ISSN: 0976-9102(ONLINE)
DOI: 10.21917/ijivp.2012.0059

OPTIMIZED IMAGE RETRIEVAL SYSTEM USING MULTIPLE THREADS

S.P. Jeyapriyamvadha[1], R. Arockia Xavier Annie[2] and P. Yogesh[3]

[1,3]Department of Information Science and Technology, College of Engineering, Anna University Chennai, India

E-mail: [1]spjeyapriya@gmail.com

[2]Department of Computer Science and Engineering, College of Engineering, Anna University Chennai, India

E-mail: annie@annauniv.edu

Abstract

Content-Based Image Retrieval is a technique used to retrieve similar images where the most challenging aspect is to bridge the gap between low level feature layout and high level semantic concepts. Efficient and effective retrieval techniques of images are desired to work out a certain image on the condition that the result would be more suitable than the input image. A novel method is proposed, which links various images as threads; where the texture features and shape features of the images are extracted and stored. The minimum distance between the query image and the thread provides the output. These threads are based upon the query result. The resultant retrieval system is found to be beneficial and interactive.

Keywords:

Content-Based Image Retrieval (CBIR), Feature Extraction, Interactive Search

1. INTRODUCTION

Many web search engines retrieve similar images by searching and matching textual metadata and the need to have a flexible and general purpose content based image retrieval (CBIR) system for a very large image database has attracted focus of many researchers of information-technology-giants and leading academic institutions for development of CBIR.

Content based image retrieval is the task of retrieving the images from the large collection of database. The ability to organise and retrieve visual information such as images and video has been a crucial problem for many researchers.

Image retrieval is the process of browsing, searching and retrieving images from a large database of digital images. The collection of images in the web are growing larger and becoming more diverse. Therefore retrieving images from such large collections is a challenging problem.

To understand this image process retrieval, the task is split into three phases; the query phase which allows the user to enter the query, next is the view phase in which the initial set of images are retrieved and finally the browse phase in which the user is allowed to choose an image from the obtained results to refine the search and get better results.

In order to categorize the images we need features as a basis, where we use multiple features of an image. Considering the visual features like color, texture, shape etc the features of a given image is extracted. The extracted features are grouped together based on the similarity space to find the images having similar features.

Here from the historical perspective the retrieval systems are mostly text based with the result from database management system since the images are required to be indexed or annotated. The result may not be accurate to the given keyword as it will retrieve the images as they are annotated; the user will be more concerned about the image content.

The drawback of the traditional technique is the retrieval of irrelevant images according to the image tags that match the input text. Hence to overcome this drawback we use certain methods like image processing for the efficiency of the search result. The results have been enhanced and another set of accurate results are got by following the methods.

2. RELATED WORK

Ork de Rooji et al. proposed a paper [1] which provides a refined browsing method to the user by creating two different threads; a dynamic retrieval process is done. The threads are calculated using the similarity and dissimilarity distances between the images that are present in the database. Based on the similarity distances, images are retrieved

Christopher C. Yang describes in his paper [6] about the image retrieval system and image processing that is to be done for effective retrieval; low level features are considered and based on the similarity measures the images are retrieved.

B.J. Lei et al. proposed a paper [7] which explains about the general and domain specific features that are used for image processing; these features provide the inventory properties of the images. The geometric descriptions are considered for the image and the gradient features are obtained.

Georgy Gimel'farb lectures a paper [8] about the texture feature extraction based on the various qualitative features of the image, each qualitative feature is considered to vary the texture.

Mohamed Rizon et al. proposed a paper[9] which explains about the invariant moments of the images to verify the shape of the object in the image, which is then used to categorize to which group they belong to.

P.B. Thawari et al. proposed a paper [10] which briefs out the concept of content based image retrieval based on the color and texture feature; where the visual patterns are observed and images are retrieved based on the similarity measures between the feature vectors.

Xiaojun Qi proposed a lecture [11] text based image retrieval and classifies the image results by finding out the precision of the results.

Ruofei Zhang et al. proposed a paper [12] on the efficiency of the retrieved results by acquiring their average precision to the image database. The accuracy is found based on the relevant images obtained in the result.

Darshak G. Thakore et al. proposed a paper [3] where image analysis is done to find out the region of interest and then feature extraction is done for further processing.

2.1 OUR CONTRIBUTION

The main contribution in our framework is to optimize the image search. Initially, three features are calculated; next they are clustered together and the similarity distance is found and a set of images are retrieved; then we refine the search by selecting an image from the retrieved results to get another set of results.

2.2 ORGANIZATION

The rest of the paper is organized as follows: Content based image retrieval is explained in section 3; section 4 explains the optimized video search; section 5 elaborates the system design, section 6 explains the experimental results and finally section 7 briefs out the conclusion.

3. CONTENT BASED IMAGE RETRIEVAL

In content based image retrieval the user gives an input either as text or an image itself. For the given input relevant images are retrieved from the image database. Finally the result is viewed by the end user. Based on the user satisfaction feedback mechanism is processed. Fig.1 explains the general framework for content based image retrieval system.

Our proposed framework will retrieve relevant images based on the input query. The framework begins with the static thread in which the user is allowed to enter the query an initial set of images are retrieved; then the user is allowed to select the image dynamically and at that instance the features are obtained and the similarity measures are calculated and now another set of images are retrieved which is the dynamic thread generation. The obtained results will be effective by the generation of threads.

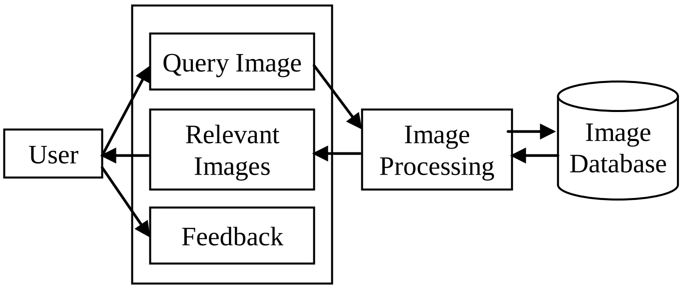

Fig.1. Frame work of Content based Image retrieval

4. OPTIMIZED IMAGE SEARCH

The optimized image search provides the user additional features for various facilities. We have defined three phases in video retrieval, the query phase, the view-phase and the browse-phase. In this section we discuss possible optimizations for each phase [1]

4.1 QUERY PHASE

In the query phase the user is allowed to enter the query by text and query by visual data. The consequence of increasing the options is provided to obtain a better set of results. Also, since the quality of individual feature extraction methods vary, retrieval sometimes leads to low quality results. These two problems increase the number of times a user has to go back and forth between the query and view phases [1]. To overcome this problem we combine the query and view phase together in the browse phase in which the user is allowed to select the image and another set of relevant images are retrieved.

4.2 VIEW PHASE

In this phase, the user is allowed to view the initial set of retrieved images. This is where the retrieved images are displayed. Similarity based visualizations have utility, but by limiting the user to view results which are already similar to each other there is a possibility that other parts of the dataset, which are relevant but not similar, are never visited [1]. View phase optimization is possible by reducing the time required for the user to decide that results are not relevant. From the query phase we obtain a single ranked list of results. In its simplest form this list is visualized as a linear list. In practice most systems extend this to a grid-based representation, such as in [2], and use implicit reading order to display the ranked list. This allows for constrained and fast navigation through the list. However, no matter how the list is displayed, it still constrains the quality of the list obtained in the query phase [1]. The only decision to be done is either be done with the results or go back to the query phase; this can be avoided by making the user to choose from the view phase itself.

4.3 BROWSE PHASE

The browse phase allows the user to navigate away from the initial set of results, and delve into the dataset based on the features themselves. An efficient browse-phase therefore reduces the number of times a user has to switch between phases, which improve search task efficiency [1]. The browse phase lists another set of results in which the refined search is done and the images are retrieved. The view phase makes the user feel convenient as the user will not have to navigate between the query and view phase. Optimizing the search phase increases the quality of the set of results. Optimizing the view phase decreases the time required to determine if the set of results is adequate. Finally optimizing the browse phase reduces the number of times a user has to switch between the query and view phases. We propose an image retrieval system which primarily focuses on browse phase optimization [1].

5. SYSTEM DESIGN

The designed framework retrieves images efficiently by the various steps as followed in the content based image retrieval. The framework is briefed out as follows,

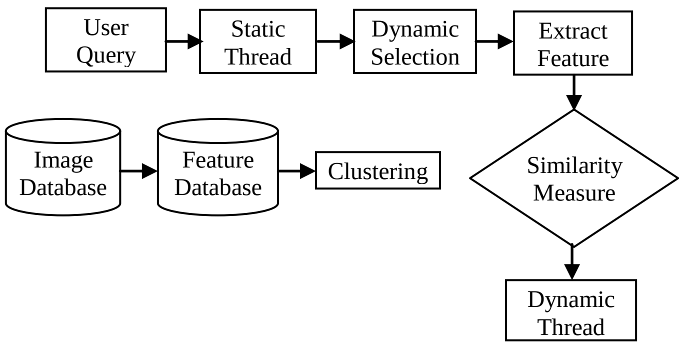

Fig.2. System Design

5.1 FEATURE EXTRACTION

General and domain specified features like texture and shape are used. Primitive features characterizing image content, such as texture and shape are computed for both stored and query images, and used to identify the stored images most closely matching the query. Each set of features vary according to the image possessed.

5.1.1 Texture Feature:

The texture is the visual pattern that relates mostly to a specific, spatially repetitive structure formed by repeating a particular element or several elements. The repetition involves local variations of scale, orientation, or other features of the elements, like coarseness, contrast, directionality, roughness, randomness, etc., they are derived from [8].

To derive the coarseness feature vector the average best size of the image is found out. At each pixel (x,y), find the value of k that maximizes the difference $Ek(x, y)$ in either direction, and at each pixel (x, y), set the best size,

$$S_{best}(x, y) = 2^k. \quad (1)$$

The contrast feature Measures how grey levels vary in the image g and to what extent their distribution is biased to black or white:

$$F_{con} = \sigma/(\alpha_4)^n \quad (2)$$

where,

$\sigma^2 = \sum_{q=0}^{q\max} (q - m)^2 \Pr(q|g)$ is the variance

$\alpha_4 = \frac{1}{\sigma^4}\sum_{q=0}^{q=0} (q - m)^4 \Pr(q|g)$ is the kurtosis

$n = 0.25$ is recommended as the best for discriminating the textures.

The directionality feature is found by considering the frequency distribution of oriented local edges against their directional angles. The edge strength $e(x, y)$ and the directional angle $a(x, y)$ are computed using approximate pixel-wise derivatives computed by the Sobel edge detector in the 3x3 moving window:

$$\begin{matrix} -1 & 0 & 1 \\ -1 & \mathbf{0} & 1 \\ -1 & 0 & 1 \end{matrix}\ \Delta x\,(x, y) \qquad \begin{matrix} 1 & 1 & 1 \\ 0 & \mathbf{0} & 0 \\ -1 & -1 & -1 \end{matrix}\ \Delta y\,(x, y)$$

$$e(x, y) = 0.5(\Delta x\,(x, y) + \Delta y\,(x, y));$$

$$a(x, y) = \tan^{-1} \Delta y\,(x, y)\, \Delta x\,((x, y))$$

The degree of directionality relates to the sharpness of peaks:

$$F_{dir} = 1 - r.n_p \sum_{p=1}^{n_p} \sum_{a=w_p} (a - a_p)^2 H_{dir}(a) \quad (3)$$

where,

n_p - The number of peaks

a_p - The position of the pth peak

w_p - The range of the angles around the pth peak

r - A normalizing factor related to quantizing levels of a

a - The quantized directional angle (modulo 180^o)

5.1.2 Shape Feature:

Geometric moment invariant produces a set of feature vectors that are invariant under shifting, scaling and rotation. Moment invariants are important shape descriptors in computer vision. GM technique is chosen to extract image features since the features generated are Rotation Scale Translation (RST) invariant, derived from [9].

Two-dimensional moments of a digitally sampled $M\times M$ image that has gray function $f(x, y)$, $(x, y=0...M-1)$ is given as,

$$m_{pq} = \sum_{x=0}^{x=M-1} \sum_{y=0}^{y=M-1} (x)^p.(y)^q f(x, y) \quad (4)$$

$p,q = 0,1,2,3\ldots$

The moments $f(x, y)$ translated by an amount (a, b), are defined as,

$$\mu_{pq} = \sum \sum (x+a)^p.(y+a)^q f(x,y) \quad (5)$$

Thus the central moments m_{pq} or μ_{pq} can be computed from Eq.(5) on substituting $a = x'$ and $b = y'$ as,

$$x' = m_{10}/m_{00} \text{ and } y' = m_{01}/ m_{00}$$

$$\mu_{pq} = \sum \sum (x-x')^p.(y-y')^q f(x,y) \quad (6)$$

Thus the shape feature is extracted using the moment invariants by scaling and translating them.

5.2 CLUSTERING

Clustering the feature vectors will group the similar features into a single cluster. Here, k-means clustering is used in which a data point is assigned to a particular cluster and points that are closest to the data point. The center of the cluster is updated and the process is repeated until all the features are clustered. After clusters are obtained from distinct features we store them for the further process. The clusters will be used to list out the images according to the input query given.

5.3 THREAD GENERATION

This section gives an overview of the proposed method for searching through the images retrieved using threads. Threads are computed from several types of features; each thread describes the image retrieval and the accuracy of the results. Finally, we describe a generic framework which uses threads to allow users in navigating image retrieval results.

5.3.1 Static Thread Generation:

A static thread is a thread where the order is determined by a structure induced by dissimilarity space only. Static threads dissect the entire dissimilarity space into individually meaningful threads [1], where the features are saved and the initial set of results are obtained by retrieving the images as per the given query. The static thread retrieves the images based on the text query given by the user. This may result in a set of retrieval system in which there may be irrelevant images; hence to aid a better option to the user we use the dynamic thread generation.

5.3.2 Dynamic Thread Generation:

The user is allowed to select the image dynamically and at that instance the features are obtained and the similarity measures are calculated and now another set of images are retrieved which is the dynamic thread generation. Dynamic thread generation reduces the semantic gap by providing more accurate results.

Summarizing the above, both static and dynamic threads are based on a dissimilarity space for a specific feature. As a consequence many different threads types are computable. The similarity space between feature of the input image and the cluster is found by using the Euclidean distance between the cluster and the input image feature, and the minimum distance is displayed out.

$$d(p,q)=sqrt(\sum_{i=0}^{n}(q_i - p_i)^2) \quad (7)$$

6. EXPERIMENTAL RESULTS AND EVALUATION

6.1 EXPERIMENTAL RESULTS

The proposed system has been implemented using different kinds of images. The framework has been implemented in Matlab (2009b). Some results are reported that are obtained in the image retrieval system.

Table.1. Minimum distance between the query image and the output images

Query Image/ output Image	1	2	3	4	5
1	1.154	1.1963	1.3090	1.4800	1.5139
2	0.0725	0.7530	0.9990	1.1176	1.1625
3	0.0678	0.5380	0.7122	0.7578	0.7834
4	1.1492	1.4026	1.8649	1.8784	1.9261
5	0.1006	1.3602	1.4297	1.8043	1.8974

Fig.3. The framework to enter the query

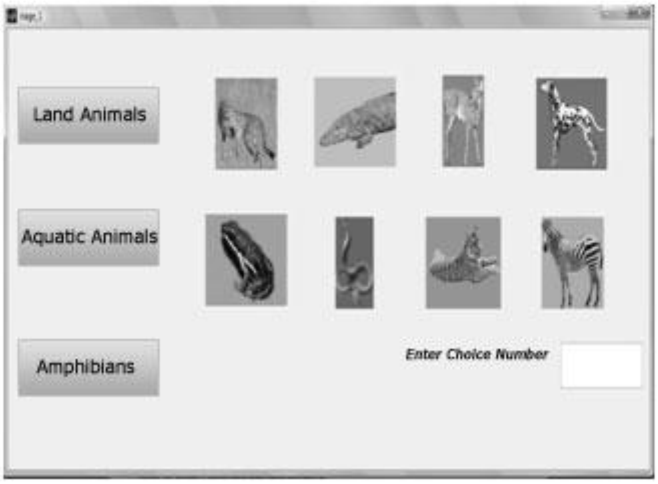

Fig.4. Initial set of retrieved images

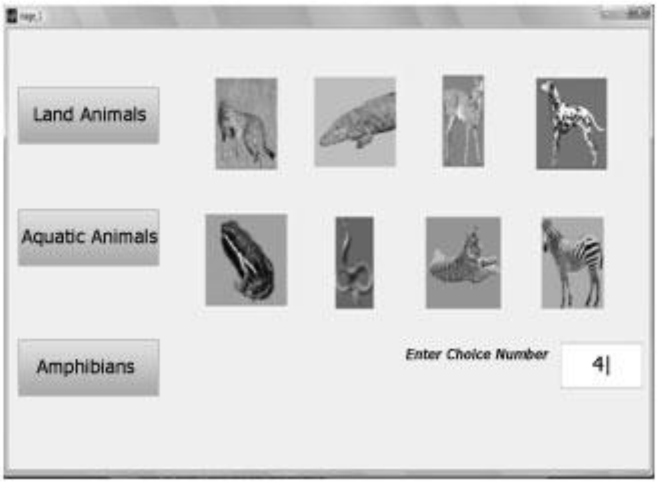

Fig.5. The user selects the image as input

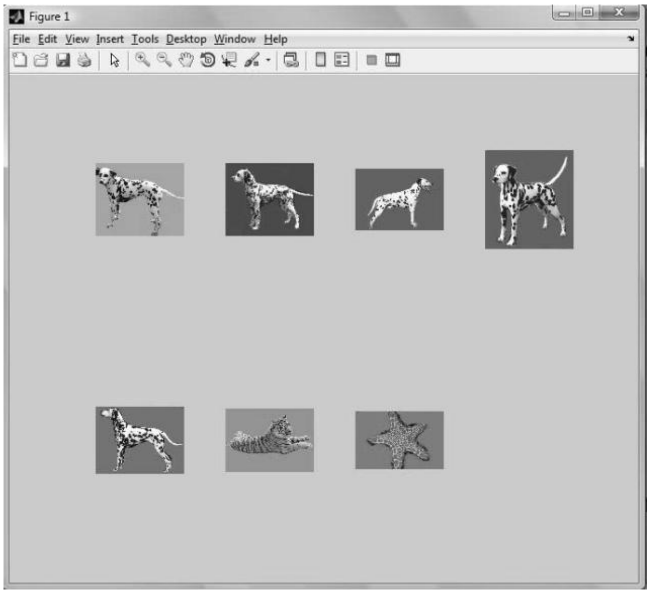

Fig.6. The refined set of images retrieved

Fig.2 shows the framework where the user is allowed to enter the query, Fig.3 shows the initial set of retrieved images, Fig.4

the user can select the image and then the final set of images are retrieved in Fig.5

6.2 EVALUATION OF THE THREADS

The performance evaluation is done by finding out the precision and recall.

$$\text{Precision} = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images retrieved}}$$

$$\text{Recall} = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images in the database}}$$

Table.2. Precision values and Recall values

Query Image	Precision values	Recall values
1	0.71	0.62
2	0.85	0.55
3	0.57	0.85
4	0.71	0.44
5	0.85	0.66

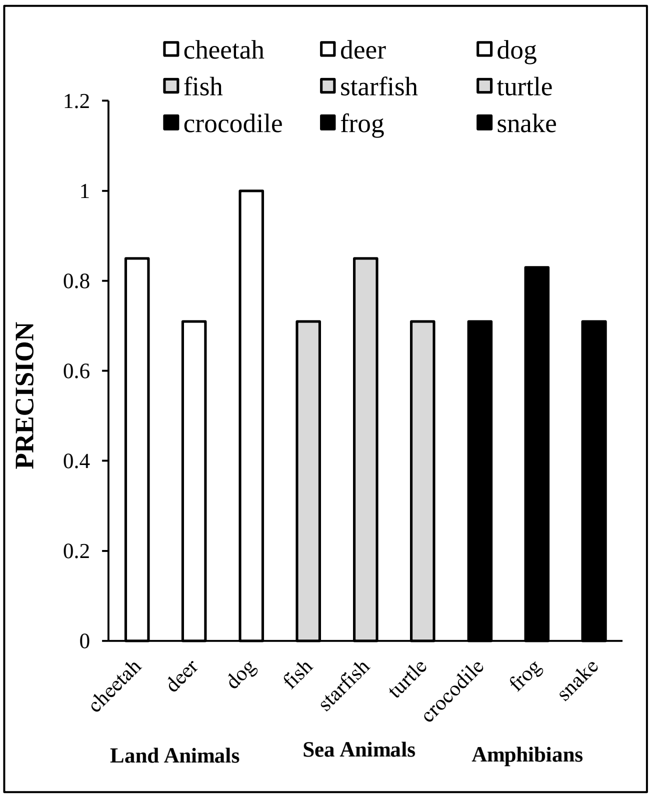

Fig.7. Precision of the result

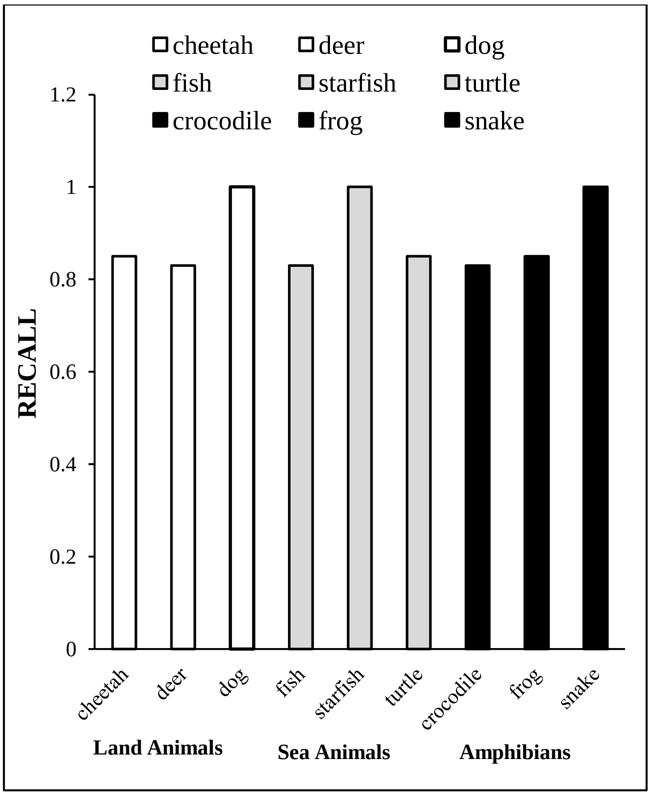

Fig.8. Recall of the result

The largest portion of results is generated from the initial query thread and the timeline. From the other threads the visual thread were used most.

7. CONCLUSION

This paper presented a method for browsing images using threads. Two types of threads are identified: static threads and dynamic threads. We have evaluated both the benefit of having these threads available during search, and the benefit of having static threads available in addition to dynamic threads. The features were extracted from the images and were grouped as threads based on the distance between them. Initial result provided a set of images, from which another query image was selected dynamically to obtain a refined search. The overall evaluation results indicate that browsing through multiple threads is beneficial for image search. It is important to offer the user to make use of the opportunity to interactively select the threads deemed useful for a certain search task. By doing so, thread-based browsing provides intuitive and flexible image retrieval.

REFERENCES

[1] Ork de Rooji and Marcel Worring, “Browsing video along multiple threads” *IEEE Transactions on Multimedia*, Vol. 12, No. 2, pp. 121-130, 2010.

[2] M.G. Christel, C. Huang, N. Moraveji, and N. Papernick, “Exploiting multiple modalities for interactive video retrieval,” *in Proceedings of the IEEE International*

Conference on Acoustic, Speech and Signal Processing, Vol. 3, pp. 1032–1035, 2004.

[3] Darshak G. Thakore and A.I. Trivedi, “Content based image retrieval techniques – Issues, analysis and the state of the art”, *Proceedings of International Symposium on Computer Engineering and Technology*, pp. 98 – 102, 2010.

[4] Gaurav Jaswal and Amit Kaul, “Content Based Image Retrieval – A Literature Review”, *National Conference on Computing, Communication and Control*, pp. 198-201, 2009.

[5] B. Li and S.D. Ma, “On the relation between region and contour representation”, *in Proceedings of the IEEE International Conference on Image Processing*, 1995.

[6] Christopher C. Yang, “Content based image retrieval: a comparison between query by example and image browsing map approaches” *Journal of Information Science*, Vol. 30, No. 3, pp. 254-267, 2004.

[7] B.J. Lei, Emile A. Hendriks and M.J.T. Reinders, “On Feature Extraction from Images” *Mediated Communication in Collaborative Work and Service project, Information and Communication Theory Group*, pp1-57, 1999.

[8] Georgy Gimel’farb, “CBIR-Texture Features-1”, Lecture notes G7, COMPSCI.708.S1C, 2006.

[9] Mohamed Rizon, et al., “Object Detection using Geometric Invariant Moment” *American Journal of Applied Sciences*, Vol. 3, No. 6, pp 1876-1878, 2006.

[10] P.B. Thawari and N.J. Janwe, “CBIR Based on Color and Texture” *International Journal of Information Technology and Knowledge Management*, Vol. 4, No. 1, pp. 129-132, 2011.

[11] Xiaojun Qi, “Content Based Image Retrieval”, Lecture on Image Processing, 1998.

[12] Ruofei Zhang and Zhongfei (Mark) Zhang “Addressing CBIR Efficiency, Effectiveness, and Retrieval Subjectivity Simultaneously”, *Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval*, pp. 71-78, 2003.

[13] M.S. Lew, N. Sebe, C. Djeraba, and R. Jain, “Content-Based Multimedia Information Retrieval: State of the Art and Challenges,”*ACM Transactions on Multimedia Computing, Communication and Applications*, Vol. 2, No. 1, pp. 1-19, 2006.

[14] B.S. Manjunath and W.Y. Ma, “Texture Features for Browsing and Retrieval of Image Data” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 8, pp. 837-842, 1996.