IJCAT - International Journal of Computing and Technology, Volume 6, Issue 4, April 2019
ISSN (Online) : 2348-6090
www.IJCAT.org
Impact Factor: 0.835

Multimedia Information Privacy Preservation with fusion of MapReduce, Fuzzy K-Means Clustering and Security for Cloud Storage

[1] Sayyada Fahmeeda Sultana; [2] Dr. Shubhangi D C

[1] Department of Computer Science & Engineering,
PDA College of Engineering, Gulbarga, India

[2] Department of Computer Science, VTU PG Center,
Gulbarga, India

Abstract - Multimedia data is expanding exponentially. The speedup growth of technology combined with storage capabilities and reasonable capacity has resulted in an explosion in multimedia availability and applications. Most data is available in the form of images and videos. Today a large amount of image data is produced through digital cameras, mobile phones and other sources. Processing this large set of images involves very complex and frequent operations in a large database that lead to challenges to improve query time and data storage capacity. Many image processing algorithms and computer vision are applicable to extensive data tasks. Mostly to run image processing algorithms on large data sets that are currently limited to the computing power of a single computer system. In order to handle such a huge data, cloud computing is used but storing data on cloud need security. Encrypting a complete image is a time consuming task to overcome the problem selective encryption is proposed based on MapReduce, Fuzzy K-Means Clustering and Data Encryption Standard(DES). The proposed scheme performs the feature extraction using MapReduce parallel speeding up the process ten times faster if ten Hadoop cluster nodes are involved. On the extracted features Fuzzy KMeans clustering is applied to segment the image and identify the region of interest(ROI). DES is applied on ROI to secure the image.

Keywords - *MapReduce, Image Feature Extraction, Mapper, Reducer, Fuzzy KMeans Clustering*.

1. Introduction

Many organizations manage sensitive multimedia information are using cloud computing as it provides resources that can be scaled easily, along with significant economic benefits in the form of reduced operational costs. However, it can be complicated to correctly handle sensitive data in cloud computing environments due to the range of privacy legislation and regulations that exist.
Cloud computing has raised several security issues including multi-tenancy, loss of control and trust. Cloud computing providers virtualize and containerize their computing platforms to be able to share them between different users (or tenants). Multi-tenancy refers to sharing physical devices and virtualized resources between multiple independent users or organizations. Loss of control is another potential breach of security that can occur where consumer's data, applications, and resources are hosted at the cloud provider's owned premises. As the users do not have explicit control over their data, this makes it possible for cloud providers to perform data mining on the users data, which can lead to security issues. In addition, when the cloud providers backup data at different data centers, the consumers cannot be sure that their data is completely erased everywhere when they delete their data. This has the potential to lead to misuse of the unerased data. In these types of situations where the consumers lose control over their data, they see the cloud provider as a black-box where they cannot directly monitor the resources transparently.

Multimedia means that computer information can be represented through audio, video, and animation in addition to traditional media text, graphics/drawings, images. Multimedia data is expanding exponentially. The speedup growth of technology combined with storage capabilities and reasonable capacity has resulted in an explosion in multimedia availability and applications. Most data is available in the form of images and videos. Today a large amount of image data is produced through digital cameras, mobile phones and other sources. Processing this large set of images involves very complex and frequent operations in a large database that lead to challenges to improve query time and data storage capacity. Many image processing algorithms and computer vision are applicable to extensive data tasks. Mostly to run image processing algorithms on large data sets that are currently limited to the computing power of a single

computer system. In order to handle such a huge data, it need to be stored on cloud, where privacy is a concern issue. To address this problems be proposed a privacy preservation scheme of multimedia information on cloud using MapReduce and fuzzy logic and security.

MapReduce is a two step process that includes a mapping step followed by a reducing step. A preliminary step in performing a MapReduce operation is to place the data in a distributed file system such that all the processing workers can access the data. The Process is transparent to customer and when the customer looks at the file in distributed file system it looks like the original file.

The mapping step is where a customer query is mapped to all the worker nodes. The complete data to be processed is sliced independently. Actually, the map is applied to logical splits of the data so that words that are physically split by data slicing are kept together. Once the mapping process is done, the output list of the map processes becomes the input list to the reduce process. The Reduction can take many forms and, in general collects and reduces the information from the mapping step.

The authors provided a brief overview of MapReduce with many design patterns related to MapReduce. Discuss both high-level theory and low-level coding of several computer vision algorithms such as classifier training, sliding windows, clustering, feature-of- bag, background subtraction, and image enrollment. Experimental evaluation is performed using 410 node cluster in Hadoop. [1].

The Split & Merge architecture was proposed for high-performance video processing, which is a generalization of the MapReduce model that justifies the use of resources by exploring demand computing. Dynamic resource sharing is used to illustrate the implementation of the Split & Merge structure, and showed a reduction in video encoding times to a specified time, regardless of the size of the video file entry on cloud storage. [2]

For data extraction, discuss issues related to emerging standards for data exchange and problem design for data integration in a distributed ownership system[3].
Compiling Parallel K-Means cluster based on MapReduce to expand data in applications makes clustering very large set of data a difficult task. The authors proposed a parallel-clustering algorithm K-Means based on MapReduce, which was widely adopted by both academia and industry. They used acceleration, scale, and scale to evaluate the performance algorithm. Results in the paper shown that parallel KMeans algorithm can be executed on commodity hardware for large size data [5].

A execution of time and space intensive computer vision algorithms on a distributed computing platform by using Apache Hadoop framework is proposed. Hadoop is a framework based on divide conquer strategy. Authors proposed to working by extracting color and texture features from inage and will be divided and assigned to equally to multiple nodes on the Hadoop cluster [7].

The MapReduce model of the Hadoop is a recent and common trend in the analysis of large data sets in a short period of time. It is important to parallel the aggregation algorithms using MapReduce to achieve efficiency in aggregation as a result of execution time. Comparing a parallel k-algorithm means using MapReduce to group documents and time to perform the assembly task with a sequential k-Means algorithm on data sets with varying sizes [11].

It is important to parallel the aggregation algorithms using MapReduce to achieve efficiency in aggregation as a result of execution time. Comparing a parallel k-algorithm means using MapReduce to group documents and time to perform the assembly task with a sequential k-Means algorithm on data sets with varying sizes [12].

In the proposed approach reduction is finding sum of intensities. On the result of Reduce step a Fuzzy KMeans clustering is applied to identify region of interest for encryption. In this paper the popular Data Encryption Standard algorithm is used to encrypt region of interest, as DES is a block ciphering technique.

2. Background and Technical Preliminaries

2.1. Clustering

Cluster analysis is used for automatic identification of natural grouping of things. It is also known as the segmentation technique. In this technique data instances that are similar to each other are categorized into one cluster. Similarity, data instances that are very different from each other are moved into different clusters. The user can specify a different, large or smaller, number of desired clusters based on their making business sense. The cluster analysis techniques will then define many distinct clusters from analysis of the data, with cluster definitions for each of these clusters. However, there are good cluster definitions, depending on how closely the cluster parameters fit the data.

The cluster can be represented by a central or modal value. A cluster can be defined as the centroid of the collection of points belonging to it. A Centroid is a measure of central tendency. It is the point from where the sum total of squared distance from all the points is the minimum.

2.2. K-Means Clustering Algorithm

KMeans is a clustering algorithm that iteratively computes the cluster and their centroids. It is a top down approach to clustering. Starting with a given number of K cluster.

Algorithm K-Means clustering[14]

Input: K number of Clusters, D list of data points
Step1: Choose K number of random data points as initial centroids
Step2: Repeat till Cluster Centers stabilize
Step 2.1. Allocate each point in D to the nearest of K centroids
Step 2.2. Compute centroid for the cluster using all points in the cluster
end

The algorithm works first by randomly selecting k clusters in a set of all clusters representing the primary cluster centers. Each remaining clusters are set to a more similar cluster, depending on the distance between the object and the middle of the cluster. The new average is then calculated for each cluster distance. This process is repeated until the standard function arrives. It is clear that the distance calculations between clusters and centers are not linked to the distance calculations between other clusters that have the corresponding positions. Therefore, calculations can be made for the distance between different clusters at the same time. In each repeat, the new positions used for the next repeat must be updated. Consequently, repeated actions must be carried out sequentially.

2.3. Data Encryption Standard (DES)

This is a same key encryption and decryption algorithm of category symmetric encryption algorithm brought by IBM. DES is one of the block-based ciphering technique, blocks of size of maximum 64 bit each, 64 bits of plain text is provided as input to DES and generates as output 64 bits of cipher text. The reapplying the same encryption algorithm decrypts the text, with minor modifications when applied to images. The key length is 56 bits [6].
DES is fully based on fundamental attributes of cryptography like substitution and transposition, it is a sixteen round process, and each round performs the steps of substitution and transposition as shown by algorithm 1.

Algorithm 1 DES encryption algorithm[13]

Step 1: The plain text block is passed to initial Permutation (IP) function.
The initial permutation performed on plain text.
Step2: initial permutation (IP) produces two halves of the permuted block; says Left Plain Text (L) and Right Plain Text (R).

Step3: Now each L and R to go through 16 rounds of encryption process as:
Round i has input L_{i-1},R_{i-1} and output L_i,R_i
$L_i = R_{i-1}$,$R_i = L_{i-1} \oplus f(R_{i-1},K_i)$
and K_i is the subkey for the 'i'th where $1 \leq i \leq 16$

$L_1 = R_0$, $R_1 = L_0 \oplus f(R_0,K_1)$
$L_2 = R_1$, $R_2 = L_1 \oplus f(R_1,K_2)$
$L_3 = R_2$, $R_3 = L_2 \oplus f(R_2,K_3)$
....
....
....
$L_{16} = R_{15}$, $R_{16} = L_{15} \oplus f(R_{15},K_{16})$

Step 4: L and R are rejoined and a Final Permutation (FP) is performed on the combined block. The result of this process produces 64 bit cipher text.

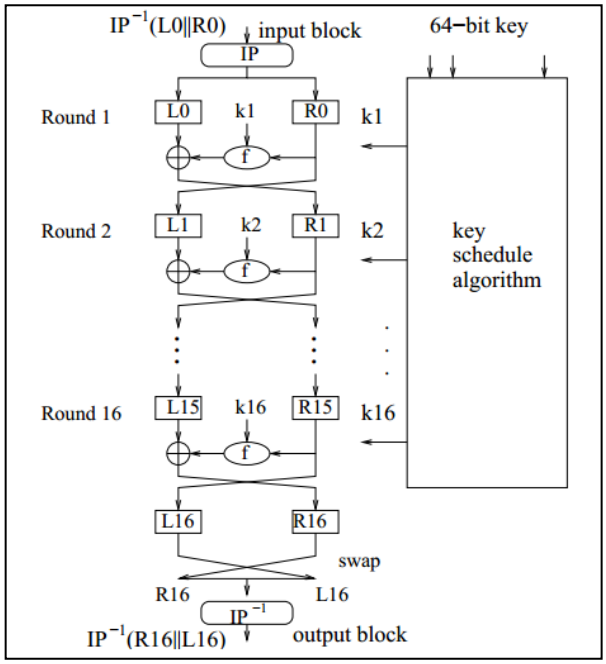


Figure 1. DES Encryption

3. Proposed methodology for Privacy Preservation with Fusion of MapReduce, Fuzzy K-Means Clustering and DES encryption algorithm

The proposed approach for privacy preservation of multimedia information like images is performed by selective encryption. To achieve selective encryption we segment image into foreground and background images, in this case we use foreground image as region of interest for privacy preservation.
The proposed methodology works to

- MapReduce Based Image feature extraction

- Fuzzy K-Means Clustering based on extracted feature from MapReduce to segment image into two clusters, fuzzy decision to identify foreground and background images
- Apply DES encryption algorithm on identified foreground image

3.1. MapReduce Based Image Feature extraction (MapReduce-IFE)

MapReduce is an Analytical programming tool to perform analysis on big data. We use MapReduce to identify feature of image. The Process starts by first representing image or any multimedia data into numerical representation as a 2D matrix. This 2D matrix is partitioned into blocks, each block is assigned as input to Mapper process. The Mapper split the assigned input into blocks of size nXn, each block is executed parallel. The Reducer is executed on individual blocks by counting the number of dark pixels to indicate variation in pixel intensity. Reducer counts the number variation in intensity and assigns the counter of number of variations, for each block.

Algorithm2 Mapper for Image Feature Extraction

Input: Block size nXn, Image size PXQ
Output: <key, value > pair which is the pair of <block_number, {start_i,end_i}>
Step1:block_number=0;
Step2: c=0, temp=0;
Step3 : for i = 1 : n : P-15
for j = 1 : n : Q-15
if(c==0)
c==1;
start_i(block_number)={ i , j }
end
block_number=block_number+1;
if(i==n && j==n)
end_i(block_number)={ i , j }
end
end
end

Mapper Algorithm2 splits and create the equal size blocks and return to Reducer process <key, value pair> which is <block number, starting & ending index of block>, where ending index is the last pixel index in a block. The process is speed up by applying parallel processing with multiple number of core processes. Further in algorithm3, Reducer reads the image count the pixels intensities in each block and returns as result <key,value > pair which in this case is <block_number,Count_i> of different intensities in block> with the help of MapReduce the difficult tast of feture extraction is executed parallel.

Algorithm3 Reducer for Image Feature Extraction

Input: Image in Numerical form I_N, block size nXn, size of I_N PXQ, Mapper output <block_number, {start_i , end_i}>
Output: <key, value > pair which is the pair of <block_number, Count_i>
Step1:for each block in <block_number, {start_i , end_i}>
Step2: Count_i=0;
Step3: for i = block_number{start_i{i}} : block_number{end_i{i}}
for j = block_number{start_i{j}} : block_number{end_i{j}}
context[nXn], con=0;
for(k=1:(nXn))
if (context(k)!=(valueof(I_N.value{i,j}))
context(k)= valueof(I_N.value{i,j});
Count_i+= valueof(I_N.value{i,j});
end
end
con.write(block_number,Count_i))
end
end

Algorithm 3 starts by reading as input I_N which is the numerical values of image, n is the number of rows and columns in a block, P,Q is the overall dimension of image with P representing number of rows and Q as columns. From Mapper process output <block_number, {start_i,end_i}> reducer uses this output to trace the blocks and count the number of varying intensities or values. To count the number of intensities in each block sum, context and con variables are used. Count_i is incremented as new intensity is encountered in a block. Context is an array of size nXn which holds the newly received intensity to check no duplicate intensities are counted, con matrix which forms the <key,value> pair of <block_number, Count_i>

3.2. Fuzzy K-Means Clustering based on extracted feature

The Second step in privacy preservation of multimedia information (image) is to segment the data into foreground and background. Image is segmented into two clusters based on features extracted from MapReduce-IFE process using Fuzzy K-Means Clustering and apply DES encryption algorithm on region of interest block a detailed procedure is explained in algorithm3.

Algorithm 4 Fusion of Fuzzy KMeans Clustering with Encryption

Input: Image I, Block sixe nXn, Image size (PXQ), <block_number,Count_i> from Algorithm 2 of MapReduce-IFE

Output: Region of interest Encrypted Image I' (foreground encrypted, background not encrypted)
Step1: Apply algorithm1 K-Means clustering on <block_number,Count_i> decision is based on Count_i with number of clusters to be two, return the block_number and cluster_number.
Step2: if number of blocks in first cluster< = second cluster

Encrypt all blocks in first cluster using DES encryption algorithm
I'=Encrypted blocks from first cluster+Plain Blocks in second cluster
(use block_number to arrange the blocks in correct position)
else
Encrypt all blocks in second cluster using DES encryption algorithm
I'=Encrypted blocks from second cluster+Plain Blocks in first cluster
(use block_number to arrange the blocks in correct position)
end
return Ciphered image I'

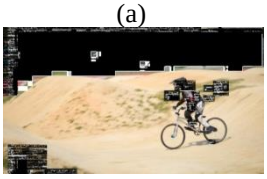
end

Algorithm 4 is the procedure to encrypt the images the reverse process applied for decryption, the keys for decryption involve the <block_number,count_i>, keys for decryption used in DES encryption, block size.

4. Results and Analysis

Simulation on the DBE encryption algorithm is done on Akiyo, Big Buck Bunny, Bridge (close), Bridge (Far), Bus, Carphone, Claire, Coastguard, Container, Elephants Dream, Waterfall, Template test video files video clips. Figure 2a shows original image, Figure 2b shows the encrypted image. The results show that perceptibility of ciphered image is decreased as the object of interest gets encrypted.

(a)



(b)

Figure 2. Image encrypted using Proposed MapReduce fuzzy KMean cluster image encrypted using DES encryption algorithm, (a) original Image (b) Region of Interest Encrypted Image

4.1. Experimental Evaluation and Configuration

K-means clustering based on the proposed scheme based on MapReduce is evaluated for efficiency, scalability, and accuracy. This experiment was implemented in JAVA to segment multimedia data into foreground and background, an experiment was executed of Amazon Web Services[16]. We used a cluster of 10 nodes with Apache Hadoop 2.6.3 [17]. One node is a master node used as head nodes and the other 9 nodes as core nodes. Each node is running with resources 4 VCore,8GiB, 32GiB memory. The local machine for the dataset owner is a desktop running OS X 10.1 with 8 3.3GHz CPU cores and 32GB memory. To support matrix related operations in our scheme, jblas library 1.2.4 [18] is adopted in the implementation. The dataset used for evaluation includes image (or audio mp3) converted into numerical values. To demonstrate that our scheme introduces reasonable computation and communication overhead, we also implemented a non-MapReduce based K-means under the local machine. All experimental results showed that the proposed MapReduce approach is more than 10 times faster than local machine implementation due to parallel execution of experiment.

5. Conclusions

A MapReduce based image Segmentation scheme is proposed that uses Fuzzy K-Means clustering to segment image into foreground and background. MapReduce perform feature extraction by first reading numerical equivalent of image as text input, then splits the input into blocks using Mapper process, the reducer process counts the number of different pixels values and return the sum of different pixels in each block with block number. This Reducer generated output gets as input to Fuzzy K-Means clustering and which gives two clusters, these are the foreground and background blocks of image. Then DES encryption algorithm is applied on foreground image. The proposed scheme is faster then executed on local machine with the utilization of parallel processes through the use of Hadoop cluster.

References

[1]. White, B., Yeh, T., Lin, J., & Davis, L. (2010, July). Web-scale computer vision using mapreduce for multimedia data mining. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining* (p. 9). ACM.
[2]. Pereira, R., Azambuja, M., Breitman, K., & Endler, M. (2010, July). An architecture for distributed high performance video processing in the cloud. In *2010 IEEE 3rd international conference on cloud computing* (pp. 482-489). IEEE.

[3]. Hawick, K. A., Coddington, P. D., & James, H. A. (2003). Distributed frameworks and parallel algorithms for processing large-scale geographic data. *Parallel Computing*, *29*(10), 1297-1333.

[4]. Amazon.com Inc. Amazon elastic compute cloud (amazon ec2), October 2012

[5]. Lv, Z., Hu, Y., Zhong, H., Wu, J., Li, B., & Zhao, H. (2010, October). Parallel k-means clustering of remote sensing images based on mapreduce. In *International Conference on Web Information Systems and Mining* (pp. 162-170). Springer, Berlin, Heidelberg.

[6]. Kumar, S., & Srivastava, S. (2014). Image encryption using simplified data encryption standard (S-DES). *International Journal of Computer Applications*, *104*(2).

[7]. Sabarad, A. K., Kankudti, M. H., Meena, S. M., & Husain, M. (2015, February). Color and Texture Feature Extraction using Apache Hadoop Framework. In *2015 International Conference on Computing Communication Control and Automation* (pp. 585-588). IEEE.

[8]. Kong, Z., Li, T., Luo, J., & Xu, S. (2019). Automatic Tissue Image Segmentation Based on Image Processing and Deep Learning. *Journal of healthcare engineering*, *2019*.

[9]. Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S., & Stoica, I. (2010). Spark: Cluster computing with working sets. *HotCloud*, *10*(10-10), 95.

[10]. Sardar, T. H., & Ansari, Z. (2018). An analysis of mapreduce efficiency in document clustering using parallel k-means algorithm. *Future Computing and Informatics Journal*, *3*(2), 200-209.

[11]. Sardar, T. H., & Ansari, Z. (2018). Partition based clustering of large datasets using MapReduce framework: An analysis of recent themes and directions. *Future Computing and Informatics Journal*.

[12]. Coppersmith, D. (1994). The Data Encryption Standard (DES) and its strength against attacks. *IBM journal of research and development*, *38*(3), 243-250.

[13]. Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, *28*(1), 100-108.

[14]. Dean, J., & Ghemawat, S. (2008). MapReduce: simplified data processing on large clusters. *Communications of the ACM*, *51*(1), 107-113.

[15]. Amazon, E. C. (2015). Amazon web services. *Available in: http://aws. amazon. com/es/ec2/(November 2012)*.

[16]. Joshi, S. B. (2012, April). Apache hadoop performance-tuning methodologies and best practices. In *Proceedings of the 3rd acm/spec international conference on performance engineering* (pp. 241-242). ACM.

[17]. Koitzsch, K. (2017). Standard Toolkits for Hadoop and Analytics. In *Pro Hadoop Data Analytics* (pp. 43-62). Apress, Berkeley, CA.