COPYRIGHT © 2010 JCIT, ISSN 2078-5828 (PRINT), ISSN 2218-5224 (ONLINE), VOLUME 01, ISSUE 01, MANUSCRIPT CODE: 100712

A Modularized Approach for Similarity Based Object Retrieval

Abu Shamim Mohammad Arif, A. M. Yunus Ali, and S.M.Rokibul Hasan

Abstract—To improve the effectiveness of similarity based object retrieval system, in this paper we proposed a modularized approaches for similarity based object retrieval. Unlike existing method it is a two stage similarity searches. In the first stage, we perform similarity comparison directly between query image with database image. After getting the output of the first stage then we perform a second stage similarity comparison by modularized query image as well as database images. Here we perform similarity comparison between corresponding module of query and database image. Due to the two stage similarity comparison with modularized approaches our system eliminating difficult false positive significantly. Another concern of our system is that it can retrieve similar object even if the images are affected with contrast and brightness variations etc.

Index Terms— Image Preprocessing, Segmentation, Feature Extraction, Similarity Comparison, Modularization.

1 Introduction

SIMILARITY-BASED retrieval of images is an important task in many image database applications. It is an object retrieval technique where an input image (which called query image that is the part of an image) is match against a collection of image from image database. It has wide application in area of crime prevention, intellectual properties, medical diagnosis, education, web searching etc. In a general object retrieval system, for a single query object identify some representative features of the object. These representing features can be used to retrieve objects that are likely to be a close match to the query object. This problem is known as Content-Based Image Retrieval CBIR [3] system. These approaches often use features that represent the entire image. But when the query object is not the whole image but a portion of an image then the content based image retrieval is complex. Then this problem is refers to as Similarity-based object retrieval (SBOR) [1]. There are two approaches to the SBOR technique: data-independent and data-dependent [3]. In data dependent approaches, apply object segmentation algorithms to extract objects from images. After extracting objects from images perform similarity search on feature vectors that are representing individual objects. On the other hand, in the data-independent approach, images are divided into overlapping or non-overlapping rectangular regions or tiles. Then feature vectors are extracted from each tile and stored in a database for similarity search. In this paper we use data independent approaches. In order to improve the effectiveness of similarity-based object retrieval system, here we have proposed a Modularized Approaches for Similarity-Based Object Retrieval (MSBOR) System.

The remainder of the paper is organized as follows. We will discuss related work in the next section. In section 3, our proposed method is introduced. In section 4, we will describe the experiments that we have performed and provide results. We conclude in section 5.

- Abu Shamim Mohammad Arif is with Computer Science and Engineering Discipline, Khulna University, Khulna-9202, Bangladesh. E-mail: shamimarif@yahoo.com.
- A. M. Yunus Ali is with Computer Science and Engineering Discipline, Khulna University, Khulna-9202, Bangladesh. E-mail: rasel_cse_ku_05@yahoo.com
- S.M.Rokibul Hasan with Computer Science and Engineering Discipline, Khulna University, Khulna-9202, Bangladesh. E-mail: rokib.cse05@gmail.com.

Related Works

This paper describes a preliminary implementation of a SBOR system for simulation data by Sen-ching S.Cheung and Chandrika Kamath [1]. Here three major modules in the system: graphical user interface, feature extraction, and similarity search. In a typical similarity search, a user first opens an image from the image database and defines a rectangular tile on the image as the query image. Then, the user specifies the types of features to be used in the similarity search. The user can select from a large array of features, ranging from simple pixel statistics to complicated visual attributes such as shape and texture. The list of features used in this experiment include: simple feature, histogram, angular radial transform and binary angular radial transform. Simple feature which is a four dimensional vector consists of mean, the standard deviation, the maximum and minimum of pixel value. Histogram is a 16 bin of pixel values in a tile image. The bins are uniform across the dynamic range. Angular radial transform belongs to a broad class of shape analysis tools based on moments [5]. ART is based on the region-shape descriptor defined in MPEG-7. To provide a description of the shape of a 2-D object independent of the internal pixel values, here proposed a slight modification of ART called the Binary ART (BART) feature. Based on the user's familiarity of the system, the user may start with simple features to obtain a quick response, and then refine the results with more sophisticated features. On the other hand, the user may choose a particular combination of features to exactly pinpoint the characteristics of interest. Based on the types of features chosen by the user, the feature extraction [9] module populates the feature database with feature vectors extracted from images in the database. A simple sliding-window approach is used in generating feature vectors from images. A tile window, with dimensions same as the query image, is moved across each image in a fixed step-size. A feature vector, which contains all features selected by the user, is computed for the part of the image under the tile window at each location. In the experiments reported here, a small step-size of two pixels is used for both the horizontal and vertical directions in order to capture spatial variations of the data. This results in overlapping tiles. Other step-sizes are also possible. The feature vector, the location of the tile image, and the ID of the original image in the image database are stored in the feature database. Even though this paper is focus on the design of features for similarity search. With the feature database in place, the similarity search module seeks out the feature vectors in the database that are “similar" to the feature vector corresponding to the query image. To properly define the notion of similarity, we assume that there is a distance, or dissimilarity, function associated with each type of feature. Two feature vectors that are a small distance apart are regarded to be more similar to each other than those with a large distance between them. Some of the most commonly used distance functions are described in detail in [3] and [6]. Based on the distances between the query feature and the features in the feature database, the similarity search module supports two types of search functions: ε-search and k-Nearest-Neighbor (k-NN) search. In a ε-search, the module returns all feature vectors in the database whose distance from the query feature is within a positive threshold ε. ε -search is intended for experienced users who can correlate distance values with the level of similarity. In a k-NN search, the k feature vectors in the database closest to the query feature are returned. When more than one feature is used a similarity search, the results on individual features can be combined by conjunction and/or disjunction. Based on the returned results, the user can refine the search by modifying different search parameters and specifying different sets of features to be used. Except for expert users who are very familiar with the system, search refinement can be a daunting task due to the large number of parameters and feature available. A more intuitive approach, called user-relevance feedback is to have the user identified relevant and non-relevant entries among all the returned results, and applies machine-learning techniques to infer appropriate modifications in search parameters.

JCIT staff will edit and complete the final formatting of your paper.

2 Proposed System

In similarity based object retrieval system, the existing system segment database images that is same size of the query image for similarity search. Then the segmented images are directly matching with the query image. As a result, the results of this system are affected by a huge number of false positive images and many positive images are treating as false negative. Here more than one feature is used (such as simple feature, histogram, ART, BART) for similarity search, so the combination of the result on individual feature is complex. Due to the contrast and brightness variation images are affected lot. In this case this system cannot retrieve similar object. For these reasons we proposed a modularized approached for retrieval of similar object.

Our proposed system consists of two stage similarity comparison. Here system are divided into various parts such as image preprocessing, image database, segmentation, primary feature extraction, primary similarity comparison, modularization, module feature extraction, final similarity comparison. Preprocessing is an important task when working with image. In our system preprocessing is not required when working with high resolution images. But when working with different type of images such as contrast and brightness variation, low resolution images we need preprocessing for those images. In our proposed system the user first select an image from a huge collection of image from image database [4] and he crop any portion of this image depends on his interest .This crop section must be square shape of any length. Then feature are extracted from this crop section according to the features available of our system that is depicted in the next section and store it in the feature database. After that the user is selects the image database where the similarity search to be performed which is treated here as searched image. The searched images are then segment [2] with the same dimensions of the query image by moving a window over the images with a step size of one pixel both horizontal and vertical direction that is show in the figure 1. For each segmented image we are extract feature which is called here primary feature extraction and also store it in the feature database. In order to find out the segmented image we are keeping information of segment as well as ID of the original image in the feature database in a good form. Then we perform a similarity comparison between query feature vector and segmented image primary feature vector using K-nearest neighbor (K-NN) search technique [7]. K-NN search is based on a distance function that measures the similarity or dissimilarity between two instances.

Figure 1: Segment image by moving a window with a step size of one pixel

The standard Euclidean distance d(x, y) is often used as K-NN's distance function. The Euclidean distance in n-dimensional feature space, which is the usual distance between two points a= (a1, a2, a3...an) and b= (b1, b2, b3…bn), is defined by

$$d_e(a,b) = \sqrt{\sum_{i=1}^{n}(b_i - a_i)^2}$$

where n is the number of features. Here K closest match are return to the search result. If any segment passes the primary similarity search by using K-nearest neighbor, we consider those segments for final similarity comparison in the next step. Before final similarity comparison we modularized primarily matched segment as well as query image with the same dimensions. For modularization of both query image and primary matched segment here we use four modules for each segment which is depicted in the figure 2.

Figure 2(a): Segmented portion of an image

Figure 2(b): Modularized this segmented portion after passing primary similarity comparison for final similarity comparison. Here four modules are showing in the figure.

Then we extract feature both for query image module and primarily matched segment module and store it in the feature database. At this stage we need to store each module with its parent segment as well as original image. Now for each of the module of a query image, we try to find corresponding module in the segmented image. For this we apply similarity comparison between each module of a segmented image with each module of query image using K-nearest neighbor search technique. Two closest K values between segmented image module and query image module ensure that they are corresponding module for each other. In this way we find the corresponding modules for a query image in the segmented image. This is done to making the whole process rotation invariant. Now for each corresponding module we find that each part crosses a positive threshold .If the answer is positive we consider this segmented image as similar image. Otherwise discard this segment treated as false positive. The whole process are continues for other segmented image. Finally the results are represented using graphical user interface for the user.

3 Experiments

To evaluate the performance of our system, we perform experiments on different categories of image. We demonstrate the effectiveness of our approach with experiments using a set of categories from the Corel Image database. Since data independent similarity based object retrieval is a system where an image need to be contain similar objects in different position, so we also use personal image and collect image from internet that has similar object of interest. The object may be mushroom shape, stop sign, car, elephant, flower or any particular portion of an image. Different types of image format are use here such as jpg, tiff, bmp etc. An image may vary due to its transformations (such as translation, rotation, flip etc), low resolution, contrast & brightness variation, overlapping etc. So we consider those types of image to our experiments.

In our proposed modularize approached for similarity based object retrieval system we use two features. For high resolution images we use histogram feature but when the images are affected with low resolution, contrast and brightness variation etc then we need some preprocessing for that images. In these case histogram is not effective of those images .To retrieve of those type of images we use binary feature.

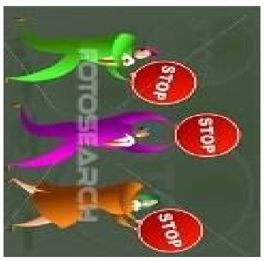
Histogram: The simplest non-parametric approach for density estimation is histogram calculation that has wide reputation for image retrieval. It is a graphic representation of the distribution of tones within an image. The horizontal axis represents each pixel value possible from black to white. The vertical values indicate the number of pixels in the image that occur at each gray level. Here we use 32-bin histogram.

Binary: For different contrast and brightness variation of images after preprocessing histogram is not suitable for similar object retrieval. In this case we convert those images into binary. A binary image is a digital image that has only two possible values for each pixel.

The following transformations and variations images are applied to our work. Examples of these transformations and variations together with the original image are shown in figure 3.

Diginal

Flip vertical



Rotated right 90^0

High brightness

Low contrast

Figure 3: Original image with the corresponding spatial transformations and variations

4 Performance Analysis Using Experimental Result

We have used five various types of images where objects may be mushroom shape, stop sign, flower, car, elephant or any particular portion of an image. We have got good result all the experiment using our proposed method. We have performed our search on various quality images. These image are includes same images, different images, rotate images, rotate but same images, different contrast images, all of the result shows in the table 1.

The performance of a similarity based object retrieval system measure in term of efficiency and effectiveness. Efficiency means time complexity; space complexity etc and effectiveness means how well a system give result. In our system we use effectiveness to evaluate our performance. To measure the effectiveness of a system recall-precision method are widely used [8]. Our performance is compares with the existing system using precision-recall graph in the figure 4 and figure 5.

Table 1: Performance Evaluation table using recall and precision:

Image size (Pixel)	**Image Description**	**Recall (%)**		**Precision (%)**	
		Cheung S.C *et al.[1]*	**Our method**	**Cheung S.C *et al.[1]***	**Our method**
361×320	Same	95	95	72	85
300×300	Different	84	90	66	79
381×360	Rotated	84	89	61	80
310×299	Rotated but same	85	92	67	83
400×400	Different contrast	0	58	0	63

Graphical Description of Precision and Recall:

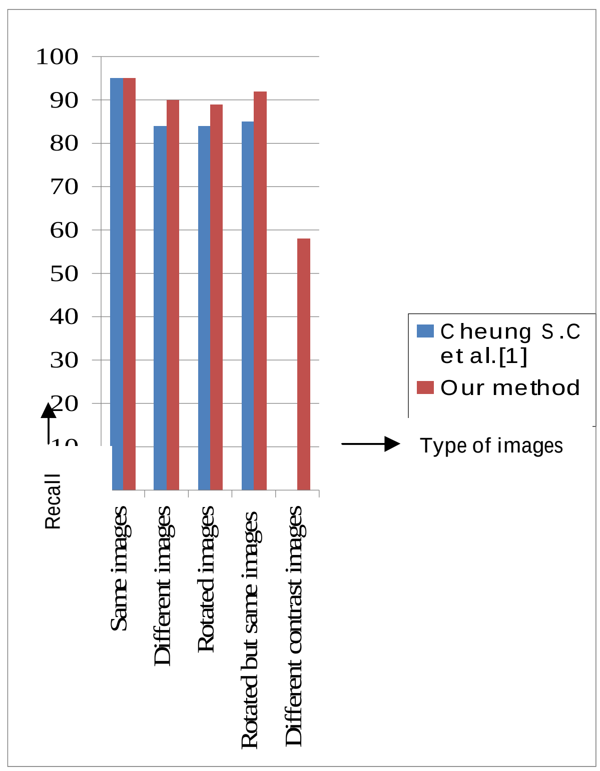


Figure 4: Graphical Description of Recall

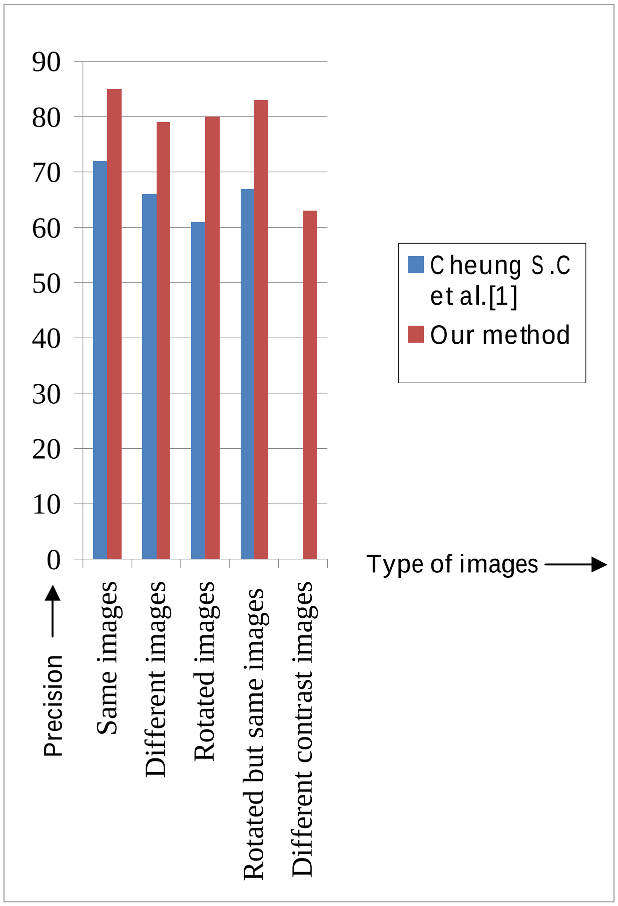


Figure 5: Graphical Description of Precision

After analyzing the experimental result we find that our proposed method effectiveness is good. The effectiveness of our result are justified by searching similarity comparison on different classes of images such as same images, different images, rotated images and rotated but same images, different contrast and brightness images. From the comparison table we find that our similarity search result on same classes images are same in case of false negative but in case of false positive its result vary significantly than existing system. On the other hand, for other classes (such as different images, rotated images, rotated but same images) our similarity search result shows better performance both case of false negative and false positive. But in different contrast and brightness images our proposed system is work greatly over existing method. Finally the results of our proposed system are compared with existing system by using precision and recall graph. From that graph we observe that both precision and recall graph shows better performance than existing method.

5 Conclusion and Future Work

Now a day's technology is improving a lot. Due to the availability of powerful computer system we are emphasizing on effectiveness of our system rather than time complexity and space complexity. Similarity based object retrieval system performance depends on variety of factors. Image variation such as transformation, contrast and brightness variation etc significantly affect on the result of similarity search. In spite of large variation of image our proposed system works effectively in many cases. It shows much better result to reduce the number of false positive which is treated as positive by the existing system. On the other hand, in case of false negative it shows slightly better result than existing system. When the contrast and brightness are varying greatly, existing system performance is very poor, but in that case our proposed method performs significantly.

Though our proposed system works effectively than existing system, but it has some limitations also. Since it is a two stage similarity compares, so it requires more searching time than existing system. After the primary similarity comparison the output of the first stage are store to make final similarity compares. Before second stage compares we need to modularize the query portion as well as database image with which query image is compared. So we need more calculation to store that module in database.

In future, we will apply image indexing using hash indexing structures to reduce two stage similarity comparison times among images and will try to apply more interactive image segmentation tools to reduce segmentation and modularization time. We also extend our task for similar object retrieval from 3D images.

References

[1] Cheung S. C. and Kamath, C., 2003, "Initial experiences with retrieving similar objects in simulation data," Proceedings, Sixth Workshop on Mining Scientific and Engineering Datasets, in conjunction with the Third SIAM conference on Data Mining, pp 11-18. UCRL-JC-151931.

[2] Derek Hoien, Rahul Sukthankar, Henry Schneiderman, Larry Huston, 2004, "Object Based Image Retrieval Using Statistical Structure of Image". Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213. Intel Research Pittsburgh, Pittsburgh, PA 15213.

[3] Castelli, V., 2002, "Multidimensional indexing structures for content-based retrieval," in Image Databases: Search and Retrieval of Digital Imagery, V. Castelli and L. D. Bergman, Eds. John Wiley & Sons, Inc.

[4] V. Castelli and D. Bergman, Eds, 2002, "Image Databases: Search and Retrieval of Digital Imagery", John Wiley & Sons, Inc.

[5] Mukundan, R. and Ramakrishnan, K. R., 1998, "Moment Functions In Image Analysis: Theory and Applications", World Scientific.

[6] Theodoridis, S. and Koutroumbas, K., 1999, "Pattern Recognition", Academic Press.

[7] Earl Gose, Richard Johnsonbaugh and Steve Jost , 1997, "Pattern Recognition and Image Analysis", Prentice-Hall.

[8] http://www.trec.nist.gov/pubs/trec10/appendices/measures.pdf

[9] F. Post et al., 2002, "Feature extraction and visualization of flow fields," in State-of-the-Art Proceedings of Eurographics" , pp.69-100.