International Journal Of Engineering Research And Development
e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 14, Issue 2 (February Ver. I 2018), PP.21-26

Cloud Storage Optimization Approach Using Compressive Sensing

M.Tanooj Kumar[1], Dr.M.Babu Reddy[2]
[1,2]Department of Computer Science, Krishna University, Machilipatnam
Corresponding Author: M.Tanooj Kumar

ABSTRACT:-Management of massive data or signals is one of the difficult problems in cloud storage. To handle this, most of the Cloud Storage Providers (CSP's) incorporate data compression methods to utilize the storage space optimally. We present a compressed sensing based method that allows performing data or signal capturing, its compression and encryption at the same time. We also present techniques specially for reconstructing the signal or data and evaluate their performance.
KEY WORDS:-Cloud Storage**,** Compressive sensing, Sparsity, Basis Pursuit, Convex optimization.

Date of Submission: 10-02-2018 Date of acceptance: 26-02-2018

I. Introduction

Cloud Storage is a remote, online, data storage service to store, access and share data over the internet. In accordance with the growing demand for cloud storage in an exceedingly safe and fast manner, researchers began to integrate cryptographic and compression techniques. The integration of these two methods may divide into three categories based on the sequence of these two processes. In the first category, cryptographic technique is applied after the compression method, which focuses more on security than the compression. In the second category, compression technique is applied after the cryptographic method, which has an advantage of choosing lossy, lossless or combination of both loss and lossless compression technique. In the third category, both compression and cryptographic techniques will be in a single process either partially or as compressive sensing (CS), which gives a very good data safety assurance with a low computational complexitythat it is qualified for upgrading the efficiency and security of data storage.

The reminder of this paper is as follows. Section II provides background and motivation of the work; Section III describes the design of proposed system; Section IV describes the implementation and experimental results; and, finally, we give concluding remarks in Section V.

II. Background

A.Basic theory of compressive sensing

Compressive sensing (CS) is one of the alternatives to capture the attributes of a signal using very few measurements, which performs both sampling as well as compression, along with encryption of input data at a time [1, 2]. Compressive sensing was introduced by Candes et al [3] and David L.Donoho in 2004 [4]. It includes the random projection of the signal and reconstructing it from far fewer samples than traditionally required. It depends on two principles: sparsity and incoherence [5].

1)Sparsity:

A signal is called sparse if most of its components are zeros. For representing a signal in a sparse form, the signal should be projected onto a convenient basis (Discrete Cosine Transform, Discrete Fourier Transform etc.) $\Psi = [\Psi_1, \Psi_2 \ldots \Psi_n]$. Using this basis, the signal can be represented as $f = \Psi . x$,where$x$ is the coefficient vector for f under the basis Ψ :

$$f = \sum_{i=1}^{n} \Psi_i . x_i \quad (1)$$

A sparse approximation of a given signal means that the signal can be represented by k (k << n) non zero coefficients and the remaining, are close to or equal to zero, can be discarded without loss of signal quality.

2) Incoherence:

This property insists that the measurement basis (Φ) and sparsifying basis (Ψ) are incoherent. The measurement basis (Φ) is used for recognizing the signal f as in (2) and the sparsifying basis is used to represent the signal f.

Data acquisition using compressive sensing is done through inner product operations between target signal and a sensing matrix $\Phi \in R^{mxn}$:

$$y = \Phi . f \quad (2)$$

$y = \Phi \,.\, \Psi \,.x$
$y = \theta \,.x$, where $\theta = \Phi.\Psi$ (3)

The largest correlation between any two elements of Φ and Ψ is computed by coherence and mathematically, the coherence between the sensing basis Φ and the representation basis Ψ is

$\mu(\Phi,\Psi) = \sqrt{n}.\max_{1\leq k,j\leq n} | < \Phi k, \Psi j > |$ (4)

3) Under-sampling and sparse recovery:

For sparse recovery, preference will be given to compute all the n coefficients of signal *f*, but subsets of these are usually observed and data is collected.

$y_k = <f, \Phi_k> , k \in M$ (5)

Where, M is a subset, such that m < n. In general, we assume that x is sparse,that is, it is a linear combination of only k << n basis vectors. Solving the equation (3) for x is equivalent to solving for f as Ψ is a known pre-determined basis.The goal of the compressive sensing is to design measuement matrix Φ and a reconstruction algorithm.

4) Measurement / Sensing Matrix (Φ):

The measurement / sensing matrix (Φ) plays an important role in the process of recovering or reconstructing the original signal. This matrix can be random ones, which is typically Guassian random matrix in numerous CS applications, or can be predefined as per some knowledge about the statistical characteristics of the original signals. The measurement matrix should satisfy the condition called Restricted Isometry Property (RIP) and is givey by

$1 - \delta_K \leq \frac{||\theta V||2}{||V||2} \leq 1 + \delta_K$ for some $0 < \delta < 1$ (6)

B. Optimization Techniques

At Receiver side, several reconstruction algorithms, either within the context of greedy approaches or convex optimization,, are used to recover or reconstruct the signal from a minimum number of measurements. Convex optimization can be formulated as follows [6]:

$\mathrm{Min}||x||_p$ such that $y = \Psi.\ \Phi.\ x = \theta\,.x$ (7)

Where, θ = Reconstruction matrix and $||x||_p$ is the p-norm given by$[\sum_{n=1}^{N} (x_n)^p]^{1/p}$ (8)

Various algorithms have been developed to solve this problem through linear programming. One solution is achieved by L_0 – norm i.e. p = 0 in equation (7), but it is NP hard problem. L_2 – norm or least squares minimization problem i.e. p = 2 in equation (7) gives not sparse solution. L_1 – norm i.e. p = 1 in equation (7) gives sparse solution and good reconstruction probabilities. Most commonly used algorithms of convex optimization are Basis Pursuit (BP) [7], Basis Pursuit De-noising (BPDN) [7], e Least Angle Regression (LARS) [8], Least absolute shrinkage and selection operator (LASSO) [9]. A greedy approach solves this problem by finding the solution, step by step, in an iterative fashion. Most popular algorithms are Matching Pursuit (MP) [10] and its derivative (OMP) [11] due to their low implementation cost and high speed of recovery. However the Basis Pursuit performance seems good as compared to OMP and its derivatives from minimum measurements perspective [12, 13]. A smaller measurement rate implies less sampling and communication cost at the transmitter side.

III. Proposedmethod

The block diagram of the proposed method is shown in figure. I. It consists of two main parts: Client and Cloud (Server). At the client side, the input signal *f* is given with N samples. Then *f* is converted into some domain in which *f* has sparse representation. After this conversion signal *f* is transformed into K-sparse signal *x*. Where K is largest coefficients obtained using thresolding. These K largest coefficients contain most of the information about signal. Then it is multiplied with sensing matrix Φ (fixed) and the result will give M- length measurement matrix *y*. The client has to Φ and Ψ. The measurement matrix *y* is uploaded to cloud. Whenever the client requests the file, in response to it the cloud sends y. The client computes θ = Φ.Ψ and executes the recovery algorithm, recovers *x*. After that, the client performs reverse sparsity operation and rebuilds the input signal $f\ (=\Psi^{-1}.\ x)$.

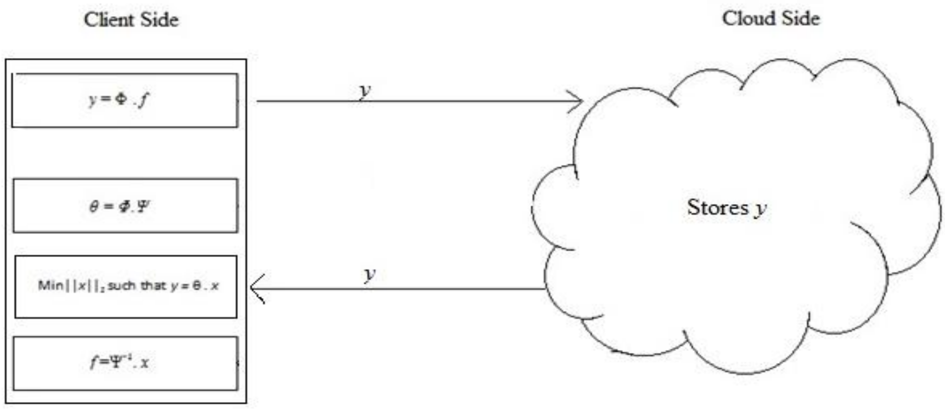

Fig.I Block diagram of proposed method, including both Client and Cloud side, with recovery task

IV. Experimental Results

Experimental results of the proposed method were as follows.

A. Text Message

Let assume that the message of the text file is
THIS IS A SECRET MESSAGE FOR BOB

T	H	I	S		I	S	
01010100	01001000	01001001	01010011	01000000	01001001	01010011	01000000

A		S	E	C	R	E	T
01000001	01000000	01010011	01000101	01000011	01010010	01000101	01010100

	M	E	S	S	A	G	E
01000000	01001101	01000101	01010011	01010011	01000001	01000111	01000101

	F	O	R		B	O	B
01000000	01000110	01001111	01010010	01000000	01000010	01001111	01000010

After assigning the Extended ASCII code to each character in the message, the binary code forms a vector of length 256; 90 vector elements are non-zero so this vector has relative sparsity 0.35. It was checked experimentally that measurement matrix (generated with secret key) must have at least compression ratio 0:8 (i. e. matrix size 205 x 256) for successful reconstruction of message. Basis Pursuit algorithm is used for the reconstruction of the message and successfully reconstructed the message.

B. Images

In this section, the experimental (simulation) results for image compression using compressive sensing are presented. These algorithms are applied on Cameraman (Figure 1) and Lena(Figure 2) images, one each, and their performance is evaluated using Compression Rate, Compression Processing Time, PSNR and upload time. Table I, II &III shows the results of PSNR and rate of compression for Cameramen and Lena standard images of different sizes. The parameters are defined as follows:

1) Rate of compression (RC):

Compression Ratio is defined as the ratio between the uncompressed file sizes in bits to compressed file size in bits. Rate of compression is defined as the reciprocal of compression ratio.

Compression rate = 1/ compression ratio

2) Compression processing time:

It is the time taken for compressing the file.

3) Peak Signal to noise Ratio (PSNR):

Estimation of the peak error between the compressed image and original image is called PSNR. Better quality image contains higher PSNR. To process the PSNR first MSE (Mean Square Error) is figured. The mean- square error (MSE) between two images $P_1(m,,n)$ and $P_2(m,n)$ is

Mean Square Error (MSE) = $1 / mxn\ \Sigma\ [P_1(m,n) - P_2(m,n)]^2$

Where, m and n are the number of rows and columns in the input images respectively.

PSNR = $10\ \log_{10}(\frac{R^2}{MSE})$

and R is maximum fluctuation in input image.
4) Upload Time:
It is time taken to upload the compressed file to storage and is calculated in terms of data size

Image Name	Sample Rate	Image size	M	Compression Ratio	Rate of compression	PSNR		DECOMPRESSION TIME	
						BP	L1	BP	L1
LINA	0.3	8x8	19	1.8529	0.53971	13.6669	8.4792	0.1092	0.0780
		16x16	77	1.75	0.57143	13.6655	9.0211	0.4992	0.546
		32x32	307	1.5588	0.64154	15.7078	8.8484	5.5224	5.5068
	0.4	8x8	26	1.7801	0.56176	15.8539	10.8289	0.1092	0.1092
		16x16	103	1.6101	0.62107	15.8645	9.0701	0.624	0.6084
		32x32	410	1.3252	0.75462	17.5817	9.5448	6.9576	6.786
	0.5	8x8	32	1.7347	0.57647	16.5932	11.6743	0.156	0.1092
		16x16	128	1.4803	0.67554	17.2657	10.3686	0.7332	0.7332
		32x32	512	1.1182	0.89431	18.2764	10.3302	8.3461	8.3305
	0.6	8x8	39	1.7128	0.58382	17.6838	11.7495	0.1028	0.1128
		16x16	154	1.3976	0.7155	17.9365	11.7273	0.6864	0.7176
		32x32	615	1.0015	0.99852	19.3594	11.5678	9.0637	9.0949
	0.7	8x8	45	1.6268	0.61471	18.9508	11.8063	0.1092	0.1248
		16x16	179	1.2826	0.77966	18.7346	12.9762	0.7332	0.7332
		32x32	717	0.87971	1.1367	21.3439	12.573	10.6861	10.5925
	0.8	8x8	51	1.5888	0.62941	21.4809	12.9512	0.1248	0.1448
		16x16	205	1.2023	0.83172	20.8318	14.299	0.85801	0.88921
		32x32	819	0.79355	1.2602	23.6184	14.6515	12.3709	12.3709
	0.9	8x8	58	1.5213	0.65735	23.2653	16.4741	0.234	0.1872
		16x16	231	1.1223	0.89104	24.8832	17.5529	0.98281	0.95161
		32x32	922	0.73934	1.3525	27.6197	18.1877	15.2101	15.1945

Table I: Results of PSNR and Rate of compression

Image Name	Sample Rate	Image size	M	Compression Ratio	Rate of compression	PSNR		Decompression time	
						BP	L1	BP	L1
CAMERAMEN	0.3	8x8	19	1.8021	0.5549	12.4273	7.1474	0.1716	0.156
		16x16	77	1.7613	0.56777	16.2968	7.2517	1.1544	1.1544
		32x32	307	1.5293	0.65391	16.9359	7.1034	10.4521	10.4209
	0.4	8x8	26	1.7468	0.57247	13.0278	7.727	0.2184	0.234
		16x16	103	1.56	0.64103	17.758	7.9666	1.326	1.326
		32x32	410	1.2749	0.78438	17.6717	7.8038	12.4021	12.3709
	0.5	8x8	32	1.7204	0.58126	16.6175	9.8533	0.234	0.2184
		16x16	128	1.4864	0.67277	17.7827	8.4881	1.17	1.17
		32x32	512	1.0884	0.91875	18.9389	8.8861	14.9293	14.8825
	0.6	8x8	39	1.6823	0.59444	17.0048	11.1175	0.2028	0.1716
		16x16	154	1.3515	0.73993	19.0737	9.2066	1.4352	1.4196
		32x32	615	0.94395	1.0594	20.106	9.5982	17.5813	17.5657
	0.7	8x8	45	1.5995	0.62518	17.8345	14.8765	0.2808	0.1872
		16x16	179	1.3021	0.76801	20.851	10.3171	1.6692	1.6848
		32x32	717	0.82315	1.2148	21.5549	11.2375	20.7169	20.6701
	0.8	8x8	51	1.5701	0.6369	20.2122	13.1436	0.3588	0.3432
		16x16	205	1.1956	0.83639	22.2294	12.6275	1.71	1.7004
		32x32	819	0.74375	1.3445	23.8745	12.5803	24.1334	24.2114
	0.9	8x8	58	1.5314	0.653	24.9998	13.9684	0.3276	0.3588
		16x16	231	1.1265	0.88767	25.4287	16.1901	1.7628	1.7628
		32x32	922	0.70562	1.4172	27.683	15.5661	26.7698	26.66.6

Table II: Results of PSNR and Rate of compression

Image Name (256x256)	Measurement Reduction M	PSNR (dB)	RC	Processing Time(sec)	Time to reconstruct (sec)
Cameraman	500	20.9208	1.26	0.06703	13.4369
	1000	22.847	1.06	0.1144	16.1084
	1500	29.4725	1.25	0.1504	23.1886
Lena	500	21.0436	0.76784	0.0543	12.0625
	1000	24.355	0.98106	0.0649	16.7437
	1500	30.7185	1.1116	0.0751	24.5863

Table III: Results of PSNR and Rate of compression

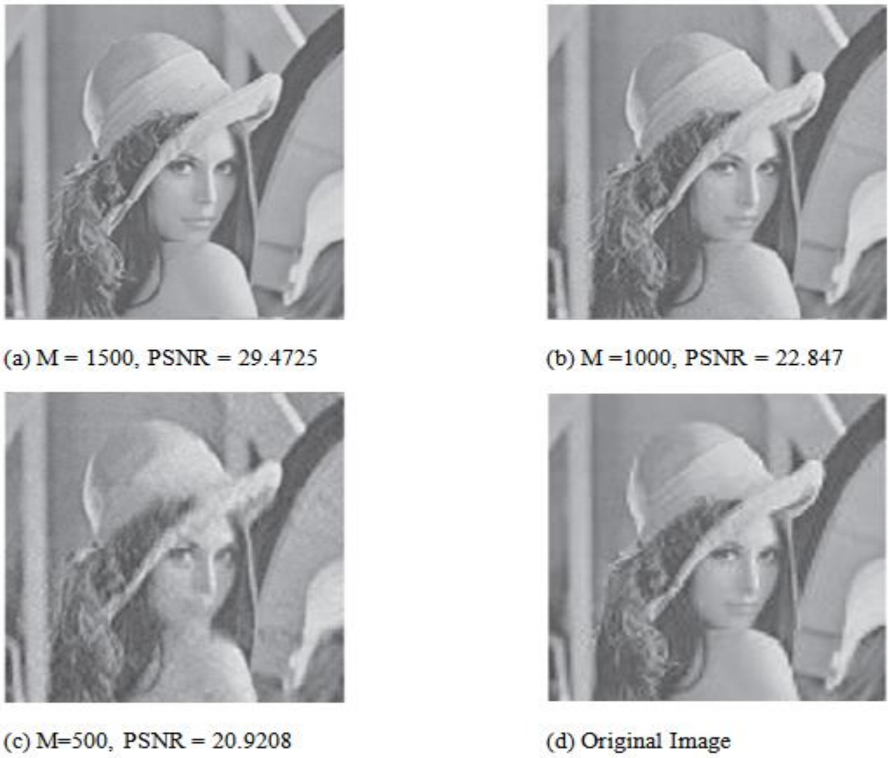

Fig..1: Recovery of Lena by compressive sensing

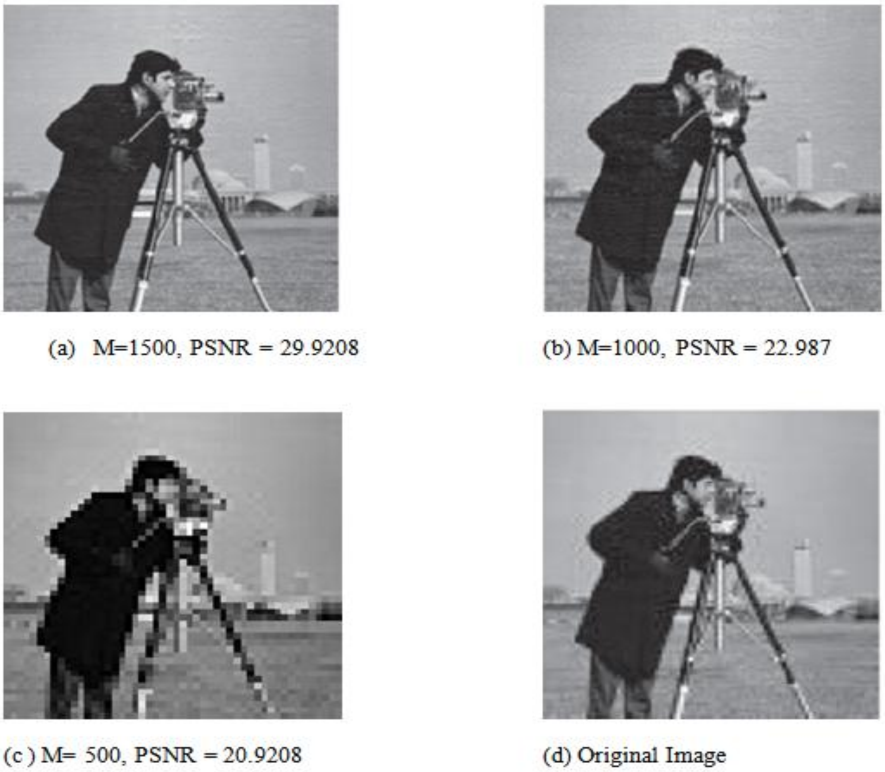

Fig 2: Recovery of Cameraman by Compressive Sensing

V. CONCLUSIONS

In this paper, we presented the implementation of Compressive sensing (CS) in cloud storage environment. After testing CS on different files of different sizes, it shows significant storage savings and a large M means that a lot of coefficients are captured; this results the high compression rate and high quality of reconstruction of data. We used both Basis Pursuit (BP) and L1-reularised least squares optimizer to perform the optimization step in CS and it is observed that BP gave better performance with respect to both PSNR and the time taken. We can conclude that compressive sensing can be used in the cloud environment to compress the data during storage so that the data transfer time and storage reduced subsequently. We presented an analysis with practical results for text and image files.

References

[1]. M. Elad, ―Optimized Projections for Compressed Sensing‖, IEEE Transactions on Signal Processing, Vol. 52, (2007).
[2]. Shuhui Bu, Zhenbao Liu, Tsuyoshi Shiina, Kazuhiko Fukutani, Matrix Compression and Compressed Sensing Reconstruction for Photoacoustic Tomography, Elektronikairelektrotechnika, Vol 18, No 9 (2012)
[3]. E. Candes, J. Romberg, and T. Tao, "*Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information*," *IEEE Transaction on Information Theory*, vol. 52, pp. 489–509, (2006).
[4]. D.L. Donoho, "Compressed Sensing," *IEEE Transactions on Information Theory*, vol. 52, pp.1289-1306, (2006).
[5]. E. Candes and J. Romberg , "Practical Signal Recovery from Random Projections," *Processing SPIE International Symposium Electronic Imaging*, pp. 76–86, vol. 5674, (2005).
[6]. Baraniuk, R.: Compressive Sensing [Lecture Notes]. IEEE Signal Processing Magazine 24(4), 118-121,(2007).
[7]. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic Decomposition by Basis Pursuit. SIAM Rev.43(1),129-159, (2001)
[8]. Tibshirani, R.: Regression Shrinkage and Selection Via the Lasso. Journal of the Royal Statistical Society, Series B 58, 267-288 (1996).
[9]. Efron, B., Hastie, T., Johnstone, I., Tibshirani, R., Ishwaran, H., Knight, K.,Loubes, J.M., Massart, P., Madigan, D., Ridgeway, G., Rosset, S., Zhu, J.I.,Stine, R.a., Turlach, B.a., Weisberg, S., Hastie, T., Johnstone, I., Tibshirani, R.: Least angle regression. Annals of Statistics 32(2), 407-499, (2004).
[10]. Mallat, S.G., Zhang, Z.: Matching pursuits with time-frequency dictionaries. IEEE Transactions on Signal Processing 41(12), 3397-3415, (1993).
[11]. Tropp, J.A., Gilbert, A.C.: Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit. IEEE Transactions on Information Theory 53(12), 4655-4666, (2007).
[12]. Dai, W., Milenkovic, O.: Subspace pursuit for compressive sensing signal reconstruction. IEEE Transactions on Information Theory 55(5), 2230 -2249, (2009).
[13]. Donoho, D.L., Tsaig, Y., Drori, I., Starck, J.L.: Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit. IEEE Transactions on Information Theory 58(2), 1094-1121, (2012).

M.TanoojKumar ."Cloud Storage Optimization Approach Using Compressive Sensing" International Journal Of Engineering Research And Development , vol. 14, no. 02, 2018, pp. 21–26.