**An auto-encoder decoder-based approach for the Hydrocephalus MRI dataset compression**

Debkumar Chowdhury1\*

1University of Engineering and Management, Kolkata, India

1[debkumar.cse@gmail.com](mailto:debkumar.cse@gmail.com)

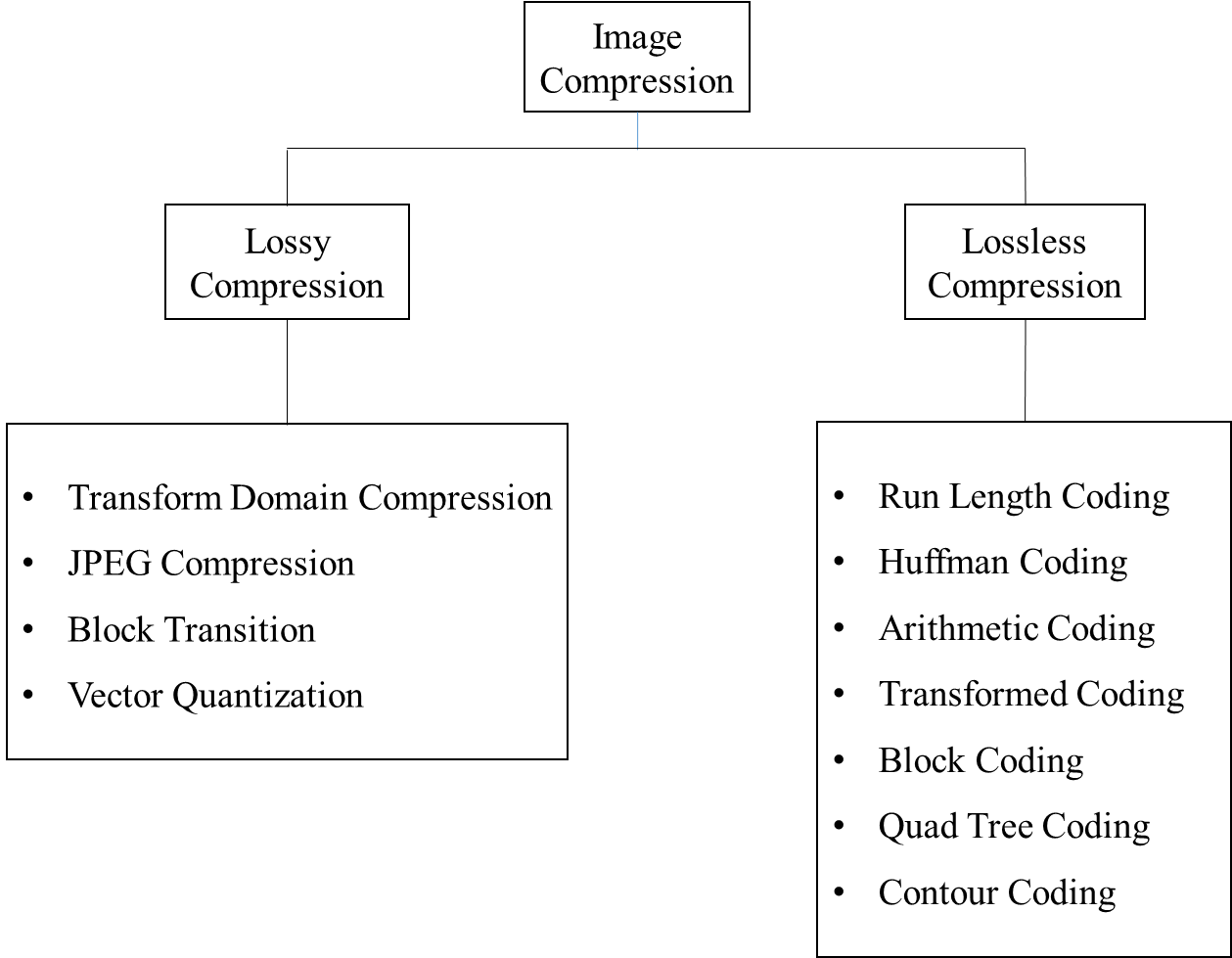
**Abstract.** Image compression is a terminology that is defined by encoding an image file in such a way that the encoded file takes lesser storage than the original file. The objective of the MRI compression is to reduce the size of multiple MRI images and to upload them to the cloud so that someone can securely store them on the web and retrieve them from there when needed. For the last few decades, different researchers have proposed various MRI compression strategies to satisfy the demand of the medical society. This article demonstrates an encoding-decoding MRI compression methodology that is applied to the Hydrocephalus MRI dataset. The compression methodology begins with an MRI preprocessing step where the raw input MRI is taken from the custom dataset and is resized, converted to a specific format, and customized so that it can be delivered to the constructed model as input. In the next steps, a custom dataset is segregated into a training and testing set, so that the proposed model can be trained and tested. In the next phase, the trained proposed auto-encoder decoder-based architecture compresses each Hydrocephalus MRI, taken from the custom dataset. The performance of the proposed auto-encoder decoder-based architecture for the MRI compression is tested in terms of multiple parameters. We compare the existing image compression algorithms with the proposed compression architecture concerning various compressions measuring metrics. From this comparative study, we can conclude that the proposed compression architecture dominates other compression approaches and it has reached the level of state-of-the-art performance. A unique compression technique, an efficient parametric performance, and its application on the Hydrocephalus custom MRI dataset established the proposed model’s uniqueness. In the future, the application of the proposed compression architecture in a cloud-based environment may be implemented.

**Keywords:** Hydrocephalus MRI**,** Auto-encoder decoder-based compression architecture, MRI Preprocessing, Lossless and Lossy MRI compression, Compression Ratio

# Introduction

Image compression is a terminology that is defined by encoding an image file in such a way that the encoded file takes lesser storage than the original file. The objective of image compression is to reduce the size of images efficiently so that we can transmit them over the internet platforms, upload them to the cloud, and serves image restoration purpose meaningfully for future use. We can categorize image compression techniques into two sub-categories, popularly known as lossless and lossy compression techniques. We classify both of the techniques into multiple categories based on the way of compression.

A full pictorial representation of image compression classification can be observed with the help of **Figure. 1.**

****

**Figure 1.** Shows various image compression classification techniques

The advantage of image compression is that it enables us to store more images with less storage usage conveniently on personal computers or servers and retrieve them as and when required. There are also several disadvantages of image compression, such as degraded image quality, and important details in the image lost that might be hard to recover. A significant amount also lost data while compressing images, and this might prove to be a deterrent to us sometimes. In a dataset, all the images need to be compressed, keeping the same compression rate in mind. Variable compression rates for different images may not be useful for us when training the model. During and after image compression, the researchers may face multiple challenges. When an image is compressed, the finer details of color, contrast, and sharpness are reduced. Hence, the encoded image quality may be drastically reduced in comparison with the original image. We may observe that even after decoding, the decoded image may not remain the same as the original image as, ideally, it should be. One such observation can be noticed in the **Figure. 2** given below.

|  |  |  |
| --- | --- | --- |
| C:\Users\User\Desktop\1.png |  |  |
| (a) | (b) | (c) |

**Figure 2** shows a brain MRI compression result (a) original image (b) encoded image (c) decoded image

For the diagnosis of various brain diseases patients in the government, public, and private hospitals multiple MRI or CT images are generated. One such brain disease is hydrocephalus. When an MRI scan is done on a hydrocephalus suspected patient, lots of images are produced as a result. These images are of extremely high quality, and hence consume a lot of space. Thus, each set of images, for just one scan on a particular area of a patient’s body, is going to consume a lot of space in the given storage medium. Our objective is to build an MRI compression model so that the MRI images of hydrocephalus suspected patients can be compressed and archived easily, effectively, and efficiently. It has been found that majorly the hospitals and medical centers in rural areas are using outdated devices, or devices having low-end specifications to store MRI images of hydrocephalus or other brain disease patients as a result of low budget and lack of funding. In this situation, we are going to propose a new auto encoding-decoding-based MRI compression model, which may prove to be beneficial for archiving and compressing MRI images of hydrocephalus patients. Our proposed model is capable of compressing the Hydrocephalus MRI images at a faster rate. Compressed images take very less time for transmission on the web, over the cloud, and on the server. Our model utilizes less storage space, which in turn helps us to store more images in the hard disk (or any other storage medium). When transferring these compressed images, less bandwidth is consumed due to its smaller file size, and as a result, less cost is incurred.

This article is divided into various sections. In section 2, some of the latest work allied to the proposed article title and some of the major and popular compression techniques as mentioned in **Figure. 1** is highlighted and briefly explained. Section 3 of the article, highlights the proposed methodology, built-in architecture, and the flow of the overall procedure. Section 4 of the article portrays various tables, graphs, and outputs to establish the performance of the compression model in comparison with some of the other popular and established compression methodologies.

# Existing Methodologies

Finding an image compression algorithm on Hydrocephalus MRI is very difficult and rare. As we are dealing with image compression algorithms and MRI images, hence this section first of all we are trying to highlight some of the existing, popular, and general image lossy and lossless compression algorithms along with their pros and cons, performance analysis in context to our problem definition. All these generic lossy and lossless image compression techniques' performance and analysis can be observed in the **Table. 1.** Secondly, it has been observed that over the years multiple methodologies related to medical image compression have been proposed by different researchers. Though it has been found that rarely a very few researchers are going on the Hydrocephalus MRI compression technique, still various researchers work on MRI compression we have considered and analyzed as well. All these existing methods' performance and analysis are highlighted in **Table. 2.**

The Lempel–Ziv–Welch (LZW) [1] is an efficient lossless generic compression strategy that provides a satisfactory compression ratio after being applied to a larger generic dataset. But it takes a larger amount of compression and decompression time and sometimes provides inappropriate and unacceptable images. The Huffman Coding [1] is a lossless generic compression technique that may be applied to MRI or CT image datasets. The technique suffers from a low compression ratio and is applicable for a small amount of data. The Embedded Zerotree Wavelet (EZW) method [2], which has been proposed by Shapiro, is a lossy generic algorithm applicable to data. Hence, the feature of the original image may not persist in the decompressed image. The Run Length Encoding (RLE) [21] is the easiest and most effective generic lossless compression algorithm applied to data and images. It has been observed that the RLE algorithm is providing almost the same Hydrocephalus image after compression from the original image. The Shannon Fano coding scheme [3] is highly recommended when the entire code block of the data is not demanded. The scheme fails to give any guarantee on the generation of codes, or generation of optimal codes and produces no unique code. Arithmetic Coding [5] is a lossless compression algorithm which is invented to compress data not images like CT or MRI. When applied to Hydrocephalus MRI it has been observed that the compression ratio is too low in contrast with other existing lossless generic compression algorithms. We have also concluded that some of the distinguishable features of the original Hydrocephalus MRI are lost due to this type of compression ratio and decompression.

A detailed analysis of these generic algorithms, discussed above is explained in **Table. 1.**The performance column of this table shows the performance of the generic compression algorithm in terms of the Hydrocephalus MRI dataset, in the representation of compression ratio.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Serial No**. | **Year** | **Author(s)/Company** | **Compression Algorithm** | **Remark** | **Performance** |
| 1 | 2020 | Abraham Lempel et. al. | LZW [1] | A generic algorithm, not Hydrocephalus MRI compression specific | Compression Ratio = 4.203 |
| 2 | 2019 | David Huffman | Huffman Coding[1] | A generic algorithm, not Hydrocephalus MRI compression specific | Compression Ratio = 6.71 |
| 3 | 1993 | Shapiro et. al. | EZW [2] | Generic algorithm, existing feature of the Hydrocephalus disease may be destroyed due to compression-decompression strategy | Compression Ratio = 3.664 |
| 4 | 1983 | Hitachi Corporation | RLE [21] | A generic algorithm, provides dissatisfactory compression results on Hydrocephalus MRI | Compression Ratio = 3.55 |
| 5 | 1948 | Claude E. Shannon et. al | Shannon–Fano[3] | A generic algorithm, invented to compress data, not Hydrocephalus MRI | Compression Ratio = 3.9825 |
| 6 | 1987 | Youngjun Yoo et. al | Arithmetic Coding[5] | A generic algorithm, huge space complexity, and lower application results in Hydrocephalus MRI | Compression Ratio = 3.36 |

**Table 1.** A detailed analysis of different generic lossless and lossy compression techniques for MRI compression

Besides, various generic lossless and lossy image compression techniques, a detailed analysis of some of the recent research works allied to this domain are highlighted in the **Table. 2.** where performance measurement is computed in terms of compression ratio, SNR, PSNR, etc. Alex Cazañas-Gordón ***et. al.***[11] explained in their article, a brief overview of different digital compression techniques applied on medical images like CT or MRI, which highlighted various digital compression tools, their efficient usage, procedures for medical image achievement, and different transmission methodologies for medical images. Despite of above advantages, the article suffers from a major drawback. It has been observed that there can be a chance of legal ramifications due to the use of lossy compression. A technique of medical image compression using DCT with entropy encoding and Huffman on MRI brain images [1] is proposed, which is capable of delivering compressed medical images with minimal quality loss and the result of this strategy shows a downfall in image quality due to decreasing encoding time. The wavelet with particle swarm optimization [13] is a strategy for medical image compression, which is capable of delivering superior performance in medical images with high detail, but it lacks intelligence techniques that accelerate the compression technique to a convergent nature. A deep auto-encoder with Boltzmann machine training architecture [14] is used to reconstruct CT and MRI images. Though this technique retains the structural quality of medical images, its computational time is comparatively higher than other transform-based methodologies. An arithmetic coding-dependent compression strategy [5] is developed which is based on automatic extraction of tumour region for ROI compression which provides information such as tumour size and location etc. Modular optimization of joint reconstruction and coding strategy [16] is proposed to perform regularized compression on MRI, where it is noticed that decompressed images are generated without additional processing and optimization strategy is suffered due to lack of automated parameters. A detailed analysis of Huffman Coding and Lempel–Ziv–Welch (LZW) Coding [1] is performed to highlight the strategy of data compression. The analysis is successful to establish, a good compression ratio, and application on a large dataset, of both the techniques but, failed to establish low time consumption for both the strategies over the small dataset. Multiple wavelets at different levels of discrete wavelets transform [17] are proposed to perform MRI image compression. The method produces significantly better results with low-frequency images but takes high computational time for high-quality images. Alzheimer's disease is characterized [18] using the image compression technique. The method avoids alteration of information present in images, but the inconsistency in the decomposition can be noticed. An improved image compression using lossless Huffman encoding [1] is proposed which requires less compression and decompression time and is appropriate for smaller data, but provides less compression ratio and proves to be inappropriate for larger data. An overview of lossless image compression techniques [3] is highlighted to show different generic lossless image compression strategies. A fast fractal-based compression [20] is used to reduce the compression time drastically, but the strategy is proved to be error-prone and low performing concerning some of the images. A combination of RLE, LZW, and Adaptive Variable Length Coding [21] is applied to MRI images for compression. Multiple Image Compression in Medical Imaging Techniques using Wavelets for Speedy Transmission and Optimal Storage [22] is capable to perform on low-resolution images at low bit rate, but at very low bit rates, coding of lossy compression scheme shows the blurry part in the background. A tri-mode dual-level 3-D image compression [23] over medical MRI images is capable of retaining the reconstructed image quality, but the technique does not work on colored images. The Embedded Zero-tree Wavelet [22] method is allowing the user to control their desired bit rate, but the properties of the image affect the performance. The SWT [24], which is used to specialize in a lossless compression function on 3D brain images, works only on brain scan images, but not on scanned images of other body parts. The Seam Carving Method [25] works well in eliminating irrelevant data, but its non-optimal compression level may lead to the loss of critical features in medical images. The Hilbert Space-Filling Curves [26] is effective in encoding because of enhancing the locality of differences due to Hilbert's space-filling curve, but it yields better results on only non-medical images such as colored human portrait images and scenery images.

A detailed analysis of these proposed algorithms, discussed above is explained in **Table. 2.** The performance column of this table shows the performance of the various type of algorithms, proposed by the researchers over the last two decades. These compression algorithms' performances are computed in terms of compression ratio. The compression ratio, of all the proposed methodologies discussed below, is calculated after the application of the proposed algorithms over various medical image datasets, except the Hydrocephalus MRI dataset.

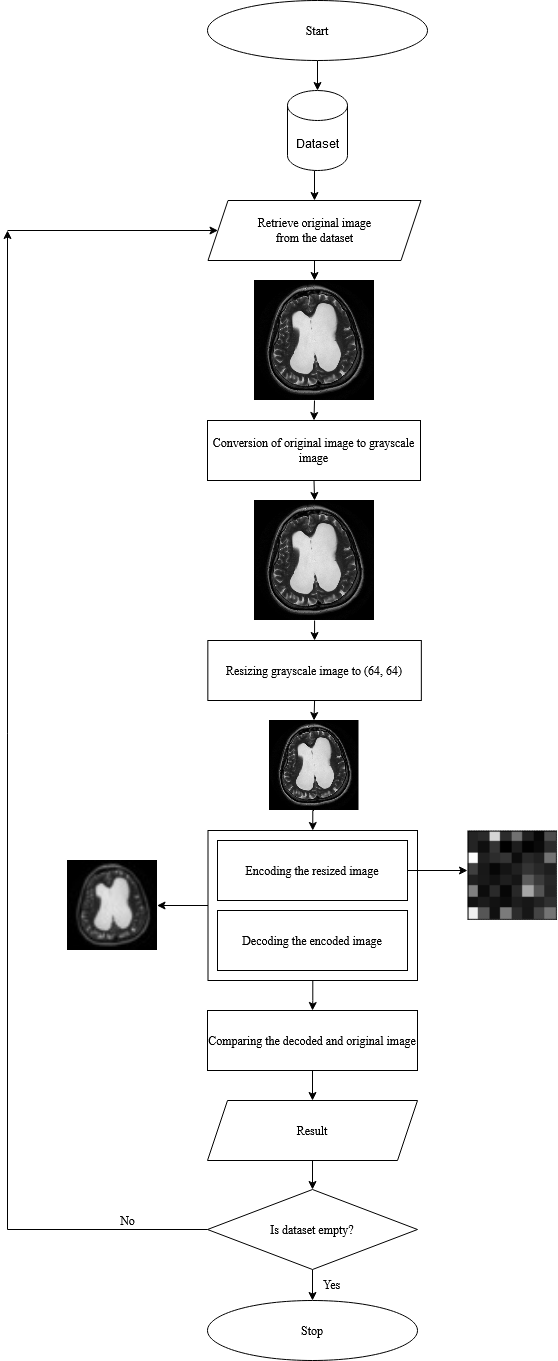
**Table 2.** A detailed analysis of different medical image compression algorithms, proposed by the researchers over the last two decades

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SNo.** | **Year** | **Author(s)** | **Main Methodology** | **Remark** | **Performance** |
| 1 | 2022 | Alex Cazañas-Gordón et. al. | Overview/Review [11] | No novel methodology is proposed, Application of compression tools on Hydrocephalus MRI images is not computed | Compression Ratio = 4.546. |
| 2 | 2022 | J.Rani et. al. | DCT with Entropy Encoding and Huffman [12] | Application of compression strategy concerning Hydrocephalus MRI images are not computed | Compression Ratio = 3.576 |
| 3 | 2021 | Monagi H. Alkinani et. al. | Wavelets with Particle Swarm Optimization [13] | Application of compression strategy concerning Hydrocephalus MRI images is not computed, Relatively less intelligence system. | Compression Ratio = 3.534 |
| 4 | 2021 | Saravanan. S et. al. | Deep  Auto-encoder with Boltzmann Machine Training  Architecture [14] | With high computational time and no application on Hydrocephalus MRI images, the Application of auto-encoder for MRI compression is allied to our defined problem | PSNR = 11.3 |
| 5 | 2021 | Prakash Tunga P. et. al. | Arithmetic coding dependent compression strategy [15] | No application on Hydrocephalus MRI images | Compression Ratio = 3.36. |
| 6 | 2020 | Veronica Corona et. al. | Modular Optimization of Joint Reconstruction and Coding [16] | No application on Hydrocephalus MRI images | PSNR = 10.41. |
| 7 | 2020 | Gajendra Sharma | Analysis of Huffman Coding and Lempel–Ziv–Welch (LZW) Coding [1] | The authors did not apply the strategies to the Hydrocephalus MRI images, Generic Strategies are applied | Huffman Coding’s Compression Ratio = 6.319 and LZW’s Compression Ratio = 4.203 |
| 8 | 2020 | Narayana Prakash S et. al. | Multiple wavelets at different  levels of discrete wavelets transform [17] | No application on Hydrocephalus MRI images | Compression Ratio = 4.005. |
| 9 | 2020 | R Pandian et. al. | Characterization of Alzheimer's MRI Image [18] | No application on Hydrocephalus MRI images | Compression Ratio = 4.3 |
| 10 | 2019 | Nirajan Bist et. al. | Lossless Huffman Encoding [19] | No application on Hydrocephalus MRI images | Compression Ratio = 5.71 |
| 11 | 2019 | Md. Atiqur Rahman et. al. | Overview/Review [3] | No application on Hydrocephalus MRI images | Shannon Fano Encoding’s Compression Ratio = 3.9825 |
| 12 | 2019 | Shuai Liu et. al. | Fast Fractal based Compression [20] | No application on Hydrocephalus MRI images | Compression Ratio = 5.75. |
| 13 | 2019 | Nassir H. Salman et. al. | Run Length Encoding + LZW + Adaptive Variable Length Coding [21] | No application on Hydrocephalus MRI images | Compression Ratio = 3.55. |
| 14 | 2019 | Ruchi Agarwal et. al. | Wavelets for Speedy Transmission and Optimal Storage [22] | No application on Hydrocephalus MRI images | Compression Ratio = 4.375 |
| 15 | 2017 | D. J. Ashpin Pabi et. al. | Tri-mode dual-level 3-D image compression [23] | No application on Hydrocephalus MRI images | Compression Ratio = 3.404. |
| 16 | 2014 | A.M.Raid et. al. | Embedded Zero-Tree Wavelet [2] | No application on Hydrocephalus MRI images | Compression Ratio = 3.664 |
| 17 | 2014 | V. Anusuya et. al. | SWT [24] | No application on Hydrocephalus MRI images | Compression Ratio = 3.752 |
| 18 | 2013 | B. D. Mokal et. al. | Seam Carving Method [25] | No application on Hydrocephalus MRI images | Compression Ratio = 3.435. |
| 19 | 2008 | Jan-Yie Liang et. al. | Hilbert Space-Filling Curves [26] | No application on Hydrocephalus MRI images | Compression Ratio = 3.77 |

All these methods are well appreciated, but in context with our problem, the results can be improved. An in-depth analysis of these methodologies has proven to be very competent, to identify the downsides of these methods. It can be concluded from the second analysis table that none of the methods are applied toHydrocephalus MRI images. Moreover, some of the researchers have used a generic compression algorithm or combination of them and calculated the compression ratio over heterogeneous medical image datasets, which clearly defines their lack of novelty. Other architectures may be unique, but they are suffered from lots of pitfalls and lack of application in the Hydrocephalus MRI dataset. Though an auto-encoder-based approach is proposed, which is allied to our objective, due to its various drawbacks and lack of application to the Hydrocephalus MRI dataset, it is incapable to fulfil the demand of our objective. Identification of these drawbacks helps us to update, and modify our proposed compression-decompression algorithm and code and to calculate the performance parameters.

# Proposed Methodology

Our proposed methodology focused on medical image compression from a custom hydrocephalus dataset using an image compression algorithm. We have proposed a block diagram to show the main concept of the methodology at a glance as shown in **Figure 3.**



**Figure 3.** The built-in architecture of MRI compression technique.

* 1. **Algorithm**

Our algorithm is divided into 8 steps. The algorithm takes a custom hydrocephalus dataset, as an input from the dataset and produces an encoded image and a decoded image. The algorithm is as follows:

**Algorithm:** An unstructured MRI compression technique

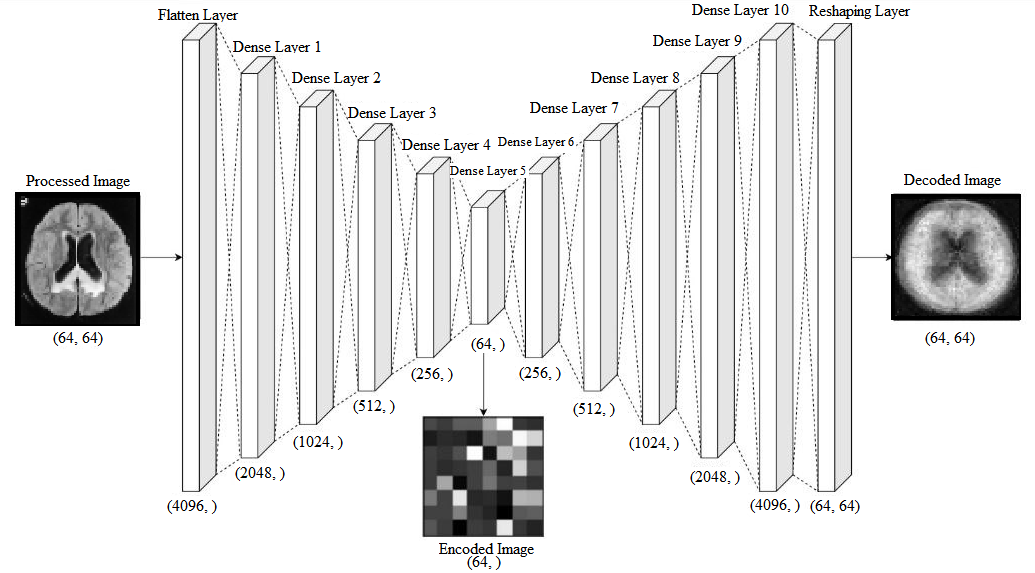
**Input:**

**Output:**  encoded image and decoded image

1. Read an image from the dataset.
2. Convert the image to a grayscale image,.
3. Resize into a 64 X 64 image.
4. The proposed architecture each has an input x of size 64x64 fed into it. In the architecture following layers are observed:
   1. Encoder Architecture
   2. One input layer which accepts input of size 64x64.
   3. One flatten layer.
   4. One dense layer of output size 2048 with an activation leakyReLU function.
   5. One dense layer of output size 1024 with an activation leakyReLU function.
   6. One dense layer of output size 512 with an activation leakyReLU function.
   7. One dense layer of output size 256 with an activation leakyReLU function.
   8. One dense layer of output size 64 with an activation leakyReLU function.
   9. Decoder Architecture
      1. One input layer which accepts input of size 64.
      2. One dense layer of output size 256 with an activation leakyReLU function.
      3. One dense layer of output size 512 with an activation leakyReLU function.
      4. One dense layer of output size 1024 with an activation leakyReLU function.
      5. One dense layer of output size 2048 with an activation leakyReLU function.
      6. One dense layer of output size 4096 with an activation leakyReLU function.
      7. One reshaping layer of output size 64x64.
5. Segregate the whole dataset into two sections. We select 80% MRI from the dataset for training purposes, 10% MRI from the dataset for validation purposes and 10% MRI from the dataset for testing purposes.
6. We feed the training and testing data into the designed model architecture to generate the result.

**3.2 Model Architecture**

The training and testing phase can be explained with the help of a model architecture as shown in **Figure 4.** The architecture accepts preprocessed MRI and displays encoded and decoded images along with performance parameters for the same.

**Figure 4.** Training and testing phase using proposed model architecture

1. **Experimental results**

In this section, first of all we are representing various types of Hydrocephalus and non-Hydrocephalus MRI image grids from our custom dataset. Secondly, we are representing a sample original image (collected from our custom dataset), a greyscale image, and a resized image in order to show the output of different image conversion stages applied in our algorithm. In the next section we have displayed **Table 3** which represents training phase analysis. After that we have demonstrated the result of our algorithm, with the help of **Table 4,** after applying it to various types of Hydrocephalus and non-Hydrocephalus MRI images which are collected randomly from our custom dataset. This section represents the original image, greyscale image, resized image, encoded image, and decoded image of each randomly collected samples. In the next section we have demonstrated various types of performance parameters such as Compression Ratio (CR) or Mean Square Error (MSE) etc. which plays a pivotal rule to measure the performance value of our proposed algorithm. The performance parameters are represented by **Table 4**. In the next sub section, the resultant value of various performance parameters such as BPP, CR, MSE, SSIM, PSNR etc. are calculated for various randomly collected MRI images on our dataset. Various performance parameters and their output values for those randomly collected MRI images are shown in **Table 5.** In the next sub section, we compare our algorithm with other popular existing compression algorithms such as LZW, EZW, RLE etc. in terms of a key performance parameter, which is known as compression ratio (CR). **Table 6** demonstrates the details analysis and explanation in this regard. In the next sub section, **Figure 7** is represented, which demonstrates 6 individual comparative graphs, such as a graph on compression ratio analysis, a graph on Mean Square Error analysis, a graph on Structure Similarity Index analysis, a graph on Peak Signal to Noise Ratio analysis, a graph on Signal to Noise Ratio analysis and a graph on Root Mean Squared Error analysis. With the help of these graphs, we have tried to establish the superiority of our proposed algorithm over other existing popular compression methodologies in terms of various performance parameters such as CR, MSE, SSI, PSNR, RMSE and SNR.

We have executed our algorithm in a Dell laptop which consists of a hardware configuration like, IntelCore i3-5005U processor, 4GB DDR3 primary memory (RAM), 480GB KINGSTON SATA-III SSD, Intel® HD Graphics 5500.

The software configuration is as follows – we have used Windows 10 Operating System. We have implemented our algorithm in Python 3. We executed our algorithm in Jupyter Notebook v6.3.0. We use Anaconda as a distributor of Python v3.8.6.

We consider a custom hydrocephalus dataset, which contains 228 images. The size, color, and format of images in the dataset are similar, whereas the resolutions of the images are different. The format of the images is ‘.jpg’ by nature.

After reading the images from the local machine we have achieved the following results as shown in **Figure. 5.**

|  |  |
| --- | --- |
| D:\Telegram Desktop\image_2022-08-08_17-40-44.png | D:\Telegram Desktop\image_2022-08-08_17-40-57.png |
| (a) | (b) |

**Figure. 5.** Samples of Hydrocephalus dataset (a) Hydrocephalus (b) Non Hydrocephalus

After reading all the images from the dataset each image will pass through the various steps of the algorithm as shown in **Figure 6.**

|  |  |  |
| --- | --- | --- |
|  | | |
| (a) | (b) | (c) |

**Figure 6.** (a) Original image (b) gray-scale image (c) Resized image

We are splitting our dataset into training and testing sets. 80% of the MRIs are used for training and the rest are used for testing. After the execution of our proposed algorithm, we observed that the number of training samples is 104, the number of testing samples is 26 whereas training shape values are (104, 28, 28). Our proposed layered architecture is trained. It is used to train on n number of samples (104 in our case). With the changes of several iterations or epochs, the total time consumed by the algorithm is calculated.

The table below shows the time taken, loss and value loss for a sample of 10 epochs from the dataset.

**Table 3:** An example of the training phase.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SNo.** | **Epoch No.** | **Time Taken (s)** | **Loss** | **Value Loss** |
| 1. | 1 | 2 | 0.2193 | 0.1604 |
| 2. | 2 | 1 | 0.1407 | 0.1219 |
| 3. | 3 | 1 | 0.1159 | 0.1086 |
| 4. | 4 | 1 | 0.0517 | 0.0842 |
| 5. | 5 | 1 | 0.0494 | 0.0835 |
| 6. | 6 | 1 | 0.0473 | 0.0831 |
| 7. | 7 | 1 | 0.0461 | 0.0828 |
| 8. | 8 | 1 | 0.0459 | 0.0821 |
| 9. | 9 | 1 | 0.0455 | 0.0817 |
| 10. | 10 | 1 | 0.0450 | 0.0814 |

We have selected 6 images each of Hydrocephalus category and non-Hydrocephalus category from the dataset. In **Table 4** we have displayed Original, Greyscaled, Resized, Encoded and Decoded images of the selected samples.

**Table 4:** Original, Greyscaled, Resized, Encoded and Decoded Image samples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Hydrocephalus** | **Original** | **Greyscaled** | **Resized** | **Encoded** | **Decoded** |
| YES | **D:\Telegram Desktop\image_2022-08-05_18-09-04.png** | | | | |
| YES | **D:\Telegram Desktop\image_2022-08-05_18-09-11.png** | | | | |
| YES | **D:\Telegram Desktop\image_2022-08-05_18-09-18.png** | | | | |
| YES | **D:\Telegram Desktop\image_2022-08-05_18-09-33.png** | | | | |
| YES | **D:\Telegram Desktop\image_2022-08-05_18-09-53.png** | | | | |
| YES | **D:\Telegram Desktop\image_2022-08-05_18-09-58.png** | | | | |
| NO | **D:\Telegram Desktop\image_2022-08-05_18-14-23.png** | | | | |
| NO | **D:\Telegram Desktop\image_2022-08-05_18-14-33.png** | | | | |
| NO | **D:\Telegram Desktop\image_2022-08-05_18-14-52.png** | | | | |
| NO | **D:\Telegram Desktop\image_2022-08-05_18-15-06.png** | | | | |
| NO | **D:\Telegram Desktop\image_2022-08-05_18-15-26.png** | | | | |
| NO | **D:\Telegram Desktop\image_2022-08-05_18-19-51.png** | | | | |

* 1. **Performance parameters**

In this work, the compression techniques use a wide number of performance measures to compute their efficiency and performance.

**Table 5:** Various performance parameters

|  |  |  |
| --- | --- | --- |
| **Serial No**. | **Name of the performance parameter** | **Equation** |
| 1. | Compression Ratio | CR= |
| 2. | Mean Square Error | MSE= |
| 3. | Bits per Pixel | BPP = |
| 4. | Structure Similarity Index | SSIM = |
| 5. | Correlation coefficient | CC = |
| 6. | Peak Signal to Noise Ratio | PSNR=20 |
| 7. | Percent rate of distortion | PRD = |
| 8. | Structural Content | SC = |

**Table 6:** Calculation of performance parameters for each image

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SNo** | **BPP** | **Compression Ratio** | **MSE** | **SSIM** | **PSNR** | **PRD** | **Structural Content** | **CC** |
| 1. | 338 | 8.047 | 3.251 | 0.070 | 10.79 | 99.665% | 10.081 | 0.0040 |
| 2. | 333 | 8.117 | 3.598 | 0.235 | 16.605 | 99.55% | 475.408 | 0.0065 |
| 3. | 333 | 7.641 | 4.144 | 0.192 | 12.585 | 99.541% | 129.134 | 0.0072 |
| 4. | 341 | 9.327 | 3.767 | 0.083 | 12.4 | 99.659% | 4.243 | 0.0020 |
| 5. | 339 | 8.605 | 4.009 | 0.227 | 12.681 | 99.625% | 3.669 | 0.0023 |
| 6. | 338 | 8.214 | 3.786 | 0.202 | 11.810 | 99.569% | 4.255 | 0.0028 |
| 7. | 337 | 7.941 | 3.995 | 0.180 | 10.25 | 99.235% | 17.490 | 0.0037 |
| 8. | 340 | 8.512 | 4.023 | 0.221 | 12.011 | 99.368% | 14.142 | 0.0069 |
| 9. | 336 | 8.018 | 3.865 | 0.199 | 13.562 | 99.415% | 9.927 | 0.0044 |
| 10. | 341 | 7.867 | 3.912 | 0.215 | 15.164 | 99.581% | 10.001 | 0.0038 |
| 11. | 340 | 7.747 | 4.109 | 0.175 | 11.899 | 99.612% | 15.819 | 0.0053 |
| 12. | 339 | 8.004 | 4.096 | 0.204 | 9.549 | 99.440% | 13.521 | 0.0036 |

**Table 6** displays the performance parameters of 12 samples shown in **Table 5** from the dataset. In the end, we have found that our proposed algorithm can be used for medical image compression from MRIs. As the data is balanced, we are considering the results to be satisfied. Comparing it with other existing methodologies a satisfactory result is observed as shown in **Table 7** to **Table 12.** We can conclude that our proposed methodology overpowers the existing compression algorithms. [1][2][3][5]

**Table 7:** Comparison chart (SNR)

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Serial No**. | **Name of the compression algorithm** | **SNR** |
| 1. | LZW [1] | 16.7 |
| 2. | Huffman coding [1] | 18.1 |
| 3. | EZW [2] | 17.7 |
| 4. | RLE [3] | 17.0 |
| 5. | Shannon–Fano [3] | 16.5 |
| 6. | Arithmetic coding [3][5] | 17.2 |
| **7.** | **Proposed methodology** | **18.6** |

**Table 8:** Comparison chart (PSNR)

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Serial No**. | **Name of the compression algorithm** | **PSNR** |
| 1. | LZW [1] | 10.321 |
| 2. | Huffman coding [1] | 11.235 |
| 3. | EZW [2] | 9.42 |
| 4. | RLE [3] | 10.505 |
| 5. | Shannon–Fano [3] | 8.663 |
| 6. | Arithmetic coding [3][5] | 12.935 |
| **7.** | **Proposed methodology** | **16.75** |

**Table 9:** Comparison chart (MSE)

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Serial No**. | **Name of the compression algorithm** | **MSE** |
| 1. | LZW [1] | 5.25 |
| 2. | Huffman coding [1] | 4.117 |
| 3. | EZW [2] | 6.996 |
| 4. | RLE [3] | 3.507 |
| 5. | Shannon–Fano [3] | 4.526 |
| 6. | Arithmetic coding [3][5] | 6.001 |
| **7.** | **Proposed methodology** | **3.166** |

**Table 10:** Comparison chart (RMSE)

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Serial No**. | **Name of the compression algorithm** | **RMSE** |
| 1. | LZW [1] | 2.291 |
| 2. | Huffman coding [1] | 2.029 |
| 3. | EZW [2] | 2.645 |
| 4. | RLE [3] | 1.873 |
| 5. | Shannon–Fano [3] | 2.127 |
| 6. | Arithmetic coding [3][5] | 2.45 |
| **7.** | **Proposed methodology** | **1.779** |

**Table 11:** Comparison chart (Compression Ratio)

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Serial No**. | **Name of the compression algorithm** | **Compression ratio** |
| 1. | LZW [1] | 6.319 |
| 2. | Huffman coding [1] | 4.203 |
| 3. | EZW [2] | 3.6647 |
| 4. | RLE [3] | 3.5766 |
| 5. | Shannon–Fano [3] | 3.9825 |
| 6. | Arithmetic coding [3][5] | 4.2059 |
| **7.** | **Proposed methodology** | **9.422** |

**Table 12:** Comparison chart (SSIM)

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Serial No**. | **Name of the compression algorithm** | **SSIM** |
| 1. | LZW [1] | 0.092 |
| 2. | Huffman coding [1] | 0.157 |
| 3. | EZW [2] | 0.296 |
| 4. | RLE [3] | 0.350 |
| 5. | Shannon–Fano [3] | 0.199 |
| 6. | Arithmetic coding [3][5] | 0.236 |
| **7.** | **Proposed methodology** | **0.070** |

|  |  |
| --- | --- |
|  | C:\Users\User\Desktop\PSNR.png |
| (a) | (b) |
| C:\Users\User\Desktop\MSE.png | C:\Users\User\Desktop\RMSE.jpg |
| (c) | (d) |
| C:\Users\User\Desktop\CR.png | C:\Users\User\Desktop\SSIM.png |
| (e) | (f) |

**Figure 7.** Comparing existing methodologies and proposed methodology based on (a) SNR (b) PSNR (c) MSE (d) RMSE (e) CR (f) SSI

1. **Conclusion**

In this paper, we have proposed a compression algorithm. This algorithm is capable of compressing and storing generated images from medical equipment contained in the custom hydrocephalus dataset, collected from the web resource. It is responsible for encoding the original image and decoding the image, and finally comparing the decoded and original image. In addition, we have also calculated BPP, Compression ratio, MSE, SSIM, PSNR, PRD, Structural content, and CC values, through which we can compare our proposed method with existing methods [1] [2] [3] [5]. The experimental result shows that after applying the proposed and existing methods [1] [2] [3] [5] to the Custom hydrocephalus dataset, our technique is producing a Compression ratio of 9.422. This result is considered to be satisfactory, and based on this result we can say that the proposed algorithm exceeds the efficiency of the existing methodologies.

Our proposed methodology is the first of its kind where image compression has been applied on the custom hydrocephalus dataset. Additionally, the compression technique has been applied for the first time in the field of medical image compression.

Due to its high performance, novelty, and ease of use, our proposed method is useful to develop any mobile or web applications in the future. Our technique can be used to encode an image and store in a remote web server and decode the requested image as and when required.

Our method can be tested on various medical equipment-generated image datasets to identify the generic performance of the proposed method in the future. The performance of our method could be increased by making necessary modifications to the algorithm.

1. **References**
2. Gajendra Sharma, “Analysis of Huffman Coding and Lempel–Ziv–Welch (LZW) Coding as Data Compression Techniques,” International Journal of Scientific Research in Computer Science and Engineering, Vol.8, Issue.1, pp.37-44, 2020.
3. A.M.Raid, W.M.Khedr, M. A. El-dosuky, and Wesam Ahmed, “Image compression using embedded zero tree wavelet” Signal & Image Processing: An International Journal (SIPIJ) Vol.5, No.6, December 2014, pp. 33-39, DOI: 10.5121/sipij.2014.5603
4. Rahman, Md. A., and Mohamed Hamada. 2019. "Lossless Image Compression Techniques: A State-of-the-Art Survey" Symmetry 11, no. 10: 1274. DOI : 10.3390/sym11101274
5. S. S. Yu, M. N. Wernick, and N. P. Galatsanos, "Lossless compression of multi-dimensional medical image data using binary-decomposed high-order entropy coding," Proceedings of 1st International Conference on Image Processing, 1994, pp. 351-355 vol.2, DOI: 10.1109/ICIP.1994.413590.
6. G. Langdon and J. Rissanen, "Compression of Black-White Images with Arithmetic Coding," in IEEE Transactions on Communications, vol. 29, no. 6, pp. 858-867, June 1981, doi: 10.1109/TCOM.1981.1095052.
7. A. K. Jain, "Image data compression: A review," in Proceedings of the IEEE, vol. 69, no. 3, pp. 349-389, March 1981, doi: 10.1109/PROC.1981.11971.
8. P. Roos, M. A. Viergever, M. C. A. van Dijke and J. H. Peters, "Reversible intraframe compression of medical images," in IEEE Transactions on Medical Imaging, vol. 7, no. 4, pp. 328-336, Dec. 1988, doi: 10.1109/42.14516.
9. Z. Xiao and C. Zheng, "Medical Image Fusion Based on the Structure Similarity Match Measure," 2009 International Conference on Measuring Technology and Mechatronics Automation, 2009, pp. 491-494, doi: 10.1109/ICMTMA.2009.558.
10. Amin Mubarak Alamin Ibrahim\* et al.,(IJITR) INTERNATIONAL JOURNAL OF INNOVATIVE TECHNOLOGY AND RESEARCH, Volume No.3, Issue No.1, December – January 2015, 1808 – 1812.
11. Pardeep Kumar, Ashish Parmar, Versatile Approaches for Medical Image Compression: A Review, Procedia Computer Science, Volume 167, 2020, Pages 1380-1389, ISSN 1877-0509, DOI: 10.1016/j.procs.2020.03.349
12. Cazañas-Gordón, Alex, Parra-Mora, Esther, "Digital Compression in Medical Images", Latin-American Journal of Computing, vol. 9, no. 1, 2022, DOI: 10.5281/zenodo.5816321
13. Sr.J.Rani, Dr.G.Glorindal, Dr.Ignatius A Herman, "Medical Image Compression using DCT with Entropy Encoding and Huffman on MRI Brain Images", Asian Journal of Applied Science and Technology (AJAST), Volume 6, Issue 2, Pages 16-25, April-June 2022 , DOI: 10.38177/ajast.2022.6203
14. Monagi H. Alkinani, E. A. Zanaty, Sherif M. Ibrahim, "Medical Image Compression Based on Wavelets with Particle Swarm Optimization", CMC, 2021, vol.67, no.2, DOI:10.32604/cmc.2021.014803
15. Saravanan.S, Sujitha Juliet.D, "CT, MRI Image Reconstruction using Deep Autoencoder with Boltzmann Machine Training Architecture", Journal of Cardiovascular Disease Research, ISSN: 0975-3583, 0976-2833, VOL12, ISSUE 02, 2021
16. Prakash Tunga P., Vipula Singh, "Compression of MRI brain images based on automatic extraction of tumor region", International Journal of Electrical and Computer Engineering (IJECE), Vol. 11, No. 5, October 2021, pp. 3964~3976, ISSN: 2088-8708, DOI: 10.11591/ijece.v11i5.pp3964-3976
17. Veronica Corona, Yehuda Dar, Guy Williams, Carola-BibianeSchonlieb, "Regularized Compression of MRI Data: Modular Optimization of Joint Reconstruction and Coding", arXiv:2010.04065v2 [eess.IV] 9 Nov 2020
18. Narayana Prakash S, A M Khan, "MRI image compression using multiple wavelets at different levels of discrete wavelets transform", Third National Conference on Computational Intelligence (NCCI 2019), IOP Publishing Journal of Physics: Conference Series, 1427 (2020) 012002, DOI:10.1088/1742-6596/1427/1/012002
19. R Pandian\* and S Lalitha Kumari, "Characterization of Alzheimer MRI Image based on Image Compression Techniques", Journal of Scientific & Industrial Research, Vol. 79, November 2020, pp. 1028-1030,
20. Nirajan Bist, Suraj Joshi, Abhishek Karki, Bharat Raj Joshi, "IMPROVED IMAGE COMPRESSION USING LOSSLESS HUFFMAN ENCODING (I2COM)", KEC Conference
21. Shuai Liu, Weiling Bai, Nianyin Zeng\* and Shuihua Wang, "A Fast Fractal based Compression for MRI Images", IEEE Access May 2019, DOI: 10.1109/ACCESS.2019.2916934
22. Nassir H. Salman, Enas Kh. Hassan, "RUN LENGTH ENCODING BASED LOSSLESS MRI IMAGE COMPRESSION USING LZW AND ADAPTIVE VARIABLE LENGTH CODING", JOURNAL OF SOUTHWEST JIAOTONG UNIVERSITY, Vol. 54 No. 4 Aug. 2019, ISSN -0258-2724, DOI：10.35741/issn.0258-2724.54.4.23
23. Ruchi Agarwal, C.S. Salimath and Khursheed Alam, "Multiple Image Compression in Medical Imaging Techniques using Wavelets for Speedy Transmission and Optimal Storage", Biomedical & Pharmacology Journal, March 2019, Vol. 12(1), p. 183-198, DOI: 10.13005/bpj/1627
24. D. J. Ashpin Pabi, P. Aruna, N.Puviarasan, "Tri-mode dual level 3-D image compression over medical MRI images", International Journal of Advanced Computer Research, Vol 7(28), ISSN (Print): 2249-7277, ISSN (Online): 2277-7970, DOI: 10.19101/IJACR.2017.728007
25. V. Anusuya, V. Srinivasa Raghavan, G. Kavitha, "Lossless Compression on MRI Images Using SWT", Society for Imaging Informatics in Medicine 2014, DOI 10.1007/s10278-014-9697-9
26. Prof. Bipin D. Mokal, Prakruti J. Joshi, Vivek P. Patkar, "Image Compression For MRI", International Journal of Scientific & Engineering Research, Volume 4, Issue 10, October-2013, ISSN 2229-5518
27. Jan-Yie Liang, Chih-Sheng Chen, Chua-Huang Huang, Li Liu, "Lossless Compression of Medical Images Using Hilbert Space-Filling Curves", DOI: 10.1.1.127.6563