Baghdad Science Journal

Volume 23 | Issue 2 Article 28

2-24-2026

EEG Lossless Signal Compression Based on Magnitude Classification and Run Length Encoding

Hala A. Jasim
Department of Remote Sensing and GIS, College of Science, University of Baghdad, Baghdad, Iraq,
hala.abd@sc.uobaghdad.edu.iq

Loay E. George
Department of Remote Sensing and GIS, College of Science, University of Baghdad, Baghdad, Iraq,
loayedwar57@uoitc.edu.iq

Eman H. Khudhair
Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq,
eman.hatem@sc.uobaghdad.edu.iq

Bushra A. Sultan
Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq,
bushra.sultan@sc.uobaghdad.edu.iq

How to Cite this Article

Jasim, Hala A.; George, Loay E.; Khudhair, Eman H.; and Sultan, Bushra A. (2026) "EEG Lossless Signal Compression Based on Magnitude Classification and Run Length Encoding," *Baghdad Science Journal*: Vol. 23: Iss. 2, Article 28.
DOI: https://doi.org/10.21123/2411-7986.5220

BAGHDAD SCIENCE JOURNAL 2026;23(2):735–746

Scan the QR to view the full-text article on the journal website

RESEARCH ARTICLE

EEG Lossless Signal Compression Based on Magnitude Classification and Run Length Encoding

Hala A. Jasim [1,*], **Loay E. George** [1], **Eman H. Khudhair** [2], **Bushra A. Sultan** [2]

[1] Department of Remote Sensing and GIS, College of Science, University of Baghdad, Baghdad, Iraq
[2] Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq

ABSTRACT

Electroencephalography (EEG) data comes with a large size due to the data's high sampling rate. Therefore, compressing EEG data is very important for storing the EEG files efficiently with less space and bandwidth capacity requirement. This research develops an efficient system for EEG data compression. The recorded EEG data are preprocessed and scaled using certain Resolution Factor and truncated to integer numbers, then the scaled EEG samples are classified into small and large vectors using a proposed adaptive thresholding which is based on using three computed factors: Standard deviation, Average of samples (Mean), and the multiplier factor (α). Then, each sample is passed through one of three procedures, then saved into the output file using multi-shift coding algorithm The best values are chosen as the tradeoff between the compression ratio and the processing time. The results indicated that the value of α parameter is significantly affects the threshold calculation, where the best-proven value for α is 1.30; the system achieves a compression gain of 65% while managing a reasonable processing time of 4.007 Second. The resolution factor affected the Mean Squared Error (MSE) and Mean Absolute Error (MEA) significantly, but it had a slight effect on the Compression Ratio (Cr). The α parameter has a great effect on Cr and a slight on MSE. The findings show a consistent trend whereby, as the resolution factor gradually decreases from 2 to 0.1, a concurrent decrease is observed in the MAE, MSE, Bitrate, Cr, and the overall processing time.

Keywords: Data scaling, Electroencephalography, EEG compression, Multi-shift coding, Run length encoding, Signal thresholding

Introduction

The European Data Format (EDF) is a standardized file format commonly used for storing and exchanging time-series data, particularly in the context of biomedical signals. [1] Originally, EDF was developed by the European Society for Biomedical Engineering. The original form of data in an EDF file is a sequence of digital samples representing physiological or biomedical signals. [2] These digital samples are acquired from various sensors or recording devices and are typically raw measurements of the underlying physiological activity. The nature of the original data depends on the type of signals being recorded. Some common types of signals stored in EDF files include: [3,4]

- Electroencephalogram (EEG): EEG recordings measure electrical activity in the brain. The original data consists of voltage measurements at different electrodes placed on the scalp. These measurements capture the fluctuations in electrical potential resulting from neuronal activity. [5]

Received 7 May 2024; revised 29 November 2024; accepted 1 December 2024.
Available online 24 February 2026

* Corresponding author.
E-mail addresses: hala.abd@sc.uobaghdad.edu.iq (H. A. Jasim), loayedwar57@uoitc.edu.iq (L. E. George), eman.hatem@sc.uobaghdad.edu.iq (E. H. Khudhair), bushra.sultan@sc.uobaghdad.edu.iq (B. A. Sultan).

https://doi.org/10.21123/2411-7986.5220

- Electrocardiogram (ECG or EKG): ECG recordings represent the electrical activity of the heart. The original data is a series of voltage measurements that correspond to the depolarization and repolarization of the heart's chambers. [6]
- Electromyogram (EMG): EMG recordings capture the electrical activity of muscles. The original data consists of voltage measurements reflecting muscle contractions and activity. [7]
- Electrooculogram (EOG): EOG recordings measure the electrical activity of the muscles controlling eye movements.

The original data includes voltage measurements associated with eye movements and positions. [8] Other Physiological Signals: EDF files can store a variety of different physiological signals, such as blood pressure, respiratory rate, and more. The original data for these signals would be specific to the physiological process being measured. [9] An EDF file consists of two main components: the header and the data records. [2] The header provides metadata about the recording and essential information to interpret the accompanying data such as version Information: Specifies the version of the EDF format used in the file. Patient Information: Includes details about the patient, such as name, identification number, birthdate, and sex. Recording Information: Describes attributes of the recording, including the start date and time, recording duration, and the number of data records. Signal Information: Specifies characteristics for each signal (channel), including the label, transducer type, physical dimension, physical minimum and maximum values, digital minimum and maximum values, and the number of samples in each data record.

While the data records contain the actual time-series data, each record typically represents a fixed duration of the recording. The samples for each signal are stored sequentially in the data records. The values of the samples are often digital representations of the physiological signals, and the header information is essential for converting these digital values to meaningful physical units. The overall structure is designed to be simple yet versatile, accommodating diverse biomedical data sources. [10] The increasing volume of these data, especially EEG data, poses significant challenges in terms of storage spaces, transmission bandwidth, and processing time. Lossless compression techniques may neglect to achieve a balance between data accuracy and compression ratio, while lossy compression may lead to losing important information needed for accurate analysis. Therefore, there is an urgent need for an efficient near-lossless compression scheme that can effectively reduce the volume of these data while maintaining integrity and facilitating rapid access for real-time processing. [5] This work proposes two-stage compression: lossy, followed by a lossless compression. The general aim of this study is to develop an efficient near-lossless compression system for EEG data to preserve the stability of data and obtain the best compression ratio and less processing time while maintaining the significant structures that are essential for accurate analysis and interpretation ensuring that the system retains the critical features of this signals, such as frequency components and events-related potentials, during the compression process to obtain the data fidelity. The other important goal of a good compression system is to enhance the speed of the entire system, i.e., improving the speed of compression and decompression operations to assist real-time analysis and compression. By achieving these objectives, the proposed near-lossless compression system aims to significantly contribute to effectively handling EEG data and enhance both research and clinical applications in neuroscience. To determine which combination of lossy and lossless compression algorithms provides the optimum performance, several combinations are investigated. The paper outlines the remaining sections as follows: Section Two introduces the related works in this domain. Section Three presents and describes the proposed compression system. Section Four shows the results of system tests and performance metrics. Finally, section Five concludes the paper.

Related works

In the field of EEG compression, researchers have attempted to develop effective techniques for reducing the massive amount of data while preserving significant neurological information. In various research the researchers aim to enhance data storage, analysis efficiency, and transmission in applications from clinical diagnostics to brain-computer interface technologies. Several efforts were undertaken to improve the EEG data processing and compression technologies. This paragraph should outline the general direction of previous research in this area conducted between 2017 to 2024.

Hejrati, Fathi, and Abdali-Mohammadi, the research presents the first learning-based adaptive transform that combines reconstruction techniques from artificial neural networks (ANNs) with DCT. In the reconstruction phase, the DCT coefficients of the EEG data are transformed using an adaptive ANN-based transform to minimize dimensionality and approximate the original DCT coefficients of the EEG. Additionally, the difference between the

estimated and original DCT coefficients is quantized to provide a novel near-lossless compression technique. Together with the predicted DCT coefficients, the quantization error was coded using arithmetic coding and delivered as compressed data. The suggested method shows a greater compression rate across a variety of datasets. While this technique effectively reduces data size, it faces limitations such as high computational requirements, and real-time processing may suffer from latency, and the complexity of neural networks.[11]

Chen, the study presents theories and fundamentals of the Free Lossless Audio Codec, a widely used audio compression tool, were covered in this thesis. A number of lossless audio compression methods employ the conventional audio compression structure, which FLAC applied. The frequency components of signals are extracted by spectral analysis, and these are then utilized to build a substitute predictor. Using a long-term EEG dataset, the developed FFT-FLAC encoder produced a compression ratio of 0.494. In order to demonstrate the promise of machine learning approaches in this field, a new predictor that employed a Wavelet Neural Network was evaluated on EEG signals. Since EEG data is almost always multi-channel, it was looked into if intra-channel redundancy could increase compression ratios. This method is intended for audio files that show a nearly constant number of peaks in the FFT spectrum. The compression ratio is reduced in EEG files that contain abnormal activities, which increases the number of peaks and negatively affects the compression ratio.[12]

Alsenwi Ismail, and Darweesh, the aim of the paper is to develop an efficient system for EEG data compression. A compression system made of both lossless and lossy compression algorithms was constructed. Using thresholding after the DCT and DWT transform is a lossy compression method. Lossless compression techniques include arithmetic encoding and run-length encoding (RLE). The employment of lossless techniques is made easier by the significant redundancy of the data generated by the lossy compression portion. The researchers test CR, RMSE, and compression time to evaluate their system performance. When compared to DWT and arithmetic encoding, they found that the greatest results come from utilizing DCT as a lossy compression algorithm and RLE as a lossless compression approach. The combination of the used algorithm may leverage the complexity of the system; also, using lossless compression may introduce a lower compression ratio.[13]

Athira and Rachana, this study proposes a lossless EEG compression that relies on a multilevel compression technique. The proposed system contains a two-stage prediction, then voting prediction and quad- encoding. The prediction two stages consist of 27 conditions and six functions, which were used to predict the current data from previous data. For the Voting prediction, the researcher applies a method to find the difference between predicted data and current data. For quad-entropy coding, the researchers use two-stage Huffman coding, Golomb rice coding, and dictionary method. The testing result shows that the multilevel compression system can achieve more compression ratio and less transmission time; however, it increases the complexity of the proposed system.[14]

Hadi, and George, they proposed EEG compression system uses the bi-orthogonal wavelet transform Tap 9/7 to compress EEG signals efficiently. It involves decomposing the signal into low and multi-high sub-bands, quantizing wavelet sub-bands, and encoding the input stream with double-shift coding. The system's efficiency is evaluated using metrics like compression ratio (CR) and mean square error (MSE). Comparative analysis shows Tap 9/7 reduces complexity and yields superior results compared to Discrete Wavelet Transform (DCT). Experimental evaluations show the system achieves a compression ratio of 7 with minimal error.[15] El Hanine et al, presented a lossless compression method to compress ECG signals by lowering the dynamic range and the number of encoded signals. This methodology used the A-law commanding algorithm in conjunction with the inter-leads correlation. The findings demonstrate that this method effectively preserves pathological data for individuals with arrhythmias or those in normal conditions.[16]

The proposed system

In this research, a new EEG data compression system was proposed; it consists of two stages: Lossy and Lossless. Its foundation is based on some statistical characteristics of the input data. The work plans can be summarized as follows:

1. Reading the data and then converting it to integer numbers based on the (Resolution Factor) that will be chosen.
2. Performing a scalar and rounding (quantization).
3. Classifying the data based on the value using some statistical criteria such as: Mean, STD, and inclusion factor (alpha).
4. After classification, all numbers are converted to positive using the mapping to positive step.
5. Adopting the encoding mechanism for the series of small and large numbers resulting from

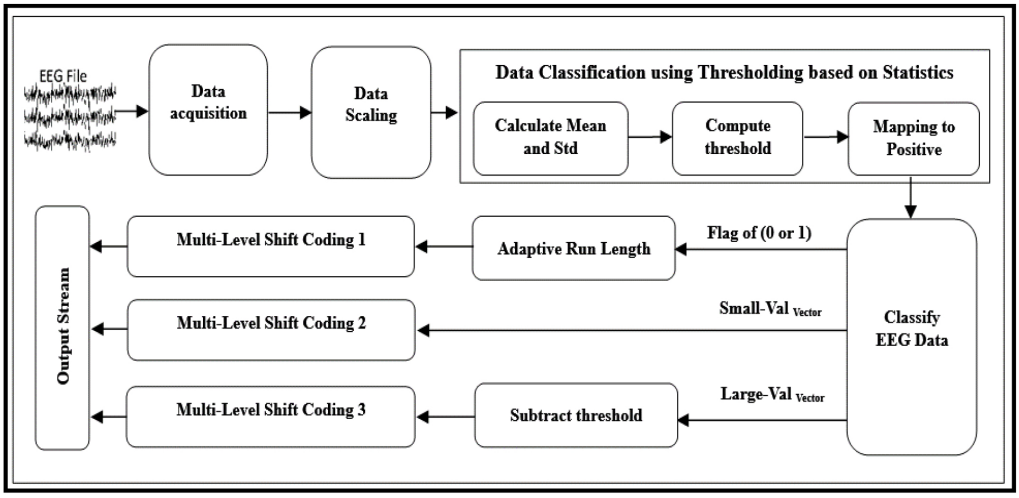


Fig. 1. The layout of the proposed compression method.

the classification using the run length encoding technique to create the three matrices:

a. Sequence of run length.
b. Sequence of small numbers.
c. Sequence of large numbers.

6. The first and second series were processed using multi-shift coding.
7. The third was processed using the value reduction stage and then multi-shift coding.
8. As a final stage, all three outputs will be collected into one file using the concatenation mechanism.

Steps 1 and 2 consider the lossy stage, the other steps are lossless. The proposed system layout is shown in Fig. 1.

Data acquisition

The EDF file contains digital samples that are usually represented as integer values. The file's header has crucial information for interpreting these samples, such as the physical units of measurement, the digital minimum and maximum values, and the physical minimum and maximum values. This information is necessary to convert the raw digital samples to meaningful physical units for analysis and interpretation, Eq. (1) can be used to carry out this conversion. [2]

$$X_{\text{physical}} = \frac{\left(X_{\text{digital}} - D_{min}\right) * \left(P_{\max} - P_{\min}\right)}{\left(D_{max} - D_{\min}\right)} + P_{\min} \quad (1)$$

where:

- Xphysical is the physical value of the signal.
- Xdigital is the raw digital sample obtained from the EDF file.
- Dmin and Dmax are the digital minimum and maximum values specified in the EDF header 2048, 2047 respectively.
- Pmin and Pmax are the corresponding physical minimum and maximum values specified in the EDF header -800,800, respectively.

To implement the proposed system, a two-dimension array was defined; it is called Signal-Samples with (Number of Channels) as a row that is defined for each of the 23 channels in the EEG file and (Number of Samples) as a column that contains the sample of each channel. The following process will be repeated sequentially for all channels in the EEG file.

Data scaling

The samples are downgraded and converted to integer data, with their fractal components truncated to reduce the required storage space per sample. This process is controlled by the resolution factor (Rs) ranging from 0.1 to 3, which plays a central role in the scaling operation applied to the values within the data array. Each sample undergoes a scaling transformation, as outlined by Eq. (2):

$$ScaledSample\,[I] = (int)\left(SignalSamples\,[I] \times Rs\right) \quad (2)$$

Each step in this and the following steps focuses on one channel individually and is repeated 23 times for each channel in the EEG file. This operation reduces the magnitude of the values, imparting a rescaled representation of the sample. The lower bound of Rs 0.1

intensifies the scaling effect, potentially heightening sensitivity to minor variations. Conversely, the upper bound set to 3, moderates the scaling, retaining more of the original magnitude and sacrificing sensitivity for data robustness. Concurrently, the opposite of Rs, denoted as Res, ensures the restoration of scaled values to their original value during the computation of error metrics. The difference between the original and the scaled samples[17] governs in Eq. (3):

$$Def\,(i) = \sum_{i=0}^{Number\ of\ Samples} \left[Original - Sample\,(i) - \left(Scaled_{Sample}\,(i) \times Res\right)\right] \quad (3)$$

The computed difference (Def) detects information loss during scaling and computes the accuracy of the system. The rest of the following procedures are lossless; this ensures data integrity by focusing on error calculation to maintain the accuracy of the original EEG data.

Data classification using thresholding based on 1st order statistics

References After the pre-processing is completed, the computation of the threshold is the current step, the input of this step is the Scaled-Sample vector, and the output is ($Flag_{vector}$, $Larg\text{-}Val_{Vector}$, $Small\text{-}Val_{Vector}$). The classification of the Scaled sample depends on three factors: first-order mean (M), Standard Deviation (Std), and Multiplication factor (α). The variable α ranges from 0.1 to 1.5. It significantly affects threshold calculation. Determines the threshold (Thr) in Eq. (4):

$$Thr = 2\ \times \left(M + \alpha \times Std\right) \quad (4)$$

Here, the lower bound of α is 0.1 downplays the role of the standard deviation in threshold computation, offering a threshold more influenced by the average. Conversely, the upper bound is 1.5 amplifies the impact of the standard deviation, resulting in a more responsive threshold to data variations. The subsequent step involves converting all values into positive values through the Mapping-to-Positive process. In this step, all sample values are transformed by multiplying positive values by 2, while negative values are multiplied by -2 and subtracting 1. The outcome of this process is a positive array where even values indicate the original positive values and odd values indicate the transformed negative values. This transformation contributes to the unification of data representation, simplifying further processing steps see Eq. (5) elucidates the procedure for converting values to their positive counterparts.

$$S\,[I] = \begin{cases} -2 * S\,[I] - 1 & If\ S\,[I] < 0 \\ 2 * S\,[I] & Otherwise \end{cases} \quad (5)$$

The following stage is Binary Flag Assignment. It performs the task of categorizing and designating the positions of specific values by labeling them as either "large" with the identifier 1 or "small" with the indicator 0 in an array known as the $Flag_{Vector}$, by comparing all values with the predefined threshold; if the value is larger than a threshold, mark 1 in the $Flag_{Vector}$ and add this value to the vector of large numbers ($Larg\_Val_{Vector}$), if the value is less than the threshold, it will marks 0 in $Flag_{Vector}$ and add it to the vector of small numbers ($Small\text{-}Val_{Vector}$). After this process, three arrays will be created ($Flag_{Vector}$, $Larg\_Val_{Vector}$, $Small\_Val_{Vector}$). Algorithm (1) summarizes the methods of thresholding and classification of samples.

Algorithm 1: Data Classification using thresholding based on statistics.

Start
Input: S (vector of Scaled-Sample).
- For $0 \le I \le$ Number of Samples:
  $Sum = Sum + |S[I]|$
  $Sum2 = Sum2 + (|S[I]| * |S[I]|)$
- Mean = Sum / $number\ of\ Samples$.
- Std = $\sqrt{(Sum2\ /\ numSamples - Mean * Mean}$
- Thr = (int)(2 * (Mean + α * Std))
- For $0 \le I \le$ Number of Samples:
  IF (S[I] < 0) S[I] = (S[I] *-2) -1
  Else S[I] = S[I] * 2
- For $0 \le I \le$ Number of Samples:
  IF (S[I] > Thr)
  Add to Larg-ValVector, FlagVector[I] = 1
  Else Add to Small-ValVector; FlagVector[I] = 0

Output: $Flag_{Vector}$, $Larg\text{-}Val_{Vector}$, $Small\text{-}Val_{Vector}$
End.

Run-length representation

Run-Length Representation is a form of lossless data compression.[18] The program identifies successive repetitions of a specific value and substitutes them with the value followed by the number of repetitions. This technique is most beneficial for data sets that have numerous redundancy values. In the proposed system, an adaptive Run-Length was used. It runs on $Flag_{vector}$ that contains 0,1, binary values only, and outputs the $RunLength_{Vector}$, which includes the numbers of runs; at the beginning, it saves the first value in the sequence, either zero or one, and then it counts the consecutive repeating occurrences of these

Table 1. Multi-shift coding parameters.

Key	Value	No. Bits	Range boundaries	Range of Values
1	V	Bt1	$R1 = 2^{Bt1}$	$0 \ldots (2^{Bt1} - 1)$
01	V-R1	Bt2	$R2 = R1 + 2^{Bt2}$	$2^{Bt1} \ldots (2^{Bt2} - 1)$
001	V-R2	Bt3	$R3 = R2 + 2^{Bt3}$	$2^{Bt2} \ldots (2^{Bt3} - 1)$
0001	V-R3	Bt4	$R4 = R3 + 2^{Bt4}$	$2^{Bt3} \ldots (2^{Bt4} - 1)$
0000	R5-R4	Bt5	$R5 = Max - R4 + 1$	$2^{Bt4} \ldots Max$

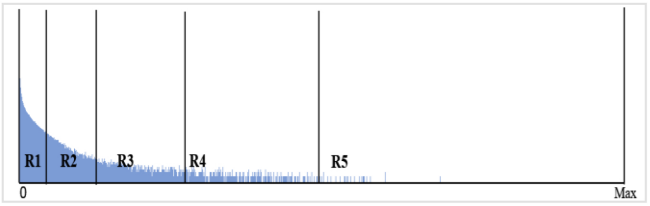


Fig. 2. Histogram of values with range boundaries.

ones and zeros. For each consecutive occurrence, it increases the counter by one; otherwise, if a change occurs 0 - 1 or 1 - 0, then it saves the counter value in the $\text{RunLength}_{\text{Vector}}$ and sets the count to zero. For example, if the following series is avaliable:

"0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0" then the resulted vector will be "0 6 13 3" where 0 is the start value, and 6 is the number of consecutive zeros 13 is the numbers of ones and so on.

Multi-shift coding

In this part of the paper an optimization algorithm aimed at minimizing the total number of bits required to represent a set of samples. It begins by calculating the maximum number (Max) and the maximum bits (MaxBt) needed to represent the maximum sample value (Max), which must be computed as shown in Eq. (6)

$$MaxBt = \left\lfloor \frac{\log Max}{\log(2)} \right\rfloor \quad (6)$$

The multi-shift coding takes the three vectors ($\text{RunLength}_{\text{Vector}}$, $\text{Larg-Val}_{\text{Vector}}$, and $\text{Small-Val}_{\text{Vector}}$) as input and generates a binary file containing these vectors as output. Each vector is processed individually in the multi-shift coding process. The algorithm categorizes the values into five ranges: Range-1, Range-2, Range-3, Range-4, and Range-5. Each range is calculated using values R1 ... R5, as shown in Table 1. The first range's upper limit must be smaller than R1, where $R1 = 2^{Bt1}$, The second range is from R1 to R2 where $R2 = R1 + 2^{Bt2}$, and so on.

After determining the boundaries of each range and the histogram of values, it shows that the first range is the smallest, but it contains the most number of values, while the last range is the larger range but it contains less number of values. Also, the values of range 1 are the smallest, while the values of range 5 are the biggest values of the sample. In Fig. 2, shows the histogram of samples and the five range boundaries. In order to get the most benefit of this technique, the values of range one will be stored as it was, while the other ranges 2 ... 5 will be reduced as if the value belongs to range two, the multi-shift coding will be subtracted R1 which is the end of range one from this value then to distinguish this value from the other values it will add the code 01 to the beginning of it. Table 1 shows the code of each range.

Algorithm 2 elucidates the step-by-step procedure for executing Multi-Shift Coding steps:

Algorithm 2: Multi-shift coding steps.

Start
Input: V (denoting the I'th value of the input Vector).
R1, R2, R3, R4, R5 (Ranges boundaries).
Bt1, Bt2, Bt3, Bt4, Bt5 (number of bits needed to save value in the specific range.
If V < R1 then: send to binary file Key $(1)_2$, V, Bt1.
Else if V < R2 then: send to binary file Key $(10)_2$, V-R1, Bt2.
Else if V < R3 then: send to binary file Key $(110)_2$, V-R2, Bt3.
Else if V < R4 then: send to binary file Key $(1110)_2$, V-R3, Bt4.
Else if V < R5 then: send to binary file Key $(1111)_2$, V-R4, Bt5.
Output: Binary file.
End.

Data set

The proposed system was applied and tested its performance on the dataset https://doi.org/10.13026/C2K01R.[19] It is a free online data collection of single-dimensional EEG recordings of 22 pediatric subjects with intractable seizures, Published on June 9, 2010.

Table 2. The effect of α variable on Error, CR, and time.

MAE	MSE	BitRate	α	Rs	Compression Ratio	Time for all channels	Gain
0.392	0.202	7.002	0.1	1.3	2.285	02.209	56.24
0.392	0.202	7.006	0.2	1.3	2.284	02.217	56.22
0.392	0.202	7.003	0.3	1.3	2.285	02.076	56.24
0.392	0.202	7.004	0.4	1.3	2.285	02.160	56.23
0.392	0.202	7.005	0.5	1.3	2.285	02.288	56.23
0.392	0.202	7.012	0.6	1.3	2.282	02.334	56.18
0.392	0.202	7.020	0.7	1.3	2.280	02.587	56.13
0.392	0.202	7.033	0.8	1.3	2.276	02.815	56.05
0.392	0.202	7.040	0.9	1.3	2.273	03.152	56.00
0.392	0.202	7.047	1	1.3	2.271	03.269	55.96
0.392	0.202	7.057	1.1	1.3	2.268	03.454	55.90
0.392	0.202	7.045	1.2	1.3	2.271	03.820	55.97
0.392	0.202	7.032	1.3	1.3	2.276	04.717	56.05
0.392	0.202	7.020	1.4	1.3	2.280	04.690	56.13
0.392	0.202	7.014	1.5	1.3	2.282	05.001	56.17
0.392	0.202	7.061	1.6	1.3	2.267	06.547	55.88
0.392	0.202	7.066	1.7	1.3	2.265	07.327	55.84
0.392	0.202	7.050	1.8	1.3	2.270	07.517	55.95
0.392	0.202	7.041	1.9	1.3	2.273	08.400	56.00
0.392	0.202	7.050	2	1.3	2.270	09.780	55.95

Version: 1.0.0. the proposed system was implemented using the Windows 10 operating system and C# programming language as the testing environment.

Performance metrics

In order to evaluate the result of the proposed system, many performance metrics have been considered as follows:

1. Compression Ratio (CR): the size of the original file in bytes divided by the size of the compressed file in bytes.[20] If the CR is a high value, it means a high compression performance. The Eq. (7) used to compute the CR of the proposed system:[21]

$$CR = \frac{Size\ of\ original\ Data}{Size\ of\ Compressed\ Data} \tag{7}$$

2. Mean Absolute Error (MAE): This indicates the average magnitude of the errors between the original and retrieved signals. It's computed by taking the average of the absolute differences between corresponding sample intensities of the original and compressed signals,[22,23] as shown in Eq. (8).

$$MAE = \frac{\sum_{I=0}^{No.\ Samples} |Original\ Samples[I] - Retrieved\ Sample[I]|}{Number\ of\ Samples} \tag{8}$$

3. Mean Square Error (MSE): measures the average squared difference between the original signals and the compressed signals;[23,24] it is calculated by averaging the squared differences between corresponding sample intensities of the original and compressed signals,[25] as shown in Eq. (9).

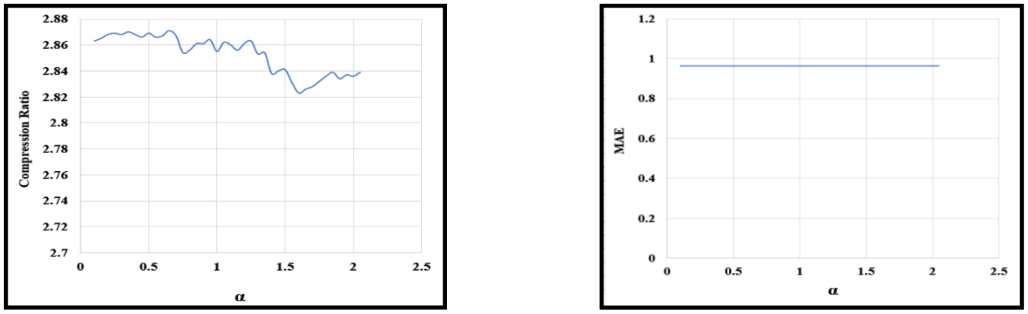
$$MSE = \frac{\sum_{I=0}^{No.\ Samples} (Original\ Samples[I] - Retrieved\ Sample[I])^2}{Number\ of\ Samples} \tag{9}$$

4. Bitrate: Bitrate signifies the average number of bits required to represent each sample in the compressed signals. It is determined by dividing the total number of bits in the compressed signals by the total number of samples, as shown in Eq. (10).

$$Bitrate = \frac{Total\ number\ of\ bits}{Number\ of\ samples} \tag{10}$$

Results and discussion

To test the efficiency of the proposed system, the values of the variables used in the algorithms were tested and the best values were chosen as the balance between the compression ratio and the time taken for the process. The first variable was the α; Table 2 encapsulates a comprehensive analysis of the compression ratio across various α values; notably, as the α increases from 0.1 to 3, the Compression Ratio (Cr) generally decreases, highlighting the expected inverse relationship between compression efficiency and α. For the α value of 1.3, the system achieves a commendable Compression Ratio



a: α effect on Cr b: α effect on MEA

Fig. 3. The relation between α and Cr and MSE.

Table 3. The effect of the resolution factor on the result.

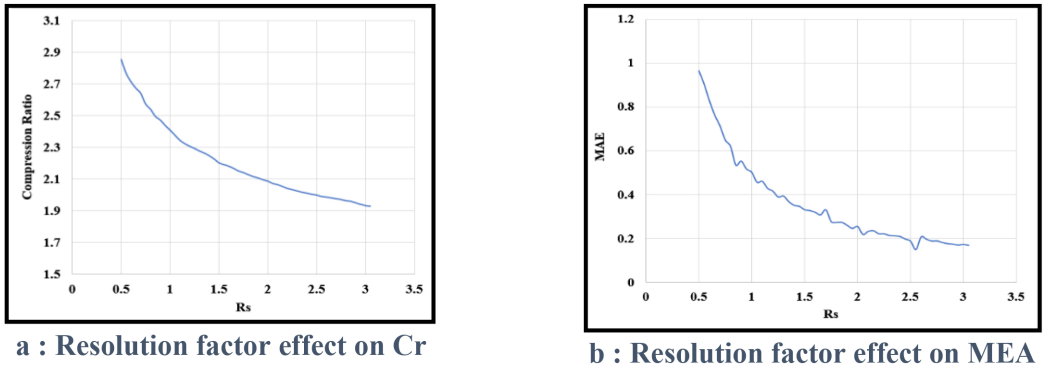
MAE	MSE	BitRate	α	Rs	Compression Ratio	Time for all channels (in Sec)	Gain
0.964	1.263	5.574	0.65	0.5	2.836	02.25	65.00
0.827	0.917	5.880	0.65	0.6	2.722	02.52	63.26
0.712	0.678	6.093	0.65	0.7	2.626	02.37	61.92
0.618	0.513	6.289	0.65	0.8	2.544	02.33	60.70
0.551	0.407	6.465	0.65	0.9	2.475	02.39	59.60
0.500	0.333	6.628	0.65	1	2.414	02.44	58.58
0.459	0.279	6.777	0.65	1.1	2.361	02.41	57.65
0.414	0.229	6.898	0.65	1.2	2.320	02.59	56.89
0.392	0.202	7.015	0.65	1.3	2.281	02.51	56.16
0.349	0.164	7.119	0.65	1.4	2.248	02.61	55.51
0.329	0.145	7.220	0.65	1.5	2.216	02.78	54.88
0.317	0.133	7.312	0.65	1.6	2.189	02.72	54.31
0.329	0.136	7.403	0.65	1.7	2.162	02.69	53.74
0.271	0.099	7.486	0.65	1.8	2.138	02.76	53.22
0.258	0.090	7.568	0.65	1.9	2.115	02.81	52.71
0.253	0.084	7.643	0.65	2	2.094	02.85	52.24
0.230	0.072	7.720	0.65	2.1	2.073	02.87	51.76
0.219	0.065	7.791	0.65	2.2	2.054	02.99	51.31
0.212	0.060	7.847	0.65	2.3	2.039	03.08	50.96
0.207	0.057	7.911	0.65	2.4	2.023	03.10	50.57
0.186	0.047	7.971	0.65	2.5	2.008	03.18	50.19
0.205	0.054	8.028	0.65	2.6	1.993	03.30	49.83
0.186	0.046	8.082	0.65	2.7	1.980	03.42	49.50
0.179	0.043	8.127	0.65	2.8	1.969	03.30	49.21
0.172	0.039	8.186	0.65	2.9	1.955	03.47	48.85
0.171	0.038	8.230	0.65	3	1.944	03.54	48.57

of 2.276 while managing a reasonable processing time of 4.717 Ms. This signifies a noteworthy balance between compression efficiency and real-time processing requirements. While slightly lower Compression Ratios can be achieved with $\alpha = 1.2$, the corresponding processing time of 3.820 Sec offers an alternative, reinforcing the application-specific nature of the "best" value for α.

The provided data suggests that α does not have a direct impact on the Mean Squared Error. The MSE remains constant at 0.202 regardless of the variations of its value. In summary, the data implies that changes in the Multiplier value have a noticeable impact on the Compression Ratio (Cr), with higher Multiplier values leading to lower compression efficiency. However, the Mean Squared Error (MSE) remains constant, indicating that the Multiplier may not be a direct factor influencing the accuracy or quality of the compressed data. The specific interpretation may vary based on the underlying algorithm or compression method being used. The Fig. 3 show relation between α and Cr and MSE, respectively.

The second variable is the resolution factor (Rs). Table 3 shows the effect of (Rs) on the result of the proposed system.

The tabulated results provide a detailed exploration of system performance across various values of Rs, with a particular focus on the impact of the resolution (Rs) parameter. The data reveals a consistent pattern



a : Resolution factor effect on Cr

b : Resolution factor effect on MEA

Fig. 4. The relation between the resolution factor and Cr and MAE.

Table 4. Compression ratio on different datasets.

Data Set	edf	Samples (CSV)	Proposed method	CR - edf	CR - CSV	Gain-edf	Gain-CSV
chb01_01	42,401,792	370,831,360	14863384.88	2.853	24.949	64.946	95.992
chb01_02	42,401,792	370,864,128	14697701.13	2.885	25.233	65.337	96.037
chb01_03	42,401,792	370,458,624	15893134.75	2.668	23.309	62.518	95.710
chb02_01	42,401,792	370,548,736	16236983	2.611	22.821	61.707	95.618

where, as the resolution progressively decreases from 3 to 0.4, there is a corresponding decrease in Bitrate, but it gave the highest Compression Ratio (Cr), and reduce time taken for the entire process. This suggests that a lower resolution, indicating a larger region for value classification Rs = 0.35, contributes to increase errors, lower data representation requirements (Bitrate), improved compression ratios, and shorter processing times. It is noteworthy that there exists a trade-off, as higher resolutions closer to 2 results in slightly lower compression ratio but may lead to reduced errors and longer processing times. The data signifies the significance of judiciously selecting the resolution parameter to align with the desired balance between accuracy and computational efficiency in the compression process. The Fig. 4 shows the relation between the resolution factor and Cr and MAE respectively.

The computation of compression ratios in Table 4 is performed by comparing the sizes of the EDF file and the CSV file, which only contains the samples recorded in double data format, with the size of the output file of the proposed method. The compression ratios obtained indicate a substantial level of compression efficiency, with the EDF file being compressed to 2.6 to 2.8 times its original size, while the CSV file achieves compression ratios of 22 to 24 times its original size. In addition, Table 4 illustrates that the compression gains for the EDF file range from 60% to 64%, whereas the compression gain for the CSV file is roughly 90% to 95%. The data shown highlights the effectiveness and efficiency of the suggested compression technique in decreasing file sizes and preserving storage capacity, particularly when dealing with CSV data. CSV data often demonstrates higher compression ratios compared to structured formats such as EDF. The results presented in Table 4 highlight the significant compression capabilities of the suggested method, offering an appealing solution for enhancing data storage and transmission efficiency in situations where reducing file sizes is of utmost importance.

Conclusion

The study paper presents a new signal-processing technique that utilizes adaptive thresholding to categorize samples into two groups to minimize the number of bits needed to save each sample, along with a flag to monitor this sample for restoration purposes. A run-length representation is used to tally the consecutive occurrences of a flag to minimize the space reserved for this flag. Subsequently, a multi-shift coding method is employed to handle and save each value in a file. The procedure was executed and assessed using a dataset. The optimal settings were chosen as a balance between compression ratio and processing time, to determine the efficacy of the proposed approach. The parameter α was the most effective in the system. The system achieves a compression ratio (CR) of 2.83 with α value equal to 0.35 while also maintaining a processing time of 2.18 Seconds. The α number has no direct impact on the Mean Squared Error, although a slightly lower Cr. The MSE remains constant at 1.263 despite fluctuations in its value. The evidence indicates that the Cr is greatly affected by variations in the Multiplier value. The data indicates a steady trend where the Mean

Absolute Error (MAE), Mean Squared Error (MSE), Bitrate, Cr, and processing time decrease proportionally as the resolution decreases from 2 to 0.3.

The limitation of the proposed system is that it is designed to focus on Single-Dimensional brain signals or on representative signals that have similar characteristics in terms of temporal change in behavior. Meanwhile, it is not suitable for Bipolar EEG, which is computed as the differential signals between pairs of electrodes, emphasizing the spatial relationships and differences in activity between adjacent brain regions bipolar and is often small in amplitude because it includes sudden and rapid rises, which makes the signal range large and not similar to the behavior of the Single-Dimensional brain signals.

Future works

Future work in this field may include the exploiting the idea of classifying the EEG signal into large and small components with the possibility of using wavelet transform to compress the large component, using other types of high-order compressors (high entropy encoders) to accomplish the final stage of compression and using non-uniform quantization to accomplish the conversion process from float to integer.

Authors' declaration

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are ours. Furthermore, any Figures and images that are not ours have been included with the necessary permission for republication, which is attached to the manuscript.
- No animal studies are present in the manuscript.
- No human studies are present in the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee at University of Baghdad.

Authors' contribution statement

L.E.G. contributed by proposing the research ideas and the algorithms used and supervising the design and implementation of work steps. Meanwhile, H.A. J. and E.H.K., contributed by implementing the idea, extracting and developing the results, and writing the manuscript. B.A.S. contributed by supervising the research from both practical and theoretical perspectives, reviewing the manuscript, and ensuring it was written in a scientific manner.

Data availability

The datasets used in this article are publicly available in the following repositories: https://doi.org/10.13026/C2K01R.

References

1. Almahdi AJ, Yaseen AJ, Dakhil AF. EEG signals analysis for epileptic seizure detection using DWT method with SVM and KNN classifiers. Iraqi J Sci. 2021;2:54–62. https://doi.org/10.24996/ijs.2021.SI.2.6.
2. Kemp B, Olivan J. European data format 'plus' (EDF+), an EDF alike standard format for the exchange of physiological data. Clin Neurophysiol. 2003;114(9):1755–1761. https://doi.org/10.1016/S1388-2457(03)00123-8.
3. Zhou Y, Soltani S, Li BM, Wu Y, Kim I, Soewardiman H, *et al.* Direct-write spray coating of a full-duplex antenna for e-textile applications. Micromachines. 2020;11(12):1056–1067. https://doi.org/10.3390/mi11121056.
4. Oostenveld R, Fries P, Maris E, Schoffelen J-M. Open-Source Software for Advanced Analysis of MEG, EEG, and Invasive Electrophysiological Data. Comput Intell Neurosci. 2011:1–9. https://doi.org/10.1155/2011/156869.
5. Kyriaki K, Koukopoulos D, Fidas CA. A Comprehensive Survey of EEG Preprocessing Methods for Cognitive Load Assessment. IEEE Access. 2024;12(4):23466–23489. https://doi.org/10.1109/ACCESS.2024.3360328.
6. Manju BR, Akshaya B. Simulation of Pathological ECG Signal Using Transform Method. Procedia Comput Sci. 2020;171:2121–2127. https://doi.org/10.1016/j.procs.2020.04.229.
7. Hambly MJ, de Sousa ACC, Pizzolato C. Comparison of filtering methods for real-time extraction of the volitional EMG component in electrically stimulated muscles. Biomed Signal Process Control. 2024;87:1–11. https://doi.org/10.1016/j.bspc.2023.105471.
8. Steele AG, Faraji AH, Contreras-Vidal JL. Electrospinography for non-invasively recording spinal sensorimotor networks in humans. J Neural Eng. 2024;20(6):1–13.https://doi.org/10.1088/1741-2552/ad1782.
9. Silva LEV, Fazan Jr R, Marin-Neto JA. PyBioS: A freeware computer software for analysis of cardiovascular signals. Comput Methods Programs Biomed. 2020;197:1–12. https://doi.org/10.1016/j.cmpb.2020.105718.
10. Liao L-D, Chen C-Y, Wang I-J, Chen S-F, Li S-Y, Chen B-W, *et al* Gaming control using a wearable and wireless EEG-based brain-computer interface device with novel dry foam-based sensors. J Neuroeng Rehabil. 2012;9(5):1–12. https://doi.org/10.1186/1743-0003-9-5.
11. Hejrati B, Fathi A, Abdali-Mohammadi F. Efficient lossless multi-channel EEG compression based on channel clustering. Biomed Signal Process. 2017;31(1):295–300. https://doi.org/10.1016/j.bspc.2016.08.024.
12. Chen Y. Investigating the lossless compression of EEG data [master's thesis]. Wolfville, NS: Acadia University; 2018.
13. Alsenwi M, Ismail T, Darweesh MS. Hybrid Compression Techniques for EEG Data Based on Lossy/Lossless Compression Algorithms. International Conference on Microelectronics (ICM). 2017:1–5. https://doi.org/10.48550/arXiv.1804.02713.
14. Athira M S, Rachana M K. Lossless EEG Compression based on Highly Efficient Multilevel Compression Method for VLSI

Implementation. Int J Eng Res Technol. 2022;11(5):323–325. https://doi.org/10.17577/IJERTV11IS050216
15. Hadi HA, George LE. A Comparative Study Using DCT, Delta Modulation, and Double Shift Coding for Compressing Electroencephalogram Data. Iraqi J Sci. 2022;63(7):3189–3199. https://doi.org/10.24996/ijs.2022.63.7.38.
16. El Hanine M, Ouldzira H, Boudaoud A, Abdelmounim E. Lossless Compression Technique For ECG Signals Using A-law Companding Algorithm. IRASET. 2023:1–5. https://doi.org/10.1109/IRASET57153.2023.10152876.
17. Walters-Williams J, Li Y. BMICA-independent component analysis based on B-spline mutual information estimation. Signal & Image Processing: Int J Res.2012;3(4):63–79. https://doi.org/10.5121/sipij.2012.3203.
18. S. Zeitz, K. Kochs, M. Dörpinghaus and G. Fettweis, "Improved Runlength-Limited Codes for Systems Employing Zero-Crossing Modulation," 2025 IEEE 36th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Istanbul, Turkiye, 2025, pp. 1–6. https://doi.org/10.1109/PIMRC62392.2025.11274776.
19. Shoeb A. CHB-MIT scalp EEG database. MIT; 2010. https://doi.org/10.13026/C2K01R.
20. AL-Khafaji GK, Rasheed MH, Siddeq MM, Rodrigues MA. Adaptive Polynomial Coding of Multi-base Hybrid Compression. Int J Eng. 2023;36(2):236–252. https://doi.org/10.5829/IJE.2023.36.02B.05.
21. Aljumaily HA, George LE. Hybrid Color Image Compression Using Signals Decomposition with Lossy and Lossless Coding Schemes. ICITAMS. 2023:225–230. https://doi.org/10.1109/ICITAMS57610.2023.10525557
22. Sanusi W, Sidjara S, Patahuddin S, Danial M. A Comparison of Spatial Interpolation Methods for Regionalizing Maximum Daily Rainfall Data in South Sulawesi, Indonesia. ITM Web Conf. 2024;58:1–8. https://doi.org/10.1051/itmconf/20245804003.
23. Rashmi CR, Shantala CP. Evaluating Deep Learning with different feature scaling techniques for EEG-based Music Entrainment Brain Computer Interface. e-Prime – Adv Electr Eng Electron Energy. 2024;7:1–15. https://doi.org/10.1016/j.prime.2024.100448.
24. Shakeel A, Tanaka T, Kitajo K. Time-series prediction of the oscillatory phase of EEG signals using the least mean square algorithm-based AR model. Appl Sci. 2020;10(10):1–11. https://doi.org/10.3390/app10103616.
25. Al-Hassani MD. A Novel Technique for Secure Data Cryptosystem Based on Chaotic Key Image Generation. Baghdad Sci J. 2022;19(4):905–913. https://doi.org/10.21123/bsj.2022.19.4.0905.

نظام ضغط مخطط كهربائية الدماغ بدون فقدان البيانات باستخدام تصنيف الحجم وترميز طول التشغيل

هلا عبد السلام جاسم [1]، لؤي ادور جورج[1]، ايمان حاتم خضير[2]، بشرى عبد الله سلطان[2]

[1] قسم التحسس النائي ونظم المعلومات الجغرافية، كلية العلوم، جامعة بغداد، بغداد، العراق.
[2] قسم علوم الحاسوب، كلية العلوم، جامعة بغداد، بغداد، العراق.

الخلاصة

تأتي بيانات تخطيط كهربائية الدماغ (EEG) بحجم كبير نظرًا لارتفاع معدل أخذ العينات في البيانات. لذلك، هناك حاجة إلى مساحة كبيرة ونطاقات ترددية أكبر لتخزين ونقل بيانات مخطط كهربائية الدماغ. لهذا السبب، يعد ضغط بيانات تخطيط كهربائية الدماغ (EEG)أمرًا مهمًا للغاية لتخزين ملفات تخطيط كهربائية الدماغ (EEG) بكفاءة مع مساحة أقل وسعة عرض النطاق الترددي. يهدف هذا البحث إلى تطوير نظام فعال لضغط بيانات مخطط كهربائية الدماغ .(EEG) تتم معالجة بيانات تخطيط كهربائية الدماغ (EEG) المسجلة مسبقًا؛ يتم قياس البيانات باستخدام عامل الدقة (Rs) واقتطاعها لتحويلها إلى أرقام صحيحة، ثم يتم تصنيف العينات المقاسة لتخطيط كهربائية الدماغ (EEG) إلى نواقل صغيرة وكبيرة باستخدام العتبة التكيفية التي يتم حسابها باستخدام ثلاثة عوامل: الانحراف المعياري(Std) ، متوسط العينات (المتوسط)، والعامل المضاعف .(α) تتم معالجة كل عينة وترميزها وحفظها في ملف الإخراج باستخدام خوارزمية ترميز متعددة التحولات. تم اختيار أفضل القيم كمفاضلة بين نسبة الضغط ووقت المعالجة. تؤثر المعلمة α بشكل كبير على حساب العتبة حيث القيمة الأفضل المثبتة لـ α هي 1.30 ويحقق النظام نسبة ضغط قدرها 65% مع زمن معالجة معقول قدره 4.007 ثانية. أثر عامل الدقة على متوسط الخطأ التربيعي (MSE) ومتوسط الخطأ المطلق (MEA) بشكل ملحوظ، في حين كان له تأثير طفيف على نسبة الضغط .(Cr) ومن ناحية أخرى، فإن القيمة α لها تأثير كبير على نسبة الضغط وليس لها تأثير مباشر على الخطأ المطلق. تظهر النتائج أنه مع انخفاض عامل الدقة تدريجيًا من 2 إلى 0.1، لوحظ انخفاض متزامن في متوسط الخطأ التربيعي (MSE) ومتوسط الخطأ المطلق (MEA) ومعدل البت ونسبة الضغط ووقت المعالجة الإجمالي.

الكلمات المفتاحية: تخطيط كهربائية الدماغ، ضغط تخطيط كهربائية الدماغ، تصنيف الإشارات، العتبة، تمثيل طول التشغيل، التشفير متعدد التحولات، التكميم العددي.