J. Integr. Sci. Technol. 2023, 11(2), 487 Article

Journal of Integrated
SCIENCE & TECHNOLOGY

Deep compressive sensing and reconstruction algorithm in wireless Internet of Things

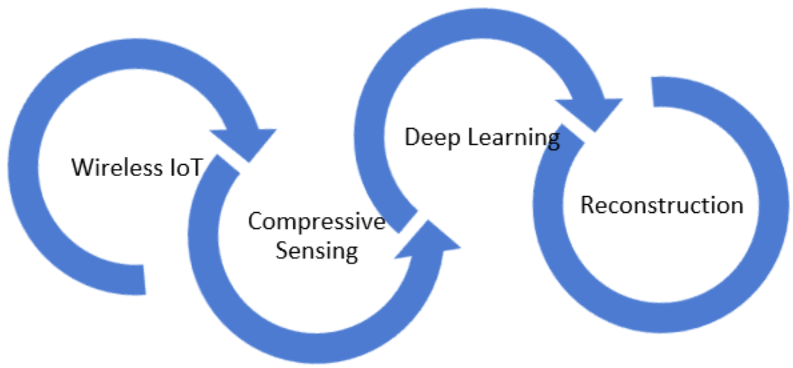
Hemant Rajoriya*, Ritesh Sadiwala

Department of Electronics and Communication, RKDF University, Bhopal, India.

Submitted on: 22-Oct-2022, Accepted and Published on: 03-Dec-2022

ABSTRACT

The wireless Internet of Things (IoT), in particular its implementation in the field, is gaining growing significance for the communication systems of the future. However, wearable gadgets normally have limited energy consumption so that the battery can last for a longer period of time. On the other hand, it has been demonstrated that compressed sensing (CS), which uses less power than traditional transform-coding-based techniques, is more efficient. Because particular transform domains, such as the discrete cosine transform, reveal specific traits that are closely related with one another in the spatial and temporal data that are collected by wireless Internet of Things sensors. Because the spatial and temporal data collected by a wireless IoT have some closely correlated structures in certain wavelet domains, such as the discrete wavelet transform (DWT) domain, we propose a new low-rank sparse deep signal recovery algorithm for recovering data in the context of compressed sensing (CS). This algorithm is designed to recover data in the context of compressed sensing. For the purpose of implementing and simulating the sparse signal deep compressed sensing (DCS) recovery technique, the simulation was carried out using the MATLAB platform. The model that is described in this work demonstrates an MSE that falls somewhere in the region of -30 to -40 dB, and the comparison analysis arrives at the conclusion that the proposed DCS algorithm is more effective than the work that has already been done.



Keywords: Wireless Network, Internet of Things (IoT), Lossy Channel, Compressed Sensing, Deep Learning

INTRODUCTION

The Internet of Things (IoT) materializes the concept of integrating real-world things with digital versions, blurring the distinction between the actual and virtual realities. Smart devices that could be implanted with software, applications, and techniques are referred to as "things." These things are frequently made up of smaller appliances with computation and storage capabilities that enable them to conduct activities and interact with other things and systems without the need for direct user participation. IoT facilitates the establishment of a new enlarged Internet by enabling things to interconnect their facilities, assets, and intellect to one another.[1,2]

The enormous amount of information created by IoT devices necessitates significant warehousing and transport expenses.[3] Communication networks are important infrastructures for IoT units, and even with the advent of low-power networking,[4] these frameworks are accountable for the considerable electricity usage of equipment. Because of the energy constraints of certain IoT devices, this usage becomes a concern that should be addressed.[5] Handling a huge dataset necessitates significantly more evaluating services and processing duration. To complicate things further, the vast amount of monitored information that must be retained in fogs and clouds imposes a substantial monetary price, as these technologies strive to charge proportionate amounts for the information retained.[6]

These concerns could be handled from the standpoint of equipment or connection, like as by adopting an appropriate

*Corresponding Author: Hemant Rajoriya, Department of Electronics and Communication, RKDF University, Bhopal, India.
email: hemantrajoriya25@gmail.com

Cite as: *J. Integr. Sci. Technol., 2023, 11(2), 487. URN:NBN:sciencein.jist.2023.v11.487*

ISSN: 2321-4635
http://pubs.thesciencein.org/jist

interaction method or enhancing bandwidth of the networking or fog and cloud storing capabilities. The most prevalent strategy to dealing with these IoT difficulties, though, is from a semantic-oriented standpoint,[7] which covers issues relevant to the optimal methods to describe, retain, and organise the data. In this environment, data compression (DC) technologies serve a crucial part in addressing communications network and accessible storing needs, and have evolved into basic techniques to handling vast amounts of produced data.

Lossless algorithms are those that can recover a signal completely as it was originally represented. Lossy algorithms, on the contrary side, can just rebuild an approximations of the actual signal,[8] allowing them to depict the signals with lesser specimens than lossless algorithms. Sensors data are typically recorded in lossy ways by removing duplicate specimens or displaying them in a compact form, with the assumption that the duplicate specimens would not contribute extra data for the app.[9]

The emphasis on lossless or lossy categorization is due to the fact that sampled IoT sensor information is frequently utilized to depict fluctuating signals,[10] and it is critical to handle DC approaches, assessing the generated signal after the compressing and decompressing procedures. The reasoning is that following the decompressing procedure, the resulting signal will be utilized for analyses (e.g., in machine learning techniques) or judgement call (e.g., to trigger alerts or regulate actuators) in different devices. There are several motives to consider lossy approaches. One of these is that they provide outcomes identical to lossless DC while using significantly less difficult methods with a higher compression ratio. Lossy algorithms can produce comparable outcomes with less specimens than lossless algorithms. For such causes, lossy DC approaches are ideal solutions for IoT equipment, enhancing power, storage, and communications resources handling.

LITERATURE REVIEW

Liang et al[11] reported the compacted signal using transfer learning, a convolution-based transfer learning CS (CTCS) framework. To assess CTCS functionality, an ultrawide band (UWB) radar echo signal and a Mnist hand-written information collection are used. Under diverse intensities of noise, measurements amounts, and signal sparsities, the suggested framework outperformed other established reconstructive techniques in 6G-IoT. Xue et al.[12] presented Kryptein, a compact encrypting technique built on compression sensing for Cloud-enabled IoT frameworks, to protect the connection between IoT gadgets and the Web. Zhang et al.[13] created a two-stage compacted information aggregating approach by combining compression sensing and a sparse autoencoder. A deep compressive sensing network (DCSNet) is meant to recover signals from compressed data utilizing a deep learning algorithm. A Compressed Sensing with Dynamic Retransmission (CSDR) technique is given in the research by Jiang et al.[14] to ensure high information reconstructive precision, long network lifespan, and efficient power consumption. The CSDR method dynamically estimates the maximum packet loss resend durations of various nodes based on their remaining energy for Internet of Things (IoT) equipment with relatively high power utilization, less max resend times are used to retain a prolonged lifespan of the network. More max resend durations are employed for energy-efficient IoT systems to optimize data transfer precision and information reconstructive quality. Prabha et al.[15] proposed a unique framework that combines grouping and compressive sensing (CS) by utilizing Block Tri-Diagonal Matrices (BDM). BDMs are measuring matrices that use grouped WSNs to generate precision and effective information refining by combining compaction, information forecasting, and retrieval. Theoretical examination served as the foundation for the development of several techniques for implementation. For simulations, real-world data were utilized, and the presented findings demonstrated that the architecture explained here gives a cost-effective alternative for apps that supervise the atmosphere in clustered WSNs. The presented IHCS accomplishes 70% energy efficiency and 93% forecasting rate. Sun et al[16] provided a Sensing Cloud-computing-based Compressed Sensing Routingcontrol-method with Intelligent Migration-mechanism (CSR-IM). Initially, the approach provides a technique for estimating the targeted node's movement speed and location using compressed sensing theories, while also providing a lower limit computation procedure of the targeted node's state estimating values at k + 1 time using probabilistic information. Furthermore, in order to reduce network load, a navigating tree with the center of fog nodes is constructed in order to efficiently gather information in the pathway and improve the information gathering navigating procedure, and the electricity expense of the entire network is then managed. Bose et al.[17] analyzed and categorized lossy compression techniques relying on sensor data signal features. The algorithms are classified according on their time and transformation areas. Tuama et al.[18] provided a summary of current DC improvements in WSN. The techniques are divided into global and localized techniques, which are further subdivided into lossy and lossless methods. Only two of the 16 publications in the review are addressing lossy DC approaches. Uthayakumar et al. [19] presented a study of DC approaches from the viewpoints of information reliability, coding schemes, kind of information, and uses. The essay focuses on WSN implementations and discusses only one lossy DC approach. A architecture that chooses between a lossless or lossy compression approach based on the power availability and the information's importance is proposed by Mohamed et al. in their article reported[20] that Information integrity and power conservation are optimized by the application of Markov Decision Process (MDP). To determine whether the information is vital (for lossless transmitting), activity identification is employed (compressed). For lossless DC, an entropy encoder is employed. The concept employs an error radius to specify the permissible level of inaccuracy. It functions as a fault bounder to determine when to encrypt and send a specimen. Giorgi[21] presented a method for zero-latency compression that integrates lossy and lossless methods. Initially, a zero-latency forecasting filter called lossy compression using differential pulse code modulation is used (DPCM). When passing the specimen to the lossless approach becomes essential, it applies a tolerance limit. The altered Exponential Golomb code used by the lossless method has a fixed-length prefix. Alsalaet and Ali[22] presented MDCT-EHCC, a DC method built on Modified Discrete Cosine Transform (MDCT) accompanied by Embedded Harmonic Components Coding (EHCC). The lossy approach used in the approach is MDCT. MDCT coefficients are used to describe an N-sample signal. EHCC is also utilized to optimize the CR. EHCC enables integrated coding, which enables the coefficients to be transmitted successively based on their relevance. This proposal's goal is to investigate harmonic redundancies.

Problem Identification

Information losses in wireless sensor networks are ubiquitous and have distinct trends owing to interference, collisions, faulty links, and unforeseen destruction, which significantly lowers restoration precision. Current interpolation algorithms do not take these trends into account and, as a result, failed to give sufficient precision when misplaced information becomes substantial.

A significant issue impacting the lifespan of wireless sensor networks is power utilization. A variety of methods, including navigation algorithms and energy-efficient media accessibility regulation, have been suggested to address this problem. The information compressing system is one of the suggested methods that may be utilized to lessen the amount of information delivered through wireless networks. The principal energy consumption in wireless sensor networks, inter-node connectivity, is reduced as a result of this technology.

Overview of Compressive Sensing

One of the crucial elements in creating forthcoming internet of things infrastructures is wireless sensor network (WSN) technologies.[23] Since smart sensing devices have started to play a significant role in our everyday activities, it has attracted a lot of interest. The storing capabilities, energy capacities, and computing capabilities of these gadgets are all limited in reality. The processing of large amounts of information, particularly video information, is becoming increasingly difficult as a result. In order to relocate the computation responsibilities from the sensor level to the decoder in WSN, compressive sensing is employed as an efficient method for reducing the ambiguity of the encoder, which implies that by improving the manner the gadgets procure and transfer information over wireless links, researchers improved the computational tools of the gadgets and improve their effectiveness. In reality, the compressive sensing method considerably improves the coding performance of wireless equipment by lowering the sampling frequency and synchronization of the information sequencing procedure. Additional issue that may be identified from a macro viewpoint in WSN systems is the irregular (rare) transmitting rates. However, it's not like all wireless sensors transmit their information to the centralized server at the exact same time, implying that the WSN architectural sparsity must be employed to provide high information dependability with a smaller amount of sensors. Furthermore, since numerous real-world databases can be effectively represented by weak signals employing a suitable transformation, Information systems can readily combine compressive sensing into their various implementations. As a result, in several WSN implementations, power utilization is a major issue since sensors must transmit sensed information to the coordination node on a regular basis. The compression sensing mechanism is depicted in Figure 1 below.

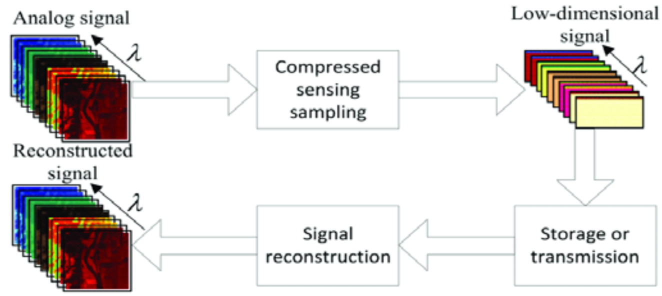


Figure 1. Compression Sensing Process

Since information delivery is thought to be the main cause of power utilization, a lot of research is being carried out to find ways to capture less information when sensors are used. One must compress the information within the network in order to limit the amounts being transmitted. As a consequence, novel approaches to constructing energy-efficient WSN with affordable information gathering have been made possible using compressive sensing (CS) algorithms.[24]

Traditional sensors are centered on the Shannon-Nyquist sampling hypothesis, which is centered on the idea that the lowest sampling rate of the signals must be twice that of its maximum carrier frequency in order to preserve its background data. For implementations that need a lot of information, this hypothesis has become obsolete since it mandates an excessively high sampling frequency. As a result, the Compressive Sensing concept aims to reduce the rates of the Shannon-Nyquist principle while also fulfilling the demands of the massive data-intensive apps. For the sake of simplicity, a CS camera for this application case records a variety of measures from the view that are coded at a scale significantly less than the overall amount of rebuilt pixels. In actuality, CS is a method that makes it possible to efficiently acquire sparse signals by performing both recognition and compression during the same instant. Conventional sampling and CS sampling strategies are contrasted in Figure 2.

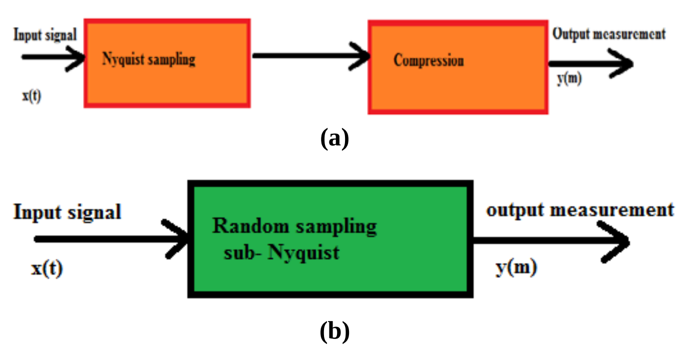


Figure 2. A Comparison of Sampling Techniques: (a) Traditional Sampling, (b) Compressive Sensing [25]

Below are a few foundational elements to remember in order to comprehend the mathematics underlying the CS approach: Rather than collecting N specimens of a signal x ∈ RN×1, M randomized measurements are collected with M N (the CS theory stipulates that M = O(Klog(N/K)) is the amount of measures required to rebuild the signal x), such that:

$$y = \varphi x_o \tag{1}$$

where y ∈ RM×1 is the known compressed measuring vector and $\boldsymbol{\varphi}$ ∈ RM×N is the sensing matrix detailed in the following subsection To retrieve the signal x given y and $\boldsymbol{\varphi}$, $\boldsymbol{x_o}$ should be sparse in a particular base Ψ:

$$\boldsymbol{x = \Psi s} \tag{2}$$

Where s is K-sparse, implying that it contains no more than K non-zero elements.

$$\boldsymbol{y = As} \tag{3}$$

Where $A = \Phi\Psi$ is an integer and the compressed sensing infrastructure is depicted in Figure 1.

Nevertheless, it is not feasible to rebuild x or s from y. As a result, addressing the underlying l_1 minimization issue[26] yields an approximated alternative:

$$s\hat{} = \text{argmin}\|s\|1 \text{ s.t. } y = \Phi\Psi s \tag{4}$$

CS algorithms employ several reconstructing methodologies to recreate s from y. Then, from given xˆ = Ψsˆ, x could be rebuilt.

Reconstruction Algorithms

The rebuilding procedure is essential for effectively incorporating compressive sensing in practical uses. As a result, the primary priority of CS investigators is the development and implementation of novel optimizing methods. There are various types of algorithms. In this section, we will look at the two basic forms of restoration techniques in CS that is convex optimization methods and greedy algorithms.

Greedy algorithms are extensively utilized in computer science implementations due to their low intricacy and quick rebuilding. The much more widely used greedy algorithms are presently divided into sequential and parallel greedy pursuit strategies. Gradient pursuit,[29] matching pursuit (MP), orthogonal matching pursuits (OMP), and orthogonal multiple matching pursuit.[30]

Methodology

By utilizing the idea of sparsely, Compressed Sensing evolves into a cutting-edge framework for signal gathering. CS demonstrates how several data points required can be significantly reduced whenever a signal is sparse (fragmented) or compressed in some manner as opposed to with the standard collecting data through compression as well as sampling process technique. In this instance, CS gathers the data directly, compressively, and slowly. It is to be supposed that the signal is k-sparse $x = (x_1, x_2, \ldots \ldots .., x_N)^t$, based on the ψ that it is depicted as

$$\boldsymbol{x = \psi S} \tag{5}$$

At this point, $\|S_0\| = \#\,\{i: S_i \neq 0\} = k \ll N$.

$y = (y_1, y_2, \ldots . y_N)^T$ is the estimation vector; alternatively, the internal product of x with a certain function constitutes the CS sampling product.

That seems to be

$$\boldsymbol{y = \varphi x = \varphi\psi S} \tag{6}$$

where $M = O(k \cdot \log(\frac{N}{k}))N$,

The measuring matrix is composed of predetermined projecting vectors (1). It is clear that M N implies a reduction in element in compared to the Nyquist sampling theorem, and that CS denotes a decreased sampling frequency. The measuring matrix () consists of projections vectors that have been predetermined. It is clear that mN represents a reduction in dimension, and that Compressed Sensing symbolizes a decreased sample frequency. To sum up, the ancient technique of high-frequency sampling which employs Richardson concept after then image compression employing the sparse conversion has been replaced in the CS procurement process by a straightforward low-rate linear prognosis. As an outcome, it is possible to reduce system complexity and improve energy efficiency. The Nyquist sampling theorem uses linear interpolated to restore signals, which is not compatible with the CS design, which is characterized by a linear system that is simple to understand. Utilizing the sparsity constraint, the intrinsic nonlinear approach resolves the following optimization problem.

$$\boldsymbol{min\|s\|_0 \; subject\; to\; y = \varphi\psi S = \odot S} \tag{7}$$

Then, using x= S, the real signal x is recovered. It is found that solving equation (4.16) requires exhaustively exploring all k sub-columns from N ones and is non-deterministic polynomial-time (NP)-hard for signals of normalized size. To guarantee that there are different as well as continuous solutions towards this issue of optimization, The sensing matrix ⊙ should adhere to the ordering 2k restricted isometric property (RIP).

Deep Compressive Estimation Scheme

This section describes the proposed deep compressive estimate (DCE) algorithm. According to the proposed method, the sensor first detects the d1 vector (x k)(i) at each node before estimating _0 with the aid of the dM sample covariance matrix. Figure 3 shows the proposed deep compressive estimator design. Figure 4 presents the flowchart of the designed compressive model.

In other words, the proposed method predicts the d1 vector (0). Use _0 as opposed to M1 as the vector, where dM and d-dimensional values are indicated by an over bar. Deep estimator _k (I) and reconstructing techniques are used in a decompression process to estimate each node's 0. Less communication of variables is needed between network nodes when using the deep compressive estimator technique. The description of the recommended deep compressive estimator technique starts with the scalar evaluation D k I given by.

$$\boldsymbol{D_k(i) = \overline{\omega_0^h x_k}(i) + n_k(i)}\text{where i = 1, 2 , and I} \tag{8}$$

The input signal vector for d1 is _0 = _k _0 and (x k) (i).

There are three steps in proposed deep compressive sensing is:

- Adaptability
- Data Exchange
- Merger

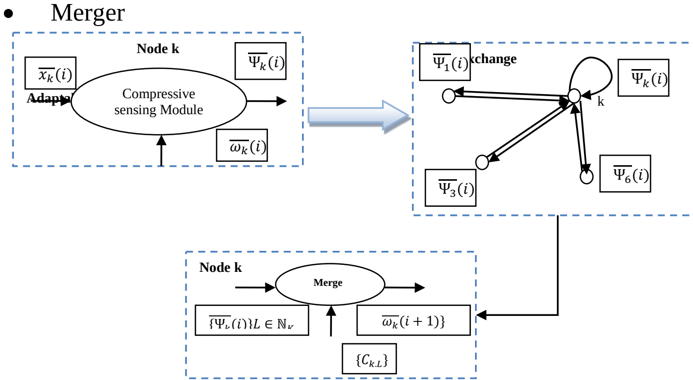


Figure 3. Proposed Deep Compressive Estimation Scheme Adaptability: Each node in k = 1, 2,... N at each and every time frame i = 1, 2,...I, in the adaptation phase.

Data Interchange: Depending on the network actually way, only the localized compressing estimator (_k) I will be shared among node k and all of its neighboring nodes. The measured matrix _k will be kept locally.

Merger: The combination phase began after the data sharing was finished at each instant I = 1, 2,..., I. In figuring out the latest compression estimation as (_k) (i+1), each node will aggregate the locally compression estimation technique from its neighboring nodes and itself. This can be used in conjunction with other reconstructive methods.

Convolution neural network technology is used to create the resilient system (CNN). Residual learning is the suggested strategy in this work. For the function of convolutional neural loss:

$$\boldsymbol{C}(\Theta) = \frac{\mathbf{1}}{\mathbf{2N}} \sum_{i=1}^{N} ||\boldsymbol{R}(\boldsymbol{y}_i; \Theta), \boldsymbol{y}_i - \boldsymbol{x}_i||_F^2 \qquad (9)$$

Where,
N= number of input training images.
y_i = Compressive data
x_i = Raw data
R= Residual learning function
ϴ re= Input parameters for CNN.

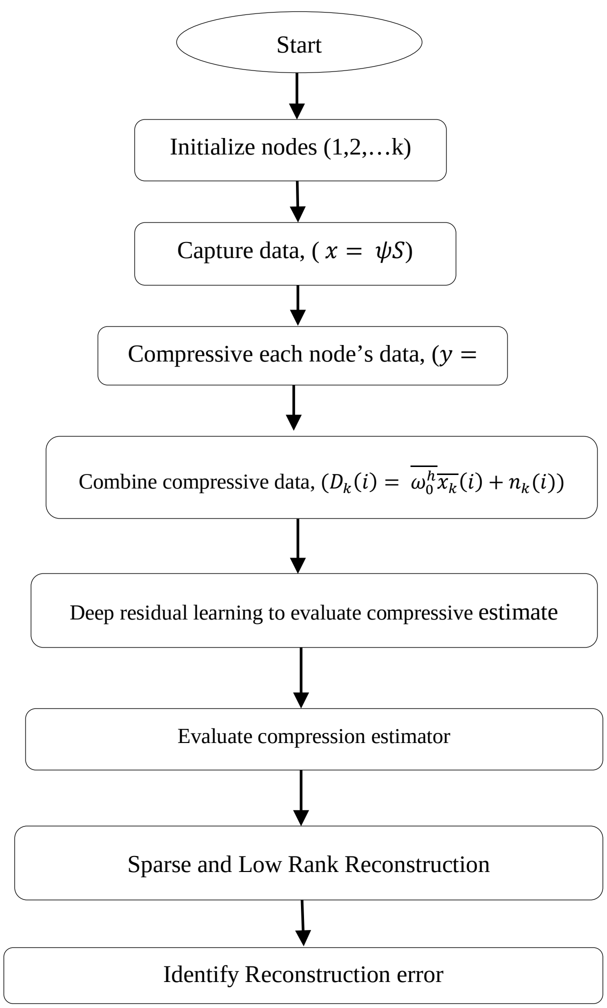


Figure 4. Flowchart of Proposed Deep Compressive Sensing

Proposed CNN Architecture: In this work, CNN layers designed as below:

- The pairing of Conv and PReLU makes up the initial layer of the envisaged CNN structure. There are 64 feature mappings with convolution filters collected in this layer of size $3 \times 3 \times c$.
  Where, c = The number of image channel. $(c = 1)$ $for\ 2D\ or\ (c = 3) for\ 3D\ data\ matrix.$
- For non-linear function PReLU activation function is used.

Sparse and Low Rank Reconstruction

Most conventional signal CS methods disregard noise. In reality, noise will always have an impact on compressive signals and will eventually reduce the effectiveness of multichannel signal CS. Therefore, taking noise into account is crucial. Noise is frequently brought on by missing data during measurement, communication problems, buffer overruns, and incorrect storage addresses. When noise enters the multichannel signal Compressed Signal system at the sender side, the compressive multidimensional signal will get distorted.

Algorithm 1: Deep CS Methodology

Initialize $\overline{\omega_k}(1) = 0\ , fork = 1,2, \ldots \ldots, p$
For every time interval i= 1,2,…., I-1
For every node k = 1,2,…,p
$\overline{\Psi_k}(i) = \overline{\omega_k}(i) + \mu_k(i) e_k^*(i) \overline{x_k}(i)$
Where, $e_k(i) = D_k(i) - \overline{\omega_0^h}(i) \overline{x_k}(i)$
For every node k = 1,2…, p
$\overline{\omega_k}(i+1) = \sum_{L \in \mathbb{N}_K} c_{kL} \overline{\Psi_k}(i)$
End
End
Following the final iteration, I
For every node k = 1,2…, p
$\omega_k(I) = f_{dl} \omega_k(I)$ Percentage deep compression of the estimation
where ω_k is the complete deep compressive estimating system.
end

Results and Discussions

In this paper following performance parameters are used:
Compression Ratio (CR): This is defined as ratio between the raw and compressed data sizes. Mathematically it is represented as:

$$\boldsymbol{CR} = \mathbf{1} - \frac{\boldsymbol{M}}{\boldsymbol{N}} \qquad (10)$$

Where, M = raw data and N= compressed data.

It is obvious that CR is a floating number less than 1, while larger CR means that less CS measurements are acquired and thus more plaintext information has been compressed.

Percentage root-mean-squared difference (PRD): It is employed to numerically measure the distortion between the reconstruction signal x' with original signal x, that is

$$\boldsymbol{PRD} = \frac{\|\boldsymbol{x}' - \boldsymbol{x}\|}{\boldsymbol{x}} * \mathbf{100} \qquad (11)$$

Mean Square Error (MSE): Mean square error is the average square of the error of the reconstructed bits as compared to the transmitted bits. The error is calculated as the difference between actual values and estimated values.

$$\boldsymbol{MSE} = \frac{|\boldsymbol{E}_v - \boldsymbol{A}_v|^2}{\boldsymbol{N}} \qquad (12)$$

Where, E_v = Estimated value
A_v = Actual value
N = Number of bits
In terms of decibel, it is represented as:

$$MSE\ (in\ dB) = 10 \log_{10} MSE \tag{13}$$

Result Analysis of Proposed Model

The result analysis is performed by simulating the scenario on MATLAB platform. In this paper, two result analysis is performed, i.e., on the basis of lossy environment and on the basis of variable compression ratio. Figure 3 shows PRD analysis with variable SNR in which SNR rate increases with PRD value and also shows that when PRD value is 11.51871672 with SNR value is 10db, 15.33652093 with SNR value is 20db and 19.51242083 with SNR value is 30db. Figure 4 shows MSE analysis with variable SNR in which SNR rate decreases with MSE value and also shows that when MSE value is -41.20924411 with SNR value is 10db, -37.16858462 with SNR value is 20db and -34.52645387 with SNR value is 30db. Figure 5 shows PRD analysis with Respect to Compression Ratio where PRD value are 17.89, 17.77, 19.57, 18.79, 20.46, 19.68, 20.79, 21.39, 22.11 with Compression Ratio 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% respectively. Figure 6 shows MSE analysis with Respect to Compression Ratio where MSE value are -38.09, -34.59, -33.97, -35.16, -33.57, -33.81, -34.88, -34.40, -34.31 with Compression Ratio 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% respectively. Figure 7 shows PRD comparison with state-of-art-model like DeepCS, OMP, BP, COSAMP, IRLS, LMS and NLMS. In which the proposed model achieved better PRD as compared to others. Figure 8 shows MSE comparison with state-of-art-model like DeepCS, OMP, BP, COSAMP, IRLS, LMS and NLMS. In which the proposed model achieved better MSE as compared to others.

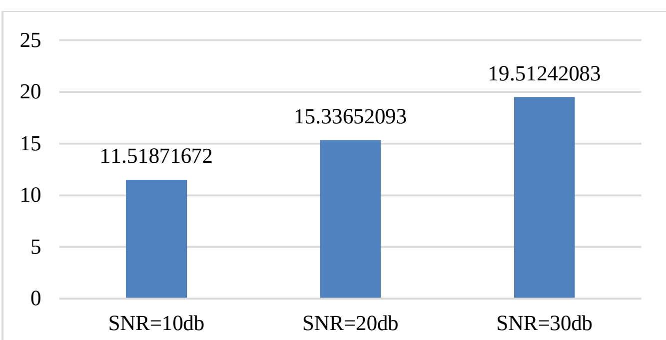


Figure 3. PRD analysis with variable SNR

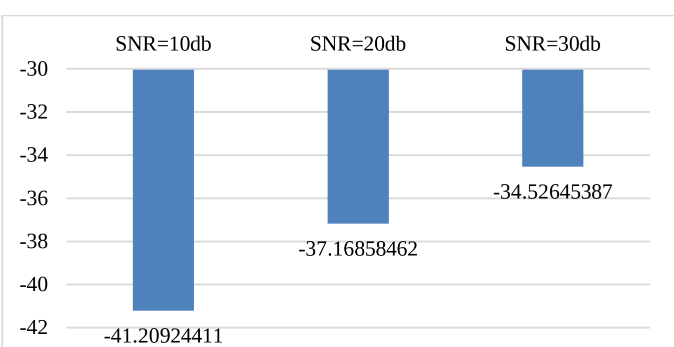


Figure 4. MSE analysis with variable SNR

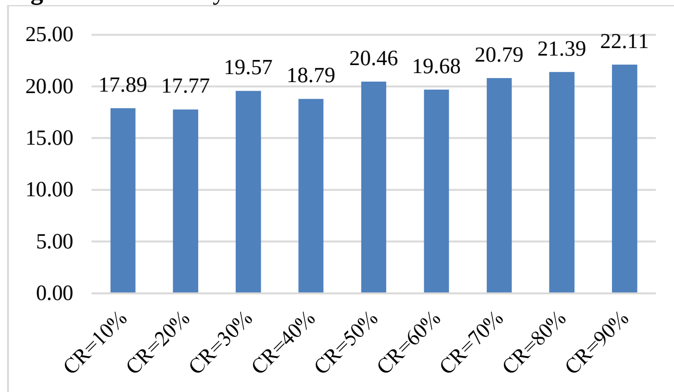


Figure 5. PRD Analysis with Respect to Compression Ratio

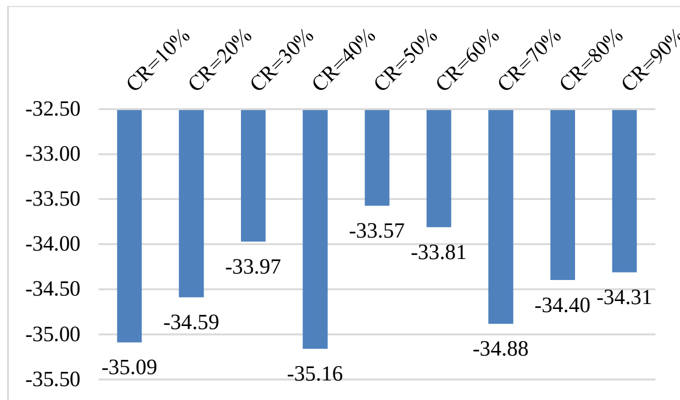


Figure 6. MSE Analysis with Respect to Compression Ratio

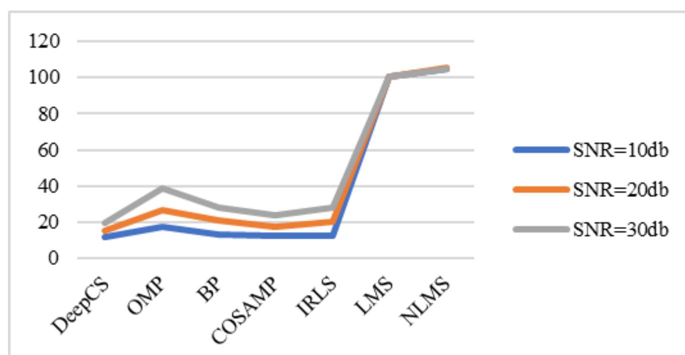


Figure 7. Comparison of PRD with State-of-Art Models

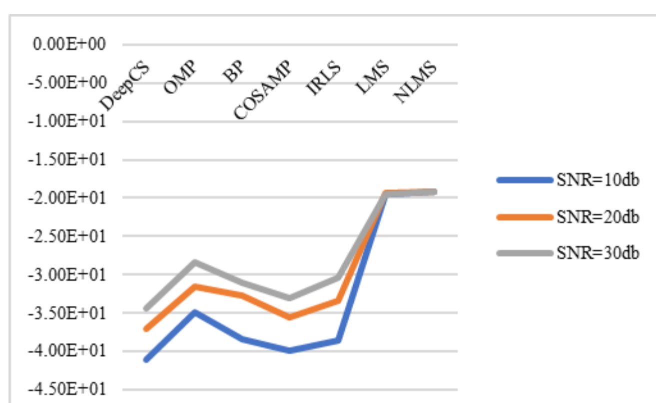


Figure 8. Comparison of MSE with State-of-Art Models

Comparative State-of-art

Table 1 shows the comparative analysis with existing work. In this table, MSE of proposed model and existing model[31] is presented and it was found that proposed algorithm have achieved better performance as compared to state-of-art model.

Table 1. Comparative State-of-Art

MSE	DeepCS	L&S [31]
SNR =10db	-41	-7
SNR =20db	-37	-14
SNR =30db	-34	-18

CONCLUSION

Recent research has demonstrated that compressive sensing, also known as CS, is an effective method for the data compression of wireless Internet of Things networks. In order to overcome the challenges outlined above, the authors of this paper suggest a deep learning-based sparse and low rank representation that operates in the presence of noise. The findings of the simulations show that the deep CS given has an edge over advanced noise methods. Several distinct situations are simulated in order to get an accurate result. The simulation was run with varying CR and SNR values for wireless IoT nodes. Additionally, the SNR values were varied. Based on the findings, it has been determined that the SNR must be increased before the PDR can rise. The MSE was determined to be between -30 and -40 db, and it appears to be getting worse. In order to facilitate comparison, this study uses a variety of statistical techniques, including basis pursuit (BP), compressive sampling matching pursuit (COSAMP), iteratively reweighted least squares (IRLS), IRLS, and OMP. In the final step, the performance of the suggested deepCS is evaluated and compared with that of previously published studies. The MSE evaluation of suggested work varies from -30 to -40, which indicates that the proposed work will be more efficient than the existing work.

REFERENCES

1. L. Coetzee and J. Eksteen. The Internet of Things - promise for the future? An introduction. IST-Africa Conference Proceedings, **2011**, 1-9.
2. C. Perera, A. Zaslavsky, P. Christen, D. Georgakopoulos. Context aware computing for the internet of things: A survey. *IEEE Commun. Surv. Tutorials* **2014**, 16 (1), 414–454.
3. A.R. Biswas, R. Giaffreda. IoT and cloud convergence: Opportunities and challenges. *2014 IEEE World Forum Internet Things, WF-IoT 2014* **2014**, 375–376.
4. G.G.M. De Jesus, R.D. Souza, C. Montez, A. Hoeller. Lorawan adaptive data rate with flexible link margin. *IEEE Internet Things J.* **2021**, 8 (7), 6053–6061.
5. J. Azar, A. Makhoul, M. Barhamgi, R. Couturier. An energy efficient IoT data compression approach for edge machine learning. *Futur. Gener. Comput. Syst.* **2019**, 96, 168–175.
6. S.K. Apat, J. Mishra, K.S. Raju, N. Padhy. The robust and efficient Machine learning model for smart farming decisions and allied intelligent agriculture decisions. *J. Integr. Sci. Technol.* **2022**, 10 (2), 139–155.
7. L. Atzori, A. Iera, G. Morabito. The Internet of Things: A survey. *Comput. Networks* **2010**, 54 (15), 2787–2805.
8. U. Jayasankar, V. Thirumal, D. Ponnurangam. A survey on data compression techniques: From the perspective of data quality, coding schemes, data type and applications. *J. King Saud Univ. - Comput. Inf. Sci.* **2021**, 33 (2), 119–140.
9. S. Pattar, R. Buyya, K.R. Venugopal, S.S. Iyengar, L.M. Patnaik. Searching for the IoT resources: Fundamentals, requirements, comprehensive review, and future directions. *IEEE Commun. Surv. Tutorials* **2018**, 20 (3), 2101–2132.
10. E. Moghadas, J. Rezazadeh, R. Farahbakhsh. An IoT patient monitoring based on fog computing and data mining: Cardiac arrhythmia usecase. *Internet of Things* **2020**, 11, 100251.
11. J. Liang, L. Li, C. Zhao. A Transfer Learning Approach for Compressed Sensing in 6G-IoT. *IEEE Internet Things J.* **2021**, 8 (20), 15276–15283.
12. W. Xue, C. Luo, G. Lan, et al. Kryptein: A compressive-sensing-based encryption scheme for the Internet of Things. *Proc. - 2017 16th ACM/IEEE Int. Conf. Inf. Process. Sens. Networks, IPSN 2017* **2017**, 169–180.
13. M. Zhang, H. Zhang, D. Yuan, M. Zhang. Compressive sensing and autoencoder based compressed data aggregation for green IoT networks. *2019 IEEE Glob. Commun. Conf. GLOBECOM 2019 - Proc.* **2019**.
14. B. Jiang, G. Huang, F. Li, S. Zhang. Compressed Sensing with Dynamic Retransmission Algorithm in Lossy Wireless IoT. *IEEE Access* **2020**, 8, 133827–133842.
15. M. Prabha, S.S. Darly, B.J. Rabi. A novel approach of hierarchical compressive sensing in wireless sensor network using block tri-diagonal matrix clustering. *Comput. Commun.* **2021**, 168, 54–64.
16. Z. Sun, J. Liu, Z. Li, et al. CSR-IM: Compressed Sensing Routing-Control- Method with Intelligent Migration-Mechanism Based on Sensing Cloud-Computing. *IEEE Access* **2020**, 8, 28437–28449.
17. T. Bose, S. Bandyopadhyay, S. Kumar, A. Bhattacharyya, A. Pal. Signal characteristics on sensor data compression in IoT - An investigation. *2016 13th Annu. IEEE Int. Conf. Sensing, Commun. Networking, SECON 2016* **2016**.
18. A.Y. Tuama, M.A. Mohamed, A. Muhammed, et al. Recent advances of data compression in Wireless Sensor Network. *J. Eng. Appl. Sci.* **2018**, 13 (21), 9002–9015.
19. U. Jayasankar, V. Thirumal, D. Ponnurangam. A survey on data compression techniques: From the perspective of data quality, coding schemes, data type and applications. *J. King Saud Univ. - Comput. Inf. Sci.* **2021**, 33 (2), 119–140.
20. M.I. Mohamed, W.Y. Wu, M. Moniri. Adaptive data compression for energy harvesting wireless sensor nodes. *2013 10th IEEE Int. Conf. Networking, Sens. Control. ICNSC 2013* **2013**, 633–638.
21. G. Giorgi. A Combined Approach for Real-Time Data Compression in Wireless Body Sensor Networks. *IEEE Sens. J.* **2017**, 17 (18), 6129–6135.
22. J.K. Alsalaet, A.A. Ali. Data compression in wireless sensors network using MDCT and embedded harmonic coding. *ISA Trans.* **2015**, 56, 261–267.
23. V. Potdar, A. Sharif, E. Chang. Wireless sensor networks: A survey. *Proc. - Int. Conf. Adv. Inf. Netw. Appl. AINA* **2009**, 636–641.
24. M. Amarlingam, P.K. Mishra, P. Rajalakshmi, M.K. Giluka, B.R. Tamma. Energy efficient wireless sensor networks utilizing adaptive dictionary in compressed sensing. *IEEE World Forum Internet Things, WF-IoT 2018 - Proc.* **2018**, 2018-January, 383–388.
25. M. Rani, S.B. Dhok, R.B. Deshmukh. A Systematic Review of Compressive Sensing: Concepts, Implementations and Applications. *IEEE Access* **2018**, 6, 4875–4894.
26. D.L. Donoho. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, 52 (4), 1289–1306.
27. S.S. Chen, D.L. Donoho, M.A. Saunders. Atomic Decomposition by Basis Pursuit. *SIAM J. Sci. Comput.* **1998**, 20 (1), 33–61.
28. D.L. Donoho, A. Maleki, A. Montanari. Message-passing algorithms for compressed sensing. *Proc. Natl. Acad. Sci. U. S. A.* **2009**, 106 (45), 18914–18919.
29. M.A.T. Figueiredo, R.D. Nowak, S.J. Wright. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE J. Sel. Top. Signal Process.* **2007**, 1 (4), 586–597.
30. E. Liu, V.N. Temlyakov. The orthogonal super greedy algorithm and applications in compressed sensing. *IEEE Trans. Inf. Theory* **2012**, 58 (4), 2040–2047.
31. Zhang, YB., Huang, LT., Li, YQ. et al. Low-rank and joint-sparse signal recovery using sparse Bayesian learning in a WBAN. *Multidim Syst Sign Process*. **2021**, 32, 359–379.