

MATLAB Implementation for Relative Assay of Compression Techniques

Jatinder Kaur¹, Sharnjeet Kaur² and Dr. Sajjan Singh³

¹ Assistant Professor, Department of ECE, Chandigarh Group of Colleges Technical Campus, Jhanjeri, jatinder.cgctc@gmail.com

² Assistant Professor, Department of CSE, Chandigarh Group of Colleges Technical Campus, Jhanjeri, sharn4765@yahoo.com

³ Professor & HOD (ECE), Department of ECE, Chandigarh Group of Colleges Technical Campus, Jhanjeri, hodecejhanjeri@cgc.ac.in

*Correspondence: jatinder.cgctc@gmail.com

ABSTRACT- With the advancement of technology, the throughput during data transfer, becomes major concern for all. In most of the system, data transfer ability is fixed so for improvement in throughput, the various compression strategies came into light. These all techniques help to increase data rate by using fixed ability. At the same time compression techniques helps to accommodate purely binary data in every system.

This paper delineates the conceptual and practical approach of compression strategies. Moreover, it also scrutinizes the future prospect of research work in the field of compression.

Keywords: Lossy Compression, throughput, efficiency, code length etc.

ARTICLE INFORMATION

Author(s): Jatinder Kaur, Sharnjeet Kaur and Dr. Sajjan Singh;

Received: 05/07/2023; **Accepted:** 19/09/2023; **Published:** 30/09/2023;

e-ISSN: XXXX-XXXX;

Paper Id: IJCSR-030101;

Citation: 10.37391/IJCSR.030101



Publisher's Note: FOREX Publication stays neutral with regard to Jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

Compression is a process of taking input and transfer that into such a form that utilize less space and bandwidth during transmission. Actually, data compression is a way of bit rate reduction. Various methodologies come into light for this purpose. Inputs for these techniques may be in any form i.e., text, image, video and audio. The only thing which keeps into mind is that the method of compression is decided on the type of input and the amount of compression ratio [1,2].

According to the information transmission, compression classified into four categories as shown in *figure 1*.

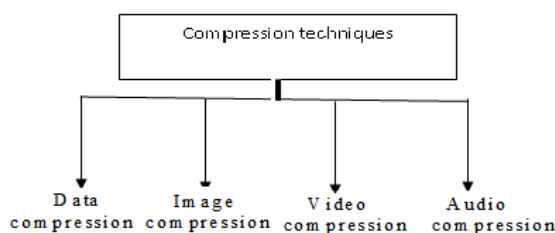


Figure 1: Classification of Compression methods on the basis of input

Figure 1 indicates that input bifurcate compression strategies as data compression, image compression, video compression and audio compression.

Again, compression techniques classification according to the compression ratio is bit preserving compression and lossy compression.

2. COMPRESSION ALGORITHM THEORY

2.1 Bit preserving compression and Lossy compression

Bit preserving compression techniques are fully consistent and are chiefly captive to data operations.

In these condensation strategies the data is compressed and can be reconstituted (uncompressed) without loss of detail or information. Decompression result is always exact reconstruction of input. Other terms that reference bit-preserving compression is lossless or reversible and non-destructive compression.

Lossy compression techniques may be bit preserving or may not be fully reproducible and are principally bounded to operations on images, video and audio. In this method, the result of condensation may not yield exact duplication of the original input, the differences between the original and reconstructed data may be so minor as to be difficult to visually perceive or hear. Since the compression ratio obtainable by the use of lossy compression can significantly exceed the compression ratio obtainable from lossless compression, the primary trade-offs concerns our need for reproducibility versus our requirements for storage and transmission. The aim of this technique is to achieve the best fidelity for a given bit rate or minimizing the

bit rate or minimizing the bit rate to achieve a given fidelity measure [3,4].

2.2 Compression methods on the basis of input

Data Compression: Data compression is the fundamental expression of information theory because it is related with redundancy. Data compression algorithms taking a stream of symbols as input and then convert that into coded form by applying information theory methods. Data compression transform data files into compact film by viewing the storage efficiency and transmission. Data compression calculation includes compression ratio, redundancy, efficiency figure of merit etc. Data compression may be lossless or lossy.

Image Compression: Image compression is a technique to reduce pixel count without affecting image quality of original image. Image compression includes image processing, image analysis and image understanding.

Video Compression: Video compression system gives very significant performance advantages for visual communications at both low and high transmission bandwidths.

Audio Compression: A speech signal is usually represented in digital format, which is a sequence of binary bits. For storage and transmission applications, it is desirable to compress a signal by representing it with as few bits as possible, while maintaining its perceptual quality.

3. PROBLEM FORMULATION

In this paper, comparative analysis of data compression techniques will be done by using MATLAB.

First, we want to transmit a message “GOOD MORNING” by using Huffman coding. Then apply Shannon code on the same message.

After the application of above defined methods, comparative analysis will be done on the basis of results obtained then apply Arithmetic coding and check the accuracy level.

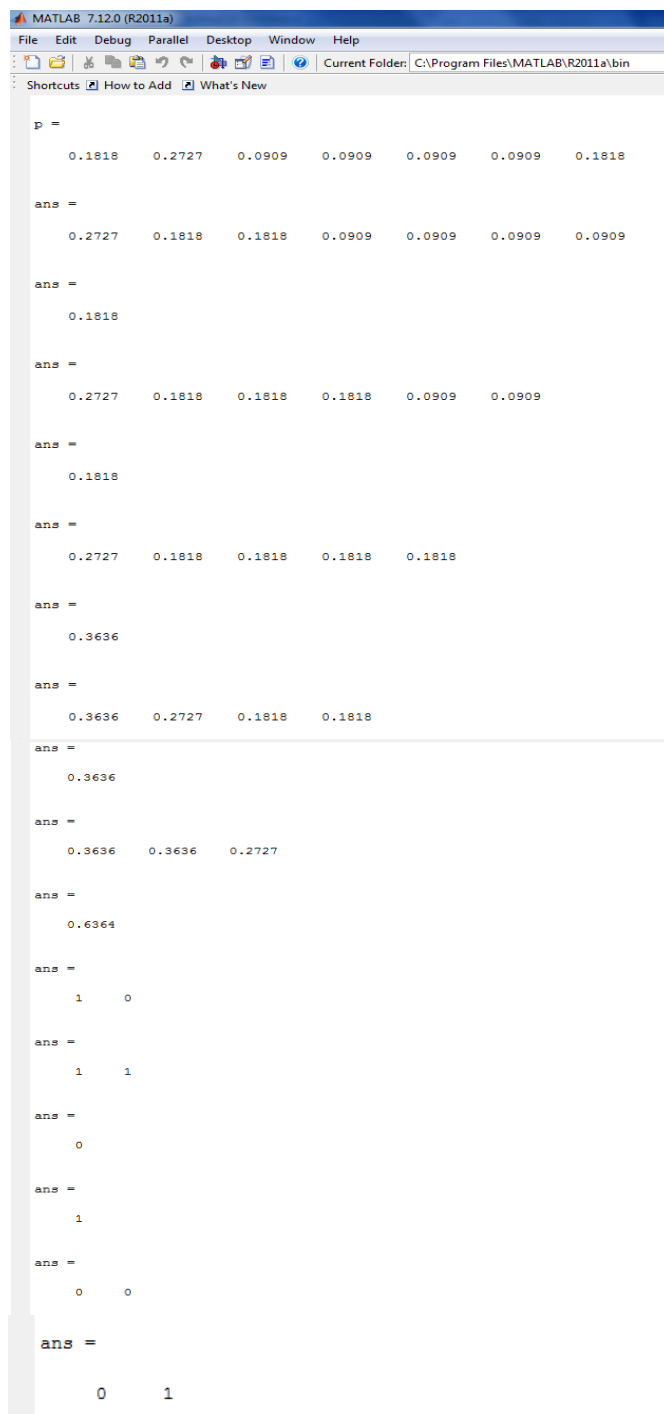
Table 1: Sample input with probabilities of occurrence

GOOD MORNING	
Symbol	Probabilities
G	0.181818182
O	0.272727273
D	0.090909091
R	0.090909091
N	0.181818182
I	0.090909091
M	0.090909091
Total Probability	1

4. RESULTS AND DISCUSSION

In this paper, For the Huffman coding implementation first step is to define the length of symbols i.e., the number of symbols. Then use ‘rand’ command which automatically select probability range and also take care to maintain maximum

probability equal to zero. After that arrange all symbols in descending order, then add two least probabilities and generate a child node. Again check that all symbols should be in descending order. This step repeats till we get only two symbols at last. For arranging all symbols in descending order after the addition of least probabilities in every step, ‘sort(descend)’ command is used.



```

MATLAB 7.12.0 (R2011a)
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Program Files\MATLAB\R2011a\bin
Shortcuts How to Add What's New

p =
    0.1818    0.2727    0.0909    0.0909    0.0909    0.0909    0.1818

ans =
    0.2727    0.1818    0.1818    0.0909    0.0909    0.0909    0.0909

ans =
    0.1818

ans =
    0.2727    0.1818    0.1818    0.1818    0.0909    0.0909

ans =
    0.1818

ans =
    0.2727    0.1818    0.1818    0.1818    0.1818

ans =
    0.3636

ans =
    0.3636    0.2727    0.1818    0.1818

ans =
    0.3636

ans =
    0.3636    0.3636    0.2727

ans =
    0.6364

ans =
    1    0

ans =
    1    1

ans =
    0

ans =
    1

ans =
    0    0

ans =
    0    1
    
```

Figure 2: Result of Huffman data compression code

As we get two symbols, the further process is to assign 0 to high probability symbol and 1 to least probability symbol.

```

MATLAB 7.12.0 (R2011a)
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Program Files\MATLAB\R2011a\bin
Shortcuts How to Add What's New

Enter the no. of message ensembles : 7
Enter the probabilities in descending order
Ensemble 1
0.181818182
Ensemble 2
0.181818182
Ensemble 3
0.272727273
Ensemble 4
0.090909091
Ensemble 5
0.090909091
Ensemble 6
0.090909091
Ensemble 7
0.090909091

Alpha Matrix
a =

    0    0.1818    0.3636    0.6364    0.7273    0.8182    0.9091

Code length matrix
n =

    3    3    2    4    4    4    4

Codeword 1
z =

    0    0    0

Codeword 2
z =

    0    0    1

Codeword 3
z =
z -
    0    1

Codeword 4
z =

    1    0    1    0

Codeword 5
z =

    1    0    1    1

Codeword 6
z =

    1    1    0    1

Codeword 7
z =

    1    1    1    0
    
```

Figure 3: Result of Shannon-Fano data compression Coding

In *Shannon-fano* algorithm, the one step of arranging all symbols in descending order is same as Huffman's. Then Recursively divide into two parts in such a way that difference between these two parts is minimum as possible. This process repeats till we get one the upper part of the list is assigned the binary digit 0, and the lower part is assigned the digit 1. This means that the codes for the symbols in the first part will all start with 0, and the codes in the second part will all start with 1.

Above two figures of data compression techniques analysed that the Huffman code provides more optimal results as compared to Shannon-fano code. As for the same probabilities we require less number of bits in Huffman as compared to Shannon.

```

MATLAB 7.12.0 (R2011a)
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Program Files\MA
Shortcuts How to Add What's New

Enter the word GOOD MORNING
Arithmetic Encoding Started
Probability for G is 0.16667
Probability for O is 0.25
Probability for D is 0.083333
Probability for I is 0.083333
Probability for M is 0.083333
Probability for R is 0.083333
Probability for N is 0.16667
Probability for J is 0.083333
The tag is 0.039541
Arithmetic Decoding Started
Received word is GOOD MORNING
Successfully Received

fx >>
    
```

Figure 4: Result of Arithmetic data compression Coding

As the above figure depicted that its procedure and calculation is much simpler as well as we can directly calculate the probabilities of occurrence. The prior knowledge of symbol probabilities is not required.

5. CONCLUSION

In the nutshell, Arithmetic coding analysis is much easier, as in this technique, prior information about the probability of occurrence is not required. Above mentioned techniques helps in minimizing bandwidth requirements during the data transmission. In data transmission, lossless compression is much prefer, since if any part of data is missing then it's tough for the receiver to interpret exact data.

By using various optimization techniques, efficiency of these all methods can be further improved.

REFERENCES

- [1] R.S. Brar and B. Singh, "A survey on different compression techniques and bit reduction Algorithm for compression of text data" International Journal of Advanced Research In Computer Science and Software Engineering (IJARCSSE) Volume 3, Issue 3, March 2013
- [2] P. Yellamma, N. Challa, "Performance Analysis of Different Data Compression Techniques on Text File" International Journal of Engineering Research and Technology (IJERT), Vol. 1 Issue 8, October – 2012, pp.1-6.
- [3] S. Porwal, Y. Chaudhary, J. Joshi and M. Jain, "Data Compression Methodologies for Lossless Data and Comparison between Algorithms" International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 2, Issue 2, March 2013
- [4] P. Yellamma, N. Challa, "Performance Analysis of Different Data Compression Techniques on Text File" International Journal of Engineering Research and Technology (IJERT), Vol. 1 Issue 8, October – 2012, pp.1-6
- [5] M. Gupta, B. Kumar, "Web Page Compression using Huffman Coding Technique" International Conference on Recent Advances and Future Trends in Information Technology (iRAFIT2012) Proceedings published in IJCA, International Journal Of Computing Applications



© 2023 by the Jatinder Kaur, Sharnjeet Kaur and Dr. Sajjan Singh. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).