Libyan Academy – Misurata
School of Engineering and Applied Science
Department of Information Technology

Content Based Color Images Retrieval Method using Discrete Cosine Transform and Histogram

A Thesis Submitted in Partial Fulfilment of the Requirements for the Master Degree in Information Technology

By:

Sumaia Ali Eltomi

Supervised by:

Dr. Mohammed Mosbah Elsheh

2019

Acknowledgment

I thank Allah for his bounty, who enabled me to accomplish this work, praise be to Allah first and last. My thanks goes to my dear parents who have learnt me the honesty in saying and working, with the request of satisfaction.

I also extend my sincere appreciation and thanks to my supervisor **Dr. Mohammed Elsheh**, who has kindly accepted the supervision on this thesis. He has been credited for guiding me through this study, and his keenness to complete and demonstrate it as required. Allah rewards him and gives him all good.

Finally, I would like to thank all those who have provided support, assistance and encouragement, which helped me in accomplishing this study to the fullest, my friends and teachers.

Contents

List of Tables

List of Figures

List of Abbreviations

TBIR : Text-Based Image Retrieval

CBIR : Content-Based Image Retrieval

SBIR : Semantic-Based Image Retrieval

QBIC : Query By Image Content

CBVIR : Content-Based Visual Information Retrieval

CH : Color Histogram

DCT : Discrete Cosine Transform

ED : Euclidean Distance

MD : Manhattan Distance

CD : City Block Distance

RGB : Red-Green-Blue

HSV : Hue-Saturation-Value

HVS : Human Visual System

ARR : Average Retrieval Rate

EDBTC : Error Diffusion Block Truncation Coding

GLCM : Gray-level co-occurrence matrix

VQ : Vector Quantization

CHF : Color Histogram Feature

EHD : Edge Histogram Descriptor

ANN : Artificial Neural Networks

CNN : Convolutional Neural Networks

Abstract

The dramatic growth in digital data has resulted in an increment in the number of images that took and stored in databases in recently years.

For this reason, researchers are working on developing image retrieval methods to help in achieving better and accurate results. In this thesis, a color image retrieval approach was proposed based on images' content. This approach is based on extracting an efficient combination of low visual features in the image; color and texture.

To extract the color feature, color histogram was used, where the RGB color space was converted into HSV color space, then the color histogram of each space was taken. To extract the texture feature, DCT transformation was used, and DC coefficients are taken meanwhile neglecting AC coefficients.

To test the proposed approach a Corel 1-k dataset is used, which is widely used in CBIR systems. It contains 1000 color images and is divided into 10 categories, each category contains 100 images.

In the CBIR system, measuring similarity is very important for evaluating retrieval performance. The experimental results were analyzed on the basis of three similarity measures. MD similarity measure proved its efficiency in retrieval process compared with other similarity measures at both the execution time and retrieval accuracy. The accuracy of the system were evaluated using the precision and recall metrics.

The results were compared with some previous studies, and they were satisfactory. The results obtained from the proposed approach showed good results when considering precision measure in evaluation process. The precision was increased by (8.3%) rate compared to best result of previous studies.

الملخص

أدى النمو الهائل في البيانات الرقمية إلى زيادة في عدد الصور التي تم التقاطها وتخزينها في قواعد البيانات في السنوات الأخيرة.

لهذا السبب، يعمل الباحثون على تطوير طرق لاسترجاع الصور للمساعدة في تحقيق نتائج أفضل ودقيقة. في هذه الرسالة، تم اقتراح طريقة لاسترجاع الصور الملونة بناءً على محتوى الصور. يعتمد هذا النهج على استخراج مجموعة فعالة من الميزات المرئية المنخفضة في الصورة؛ اللون والملمس.

لاستخراج ميزة اللون، تم استخدام الرسم البياني للألوان، حيث تم تحويل فراغ اللون RGB إلى فراغ اللون HSV، ثم تم أخذ الرسم البياني للألوان لكل فراغ. لاستخراج ميزة النسيج، تم استخدام تحويل DCT، ويتم أخذ معاملات التيار المستمر مع إهمال معاملات التيار المتردد.

لاختبار النهج المقترح، تم استخدام مجموعة بيانات Corel 1-k، والتي تستخدم على نطاق واسع في أنظمة CBIR. تحتوي على 1000 صورة ملونة وتنقسم إلى 10 فئات، كل فئة تحتوي على 100 صورة.

في نظام CBIR، يعد قياس التشابه مهمًا جدًا لتقييم أداء الاسترجاع. تم تحليل النتائج التجريبية على أساس ثلاثة تدابير تشابه. أثبت مقياس التشابه MD فعاليته في عملية الاسترجاع مقارنةً بتدابير التشابه الأخرى في كل من وقت التنفيذ ودقة الاسترجاع. تم تقييم دقة النظام باستخدام مقاييس الدقة والاستدعاء.

تمت مقارنة النتائج مع بعض الدراسات السابقة، وكانت النتيجة مرضية، حيث أظهرت النتائج التي تم الحصول عليها من النهج المقترح نتائج جيدة عند النظر في قياس الدقة في عملية التقييم. تمت زيادة الدقة بنسبة (8.3٪) مقارنة مع أفضل نتيجة للدراسات السابقة.

CHAPTER ONE

INTRODUCTION

1.1 Introduction

In recent years, large collection of digital images have been created and dramatically increased, this includes many academic areas, trade/business, government sectors, medical applications, and traffic control. Technology has played a major role in many inventions, such as photography and television, which has facilitated the capture and communication of image data. The computer is the main engine of the revolution of photography, many technologies and devices that brought with it to capture, processing, storage and transfer images. The usage of these devices was limited until the mid-eighties because of the high cost of these devices. After the process of computerizing photography became easy and accessible to everyone, rapidly and quickly spread in fields that have been in one way or another depending heavily on the images in the process of communication between them, such as engineering, architecture and medicine.

Prior to the 1990s, access to digital images was limited. After the creation of the World Wide Web, it provided a great incentive for easy access to digital image data for users from anywhere in the world and enabled them to make optimal use of it in many areas[1].

A huge amount of digital images become accessible to the public usage. However, we may not be able to get benefit from them unless the review, inquiry, search and recovery process is efficient. The main problem is the difficulty of identifying the desired image in a large variety of image data set. While it is very practical and it is possible to select the desired image from a small set of images as soon as browsing, more effective techniques are needed with large sets of digital images[2] .

Image retrieval is one of the most important areas of research among researchers in the field of image processing. Researchers are focusing on new ways by which images can be easily, quickly and accurately retrieved and accessed from large databases. The retrieval mechanism and processing of the desired image from the database are important. At early stage, a major focus was placed on the process of retrieving images in what is now known as Text-Based Image Retrieval (TBIR), also known as concept based image retrieval[3].

Retrieving images based on TBIR with a small database is a straightforward way method. But the drawback of TBIR is a manual suspension, impossible and expensive task for a large database[4].

The methods used to retrieve images using text search techniques may suffer from inconsistencies between text and visual content if visual content is ignored as a classification guide. Attract Content-Based Image Retrieval (CBIR) which depends on identifying relevant images on visual content representation has been a constant concern in the past two decades[5].

Many content-based technologies have been developed in the last decade. CBIR is a field and a set of technology algorithms that enable the user to query the image databases using image content such as color, texture, and shape without using text attributes such as image name or other keywords.

1.2 Image Retrieval

Due to the rapid growth of the World Wide Web, and moving the world very fast because of the internet, image retrieval systems became important, also the retrieval mechanism processing of the desired image from the database has become very important.

The general target of image retrieval systems is: a system must be able to process language query, search must be performed among all image database, and system must take in account all the features of image. The image can be automatically indexed by summarizing their visual features in image retrieval systems[6].

With the focus on two major research communities: database management and computer vision, image retrieval can be defined as the task of searching for images in an database. Image retrieval system can be classified into three categories: text-based image retrieval (TBIR), content-based image retrieval (CBIR) and Semantic-Based Image Retrieval (SBIR),as shown in Fig 1.1.

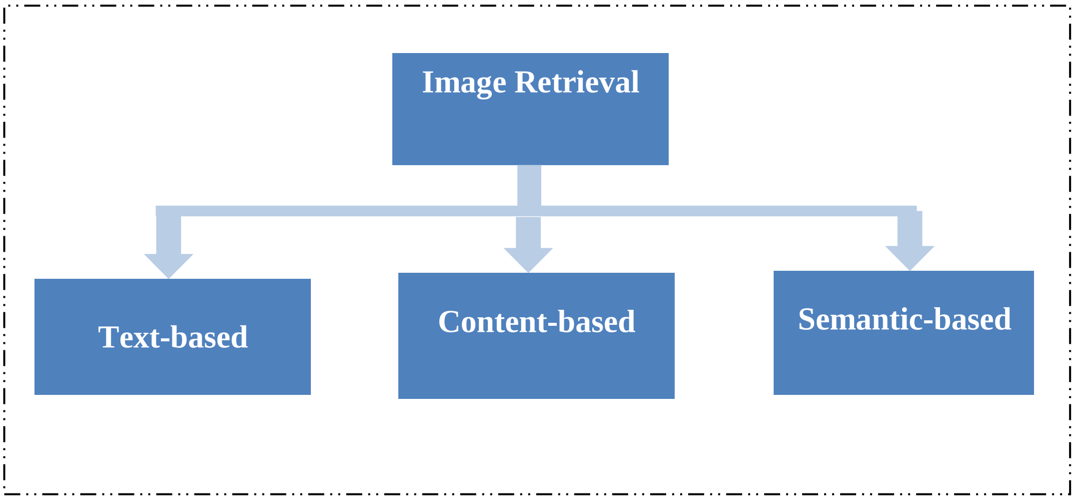

Fig 1.1: Classification of image retrieval system

1.2.1. Text-Based Image Retrieval (TBIR)

TBIR uses the text associated with the image to determine what the image contains. This text can be text around the image, image file name, hyperlink, image annotation, or any other text that can be linked to the image[3].

Metadata is used to index images. Google, Yahoo Search Engine Images are examples of systems that use this approach. These search engines have indexed more than a billion images.

These search engines often fail to retrieve relevant images, although they are fast and powerful, and this is for many reasons. First, there are a lot of irrelevant words in the surrounding text descriptions, resulting in a low resolution of image search. Second, the surrounding text does not seem to fully describe the semantic content of web images, resulting in a low call rate for image search[7].

The third problem is that the image annotation is never complete, a process that takes time because human cognition can lead to a number of errors. Therefore, there is a need for a new way to retrieve images where the human factor is mitigated from the annotation task and done automatically.

1.2.2. Content-Based Image Retrieval (CBIR)

CBIR is the modern image retrieval system. CBIR is also known as Query By Image Content (QBIC) and Content-Based Visual Information Retrieval (CBVIR). The term (CBIR) has been used widely for the process of retrieving images from a large collection of images based on visual features (color, shape, and texture) that is the signature of the image.

CBIR systems are used to extract image features, index those images using appropriate structures and efficiently process user queries providing the required answers[3].

CBIR uses the visual content to search images from large scale image database according to the user’s interest, it covers versatile areas, such as image segmentation, image feature extraction, representation, and mapping of features to semantics[6, 8] .

In a typical CBIR systems, the visual content of images in the database is extracted and described by multidimensional feature vectors. The color content of an image is the most widely used feature for CBIR, while texture and shape feature are also used to a lesser degree. A single feature is not enough to distinguish among a homogenous group of images. In such cases, either pairs of these features or all of them are used for the purpose of indexing and retrieval. Similarity matching, through matrices called similarity measures determine the degree of relevance of an image in a collection to a query. This is the key component of CBIR system because finding a set of images similar to the image the user had in mind is its primary goal[6].

CBIR involves the following four parts in system realization: data collection, build up feature database, search in the database, arrange the order and results of the retrieval images.

Fig 1.2 shows architecture of content based image retrieval system.

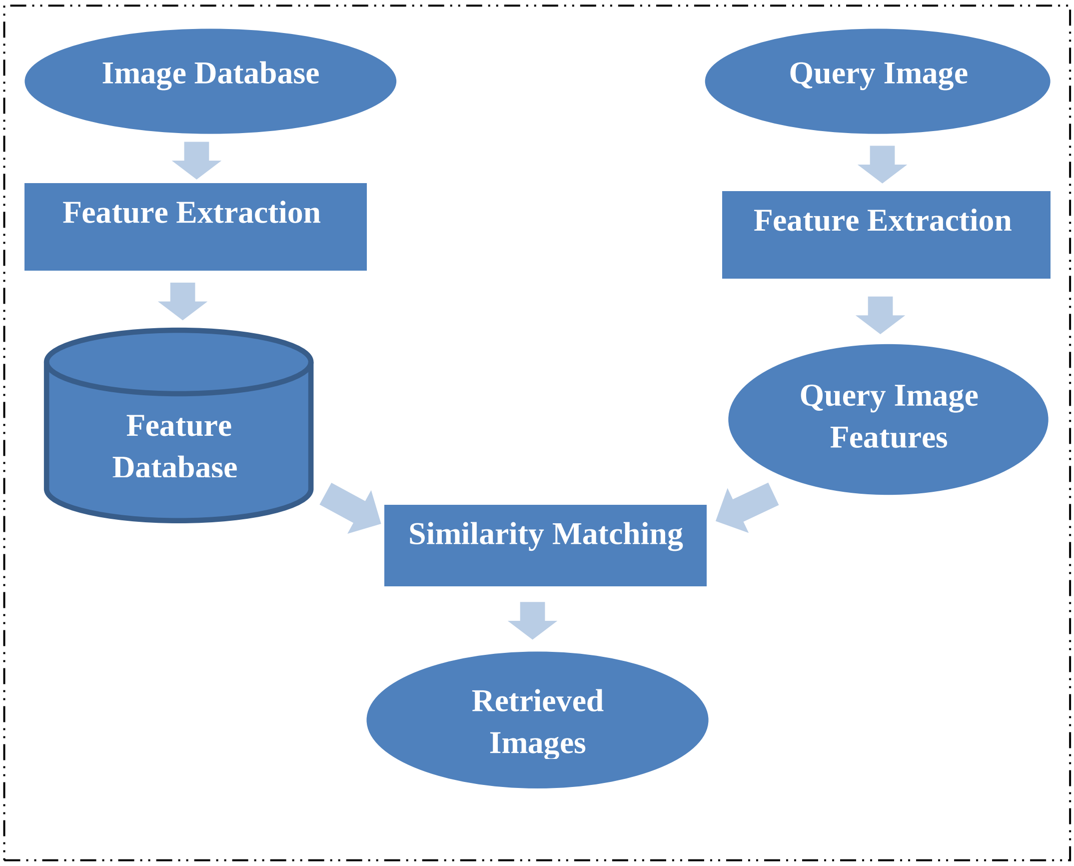

Fig 1.2: Architecture of content based image retrieval system

1.2.3. Semantic-Based Image Retrieval (SBIR)

Basically, the CBIR drawback lies in the semantic gap between high-level features and low-level features of the image. SBIR can be performed via extracting low-level image features to specify areas or objects with meaningful and interesting characteristics using similar features. Following, region attributes or objects will go through semantic process to acquire the description of the semantic images to be stored in the database. High-level concept is used to query Image retrieval[8, 9].

1.3 Image Features

The feature can be defined as capturing a specific visual property of an image. In general, picture features can be global or local. Global features describe the visual content of the entire image, where local features describe areas or objects (a small set of pixels) of the image content.

The feature is defined as an interesting part of the image, and features are used as the starting point for many computer vision algorithms. Because features are used as a starting point and basic priorities for the algorithms, the general algorithm is often only as good as the feature detector[3].

1.4 Features Extraction

Extracting the feature means getting useful information that can describe the image with its content. By means of image features, we mean special characteristics. Objects in the image can be considered as shapes that can be an advantage of the image. To describe the image, we must consider its main features. Specifying image features is an important step so that the image content can be well represented[1].

1.5 Color Histogram

One of the most visible and visual features of the image is color, it is the basic feature of the image contents because the human eye is sensitive to color and can recognize the images and objects contained in the image using color features. The color histogram (CH) is used to describe and represent colors in the image, which displays the pixel ratio of each color within the image. For the color histogram, the data range is divided into boxes of equal size and the number of pixels that have the same color value per bin is calculated[1].

In CBIR systems, the color histogram is used more frequently to calculate the distance criteria based on the chromatic similarity of each image, given its features such as high efficiency[3].

1.6 Discrete Cosine Transform

The discrete cosine transform (DCT) helps in separating the image into parts (or spectral sub-bands) of differing importance (with respect to the image's visual quality). The DCT is similar to the discrete Fourier transform, it transforms a signal or image from the spatial domain to the frequency domain[10]. For example, an image is a two-dimensional signal that is perceived by the human visual system. The DCT Transformation can be used to convert the spatial information into numeric data (frequency or spectral information), where the image’s information exists in a quantitative form (coefficients) that can be manipulated. In the literature the DCT has been widely used for efficient texture feature extraction[11].

1.7 Feature Similarity Measurement

Measurement of similarity is the process of approximation of the solution, based on the calculation of the function of the similarity between a pair of images. The result is a set of possible values. Once the database features are created, the user can give an image as input to the application to retrieve similar images from the database. The object vector is calculated for the query image using the same procedure. To calculate the similarity or congruence between the input query image and the database image, the difference between the feature vector of the query image and the vector of the database image is calculated using different distance metrics, such as Euclidean Distance (ED), Manhattan Distance (MD) and City block distance (CD)[12, 13] .

Measuring similarity is another important issue in CBIR where the query image is compared with other database images for similarity.

1.8 Problem Statement

The most common problem for retrieving images, is to extract image information to match it with images stored in a database containing large number of images. In order to extract texture and color attributes in retrieval systems, statistical comparisons are used, but in these ways the calculations may be very complex especially if there is a need to cover a wide range of data. When using these calculations for image analysis, this requires large storage space, and a long time to calculate the image attributes matrix.

The main research issue is to develop CBIR approach, which is based on extracting image information (color and texture) in an effective way that improves the search and retrieval of images, and try to overcome the problems facing these systems.

1.9 Goal and Objectives

This dissertation aims to develop an image retrieval approach based on the color histogram and DCT techniques to extract image's information according to the color and texture features of the retrieved images, to enhance the efficiency of CBIR systems in terms of the accuracy of retrieved data. The mentioned aim will be achieved by considering the following objectives:

1. To explore the area of image retrieval approaches based on the color histogram and DCT algorithms.
2. To develop an approach to retrieve images based on the color histogram and DCT techniques.
3. To implement a prototype system that put in action the proposed approach.
4. To evaluate the obtained results from the prototype system by comparing them with the results of previous studies.

1.10 Thesis Structure

The rest of this dissertation is structured as follows:

Chapter 2: Presents the related works regarding to image retrieval techniques in general as well as a detailed discussion about relevant research papers regarding retrieving digital images based on color histogram and DCT.

Chapter 3: Architecture of the proposed approach is described in detail to clear out how the color histogram and DCT is used to enhance for color images retrieval.

Chapter 4: Analysis and discussion of findings is explained in this chapter.

Chapter 5: Conclusions drawn and suggestion possible directions for further research is given in this chapter.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

The main reason behind image feature extraction is to discover image properties that describe the image in clear and precise manner. Feature extraction plays a very crucial role in image processing, especially when we need to make adjustment in the process. Since frequency conversions is widely used in image processing, extracting the required features from the image frequency information rid out the need to add spatial domain techniques which puts extra process cost. DCT is considered a very useful tool in frequency conversions, it suitable to extract low-level properties directly from DCT data in an image.

Histogram-based features commonly used to match and retrieve images from color histograms to more complex histograms. The histograms can be easily and quickly extracted. They are invariant to rotation and robust to occlusion and changes of view. For these reasons, histograms are the most popular features used for image matching[14]. Combining color and texture content provides a powerful feature set to restore color images.

In this chapter, the extraction and representation of image features will be discussed, focusing on extracting the color and texture feature based on the color histogram and DCT transform, and reviewing the related works.

2.2 Visual Features Representation

The importance of features extraction in image engines is very obvious. It helps in finding or search matching features from the database. The visual features that CBIR trusts including shape, semantic elements, structure, texture and color[15]. However, the proposed work deals with color and texture features, the rest of other features, are beyond of this study.

2.2.1 Color Feature

Color is a grasp depends on the human visual system's response to light and it interacts with our eyes and brain.

Color features are the basic components of image content, and are widely used in image processing, because they provide valuable information about images, so they are the most common visual features of CBIR. The main key issues in color extraction feature are color space, quantitative color, selection and similarity function[3].

2.2.1.1 Color Feature Extraction Techniques

The color feature can be represented by several methods, such as color histograms, color moments, color correlograms, coherence vectors, etc. The most commonly used descriptors of color are:

- **Color Moments**

  Color moments are the statistical moments of the probability distributions of colors and have been successfully used in many retrieval systems, especially when the image contains just the object. The color moments feature extraction process involves computation of mean and standard deviation of each component. These stored feature vectors are then compared with the feature vectors of the query image [13, 16].

- **Color Correlogram**

  To integrate spatial information with color histograms a color correlogram technique is used. This method is based on the combination of spatial information in the encoded color and can be used to describe the global distribution of local spatial correlation of colors and it is easy to calculate[13].

- **Color Histogram**

  Color histogram is the main way used to representing the color information of images in CBIR systems. Statistically, a color histogram is a way to approximate the joint probability of the values of the three color channels. Splitting the range of the data into equally sized bins is the most common way to form the histograms. For each bin, the number the colors of the pixels in an image that fall into each bin are counted and normalized to total points, which supply us the probability of a pixel falling into that bin[16].

However, several academic works have been done in the last view years related to CBIR using a color histogram. Among of them are:

The authors in [17] discussed the effectiveness of using the global HSV color space histograms of images as the descriptors in image clustering. Both the Red-Green-Blue (RGB) and Hue- Saturation-Value (HSV) color spaces define a method of uniquely specifying colors via three numbers. Color has been taken as the property for searching. For efficient way of searching, local histogram searching has been used. So, it has advantages than global histogram. The HSV global histograms are calculated for all the images to reduce the dimensions of the image descriptor vectors using Principal Component Analysis and calculating the similarity measures between the images. Once the histograms have been created, Euclidean distances are calculated for comparing the histograms of the images. The efficiency of this system is also measured by calculating precession and recall values.

In [18] the researchers proposed a CBIR system based on a color histogram feature. To compare the histogram and find the errors for that histogram, if the error is beyond the threshold then the images will not retrieved, otherwise images will be retrieved. After extracting the color histogram feature for database images and the query image, Euclidian Distances between the feature vectors of the query image and the feature vectors of images in the database are calculated as a similarity measure, then they apply threshold. For testing the performance of each proposed CBIR technique, five queries are fired on the generic image database of 650 images variable size and across five categories images databases. The query and database image matching is done using Euclidean distance. Precision and recall are used as statistical comparison parameters for the proposed CBIR techniques. The result is depending on the color present in image. If it is proper color image then good results are obtained.

Mohammed and Dawood in [19] suggested three ways to retrieve image from the database of images, all of them depending on color histogram, a histogram of prime value and color image moment by adding some extra features to increase the efficiency of work. In image retrieval based on color histogram method, the histogram is divided into sub-block histogram with 17 blocks, each block contains 15 grayscale colors for each of the RGB component of each image read from the database. Thus each image has three histograms associated with it. In image retrieval using the Prime value of color histogram, the frequency of the primes number between 0-255 is taken for all images in database and query image, then applying the Euclidian Distance between two features vector for each color (R, G and B). In image retrieval using color moment, the histogram value for three band (Red, Green and Blue) computes the set of moment (Mean , Entropy , Variance and Standard Deviation) of the color image for each band and applying the measurement distance to check which image is more closer to the query image from database. Experiment results show that the Prime algorithm is relatively easy, and it is effective among the other three techniques.

The authors in [20] presented an effective image retrieval method which is based on the color feature. Three dimension color space HSV is used and a (16:4:4) non-uniform quantization method is adopted in which H vector is divided into 16 values and S, V is divided into 4 values separately. The Minkowski distance is used to compare only the same bins between color histograms. For training purpose, almost 700 images have been used for populating the database. For each image, a 3-D histogram of its HSV values is computed. At the end of the training stage, all 3D HSV histograms are stored in the same file. For an image set of 606 images, the average of retrieval time was four seconds which is very fast.

2.2.2 Texture Feature

The texture refers to visual patterns consisting of entities or regions with subtypes, with homogeneous characteristics that are not caused by only one color or intensity. The texture is a property that represents the surface and structure of the image. It is a natural property of almost all surfaces. It contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment. Texture can be seen as a congruence grouping in an image[16, 21]. Fig 2.1 shows some types of textures.

Fig 2.1: Examples of Texture

2.2.2.1 Texture Feature Extraction Techniques

Texture representation techniques can be classified into three categories:

a) Statistical Techniques

They characterize texture using the statistical properties of the gray levels of the pixels comprising an image. Generally, in images, there is the periodic occurrence of certain gray levels. The spatial distribution of gray levels is calculated. Texture is measured statistically using a moving window throughout the image. Statistical operators including skewness, kurtosis, variances, standard deviation, maximum and mean, Euclidean distances are used for texture analysis.

b) Structural Techniques

characterize texture as being composed of texels (texture elements). These texels are organized ordinarily on a surface according to several definite arrangement rules.

c) Spectral Techniques

They are based on properties of the Fourier spectrum and depict global periodicity of the grey levels of a surface by recognizing high-energy peaks in the Fourier spectrum. The spectral approach to texture analysis deals with images in the frequency domain. Consequently, this method needs Fourier transform to be carried out on the original images to obtain their corresponding representations in the frequency space[16].

There are many mathematical transforms that are used in texture representation. The discrete cosine transform is remarked to be the best in image power compression in very few conversion coefficients. The DCT has been widely used for efficient texture feature extraction.

2.2.2.2 Texture feature extraction using DCT

In sake of efficient texture feature extraction, some DCT coefficients are used in the compressed domain as the feature vectors[22]. The DCT coefficients are acquired covering different spectral bands, to gain a fast feature extraction for the compressed domain. For texture images, much of the signal energy lies at low frequency components, which appear in the upper left corner of the DCT[23]. Texture features can be defined as the spectrum energies in different localizations of a local block.

The DCT decomposes the signal into underlying spatial frequencies, which then allow further processing techniques to reduce the precision of the DCT coefficients consistent with the Human Visual System (HVS) model. The DCT coefficients of an image tend themselves as a new feature, which has the ability to represent the regularity, complexity and some texture features of an image and it can be directly applied to entire image data or to sub-image of various sizes in the compressed domain[22].

However, several academic works have been done in the last view years related to CBIR using a DCT technique. Among of them are:

The study in [24] introduced a CBIR system based on two different approaches, DCT and DCWT. In this system, the image is divided into R, G and B color space. The plane of image is divided into four blocks of all equal size, then the row mean vectors for each block is calculated. DCT is applied overall row mean vectors of each block of each plane of the all the database images and DCT feature database is prepared. Similarly, the DCT wavelet applied over all row mean vectors of all four blocks of each plane of all database images, and new DCT wavelet feature database is prepared. Feature extraction of query image is done in the same manner as it does for the database images, and Euclidean Distance is used as the similarity measure to compare the image features. Obtained results are indirectly compared with the traditional parameters precision and recall. On the basis of comparison of this work with existing systems, they found that results are better in terms of similarity retrieval and also in terms of computational time required.

The authors in [25] presented a simple but effective approach to construct a descriptor from DCT coefficients for image retrieval, which selects part of DCT coefficients inside each block to construct AC-Pattern and using DC coefficients between neighboring blocks to construct DC-Pattern . The luminance normalization method adopted to eliminate the effect of luminance variations. Then, a 4*4 block DCT transform is used to get 1 DC coefficient and 15 AC coefficients for each block. For each block, select 9 AC coefficients to construct AC-Pattern, and use DC coefficients of the block itself and DC coefficients of its 8 neighboring blocks to build DC-Pattern. Finally, using the concatenation of AC-Pattern histogram and DC Pattern histogram as the descriptor of the image to do the retrieval task. Experiments are done on ORL face database (AT&T Laboratories Cambridge) and VisTex texture database (Media Laboratory MIT), and to evaluate the performance, the average retrieval rate (ARR) is used. Compared to other methods, the experimental results show higher performance to the proposed method compared to classical and state-of-art methods.

In [26] Hemalath Proposed a method that uses the shape as a feature to be extracted from (Statistical Region Merging) SRM algorithm and from DCT. The three feature databases are extracted as edge images by using SRM and DCT and the DCT images itself. Feature databases are considered from three different processing of the query image. The first feature database is considered from the border images extracted using SRM, this algorithm focuses on regions where it segments the given image by merging the similar colors together. The second feature database is generated by applying DCT on the query image, and the third feature database is generated by obtaining the edge images from DCT by using Sobel in Black and White images. Thus the processing takes place in three feature databases. The work was done on database consists of 186 leaf images with three different categories, the similarity measurement is given by RGB projection which determines the size of the image and compares the images in the database with the query image. The proposed work contributed much towards the accuracy by treating the images in three different feature databases and found the similarity between the images in analogy with the query image indicating the percentage of similarity between the images.

The researchers in [27] presented two grading retrieval algorithms based on DCT compressed domain and DWT compressed domain, respectively. Firstly, they use a 2-level grading image retrieval algorithm to realize image retrieval based on DCT compressed domain. Also they use a new dynamic color space quantization algorithm based on color distribution to improve retrieval accuracy and reduce dimensions of the histogram. The work is conducted on three stages:

The first level, feature vector is obtained by using the mean and standard deviation of low-frequency information as the texture feature vector. The distance is used to measure the similarity between the first level feature vectors in DWT domain, like the first level feature vector in DCT domain.

In the second level, the features are extracted from high frequency of wavelet sub-bands which describes the details of images (horizontal edge, vertical edge, and diagonal edge). Thus, it possible to use the mean and standard deviation of high-frequency information as the texture feature vector to improve retrieval accuracy further.

In the third level, in order to improve retrieval accuracy, they use fast wavelet histogram techniques to construct wavelet histogram to describe texture feature of images further. Distance was used to measure the similarity between the third level feature vectors. The obtained results are evaluated by using the performance measurement methods, namely precision, and recall. Experiments made on a dataset consisting of 600 color images. The results show clearly that two grading image retrieval algorithms work better than other algorithms. Store memory is reduced and retrieval accuracy is improved.

2.3 Query by Color and Texture Content

Several papers discussed the issues surrounding digital image retrieval by integrating color and texture features and enhanced by many of the techniques proposed in the literature. The reminder of this chapter introduces some of them.

The authors in [28] presented a novel approach for CBIR by combining the color and texture features. The texture and color features are extracted through wavelet transformation and color histogram. The histogram is applied to extract color features using (8,8,8) color quantization bin, and discrete wavelet transform to extract texture features. Haar wavelets is used to compute feature signatures because they are the fastest to compute. Then the similarity matrix of the query image and the image present in the database is calculated by Histogram Intersection Distance method, and the performance of retrieval of the system is measured in terms of its recall and precision. The experiments are performed on a general-purpose WANG database containing 1000 images. The experimental result shows that the proposed method outperforms the other retrieval methods in terms of average precision. Also, the whole indexing time for the 1000 image database takes 5-6 minutes.

The researchers in [29] proposed a method for image retrieval based on color, texture and edge descriptor features which require very low computational complexity. The Error Diffusion Block Truncation Coding (EDBTC) compresses an image in an effective way by incorporating the error diffusion kernel to generate a bitmap image. YCbCr color space is used for encoding RGB information, then they characterize the edges by Binary Histogram Feature. Finally, Gray-level co-occurrence matrix (GLCM) is used to extract a number of texture features. Four second order features namely inverse difference moment, correlation, angular second moment and entropy are computed. High discrimination accuracy is provided by these four measures. The features of both query image and database images are compared based on Euclidian Distance. The retrieval accuracy is measured using the average precision, average recall value over all query images.The proposed method provides higher average precision while preserving the low computational complexity, also provides performance gain of average retrieval time better than other methods.

Another study in [30] presented a novel approach called Error Diffusion Block Truncation Coding (EDBTC) to extract the texture and features of an image. A feature descriptor obtained from a color image is constructed from the EDBTC encoded data by incorporating the Vector Quantization

(VQ). The two methods introduced are Color Histogram Feature (CHF) and Bit Pattern Histogram Feature. The CHF effectively represents the color distribution within an image, while the BHF characterizes the image edge and texture. The successfulness of the proposed EDBTC retrieval system is measured with the precision, recall, and Average Retrieval Rate (ARR) value. Experimental result shows, the proposed indexing method outperforms the former BTC-based image indexing and the other existing image retrieval schemes with natural and textural datasets.

In [31] the researchers proposed a new CBIR technique to fuse color and texture features. Color Histogram (CH) is used to extract a color information. Texture features are extracted by DWT and Edge Histogram Descriptor (EDH). This technique is applied through several steps. First, it must apply a low pass filter to preserve the low frequency and high pass filter to preserve the high frequency of an image. Then, applying four level DWT to image and construct a feature vector for the first two moments those are the mean and standard deviation. Finally, the feature vector is constructed by Edge Histogram Descriptor (EHD). Query image is taken and the previous steps are repeated to build a feature vector based on color and texture feature. The distance between the query image and database image is calculated using Manhattan distance. The work was evaluated using Corel l-k dataset. To examine the accuracy of the other proposed systems, precision and recall methods are used that provides a competitive and efficient result. The experimental results show that, the proposed method outperforms with existing CBIR systems.

CHAPTER THREE

RESEARCH METHODOLOGY

3.1 Introduction

To query the images from large image databases, CBIR uses visual image features such as color, shape and texture to retrieve desired images based on user's interest for a given input query image.

Based on pixel values for image and using specific techniques these features are extracted directly from the image then stored on storage media.

Searching on desired image depends on matching process between the query image features and the image features in the database. These features are calculated and used to identify and retrieve images in the database that closely match the query image. The color feature is one of the most reliable visual features in image retrieval. This feature is extracted using different methods, the most important of which is the color histogram. Also the texture is a powerful and reliable feature in the retrieval process, it can be used with color feature to improve image retrieval performance[32].

This chapter presents the proposed method of color images retrieval system based on color histogram and DCT techniques for color and texture features extraction, then applying Manhattan Distance, Euclidean Distance and Mean Square Error to measure the similarity of feature vectors in features database and feature vector for the query image.

3.2 Thesis Approach

The Proposed retrieval approach goes through several steps as shown in Fig 3.1.

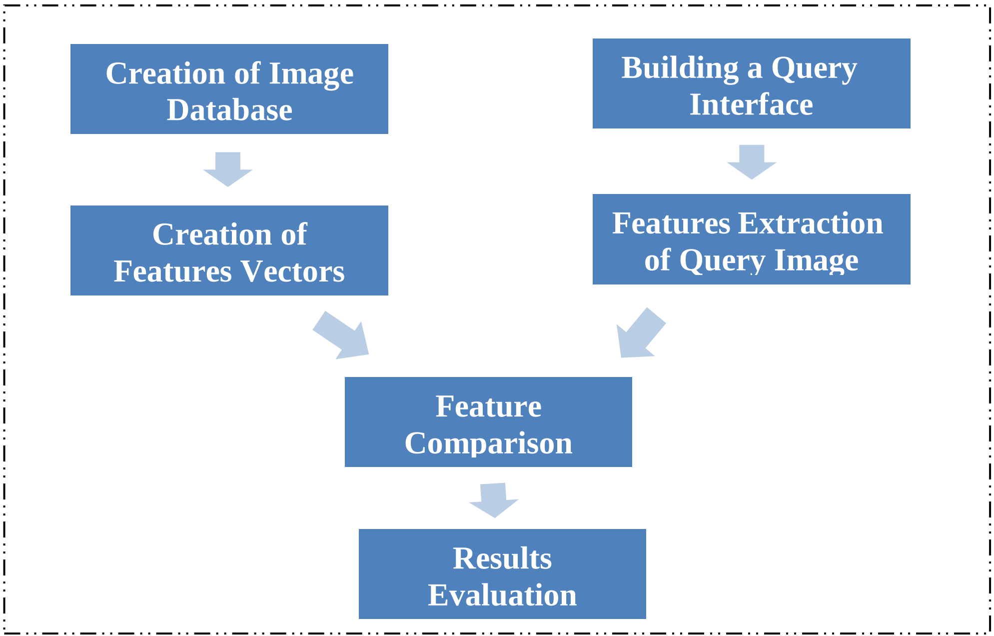

Fig 3.1 Proposed Retrieval Approach

- **Creation of Image Database**

  Creation of RGB image database, and data will be classified, then implementation of some pre-processing to standardize the dataset is performed.

- **Creation of Features Vectors**

  Applying histogram for color feature extraction and DCT for texture feature extraction.

- **Building a Query Interface**

  In the query Interface, a query image is input, and its feature vector is extracted by applying the same steps used to create the features vectors.

- **Feature Comparison**

  Applying measurement of the similarity of feature vectors in features database and the feature vector of the query image by MD, ED and MSE similarity measures.

- **Results Evaluation**

  The evaluation of the proposed work is carried out on based of returning the most relevant images that are similar to the query image, and the common equations of precision and recall.

3.2.1 Feature Vector Generation

In CBIR systems, feature vectors are generated from important information in the image such as color and texture, because they have a significant impact on the overall appearance of the image. Some systems use directly pixel information of images, for example RGB values or gray level information.

This proposed study is based on color histogram and DCT techniques for features extraction and creation the attributes vectors.

3.2.2 HSV Vector Generation

When working with color properties in images, it is important to look at the color representation and human color perception. For color representation, many color spaces have been developed. RGB color space is being the most widely known technique. However, for image retrieval purposes, RGB color data is often converted to another color system that is more adaptable to the human visual system, such as HSV (Hue - Saturation – Value)[33].

The HSV provides the perception representation according with human visual feature. The HSV model, defines a color space in terms of three constituent components: Hue, the color type range from 0 to 360 relative to the red primary at 0°, passing through the green primary at 120° and the blue primary at 240°, and then back to red at 360°. Saturation, the "vibrancy" of the color: Ranges from 0 to 100%. Value, the brightness of the color: it ranges from 0 to 100%. The HSV color space is used instead of the RGB color space due to two reasons: the lightness component is independent factor of images and the components of hue and saturation are so closely link with the pattern of human visual perception[34].

The conversion process of RGB to HSV color space is defined in Equations 1,2 and 3.

$$H = cos^{-1} \frac{\frac{1}{2}[R-G]+[R-B]}{\sqrt{(R-G)^2-(G-B)(R-B)}} \quad (1)$$

$$S = 1 - \left(\frac{3[\min(R,G,B)]}{R+G+B} \right) \quad (2)$$

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

To generate the color histograms, the color space is quantized into a finite number of discrete levels. Each of these levels becomes a bin in the histogram. The color histogram is then computed by calculating the number of pixels in each of these discrete levels.

Each histogram bin corresponds to a color in the used quantized color space. A color histogram for a given image is represented by a vector as shown in Equation 4.

$$H = [H[0], H[1], H[2], H[3], \ldots, H[\boldsymbol{i}], \ldots, H[\boldsymbol{n}]] \quad (4)$$

Where $\boldsymbol{i}$ is the color bin in the color histogram and H[$\boldsymbol{i}$] represents the number of pixels of color $\boldsymbol{i}$ in the image, and $\boldsymbol{n}$ is the total number of bins used in the color histogram. The normalized color histogram is calculated as displayed in Equation 5.

$$H' = [H'[0], H'[1], H'[2], H'[3], \ldots, H'[\boldsymbol{i}], \ldots, H`[\boldsymbol{n}]] \quad (5)$$

Where $H'[\boldsymbol{i}] = \frac{H[i]}{p}$, and $\boldsymbol{p}$ is the total number of pixels of an image[32].

The histogram of an image in a HSV space color is shown in Fig 3.2.

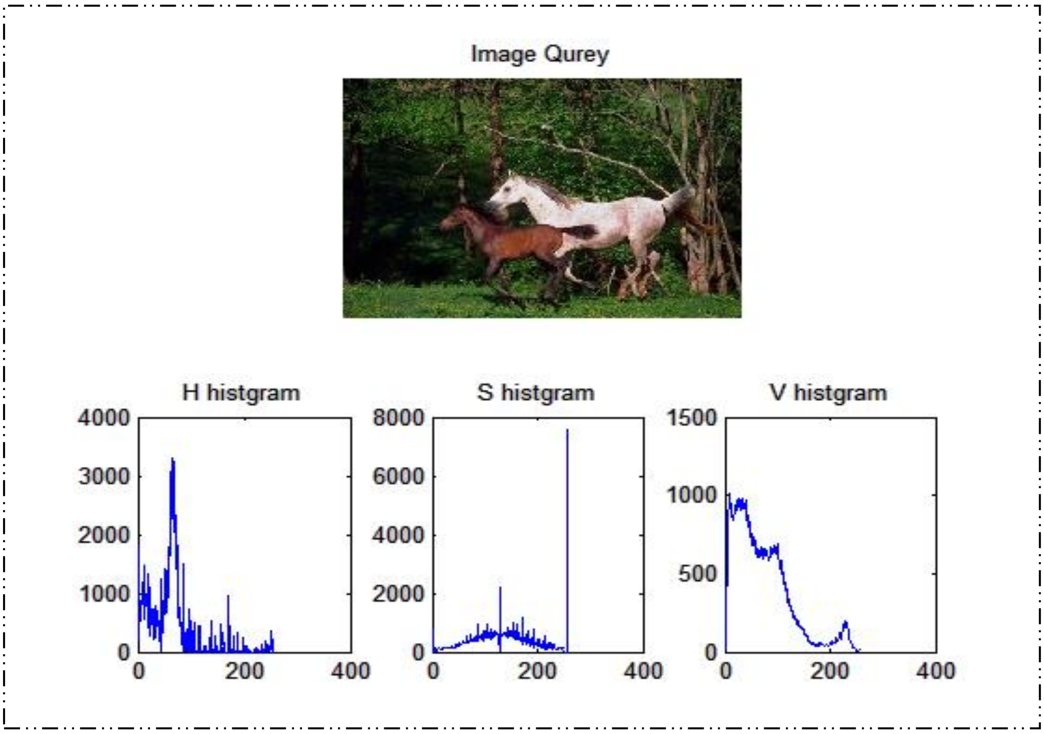

Fig 3.2 An image and its histogram

The algorithm for color feature vector generation is shown in Fig 3.3 and formulated in these following steps:

Step 1: Read the image.

Step 2: Convert RGB color space image into HSV color space.

Step 3: Color quantization is carried out using color histogram by assigning eight levels for each to Hue, Saturation and Value to give a quantized HSV space with 8*8*8=512 histogram bins.

Step 4: Histogram is obtained by dividing the pixel which represents the color on the total number of pixels.

Step 5: Compute HSV histogram for all color intensities in the image.

Step 6: Store the value of bins of color histograms in three vectors, one for each HSV color space.

Step 7: Repeat step1 to step 6 on all images in the database.

Step 8: All these color histograms are combined after then in one vector with the values of DCT to search for similar images in database.

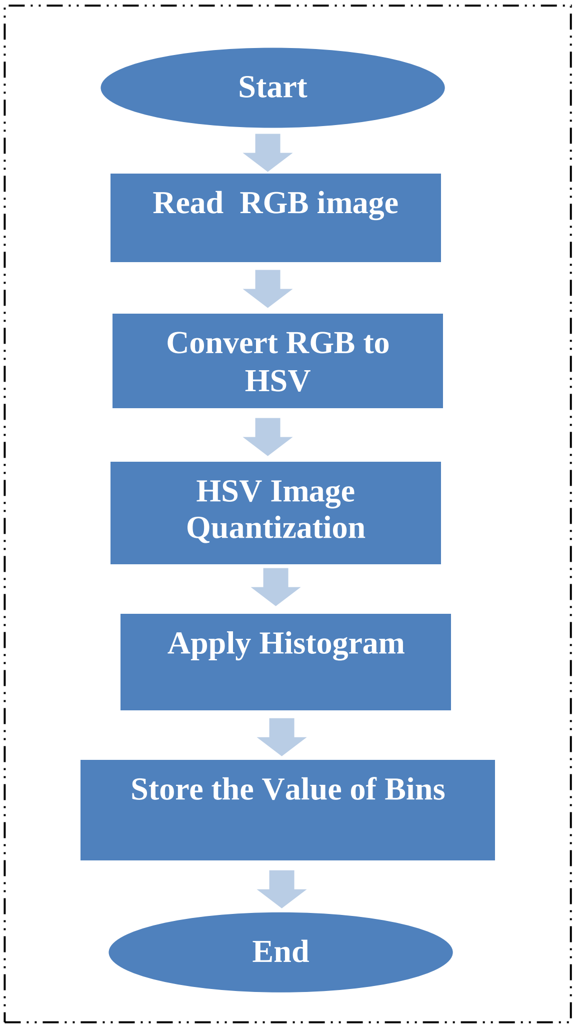

Fig 3.3: Block diagram of the color feature extraction using HSV histogram

3.2.3 DCT Vector Generation

To ease differentiation among several images with similar color, the need to make use of texture is very crucial. For example of these images, sea and sky or grass and leaves. Pixel information is sometimes filtered for gaining better analyze texture properties[33].

For the analysis of a texture image, it requires large storage space and a lot of computational time to calculate the matrix of features. For solving this problem, some researchers proposed using DCT for texture representation. It is one of the most popular techniques used for feature extraction. DCT is shift variant ie, it decomposes the spatial frequency depending on the position of the features in the image. Also it affords high energy compaction[26].

For the DCT transform, a query image is given and converted into a gray level image. The texture feature vector is gained from some DCT coefficients. It is computed directly from the DCT coefficients and the spatial localization using sub blocks. Each image is divided into ***N*N*** sized sub-blocks. The two dimensional DCT can be defined in terms of pixel values ***f***(***i, j***) for ***i,j*** = 0,1,… ***N***-1 and the frequency-domain transform coefficients ***C***(***u***,***v***) as explained in Equation 6[35].

$$c(u,v) = \alpha(u)\alpha(v)\sum_{X=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left(\frac{\pi(2x+1)u}{2N}\right)\cos\left(\frac{\pi(2y+1)v}{2N}\right) \quad (6)$$

Where

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}}, for\ u = 0 \\ \sqrt{\frac{2}{N}}, for\ u \neq 0 \end{cases}$$

Where,

u : indicates regular frequency spatially,

v : indicates perpendicular frequency spatially,

f(x, y) : the pixel value at (x, y),

C(u, v) : DCT coefficient at (u, v).

The algorithm for texture feature vector generation is illustrated in Fig 3.4 and goes through the following steps:

Step 1: Read the image.

Step 2: Convert RGB into gray scale.

Step 3: Partition the image into 8x8 blocks.

Step 4: Apply DCT on each block to acquire DC coefficients.

Step 5: Store the value of DC coefficients in one vector.

Step 6: Repeat step1 to step 5 on all images in the database.

Step 7: Combine the vector of DC coefficients with the vectors of color histograms in one vector.

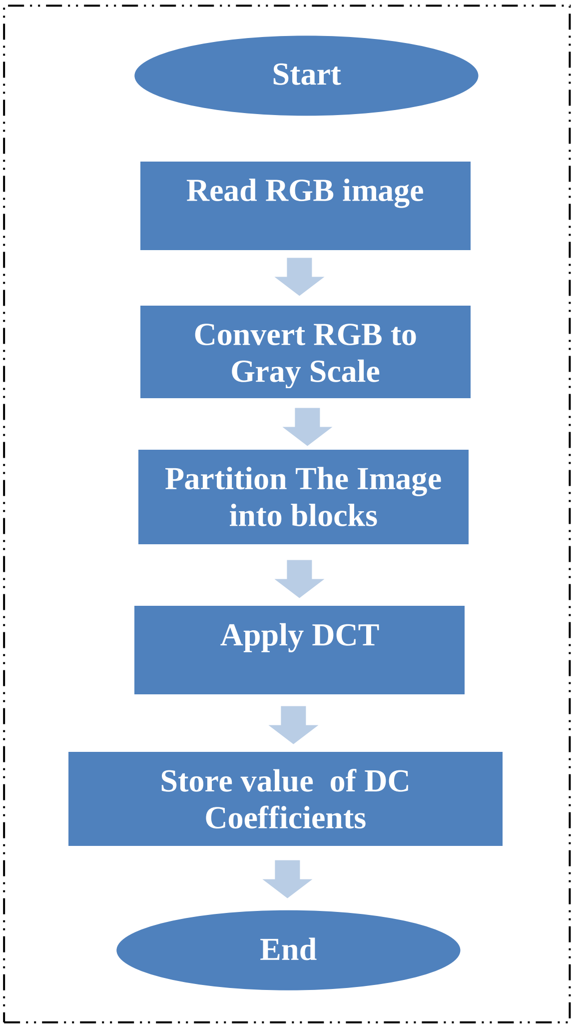

Fig 3.4: Block diagram of the texture feature extraction using DCT

3.3 Feature Similarity Measurement Process

For evaluation purpose, similarity measurement is conducted to compare query image with other images resided in images database. Several distance metrics are in use to distinguish between the query image feature vector and the database image feature vector. To compute the similarity between the input query image and the database images, the difference between the query image feature vector and the database image feature vector is computed by using various distance metrics[13]. For this purpose, the proposed work uses MD, ED and MSE distance metrics for experimentation.

- **Manhattan Distance (MD)**

The Manhattan distance, also known as rectilinear distance, or city block distance. Manhattan Distance between two points is the sum of the absolute differences of their coordinates. The Manhattan Distance is shown in Equation 7.

$$MD_{(x,y)} = \sum_{i=1}^{n} |x_i - y_i| \qquad (7)$$

Where $\boldsymbol{n}$ is the number of variables in each vector, $\boldsymbol{i}$ denotes the range 1....$\boldsymbol{N}$, and $\boldsymbol{x_i}$ and $\boldsymbol{y_i}$ are the values of the $\boldsymbol{i}^{th}$ variable, at points $\boldsymbol{x}$ and $\boldsymbol{y}$ respectively[36].

- **Euclidean Distance (ED)**

Because of its efficiency and effectiveness, Euclidean Distance metric is the most widely used for similarity measurement in image retrieval. It measures the distance between two vectors by computing the square root of the sum of the squared absolute differences, its shown in Equation 8[13].

$$ED_{(x,y)} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \qquad (8)$$

- **Mean Square Error (MSE)**

The mean-squared error (MSE) calculates the average squared difference between the arrays X and Y. Where X and Y can be arrays of any dimension, but must be of the same size and class. It is calculated as in Equation 9.

$$MSE_{(x,y)} = \frac{1}{n} \sum_{i=1}^{n} (x_i - y_i)^2 \qquad (9)$$

Where $\boldsymbol{x}$ and $\boldsymbol{y}$ can be any arrays of any dimension, but must be of the same size $\boldsymbol{n}$.

MSE measures the average of the squares of the errors that is, the average squared difference between the estimated values and what is estimated. MSE is always non-negative, and the smaller value of the MSE represents the better result[37].

In this approach, that matching is done on the distance measurement between the vector of the query image from the histogram and DCT values, and the vector values of the database image. All images are matched by distance measurement. The smaller distance between the vectors, the best match obtained is.

The result is a list of 10 or 20 images based on user interest, and ordered by their similarities with the query image.

If ***I*** is the database image and ***Q*** is the query image, then the algorithm of similarity measure is calculated as shown in Fig 3.5 and illustrated in the following steps:

Step 1: Compute color histogram vector ***vI*** and DCT vector ***dI*** of the database images then combine them into a single vector.

Step 2: Calculate the vectors ***vQ*** and ***dQ*** for the query image also.

Step 3: One measure of distance between two feature vectors will be used to the similarity measurement.

Step 4: From all, the matching images are the top 10 or 20 images which displayed as a result.

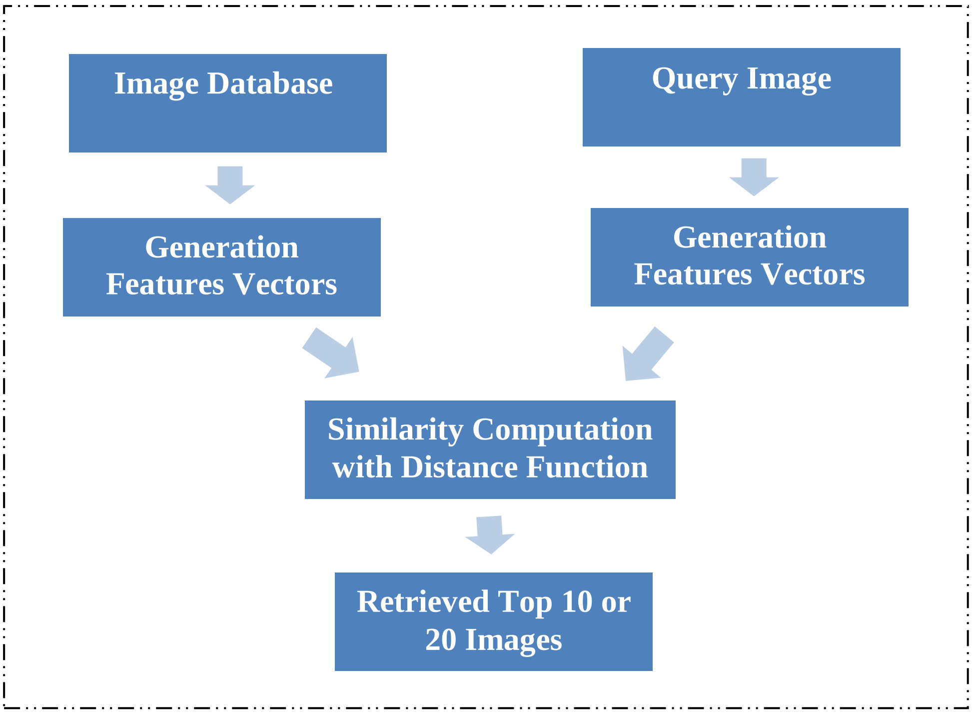

Fig 3.5: Block diagram for similarity measure algorithm

CHAPTER FOUR

IMPLEMENTATION, RESULT AND DISCUSSION

4.1 Introduction

This chapter demonstrates the conducted experiments to test and evaluate the proposed approach. To implement the proposed approach, a prototype system is designed to select a query image, then search in database to retrieve the most similar images of the query image using the necessary tools and programs.

4.2 Software Environment and Dataset

The proposed color image retrieval approach based on feature extraction is implemented using Matlab of version 8.1.0.604 (R2013a) and the hardware architecture used is a workstation with 4GB RAM, intel (R) Pentium (R)2.30 GHz CPU and 32 bit operating system Windows 7 Ultimate.

To test the proposed approach and analyze the retrieval results, a Corel 1-k dataset is used. It contains 1,000 color images divided into 10 different categories: African people, Beach, Building, Buses, Dinosaur, Elephant, Flowers, Horse and Mountain, each category contains 100 photos.

The main reason behind choosing this dataset is that, it is free access, widely used and contains a considerable data. In addition, since the chosen related previous study[31] has evaluated its study using this dataset, it make sense to choose the same dataset.

All images are in RGB color space. Image size was standardized to 384 x 256 pixels. Fig 4.1 shows some images from a Corel l-k dataset.

Fig: 4.1 Sample images for Corel-1k dataset

4.3 Overall Scheme of Implemented Approach

The proposed CBIR approach is divided into two main phases. Off - line process phase and on-line process phase.

4.3.1 Off - line Process Phase

This phase consists of two Stages, which are:

Stage 1: CBIR algorithm for loading dataset images.

This stage contains two steps:

Step 1: Load Corel l-k dataset which contains 1000 color images.

Step 2: At pre-processing stage, image resized into 384 x 256 pixels and converted into HSV color space to extract color feature, and converted into gray level to extract texture feature.

Stage 2: CBIR algorithm for generating features vectors.

This stage contains three steps:

Step 1: Color feature is extracted by applying HSV histogram and the values are stored in a single feature vector.

Step 2: Texture feature is extracted by applying DCT transform for grayscale images and values of DC coefficients are saved as a single feature vector.

Step 3: Combining two feature vectors (color and texture information) in one vector and keep it in database.

4.3.2 On - line Process Phase

This phase consists of two Stages, which are:

Stage 1: CBIR algorithm for generating features vectors.

Query image is selected and the stages one and two in off - line process phase are repeated to build a query feature vector based on color and texture feature.

Stage 2: CBIR algorithm for similarity measurement.

This stage contains two steps:

Step 1: Distance between the query image and database image is calculated by one of similarity matching distance measures.

Step 2: Top 20 images that are similar to the query image based on the small value of distance are returned.

The proposed approach architecture is explained in Fig 4.2.

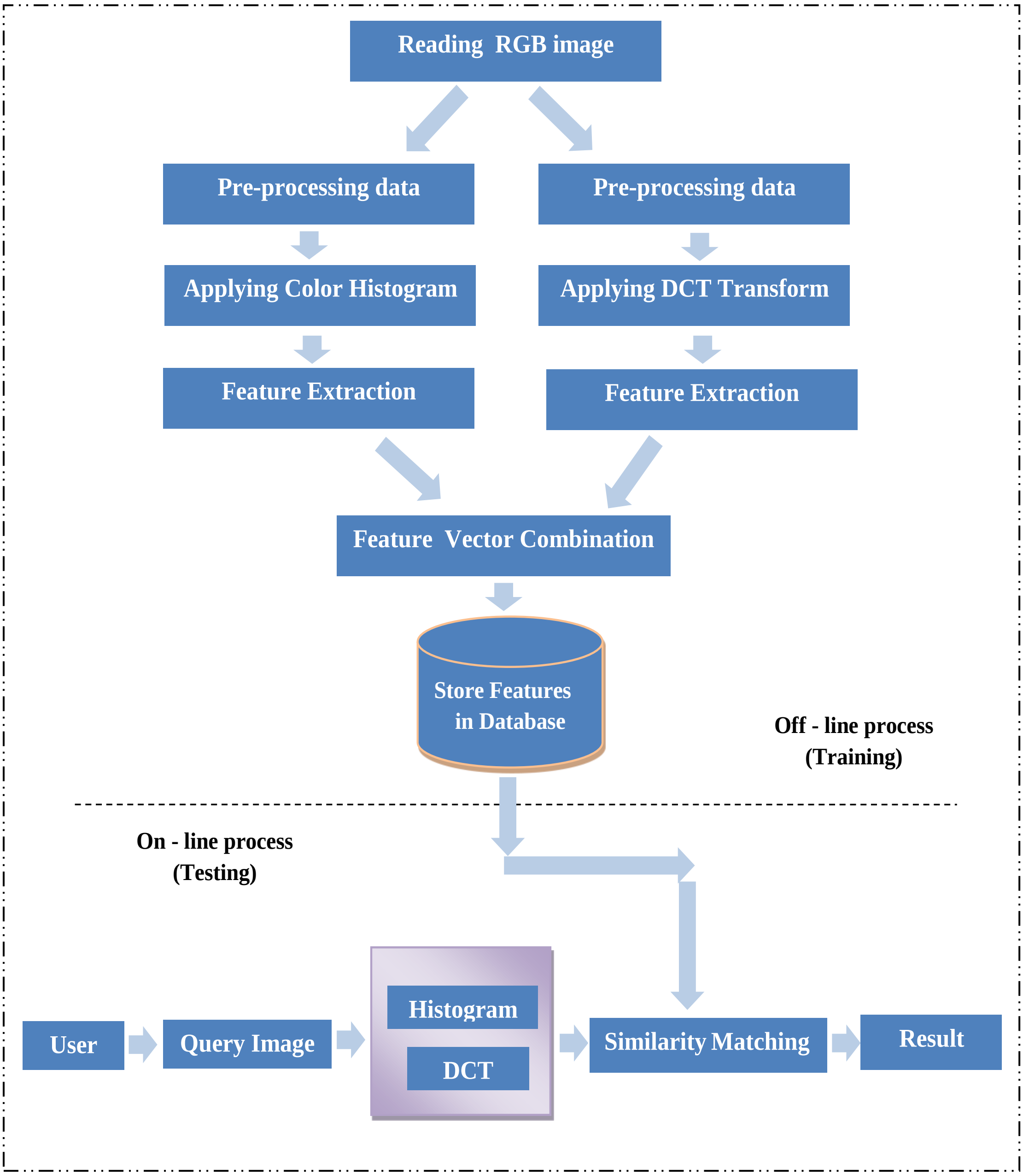

Fig: 4.2 Architecture of The Proposed Approach

4.4 Results and Discussion

The performance efficiency of the prototype system and execution time are tested using some similarity measures.

4.4.1 Evaluation Measures for CBIR System

There are several ways to evaluate the performance of CBIR systems and measure their efficiency, the most famous are precision and recall.

The precision is defined as the ratio of the number of related images retrieved to the total number of images retrieved, while the recall is the ratio of the number related images retrieved to total the relevant images in the database.

Precision can be seen as a measure of exactness or quality. On the other hand, recall is a measure of completeness or quantity. Simply, high precision denotes that an algorithm returned substantially more relevant results than irrelevant ones, while high recall means that an algorithm returned most of the relevant results. Precision and recall are defined in Equations 10 and 11.

$$\text{Precision} = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images retrieved}} = \frac{\boldsymbol{A}}{\boldsymbol{B}} \qquad (10)$$

$$\text{Recall} = \frac{\text{Number of relevant images retrieved}}{\text{Total number of relevant images in the dataset}} = \frac{\boldsymbol{A}}{\boldsymbol{C}} \qquad (11)$$

Where ***A*** is the set of retrieved images matching the query, ***B*** is the set of returned images, and ***C*** is the set of images matching the query in the database[1].

4.4.2 The proposed Approach Evaluation

In this section the proposed prototype system is tested the results are shown and discussed. First, the proposed system is evaluated using several distance measures, then it is compared with previous studies.

To find the similar images, the feature of query image is compared with feature of images database by Manhattan, Euclidean and Mean Squared Error methods which calculate the minimum distance. The prototype system retrieves the top 10 or 20 images similar to the query image depending on the user's interest. The retrieved results are a list of images arranged by their similarity distances to the query image. For each category, four images are selected randomly, and calculating the average for them.

In order to test and evaluate the proposed prototype system, a graphical user interface described in Fig 4.3 is designed to allow the selection of the required similarity measurement unit as well as the number of images retrieved in the retrieval process.

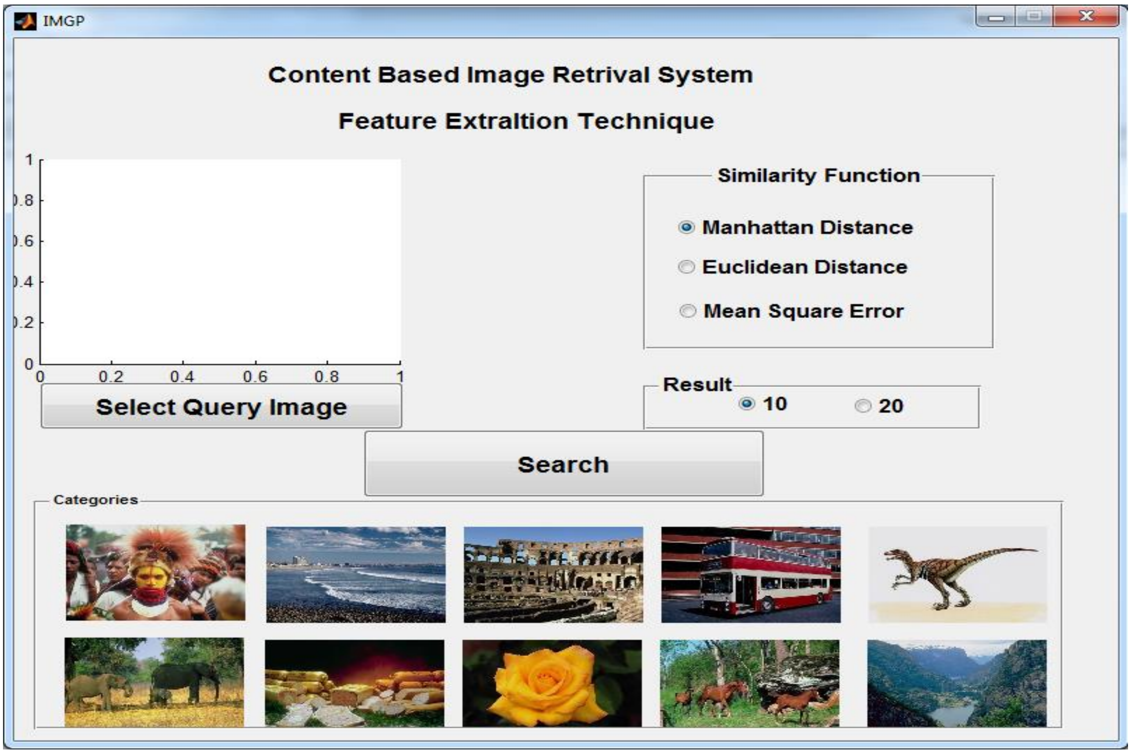

Fig: 4.3 Main user interface for CBIR prototype system

The first test is selecting the required similarity measurement method, then selecting a random image from African category, for example. The query image is loaded in the framework. When clicking on search button, the system retrieves the best 10 or 20 images similar to image query by selecting the number of images results, which show in Fig 4.4. The result shows the system retrieved all images similar to the query image. Similarly, when choosing 20 images as a result, the system proved effective retrieval all images similar to the image of the query as shown in Fig 4.5, with different retrieval time.

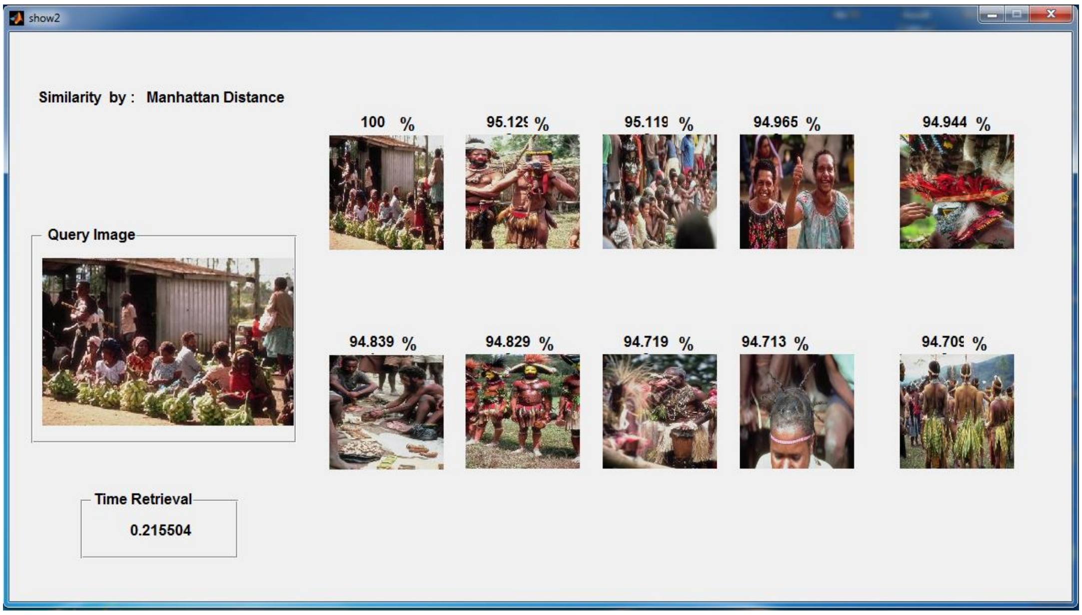

Fig: 4.4 Top-10 retrieval result for African image

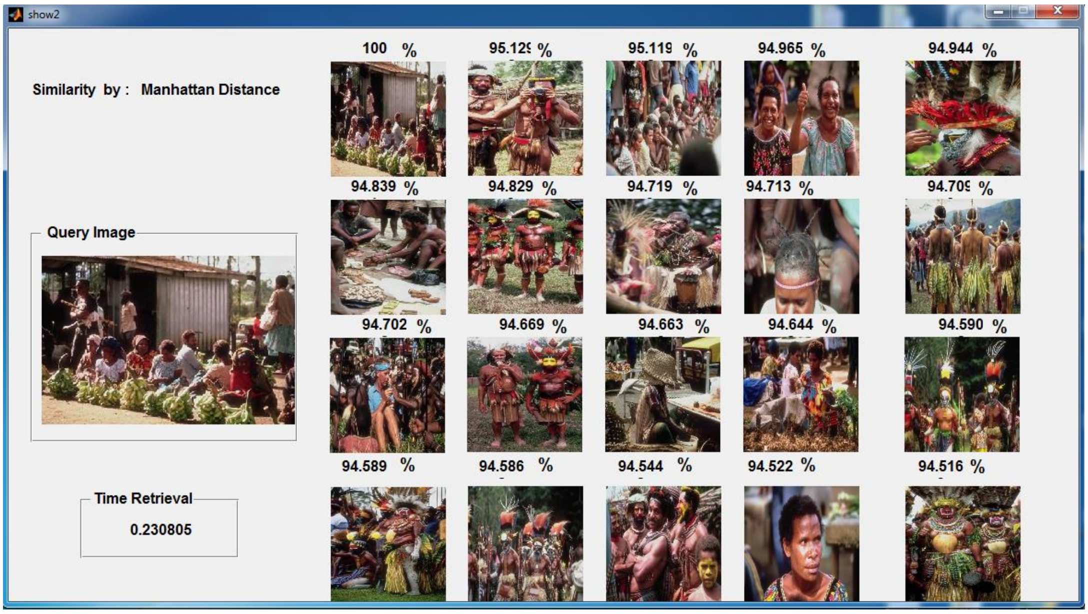

Fig: 4.5 Top-20 retrieval result for African image

In Fig 4.6, a second query image and its results appear. This is the second test for the system. A random image was selected from Bus category and 10 images were retrieved as a result. All retrieved images belong to the same category as the query image. Fig 4.7 shows top 20 retrieved Images for the same query image. The test shows that, the system's efficiency in retrieving images is almost similar to the query image when retrieving ten images. The results are good, even with different color of the Bus, because the system does not rely on the color feature only, thus the result of retrieval is improved by integrating the texture feature. Most Buses show the same size and shape as the bus query image.

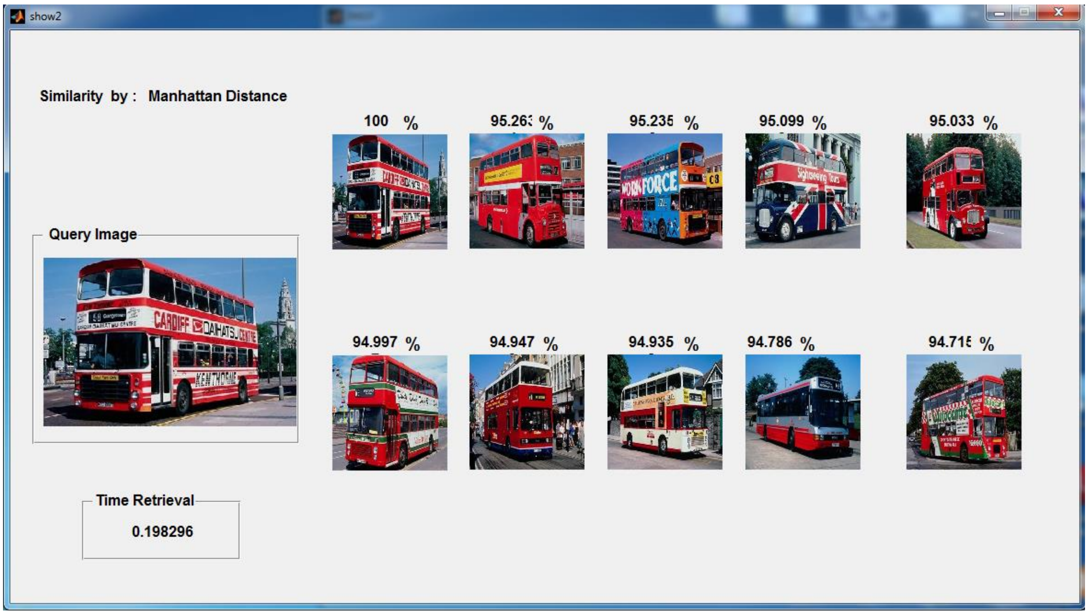

Fig: 4.6 Top-10 retrieval result for Bus image

Fig:4.7 Top-20 retrieval result for Bus image

The third test of the prototype system is based on Dinosaur category. All retrieved images are similar to query image, whether 10 or 20 images result, as shown in Fig 4.8 and Fig 4.9. It is noted that Dinosaur category in the most retrieval systems are with good results, due to the nature of the image, where Dinosaur object is clear in image, which facilitates the process of finding the histogram accurately, as well as when converting the image into image in HSV color space. In proposed prototype system, the retrieved results in Dinosaur category are very high with all used similarity measures.

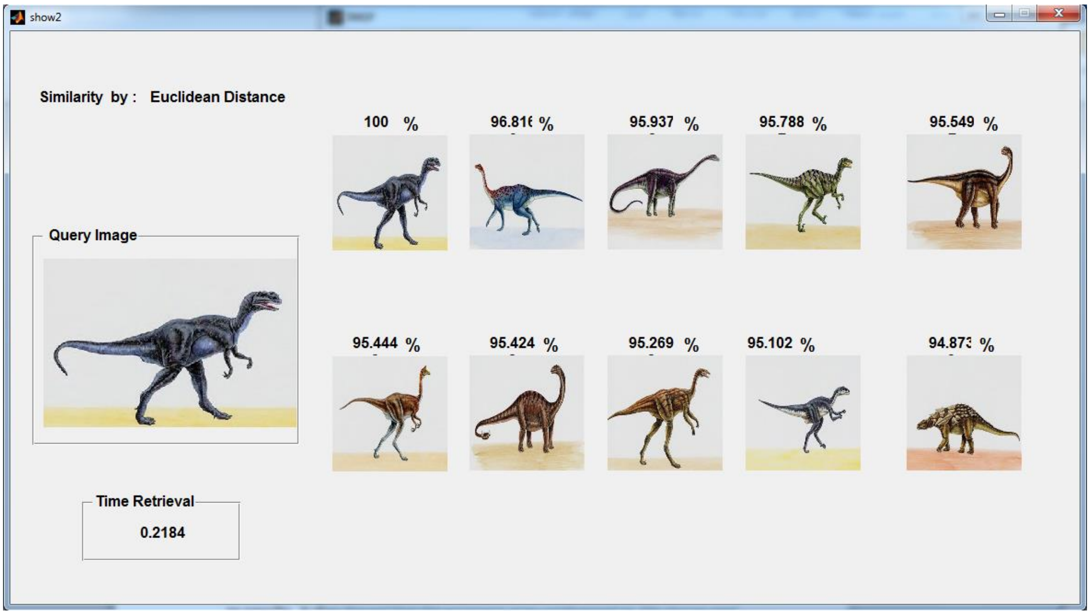

Fig: 4.8 Top-10 retrieval result for Dinosaur image

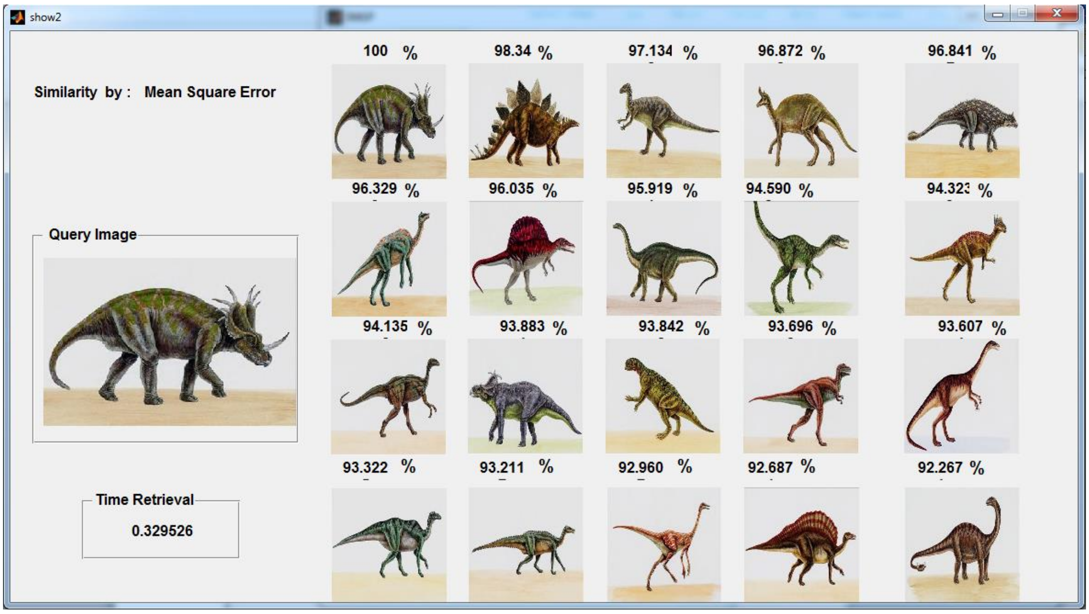

Fig: 4.9 Top-20 retrieval result for Dinosaur image

4.4.2.1 (MD) Similarity Measure

When applying MD similarity measure, and comparing the precision and recall values between results of 10 and 20 images, it was found that the accuracy was quite equal in Horse and Dinosaur categories, and was slightly lower in African, Bus and Food categories, and slightly increased in the other categories. The recall values are increased when retrieving 20 images comparing with 10 images for all categories. These values are given in Table 4.1.

Table 4.1: Precision and recall of all image categories using MD similarity measure

	**Precision**		**Recall**	
Category	**10 images**	**20 images**	**10 images**	**20 images**
African	**0.925**	**0.875**	**0.093**	**0.175**
Beach	**0.775**	**0.650**	**0.078**	**0.130**
Building	**0.900**	**0.763**	**0.090**	**0.153**
Bus	**0.875**	**0.863**	**0.088**	**0.173**
Dinosaur	**1**	**1**	**0.1**	**0.2**
Elephant	**0.700**	**0.600**	**0.070**	**0.120**
Flower	**0.975**	**0.863**	**0.098**	**0.173**
Horse	**1**	**1**	**0.1**	**0.2**
Mountain	**0.725**	**0.638**	**0.073**	**0.128**
Food	**0.975**	**0.925**	**0.098**	**0.185**
Average	**0.885**	**0.818**	**0.084**	**0.164**

Example of results when applying the proposed framework are shown in the linear relationship between precision of two results retrieving, also the recall in Fig 4.10 and Fig 4.11.

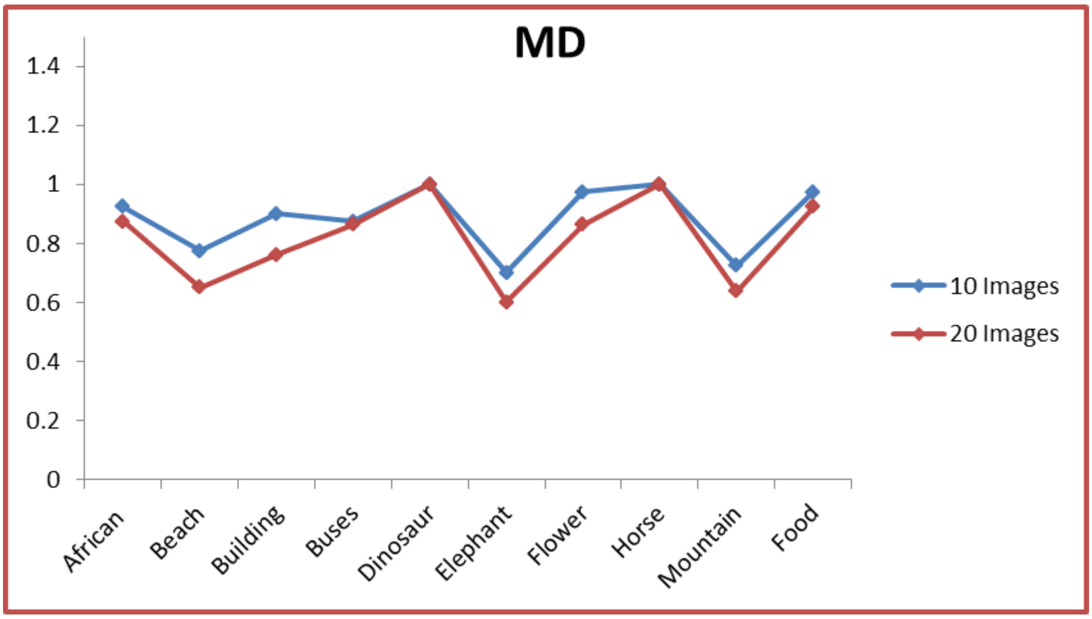

Fig 4.10: The precision by MD

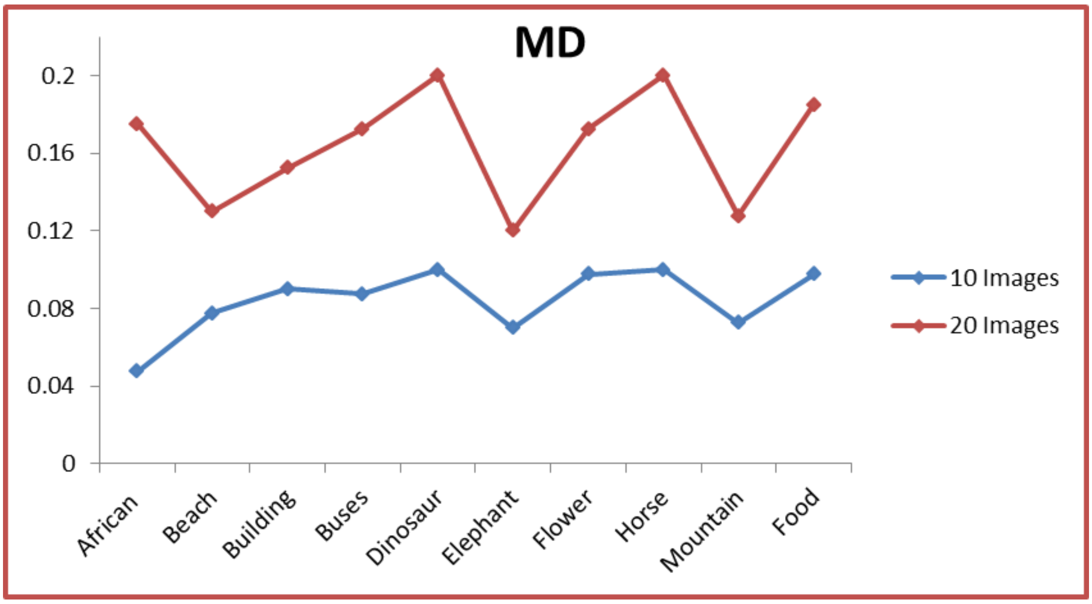

Fig 4.11: The recall by MD

4.4.2.2 (ED) Similarity Measure

A comparison of the precision and recall values when the ED Similarity Measure is used shows that, the Dinosaur category has the same of precision in two the results, and African, Horse and Bus categories were slightly better when retrieving 10 images, while the other categories varied in accuracy. Table 4.2 shows the values and average for all image categories.

Table 4.2 : Precision and recall of all image categories using ED similarity measure

	Precision		Recall	
Category	10 images	20 images	10 images	20 images
African	1	0.900	0.1	0.180
Beach	0.575	0.413	0.058	0.083
Building	0.850	0.650	0.085	0.130
Bus	0.920	0.888	0.095	0.178
Dinosaur	1	1	0.1	0.2
Elephant	0.650	0.475	0.065	0.095
Flower	0.925	0.775	0.093	0.155
Horse	1	0.963	0.1	0.193
Mountain	0.700	0.625	0.070	0.125
Food	0.975	0.850	0.098	0.170
Average	0.8595	0.7540	0.0863	0.1508

The linear relationship between two the results which are retrieved by ED similarity measure are shown in Fig 4.12 and Fig 4.13 respectively.

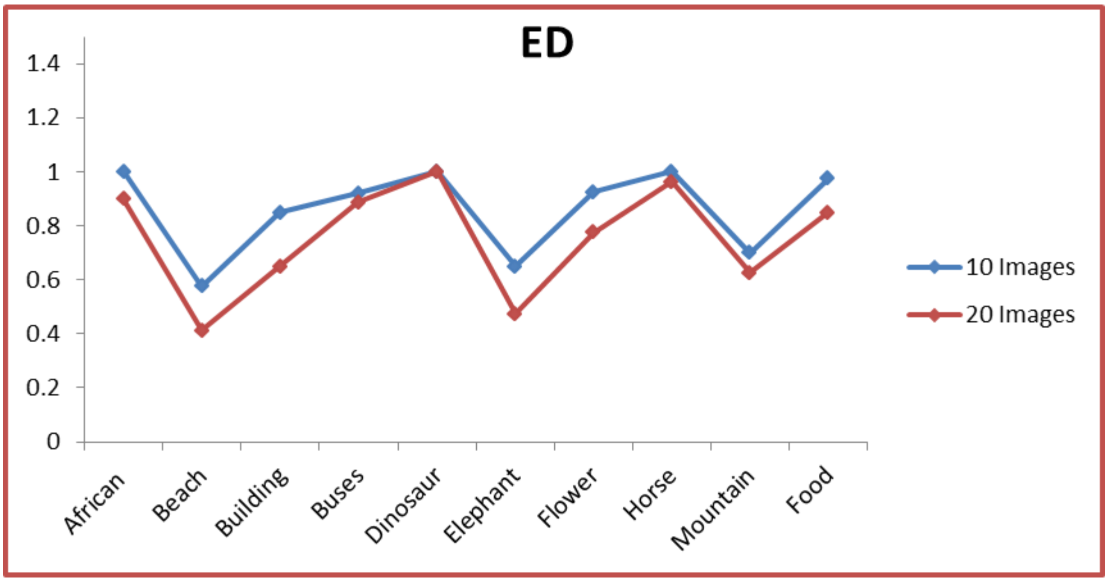

Fig 4.12: The precision by ED

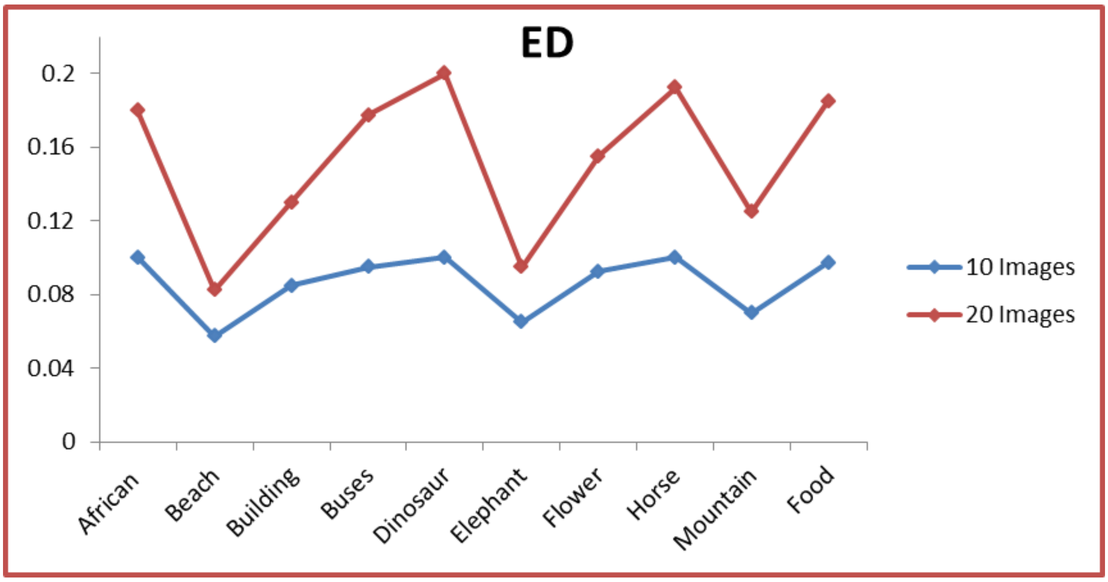

Fig 4.13: The recall by ED

4.4.2.3 (MSE) Similarity Measure

The difference between precision of results retrieved by applying MSE similarity measure are shown in Table 4.3, where Dinosaur category is the same. It can also be observed that it low in African, Horse and Bus categories. The precision in rest of categories were varying accuracy. The recall values are increased when retrieving 20 images for all categories.

Table 4.3 : Precision and recall of all image categories using MSE similarity measure

	Precision		Recall	
Category	**10 images**	**20 images**	**10 images**	**20 images**
African	**1**	**0.900**	**0.1**	**0.180**
Beach	**0.525**	**0.363**	**0.053**	**0.072**
Building	**0.850**	**0.65**	**0.085**	**0.130**
Bus	**0.920**	**0.888**	**0.095**	**0.178**
Dinosaur	**1**	**1**	**0.1**	**0.2**
Elephant	**0.575**	**0.425**	**0.058**	**0.085**
Flower	**0.925**	**0.775**	**0.093**	**0.155**
Horse	**0.975**	**0.938**	**0.098**	**0.188**
Mountain	**0.700**	**0.625**	**0.0700**	**0.125**
Food	**0.975**	**0.775**	**0.098**	**0.078**
Average	**0.8445**	**0.7338**	**0.0750**	**0.1390**

The retrieval results are graphically represented in the two following Fig 4.14 and Fig 4.15.

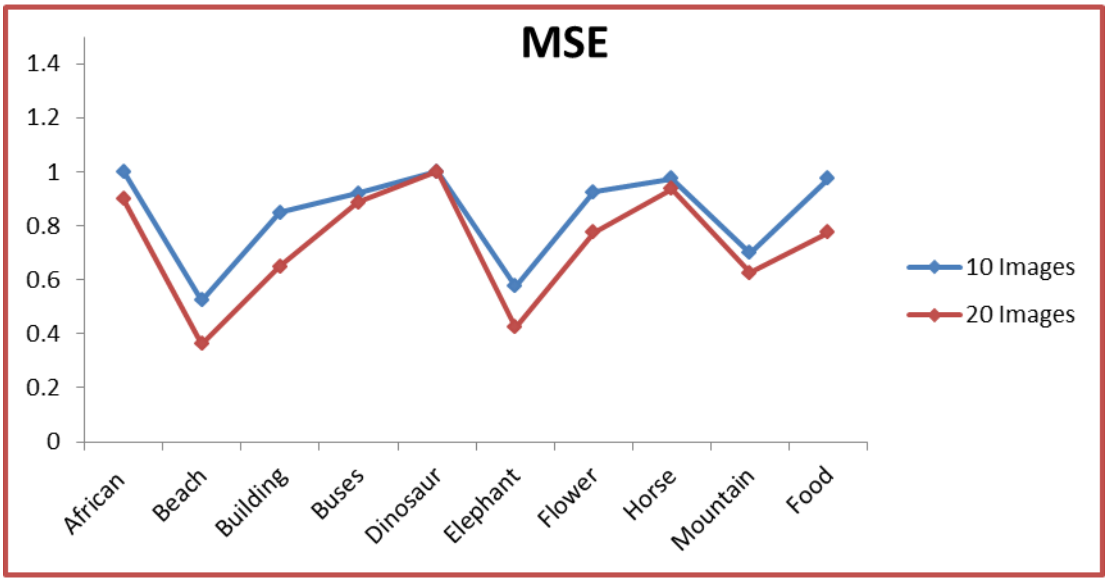

Fig 4.14: The precision by MSE

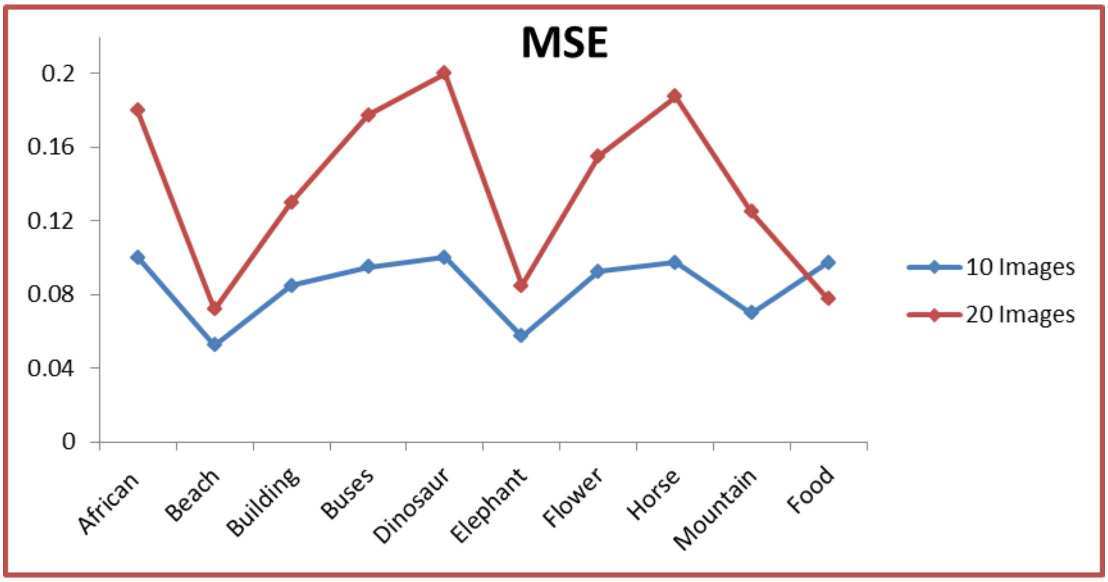

Fig 4.15: The recall by MSE

4.4.2.4 Overall Average of System Evaluation

The general average for precision and recall was calculated to retrieve the image using the three methods for measuring similarity. In terms of precision, MD similarity measure was the best, followed by the ED similarity measure and finally the MSE similarity measure with little difference. The average values are shown in Table 4.4 The recall values also shown in this table.

Table 4.4 : Average precision and recall of all image categories using three similarity measures

	**Precision**		**Recall**	
similarity measure	**10 images**	**20 images**	**10 images**	**20 images**
MD	**0.8850**	**0.8180**	**0.0840**	**0.1640**
ED	**0.8595**	**0.7540**	**0.0863**	**0.1508**
MSE	**0.8445**	**0.7338**	**0.0750**	**0.1390**

The precision and recall values of the retrieved results by three similarity measures are represented graphically in Fig 4.16, Fig 4.17,Fig 4.18 and Fig 4.19.

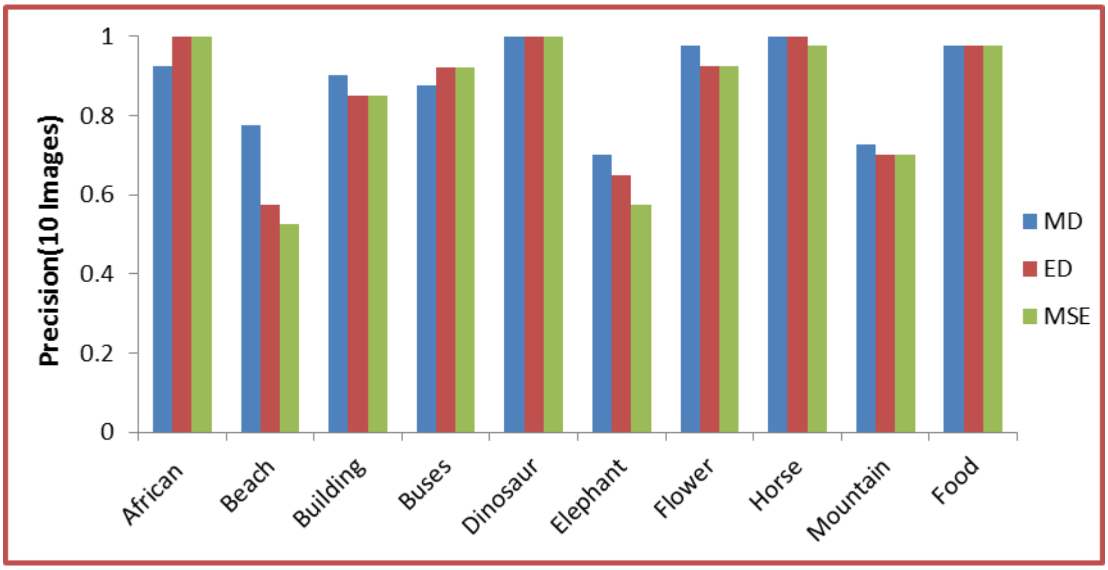

Fig 4.16: Comparison of Precision of 10 images

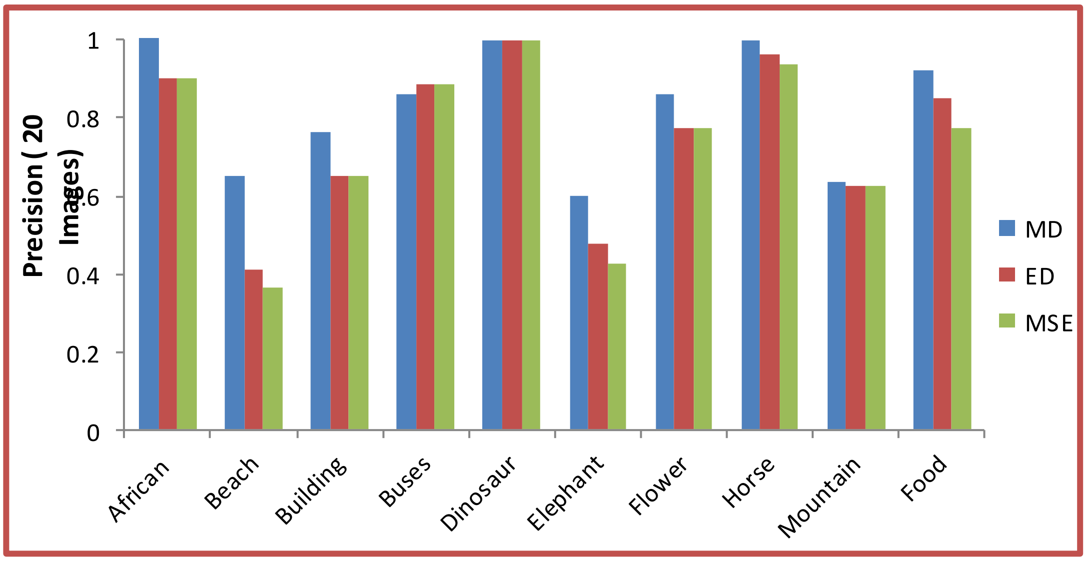

Fig 4.17: Comparison of Precision of 20 images

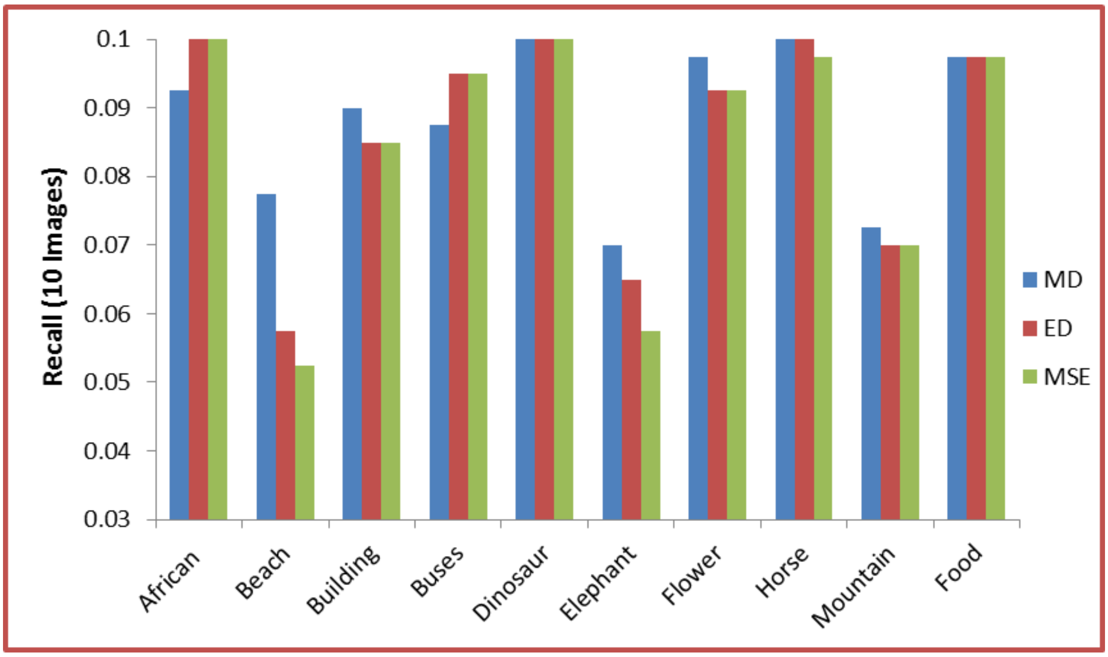

Fig 4.18: Comparison of recall of 10 images

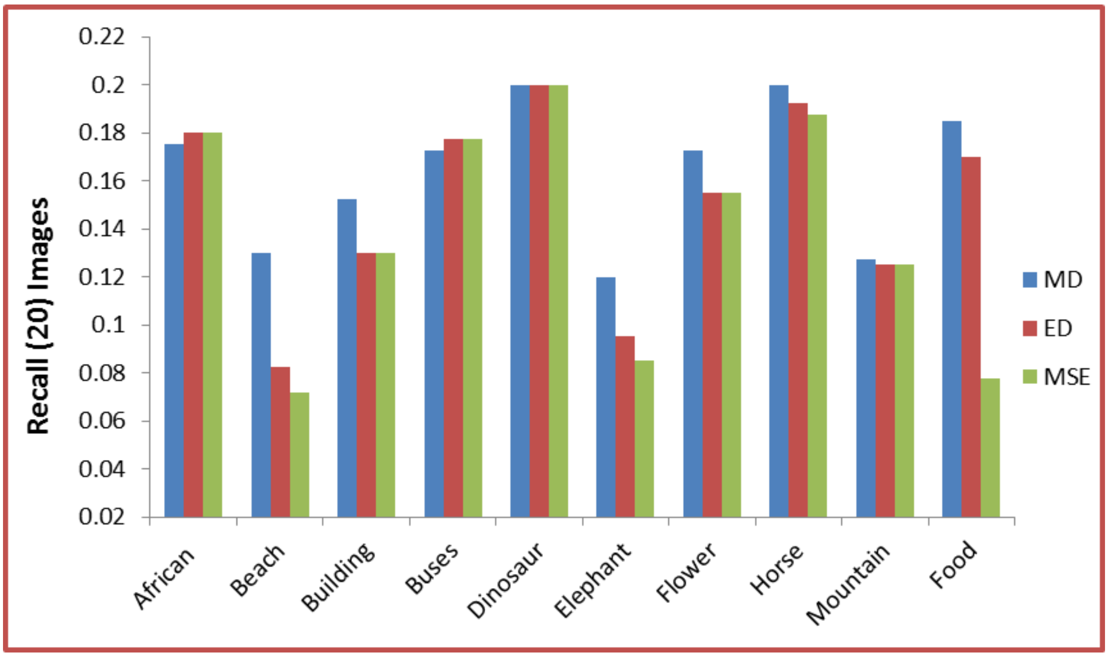

Fig 4.19: Comparison of recall of 20 images

4.4.3 Execution Time

Time retrieval in this prototype system was measured by tic and toc command in Matlab. The three similarity measures in the retrieval process are compared and it is observed that MD similarity measure is the best. Time retrieval of all image categories and average using three similarity measures are shown in Table 4.5.

Table 4.5 : Time retrieval of all image categories using three similarity measures

	MD		ED		MSE	
Category	**10 images**	**20 images**	**10 images**	**20 images**	**10 images**	**20 images**
African	**0.185821**	**0.185191**	**0.232563**	**0.215342**	**0.253046**	**0.223822**
Beach	**0.189025**	**0.193698**	**0.210779**	**0.218085**	**0.220224**	**0.226262**
Building	**0.190903**	**0.191038**	**0.214922**	**0.217183**	**0.235264**	**0.230030**
Bus	**0.189475**	**0.185002**	**0.217031**	**0.224226**	**0.221686**	**0.231445**
Dinosaur	**0.199029**	**0.193279**	**0.220731**	**0.240948**	**0.219991**	**0.224713**
Elephant	**0.199727**	**0.169523**	**0.225568**	**0.215444**	**0.335714**	**0.225218**
Flower	**0.202410**	**0.199885**	**0.234206**	**0.222322**	**0.225289**	**0.228048**
Horse	**0.195588**	**0.196394**	**0.228345**	**0.212715**	**0.226906**	**0.232744**
Mountain	**0.188544**	**0.196420**	**0.221880**	**0.226041**	**0.243296**	**0.221207**
Food	**0.183688**	**0.186412**	**0.215628**	**0.217739**	**0.230341**	**0.226685**
Average	**0.172518**	**0.189684**	**0.222165**	**0.221005**	**0.241186**	**0.227017**

The comparison of run time by three similarity measures are represented graphically in Fig 4.20 and Fig 4.21.

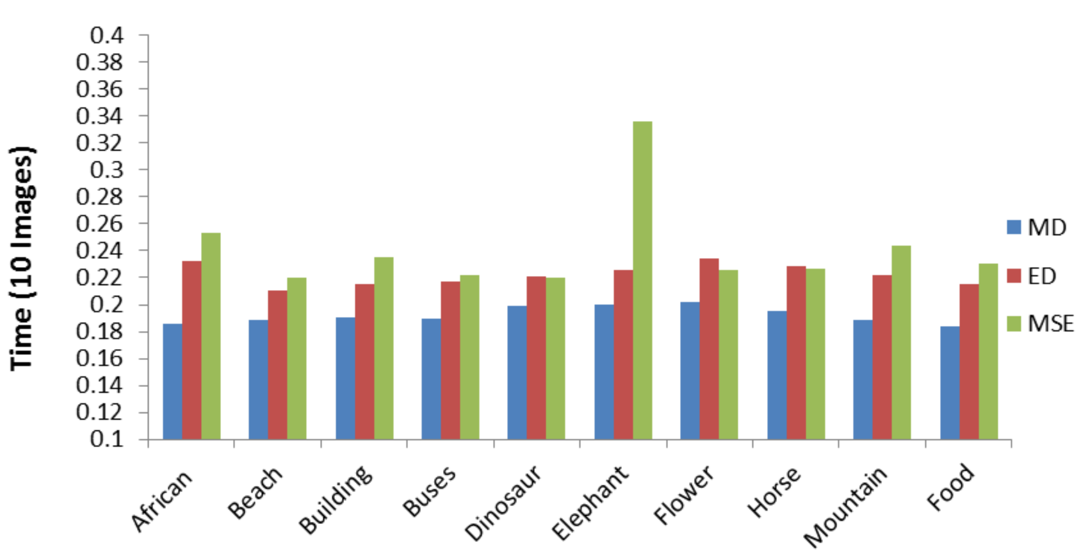

Fig 4.20: Comparison of run time of 10 images

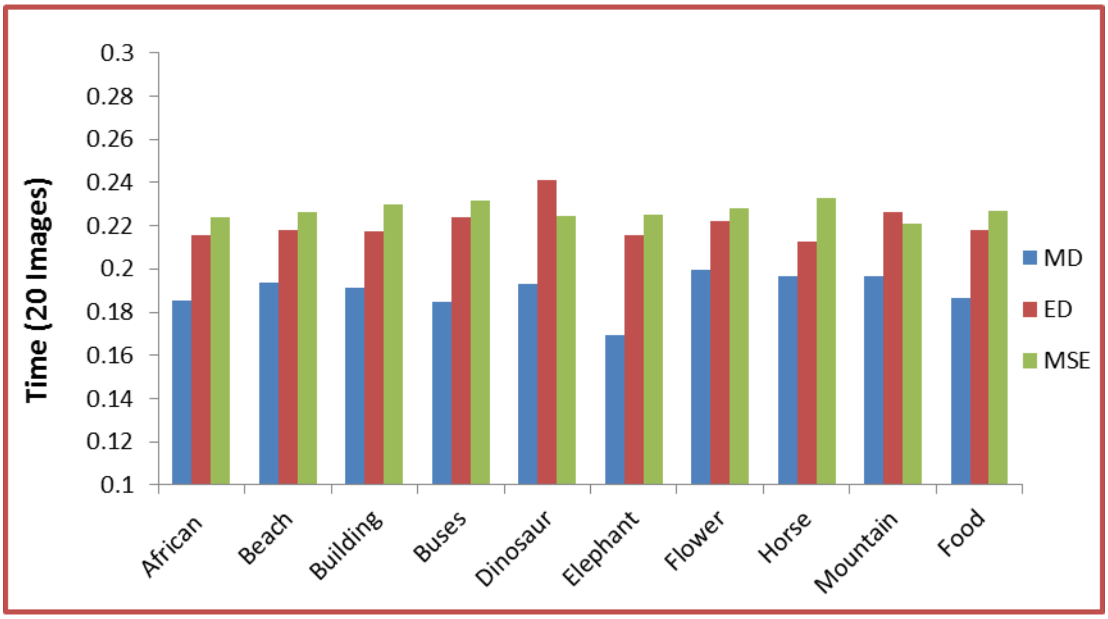

Fig 4.21: Comparison of run time of 20 images

4.4.4 Comparison of the proposed Approach with previous studies

The results of the proposed approach are compared with other previous studies which are selected for performance comparison as reported by[31]. A summary of previous works and proposed approach are shown in Table 4.6. For comparison purpose with some previous studies whose results were the same number of images, MD similarity measure is chosen for comparison because it produced better results than others in the proposed approach. The comparison results are shown in Table 4.7, which clarify that the accuracy of the performance of the proposed approach was better than other previous studies in most categories, except for Bus category, while the accuracy was equal in Dinosaur category. The accuracy was improved by (8.3%) compared with the best results of the previous studies, where the rate of accuracy of the retrieval in those the study was (73.5%), while the value in the proposed system was (81.8%).

Table 4.6 : A summary of previous works and proposed approach

Author (Year)	Techniques	Similarity metric(s)	Performance measure (s)	A sample research	Results
M.E. Elalami (2011)	3D color histogram and Gabor filter. A genetic algorithm (GA).	Euclidean Distance	precision and recall	Wang database	Effectiveness and efficiency of the proposed model and a precise image retrieval in a short time.
J.Yue etl (2011)	Color histogram and a co-occurrence matrix	Euclidean Distance	precision and recall	NA	The fused features retrieval brings better visual feeling than the single feature retrieval.
J. Yu etl (2013)	Integrating the SIFT,HOG with the LBP descriptor. K-means clustering algorithm	Histogram intersection	Average retrieval precision (ARP)	Corel database	The feature extraction module takes a longer time, but the integration and similarity measure modules are performed separately and run fast.
S.Somnug etl (2016)	Color correlograms and Edge Direction Histogram (EDH)	Euclidean Distance	precision and recall	Wang database	Combining feature yields a good result when compares with the other combining scheme.
A. Nazir etl (2018)	CH, DWT and EDH	Manhattan Distance	precision and recall	Corel l-k dataset	considerable performance with existing CBIR systems
Proposed approach	Color histogram and Discrete Cosine Transform	Manhattan Distance, Euclidean Distance and Mean Square Error	precision and recall	Corel l-k dataset	The precision was increased by (8.3%) rate compared to best result of previous studies.

Table 4.7 : Average precision of all image categories with other previous studies

Category	M.E. Elalami[32]	J.Yue etl [4]	J. Yu etl [7]	S.Somnug etl [18]	A. Nazir etl [31]	Result of Approach
Africa	0.58	0.53	0.57	0.676	0.85	0.875
Beach	0.41	0.45	0.58	0.598	0.50	0.650
Building	0.42	0.46	0.43	0.58	0.75	0.763
Bus	0.71	0.84	0.93	0.94	1	0.863
Dinosaur	0.74	0.90	0.98	0.998	1	1
Elephant	0.65	0.72	0.666	0.58	0.55	0.600
Flower	0.83	0.74	0.83	0.886	0.95	0.863
Horse	0.69	0.72	0.68	0.938	0.90	1
Mountain	0.44	0.53	0.46	0.478	0.30	0.638
Food	0.44	0.46	0.53	0.492	0.55	0.925
Average	0.595	0.641	0.650	0.725	0.735	0.818

Table 4.8 : Average recall of all image categories with other previous studies

Category	M.E. Elalami[32]	J.Yue etl [4]	J. Yu etl [7]	S.Somnug etl [18]	A. Nazir etl [31]	Result Approach
Africa	0.12	0.11	0.11	0.13	0.17	0.175
Beach	0.08	0.09	0.12	0.12	0.10	0.130
Building	0.08	0.09	0.08	0.12	0.15	0.153
Bus	0.14	0.17	0.19	0.19	0.20	0.173
Dinosaur	0.15	0.18	0.19	0.19	0.20	0.2
Elephant	0.13	0.15	0.12	0.13	0.11	0.120
Flower	0.17	0.15	0.16	0.18	0.19	0.173
Horse	0.14	0.14	0.13	0.19	0.18	0.2
Mountain	0.09	0.11	0.09	0.09	0.06	0.128
Food	0.09	0.09	0.10	0.10	0.11	0.185
Average	0.119	0.128	0.129	0.144	0.147	0.164

The graphical representation of precision and recall for the proposed approach with the previous systems is shown in Fig 4.22 and Fig 4.23.

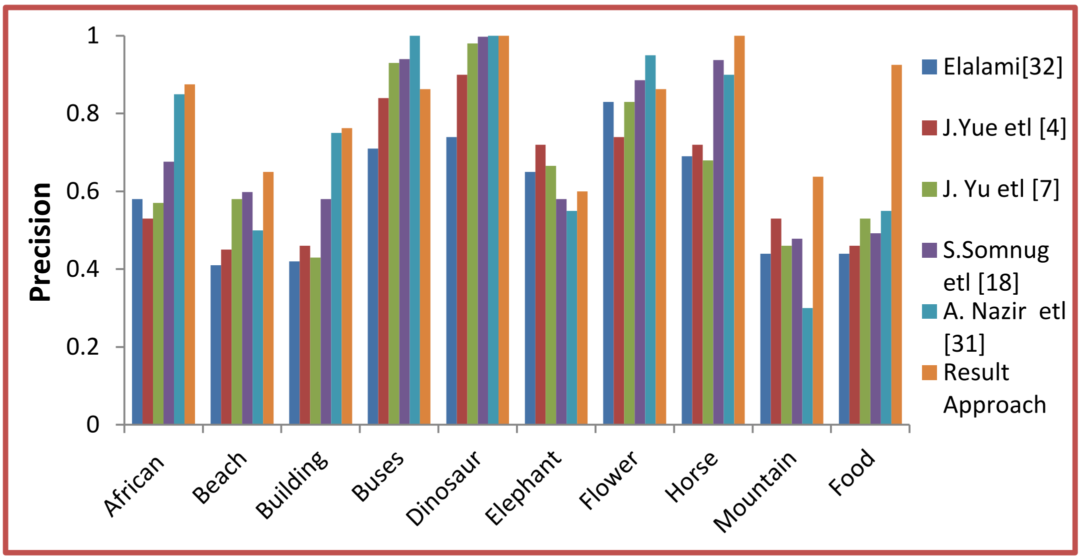

Fig 4.22: Comparison of precision measure of the proposed approach with previous studies

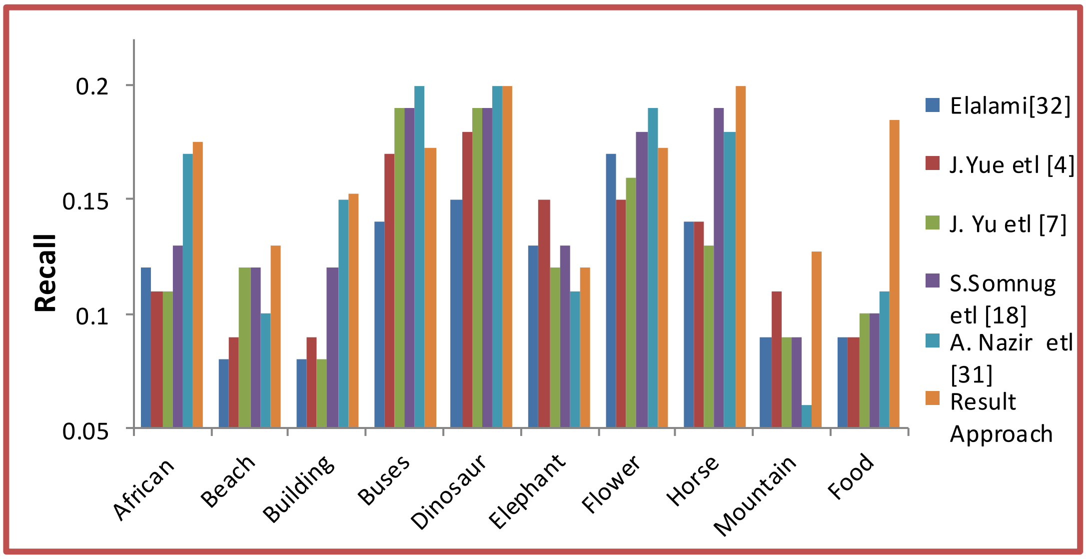

Fig 4.23: Comparison of recall measure of the proposed approach with previous studies

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

This chapter, presents a summary of CBIR proposed system, casts light on the contribution it has achieved and a conclusion for the work. Also presents some suggestions, recommendation for future works.

5.1 Conclusions

This dissertation suggests a color images retrieval approach based on the image content. Color and texture features are extracted to represent the image content, and various distance metrics for the performance analysis are used.

In order to extract color and texture features, the color histogram and DCT techniques are used respectively, then integrating these features into a single vector representing the image in numerical values to compare it with the vector of the query image using MD, ED and MSE functions to measure the similarity.

To evaluate the performance of the proposed approach, the precision and recall values are calculated.

Performance of the proposed approach was evaluated using three similarity measures. MD similarity measure proved its effectiveness in the retrieval process, as well as in the response time compared to other similarity measures.

The proposed approach was evaluated and compared with some previous studies. It has proved its effectiveness in retrieval process, and it has good performance in precision with rate (8.3%) compared with the best result of other studies. This means that using the proposed approach has improved the process of retrieving color images in means of accuracy and response time.

5.2 Future Work

Although the proposed approach proved its effectiveness in retrieving color images based on their content, however, there is a lot of work that can be done to increase the efficiency of this approach.

1. A CBIR proposed approach works on the low-level visual features which are color and texture. Combining low-level and high-level features may give good results in retrieving color images.

2. To further improvement, the researcher recommend using fuzzy logic technique and Artificial Neural Networks (ANN) such as Convolutional Neural Networks (CNN) to classify color images based on their content.

References

[1] A. J. Afifi and W. M. Ashour, "Image retrieval based on content using color feature," *International Scholarly Research Notices*, vol. 2012, 2012.

[2] H. H. Wang, D. Mohamad, and N. A. Ismail, "Approaches, challenges and future direction of image retrieval," *arXiv preprint arXiv:1006.4568,* 2010.

[3] S. Lata and P. P. Singh, "A Review on Content Based Image Retrieval System," *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), ISSN,* vol. 2277, 2014.

[4] R. S. Patil and A. J. Agrawal, "Content-based image retrieval systems: a survey," *Advances in Computational Sciences and Technology,* vol. 10, pp. 2773-2788, 2017.

[5] W. Zhou, H. Li, and Q. Tian, "Recent advance in content-based image retrieval: A literature survey," *arXiv preprint arXiv:1706.06064,* 2017.

[6] M. Singh, "Under the section of Computer and Information Technology Sciences, Image Retrieval- an Overview," *Research Journal of Recent Sciences,* vol. 4(ISC-2014), 2015.

[7] A. M. Riad, H. K. Elminir, and S. Abd-Elghany, "A literature review of image retrieval based on semantic concept," *International Journal of Computer Applications,* vol. 40, pp. 12-19, 2012.

[8] M. Alkhawlani, M. Elmogy, and H. El Bakry, "Text-based, content-based, and semantic-based image retrievals: A survey," *Int. J. Comput. Inf. Technol,* vol. 4, 2015.

[9] K. Akhilesh, R. R. Shedamkar, and S. Sharma, "Comparison and Analysis of Different Image Retrieval Systems," *International Journal of Recent Trends in Engineering & Research (IJRTER),* vol. 02, 2016.

[10] M. Gupta and A. K. Garg, "Analysis of image compression algorithm using DCT," *International Journal of Engineering Research and Applications (IJERA),* vol. 2, pp. 515-521, 2012.

[11] D. T. i Hasta, "Fast Discrete Cosine Transform Algorithm Analysis on IJG JPEG Compression Software," Faculty of Industrial Technology, Gunadarma University, Gunadarma University, 2012.

[12] F. Malik and B. Baharudin, "Analysis of distance metrics in content-based image retrieval using statistical quantized histogram texture features in the DCT domain," *Journal of king saud university-computer and information sciences,* vol. 25, pp. 207-218, 2013.

[13] Y. Mistry, D. Ingole, and M. Ingole, "Content based image retrieval using hybrid features and various distance metric," *Journal of Electrical Systems and Information Technology,* 2017.

[14] D. He, "Three new methods for color and texture based image matching in Content-Based Image Retrieval," 2010.

[15] A. S. GOMASHE and R. KEOLE, "A Novel Approach of Color Histogram Based Image Search/Retrieval," *International Journal of Computer Science and Mobile Computing,* pp. 57-65, 2015.

[16] A. Khokher and R. Talwar, "Content-based image retrieval: Feature extraction techniques and applications," in *International Conference on Recent Advances and Future Trends in Information Technology (iRAFIT2012)*, 2012, pp. 9-14.

[17] C. A. Hussain, D. V. Rao, and T. Praveen, "Color histogram based image retrieval," *Int J Adv Eng Technol IV/III,* pp. 63-66, 2013.

[18] S. M. Maniar, D. Pathak, M. Kadvani, and J. S. Shahsir, "Query by Image Content Using Color Histogram Techniques," *International Journal of Engineering Research,* vol. 2, 2013.

[19] M. H. Abed and D. S. J. Al-Farttoosi, "Content based Image Retrieval based on Histogram," *International Journal of Computer Applications,* vol. 110, 2015.

[20] P. Kar and L. Kumari, "Feature Based Image retrieval based on Color," *International Research Journal of Engineering and Technology* vol. 05, 2018.

[21] N. Chauhan and M. Goyani, "Enhanced multistage content based image retrieval," *Int. J. Comput. Sci. Mobile Comput,* vol. 2, pp. 175-179, 2013.

[22] G. Sorwar and A. Abraham, "DCT based texture classification using soft computing approach," *arXiv preprint cs/0405013,* 2004.

[23] T. Tsai, Y.-P. Huang, and T.-W. Chiang, "Image retrieval based on dominant texture features," in *2006 IEEE International Symposium on Industrial Electronics*, 2006, pp. 441-446.

[24] H. Kekre and K. Sonawane, "Retrieval of images using DCT and DCT Wavelet over image blocks," *(IJACSA) International Journal of Advanced Computer Science and Applications,,* vol. Vol. 2 2011.

[25] C. Bai, K. Kpalma, and J. Ronsin, "A new descriptor based on 2d dct for image retrieval," in *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2012, p. 4 pages.

[26] P. Hemalath, "Image Retrieval by content using DCT and RGB Projection," *International Journal of Computer Science & Communication Networks,* vol. 3, p. 134, 2013.

[27] C. Wang, X. Zhang, R. Shan, and X. Zhou, "Grading image retrieval based on DCT and DWT compressed domains using low-level features," *Journal of Communications,* vol. 10, pp. 64-73, 2015.

[28] M. Singha and K. Hemachandran, "Content based image retrieval using color and texture," *Signal & Image Processing,* vol. 3, p. 39, 2012.

[29] J. K. S and N. M. Jose, "An image Retrieval using EDBTC with YCbCR Features," *International Journal of Engineering Science and Computing,* vol. 6, 2016.

[30] L. V. Sree and K. Chaitanya, "Color Image Indexing by Exploiting the Simplicity of the EDBTC Method," *International Journal of Research,* vol. 04, 2017.

[31] A. Nazir, R. Ashraf, T. Hamdani, and N. Ali, "Content based image retrieval system by using HSV color histogram, discrete wavelet transform and edge histogram descriptor," in *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 2018, pp. 1-6.

[32] M. R. Girgis and M. S. Reda, "Content-based image retrieval using image partitioning with color histogram and wavelet-based color histogram of the image," *International Journal of Computer Applications,* vol. 104, 2014.

[33] A. S. Teynor, "Image Retrieval in the Compressed Domain Using JPEG2000," Department of Computer Science, Augsburg University of Applied Sciences, 2003.

[34] C.-H. Su, M. H. A. Wahab, and T.-M. Hsieh, "Image Retrieval based on color and texture features," in *2012 9th International Conference on Fuzzy Systems and Knowledge Discovery*, 2012, pp. 1816-1819.

[35] P. H. Chandankhede, "Soft Computing Based Texture Classification with MATLAB Tool," *International Journal of Soft Computing and Engineering,* vol. 2, 2012.

[36] K. Ponnmoli and D. S. Selvamuthukumaran, "Analysis of Face Recognition using Manhattan Distance Algorithm with Image Segmentation," *International Journal of Computer Science and Mobile Computing,* vol. 3, pp. 18-27, 2014.

[37] F. Memon, M. A. Unar, and S. Memon, "Image quality assessment for performance evaluation of focus measure operators," *arXiv preprint arXiv:1604.00546,* 2016.