Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 3, No. 2, March : 2023

LOSSLESS MEDICAL IMAGE STORAGE USING MODIFIED LZW COMPRESSION

Jillapalli Nageswara Rao Research Scholar, Department of Information and Technology, University College of Engineering, Osmania University, Hyderabad, India.
Dr. V. B. Narasimha Assistant Professor, Department of Computer Science and Engineering, University College of Engineering, Osmania University, Hyderabad :
Email: [1]nenusitam.rao@gmail.com, [2]vbnarasimha@gmail.com

Abstract
Applications for medical imaging create vast collections of pictures that are quite similar to one another. As a result, a compression method is helpful for reducing the amount of space required for storage. In such important applications, lossless compression techniques are an absolute need. Lossless compression is a category of data compression methods that enables the original data to be fully rebuilt from the compressed data using the compressed data. This paper presents a lossless compression technique using LZW compression. The input image is pre-processed by subtracting the consecutive pixels. This increases the redundancy in the input image thereby increasing the compression efficiency. The LZW compression is much faster and simpler compared to the existing algorithms. The compression ratio of the proposed model on medical images is above 2.

Keywords: lossless compression, LZW, compression efficiency, redundancy.

I. Introduction

Information pertaining to a patient that includes physical examinations, such as glucose levels, arterial blood pressure, blood particle ratio, and so on is referred to as healthcare data. These data come straight from the Internet of Things (IoT) and various other pieces of medical equipment. The digitalization of people's medical records offers direct assistance in the conduct of their everyday lives [1-2]. For the sake of providing better care and more sophisticated services, it is essential that remote specialists have the ability to handle and exchange patient data. The availability of such digital data can help the patient save both time and money. In networks with a limited bandwidth, the transmission of patient data is an extremely important process [3-4]. This communication becomes essential in order to transmit accurate data in a timely manner to care providers for the urgent treatment of patients who get ill or wounded while travelling or who have not taken medicine as directed by their care providers. However, the poor quality of networks and the lack of security provide substantial obstacles in the way of delivering such services, particularly telemedicine services supplied by businesses located in remote areas. Previous studies have shown that massive amounts of data being dumped all at once leads to an increase in traffic, which in turn causes significant latency and network congestion during the time spent processing and distributing data. They also show that when a rising round-trip time delay is established to transport substantial data from the source to the destination, healthcare data becomes insufficient for end-users. When transferring huge amounts of data across a network with a sluggish connection speed, there is a greater chance that errors may occur.

In addition, there is an increase in the risk that pertains to the data's security when it is accessed from a geographically remote location. The requested data can originate from a source that is not trusted or might have been tampered with by the attackers. A consultation that is based on these falsified facts may direct the patient to the incorrect therapy and lead the patient to develop a serious disease. Therefore, the use of distant data sharing may result in problems when trying to guarantee the authenticity and integrity of data. Authentication allows for the identification of the source of the data, while integrity ensures that data may be sent across sites without error. These security flaws need to be fixed as soon as possible since patient data are very sensitive. As a result, one of the most important concerns is the development of a system that can ensure the authenticity and integrity of data while simultaneously passing data with a low latency over the communication medium.

As was said earlier, the process of compressing medical data is necessary in order to ensure that the transfer of medical data is both quick and secure [5-6]. By adding compression to media files, you may make them smaller than they were before and much simpler to access. The medical reports and findings, such as X-rays, scans, MRIs, and electrocardiograms, will be in a large format. The cloud now stores the medical records of thousands upon millions of patients. Each day, a substantial number of medical files are uploaded to the cloud. Therefore, we need a big amount of storage capacity. Accessing such large medical data files might sometimes be a time-consuming process [7-8]. The answer to this predicament lies in the use of compression strategies.

II. Literature

Ata Ullah et al [9] provided a strategy that makes use of fog computing to aggregate healthcare data in a way that is both safe and effective. Peer-to-peer communication between healthcare sensing devices and wearables has been facilitated by the team in order to facilitate the exchange of confidential data with an aggregating node that is able to share data with a FoG server. In this case, an aggregator may be physically separated from the FoG server and thus unable to transfer data in a direct manner. However, it is able to share the encrypted data with the surrounding aggregator so that the neighbouring aggregator may send data to the FoG server by adding in the data that it has already gathered. The FoG server is able to extract the necessary values from the data and store them in the local repository so that they may be further modified and updated in cloud repositories at a later time. In order to provide these functions, the authors have provided two algorithms: one for receiving messages at the aggregator, and the other for extracting messages at the FoG server.

Aparna Kumari et al [10] suggested that a patient-driven healthcare architecture with three layers be developed for the gathering, processing, and transmission of real-time data. End users are provided with information into the suitability of fog devices and gateways in the Healthcare 4.0 environment for both present and future applications thanks to this.

Joseph Azar et al [11] provided a method that is both data gathering and analysis friendly to the environment. The first thing that the authors do is apply a fast error-bounded lossy compressor to the data that has been gathered before it is sent. This is done since the transmission process is often regarded as the IoT device that uses the most energy. In the second step, they reconstruct the data that was communicated on an edge node so that it may be processed using supervised machine learning algorithms. A Wireless Body Sensor Network (WBSN) and wearable devices are used to collect vital signs data from the driver, and this information is then transmitted to an edge node for the purpose of stress level detection. In order to validate the approach, they consider the context of driving behaviour monitoring in intelligent vehicle systems.

Ozal Yildirim et al [12] developed a deep network topology with 27 layers that is planned to consist of encoder and decoder component. In the component of this model referred to as the encoder, the signals are transformed into low-dimensional vectors. The signals are then reconstructed in the section referred to as the decoder. The representations of the low and high levels of signals in the model's hidden layers may be obtained via the use of the deep learning methodology. As a result, the original signal may be rebuilt with just a little amount of data being lost. A deep compression strategy is quite different from more conventional linear transformation approaches since it means that the system is able to learn to utilise various ECG data in an automated manner.

Sara Kadhum Idrees et al [13] presented a brand new approach for lossless EEG data compression that is enabled by fog computing to reduce the quantity of IoT EEG data that is transferred to the cloud. Clustering is the first step in the EEG data compression strategy, which is followed by the compression phase. Using a technique known as agglomerative hierarchical clustering, the data that has been received is first organised into clusters. In the second step, the Huffman encoding is used on each cluster that was produced as a consequence. At last, the files that were compressed into smaller clusters are reassembled, and then they are transferred from fog to cloud.

Alaa Awad Abdellatif et al [14] presented and implement an accurate and lightweight classification mechanism that, by leveraging some time-domain features extracted from the vital signs, enables a reliable seizures detection at the network edge with precise classification accuracy and low computational requirement. In other words, this mechanism should be able to present and implement an accurate and lightweight classification mechanism. After that, the authors come up with a plan for selective data transfer and put it into action. This plan chooses the method of data transmission that will be the least disruptive to the identified patient's symptoms and uses it. In addition to this, they offer a dependable and energy-efficient emergency warning system for the detection of epileptic seizures. This system is based on conceptual learning and fuzzy categorization.

Ashima Anand et al [15] proposed an enhanced method of watermarking that is able to secure patient data by embedding multiple watermarks in medical cover images utilising the DWT-SVD domain. This method may be found in the presentation. The Hamming code is applied to the text watermark prior to embedding it in order to decrease the channel noise distortion that would otherwise occur for the sensitive data. Following the process of embedding, the medical picture will be encrypted before being compressed. The combination of Chaotic-LZW displays the greatest performance out of two different encryption methods and three different compression schemes that were examined. Nevertheless, the HyperChaotic-LZW combination is more resistant to Gaussian assaults, as well as JPEG compression attacks, speckle noise attacks, and histogram equalisation attacks.

Amir M. Rahmani et al [16] proposed to use the idea of fog computing in healthcare IoT systems by creating a geo-distributed layer of intelligence that acts as an intermediate between sensor nodes and the cloud. The Fog-assisted system architecture is able to deal with many of the challenges that are present in ubiquitous healthcare systems, such as mobility, energy efficiency, scalability, and reliability issues. This is accomplished by taking on some of the responsibilities of the sensor network and a remote healthcare centre. If Smart e-Health Gateways are successfully implemented, it may be possible to facilitate the widespread deployment of ubiquitous health monitoring systems, particularly in clinical settings. In addition to this, the authors provide a prototype of a Smart e-Health Gateway known as UT-GATE. In this version of the gateway, several of the higher-level functionalities that have been mentioned have been implemented. In order to demonstrate the effectiveness and applicability of the system in the context of resolving a medical case study, they additionally develop an Internet of Things-based Early Warning Score (EWS) health monitoring system.

C.K. Jha et al [17] used EMD In order to get the first intrinsic mode function (IMF), which is based on the sifting function. Following EMD, the first IMF and four important filtering functions are merged together to form a single function. This combination eliminates a large number of the signal's components that are not important. This combination has had the Discrete Wavelet Transform, or DWT, applied to it with the mother wavelet set to "bior4.4." After performing DWT, the transform coefficients that were generated are next subjected to dead-zone quantization. It throws out any little transform coefficients that are clustered close to zero. In addition, the coefficients are converted to integers, and run-length encoding is used, so that the resulting ECG data may be stored in a compressed format.

Mrinai M. Dhanvijay et al [18] Described the WBAN-based Internet of Things healthcare system and conducts a state-of-the-art analysis of the network architecture topology and applications used in Internet of Things solutions for healthcare. In addition, this article examines the security and privacy features that are quite problematic in a variety of IoT healthcare architectures. These features include real-time wireless health monitoring, privacy, authentication, energy management, power management, resource management, Quality of Service, and resource management. The limitation of data and the maintenance of its integrity are both difficult tasks at this time since the architecture of the system is not fully defined. At this time, ninety percent of the information that is currently accessible has been gathered during the last two years. This survey's primary objective is to investigate healthcare purposes from the perspective of digital healthcare delivery systems. In addition, it reports a variety of IoT rules and systems, as well as those related to e-healthcare, which determine how to make all tolerable development easier.

Rafik Hamza et al [19] suggested a chaos-based encryption cryptosystem for the purpose of protecting the patients' confidentiality. The suggested cryptosystem has the capability of protecting patient photos from being seen by a broker who has their security breached. In specifically, the authors propose a rapid probabilistic cryptosystem to secure medical keyframes that are derived from a wireless capsule endoscopy process using a prioritising mechanism. This system would be used to protect the data. The encrypted pictures that are created by the cryptosystem display a feature known as unpredictability. This behaviour guarantees both computational efficiency and the maximum degree of security possible for the keyframes against a variety of different types of assaults.

Bassam A. Y. Alqaralleh et al [20] provided a deep learning (DL) model for the Internet of Medical Things environment that includes blockchain-assisted secure picture transfer and diagnosis. The model that is being given includes a few different operations, including data gathering, secure transaction, hash value encryption, and data categorization. In the first place, elliptic curve cryptography (ECC) is used, and the optimum key generation for ECC is accomplished using an algorithm that is a hybridization of grasshopper and fruit fly optimization (GO-FFO). After that, the hash values are encrypted using a combination of the neighbourhood indexing sequence (NIS) and the burrow wheeler transform (BWT), which is referred to as NIS-BWT. In the end, a deep belief network, also known as a DBN, is used as part of the classification process in order to identify the presence of illness.

III. Proposed Model

Abraham Lempel, Jacob Ziv, and Terry Welch are the ones responsible for developing the global lossless data compression technique known as LZW (Lempel-Ziv-Welch). In 1984, it was released by Welch as an enhanced implementation of the LZ78 method, which had been published the year before by Lempel and Ziv. The technique was developed to be easy to deploy quickly, but it is not guaranteed to provide the best results since it does not do any analysis on the data. Large English texts would normally be reduced to around half their original size after being compressed using this method.

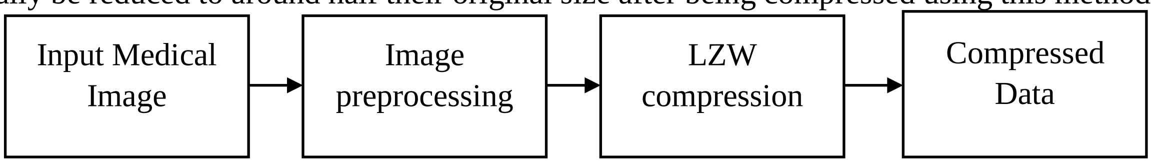


Figure 1: Proposed model

Every day, the digital world creates data at a pace that is measured in the billions of gigabytes. Either this data must be successfully kept, or it must be transferred. Utilizing mathematical analysis and making use of the patterns that are there in the data is one way to accomplish this goal. These patterns may be applied to the data in order to achieve the greatest potential reduction in size. The task at hand is to carry out this action in order to rebuild the initial data whenever it is required without discarding any of the data in the process. Lossless compression is another name for this data management strategy. In the days before the widespread use of computers, this issue arose anytime it was necessary to transmit data across extensive distances. Codebooks were developed as a means of condensing communications without sacrificing any of the information contained inside them. These codebooks transform a certain group of frequently used words into a series of brief letter sequences, sometimes known as codes. This was accomplished with the use of lookup tables. When two persons have access to the same codebooks, they are able to exchange lengthy communications using just the codewords without any of the information being compromised. Specific phrases that are often transmitted are the only ones that are coded, since the codebooks do not include every potential sentence that has to be sent.

In 1978, Lempel Ziv introduced a system in which a fresh codebook is built for each message that has to be conveyed. This approach was conceived by Lempel Ziv. The message is checked for any instances of duplication in advance of its transmission, and an ideal codebook is created to correspond

with that specific message. Following the completion of the message compression process, the codebook is required to be sent together with the compressed message. When added together, the length of the codebook and the data that has been compressed must be less than the length of the initial message. After it has been sent the data, the receiver will consult the codebook in order to decompress the message.
The writers also desired to do away with the step of sending the codebook together with the message that was being sent. In this kind of situation, the codebook will need to be improvised as it goes along. This results in a shorter amount of time needed to develop the codebook. The solution consisted in rethinking the structure of the codebook in its entirety. It is recommended that the codebook and the compressed message be constructed on the fly concurrently rather than scanning the complete message in order to build the codebook.
While the message is being sent, the codebook is being pieced together progressively depending on the letter sequences that are appearing at the appropriate times in the message. At the same time, the compressed message is also constructed by making use of the codebook that has been built up until that point. Because of this, the sender is given the ability to transmit merely the compressed message and not the codebook. During the process of decompressing the message, the receiver is able to construct the identical codebook using the compressed data from the message.

3.1 Encoding

Consider a simple message “PQPQQPQRPQPQQP” that needs to be encoded.
The initial entries in the codebook are:

- One is assigned to P.
- Two is assigned to Q and three is assigned to R. While traversing the message.
- Unique combinations are added to the entry.
- Their corresponding index and encoded output are updated.

The first new item in the codebook is the letter combination PQ, the index that corresponds to it has been changed to four, and the output that has been encoded is represented by the first entry that has been discovered in the code (which is P). As a result, the encoded output is changed to the value one. The next combination that is completely unique is QP; the index is then changed to five, and the encoded output is changed to two to reflect this change. The next sequence in the message that is completely unique is PQQ; hence, the index is changed to six, and the encoded output is changed to five. The procedure is carried out repeatedly until all of the distinct permutations have been brought up to date, after which the complete message is processed. The decoded data that was produced are as follows: 1,2,4,5,2,3,4,6,1. This communication is sent to the transmitter.

Table 1: Initial dictionary

Index	Codeword
1	P
2	Q
3	R

Table 2: codebook

Encoded Output	Index	Entry
-	1	P
-	2	Q
-	3	R
1	4	PQ
2	5	QP
4	6	PQQ
5	7	QPQ
2	8	QR
3	9	RP

4	10	PQR
6	11	PQQR
1	-	-

3.2 Decoding

The received compressed code is 1,2,4,5,2,3,4,6,1. The decoding is illustrated in table 3.

Table 3: LZW decoding data compression

Received	Decode	Update dictionary			
		Index	Entry	index	Partial entries
1	P	--	-	4	P_
2	Q	4	PQ	5	Q_
4	PQ	5	QP	6	PQ_
5	QP	6	PQQ	7	QP_
2	Q	7	QPQ	8	Q_
3	R	8	QR	9	R_
4	PQ	9	RP	10	PQ_
6	PQQ	10	PQP	11	PQQ_
1	P	11	PQQP	12	P_

The first piece of data that was received was a one, and it was decoded as the letter P. The index and entry of the receiver dictionary are both left blank since P is already present in the initial dictionary. The partial index has been changed to reflect the value four, and the partial entry has been given the value P_. The decoded value is now updated to be Q, and the index is now updated to be four since the next number that was received was number two. The entry is updated by adding the first digit of the decoded message to the partial entry; as a result, the entry now has the value PQ (the previous partial entry had the value P_, and the decoded message had the value Q). The value of five is entered into the partial index, and the value of Q_ is entered into the partial entry. The following data that was received is the number four. The decoded value, which is changed to be QP, the index, which is updated to be five, and the entry, which is updated to be PQ have all been modified. The revised value for the partial index is six, and the updated value for the partial entry is PQ_. This procedure is repeated until the whole of the message has been correctly deciphered.

3.3 Proposed LZW compression algorithm

The proposed LZW compressed data is described in this section.

- **Step 1**: Read the input Image
- **Step 2**: Convert the image into a column vector of length N data type *double*
- **Step 3**: Subtract every pixel from its predecessor.

$$Im(i) = Im(i) - Im(i-1)$$

Where Im is the input image vector of length N
i is the pixel index ranging from 2 to N

- **Step 4**: Split the image into two parts of length N

for every pixel in Im

$$Im\_1(i) = Im(i), \quad if \;\; -50 < Im(i) < 50$$
$$Im\_1(i) = 0, \quad otherwise$$

for every pixel in Im

$$Im\_2(i) = 0, \quad if \;\; -50 < Im(i) < 50$$
$$Im\_2(i) = Im(i), \quad otherwise$$

- **Step 5**: Convert Im_1 and Im_2 into signed 8-bit notation (Im_1S and Im_2S), where MSB of the data indicated the sign bit

$$MSB = 1, \;\; Negative\ Number$$

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 3, No. 2, March : 2023

$MSB = 0$, *Positive Number*

- **Step 6**: Compress *Im_1S* and *Im_2S* using LZW compression.

IV. Experimental Results

This section presents the experimental results carried out to validate the proposed model. In the proposed model, an adaptive LZW code is used to dynamically assign codes while monitoring the frequency of occurrence. Any compression technique's effectiveness is determined on the amount of redundancy in the input data. The suggested model adds a pre-processing stage to increase the input data's redundancy. Pre-processing is a reversible process in which the input data can be completely recreated without error. The parameters used to evaluate the obtained results are as follows:

- **Compressed data size in bits**: The number of bits present in the compressed data.
- **Compression ratio**: Compression ratio is defined as the ratio of original size of the data to the compressed size of the data.

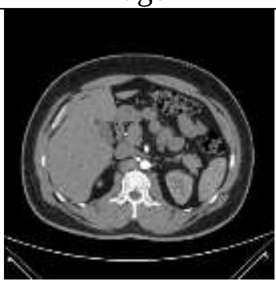
$$Compression\ ratio = \frac{Original\ size\ of\ the\ data}{Compressed\ size\ of\ the\ data}$$

- **Bits per pixel (BPP):** BPP is defined as the ratio between the number of bits in the compressed data to the number of pixels in the input image.

$$Bits\ Per\ Pixel = \frac{Number\ of\ Bits}{Number\ of\ Pixels}$$

The images for the experimental analysis are procured from kaggle.com. The images 1,2 and 3 shown in table 1 are taken from CT KIDNEY DATASET: Normal-Cyst-Tumor and Stone and the images 4,5 and 6 are obtained from Tuberculosis (TB) Chest X-ray Database. The dimension of images in both datasets are 512x512 pixels.

Table 1: Compression results

S. No.	Input Image	Image Dimensions	Size in pixel	Compression size (pixels)	Compression Ratio	BPP
1	 Image 1	512 x 512	2,62,144	94752	2.76	0.3614
2	 Image 2	512 x 512	2,62,144	94591	2.77	0.3608

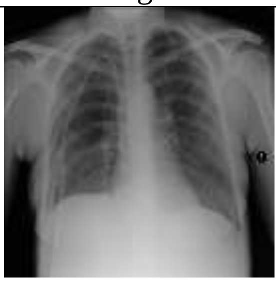
3	 Image 3	512 x 512	2,62,144	91088	2.87	0.3474
4	 Image 4	512 x 512	2,62,144	75746	3.46	0.2889
5	 Image 5	512 x 512	2,62,144	94091	2.78	0.3589
6	 Image 6	512 x 512	2,62,144	73700	3.55	0.2811

Figure 2 shows the size of images (in pixels) after compression.

Figure 2: Compression size of input images

Figure 3 shows the Compression ratio (original size of data/compressed size of data) of the images.

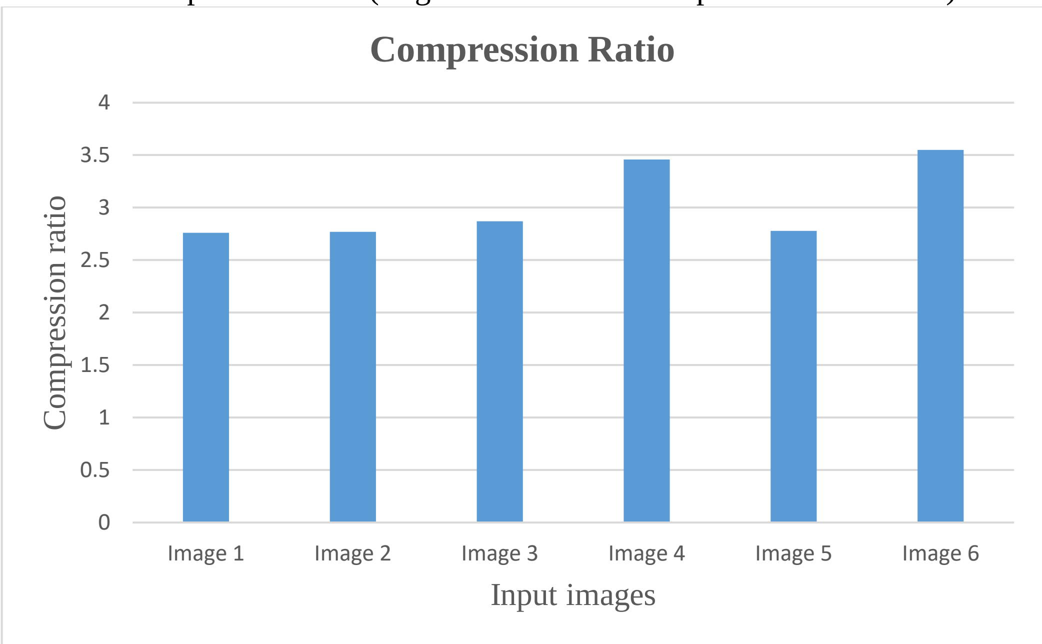


Figure 3: Compression ratio of input images

Figure 4 shows the Bit per pixel (Number of bits/Number of pixel) of the images.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 3, No. 2, March : 2023

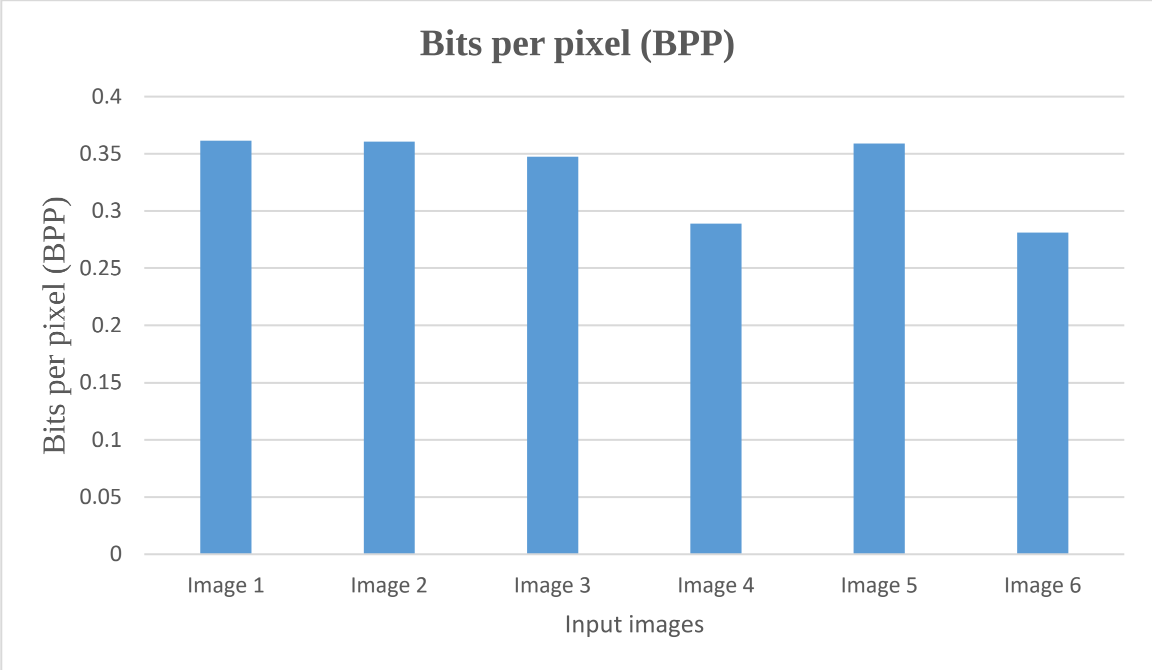


Figure 4: Bits per pixel of input images

The proposed method is compared with existing approach and corresponding results are reported in Table 2. By comparing a new image compression model with existing works, we can evaluate its performance and determine whether it is an improvement over previous method. Performance can be evaluated in terms of image quality, compression ratio, computational complexity, and other metrics. In this paper, compression size and compression ratio considering as performance metrics.

Table 2: Comparison Results

S. No.	**Input Image**	**Size in pixel**	**Compression size(pixels) of Existing Method [21]**	**Compression size (pixels) of Proposed Method**	**Compression Ratio of Existing Method [21]**	**Compression Ratio of Proposed Method**
1	Image 1	2,62,144	1,41,150	94752	1.85	2.76
2	Image 2	2,62,144	1,23,252	94591	2.12	2.77
3	Image 3	2,62,144	1,23,980	91088	2.11	2.87
4	Image 4	2,62,144	1,20,745	75746	2.17	3.46
5	Image 5	2,62,144	1,22,837	94091	2.13	2.78
6	Image 6	2,62,144	1,19,231	73700	2.19	3.55

The proposed LZW compression algorithm is known to provide high compression ratios, meaning that it can compress data to a smaller size without losing any information. This makes it ideal for compressing large files like images. As compared to previous work described in the paper [21], the proposed approach offers a superior compression ratio. The proposed LZW compression and decompression processes are fast and efficient, making it ideal for real-time applications. The proposed LZW compression is a suitable compression technique for medical images due to its ability to provide high compression ratios without any loss of data. It is a critical tool for medical professionals to manage large image datasets and improve patient care.

V. Conclusion

A significant role is played by medical imaging in the process of illness diagnosis. To aid in the diagnosis and treatment of patients, its lossless compression is quite important and directly influences the amount of local storage space and transmission bandwidth that is required by distant medical systems. The LZW compression method is used to demonstrate a lossless compression methodology that is presented in this work. After the initialization procedure, the input picture has the consecutive pixels subtracted off. This results in an increased level of redundancy within the input picture, which in turn results in an increased level of compression efficiency. In comparison to the other compression methods, the LZW compression is much quicker and less complicated. Above 2 is the compression ratio that the suggested approach achieves when applied to medical photos.

References

[1] Bohr, Adam, and Kaveh Memarzadeh. "The rise of artificial intelligence in healthcare applications." In *Artificial Intelligence in healthcare*, pp. 25-60. Academic Press, 2020.
[2] Greiwe, Justin, and Sharmilee M. Nyenhuis. "Wearable technology and how this can be implemented into clinical practice." *Current allergy and asthma reports* 20, no. 8 (2020): 1-10.
[3] Furtado, Denzil, André F. Gygax, Chien Aun Chan, and Ashley I. Bush. "Time to forge ahead: The Internet of Things for healthcare." *Digital Communications and Networks* (2022).
[4] Maseleno, Andino, Wahidah Hashim, Eswaran Perumal, M. Ilayaraja, and K. Shankar. "Access control and classifier-based blockchain technology in e-healthcare applications." In *Intelligent Data Security Solutions for e-Health Applications*, pp. 151-167. Academic Press, 2020.
[5] Hurrah, Nasir N., Shabir A. Parah, Javaid A. Sheikh, Fadi Al-Turjman, and Khan Muhammad. "Secure data transmission framework for confidentiality in IoTs." *Ad Hoc Networks* 95 (2019): 101989.
[6] Saravanan, S., and D. Sujitha Juliet. "A Hybrid Approach for Region-Based Medical Image Compression with Nature-Inspired Optimization Algorithm." In *Innovations in Computer Science and Engineering*, pp. 225-233. Springer, Singapore, 2021.
[7] Kouanou, Aurelle Tchagna, Daniel Tchiotsop, Romanic Kengne, Djoufack Tansaa Zephirin, Ngo Mouelas Adele Armele, and René Tchinda. "An optimal big data workflow for biomedical image analysis." *Informatics in Medicine Unlocked* 11 (2018): 68-74.
[8] Drăgulinescu, Ana Maria Claudia, Adrian Florin Manea, Octavian Fratu, and Andrei Drăgulinescu. "LoRa-based medical IoT system architecture and testbed." *Wireless Personal Communications* 126, no. 1 (2022): 25-47.
[9] Ullah, Ata, Ghawar Said, Muhammad Sher, and Huansheng Ning. "Fog-assisted secure healthcare data aggregation scheme in IoT-enabled WSN." *Peer-to-Peer Networking and Applications* 13, no. 1 (2020): 163-174.
[10] Kumari, Aparna, Sudeep Tanwar, Sudhanshu Tyagi, and Neeraj Kumar. "Fog computing for Healthcare 4.0 environment: Opportunities and challenges." *Computers & Electrical Engineering* 72 (2018): 1-13.
[11] Azar, Joseph, Abdallah Makhoul, Mahmoud Barhamgi, and Raphaël Couturier. "An energy efficient IoT data compression approach for edge machine learning." *Future Generation Computer Systems* 96 (2019): 168-175.
[12] Yildirim, Ozal, Ru San Tan, and U. Rajendra Acharya. "An efficient compression of ECG signals using deep convolutional autoencoders." *Cognitive Systems Research* 52 (2018): 198-211.
[13] Idrees, Sara Kadhum, and Ali Kadhum Idrees. "New fog computing enabled lossless EEG data compression scheme in IoT networks." *Journal of Ambient Intelligence and Humanized Computing* 13, no. 6 (2022): 3257-3270.
[14] Abdellatif, Alaa Awad, Ahmed Emam, Carla-Fabiana Chiasserini, Amr Mohamed, Ali Jaoua, and Rabab Ward. "Edge-based compression and classification for smart healthcare systems: Concept, implementation and evaluation." *Expert Systems with Applications* 117 (2019): 1-14.

[15] Anand, Ashima, and Amit Kumar Singh. "An improved DWT-SVD domain watermarking for medical information security." Computer Communications 152 (2020): 72-80.

[16] Rahmani, Amir M., Tuan Nguyen Gia, Behailu Negash, Arman Anzanpour, Iman Azimi, Mingzhe Jiang, and Pasi Liljeberg. "Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach." Future Generation Computer Systems 78 (2018): 641-658.

[17] Jha, C. K., and M. H. Kolekar. "Empirical mode decomposition and wavelet transform based ECG data compression scheme." *IRBM* 42, no. 1 (2021): 65-72.

[18] Dhanvijay, Mrinai M., and Shailaja C. Patil. "Internet of Things: A survey of enabling technologies in healthcare and its applications." *Computer Networks* 153 (2019): 113-131.

[19] Hamza, Rafik, Zheng Yan, Khan Muhammad, Paolo Bellavista, and Faiza Titouna. "A privacy-preserving cryptosystem for IoT E-healthcare." *Information Sciences* 527 (2020): 493-510.

[20] Alqaralleh, Bassam AY, Thavavel Vaiyapuri, Velmurugan Subbiah Parvathy, Deepak Gupta, Ashish Khanna, and K. Shankar. "Blockchain-assisted secure image transmission and diagnosis model on Internet of Medical Things Environment." *Personal and ubiquitous computing* (2021): 1-11.

[21] Sridhar, A. Phani, and P. V. Lakshmi. "An efficient lossless medical data compression using lzw compression for optimal cloud data storage." Annals of the Romanian Society for Cell Biology 25, no. 6 (2021): 17144-17160.