Unsupervised CBIR by Combining Color, Shape (Features with a Threshold) and Lossless Gray Image Compression

Raj Kumar Mishra, Rajni Singh

DBGI, Dehradun, TMU, Moradabad; er.rajnisingh25@gmail.com

Abstract: Content-based image retrieval (CBIR) uses the visual features of an image such as color, shape and texture to represent and index the image. In a typical content based image retrieval system, a set of images sorted by similarities of their visual features with that of the query image are returned in response to a query. CLUE is a popular CBIR technique that retrieves images by clustering. In this paper, we propose a CBIR system that also retrieves images by clustering just like CLUE. But, the proposed system combines the color and shapes features with a threshold and lossless gray image compression for the purpose. The combination of the colored shape features and compression provides a robust feature set for image retrieval. We evaluated the performance of the proposed system using images from COREL database and compared its performance with that of the other two existing CBIR systems namely UFM and CLUE. Experimentally, we find that the proposed system outperforms the other two existing systems.
[Raj Kumar Mishra. Rajni Singh. Unsupervised CBIR by Combining Color, Shape (Features with a Threshold) and Lossless Gray Image Compression. Academia Arena, 2012;4(2):37-41] (ISSN 1553-992X). http://www.sciencepub.net.

Keywords: Content based image retrieval, image classification, unsupervised learning, spectral graph clustering, lossless gray image compression.

1 INTRODUCTION

The creation of the World Wide Web (in short, WWW) has enabled users to access data in a variant of media formats. This served as a stimulus for organizations having large image collections to convert their collections to digital formats. The number of digital images on the WWW is estimated to be more than hundred of millions. This creates a need for development of novel techniques for efficient storage and retrieval of images.

Content-based image retrieval (CBIR, in short) uses the visual contents of an image such as color, shape and texture to represent and index the image. In a typical content-based image retrieval system (see Figure 1), the visual contents of the images in the database are extracted and described by multi-dimensional feature vectors. The feature vectors of the images in the database form a feature database. To retrieve images, users provide the retrieval system with example images or sketched figures. The similarities/distances between the feature vectors of the query example or sketch and those of the images in the database are then calculated and retrieval is performed with the aid of an indexing scheme. The indexing scheme provides an efficient way to search the image database for images similar to the query images in order to return the relevant images. Generally speaking, content-based image retrieval (CBIR) aims at developing techniques that support effective searching and browsing of large image digital libraries on the basis of automatically derived image features [1].

Unsupervised learning is applied to the class of problems, where one seeks to determine how the data are organized. Here, the system discerns the objects under consideration in different categories on the basis of some similarity measures. The objects that are similar to each other are put in one group (also called a cluster) and the objects that are dissimilar are put into different clusters. CLUE, cluster-based retrieval of images by unsupervised learning, proposed by Chen et al.. [17, 18] is an example of CBIR technique based on unsupervised learning.

In this paper, we propose a CBIR system that is also based on unsupervised learning and combines the color and shape features with a threshold to compute the similarity of the query image with the images in the database.

This paper is organized as follows. In the next section, we discuss the background and related work. In Section 3, we discuss the details of unsupervised content based image retrieval and present the architecture of our proposed CBIR system. In section 4, we present our experimental results. Finally, we conclude in section 5.

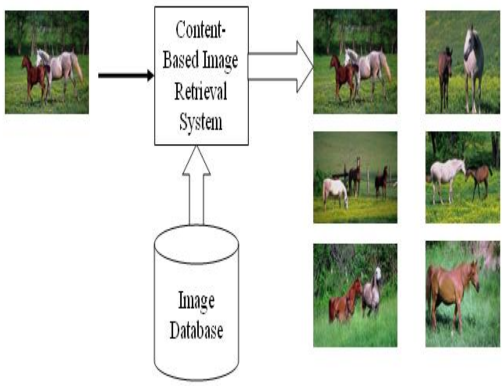


Figure 1. A Content Based Image Retrieval System

2 BACKGROUNDS AND RELATED WORK

In the past fifteen years, many general-purpose image retrieval systems have been developed. Some of these are QBIC System [8], Photobook System [9], Blobworld System [10], Virage System [11], VisualSEEK and WebSEEK Systems [12], the PicHunter System [13], NeTra System [14], MARS System [15], and SIMPLIcity Systems [16].

Existing CBIR systems can be grouped into two major categories: *full-image* retrieval system and *region-based* image retrieval system. Some of the existing CBIR systems may also belong to the both categories. Most of the existing CBIR systems are region-based systems because region-based systems are better than full-image retrieval systems.

In a CBIR system, to search images by their content, two things have to be done [23].

1. The image has to be re-encoded into some mathematical form and stored in a database.
2. There should be a mechanism to compare these mathematical forms.

Re-encoding is needed because an image is a collection of pixels with no meaning by itself. There is a gap between the visual information conveyed by the image and the way it is encoded. The process of re-encoding the image into a mathematical form suitable for comparison purpose is called feature extraction.

Features can also be grouped as low-level and high-level features. Low-level features are features that can be obtained from the pixel itself. Examples are color and texture. High-level features are features obtained from the combination of low-level features. Examples are edge and shape. But, the three of the most widely used features are (i) color (ii) texture and (iii) shape. Details of these features are discussed in [26].

A typical CBIR system views the query image and images in the database (target images) as a collection of features, and ranks the relevance between the query image and any target images in proportion to feature similarities. Images with high feature similarities to the query image may be very different from the query in terms of the interpretation made by a user. This is referred to as the *semantic gap*, which reflects the discrepancy between the relatively limited descriptive power of low level imagery features and the richness of user semantics [17]. Statistical classification methods group images into semantically meaningful categories using low level visual features so that semantically-adaptive searching methods applicable to each category can be applied [19, 20, 16, 21]. The Simplicity system [16] classifies images into graph, textured photograph, or non-textured photograph, and thus narrows down the searching space in a database. There has been work on attaching words to images by associating the regions of an image with object names based on region-term co-occurrence [22]. And semantically precise image segmentation by an algorithm is still an open problem in computer vision [23, 24].

Cluster based retrieval of images by unsupervised learning (CLUE) is an important CBIR technique based on unsupervised learning. CLUE retrieves image clusters by applying a graph–theoretic clustering algorithm to a collection of images in the vicinity of the query. Clustering in CLUE is dynamic.

In this paper, we propose a CBIR system that is also based on unsupervised learning. Color features are computed by color moment and color histogram [2, 3]. Shape features are calculated after images have been segmented into regions or objects [4, 5]. Shape information is captured in terms of edge images computed using Gradient Vector Flow fields [6]. Invariant moments are then used to record the shape features [7]. The proposed system sums up the values of color and shape features, after applying the threshold, for assigning weights to different images. On the basis of these weights, the relevant images are extracted from the image database.

3. UNSUPERVISED CONTENT BASED IMAGE RETRIEVAL

A CBIR system based on CLUE is shown in Figure 2. In this, the retrieval process starts with feature extraction. The features for target images (images in the database) are usually computed beforehand and stored as feature files. Using these features together with an image similarity measure, the resemblance between the query image and target images are evaluated and sorted. Next, a collection of target images that are "close" to the query image are selected as the neighborhood of the query image. A clustering algorithm is then applied on these target images. Finally, the system displays the image clusters and adjusts the model of similarity measure.

The major difference between CBIR system based on CLUE and the other two CBIR systems lies in the two processing steps, selecting neighboring target images and image clustering, which are the major components of CLUE[18].

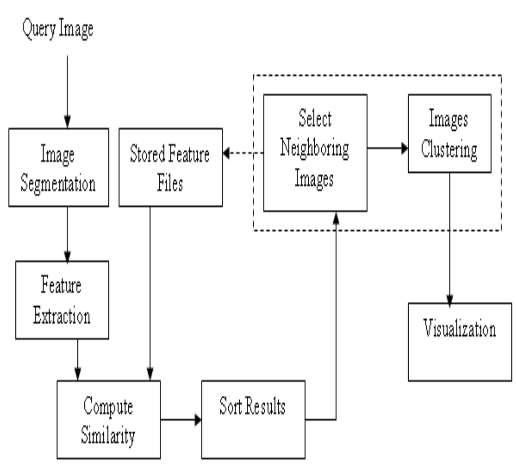


Figure 2. A CBIR system based on CLUE

There are two simple methods to select a collection of neighboring target images for query image [16].

- *Fixed-radius method* (FRM) takes all target images within some fixed radius ε with respect to *i*. For a given query image, the number of neighboring target images is determined by ε.
- *Nearest-neighboring method* (NNM) first chooses k NN of *i* as seeds. The *r* NN for each seed is then found. Finally, the neighboring target images are selected to be all the distinct *target* images among seeds and their *r* NN, i.e., distinct images in $k(r+1)$ target images. Thus, the number of neighboring target images is bounded above by $k(r+1)$.

In the field of computer vision, two types of representations are widely used. One is called the *geometric representation,* in which data items are mapped to some real normed color space. The other is referred to the *graph representation* emphasizing the pair wise relationship. Graph representation of neighboring target images is as follows.

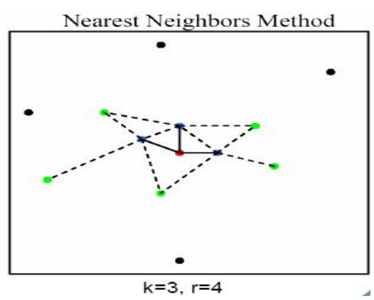


Figure 3 Example of Nearest Neighbor Selection of Images

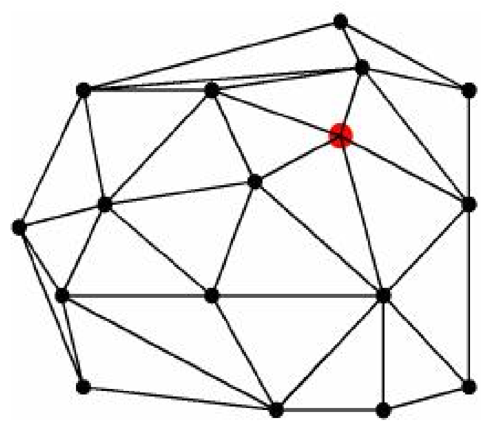


Figure 4 Example of Weighted Graph Representation of Images

A set of *n* images is represented by a weighted undirected graph G = (V, E).The nodes V = {1, 2, …., *n*} represent images, the edge E = {(*i*, *j*) : *i*, *j* ∈ V} are formed between every pair of nodes, and the nonnegative weight w_{ij} of an edge (*i, j*), indicating the similarity between two nodes, si a function of the distance(or similarity) between nodes(images) i and j. Given distance d(i, j) between images i and j, the nonnegative weight w_{ij} is given by

$$w_{ij} = e^{\frac{-d(i,j)^2}{s^2}}$$

where, s is a scaling parameter that needs to be tuned to get a suitable locality. The choice of exponential decay is based on support from psychological studies. The weight can be organized into a matrix W, named affinity *matrix* with *ijth* entry given by w_{ij}

Under a graph representation, clustering can be naturally formulated as a graph partitioning problem. The CLUE uses spectral graph partitioning methods called the normalized cut (N_{cut}) method for image clustering. A graph partitioning method attempts to organize nodes into groups so that the within-group similarity is high, and/or the between-groups similarity is low.

Given a graph G = (V, E) with affinity matrix W, a simple way to quantify the cost for partitioning nodes into two disjoint sets A and B ($A \cap B = \Phi$ and $A \bigcup B = V$) is the total weights of the edges that connecting the two sets. In graph theory, this cost is called a *cut*

$$cut(A, B) = \sum_{i \in A, j \in B} w_{ij}$$

which can also be viewed as a measure of the between-groups similarity.

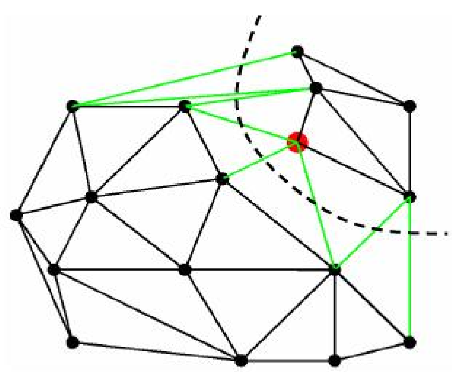


Figure 5 Normalized cut of weighted graph of Images

Finding a bipartition of the graph that minimizes this cut value is known as the *minimum* cut problem. However, the minimum cut criterion favors grouping small sets of isolated nodes in the graph because the cut defined above, does not contain any within-group information. This motivates several modified graph partition criteria including the N_{cut}

$$N_{cut} = \frac{cut(A,B)}{cut(A,V)} + \frac{cut(A,B)}{cut(B,V)}$$

An unbalanced cut would generate a large N_{cut} value.

Finding a bipartition with minimum N_{cut} value is an NP-complete problem. Shi and Malik [27] proposed an approximated solution by solving a generalized eigenvalue problem

$$(D - W)y = \lambda D y$$

Where W is an n X n affinity matrix, D = diag[s_1, s_2,…., s_n] is a diagonal matrix with $s_i = \sum_{j=1,\ldots,n} w_{ij}$.

Given a graph representation of images G = (V, E) with affinity matrix W, let the collection of image clusters be {C_1, C_2, …, C_m}, which is also partition of V, i.e.,

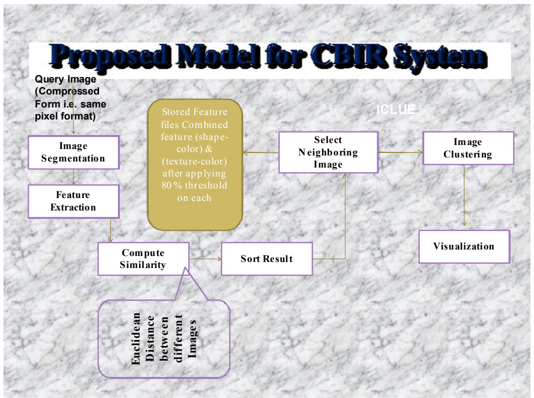
$$C_i \cap C_j = \Phi \text{ for } i \neq j \text{ and } \bigcup_{i=1}^{m} C_i = V.$$

Then the representative node (image) of C_i is

$$\arg\max \sum_{j \in C_i, t \in C_i} w_{jt}$$

which can also be viewed as a measure of the between-groups similarity.

Now, we propose the architecture of a CBIR system based on unsupervised learning as shown in Figure 6. The major difference between the proposed CBIR system and the CBIR system based on CLUE lies in the stored features files. In the proposed CBIR system, we store the values of features in the stored features files after combining values of shape and color features of an image with the 80% & 20% (for sketch diagram) threshold. In other words, we take 80% & 20% (for sketch diagram) of the total value of color features and 80 % of the value of shape features for an image and combine the two values and store that combined value into the stored features files as the feature values for the image.



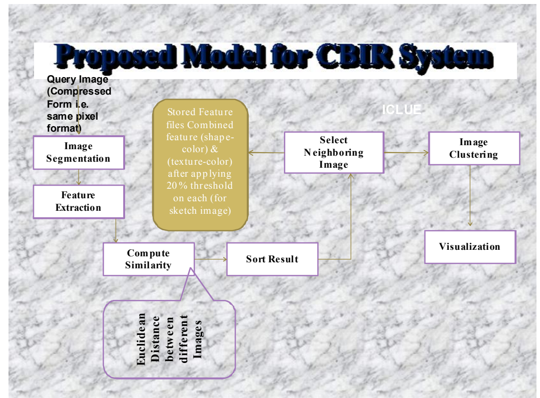


Figure 6 (a) & 6 (b) The Proposed CBIR system

12/12/2011