Ad Hoc Networks 35 (2015) 137–148

Contents lists available at ScienceDirect

Ad Hoc Networks

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

Distributed in-memory vocabulary tree for real-time retrieval of big data images

Hancong Duan[a], Yubing Peng[a,*], Geyong Min[b], Xiaoke Xiang[a], Wenhan Zhan[a], Hao Zou[a]

[a] *University of Electronic Science and Technology of China, Department of School of Computer Science and Engineering, Qingshuihe Campus: No. 2006, Xiyuan Ave, Chengdu 611731, PR China*
[b] *College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter EX4 4QF, United Kingdom*

ARTICLE INFO

Article history:
Received 5 February 2015
Revised 7 April 2015
Accepted 1 May 2015
Available online 19 May 2015

Keywords:
Image retrieval
Distributed systems
Vocabulary tree
MapReduce
Big data

ABSTRACT

Real-time precision retrieval based on big data images has become a key technical issue recently. The vocabulary tree is an efficient method for addressing this issue owing to high precision and fast retrieval time. Most of the existing construction methods for the vocabulary tree are centralized. However, under a centralized scheme, it is almost impossible to train a big vocabulary tree with limited memory to retrieve a similar image with high precision. In this paper, a new scheme of the distributed in-memory vocabulary tree based on MapReduce model for massive image training and retrieval is proposed. Firstly, the distributed image feature exaction mechanism is presented to preprocess massive images. Secondly, a distributed K-means algorithm based on MapReduce model is proposed to build the first level of the vocabulary tree concurrently. Thirdly, the big vocabulary tree is divided into many subtrees. The entire training task for computing the vocabulary tree is divided into many subtasks. These training subtasks are performed in parallel in the memory of the cluster nodes. This distributed vocabulary tree strategy can support massive image training in memory. Therefore, a similar image can be retrieved in a distributed manner based on MapReduce model. Besides, the training time and memory overhead of our proposed scheme are analyzed in detail. The experimental results demonstrate that, with an increase in computer nodes, the training time and memory overhead on each node are linearly reduced, and the retrieval time is relatively reduced compared with centralized scheme without a loss of retrieval precision.

1. Introduction

With the rapid development of the Internet, the number of images available is growing explosively. Both the task of quickly and accurately retrieving similar images from a massive image library and the task of completing a mass image training using limited memory and time have both become research hotspots.

Content-Based Image Retrieval (CBIR) [1], which constructs indices by the Scale Invariant Feature Transform (SIFT) [2] or Speeded Up Robust Features (SURF) [3] extracted from images, can mainly be classified into three categories of indices: tree-based index, hash-based index, and visual-word-based inverted index [4–7]. In a tree-based indexing structure, such as KD-tree [8] or R-tree [9], when the dimension of a feature descriptor grows greater than 20, the construction efficiency deteriorates rapidly. Both categories of hash-based index – the Euclidian Locality Sensitive Hashing (E2LSH) [10] related methods and the spectral hash methods [11–13] – are inefficient on sparse descriptors. In a visual-word-based inverted index, the traditional flat K-means [14–17] clustering algorithm and the improved Approximate K-Means

[*] Corresponding author.
E-mail address: yubingpenguestc@163.com (Y. Peng).

http://dx.doi.org/10.1016/j.adhoc.2015.05.006
1570-8705/© 2015 Published by Elsevier B.V.

(AKM) [5], Robust Approximated K-Means (RAKM) [18] and Approximate Gaussian Mixture (AGM) [19] lead to heavy training and retrieval time overhead. Additionally, in the index method, which is based on a vocabulary tree [20–22], the scale (the number of branches and levels) of a vocabulary tree is needed to ensure the image matching precision. While the image training set is huge (say millions or billions), it is almost impossible to train such a huge vocabulary tree in centralized memory. Moreover, the centralized training time of a huge vocabulary tree is too long.

In this paper, in order to reduce the memory and time overhead for centralized training of a huge vocabulary tree so that it can train a huge image training set, a new distributed vocabulary tree image training and retrieval scheme is proposed.

The major contributions of this paper are:

1. We propose a distributed in-memory vocabulary tree based on MapReduce model for big data images, which is composed of two parts: the distributed K-means algorithm and the distributed multiple subtrees method.
2. We propose a distributed retrieval approach which divides the retrieval task into many subtasks, and these subtasks are executed in multiple nodes concurrently.
3. We present the mathematical model of our method and analyze the training time and the memory overhead by comparing our scheme with the legacy centralized proposal. We found that the training time and the memory overhead of our scheme are all reduced to about $1/n$ of legacy one on single compute node, where n is the number of computer nodes.

The rest of the paper is organized as follows: Section 2 reviews the research status and existing problems of the legacy CBIR. Section 3 presents the useful preliminaries. Section 4 introduces our method, including the overall framework of the system, training process, retrieval process and discusses the advantages compared with the centralized single node. Section 5 presents the experimental analysis of the scheme proposed in this paper by comparing the training time, training memory overhead, retrieval precision, and retrieval time under various working conditions. Section 6 summarizes this paper.

2. Related work and motivation

2.1. Related work

In the 1990s, the CBIR method was proposed to overcome the difficulties encountered with text-based image retrieval. There are three methods of CBIR – tree-based index, hash-based index, and visual-word-based on inverted index.

Tree-based index was proposed by Silpa-Anan and Hartley [23]. They created multiple KD-trees along with a same dataset to improve the performance. Although the tree-based indexing structure can successfully divide data space into a hierarchical tree structure, when the dimension of the descriptor exceeds 20, the efficiency of indexing decreases dramatically, suffering from “the curse of dimensionality”. Because a KD-tree usually takes a lot of time to backtrack a tree to get optimal solution in high dimensional space.

Besides tree-based index, there is hash-based index. In 2004, Ke et al. [24] successfully adopted an E2LSH index feature descriptor. E2LSH is efficient with high dimension and dense feature points, but the LSH-like scheme has many shortcomings: firstly, large numbers of hash tables are needed to get high retrieval precision. Secondly, the filtration effect is poor when the point is far from the query point. Thirdly, it does not make full use of the properties of the hash function.

The third type of CBIR is visual-word-based on inverted index. It contains two different methods – flat vocabulary tree and hierarchical vocabulary tree. Flat vocabulary tree is proposed by Sivic and Zisserman [6] in 2003. They applied a standard flat K-means algorithm to train a vocabulary tree. Although using a vocabulary tree can get good retrieval precision when the scale (the number of branches and levels) of the vocabulary tree is big enough, flat K-means needs a large number of cluster centers, which leads to low time efficiency of clustering. Philbin et al. [5] proposed the AKM algorithm. In the algorithm, at the beginning of each iteration, eight random KD-trees are used to build the cluster centers. Gu and Zhu [25] presented the FAKM algorithm based on AKM, using the idea of classifying the cluster centers. While in the clusters results of AKM, there are some cluster centers which have only a few samples. FAKM discards these centers to reduce the quantity and category of the samples to be clustered.

Hierarchical vocabulary tree is presented by Nister and Stewenius [7]. They used a hierarchical K-means cluster algorithm to build a vocabulary tree, which can efficiently cluster a large number of visual words. Chen and Sheng [20] overcame the shortcoming of low recall rates of retrieval results by using a fuzzy quantization method. By means of fuzzily quantizing the SIFT features exacted from images into words, images are converted into vectors, which can be compared with each other to test the similarity of images. Huo [26] presented a scheme to reduce the calculation of clustering by first reducing the dimension of the exacted SIFT features.

To the best of our knowledge, the existing image training and retrieval systems are often deployed on a single server (centralized). But when the number of images available grows explosively, a single server is almost impossible to train a massive image set (say millions or billions), because the feature points and the inverted index are too big to load in memory. So how to train and search a huge image set in parallel is very necessary and important. King et al. [27] proposed a Firework Query Model for distributed retrieval using a P2P network. But P2P needs a lot of time to route to the right node. He and Lin [28] proposed a distributed parallel method which is based on LSH using a Hadoop distributed system. They used a Hadoop distributed system which will write middle results into disks, which is much slower than our in-memory scheme. In our scheme, all of the retrieval steps are processed in memory. Meanwhile [27,28] are based on LSH [29], but