it has to be quantized into one black color with C=0.

2. for sϵ [0, 0.2] and $v\epsilon[0.2, 0.8)$, C is quantized to a gray color:

$$C = floor((v - 0.2)\times 10) + 1$$

This step quantizes area 2 in Figure 4.5, this area is gray gradient, it changes from the dark gray to the light gray as v increases. In the above equation the max value of v is 0.8, which will result in $((0.8 - 0.2))\times 10) + 1 = 7$, and the minimum value of v is 0.2, which will result in $((0.2 - 0.2)\times 10) + 1 = 1$, Now we understand that this step makes a result in the period [1, 7], where 1 means dark gray, 7 means light gray and other values like 2, 3, 4, 5, 6 are just gray level gradients.

In this step we used the floor to convert numbers into digits only, we also added 1 to the equation to avoid the result $C=0$ which means the black quantized color.

3. when $s\epsilon[0.2,1]$ and $v\epsilon(0.2, 1]$ it is a real color area, in this area the colors are segmented as shown in Figure. 4.3 (c). In this step the first thing we have to do is to quantize each plane in the HSV planes into a specific number of values. Saturation and Value will be quantized into 2 values (0, 1) for each, the Hue will be quantized into 7 values (0, 1, 2, 3, 4, 5, 6), as shown in the following steps:

a. $S = 0$ if $s\epsilon(0.2,0.65]$ and S=1 for $s\epsilon(0.65,1]$

b. $V=0$ if $v\epsilon(0.2, 0.7]$ and $V=1$ for $v\epsilon(0.7,1]$

The following formulas elaborate the quantization of the Hue spectrum (as seen in figure 4.3 (c)):

2, $H\epsilon[0, 22]$ U $H\epsilon$ (330, 360], (Red Color).

1, H (22, 45], which is Orange.

0, $H\epsilon(45, 70]$, which is Yellow.

5, $H\epsilon(70, 155]$, which is Green.

4, $H\epsilon$ (155, 186] ,which is Cyan.

6, $H\epsilon$ (186, 260] , which is Blue.

3, $H\epsilon$ (260, 330], which is the Purple color.

Now the Hue value is quantized into 7 values from 0 to 6, these values are arranged to make close colors next to each other, i.e. the yellow color is closer to the orange for the human eye, so they are close to each other in the quantization.

The final measurement of the color C is:

$C = 4H+2S+V+8$ …………………………………………………………..(4.4)

Where H stands for quantized Hue values in the range [0, 6], S for saturation and V for Value. The quantization of the main color area will result in 28 colors (from 8 to 35), we added 8 to our equation to avoid the results in step 1 and 2 where C=0 (black color) and C= (1, 2, 3, 4, 5, 6, 7) for the gray level gradients.

In equation 4.4 the first quantized color will be 8 (when H, S and V are zeros) and the last one will be 35 (H=6, S=1 and V=1).

4. When $s\epsilon[0, 0.2]$ and $v\epsilon(0.8, 1]$, which is area number 3 in Figure 4.5, this part is quantized into a white color and C=36. And then we have a total of 37 colors, from 0 to 36.

Figure 4.6 shows the result of quantizing an image using the abovementioned technique. The quantized image in the figure is displayed in gray scale mode.

Figure 4.6: Quantized image with 37colors converted to gray scale image

4.3 Representing image as Color and Angle Histogram CAH

The main idea in this step is to select a reference point in the center of the image, then according to this point the image is divided into 8 regions (We used 8 regions in our CBIR system, but other divisions can be used), these regions will make our CBIR system able to detect rotated images if the rotation was in the increments of 45°, Figure 4.7 illustrates the meaning of the angles in this context.

Figure 4.7: Image is divided into 8 regions

Then the image representation process has to count the number of each quantized color in each angle, to convert the image into a color and angle histogram (the proposed CAH signature).

The following example divides the image into 4 angles, and 3 quantized colors to describe the process of the proposed technique. The angles are quantized in this example as follows:

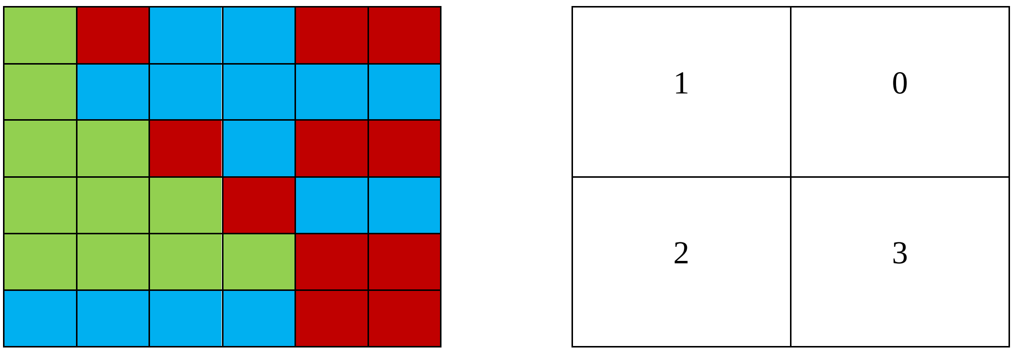
1. 0 to 90 is quantized as 0
2. 90 to 180 is quantized to 1
3. 180 to 270 is quantized to 2
4. 270 to 360 is quantized to 3

Now if we have two vectors *U* and *V*, with *U* is a horizontal vector extends from the reference point in the origin and the farthest point to the right, and *V* is the vector extends from the reference point in the origin and our point of test $P\ (x_1, y_1)$, then equation 4.5 will be used to find the angle between the two vectors:

$$\theta = \tan^{-1}\frac{x_1-0}{y_1-0} = \tan^{-1}\frac{x_1}{y_1} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.5)$$

Then ϴ is quantized to the four abovementioned values (0, 1, 2 and 3) according to its value.

For this elaboration assume that we used the center point as the main point, and assume that the image was quantized for 3 arbitrary colors (0,1,2), and let the angel be divided into 4 regions as shown in figure 4.8 (b) , then let figure 4.8 (a) is the quantized and processed image.



(a) Quantized Image

(b) Image divided into 4 regions

Figure 4.8: Image quantized into three colors Red = 0, Green = 1, Blue = 2

After preparing the image suppose that we have a camera on the center (the supposed reference point), from this camera we count the number of pixels with specific color in every angle, the color/angle histogram (CAH) will be represented as shown in Figure 4.9, in the first row number 4 (in the first cell) means that there are 4 pixels colored red and fall in the angle between 0 and 90 degree, the number 3 in the first row (in the last cell) means the number of blue pixels that falls in the angle from 270 to 360. The first row (the grayed row) will represent the final CAH signature in the end.

Number of Red pixels in the angle 0° to 90° = 4

# of pixels	4	2	0	5	0	4	6	1	5	3	3	3
Angle	0 0° to 90°	1 90° to 180°	2 180° to 270°	3 270° to 360°	0 0° to 90°	1 90° to 180°	2 180° to 270°	3 270° to 360°	0 0° to 90°	1 90° to 180°	2 180° to 270°	3 270° to 360°
Color	0 for Red				1 for Green				2 for Blue			

Figure 4.9: The final histogram for the image (the gray row)

Histogram will be represented in *x* values, where *x* is the result of multiplying the number of quantized colors and the number of the quantized angles. In our example we have three quantized colors and 4 angles, so we have 3×4 values or 12.

In the real application, the number of quantized colors will be 37 colors and the number of angles will be 8 angles(total number of CAH values =37×8 = 296 value).

In the previous example the image is represented as the vector (4, 2, 0, 5, 0, 4, 6,1,5,3,3,3), this vector will be used later for measuring the similarities among images.

However, to make this Color and Angle Histogram (CAH) invariant to scale changes we have to make all the numbers in this CAH in the period [0, 1], this can be achieved by dividing all the numbers by the maximum number in our CAH, which is 6 in our example, and other techniques will be used in section 4.4 to make a rotation resistant CAH.

4.4 Rotation Resistant CAH

One of the important goals for any CBIR system is to be able to detect similarity between an image and its rotated copy, and the power of this rotation invariant system depends on the rotated angles that can be detected, for example some CBIR systems can detect flipped images (horizontally or vertically) only, other approaches can detect images if rotated 90° or its multiples (90°, 180°, 270° or 360°). This Rotation invariant CBIR system needs additional techniques or modifications for the image or the signature.

The histogram or CAH must be adapted to be tolerant for specific incremental rotations and to be scale invariant too, in the end of the previous section we illustrated how to make this CAH scale invariant, now we are focusing on the rotation invariance problem, to fix this problem we will find the highest value in the CAH, which is 6 in our example, this value has to be rotated to be the first in the group (0 ,4 ,6 ,1) (in the previous example we divided the image into 4 angles) , so it has to be rotated two stepsto left, the new block will be (6, 1, 0, 4), and all the other colors has to be rotated two steps too, so the histogram will look like the array (0, 5, 4, 2, 6, 1, 0, 4, 3, 3, 5, 3), this rotation for each image in the collection will make the similarity measurement resistant for rotations in any angle of the sampled angle (which is 90° in our example and 45° in our approach), in our approach rotating the image to 45°, 90°, 180° or 270° will result in the same vector (with some differences because of the distribution of the pixels), this will make the similarity high between the original image and its rotated copy.

Figure 4.10 shows an image from the Corel-1000 database, this image has been rotated into 4 images each one is rotated with a specific angle (90°, 180°, 270° and 360°), for each image the corresponding CAH and the rotation tolerant CAH are shown. Similarity measurements showed that the similarity between the rotation tolerant CAHs are higher than the original CAH which will be helpful for detecting rotated images in our approach.

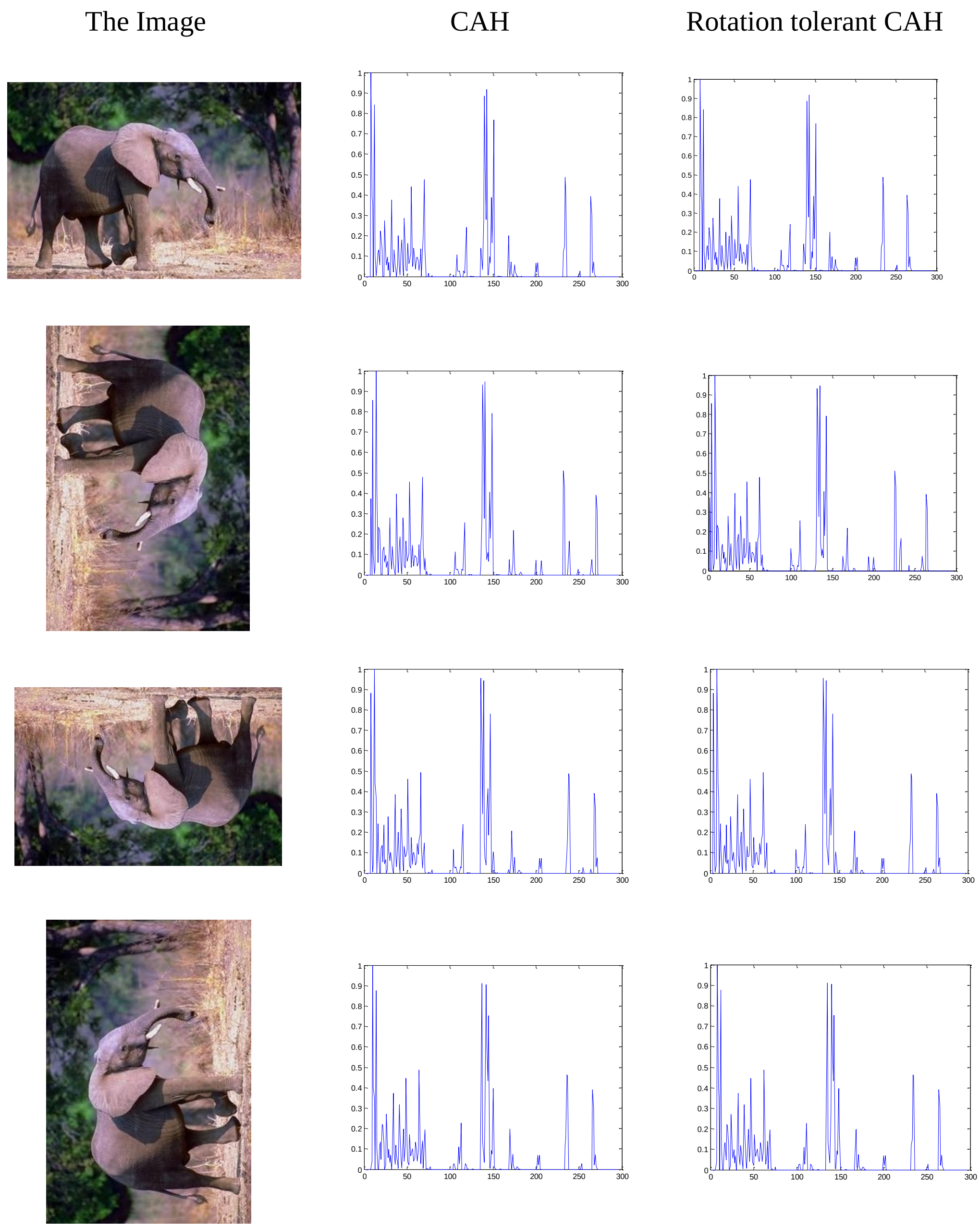


Figure 4.10: An image rotated into 90° multiplications and the corresponding CAH

The rotation of the CAH is not applied on the whole vector (the CAH), it is applied on each block (each block represents a color), so we have to make 37 rotations one for each block.

4.5 Similarity Measurement

Each Image is represented as an array of values (the CAH), each value lies in the range [0, 1], The Euclidian Distance (and the simplified form of it without the square root) has showed insufficient results in many tests, the main reason of weakness in this method is its equality in measuring differences, for example the distance between the following couples ((0, 0.2), (0.4, 0.6) and (0.8, 1)) is 0.2 for each couple, but this is not perfect, because the value of the numbers has different impact on the similarity measurement (higher numbers have more impact than small numbers that might be just noisy colors). The following equation is used to measure the Euclidian distance:

$$|x - y| = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.6)$$

The main argument in our approach with the Euclidian distance is that zero and low numbers do not deserve the same interest of high numbers in our calculations.

Another similarity measurements like the chi-square distance [61] added the sum of the two values to the denominator, this addition will increase the similarity when the two values are the higher, and vice versa the similarity will be lower when the two values are lower.

$$dissimilarity = \sum_{i=1}^{n}\left(\frac{Fq(i) - Ft(i)}{Fq(i) + Ft(i)}\right)^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4.7)$$

However the above distance in equation 4.7 from reference [61] needs more modifications to match our idea, and it also has to avoid the zero value of the denominator, when the two values are zeros.

In the modified technique used in this research, the similarity is adjusted to make non-uniform distances according to the effect of each value in the represented image. Note that in the following equations we depend on the concept of similarity and not the distance (like the Euclidian and the chi-square distances), so, two identical images have the similarity 1, and two totally different images have the similarity value of zero.

In our similarity measurement note that the result is a variable number in the period (-∞,∞), in the end of the calculations the results will be converted to be in the period [0, 1], with 0 is two different images and 1 means two identical images.

The similarity s_i measurement between two values a_i and b_i *in* two vectors A and B is calculated as follows:

a) If a_i or b_i equals zero

$$s_i = -|a_i - b_i| \qquad (4.8)$$

b) If a_i and $b_i \neq 0$

$$s_i = (1-|a_i - b_i|) \times min(a_i, b_i); \qquad (4.9)$$

where the total similarity S is shown in the equation 4.10 , i is the location of the feature in the vector, and n is the number of values in the signature of the image or the CAH in our research.

$$S = \sum_{i=1}^{n} s_i \qquad (4.10)$$

In eq. 4.8 the difference between two values with one of them is zero is considered as a negative similarity, because in each value in the first image must has a response in the 2nd one, if there is no existence for this color in this angle it is considered negative response. Note that this condition can never exist in two identical images.

In eq. 4.9 the similarity is multiplied by the lowest number to make low importance for small numbers like 0.1 and 0.2 when it has to be compared with 0.88 and 0.92 for example, and to give less importance for comparisons between 0.9 and 0.07 for example (considering the small values are expected to be noise).

After measuring the similarity between the query image and each image in the database, similarity then is converted to be in the range [0, 1] with 1 means identical images, and 0 means no similarity at all.

After retrieving a number n of images, the similarity between each retrieved image and the query image is saved in the vector $R = (r_1 , r_2, \ldots\ldots, r_n)$, where r_1 is the similarity between the first retrieved image and the query image.

From the similarity measurement equations 4.8, 4.9 and 4.10 we know that the values in the vector R are not located in a specific period, and this problem has to be solved to be able to rank the images later.

The equations used to normalize the vector of results *R* to be in the period [0,1] are as follows:

1. $R - min(R)$, Convert all the values in R to be in the period $[0, \infty)$, no matter if the lowest value is positive or negative.
2. $R / max(R)$, this will convert all the values in the results vector to be in the period [0,1], because the maximum value has been divided by itself to be 1, and all the other values will be ≤ 1.

The result of our calculations are similarities, and they can be converted to distances only by making the subtraction $1\text{-}R$.

The above similarity measurement showed an enhanced results in retrieving images in the proposed system, in the simulation and results chapter (Chapter 5), there is a comparison between our proposed similarity and the results of the Euclidian Distance similarity.

4.6 Clustering to increase the speed of Retrieval

The images database is converted to a text database or vectors of digits actually (the CAH), this process does not affect the retrieval time because it is an offline process, each line or CAH (vector of 296 values) represents an image, the similarity measurement time for any query image can be enhanced by clustering the images database into a specific number of clusters, then similarity measurements are applied to the centroids of the clusters only, instead of finding similarity with each image in the database. The clustering technique used in this paper is similar to the K-means algorithm, but with similarity equations used in section 4.5 instead of using the Euclidian distance, other ideas like the number of centroids and eliminating outliers have been used from[52].

After finding the most similar centroid, the CAH similarity is measured with each image in the cluster, this will eliminate finding similarity with many images, and it will lead to decreased computation time.

In our clustering algorithm the number of clusters depend on the number of images (or objects) in the database, the equation 4.11 will be used to define the number of our clusters (and centroids)):

$f(N) = \ln(N)$ ……………………………………………………………….(4.11)

Equation 4.11 used in [52] to avoid the time consuming process of finding the perfect number of clusters within data, the equation depends on a logarithmic function that increases slowly while the number N increases dramatically, this will keep the number of our clusters small enough to deal with.

In Table 4.1 the row N is the number of images in the database, and the value f(N) is the number of clusters for our clustering algorithm [52].

Selecting the number of clusters as static number can be a negative choice because it might not be able to detect the real number of typical clusters, it can also be tempted to choose weak connections because it has to commit to the clusters' number. But however the static selection of the number of clusters will increase the speed of our CBIR system and the clustering algorithm because it will avoid huge volume of calculations involved in selecting clusters in finding error ratios.

Table 4.1: Number of clusters for different numbers of images

N	*Billion*	*Million*	*1000*	*100*	*20*	*10*
f(N)	*143*	*63*	*15*	*7*	*2*	*1*

Now from the table if we have one Million images, then the direct number of clusters will be 63 (from the table), to find the centroid of each cluster the algorithm will select a random number of points from the whole database.

After finding the number of our clusters (which is the number of centroids also), and considering the huge size of our database (i.e. R points $= f(N) * 10$), and calculate the average of the selected points as our centroid, then all the points in our dataset will be assigned to one of the centroids, the value of the centroids will be measured again and again until we reach the convergence case.

For example if we have to select two clusters *k=2* and the number of attributes (the length of the signature CAH) p=2, then to calculate the first centroid we have to select two random points (when R=2) one for each cluster, for example:

P1={ 5, 8} and P2={ 7, 10}

The first centroid M1 is the average of the selected points:

M1 = { (5+7)/2 , (8+10)/2 } = {6, 9}

The second centroid will be calculated by selecting other two random points, and calculating their average.

After finding our final centroids, we can repeat this step for *S* times, then we have to find the error ratio for each step, this can be done by measuring the average distance between each centroid and the others, then the most suitable solution for our clustering is the step with the highest average distance.

Applying the clustering algorithm in this section for the Corel-1000 database, which includes 1000 image, and according to Table 4.1 the number of clusters will be 15 (15 centroids), each cluster will include 67 images nearly, this will add more overhead to the offline process during the clustering process, but it will decrease the online time used in the image retrieval process, because we have to make measurements for the query image with 67 images instead of 1000 images.

Chapter 5

SIMULATION AND RESULTS

5.1 Experimental Environment

The Corel 1000 images database which includes 1000 images has been used as the dataset to find the ratio of the image retrieval success for the ten types in the database (Africa , Beaches, Building, Buses, Dinosaurs, Elephants, Flowers, Horses, Mountains and Foods), the database includes 100 images from each type, Figure 5.1 shows a sample of the Corel 1000 images database.

Figure 5.1: Corel 1000 images database samples

In the first step Matlab code has been written to read the whole database and convert each image to the representing vector ,or the CAH, which depends on the color and angle, the output is saved to a text file, then the query image has to be represented in the same way, then the resulting vector has to be compared with each vector in the text file, the Matlab code has been adapted to generate the CAH signatures using either the uniform or non-uniform color quantization techniques.

To find the precision of our system for each type of images, 10 images are selected from each type randomly as query images, the search algorithm will start and retrieve the top 20 images with the highest similarity (for each image of the 10 images), then the precision value is measured (how many relevant image retrieved in the top 20 images), the average for the ten images is considered as the final precision value for the type of images.

The results showed very good enhancement in finding similarity among different types of images, for other types the enhancement was slight because of the interference among similar colors in some types of images.

In the following sections we will work on comparing the CBIR system using uniform and non-uniform color quantization techniques, we will also compare the results when using the Euclidian distance and our proposed similarity measurement technique.

To add noise and make some modifications on the images like rotation, translation and resizing, the well-known image processing application Adobe Photoshop was used.

The laptop used for building and running the codes is an i7 CPU with 2.4 GHz core 2 quad, and 4GB RAM.

5.2 Euclidian Distance and Uniform Quantization

The first similarity measurement will depend on the Euclidian distance to find the top ten retrieved images. In this section two tests will be applied, the first will be applied on the CAH signatures database generated using the uniform color quantization technique described chapter 3, and another test will be applied to the database generated with the non-uniform color quantization technique from chapter 4.

Figures 5.2 through 5.5 shows the result of retrieving the top 10 relevant images using the uniform color quantization and the Euclidian distance. In Figure 5.2 the query image used is an image for a historical building, in Figure 5.3 an image for a dinosaur has been used, this image in the Corel-1000 database have a special case because they have a common background with pure color, in Figure 5.4 an image for a horse has been used, and the last Figure 5.5 used an image for a flower, this image has a specific region of interest in the center of the image and a dark background around the region (the flower).

In Figure 5.2 we can see that some of the retrieved images are irrelevant, but they have been considered as relevant images like the African, the Elephant and the Bus.

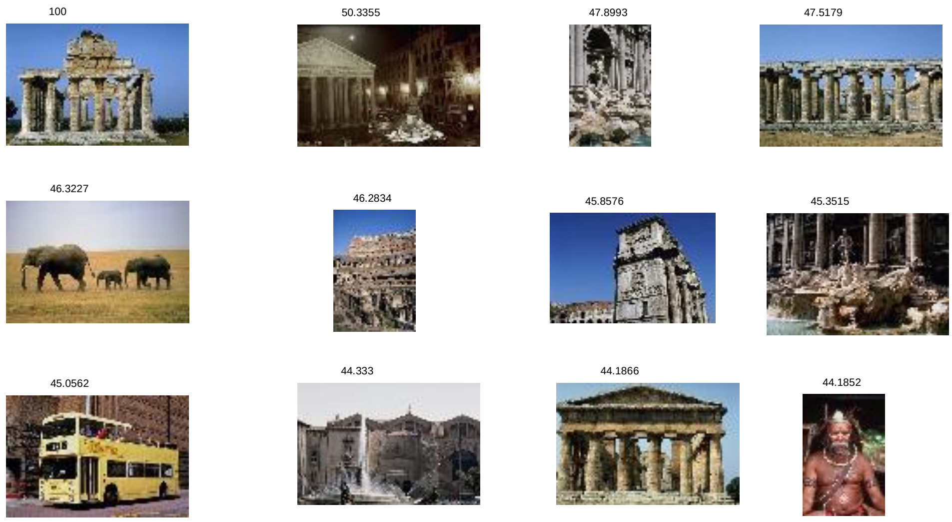


Figure 5.2: Euclidian distance and uniform color quantization test 1

In Figure 5.3, a query image for a Dinosaur, all the retrieved images in this example are relevant images, the main reason for the success in this test is the common background color.

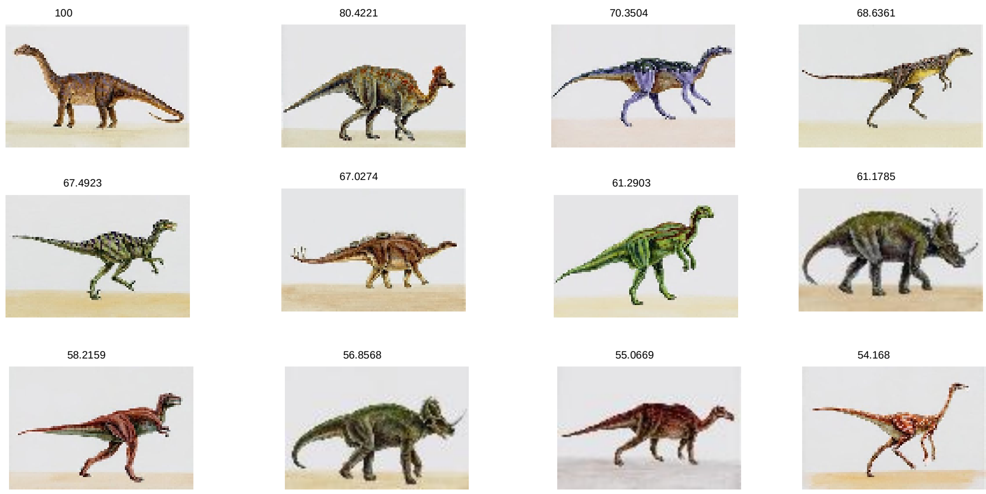


Figure 5.3 : Euclidian distance and uniform color quantization test 2

In Figure 5.4 another query image for a horse has been used, some of the retrieved images are also irrelevant in this test like the red and the yellow flowers.

Figure 5.4: Euclidian distance and uniform color quantization test 3

In Figure 5.5 an image for a yellow flower has been used, in this test many irrelevant images also retrieved, another problem appeared here, which is the similarity with flowers with different colors, instead of retrieving the yellow flowers.

From the retrieved images in this section we can conclude that we still have a problem in the similarity measurement in addition to the color quantization technique.

Figure 5.5: Euclidian distance and uniform color quantization test 4

5.3 Euclidian Distance and non-uniform Quantization

In the second test we will use the Euclidian distance with the non-uniform color quantization. Figures 5.6 through 5.9 illustrates the retrieval of the top 10 images when using the Euclidian distance to measure the similarity among the CAH images' signatures generated using the non-uniform color quantization technique.

In Figure 5.6 we can see that some of the retrieved images are irrelevant to the query image of the historical building, But however it has a slight enhancement on the first approach that used the uniform color quantization.

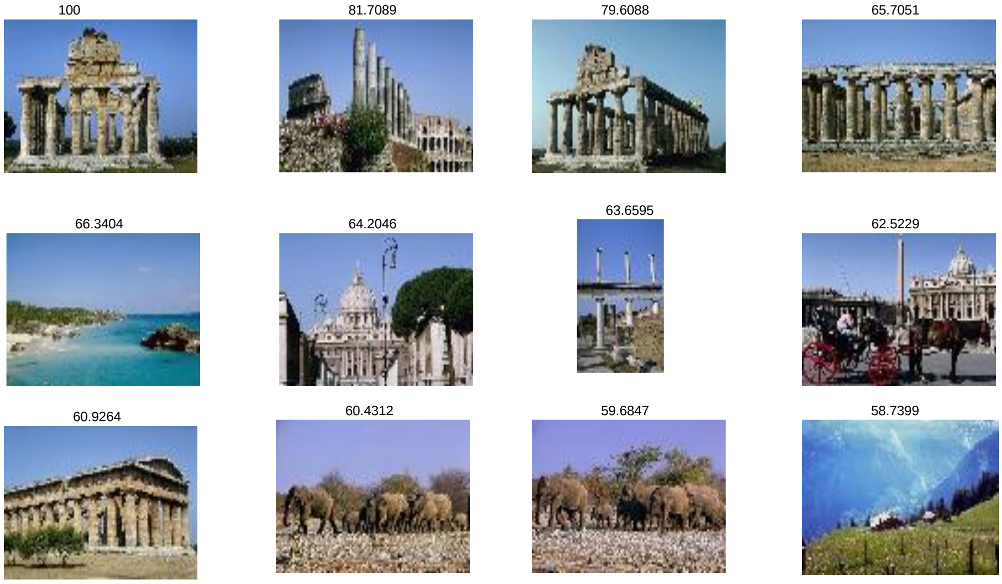


Figure 5.6: Euclidian distance and non-uniform color quantization test 1

In Figure 5.7 the CBIR system could retrieve the relevant images for the selected Dinosaur image.

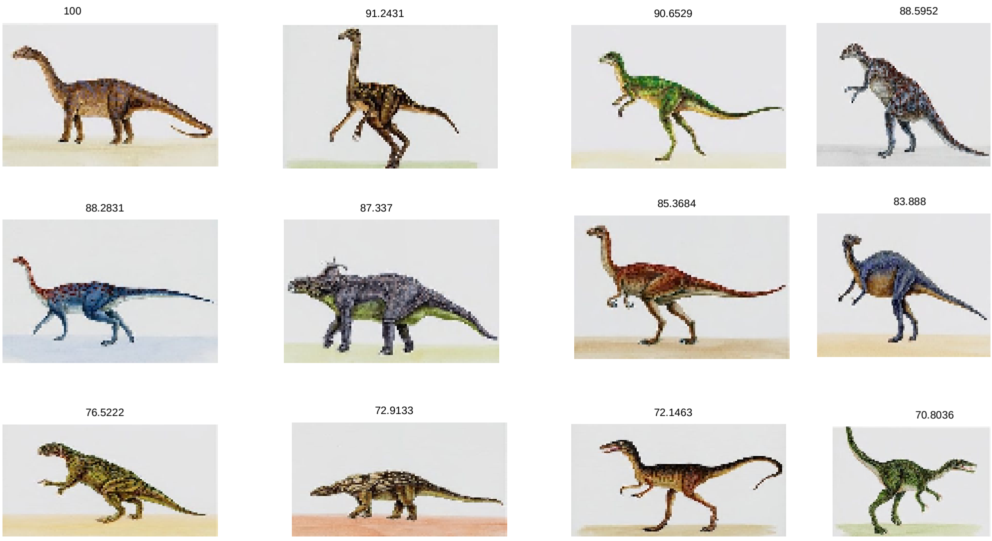


Figure 5.7: Euclidian distance and non-uniform color quantization test 2

In Figure 5.8, we can see an improvement on the retrieved images compared with Figure 5.4 that used the uniform color quantization, and the flowers images did not appear in the retrieved images.

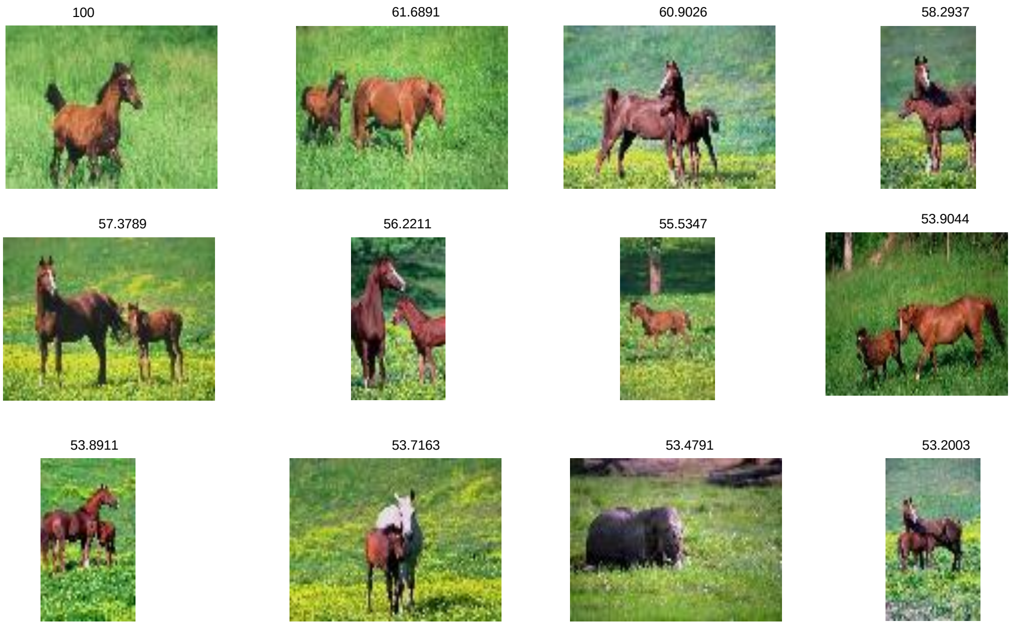


Figure 5.8: Euclidian distance and non-uniform color quantization test 3

In Figure 5.9, the results are much more better than the results in Figure 5.5, because the retrieved relevant images are more in the number and higher in the similarity for the human eye perception.

Figure 5.9: Euclidian distance and non-uniform color quantization test 4

5.4 Proposed Similarity and Uniform Color Quantization

In this section we are going to use our proposed similarity measurement technique to use another form instead of the Euclidian distance, our proposed similarity (discussed in section 4.5) showed better results with more precision ratios.

In this section another test will be applied using the CAH signatures database generated using the uniform color quantization, and the second test will use the CAH signatures database generated using the non-uniform color quantization. Figures 5.10 through 5.13 use the same query images used in the previous section to illustrate the difference when the new similarity measurement is applied to the uniform quantized CAHs.

In Figure 5.10 the results are not interesting because many of the retrieved images are irrelevant to the query image.

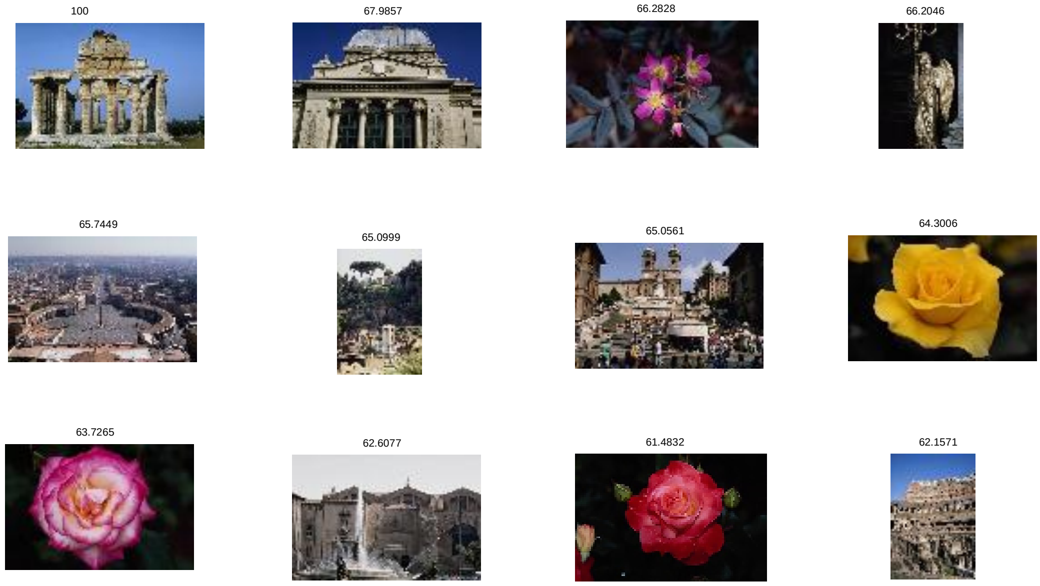


Figure 5.10: Historical building to test proposed similarity and uniform color quantization

The CAH signatures are making good results in the case of the Dinosaurs images for the two similarity measurement techniques, as shown in Figure 5.11.

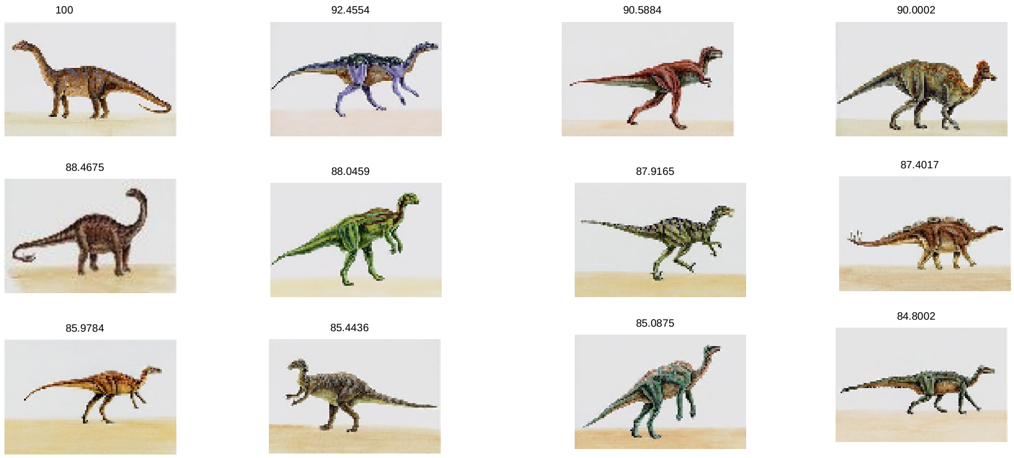


Figure 5.11: Dinosaur to test proposed similarity and uniform color quantization

In Figure 5.12, the retrieved relevant images are satisfying, but the appearance of the white horse in the image briefs the color quantization problems.

Figure 5.12: Horse image to test proposed similarity and uniform color quantization

In Figure 5.13, the retrieved images are all relevant, but we still face the same problem that we have red retrieved images for a yellow query image (which is a color quantization problem), this problem will be solved later when using the non-uniform color quantization combined with other similarity measurement technique.

Figure 5.13: Flower image to test proposed similarity and uniform color quantization

5.5 Proposed Similarity and Non-Uniform Color Quantization

Now we have to test our proposed similarity measurement technique to measure the similarity among CAH signatures generated using non-uniform color quantization, Figures 5.14 to 5.17 use the proposed similarity and the non-uniform color quantization.

The retrieved images shown in Figure 5.14 showed a great improvement on the results shown in Figures 5.2, 5.6 and 5.10, the results are very promising because all the retrieved images are considered relevant images.

Figure 5.14: Historical buildingto test proposed similarity and non-uniform color quantization

In Figure 5.15 the retrieved images are also relevant, and it is obvious that this technique is able to retrieve relevant images when we have an image with specific background like the Dinosaurs shown in Figure 5.15.

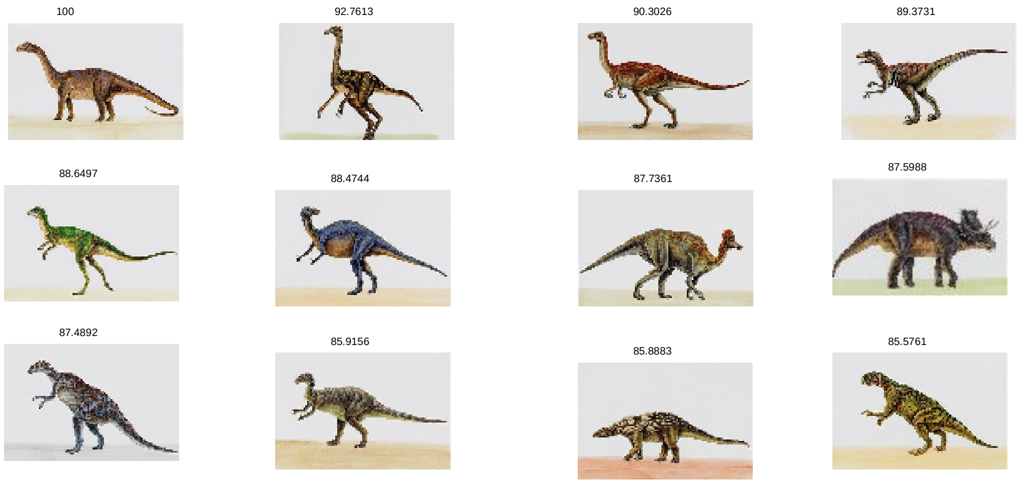


Figure 5.15: Dinosaurto test proposed similarity and non-uniform color quantization

All the retrieved images in Figure 5.16 are relevant images, the results are showing good enhancement compared with results in Figures 5.4, 5.8 and 5.12.

Figure 5.16: Horse image to test proposed similarity and non-uniform color quantization

In Figure 5.17 we have some irrelevant images appeared, but we have good improvement here, because we have no red flowers retrieved, and all the retrieved images are more similar to the yellow flower. The results in Figure 5.17 are very great compared to the results in Figures 5.5, 5.9 and 5.13.

Figure 5.17: Flower to test proposed similarity and non-uniform color quantization

In the Figures 5.18 to 5.22 other tests have been applied to illustrate the retrieved images for randomly selected images, the query image is retrieved with similarity of

100%, which is an important thing for applications interested in finding identical images for some purposes like intellectual rights.

In Figure 5.18 the query image for a bus, and all the retrieved images are very similar for our query which indicates a very interesting output.

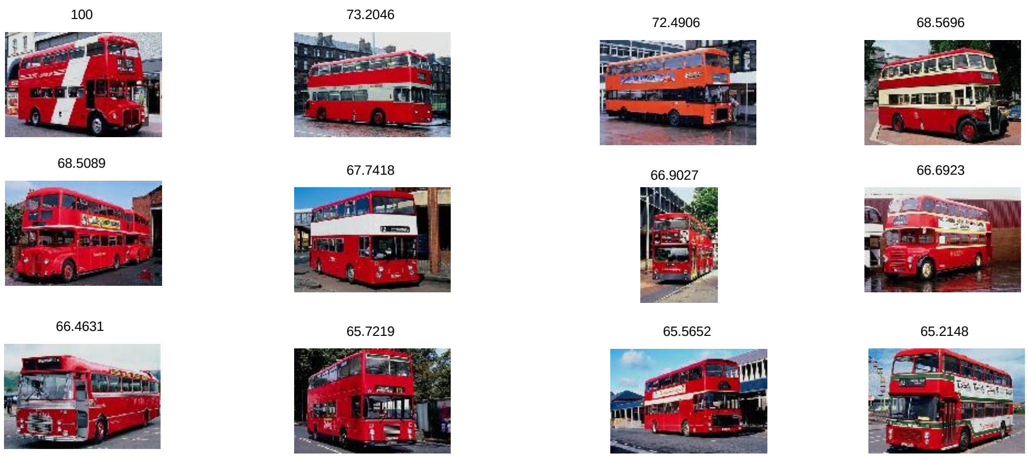


Figure 5.18: Red bus is used as query image

Some techniques are able to detect similarity for images like the previous–Bus Image- but they are not able to detect similarity for images with vast backgrounds like the Dinosaurs images, but the proposed technique in Figure 5.19 and many other tests showed nearly perfect retrieval for this type of images.

Figures 5.20, 5.21 and 5.22 are other examples for applying this technique for other types of images.

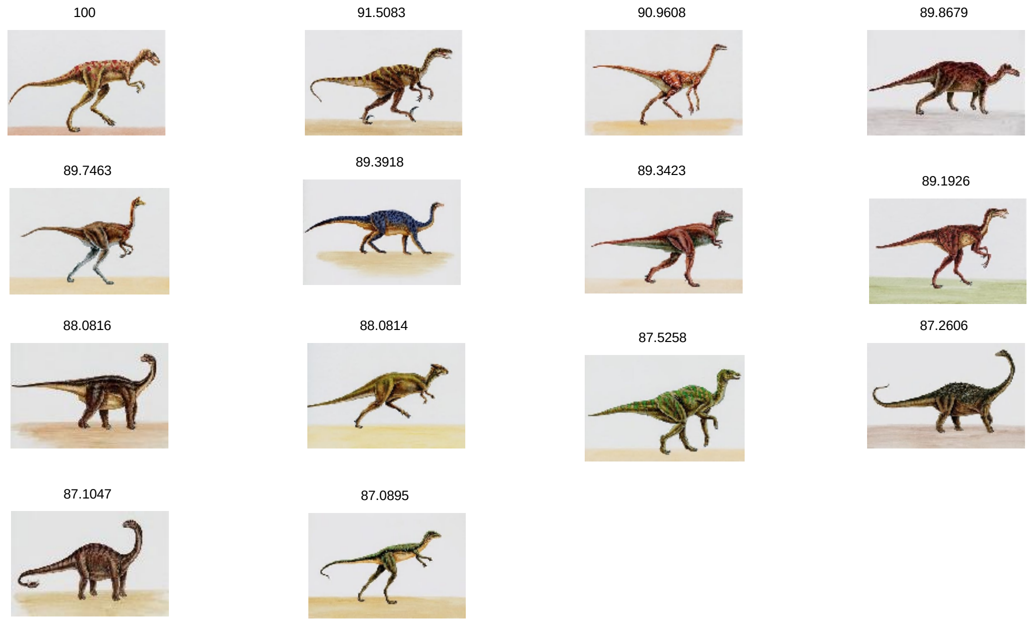


Figure 5.19: Similarity for Dinosaurs

Figure 5.20: Image retrieval for yellow rose

Figure 5.21: Horses in green fields

Figure 5.22: Historical buildings retrieval

The previous examples showed that the scale of the image does not matter because the similarity measurement depends on the normalized distribution of the colors across the image.

5.6 Rotation and Translation Tests

One of the important factors in the CBIR system retrieval is to be able to find similar images even if the image is rotated, resized or even translated, In Figure 5.23, the query image has been rotated in different angles (90°, 180°, 270°), such rotations must be able to be detected in our CBIR system, because it tolerates the increments of 45° rotations.

In the results, the three rotated images have been retrieved with high similarity, which means that the rotation tolerant technique is working well and it is able to detect an image if rotated with incremental degrees of 45°.In the figure The rotated retrieved images are located in red rectangles.

Figure 5.23: Check rotation similarities

From the previous results in Figure 5.23 we can conclude that our rotation tolerant technique is retrieving rotated images as similar and not identical images (with similarity 100%).

Retrieving the rotated images with angles other than 90° and 180° will result in additional spaces on the corners, this will make the similarity decrease because the image is actually changed.

In Figure 5.24, another image has been picked, the image has been subject to some modifications, the image is flipped horizontally and vertically, it has been rotated in 45° increments and in other increments, also the query image has been subject to some noise using the Adobe Photoshop filters (the well-known image processing program).

Note that the modified images in figures 5.23, 5.24 and 5.25 all have been added to the Corel 1000 images database, then the query image is checked against the whole database (including the modified images).

Query Image (Similarity = 100%)

95.7951

Query image with some noise

92.3708

Query image with other noise

91.2179

Image rotated 90°

85.5672

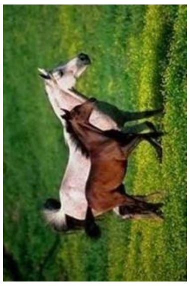
Image rotated 25°

85.2674

Image rotated -30°

83.4221

Image rotated 180°



81.6017

Image rotated -90°

76.4885

Image translated with keeping and cutting parts of the image

75.3779

Image translated

72.2475

Image rotated 40°

Figure 5.24: Check rotation and translation for modified horse image

In figure 5.25 another image has been used, the image was resized, rotated, translated and additional noise were added to the image, all the processed images have been added to the Corel-1000 database, then we retrieved the top 20 images from the whole images' database.

In figure 5.25 we can see that the first image is the original image, returned with similarity of 100%. All the processed noisy images are retrieved in our test (images in dashed and dotted red rectangles), and all the rotated images (shown in dashed green rectangles) retrieved too, the translated images with small translations also retrieved (images in the lined blue rectangles). The image shown in the orange (dotted) rectangle is the original image in the Corel-1000 database (our query image is clipped from this image), and the other images without rectangles are only similar images.

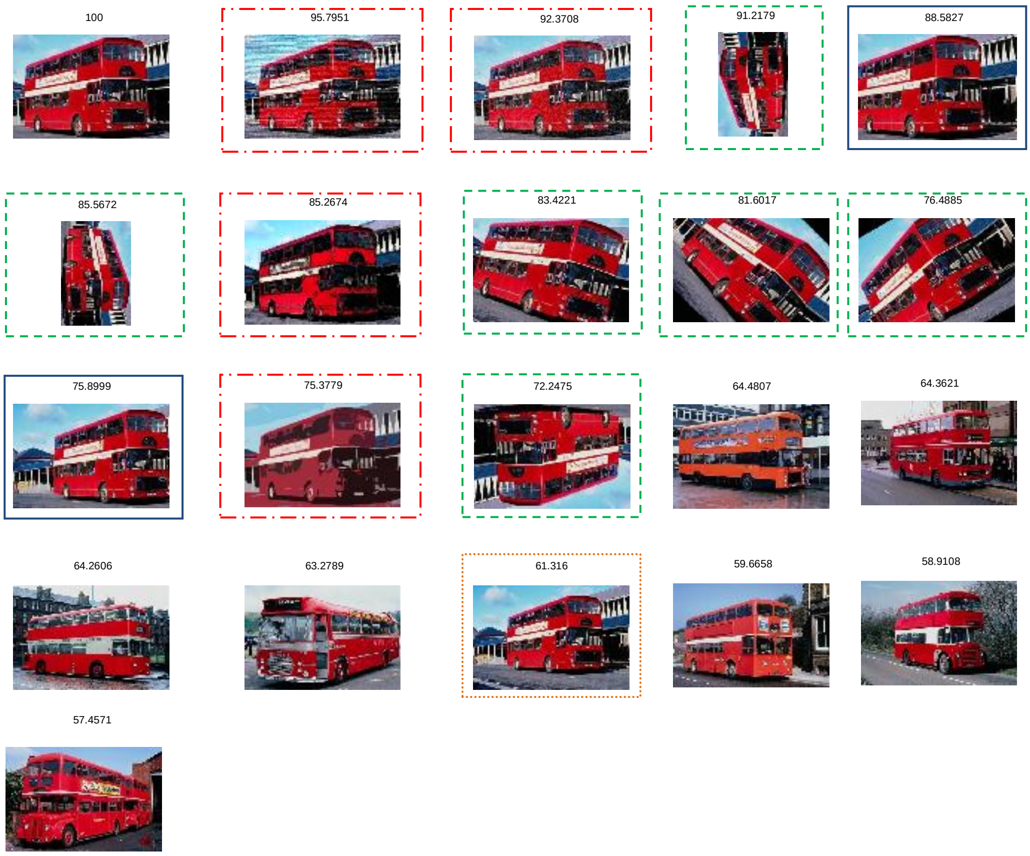


Figure 5.25: Bus image with different tests (rotation, translation, resize and additional noise).

From the results shown in figures 5.23, 5.24 and 5.25 we can conclude that our CBIR system can find rotated images with different rotation angles, even though the perfect retrieval is for 45° increments. Our approach didnot add specific solutions to retrieve translated images, but however the results in figures 5.25 and 5.26 showed that the distribution of the colors in the image might ignore little translations for the query image. The results also showed that our approach is able to retrieve images even if some noise were added.

The last interesting notice that if the image was cropped form another image (from the center) then our CBIR system will be able to find the new image, and if the image is resized the image will be retrieved as an identical or at least similar image with no problems.

5.7 Final results and Comparisons

In this section we document our results and we make comparisons with other CBIR approaches to evaluate our approach. Table 5.1 documents the comparisons between precision ratios for each type of images in the Corel-1000 Database. The table shows the comparisons between using the Euclidean similarity (first two rows) and our proposed similarity (last two rows), and each similarity is used with uniform and non-uniform color quantization. The results showed that the using of the non-uniform color quantization with our proposed similarity has enhanced the retrieval efficiency of the CBIR system.

The columns precision ratios are the percent of relevant images retrieved within the top 20 retrieved images.

Table 5.1: Precision Ratios Comparisons

	UCQ precision using ED	NUCQ precision using ED	UCQ Precision	NUCQ precision
Africa	0.264	0.328	0.539	0.756
Beaches	0.357	0.419	0.482	0.532
Buildings	0.411	0.527	0.544	0.613
Buses	0.357	0.729	0.715	0.878
Dinosaurs	0.905	0.917	0.869	0.9934
Elephants	0.504	0.859	0.643	0.881
Flower	0.614	0.827	0.688	0.855
Horses	0.731	0.759	0.879	0.938
Mountains	0.251	0.314	0.482	0.527
Foods	0.352	0.529	0.548	0.649
Average	0.4746	0.6208	0.6389	0.76224

From the previous results in Table 5.1 we can conclude that the non-uniform color quantization adapted in this approach has made better retrieval results in our CBIR system, and we have to note that the used uniform color quantization is an adapted technique that works smarter than the traditional uniform quantization technique, but eventually, our technique could overcome this smart adapted quantization technique.

Figure 5.26 illustrates the comparison in table 5.1 between the four tests (Euclidian distance and our proposed similarity for the uniform and non-uniform color quantization databases).

From Figure 5.26 we can see that the non-uniform color quantization with our proposed similarity measurement has the highest efficiency against the Euclidian distance and the uniform color quantization.

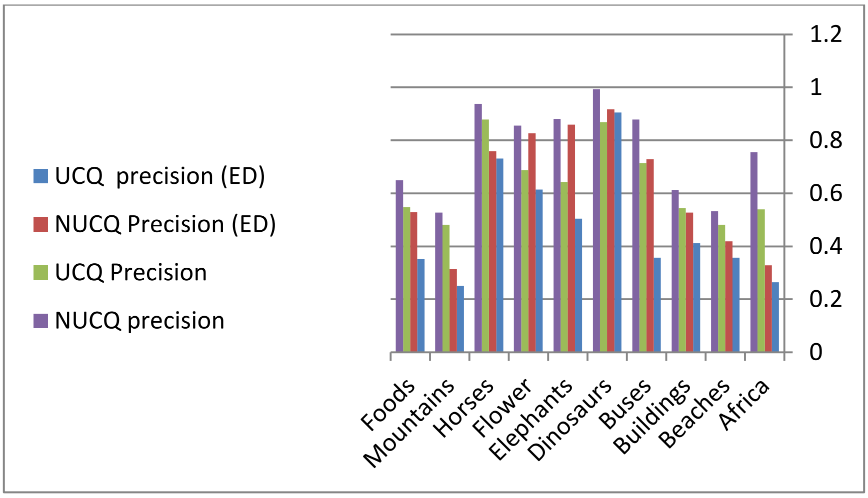


Figure 5.26: Comparison between the uniform, non-uniform, Euclidian and our proposed similarity

Table 5.2 elaborates the comparison of the proposed technique with other techniques used in previous researches, the results showed significant optimization in many types of images.

Table 5.2 compares our results with other approaches like IRM [56], Fuzzy Club [58], Signature Based [59], Geometric Histogram [60] and ERBIR [43].

Figure 5.28 illustrates the results of Table 5.2, from the figure we can see that our proposed approach could overcome the other approaches in many cases, the results are promising and they have overcomethe other CBIR approaches.

Table 5.2: Comparisons with other CBIR systems precisions for the top 20 retrieved images

	Fuzzy Club	IRM	Geometric Histogram	Signature Based CBIR	Modified ERBIR	Proposed Technique
Africa	0.65	0.47	0.125	0.42	0.7025	0.756
Beaches	0.45	0.32	0.13	0.46	0.57	0.532
Building	0.55	0.31	0.19	0.25	0.4925	0.613
Buses	0.7	0.61	0.11	0.83	0.8675	0.8788
Dinosaurs	0.95	0.94	0.16	0.92	0.9925	0.9934
Elephants	0.3	0.26	0.19	0.95	0.5725	0.881
Flowers	0.3	0.62	0.15	0.96	0.835	0.855
Horses	0.85	0.61	0.11	0.89	0.9275	0.938
Mountains	0.35	0.23	0.22	0.32	0.4975	0.527
Foods	0.49	0.49	0.15	0.28	0.655	0.6495
Average	0.559	0.486	0.1535	0.628	0.71125	0.76237

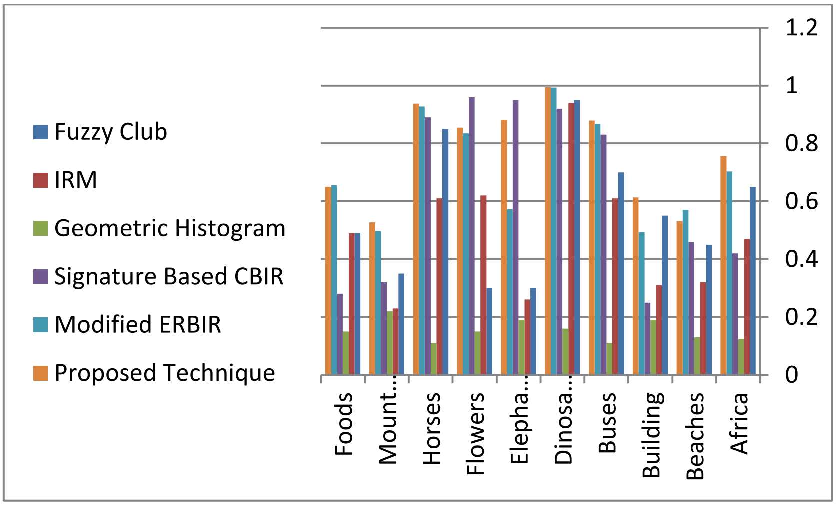


Figure 5.27 CBIR techniques comparison

In the previous comparisons we used the top 20 retrieved images to measure the precision ratios, in the next comparisons we will measure the precision ratio for different numbers of retrieved images, to make this comparison we will select 10 random images from each type of images in the Corel 1000 images database (the database includes 10 types of images, return to Figure 5.1), so the total number of selected images will be 100, for each type of images we will find the relevant images in the top 5, 10, 20 and 30 images, then we will take the average for every 10 images from the same type to consider this average as the final precision ratio for each type. Table 5.4 shows that we got high precision ratios when we make our measurements on the top 5 retrieved images, the precision ratios have decreased for larger numbers like 10, 20 and 30.

However, in our research we used the values in the third column (20 images), to compare our results with other researches, because the top 20 images retrieved were used in the other researches, and they are the most interesting retrieved images for the client (we suppose that).

Table 5.3: Comparison between different precision ratios for different number of retrieved images

	Top 5 relevant	Top 10 relevant	Top 20 relevant	Top 30 relevant
Africa	0.885714	0.814	0.756	0.417
Beaches	0.8	0.739	0.532	0.389
Buildings	0.76	0.693	0.613	0.429
Buses	0.92	0.892	0.8788	0.79
Dinosaurs	1	1	0.9934	0.908
Elephants	0.971429	0.843	0.4459	0.382
Flowers	0.914286	0.904	0.855	0.788
Horses	0.9824	0.972	0.938	0.81
Mountains	0.7388	0.7019	0.527	0.48
Foods	0.8926	0.7693	0.6495	0.52

Figure 5.28 illustrates the difference among several precision ratios for different numbers of results (The results shown in Table 5.3), and the figure shows that the higher the number of the retrieved images the lower the precision ratio.

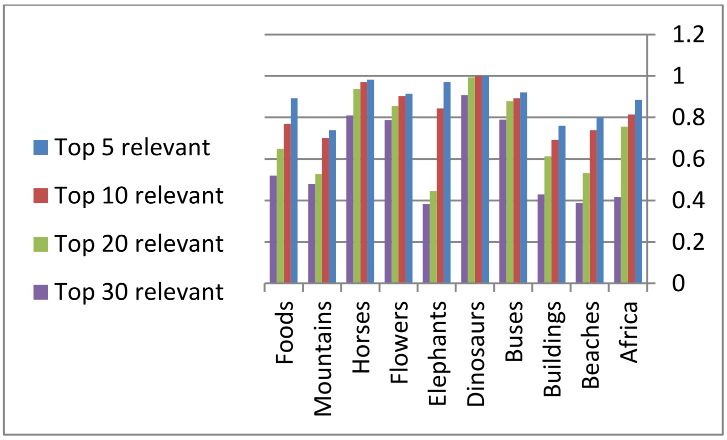


Figure 5.28: Comparison between different precision ratios for different number of results

Before the end of this section we can talk about the time used in different steps of this system (using an *i7* 2.4GHz CPU and 4GB machine):

1. Time used to convert all the Corel-1000 images into CAH signatures text file was 15 minutes for 1000 images (less than 1second per image).
2. Time used to compare a CAH signature with all the signatures was ≈ 0 seconds.
3. Time used to upload an image, compute its CAH and measure similarity with all the signatures and retrieve the result of top 20 images is less than 1 second.

The query image and all the images can be resized to predefined size (smaller than the original), this preprocessing step can decrease the time used in finding the CAH, and it has an intangible effect on the retrieval efficiency.

Table 5.4 shows some comparisons between our proposed approach and other results from different researches, but we have to notice that the time consumption depends on the used machine specifications.

Table 5.4: Average retrieval time for different approaches

Approach	Proposed Approach	ERBIR with SOM [43]	ERBIR without SOM [43]	Fractal Coding Signature [66]	MSD [22]
Retrieval time Average/second	0.9	0.23	0.78	1.766	Not available

In the end of this chapter we can conclude that representing an image as a color and angle histogram or signature is a very powerful approach in CBIR systems, and the comparisons proved that our approach using the proposed non-uniform color quantization technique and our proposed similarity measurement is a promising approach and can be used in many and different types of CBIR systems.

The results also showed that our approach is much better than using techniques like the color histogram in its traditional form, the results also showed that our approach is a scale invariant approach because the CAH is normalized, which makes the size of the image not a concern, and this solves the problem of using texture based or object based approaches. Another achieved goal is that our approach is able to tolerate specific rotations according to our angles' divisions, which make the system able to find the image if rotated with specific degrees or if flipped horizontally or vertically.

Chapter 6
CONCLUSION

6.1 Summary and Concluding Remarks

In this research a new idea is proposed to represent a colored image as a signature describes the color and angle distribution for each color within the image. The first step in this approach depends on quantizing the image into a number of colors, this step eliminates the number of the colors in the image from millions to dozens, which makes the other steps simpler and faster.

The quantization operation also makes the process more accurate, because it makes our system ignores fine gradients in the image, that will make the blue sky looks as one color instead of thousands of similar colors.

The quantization approach used in this research is the non-uniform color quantization technique, this technique divides the Hue plane into non-similar blocks, according to the distribution of each color, this approach also depends on the Saturation and Value to mark dark colors as black, and light colors as white, while colors' gradients between the previous two colors will be considered as gray color gradients, other colors with specific conditions will be considered as true colors.

The non-uniform color quantization technique used in this research is very close to the human eye vision system, which makes the CBIR system works more closer to the human eye system.

Comparisons between the CBIR system using this quantization technique and other traditional techniques showed that this technique is more accurate and simulates the human eye in its mechanism, the numbers showed good and interesting results using this approach.

The second step in our CBIR system divides the image into a number of regions, then, for each region, the number of each quantized color has to be found to make a histogram for each division, then a complete color and histogram CAH representation for the whole image will be the concatenation for all the generated histograms for each angle.

This CAH will make the CBIR system depends on the color and the location of the pixels in the image, this will overcome traditional color histogram systems that depend on the whole image as a unit.

We know that two images might look totally different, but they might have similar histograms, this is the main problem in the CBIR approaches that depend on the histogram, however, the proposed CAH has overcome this problem.

Another technique is used to make the CAH independent from the per-cell rotation, so if the division number one (which is the angle from 0° to 45°) rotated, that will not affect the CAH, this idea made the proposed CAH rotation tolerant, and images rotated with increments of 45° (our used angle) can be retrieved as similar or high relevant images.

To measure the similarity between two signatures we used a new proposed similarity measurement technique, the proposed technique showed results much more better than the traditional Euclidean distance.

The results and comparisons with other CBIR approaches showed that our proposed technique was efficiently able to detect similar images for different types, colors and sizes, the results showed that this technique could overcome many other CBIR techniques and can be used as a reliable and efficient technique for retrieving similar images depending on their contents.

The proposed technique is able to tolerate rotated or resized images, and rotated images in 45° increments are retrieved with high similarity values, identical images are retrieved with similarity of 100% which makes it more efficient in finding identical images.

The comparisons and results shows that our approach overcomes many other approaches like Fuzzy Club, IRM, Geometric Histogram Signature Based CBIR and Modified ERBIR, in almost all the tests.

In terms of time and space the technique was not consuming because the conversion of the images can be done in the background without impacting the client's time, and the representatives of the images are simple vectors that can be compressed with the traditional compression techniques.

The converted database can be clustered to get more efficient and time efficient retrieval for image retrieval systems with huge databases, after clustering the similarity measurements will be done between the query image representing vector and each centroid instead of measurements against all the images in the database.

6.2 Recommendations and Future Work

Another enhancements can be applied to optimize the work of this CBIR system is to use another technique that depends on shape or texture for example, for example we can use object detection approaches, then after finding the new objects within the image we can find the CAH for every object, that will make each image has one or more CAH signatures, and many-to-many relations will be measured between each couple of images.

More comparisons can be applied on the effect of different sizes on the precision ratio of the retrieved images, this can be tested by making many CAH databases with different sizes.

Other enhancements can be done by converting the image to a gray scale image, the gray image can be used as another image for another signature, and the query image will also be converted to a gray scale image and the similarity between the two images will be considered as another factor for the similarity between the original images.

Like the rotation-tolerant technique applied in this research we can apply the same idea to make a color-invariant CAH, in this approach we have to find the color with the highest response and to rotate this color to the left (instead of applying the rotation within the block), this rotation will make our CAH signature connected to the distribution and the response of the colors instead of the color itself, but it is not guaranteed that this will be convincing for the human visual system.

REFERENCES

[1] Moravec, Hans P, "Rover Visual Obstacle Avoidance,"International Joint Conference on Artificial Intelligence, pp. 785-790. 1981.

[2] Kato, Toshikazu, "Database architecture for content-based image retrieval," SPIE/IS&T 1992 Symposium on Electronic Imaging: Science and Technology, pp. 112-123. International Society for Optics and Photonics, 1992.

[3] Quellec, Gwénolé, Mathieu Lamard, Guy Cazuguel, BéatriceCochener, and Christian Roux, "Adaptivenonseparable wavelet transform via lifting and its application to content-based image retrieval," Image Processing, IEEE Transactions on 19, no. 1 (2010): 25-35.

[4] Murala, Subrahmanyam, R. P. Maheshwari, and R. Balasubramanian, "Local tetra patterns: a new feature descriptor for content-based image retrieval," Image Processing, IEEE Transactions on 21, no. 5 (2012): 2874-2886.

[5] He, Xiaofei, "Laplacian regularized D-optimal design for active learning and its application to image retrieval," Image Processing, IEEE Transactions on 19, no. 1 (2010): 254-263.

[6] Androutsos, Dimitrios, Konstantinos N. Plataniotis, and Anastasios N. Venetsanopoulos, "Image retrieval using the directional detail histogram,"Photonics West'98 Electronic Imaging, pp. 129-137. International Society for Optics and Photonics, 1997.

[7] D. Mahidhar, "MVSN, Maheswar a novel approach for retrieving an image using cbir,"International Journal of computer science and information technologies vol. 1, 2010.

[8] Del Bimbo, Alberto, and PietroPala, "Visual image retrieval by elastic matching of user sketches," Pattern Analysis and Machine Intelligence, IEEE Transactions on 19, no. 2 (1997): 121-132.

[9] Carson, Chad, Serge Belongie, Hayit Greenspan, and Jitendra Malik, "Blobworld: Image segmentation using expectation-maximization and its application to image querying," Pattern Analysis and Machine Intelligence, IEEE Transactions on 24, no. 8 (2002): 1026-1038.

[10] R. Venkata Ramana Chary, Dr. D. Rajya Lakshmi, Dr. K. V. N Sunitha, "Image Retrieval and Similarity Measurement based on Image Feature," IJCST Vol. 2, ISSue 4, 2011.

[11] Smeulders, Arnold WM, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain, "Content-based image retrieval at the end of the early years," Pattern

Analysis and Machine Intelligence, IEEE Transactions on 22, no. 12 (2000): 1349-1380.

[12] Antani, Sameer, RangacharKasturi, and Ramesh Jain, “A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video,” Pattern recognition 35, no. 4 (2002): 945-965.

[13] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, P. Yanker, “Query by image and video content: the QBIC system,”IEEE Computer 28, 1995

[14] Ma, Wei-Ying, and Bangalore S. Manjunath, “Netra: A toolbox for navigating large image databases,” Multimedia systems 7, no. 3 (1999): 184-198.

[15] Pentland, Alex, Rosalind W. Picard, and Stan Sclaroff, “Photobook: Content-based manipulation of image databases,” International Journal of Computer Vision 18, no. 3 (1996): 233-254.

[16] J.R. Smith, S.F. Chang, “Querying by Color Region Using the VisualSEEK Content Based Visual Query System, Intelligent Multimedia Information Retrieval,”AAAI Press, 1997.

[17] Liu, Guang-Hai, Lei Zhang, Ying-Kun Hou, Zuo-Yong Li, and Jing-Yu Yang, “Image retrieval based on multi-texton histogram,” Pattern Recognition 43, no. 7 (2010): 2380-2389.

[18] Palm, Christoph, “Color texture classification by integrative co-occurrence matrices,” Pattern Recognition 37, no. 5 (2004): 965-976.

[19] Liu, Guang-Hai, and Jing-Yu Yang, “Image retrieval based on the texton co-occurrence matrix,” Pattern Recognition 41, no. 12 (2008): 3521-3527.

[20] Huang, Jing, S. Ravi Kumar, MandarMitra, and Wei-Jing Zhu, “Image indexing using color correlograms,” U.S. Patent 6,246,790, issued June 12, 2001.

[21]Wang, Xingyuan, and ZongyuWang, “A novel method for image retrieval based on structure elements' descriptor,” Journal of Visual Communication and Image Representation 24, no. 1 (2013): 63-74.

[22] Liu, Guang-Hai, Zuo-Yong Li, Lei Zhang, and Yong Xu, “Image retrieval based on micro-structure descriptor,” Pattern Recognition 44, no. 9 (2011): 2123-2133.

[23] Liu, Zhuoxin, and JinshanJiang, “An approach to image retrieval using Distance-Angle Pixel Array,” Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference on, vol. 6, pp. 2579-2583. IEEE, 2010.

[24] J. Ashley, M. Flickner, J. Hafner, D. Lee, W. Niblack, and D. Petkovic, "The query by image content (QBIC) system," Proc. ACM SIGMOD Int. Conf. Manag. Data, 1995, p. 475.

[25] Bao, Qian, and Ping Guo, "Comparative studies on similarity measures for remote sensing image retrieval," Systems, Man and Cybernetics, 2004 IEEE International Conference on, vol. 1, pp. 1112-1116. IEEE, 2004.

[26] Bretschneider, T., R. Cavet, and O. Kao, "Retrieval of remotely sensed imagery using spectral information content," Geoscience and Remote Sensing Symposium, 2002. IGARSS'02. 2002 IEEE International, vol. 4, pp. 2253-2255. IEEE, 2002.

[27] T. Bretschneider and O. Kao, "A retrieval system for remotely sensed imagery," Proc. Int. Conf. Imag. Sci., Syst., Technol., 2002, vol. 2, pp. 439–445.

[28] Scott, Grant J., Matthew N. Klaric, Curt H. Davis, and Chi-RenShyu, "Entropy-balanced bitmap tree for shape-based object retrieval from large-scale satellite imagery databases." Geoscience and Remote Sensing, IEEE Transactions on 49, no. 5 (2011): 1603-1616.

[29] Raghunathan, Badrinarayan, and Scott T. Acton, "Content based retrieval for remotely sensed imagery," Image Analysis and Interpretation, 2000. Proceedings. 4th IEEE Southwest Symposium, pp. 161-165. IEEE, 2000.

[30] Ferecatu, Marin, and NozhaBoujemaa, "Interactive remote-sensing image retrieval using active relevance feedback," Geoscience and Remote Sensing, IEEE Transactions on 45, no. 4 (2007): 818-826.

[31] Tobin, Kenneth W., Budhendra L. Bhaduri, Eddie A. Bright, Anil Cheriyadat, Thomas P. Karnowski, Paul J. Palathingal, Thomas E. Potok, and Jeffery R. Price, "Large-scale geospatial indexing for image-based retrieval and analysis," Advances in Visual Computing, pp. 543-552. Springer Berlin Heidelberg, 2005.

[32] Tobin, Kenneth W., Budhendra L. Bhaduri, Eddie A. Bright, Anil Cheriyadat, Thomas P. Karnowski, Paul J. Palathingal, Thomas E. Potok, and Jeffery R. Price, "Automated feature generation in large-scale geospatial libraries for content-based indexing," Photogrammetric engineering and remote sensing 72, no. 5 (2006): 531.

[33] Li, Y., and T. Bretschneider, "Semantics-based satellite image retrieval using low-level features," Geoscience and Remote Sensing Symposium, 2004. IGARSS'04. Proceedings. 2004 IEEE International, vol. 7, pp. 4406-4409. IEEE, 2004.

[34] Samal, Ashok, Sanjiv Bhatia, PrasanthVadlamani, and David Marx, "Searching satellite imagery with integrated measures," Pattern Recognition 42, no. 11 (2009): 2502-2513.

[35]Newsam, Shawn D., and ChandrikaKamath, “Retrieval using texture features in high-resolution multispectral satellite imagery,” Defense and Security, pp. 21-32. International Society for Optics and Photonics, 2004.

[36] Harris, Chris, and Mike Stephens, “A combined corner and edge detector,”Alvey vision conference, vol. 15, p. 50. 1988.

[37] Zhang, Zhengyou, RachidDeriche, Olivier Faugeras, and Quang-Tuan Luong, “A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry,” Artificial intelligence 78, no. 1 (1995): 87-119.

[38] Schmid, Cordelia, and Roger Mohr, “Local grayvalue invariants for image retrieval,” Pattern Analysis and Machine Intelligence, IEEE Transactions on 19, no. 5 (1997): 530-535.

[39] Lowe, David G, “Object recognition from local scale-invariant features,” Computer vision, 1999. The proceedings of the seventh IEEE international conference on, vol. 2, pp. 1150-1157. Ieee, 1999.

[40] Cross, George R., and Anil K. Jain, “Markov random field texture models,” Pattern Analysis and Machine Intelligence, IEEE Transactions on 1 (1983): 25-39.

[41] M. Porat and Y. Zeevi., “The generalized Gabor scheme of image representation in biological and machine vision,” IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 10, no.4, pp. 452-468, July 1988.

[42] Ojala, Timo, Matti Pietikainen, and Topi Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” Pattern Analysis and Machine Intelligence, IEEE Transactions on 24, no. 7 (2002): 971-987.

[43] Abuhaiba, Ibrahim S I, and Ruba A A Salamah, “Efficient global and region content based image retrieval,”Int J ImageGraphics Signal Process 4, no. 5 (2012): 38-46.

[44] B.S. Manjunath, P. Salembier, T. Sikora, “Introduction to MPEG-7: Multimedia Content Description Interface,” John Wiley & Sons Ltd., 2002.

[45] Julesz, Bela, “Textons, the elements of texture perception, and their interactions,” Nature (1981).

[46] L. Chen, “Topological structure in visual perception,” Science 218 (4573) ,1982

[47] P.S.Suhasini, K.Sri Rama Krishna, and I. V. Murali Krishna, “CBIR using color histogram processing, ” Journal of Theoretical and Applied Information Technology, 2005-2009.

[48] Pass, Greg, and RaminZabih, "Histogram refinement for content-based image retrieval," Applications of Computer Vision, 1996. WACV'96., Proceedings 3rd IEEE Workshop on, pp. 96-102. IEEE, 1996.

[49] Mahmoudi, Fariborz, JamshidShanbehzadeh, Amir-Masoud Eftekhari-Moghadam, and Hamid Soltanian-Zadeh, "Image retrieval based on shape similarity by edge orientation autocorrelogram," Pattern recognition 36, no. 8 (2003): 1725-1736.

[50] Lowe, David G, "Distinctive image features from scale-invariant keypoints,"International journal of computer vision 60, no. 2 (2004): 91-110.

[51] Lei, Zhang, Lin Fuzong, and Zhang Bo, "A CBIR method based on color-spatial feature." TENCON 99. Proceedings of the IEEE Region 10 Conference, vol. 1, pp. 166-169. IEEE, 1999.

[52] Hadi A Alnabriss and Wesam M. Ashour, "Avoiding Objects with few Neighbors in the K-Means Process and Adding ROCK Links to Its Distance," International Journal of Computer Applications (IJCA), 28(10):12-17, 2011.

[53] Smeulders, Arnold WM, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain, "Content-based image retrieval at the end of the early years," Pattern Analysis and Machine Intelligence, IEEE Transactions on 22, no. 12 (2000): 1349-1380.

[54] Meskaldji, K., S. Boucherkha, and S. Chikhi, "Color quantization and its impact on color histogram based image retrieval accuracy," Networked Digital Technologies, 2009. NDT'09. First International Conference on, pp. 515-517. IEEE, 2009.

[55] Afifi, Ahmed J., and Wesam M. Ashour, "Content-Based Image Retrieval Using Invariant Color and Texture Features," Digital Image Computing Techniques and Applications (DICTA), 2012 International Conference on, pp. 1-6. IEEE, 2012.

[56] J. Li, J. Wang, and G. Wiederhold, "Integrated Region Matching for ImageRetrieval," Proceedings of the 2000 ACM Multimedia Conference , Los Angeles, October 2000, pp. 147-156.

[57] Krishnamoorthi, R, "A multi-resolution approach for rotation invariant texture image retrieval with orthogonal polynomials model," Journal of Visual Communication and Image Representation 23, no. 1 (2012): 18-30.

[58] R. Zhang, and Z. Zhang, "A Clustering Based Approach to Efficient Image Retrieval," Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'02), Washington, DC, Nov. 2002, pp. 339-346.

[59] D. Lakshmi, A. Damodaram, M. Sreenivasa and J. Lal, “Content based image retrieval using signature based similarity search,” Indian Journal of Science and Technology, Vol.1, No 5, pp.80-92, Oct. 2008

[60] A. Rao, R. Srihari, Z. Zhang, “Geometric Histogram: A Distribution of Geometric Configuration of Color Subsets,” Internet Imaging, Proceedings of the SPIE Conference on Electronic Imaging 2000, Vol. 3964-09, San Jose, CA, pp.91-101,Jan 2000.

[61] Jain, Anil K., and AdityaVailaya, “Image retrieval using color and shape,” Pattern recognition 29, no. 8 (1996): 1233-1244.

[62] Ojala, T., M. Rautiainen, E. Matinmikko, and M. Aittola, “Semantic image retrieval with HSV correlograms,” Proceedings of the Scandinavian conference on Image Analysis, pp. 621-627. 2001.

[63] Vadivel, A. K. M. S. S. A., A. K. Majumdar, and ShamikSural, “Performance comparison of distance metrics in content-based image retrieval applications,” Proc. of Internat. Conf. on Information Technology, Bhubaneswar, India, pp. 159-164. 2003.

[64] Lin, Chuen-Horng, Der-Chen Huang, Yung-Kuan Chan, Kai-Hung Chen, and Yen-Jen Chang, “Fast color-spatial feature based image retrieval methods,” Expert Systems with Applications 38, no. 9 (2011): 11412-11420.

[65] Manjunath, Bangalore S., J-R. Ohm, Vinod V. Vasudevan, and Akio Yamada, “Color and texture descriptors,” Circuits and Systems for Video Technology, IEEE Transactions on 11, no. 6 (2001): 703-715.

[66] Xiaoqing Huang, Qin Zhang, and Wenbo Liu, “A new method for image retrieval based on analyzing fractal coding characters,” Journal of Visual Communication and Image Representation, 24(2013)1, 42–47.