J. Vis. Commun. Image R. 36 (2016) 40–55

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Indexing and encoding based image feature representation with bin overlapped similarity measure for CBIR applications ☆

S.G. Shaila, A. Vadivel *

Information Retrieval Group, Department of Computer Applications, National Institute of Technology, Tiruchirappalli 620015, India

ARTICLE INFO

Article history:
Received 16 March 2015
Accepted 6 January 2016
Available online 11 January 2016

Keywords:
Indexing
Encoding
Distance measure
Image retrieval
Color features
GR Coding
BOSM
Histogram dimension

ABSTRACT

In Content Based Image Retrieval (CBIR) system, the exhaustive search for a given query image to find the relevant images in the database are non-scalable. In this paper, we propose indexing, coding technique and similarity measure to address the above mentioned problem. We consider the color histogram of the image and its bin values are analyzed to understand the color information in the image. The histogram dimension is reduced by removing trivial bins and only those bins that represent color information significantly are considered. Based on the dimensions of the histogram, it is clustered and indexed. The Golomb–Rice (GR) coding is used to encode the indexed histograms. The Bin Overlapped Similarity Measure (*BOSM*) is proposed to compute the distance values between query and database image histograms. The performance of proposed approach is evaluated on benchmark datasets and found that the performance of the proposed approach is encouraging.

1. Introduction

Enormous growth in digital technology and higher storage capacity with low cost has resulted in the development of huge digital media devices. These devices deploy new applications in the area of multimedia information systems [1], spatial information systems [2], medical imaging [3], time-series analysis [4], image retrieval systems [5], storage and compression [6,7] etc. The explosion of these inexpensive digital equipment and storage devices made users to easily own and access huge amount of digital images. Also, the rapid growth of the Internet has encouraged the real time processing and the retrieval of digital images as one of the most important communication media for daily life. As a result, the visual information retrieval has become an active research area. The Text-Based Image Retrieval (TBIR) is a traditional approach employed by the user is sensitive to the keywords [8] and this limitation is handled by CBIR approach, which is proposed in the early 1990s [9]. The CBIR has attracted wide interest from the researchers in the areas of computer research and proved its dominating tendency in the image retrieval systems. The CBIR uses the visual features such as color, texture and shape for image representation and these features are indexed and stored in the feature database for retrieval applications. The users are interested in searching similar images for a given query image and hence, the visual features of query image are compared with the visual features of all the images features in the database. Among various features, the color is considered as the most important feature and the reason is that the human visual system plays an important role in understanding the color theory [10]. Most of the times, image have less information content and however includes high-level of visual complexity. This is due to the fact that the image features are high dimensional in nature and represented as n-dimensional feature vector such as color histograms [11] and color layout [12].

Due to the exponential growth of World Wide Web (WWW), the size of the image data is rapidly increasing, which results in storage complexity, access time and transmitting issues. This has further reduces the performance of any real-time system. During retrieval, though, the time for comparing two image features is less, the cumulative comparison time is relatively large and increases the user's waiting time considerably. The search time increases linearly with the size of the database and thus, a suitable indexing technique is required to index large feature databases to ensure that the search time is not increasing linearly with the feature database size. Moreover, indexing plays a significant role in the area of computer vision such as remote sensing [13], image analysis [14,15], pattern recognition [16,17] and is a challenging task for CBIR to minimize search time with higher precision of retrieval. In addition, suitable feature representation scheme is also required to improve the search process [18]. Usually, research focus on large databases and image analysis in the applications

☆ This paper has been recommended for acceptance by M.T. Sun.

* Corresponding author. Fax: +91 431 250 0133.
E-mail address: vadi@nitt.edu (A. Vadivel).

http://dx.doi.org/10.1016/j.jvcir.2016.01.003
1047-3203/

of facial recognition system, image and video compression and scientific data involved applications such as content based medical and biological image indexing and retrieval. Many images are currently stored in unstructured, non-indexed form and has become hard to analyze due to increased size of their databases. These images are compared using their features in the computational analysis, which involves clustering, building cluster representatives, similarity search, data visualization, etc. This leads to the similarity search using certain features such as color, texture size, and depth, depending on the domain-specific observations of the images. In various applications, this feature size is large and cause curse of dimensionality issues, which degrades the performance of the retrieval system. As a result, the dimensions of the features has to be reduced without losing the original meaning of the features. This can be achieved by removing the irrelevant and redundant information [19] and by using a suitable encoding scheme. Most of the current research deals with per-picture based image encoding algorithms. However, the entire image database has similarities in terms of objects, persons, locations, etc. It is desirable to encode a group of images to gain effective bit ratio by exploiting inter-image redundancies. In addition, there is a need for a useful nearest neighbor search and a suitable similarity measure that uses less retrieval time and maintains good precision of retrieval. Since, similarity measure is considered as computationally expensive, the quick performance of the similarity measure algorithms becomes a key issue. Due to high dimensional features, most of the nearest neighbor search algorithm that is found to be not exact and do not have acceptable performance. Many real time applications are forced to use an approximate search to match the user's given query. As a result, building an efficient algorithm that performs fast nearest neighbor search in large database is a challenging issue in CBIR. This is necessary to fasten up applications such as machine learning [20], document retrieval [21], data compression [22], bioinformatics [23,24] and data analysis [25].

Based on the above discussion, this paper addresses the curse of dimensionality issues in terms of indexing, feature storage and retrieving relevant information. A new approach for indexing and encoding and a distance measure is proposed. The indexing scheme clusters the images based on their feature dimension size to avoid linear search process. Next, we propose an encoding scheme in which Golomb–Rice coding (GR) is applied to code the bin values of the histogram for achieving low bit length. The proposed approach use the quotient part of the code for achieving low bit length with good precision of retrieval. Finally, *BOSM* is proposed as similarity measure to handle the variation in the bin dimensions between query and database image histograms. The proposed distance measure considers the values of common bin indices to compute the distance between query and database images to decrease the disk computation rate and retrieval time. The rest of the paper is organized as follows. In Section 2, we review the related work and Section 3 explains the proposed approach in detail. Section 4 presents the experimental results and we conclude the paper in the last Section.

2. Materials and methods

The related work is divided into three sub-sections to present the reviews on indexing the data and space partitioning along with reviews on reducing the dimension of the features and reviews on distance measure. The related work initially starts on various indexing approaches, continued on discussing encoding approaches and ends with reviews on various distance measures to establish the research gaps.

2.1. Reviews on indexing the data and space partitioning

Index structure is an effective tool that prunes the feature space and avoids unnecessary computations. Nowadays, the CBIR with Graphical Processor Units (GPU) are used in High Performance Computing (HPC) system because of its high parallel structure, multithread execution of algorithms, programmability and low cost. Many researchers have produced promising results on parallelizing sequential algorithm on GPU. The direct parallelization of traditional indexing structure cannot make complete use of GPU resources. In general, these kind of indexing approaches are divided into two main categories namely tree-based and hash-based indexing structure. Tree-based structures are used in-memory indexing and they are categorized as balanced and unbalanced tree. The binary tree [26] is an unbalanced tree where in a node has at most two children and this structure may degenerate, causing more nodes to be accessed. In addition, especially for in-memory indexing, balanced tree-structures such as B-trees [27], B+-trees [28], red-blacktrees [29], AVL-trees [26] and T-trees [30] are being widely used. Current research focuses on cache conscious tree-structures say T-Trees [31] as well as B-Trees [32] and both of them exploits SIMD instructions [33] or architecture-aware tree indexing on CPUs and GPUs [34]. The major disadvantages of the tree-based structure is that reorganization of nodes and key comparisons. In addition, there is a need for variable prefix lengths to adjust the tree height and its memory consumption for indexing arbitrary data types of fixed or variable length. Tree-based structures require re-organization for tree balancing by rotation or node splitting/merging and many key comparisons. All these procedure degrades the performance of the tree-based techniques in terms of large running time, large space requirements and adverse properties of retrieval system for higher dimensions. This is due to the fact that higher feature dimensions are required to describe the complex visual content of image in CBIR. Hence, the retrieval performance may be degraded whilef tree-based indexing structure is used [35]. Moreover, tree-based index structure involves interleaved series of similarity computations and causes ineffectiveness for parallelization [36]. In fact, tree-based structures perform well for low-dimensional feature space. Although, hash-based index structures, such as Local Sensitive Hashing (LSH) [37] and Spectral Hashing (SH) [38] have been proposed to handle the issues, however, they are not suitable for the generic distance types and thus the application of these algorithms is largely limited. In hash-based structures also, there are limitations in terms of reorganization of nodes and key comparisons. It involves many parameters for its construction and needs expert to adjust the parameters. Hash-based techniques heavily rely on assumption parameters about the data distribution of keys and require reorganization (rehashing) as well. Hash-based structures rely on a hash function to determine the slot of a key within the hash table (an array). Depending on the used hash-function, the approaches make assumptions about the data distribution of keys and can be order-preserving. For chained bucket hashing [26], no reorganization is required because the size of the hash table is fixed. However, in the worst case, it degenerates to a linked list and thus, the worst-case search time complexity is not good. In contrast to this, there are several techniques that rely on dynamic reorganization such as extendible hashing [39], linear hashing [40] and modified linear hashing [30]. Current research focuses on efficiently probing multiple hash buckets using SIMD instructions [41]. Due to reorganization, those structures exhibit a higher search time complexity. However, additional overhead for dynamic hash table extension and re-hashing is required and thus it can be slower than tree-based structures. The authors of [42,43] have proposed an effective parallelization of brute force retrieval using Compute Unified Device Architecture (CUDA) and CUDA Basic Linear Algebra