International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 14 (2018) pp. 11471-11478

Searching Text on 2D Image Over Encrypted Cloud Data using OCR

Suresha D[1], and K Karibasappa[2]

[1]Department of Computer Science & Engineering,
Dr.Ambedkar Institute of Technology, Bengaluru – 560056, Karnataka India.

[2]Professor, Department of Computer Science & Engineering,
Dayanand Sagar Academy of Technology and Management,
Bengaluru – 560082, Karnataka, India.

Abstract

In this paper, we propose a scheme to search a text keyword on 2D image over encrypted cloud data using OCR conversion. Nowadays, Storage as a Service (SaaS) has become a reliable method of storing the individual and organizations huge volumes of confidential data in the cloud server. The data stored on the cloud may be a text file with normal plain text and an image file that contains text or scanned image of a text file. It is also an essential to store this kind of image document into cloud server and provide security to it. There are many traditional schemes to search keyword over cloud data and retrieve the required files. The scheme should also support keyword search over image documents when it is stored on the cloud. In this proposed scheme, we are using OCR technique for image document to text file conversion.

Keyword: Keyword Search, Index Generation, Cloud Computing, Image Search

INTRODUCTION

CLOUD service is easily accessible and approachable, enabling many institutions and organizations including small revenue companies to store their confidential data in cloud as it are pay-on-use basis for storage. In cloud computing, more number of customers/tenants share the same storage space where organizations and individuals can get storage space on-demand and pay-as-you-use basis from cloud service providers (CSPs) like Amazon, Google, Dropbox, Zoho or any other. More customers prefer cloud storage due to the advantages of cloud computing like network access, storage space elasticity [1] etc. More number of data owners outsource their data into the shared cloud server. As a result of which one user may have control over the data stored by another user. This may lead to data loss or data may be altered by some unauthorized users or may be from CSPs as cloud server is maintained by them only. There are number of security threats posed for the outsourced cloud data by different users. So, it is always data owner's risk and the users may lose control over their data. Hence, the data owners need to encrypt their data using some private key before outsourcing their data into cloud, so that for other users without private key they cannot understand the original data [2]. The user stores not just the text files/documents but also the image documents where the image contains the text information. The main goal is not just data storage, but it is about to search the keyword on the encrypted data [3]. Once the data is downloaded from the cloud into the local machine, decrypt it and then search the plain text. It occupies more bandwidth of cloud and also decreases the performance. So, the challenge is to search on encrypted data in the cloud for better performance.

The Cloud Service Providers (CSPs) uses firewall mechanism for security of data stored by users or customers, but still the cloud server may not be secure. It is recommended that data owner should encrypt their data with the preferred encryption algorithm [4], [5] and then outsource the data into cloud server. Sometimes, not just the text documents but also image documents are added. It is easy to read the text content from image, if image file is retrieved. So, the text on image document needs to be encrypted before outsourcing into cloud. The challenge is to retrieve the relevant files from the huge set of files stored in the cloud with the help of keyword search method. Users can retrieve only the needed files using this technique. But, this technique may pose challenges to retrieve needed files when there are large number of outsourced encrypted image documents and enormous number of data users on same server, because it has to meet the requirements like performance with respect to search time, storage space and system stability.

Earlier most of the information were stored in textual format. Nowadays, information is stored in image format also where text data may be present. The data may include both alpha numeric and special characters. So, the system has to support searching keyword which exists in the image too. Directly storing the image with text may lead to information leak if the security is not provided for the image. Hence, text in the image is retrieved and stored in textual format, then encrypt and store in the cloud space along with image identifier for all textual information. Whenever the user searches for the keyword, the system has to retrieve the images which contain the input keyword and has to give the relevant files. Though the keyword search is possible on plain text, here it has to support search over encrypted data as the text on image is retrieved in a text format and that text file is encrypted before outsourcing into cloud. Also it has to support ranked search to find relevant results. As the cloud is pay as you use model, finding relevant

result is more important. For preserving privacy, system should not leak any information related to keyword and also details of image.

Motivation: In the previous scheme, the keyword search technique for text document over encrypted cloud data did not focus on scanned script documents. The main issue of the previous scheme is that the keyword is not extracted from the image document for indexing and retrieval of image document through query keyword request over encrypted data. We need to provide accurate and secure keyword search over encrypted image documents.

Contribution: In this paper, we developed a new model for keyword search on text image data over encrypted cloud environment. Binary search is used to search the keyword in the index file and retrieval of the top-k files sorted in descending order using heapsort. The sensitive data is protected from the cloud service provider and unauthorized users. Here, we address the challenges of secure search over encrypted image documents. The importance of our contributions is:

1) OCR based novel Image indexing and retrieval scheme is introduced on the text image.
2) A new algorithm is developed for indexing and searching over encrypted cloud data that reduces search time by O (log n).
3) A mathematical model is introduced for computation of each word in the index file.
4) Performance of the proposed work reduces the search time, search accurate results and consumes less storage space.
5) The proposed scheme is secure and demonstrated on real data sets.

Organization: In the remaining of the paper, the following information is presented: In Section II, related research works are discussed. Then, necessary background work for this paper is described in Section III. In Section IV, problem formulation and system model are explained. This section also has the detailed description on OCR Technique. In Section V overview of proposed search schema has been described. In Section VI, performance analysis for index time construction, storage cost of index and search time and security analysis are presented. Finally, in Section VII, the paper concludes with some suggestions for future work.

RELATED WORK

Bakhtiari et al., [6] have enhanced Secure Search over Encrypted Data in Cloud Computing. Secure Searchable Based Asymmetric Encryption (SSAE) algorithm gives Indistinguishability under adaptive chosen ciphertext attack (IND-CCA2). SSAE is numerically demonstrated secure and it has the capability to search without unravelling information. SSAE takes steady time in searching for large datasets. Computation cost, unique array is used for all documents and array updating is major problem in SSAE algorithm.

Zi et al., [7] have developed a science and technology dissertations retrieval (STDR) system based on cloud computing technology. Vector Space Model (VSM) and Hbase architecture are used to develop STDR retrieval system. The STDR system improves in computing and search time. The STDR system provides effective exploration but occupies more storage space and complexity is high.

Hassan et al., [8] have defined a novel word image-based document image indexing and retrieval framework. Distance Based Hashing (DBH) approach is used for indexing the word image. The time complexity of retrieval is less and size of hashing reduced in data structure by using Hierarchical hashing, Binary mapping functions extends to multi-probe hashing. The experiment done on different language scripts like Devanagari, Bengali and English. The entire framework only works on regular data not supporting for encrypted.

Wei et al., [9] have described a watchword recovery framework for finding words in chronicled Mongolian report pictures in light of the word spotting innovation. Off-line part and online part are included in keyword retrieval system. Off-line part creates word image indexing and each index term with a fixed-length feature vector developed by gaining the right range of the advanced coefficients of separate Fourier remodel on every profile feature.

Online part support for queried word image retrieval by calculating similarities and return the ranked results in decreasing order. [9] Takes more storage space, computation cost and not done on encrypted data. Sankar et al., [10] have presented a new framework for image document retrieval system based on word annotation. Word annotation framework algorithm avoids repetitive classification on similar features and results are improved by $O(\log N_1 \cdot \log N_2 \cdot K^2)$, where N_1, N_2 are training data size and test datasets and the cluster hierarchy branching factor is K. [10] have developed feature classifications and indexing scheme for image retrieval system. The retrieval system average precision performance is 0.8, lesser effects in segmentation and Limited by training data. Computation and storage cost is more and not equipped on encrypted image document.

Li et al., [11] have proposed FastRange Query Processing with Strong Privacy Protection for Cloud Computing that accomplishes indistinguishability against chosen keyword attack (IND CKA). Privacy Bloom filter tree (PBtree) algorithms (i.e., PBtree traversal width minimization and PBtree traversal depth minimization algorithms) are utilized to enhance query processing efficiency. Algorithms associated with PBtree construction, searching and optimization. The worst case complexity of PBtree query processing algorithm is $O(|R|\log n)$, where R is query result having set of data items and n is the total number of data items. PBtree algorithms have strong privacy and one time construction overhead.

Lu et al., [12] have studied on efficient top-K approximate search in the context of a relation with various attributes. ScanIndex and Top-Down (TD) are efficient algorithms used to compute top-K documents. If the ScanIndex similarity score is zero for distinct attribute than they are free from computation and below threshold (rather than zero) similarity scores are skipped with respect to TD. ScanIndex and TD improved the

performance by two order magnitude and experiments demonstrate on real life datasets. Privacy is lacking in top-K approximation search.

Pervez et al., [13] have addressed Privacy-aware searching issue and resolved by Oblivious Term Matching (OTM) on encrypted cloud storage data. OTM allows only the authorized users to search on encrypted cloud data and without revealing any information to the third party. Homomorphic encryption and proxy re-encryption techniques are used to hide the information for others. Performance improved by computation cost of searching query on encrypted data. Homomorphic encryption and proxy re-encryption techniques take more storage space to store 1248 bits long keys.

Kesidis et al., [14] have described a framework for word stopping method on historical machine-printed documents. Natural Language Processing (NPL) is used to access the content of Greek printed documents by searching keywords directly in digitalized documents based on stopping keywords. The quality of image improved by pre-processing, accessing printed document and searching is efficient. Pre-processing and two complementary segmentations increases in computation cost.

Roy et al., [15] have presented an efficient indexing and retrieval unconfined document layouts using Character Spatial Feature Table (CSFT). The Support Vector Machine (SVM) is employed to perform character labeling of multi-scaled and multi-oriented component. The positional information of the queried string splits into character pairs for searching process. String matching algorithm is employed to match the queried word and character pair sequence in documents. The advantage of [15] is detecting the queried word without extracting all the characters of the queried string. Orientation information technique is not used and not done on massive assortment of graphical documents.

Guo et al., [16] have proposed an efficient secure-channel free public key encryption with keyword search (SCF-PEKS) scheme on encrypted medical records in cloud environment. SCF-PEKS scheme secure against chosen keyword and ciphertext attacks (IND-SCF-CKCA) and keyword guessing attacks (IND-KGA). SCF-PEKS scheme compare to [17] less ciphertext length and computation overhead. If [16] and [17] are constructing pairing operation than its good for (IND-KGA). SCF-PEKS scheme is secure, efficient and well suited for medical data records.

Liu et al., [18] have addressed the issues of search pattern leakage and demonstrated the risk in applications by two concrete attack methods. [18] developed a grouping-based construction (GBC) to overcome with search pattern leakage issue and hiding the search pattern. GBC have strong security guaranty and reduces search pattern leakage. GBC generates fake queries to confuse attackers and incurs computation overhead.

Liu et al., [19] have developed a system to detect the equations region on the printed image document. The two major contributions are: 1) The system builds a simple algorithm for classification based on density of the symbol and no additional classifier required 2) the algorithm built into open source OCR engine to access by OCR community. OCR Tresseract improved performance after enabling equation detector model. The equation detection algorithm not having equation region parser runs more than once to get recognition stage that takes 20 – 30% increases in time and present precision rate is 74.3%.

Schreiber et al., [20] have presented a system to detect symbols on roads by using monoscopic or stereoscopic camera system. Monoscopic camera is used to estimate the vanished points and a distortion free top view by applying inverse perspective transformation. Stereoscopic camera is used for 3D reconstruction on captured image. Tresseract OCR system is used to categorize the symbols. The system improved in road surface mapping and localization using OCR. Both cameras detect 80% accuracy and the difficultly is non-standard road paintings.

Chen et al., [21] have proposed a new position index (P-index) scheme that support for encrypted Electronic Medical Records. P-index scheme is secure and privacy persevering against the adaptive chosen keyword attacks. P-index pro-vides secure search, flexible spacing and less false position rate. P-index is efficient to work on cloud environment because it does not rely on pairing. The complicate computation occupies more computation time for index file. In the event that nor is accessible on your word processor, please utilize the text style nearest in appearance to Times. Abstain from utilizing bit-mapped text styles. Genuine Type 1 or Open Type text styles are required. If you don't mind insert all text styles, specifically image textual styles, also, for math, and so forth.

BACKGROUND

There are many interesting research works being carried out to find the efficient way of searching the keyword on image documents over encrypted cloud data. Various techniques are used to achieve the keyword search on image data like Optical Character Recognition, Optical Word Recognition, Intelligent character and word recognition. In recent days, handwriting recognition is also being used. Also, research is being carried out on different languages and scripts that are available in various countries. In [4] the authors have used word spotting technique to retrieve Mangolian words. Similarly, in [9], they access the Greek documents. Some of these techniques are used not just to retrieve text but also the equations in [14] and symbols in [15]. All these schemes focused on single type of problem statements or situation and they have not concentrated on encrypting the text data after retrieving the image. Our proposed work focusing on searching text on 2D image over encrypted cloud data and the OCR technique is used to retrieve the image documents.

PROBLEM STATEMENT AND SYSTEM MODEL

A. System Model

There are three entities in the system architecture i.e., Information owner, Cloud Server and Information user shown in fig 1. Information owner wants to outsource 'n' image documents $I = I_1, I_2, \ldots, I_n$ to the cloud server. The scanned

image document converts and stores information into a text document by using (Optical Character Recognition) OCR Tressact opensource tool. Extract keywords from the text document and remove stop words from the extracted keywords then compute a mathematical model for security concern. The computed keywords list is stored in the index file I'. Finally outsource the encrypted index file and encrypted image documents to the cloud server.

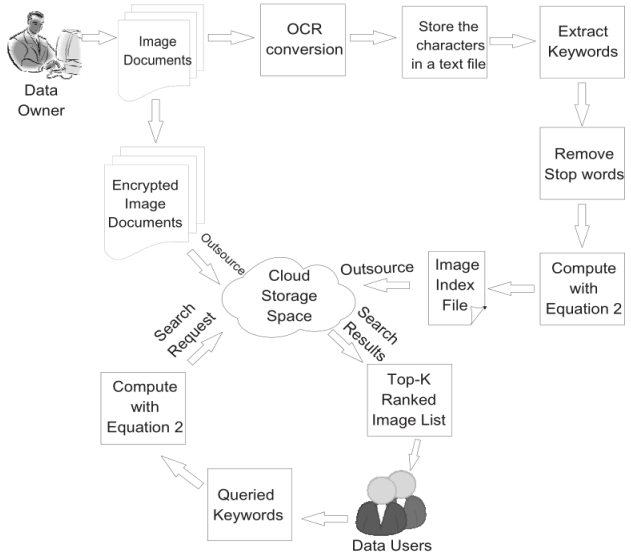

Figure 1. System Model for searching text in Image Over Encrypted Cloud Data using OCR

Cloud server provides space for storing the outsourced information from Information owner. The cloud server establishes communication between the Information owner and Information users without leaking any information to the unauthorized users. Information users need to find out the document from the cloud with the assistance of queried keywords. The queried keywords calculate with a same mathematical model which is used from the data owner then search for the queried keyword within the index file. The matched top-k documents are returned to the Information users in decreasing order based on the frequency of the queried keywords with top-k matched files.

Information owner encompasses a set of n image documents ID=I_1, I_2, I_3...I_n. These image documents are encrypted by using blowfish algorithm and encrypted image documents EID=EI_1, EI_2, ..., EI_n are outsourced into cloud server. Parallelly, data owner will convert the image documents into text file using OCR (Optical Character Recognition) conversion technique and stores the characters and words of image documents in the text file F. Then the data owner extracts keywords from the text file and removes the stopwords. Data owner then computes keywords using equation 2 to generate image index file and the image index file is outsourced into cloud storage space.

When the user inputs the query keyword (QKW) to perform the search operations, system generates the search request by using equation 2. The search request is forwarded to the cloud server. The server will reply to the query by generating Top-K ranked image list. The search results displayed based on any matches of the query keyword in the index, and then the server returns the image documents which contain that keyword.

The OCR works as follows (fig.2): Initially the image document is retrieved. After scanning the image document, crop the image that has only the text region. Detect the characters by lining and separating each character. After character classification is done, the characters are merged to form one string. The string data is generated from the number of characters retrieved from image document and stored it in text file which is used to generate index file.

B. Design Goals

The outline of the system model offers keyword search over outsourced image document within the cloud with the subsequent security and performance problems.

1) Privacy-preserving: The model is intended to satisfy the privacy challenges and avoids a cloud service provider and alternative cloud users from accessing any info from any image documents or from index kept in the cloud.
2) Text search on image document: Our search theme supports keyword seek for the input keyword and from the image files containing the keyword.
3) Efficiency: The above functionalities are achieved with low storage, low network traffic, low computation and search time.

C. Computation Method

The Score S is calculated using the frequency of every term within the individual file. The expression for normalized Score calculation is as follows:

$$S = \frac{freq}{\max - freq} \qquad (1)$$

Where freq → frequency of every term in a file, max_freq → maximum frequency after considering all the files within the folder and S → is Score obtained by $^{freq}/_{\max - freq}$. A new mathematical model for encrypting the keyword is given below:

$$a(w_i) = (a_0 x^k + a_1 x^{k-1} + \cdots + a_n x^{k-n}) \qquad (2)$$

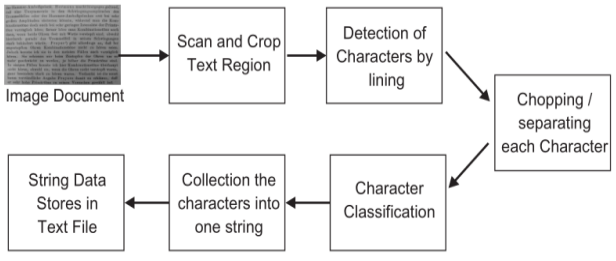

Figure 2. OCR

$$\alpha(w_i) = \sum_{p=0}^{n} a_b x^{k-p} \quad (3)$$

where x - may be a real and it ought to be same for each index keyword and queried keyword, k - may be a length of the keyword (i.e., if the keyword is Network than the length of the keyword is 7) and p - is that the position of every letter ($0 \leq p \leq n$) (if the keyword Network position of letter e is two and r is 6).

D. OCR

The Tesseract OCR is an open source tool for recognition of character from scan or printed image. OCR scans and crops the selected text region from the entire 2D image. Optical Character Recognition (OCR) is a technology which converts different type of documents like scanned image documents, pdf files or text image captured by camera into editable and searchable data. The OCR technology works as follows:

Initially the image document is retrieved; then OCR pre-processes the images to improve the quality of data recognition including techniques like line removal, script recognition, layout analysis or segmentation. In this process, after scanning the image document, it crops the image to extract only the text region, and then it detects the characters by using line and word detection technique and separates each character. After character classification is done, the characters are merged to form one string. The string data is generated from the number of characters retrieved from image document and stored it in text file which is used to generate index file.

PROPOSED SCHEME

Algorithm: This section provides a summary of our theme which is proficient graded keyword search scheme over image document. This theme is designed as follows.

1) Initialization: This section dealt by information owner IO to initialize the theme. It performs OCR conversion on image documents to extract the string data from the image.
2) Index generation: This section dealt by information owner to make the image index and set of distinct keywords of image documents and outputs the index file.
3) Query generation: The query is generated by information user IU to get the query out of the keyword being given as input for search. The query is generated using equation 2.
4) Search: The search section is executed by CS to search for the image files which contain the keyword. It takes query and index as input and returns the image documents where the keyword is present.

```
Index Generation Phase
Input: Directory name which contains images
Output: frequency file
Function: create _ frequency _ file()
for each image file in directory;
command tesseract imagename text_ file _ name;
for file in extracted _ files;
Read contents of all file;
Convert the contents to lowercase;
Remove stopwords from the contents;
Remove punctuations from the contents;
for each keyword in contents;
if(len(keyword)<2);
Remove the keyword from contents;
wordsset = Remaining keywords from contents;
/*Adding to dictionary for each keyword in wordsset;
if keyword already added in keydictionary;
append(fileid, frequency);
else add(keyword: fileid, frequency);
Writing to file;
for each keyword in keydictionary;
frequencyfile.write(convert_to_number(keyword));
sorting the file;
frequencyfile.sort();
OUTPUT: frequencyfile;
```

The directory name which contains the images is given as input to the system. It then extracts the entire image and the text files from the input directory. Then the system reads all the contents from each file in the set of extracted files one after the other. First, it converts all the contents into lowercase, then it removes stopwords and punctuation marks from each file. If length of any word is less than two, then such words are removed from contents. Then the remaining keywords are stored in separate file. Once counting the frequencies is done, next step is to add the keywords into dictionary along with the file id and frequency. If keyword is already added in key dictionary, then append file id and frequency. Otherwise, add the keyword along with file id and frequency.

```
Query Search Phase
Input: Queried keyword
Output: topk fileids
Function: Keyword _ Search(number _ of _ lines)
contents = frequencyfile.readlines();
low = 0;
high = number _ of _ lines− 1;
while (low ≤ high)
mid = (low+high)/2;
ifcontents[mid] == key;
break;
else if contents[mid]<key;
low <- mid+1;
else;
high <- mid-1;
for i=1 to contents[mid][i]!=',';
append(content[mid][i]) to fileid;
for j=i to contents[mid][j]!=NULL;
append(content[mid][j]) to frequency;
heapify(frequency);
for i = 0 to k;
topk.append(frequency.pop());
```

Next step is to write the keyword details into frequency file. Here, we convert the keyword into number and this number is unique across all the keywords available in the frequency file. Then, sort the frequency file based on unique number for each keywords stored in separate line in the file and this sorted frequency file is the output. The queries keyword is given as the input to the system. Each line is read from the frequency file and stored in variable called contents. As each keyword is converted into unique number and stored in a separate line of the file. Then, by using binary search technique, we search for the queried keyword. Then extract the file ids and frequencies for each keyword from the file. Once the frequency for all the key-words are extracted, build a max-heap of frequencies. Then retrieve the top-k elements from the frequency heap and return back the file ids of top k elements which are top-k elements with maximum frequency.

PERFORMANCE

Our proposed system is verified by implementing the search system on the cloud server. Our experiment includes a user and a server. The search system is implemented using Java on windows platform with Quad core CPU running at 1.46 GHz machine and the encrypted image documents are stored on the commercial public cloud Amazon cloud services like S3(Simple storage service). The performance is evaluated for index generation time, index storage space, search time, OCR generation and security analysis.

A. Index Storage Space

As the index file size is very less, the storage space for index file is not an issue in cloud server which is always bigger in size. Rather than storage space for index, we have to give importance to data security.

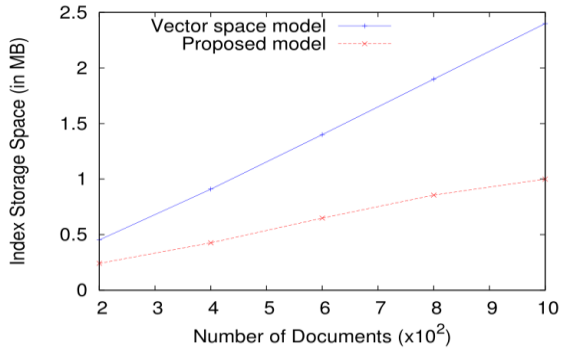

Figure 3: Index Storage Space vs Number of Documents

Fig 3 indicates the index storage space for both the schemes. The storage cost for index of 400 image documents in vector space model is 986.2kb whereas in proposed scheme it costs just 489.6kb. Vector space model consumes 1.5 MB for 600 image documents whereas it costs 676.3 kb with our proposed scheme. Continuing our experiment with still more number of documents, for 800 image documents vector space model consumes 1.9 MB for index file whereas proposed scheme costs just 859.0 kb for same number of documents. Extensive experiment shows that proposed scheme consumes very less space for storing file compared to VSM method.

B. Index Search Time

In this section, the performance of proposed search scheme is evaluated with increasing number of image documents in each iteration. The keyword search process is initiated by cloud server. Fig 4 shows the search time for both vector space model and proposed scheme. It shows that search time is less and more efficient in our proposed scheme compared to VSM scheme. When the index is generated for 400 image documents, the VSM takes 0.0112ms to search for a keyword whereas our proposed scheme consumes just 0.0028ms. Vector space model consumes 0.0172ms to search for a keyword with 600 image documents while our proposed scheme takes 0.0019ms. With 800 image documents, the VSM scheme takes 0.0174ms and our proposed scheme takes just 0.0046ms to search a keyword.

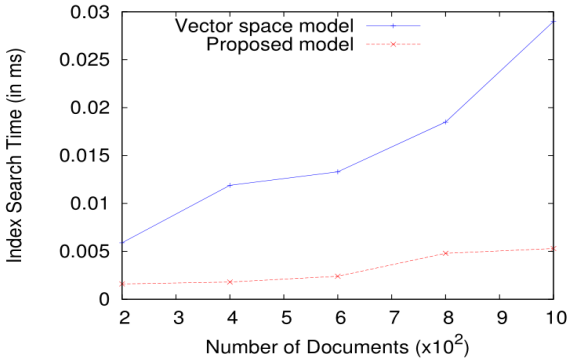

Figure 1. Index Search Time vs Number of Documents

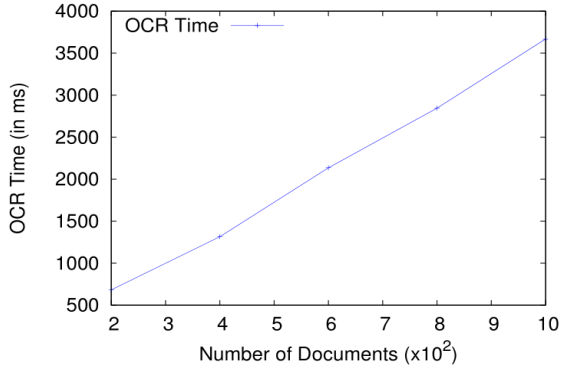

Figure 2. OCR Time vs Number of Documents

The extensive experimental results show that the proposed search scheme consumes less time for searching the keyword over an image document compared to vector space model scheme.

C. Security Analysis

We evaluate the security of the proposed scheme by finding out the number of data files which are stored in cloud server and they should not be read or any modifications should not be done on data by any unauthorized data user or one who belong to service provider. The data files D which are converted to text document from image document using OCR technique are encrypted by using bluefish algorithm and after encryption only the documents are outsourced into cloud server. Though the text document contains normal plain text after OCR conversion from image to text file, it will not have any plain text after they are all encrypted. As we are outsourcing the encrypted files into cloud server, it is difficult to read and understand the data even if it is attacked by any unauthorized data user. Even though the index file is also outsourced into cloud server, it is not possible to decrypt the keywords in index file without secret key. Also, the file ids and frequency of keywords are encrypted in index file. Hence, it is difficult to decipher and arrange in proper readable format. Thus, our proposed scheme preserved the privacy of data.

CONCLUSION AND FUTURE WORK

The proposed model is more efficient than the existing vector space model and also supports far larger data sets. Finally, the proposed model evaluated on the real data set which shows reduction in search time and index storage space. In this paper, we solve the problem of searching text keyword on 2D image over encrypted cloud data. The proposed scheme includes OCR conversion for image document to text file. Since the text files are encrypted and also the file ids and text in index file is encrypted, this schema provides better security to the data. This scheme is evaluated with real-time datasets to evaluate the efficiency of storage cost and keyword search time. The efficiency of proposed scheme with respect to index storage space and keyword search time in index is high compared to vector space model. Further, we would like to explore synonym based search over image document as done in our other work with encrypted cloud data. Also, we will be exploring with searching the symbols mainly scientific symbols over image document. Further, would like to do research on searching the keywords of other Indian languages which supports Unicode format using techniques other than OCR, as OCR faces difficulty in recognizing words of other Indian language scripts.

REFERENCES

[1] S. Raghavendra, C. S. Reddy, C. Geeta, R. Buyya, K. Venugopal, S. Iyengar, and L. Patnaik, “Survey on data storage and retrieval techniques over encrypted cloud data,” International Journal of Computer Science and Information Security, vol. 14, no. 9, p. 718, 2016.

[2] S. Raghavendra, S. Girish, C. Geeta, R. Buyya, K. Venugopal, S. Iyengar, and L. Patnaik, “Split keyword fuzzy and synonym search over encrypted cloud data,” Multimedia Tools and Applications, pp. 1–22, 2017.

[3] S. Raghavendra, C. Geeta, R. Buyya, K. Venugopal, S. Iyengar, and L. Patnaik, “Drsms: Domain and range specific multi-keyword search over encrypted cloud data,” International Journal of Computer Science and Information Security, vol. 14, no. 5, p. 69, 2016.

[4] S. Raghavendra, K. Meghana, P. Doddabasappa, C. Geeta, R. Buyya, K. Venugopal, S. Iyengar, and L. Patnaik, “Index generation and secure multi-user access control over an encrypted cloud data,” Procedia Computer Science, vol. 89, pp. 293–300, 2016.

[5] S. Raghavendra, G. Mara, R. Buyya, V. K. Rajuk, S. Iyengar, and L. Patnaik, “Drsig: Domain and range specific index generation for encrypted cloud data,” in Computational Techniques in Information and Communication Technologies (ICCTICT), 2016 International Conference on. IEEE, 2016, pp. 591–596.

[6] M. Bakhtiari, M. Nateghizad, and A. Zainal, “Secure Search over Encrypted Data in Cloud Computing,” 2013 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), pp. 290–295, 2013.

[7] J. Zi-long, G. Shu, and X. Xiong-wei, "The Design and Realization of a Science and Technology Dissertations Retrieval System based on Vector Space Model under Cloud Computing Environment," 2013 Fifth International Conference on Computational and Information Sciences (ICCIS), pp. 329–331, 2013.

[8] E. Hassan, S. Chaudhury, and M. Gopal, "Word Shape Descriptor Based Document Image Indexing: A New DBH-Based Approach," International Journal on Document Analysis and Recognition (IJDAR), vol. 16, no. 3, pp. 227–246, 2013.

[9] H. Wei and G. Gao, "A Keyword Retrieval System for Historical Mongolian Document Images," International Journal on Document Analysis and Recognition (IJDAR), vol. 17, no. 1, pp. 33–45, 2014.

[10] K. P. Sankar, R. Manmatha, and C. Jawahar, "Large Scale Document Image Retrieval by Automatic Word Annotation," International Journal on Document Analysis and Recognition (IJDAR), vol. 17, no. 1, pp. 1–17, 2014.

[11] R. Li, A. X. Liu, A. L. Wang, and B. Bruhadeshwar, "Fast range query processing with strong privacy protection for cloud computing," Proceedings of the Very Large Data Bases Endowment, vol. 7, no. 14, pp. 1953–1964, 2014.

[12] W. Lu, J. Chen, X. Du, J. Wang, and W. Pan, "Efficient Top-k Approximate Searches Against A Relation with Multiple Attributes," World Wide Web, vol. 14, no. 5-6, pp. 573–597, 2011.

[13] Z. Pervez, A. A. Awan, A. M. Khattak, S. Lee, and E.-N. Huh, "Privacy-Aware Searching with Oblivious Term Matching for Cloud Storage," The Journal of Supercomputing, vol. 63, no. 2, pp. 538–560, 2013.

[14] A. L. Kesidis, E. Galiotou, B. Gatos, and I. Pratikakis, "A Word Spotting Framework for Historical Machine-Printed Documents," International Journal on Document Analysis and Recognition (IJDAR), vol. 14, no. 2, pp. 131–144, 2011.

[15] P. P. Roy, U. Pal, and J. Lladós, "Word Searching in Unconstrained Layout using Character Pair Coding," International Journal on Document Analysis and Recognition (IJDAR), vol. 17, no. 4, pp. 343– 358, 2014.

[16] L. Guo and W.-C. Yau, "Efficient Secure-Channel Free Public Key Encryption with Keyword Search for EMRs in Cloud Storage," Journal of medical systems, vol. 39, no. 2, pp. 1–11, 2015.

[17] L. Fang, W. Susilo, C. Ge, and J. Wang, "Public Key Encryption with Keyword Search Secure Against Keyword Guessing Attacks without Random Oracle," Journal of Information Sciences, vol. 238, pp. 221–241, 2013.

[18] C. Liu, L. Zhu, M. Wang, and Y.-a. Tan, "Search Pattern Leakage in Searchable Encryption: Attacks and New Construction," Journal of Information Sciences, vol. 265, pp. 176–188, 2014.

[19] Z. Liu and R. Smith, "A Simple Equation Region Detector for Printed Document Images in Tesseract," 12th International Conference on Document Analysis and Recognition (ICDAR), pp. 245–249, 2013.

[20] M. Schreiber, F. Poggenhans, and C. Stiller, "Detecting Symbols on Road Surface for Mapping and Localization using OCR," IEEE 17th International Conference on Intelligent Transportation Systems (ITSC), pp. 597–602, 2014.

[21] Y.-C. Chen, G. Horng, Y.-J. Lin, and K.-C. Chen, "Privacy Preserving Index for Encrypted Electronic Medical Records," Journal of medical systems, vol. 37, no. 6, pp. 1–7, 2013.