

An Efficient Scalable Graph Based Ranking Model for Content Based Image Retrieval

V. Lokeswara Reddy*

Department of C.S.E, K.S.R.M College of Engineering, Kadapa, Y.S.R. District., A.P, India

*Corresponding author: VL_REDDY@YAHOO.COM

Abstract As the multimedia system technologies have become more popular, users do not satisfied with the standard retrieval techniques, thus today the content based image retrieval is becoming source of exact and for quick retrieval. The last decade has witnessed great interest in research on content based image retrieval. In this paper a graph based ranking model has been proposed and successfully applied to content-based image retrieval, due to its outstanding ability to find underlying geometrical structure of the given image database. The Admin have control to add, delete and modify the image database and therefore the user will search the image that need to be accessed and later the graph is generated based on user search. Experimental results show that the proposed technique has high accuracy than other conventional methods for generating the graph.

Keywords: content based image retrieval, ranking model, image retrieval, efficient ranking model

Cite This Article: V. Lokeswara Reddy, "An Efficient Scalable Graph Based Ranking Model for Content Based Image Retrieval." *Journal of Computer Sciences and Applications*, vol. 4, no. 3 (2016): 52-58. doi: 10.12691/jcsa-4-3-1.

1. Introduction

Digital imaging was a great invention within twentieth century. Since digital cameras became popular, a large amount of digital images emerged in the late of the twentieth century. The way to manage the huge amount of images and find desired images among them became an urgent issue during the same period. Techniques of retrieving a desired image [1] are generally categorized into two basic classes. One is based on text-based key words to retrieve desired images in the image database. The other one based on image-based queries to retrieve desired images in the image database which is usually named as content-based image retrieval (CBIR) technique [2].

The Graph-Based ranking models are being used in information retrieval area. This paper mainly focused on the problem of applying a novel and efficient graph-based model for content based image retrieval (CBIR) [3]. Traditional image retrieval systems are based on keyword search, such as Bing and Google image search. In these systems, a user keyword (query) is matched with the context around an image including the title, manual annotation, web document, etc. These systems don't utilize information from images. However these systems suffer many problems, such as shortage of the text information and inconsistency of the meaning of the text and image. Content-based image retrieval is a considerable choice to overcome these difficulties. CBIR has drawn a great attention in the past two decades.

CBIR methods are viable solutions to find desired images from digital image libraries. In a basic CBIR system, all digital images in a library are represented by

their visual features (e.g., visual contents of images). Typical visual features include colors, shapes, edges, and textures to represent an image from different visual perspectives. Initially, these visual features are extracted from each image and stored in a feature database corresponding to the digital image library to facilitate the future use. When a query image is submitted to the system, visual features of the query image are first extracted. A matching technique is then utilized to compare the similarity between visual features of the query image and visual features of all digital images in the image database. Only those images having higher similarity scores are returned to the user as the retrieval results.

However, as the ranking of retrievals is calculated based on selected image features, the retrieval accuracy could also be unsatisfactory due to the semantic gap between low-level visual features and high-level semantic concepts. This semantic gap exists because images of similar semantic content may be scattered far away from each other in the feature space, while images of dissimilar semantic content may share similar low level features. For example, given a query image with a black horse in the front view, an image with a white horse in a side view is considered similar to the query image from the view point of the semantic concept. However, the front view of a black horse looks very different from the side view of a white horse, so are their visual features (i.e., their visual features are different).

2. Proposed System

The proposed mechanism is an extension of manifold ranking. In this paper, the original manifold ranking is extended and proposed an Efficient Model to achieve the

shortcomings of manifold ranking from two perspectives: the first is scalable graph construction; and the second is efficient computation, especially for out-of-sample retrieval. Specifically, we build an anchor graph on the database instead of the traditional k -nearest neighbor graph, and design a new form of adjacency matrix utilized to speed up the ranking

computation. The model has two separate stages: an offline stage for building (or learning) the ranking model and an online stage for handling a new query. With this model, can handle a database with 1 million images and do the online retrieval in a short time. The high-level block diagram of a basic CBIR system [3,5] is shown in Figure 1.

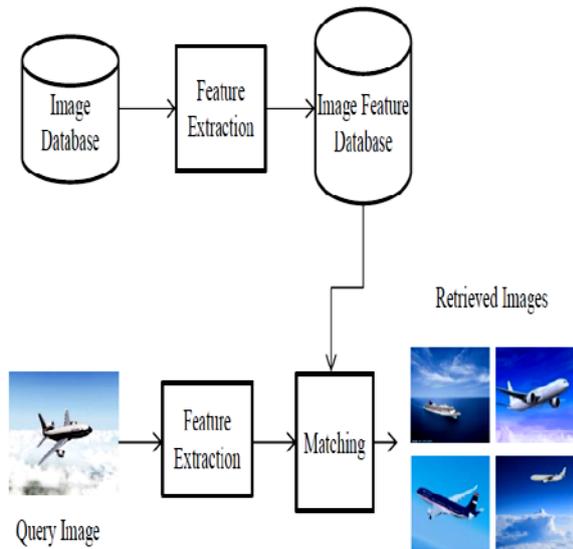


Figure 1. Block diagram of basic CBIR system

2.1. Efficient Ranking Model

In this Efficient Ranking Model the shortcomings of Manifold Ranking model by constructing a scalable graph and efficient ranking [6] computations are addressed. The process of scalable graph construction and ranking computation is as follows.

2.2. Scalable Graph Construction

In this Scalable Graph Construction to handle huge databases, we want the graph construction cost to be sub-linear with the graph size. That means, for each data point; we can't search the whole database as k -NN strategy does. To achieve this requirement, an anchor graph has been constructed and proposed a new design of adjacency matrix W .

2.2.1. Anchor Graph Construction

Anchor graph [7] is a low-rank approximation of neighborhood graph (such as k -NN), where the similarity between data points are measured by a small number of anchor points. In this paper, k -means algorithm has been used and selected the centers as anchors. The main advantage of building an anchor graph is separating the graph construction into two parts – anchor selection and graph construction. Each data sample is independent to the other samples but related to the anchors only. The construction is always efficient since it has linear

complexity to the data size. Note that we don't have to update the anchors frequently, as informative anchors for a large database are relatively stable (e.g., the cluster centers) even if a few new samples are added.

2.2.2. Design of Adjacency Matrix

In this paper, a new approach presented to design an adjacency matrix W and which is big advantage to handle some high dimensional data. The weight matrix $Z \in R^{d \times n}$, can be seen as a d dimensional representation of the data $X \in R^{m \times n}$, d is the number of anchor points. That is to say, data points can be represented in the new space, no matter what the original features are.

Then, with the inner product as the metric to measure the adjacent weight between data points, the adjacency matrix to be a low-rank form [8] is designed.

$$W = Z^T Z \quad (1)$$

Which means that if two data points are correlative ($W_{ij} > 0$), they share at least one common anchor point, otherwise $W_{ij} = 0$. By sharing the same anchors, data points have similar semantic concepts in a high probability as our consideration. Thus, our design is helpful to explore the semantic relationships in the data.

2.3. Efficient Computation

After graph construction, the main computational cost for manifold ranking is the matrix inversion whose complexity is $O(n^3)$. So the data size n cannot be too large. Although we can use the iteration algorithm, it is still inefficient for large scale cases.

During the computation process, the adjacency matrix W does not used. So the matrix W is not saved in memory, but save matrix Z instead.

2.4. Efficient Ranking for CBIR

In this part, efficient ranking applied to pure content-based image retrieval have been summarized. For more information, the data features are extended as follows.

Step 1: Extract the low-level features of images in the database, and use them as coordinates of data points in the graph.

Step 2: Select representative points as anchors and construct the weight matrix Z with a small neighborhood size s . Anchors are selected off-line and does not affect the on-line process.

Step 3: When the user specifying or uploading an image as a query, get or extract its low-level features, update the weight matrix Z

Step 4: At last, Images with highest ranking scores are considered as the most relevant and return to the user.

2.5. Complexity Analysis

In this section, the complexity analysis of Efficient Ranking is represented; in this both computation cost and storage cost of Efficient Ranking are included.

- In this Efficient Ranking an anchor graph has been generated, *i.e.*, for each data sample, the relationships to its s -nearest anchors are calculated. The computation cost is $O(nd \log s)$. The k -means is used to select the anchors, and need a cost of $O(Tdn)$, where T is the iteration number. But this selection step can be done off-line and unnecessarily

updated frequently. At the same time, the sparse matrix $Z \in R^{d \times n}$ with a storage cost $O(sn)$ has been saved.

- In the ranking computation stage, complexity is $O(dn + d^3)$.

3. Experimental Work & Results

The proposed system is implemented by using JSP, HTML languages as frontend and database MySQL is used as backend to store the data of images. In this system

there are two login windows, which are represented as Admin login window and User login window. In Admin login window the Admin can login and add, delete and modify the data that is to be accessed by the user. In user login window the user can login and access the data. The proposed system Admin login window is shown in Figure 3. In this window the Admin can login by providing username and password to access the database. The login window is represented in Figure 2 as follows.

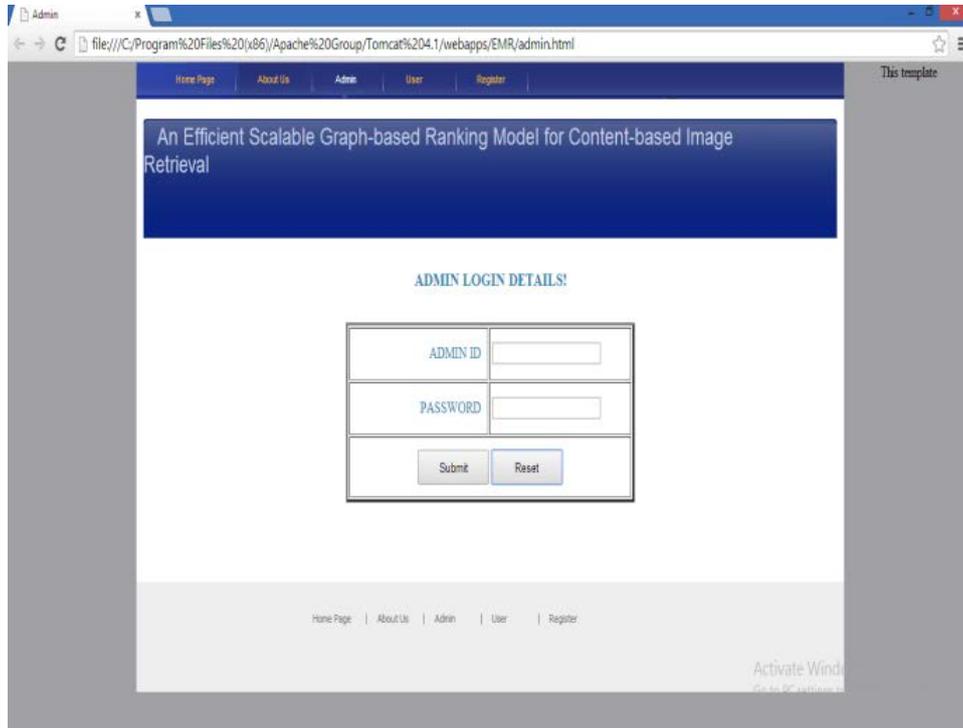


Figure 2. Admin login window

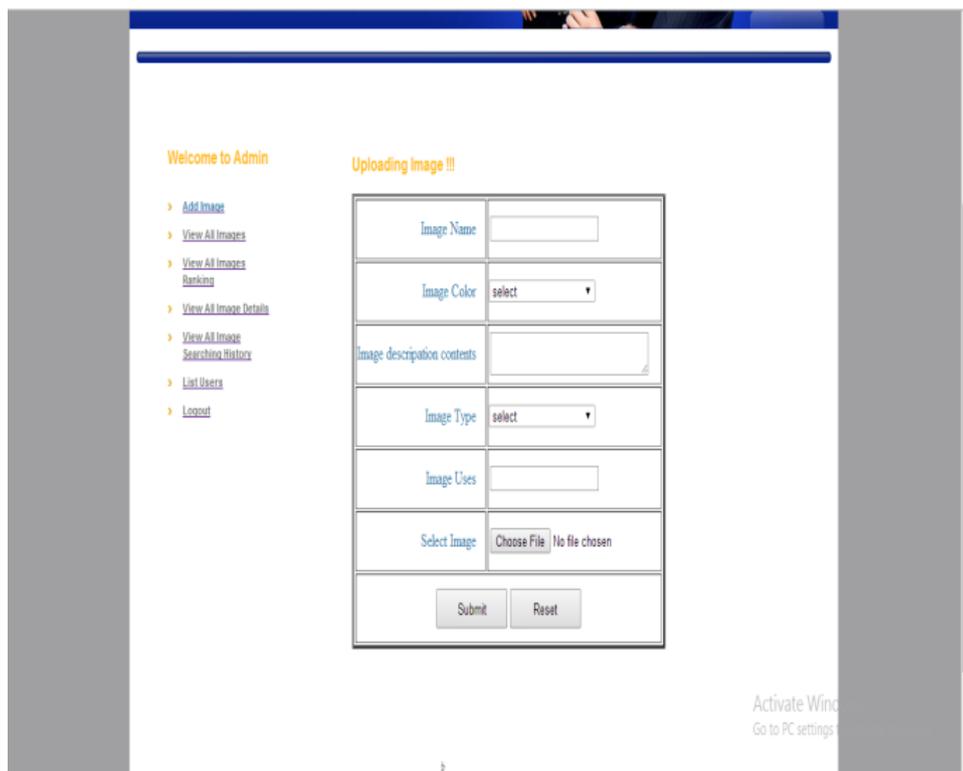


Figure 3. Image adding window

After logging the admin has features like admin can add images of different types, view all images in the database, view all image rankings, view all image details, view all

searching history by users and also view list of users accessing the database.

Figure 3: Represents the window in which the admin can add the images to the database.

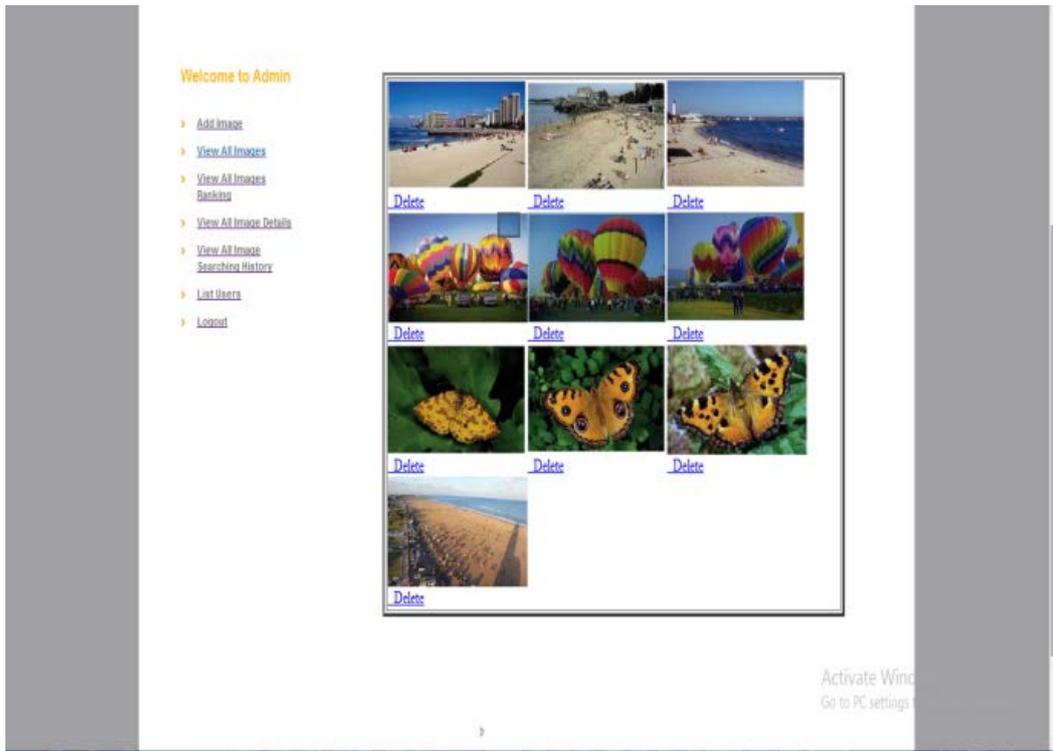


Figure 4. Window to view Images in the database

The Figure 4 represents the window in which the admin can view the list of images in the image database.

The Figure 5 represents the window which can be accessed by admin to view the rank of the image in the databases which are searched by the user.

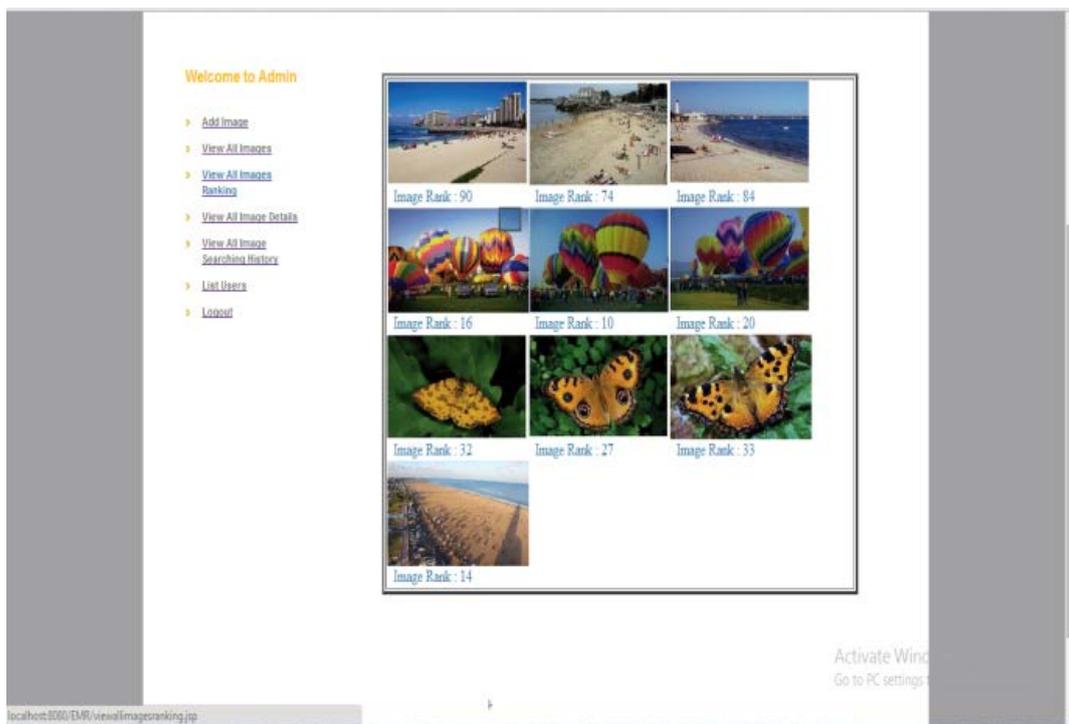


Figure 5. Window to view Rank of the image

The Figure 6 represents the window that can be accessed by the admin to edit or to view the complete

description of the image like image type, image description, image rank, image points etc.

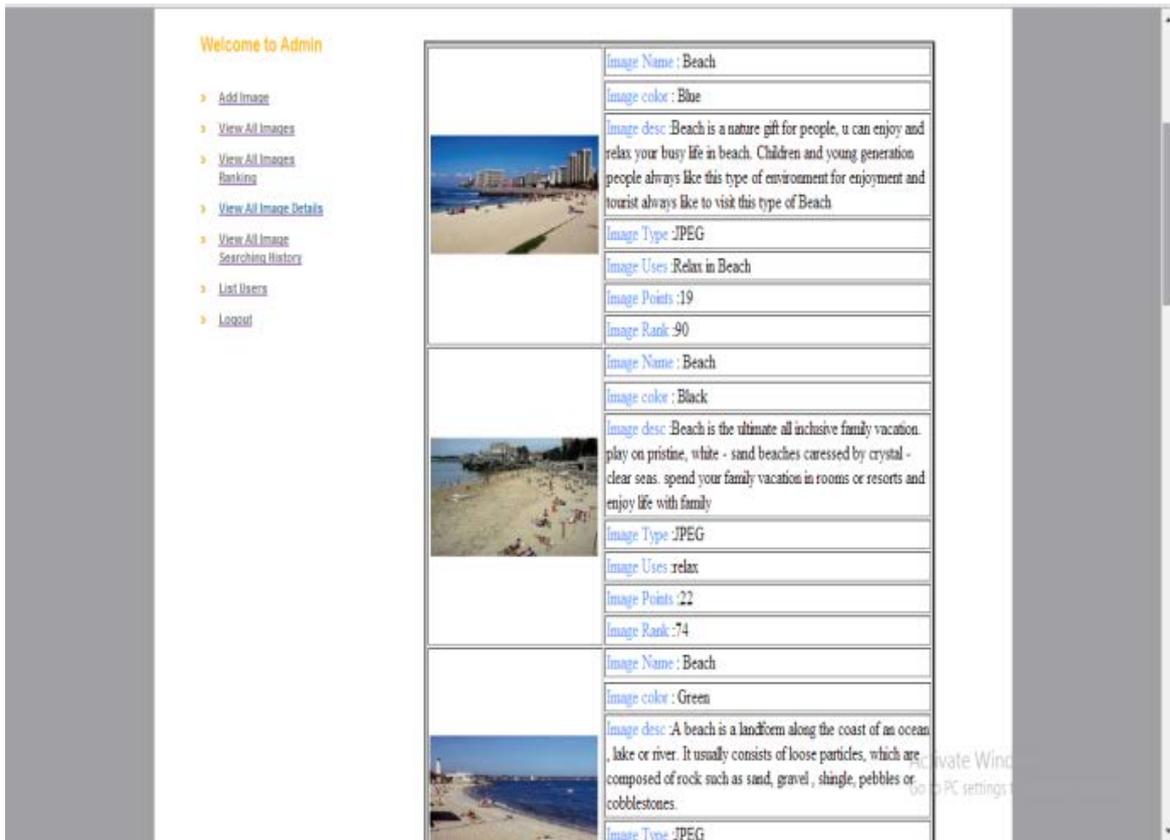


Figure 6. Window to view complete description of images

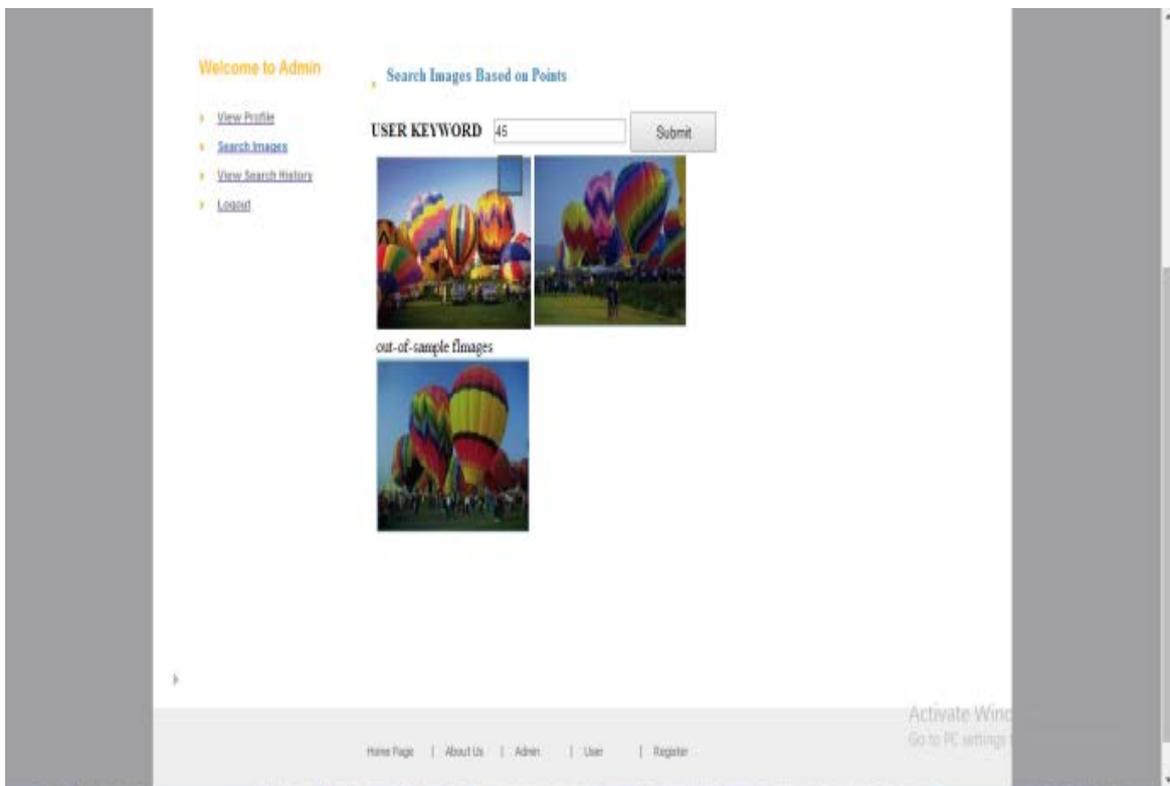


Figure 7. Search results after performing point search

The Figure 7 represents the search results after performing the point based search by the user. The user insert the points that are given to the image, if the points match with the image then the respective image with points can be shown to the user otherwise it shows null to the user.

The Figure 8 represents the search results after performing the content based search by the user. The user insert the query like “Beach” then search result with query will be shown to user.

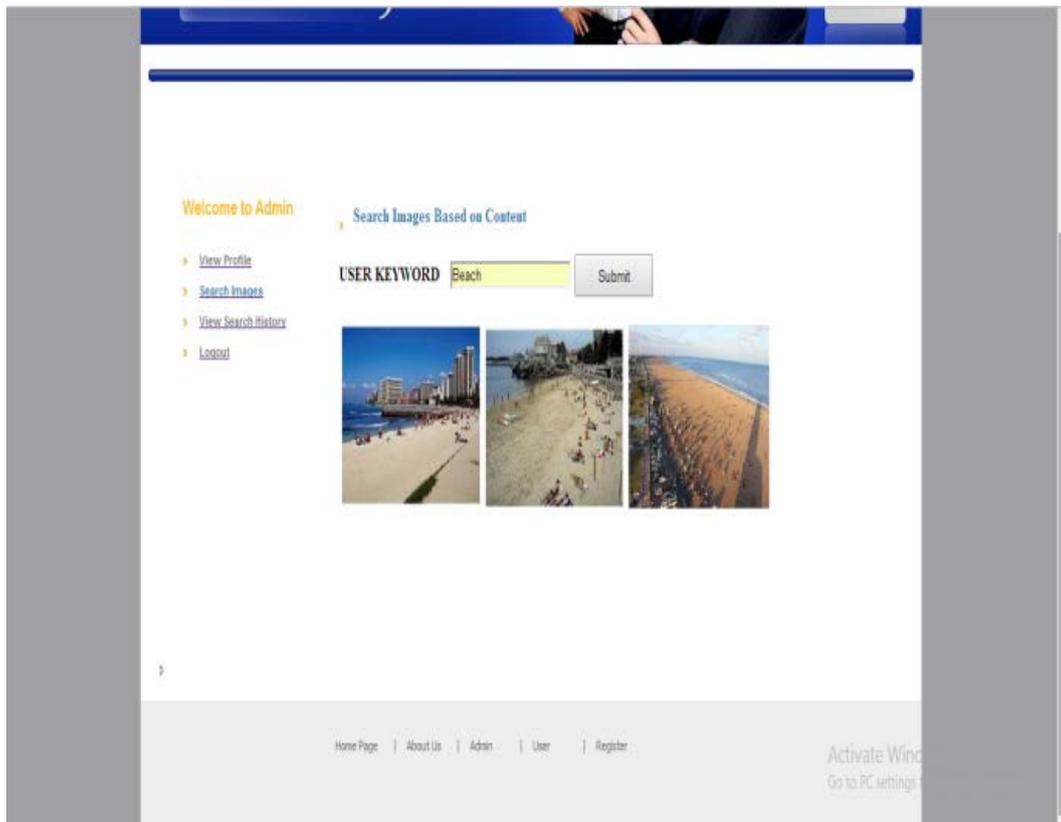


Figure 8. Search results after performing content based search

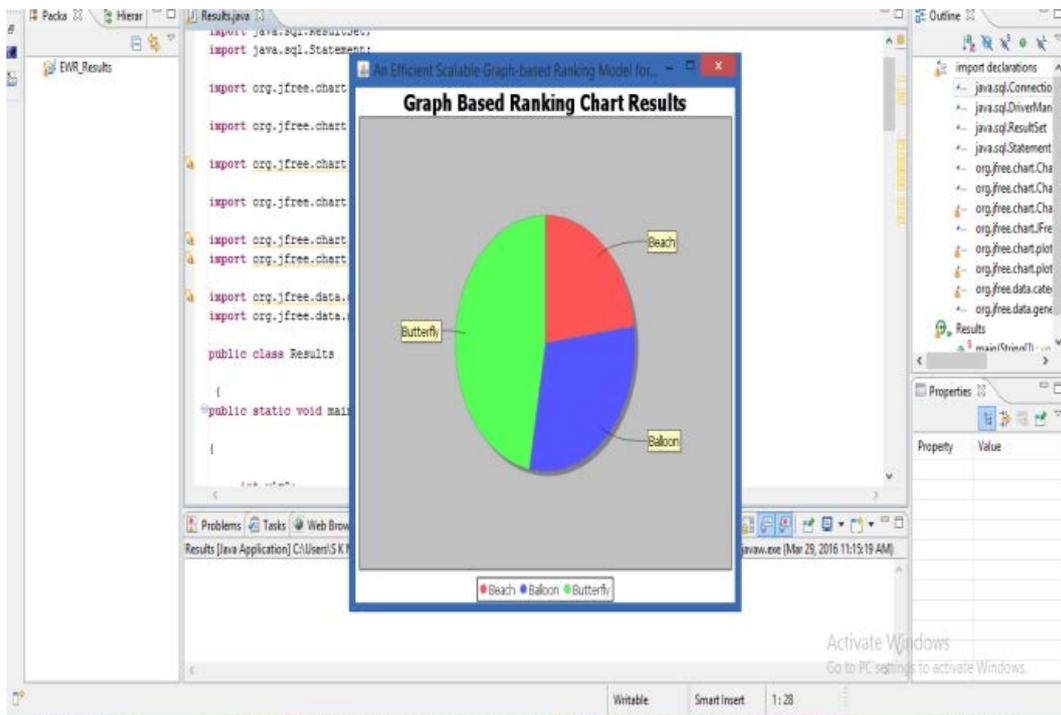


Figure 9. Graph ranking chart for search results

The Figure 9 represents the generated graph after performing the search by the user. This graph can be viewed by the admin to know the user interest.

4. Conclusion

In this paper, an efficient scalable ranking model to handle the large scale database effectively when compared

to original Manifold Ranking Model has been proposed. Efficient scalable graph-based ranking model to a content based image retrieval application based on a real world image database has been applied. By implementing this system, it is easy to manage huge database and generate graph related to image searched by the user effectively. Efficient scalable graph based ranking model tries to overcome the disadvantages of Manifold Ranking by generating effective scalable graph and also reduces the

computational time. This mechanism is feasible to large scale image retrieval systems and also reduces the storage space.

References

- [1] R. Datta, D. Joshi, J. Li, and J. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM CSUR*, vol. 40, no. 2, pp. 1-60, 2008.
- [2] R. C. Veltkamp and M. Tanase, "Content-based image retrieval systems: A survey," Dept. Computing Science, Utrecht University, Utrecht, The Netherlands, Tech. Rep. UU-CS-2000-34, 2000.
- [3] Chang, Ran, "Effective Graph-Based Content-Based Image Retrieval Systems for Large-Scale and Small-Scale Image Databases" (2013). All Graduate Theses and Dissertations. Paper 2123.
- [4] Y. Liu, D. Zhang, G. Lu, and W. Ma, "A survey of content-based image retrieval with high-level semantics," *Pattern Recognit.*, vol. 40, no. 1, pp. 262-282, 2007.
- [5] J. He, M. Li, H. Zhang, H. Tong, and C. Zhang, "Manifold ranking based image retrieval," in *Proc. 12th Annu. ACM Int. Conf. Multimedia*, New York, NY, USA, 2004, pp. 9-16.
- [6] B. Xu et al., "Efficient manifold ranking for image retrieval," in *Proc. 34th Int. ACM SIGIR*, New York, NY, USA, 2011, pp. 525-534.
- [7] W. Liu, J. He, and S. Chang, "Large graph construction for scalable semi-supervised learning," in *Proc. 27th Int. Conf. Mach. Learn.*, Haifa, Israel, 2010, pp. 679-686.
- [8] D. Cai, X. He, and J. Han, "Using graph model for face analysis," *Comput. Sci. Dept., UIUC*, Champaign, IL, USA, Tech. Rep. UIUCDCS-R-2005-2636, 2005.