ABHIYANTRIKI
An International Journal of Engineering & Technology
(A Peer Reviewed & Indexed Journal)
Vol. 4, No. 5 (May, 2017) http://www.aijet.in/ eISSN: 2394-627X

Time Efficient Image Processing Framework using MapReduce

Sanchari Saha
Assistant Professor
Deptt. of Computer Science Engineering,
MVJCE, Bangalore, India

Sanketh T*
M. Tech.
Deptt. of Computer Science Engineering,
MVJCE, Bangalore, India

Abstract

Paper introduces an effective processing framework nominated Image Cloud Processing (ICP) to powerfully cope with the data explosion in image processing field. While most previous researches focus on optimizing the image processing algorithms to gain higher efficiency, our work dedicates to providing a general framework for those image processing algorithms, which can be implemented in parallel so as to achieve a boost in time efficiency without compromising the results performance along with the increasing image scale to accomplish processing, two novel data representations named P-Image and Big-Image are designed to cooperate with MapReduce to achieve more optimized configuration and higher efficiency. It is implemented through a parallel processing procedure working with the traditional processing mechanism of the distributed system. Representative results of comprehensive experiments on the challenging ImageNet dataset are selected to validate the capacity of our proposed ICP framework over the traditional state-of-the-art methods, both in time efficiency and quality of results.

Keywords*: Image Processing, MapReduce.*

****Author for correspondence*** sankusgr@gmail.com

1. Introduction

Over several years, image processing has gained wide attention due to its comprehensive applications in various areas, such as engineering, industrial manufacturing, military, and health, etc. However, in spite of its expansive development prospect, huge data amount comes along and hence triggers severe constraints on data storage and processing efficiency, which calls for urgent solution to relieve such limitations. Particularly, since Web age and search. Furthermore, the prosperity of big image data over recent years has undoubtedly aggravated the challenge that current image processing field commonly faces. To this end, arduous efforts from related research fields have been made so far to propose high-efficiency image processing algorithms.

It is therefore significant to take full advantages of the abundant computing resources that the distributed system offers in a cloud processing manner. Given the distributed resources, parallel processing can undoubtedly achieve state-of-the-art improvements when compared with the traditional processing methods limited to a single machine, yet meanwhile, this attempt is a

demanding challenge. In recent years, considering the high efficiency that parallel processing brings, researchers have attempted to propose image processing algorithms that can be implemented in parallel, among which image classification , feature extraction and matching can serve as representative instances. All those algorithms can run on multiple nodes in parallel and hence significantly improves the time efficiency.

2. Review of Literature

Liu [2] proposed a scalable two-phase top-down specialization approach for data anonymization using MapReduce on cloud. Releasing person-specific data in its most specific state poses a threat to individual privacy. Paper presents a practical and productive algorithm for determining an abstract version of data that masks sensitive information and remains useful for standardizing organization. The classification of data is implemented by specializing or detailing the level of information in a top-down manner until a minimum privacy requirement is compromised. This top-down specialization is practical and efficient for handling both definitive and continuous attributes. Method exploits the scenario that data usually contains redundant structures for classification. While generalization may remove few structures, other structures emerge to help. Results show that standard of classification can be preserved even for highly prohibitive privacy requirements. Work has great applications to both public and private sectors that share information for mutual advantage and productivity. Xu [3] proposed multi-view intact space learning, which integrates the encoded complementary information in multiple views to discover a latent intact representation of the data. Even though each view on its own is insufficient, we show theoretically that by combing multiple views we can obtain abundant information for latent intact space learning. Employing the Cauchy loss (a technique used in statistical learning) as the error measurement strengthens robustness to outliers. Proposes a new definition of multi-view stability and then derive the generalization error bound based on multi-view stability and Rademacher complexity, and show that the complementarities between multiple views is beneficial for the stability and generalization. MISL is efficiently optimized using a novel Iteratively Reweight Residuals (IRR) technique, whose convergence is theoretically analyzed. Experiments on synthetic data and real-world datasets demonstrate that MISL is an effective and promising algorithm for practical applications. Yang [4] proposed robust discrete spectral hashing for large-scale image semantic indexing. The ever-increasing image data has posed significant challenge on modern image retrieval. It is of great importance to index images with semantic keywords efficiently and effectively, especially confronted with fast-evolving property of the Web. Learning-based hashing has shown its power in handling large-scale high-dimensional applications, such as image retrieval. Existing solutions normally separate the process of learning binary codes and hash functions into two independent stages to bypass challenge of the discrete constraints on binary codes. This work proposed a novel unsupervised hashing approach, namely robust discrete hashing (RDSH), to facilitate large-scale semantic indexing of image data. Specifically, RDSH simultaneously learns discrete binary codes as well as robust hash functions within a unified model. In order to suppress the influence of unreliable binary codes and learn robust hash functions and also integrated a flexible loss with nonlinear kernel embedding to adapt to different noise levels and finally devised an alternating algorithm to efficiently optimize RDSH model. Given a test image, conducted r-nearest-neighbor search based on Hamming distance of binary codes, and then propagated semantic keywords of neighbors to the test image. Extensive experiments have been conducted on various real-world image datasets to show its superiority to the state-of-the-arts in large-scale semantic indexing.

3. Implementation

MapReduce is recognized as a popular framework to handle huge data amount in the cloud environment due to its excellent scalability and fault tolerance. Application programs based on MapReduce can work on a huge cluster of thousands of desktops and reliably process Peta-Bytes data in parallel. Owing to this, time efficiency successfully gains a desirable improvement. Until now, MapReduce has been widely applied into numerous applications including data anonymization, text tokenization, indexing and searching, data mining, machine learning, etc. Aimed at achieving higher time efficiency on the associated applications, recent endeavors involve many industry giants to make efforts by leveraging MapReduce. For example, Yahoo has been working on a couple of real-time analytic projects, including S4 and MapReduce Online. In addition, IBM has been devoted to developing real-time products such as InfoSphere Streams and Jonass Entity Analytics software used to analyze stream data more accurately. Despite that these frameworks have successfully implemented the efficient processing of text data and stream data; they do little contribution to image processing field. Motivated by these cases achievements and restrictions, we aim to implement an effective processing framework for big image data by the utilization of the cloud computing ability that MapReduce provides.

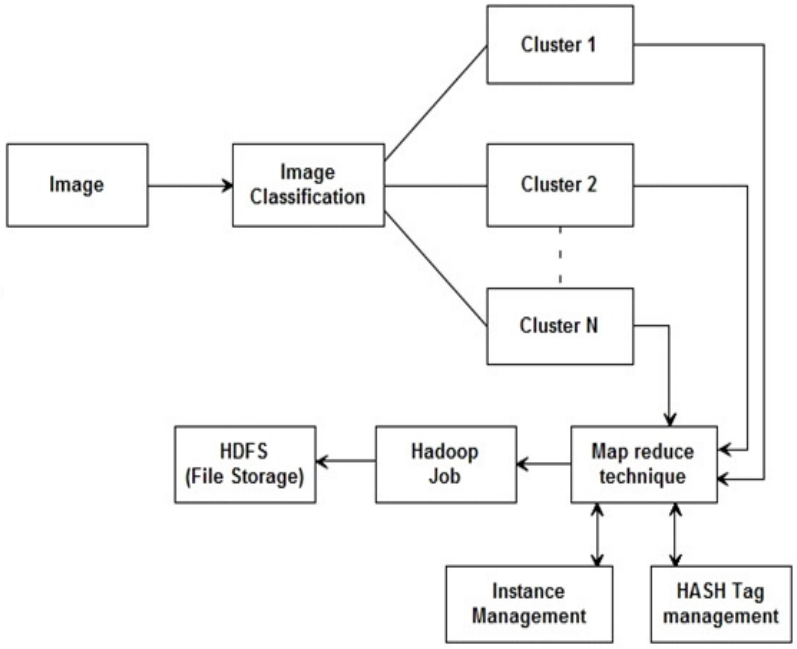

Fig. 1: System architecture

In this proposed system, we present and analyze a novel effective distributed framework named Image Cloud Processing (ICP) which is dedicated to offering a reliable and efficient model for vision tasks. The core design of ICP is to utilize the affluent computing resources provided by the distributed system so as to implement effective parallel processing. The elegant distributed processing mechanism that ICP contains is defined from two comprehensive perspectives:

a) Efficiently processing those static big image data already stored in the distributed system, such as the task of image classification, image retrieval, etc. that do not demand immediate response to the users but an efficient processing instead;
b) Timely processing that dynamic input which needs to be processed immediately and return an immediate response to the users, especially for the requests from the user terminal, e.g., the image processing software in the users' laptop / desktop.

System Architecture

System Architecture shows how Image Processing concept is combined with Hadoop Architecture. Images are classified into different classifications based on their catagories. Once the image is uploaded, it will be compared with clusters and stored in HDFS.

We use RANSAC algorithm [5] for comparison purpose. Random sample consensus (RANSAC) is an iterative method to estimate parameters of a mathematical model from a set of observed data that contains outliers, when outliers are to be accorded no influence on the values of the estimates. Therefore, it also can be interpreted as an outlier detection method. It is a non-deterministic algorithm in the sense that it produces a reasonable result only with a certain probability, with this probability increasing as more iteration are allowed. A basic assumption is that the data consists of "inliers" i.e., data whose distribution can be explained by some set of model parameters, though may be subject to noise, and "outliers" which are data that do not fit the model. The outliers can come, e.g., from extreme values of the noise or from erroneous measurements or incorrect hypotheses about the interpretation of data. RANSAC also assumes that, given a (usually small) set of inliers, there exists a procedure which can estimate the parameters of a model that optimally explains or fits this data.

The RANSAC algorithm is a learning technique to estimate parameters of a model by random sampling of observed data. Given a dataset whose data elements contain both inliers and outliers, RANSAC uses the voting scheme to find the optimal fitting result. Data elements in the dataset are used to vote for one or multiple models. The implementation of this voting scheme is based on two assumptions: that the noisy features will not vote consistently for any single model (few outliers) and there are enough features to agree on a good model (few missing data). The RANSAC algorithm is essentially composed of two steps that are iteratively repeated:

a) In the first step, a sample subset containing minimal data items is randomly selected from the input dataset. A fitting model and the corresponding model parameters are computed using only the elements of this sample subset. The cardinality of the sample subset is the smallest sufficient to determine the model parameters.
b) In the second step, the algorithm checks which elements of the entire dataset are consistent with the model instantiated by the estimated model parameters obtained from the first step. A data element will be considered as an outlier if it does not fit the fitting model instantiated by the set of estimated model parameters within some error threshold that defines the maximum deviation attributable to the effect of noise.

The set of inliers obtained for the fitting model is called consensus set. The RANSAC algorithm will iteratively repeat the above two steps until the obtained consensus set in certain iteration has enough inliers. The input to the RANSAC algorithm is a set of observed data values, a way of fitting some kind of model to the observations, and some confidence parameters. RANSAC achieves its goal by repeating the following steps:

1) Select a random subset of the original data. Call this subset the hypothetical inliers.
2) A model is fitted to the set of hypothetical inliers.
3) All other data are then tested against the fitted model. Those points that fit the estimated model well, according to some model-specific loss function are considered as part of the consensus set.
4) The estimated model is reasonably good if sufficiently many points have been classified as part of the consensus set.
5) Afterwards, the model may be improved by re estimating it using all members of the consensus set.

This procedure is repeated a fixed number of times, each time producing either a model which is rejected because too few points are part of the consensus set, or a refined model together with a corresponding consensus set size. In the latter case, we keep the refined model if its consensus set is larger than the previously saved model. Once the image processing is done, block wise storage will happen in HDFS of Hadoop Architecture. MapReduce will be used for implementing accessing and storage purposes.

Fig. 2: Inliers and outliers

Acknowledgement

It is my privilege to acknowledge with deep sense of gratitude towards project guide Mrs. Sanchari Saha for her valuable suggestions and guidance throughout course of study and timely help in completion of the preliminary project work on "Time Efficient Image Processing Framework using MapReduce". I would also like to thank project Co-ordinator Prof. Usha Ruby and all other faculty members of Computer Science and Engineering Department who directly or indirectly kept the enthusiasm and momentum required to keep the work done. I hereby extend my thanks to all concerned person who co-operated with me in this regard.

References

[1] Le Dong. A hierarchical distributed processing framework for big image data. IEEE Transactions on Big Data. DOI: 10.1109/TBDATA.2016.2613992.

[2] Tongliang Liu. A scalable two-phase top-down specialization approach for data anonymization using map reduce on cloud. IEEE Transactions on Pattern Analysis and Machine Intelligence. DOI: 10.1109/TPAMI.2016.2544314.

[3] Chang Xu. Multi-view intact space learning. IEEE Transactions on Pattern Analysis and Machine Intelligence. DOI: 10.1109/TPAMI.2015.2417578.

[4] Yang Yang. Robust discrete spectral hashing for large-scale image semantic indexing. IEEE Transactions on Big Data. DOI: 10.1109/TBDATA.2016.2516024.

[5] RANSAC Algorithm. Available from Wikipedia.