SJST
sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

Data Compression and Transform Coding for Improved Performance in Optical-Fiber Communication

Abdulmanam S. Abdulwhab[1], Zeyad Moh. Elkwash[2], Jamal Abd Eltayef Esghaer[3], Ahmed Alhadi Ahmed[4]

[1]Sabratha Higher Institute of Marine Science Technologies, Sabratha, Libya
[2]Faculty of Engineering, Sabratha University, Sabratha, Libya
[3,4]Surman College Science and Technology, Surman, Libya

rdsahl305@gmail.com[1], zeyad.alkawash@sabu.edu.ly[2], Jamal-esghaer@scst.edu.ly[3], ahmedalhrare@gmail.com[4]

Abstract

The ever-growing volumes of data demand efficient storage and reliable communication solutions. Compressing information without compromising quality is a crucial challenge faced by modern systems. Meanwhile, the need for high-speed, long-distance data transfer continues to push the boundaries of optical transmission technologies. Overcoming these dual obstacles is essential for enabling innovative applications across diverse sectors. in this research, Huffman and LZW algorithms were used and achieved good results. An average compression ratio of 1:47 was obtained for text files, while an average compression ratio of 1:10 was achieved for audio files. Huffman coding, which is a lossless data compression method, provided a higher throughput rate, making it suitable for applications where processing speed is a priority. On the other hand, the LZW algorithm, which is also a lossless technique, provided a higher compression ratio without adding additional encoding to the files, although this came at the cost of higher (RAM) Random Access Memory consumption. In the field of audio compression, a down sampling algorithm was used along with the Fast Fourier Transform (FFT) to effectively reduce the sampling rate of audio files. This approach resulted in good compression ratios while preserving the sound characteristics, as evidenced by the obtained Signal-to-Noise Ratio (SNR), Mean Square Error (MSE), and Peak Signal-to-Noise Ratio (PSNR) values. By fine-tuning the down sampling factor, which was found to give good results and provide, an excellent balance between compression efficiency and sound quality when its value was between 3 and 10. The bidirectional fiber optic link also demonstrated good performance in data transmission, with low loss values, excellent Bit Error Rate (BER) value less than $1\times 10^{-15}$and Q-factor higher than 7, high Optical Signal-to-Noise Ratio (OSNR) levels above 18dB, This optical communication system can find applications in high-speed data transmission, long-distance communications, and other areas that require reliable and high-bandwidth data transfer. Overall, this research highlights the multi-faceted capabilities in both data compression and optical communications. By the seamless integration of Huffman and LZW algorithms, audio down sampling, and fiber optic technologies, a system was implemented and simulated to provide a solution that can meet a wide range of applications, from efficient data storage and transmission to high performance communication systems.

Submitted: 21/03/2025 *Accepted: 27/04/2025*

SJST
sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

الملخص

تتطلب الكميات المتزايدة من البيانات حلولاً فعالة للتخزين والاتصال الموثوق. إن ضغط المعلومات دون المساس بالجودة هو تحد حاسم تواجهه الأنظمة الحديثة. في الوقت نفسه، لا يزال هناك حاجة متزايدة لنقل البيانات بسرعة عالية وعلى مسافات طويلة عبر تقنيات الإرسال البصري. التغلب على هذين العائقين أمر بالغ الأهمية لتمكين التطبيقات المبتكرة في مختلف القطاعات. في هذا البحث، تم استخدام خوارزميات هافمان و(LZW) التي حققت نتائج جيدة. حيث تم الحصول على متوسط ضغط بمعدل (1:47) للملفات النصية بينما تم الحصول على متوسط ضغط (1:10) للملفات الصوتية، ترميز هافمان، وهو طريقة ضغط بيانات بدون فقد، قد وفر معدل معالجة أعلى، مما جعله مناسبًا للتطبيقات التي يكون فيها سرعة المعالجة ذات أولوية. من ناحية أخرى، وفرت خوارزمية LZW، وهي أيضًا تقنية بدون فقد، نسبة ضغط أعلى بدون اضافة ترميزات اضافية للملفات، على الرغم من أنه جاء على حساب استهلاك ذاكرة عشوائية أعلى. في مجال ضغط الصوت، تم استخدام خوارزمية التخفيض مع تحويل فوريير السريع (FFT) لتقليل معدل أخذ العينات للملفات الصوتية بفعالية. أسفر هذا النهج عن نسب ضغط جيدة مع الحفاظ على خصائص الصوت بجودة جيدة، كما يتضح من قيم نسبة الإشارة إلى الضوضاء (SNR) والخطأ المربع المتوسط (MSE) للإشارة والنسبة الإشارة إلى الضوضاء المكتسبة (PSNR) التي تم الحصول عليها. من خلال التعديل الدقيق على معامل التخفيض، الذي وجد أنه يعطي نتائج جيدة عندما تكون قيمته بين 3 و10، هذه القيم اعطت توازن ممتاز بين كفاءة الضغط وجودة الصوت. أيضا أظهر رابط ألياف البصرية ثنائي الاتجاه. أداءً جيدا في نقل البيانات، مع قيم فقد منخفضة، وقيم BER اقل من (1×10^{-15}) ومعامل جودة Q ممتاز اعلى من 7، وقيم OSNR اعلى من 18dB، يمكن أن يجد هذا النظام الاتصالي البصري تطبيقات في نقل البيانات عالية السرعة والاتصالات طويلة المدى والمجالات الأخرى التي تتطلب نقل بيانات موثوق وعالي النطاق الترددي. على الصعيد الإجمالي، يسلط هذا البحث الضوء على قدرات متعددة الجوانب في كل من ضغط البيانات والاتصالات البصرية. من خلال الدمج السلس لخوارزميات هافمان وLZW وتخفيض معدل اخذ العينات للملفات الصوتية وتكنولوجيات الألياف البصرية، حيث تم تنفيذ ومحاكاة النظام بحيث يقدم حلاً يمكن أن يلبي مجموعة واسعة من التطبيقات، بدءًا من التخزين والإرسال الفعال للبيانات وصولاً إلى أنظمة الاتصالات عالية الأداء.

Introduction

The last decade has been witnessing a transformation some call it a revolution in the way we communicate, and the process is still under way. This transformation includes the ever-present, ever-growing Internet; the explosive development of mobile communications; and the ever-increasing importance of video communication. Data compression is one of the enabling technologies for each of these aspects of the multimedia revolution. It would not be practical to put text, let alone audio and video, on websites if it were not for data compression algorithms. Cellular phones would not be able to provide communication with increasing clarity was it not for compression. The advent of digital TV would not be possible without compression. Data compression, which for a long time was the domain of a relatively small group of engineers and scientists, is now ubiquitous, Long-distance calls make use of compression. Modems and fax machines benefit from compression. Listening to music on an MP3 player or watching a DVD results in entertainment courtesy of compression. [1]

In brief, data compression is the art or science of representing information in a compact form. These compact representations are created by identifying and using structures that exist in the data. Data can be characters in a text file, numbers that are samples of speech or image waveforms, or sequences of numbers that are generated by other processes. The need for data compression arises from the fact that an increasing amount of information that is generated and used in digital form, represented by bytes of data, is increasing, the number of bytes required to represent multimedia data can be enormous. For example, in order to digitally represent 1 second of video without compression (using the CCIR 601 format), More than 20 megabytes, or 160 megabits, are needed. The number of seconds in a movie makes it easy to see why compression is required. To represent 2 minutes of uncompressed CD-quality music (44,100 samples per second, 16 bits per sample) requires more than 84 million bits. Downloading music from a website at

SJST
sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

these rates would take a long time. An early example of data compression is Morse code1, developed by Samuel Morse in the mid-19th century. Letters sent by telegraph are encoded with dots and dashes. Morse noticed that certain letters occurred more often than others. In order to reduce the average time required to send a message; he assigned shorter sequences to letters that occur more frequently, such as e (·) and a (· −), and longer sequences to letters that occur less frequently, such as q (−−·−) and j (·−−−). This idea of using shorter codes for more frequently occurring characters is used in Huffman coding,

System Design & Implementation

This system is a comprehensive data compression and transmission system that leverages a combination of proven compression algorithms and optimized transport mechanisms. At the core of the system are two powerful compression techniques - Huffman coding and LZW for text compression. These algorithms work to significantly reduce the file sizes of the source materials, enabling faster and more reliable transmission over the network.

Furthermore, a downs sampling algorithm tailored for audio data is incorporated into the system. This technique intelligently reduces the sampling rate and bit depth of the audio signals, preserving the essential perceptual characteristics while dramatically decreasing the overall data footprint. This is particularly beneficial for low-bandwidth scenarios where full-fidelity audio may not be feasible. The compressed data is then transmitted over an optical fiber link, taking advantage of the inherent high-speed and low-latency properties of fiber optic technology. This ensures that the compressed data reaches the destination quickly and with minimal degradation, providing a seamless user experience.

At the receiving end, the system employs complementary decompression algorithms to restore the original text and audio data, allowing for faithful reconstruction of the transmitted content. This end-to-end solution combines the power of advanced compression techniques with the reliability of fiber optic connectivity, delivering a robust and efficient platform for the transmission of digital information. The Figure (3-1) shows the general block diagram for the proposed system.

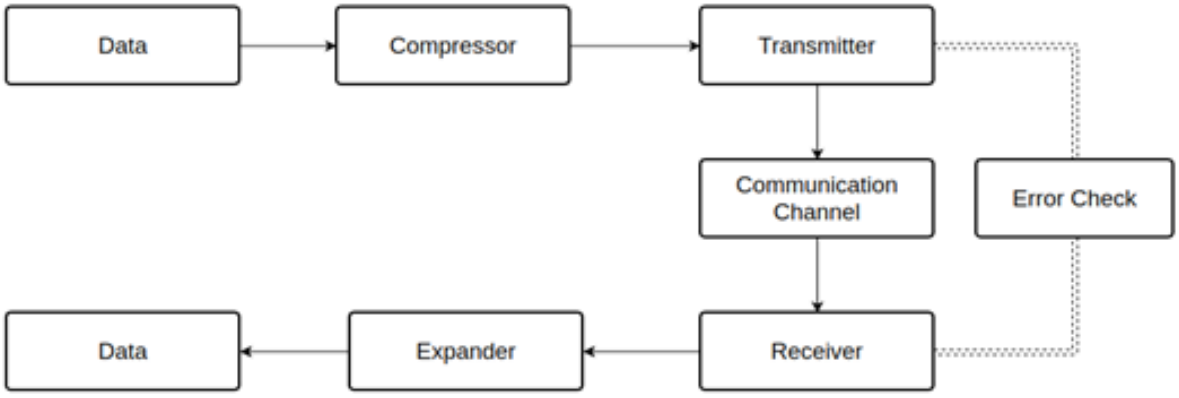


Figure 1-1: simple system general block diagram

sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

The Figure (1-2) Shows more details block diagram for the proposed System.

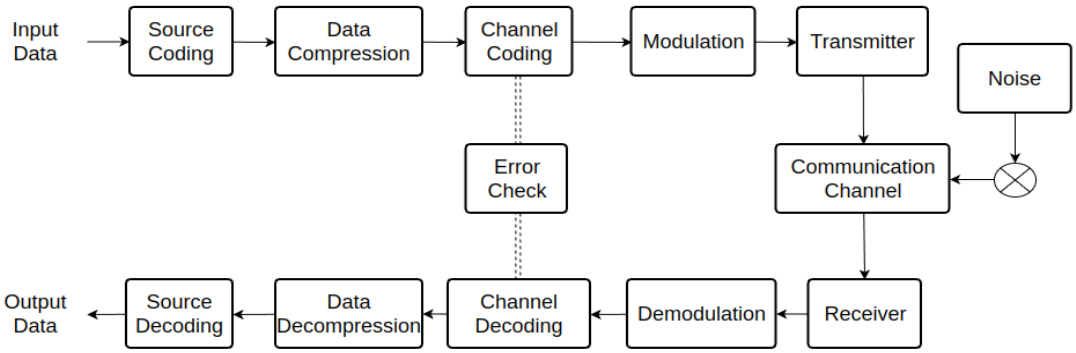


Figure 1-2: Full system Block diagram

Proposed Compression / Decompression methods

Highly efficient, lossless compression of text data is enabled by the Huffman coding algorithm and LZW algorithm at the heart of the system, substantially reducing file sizes without sacrificing any of the original content or quality. The full integrity and fidelity of the transmitted information is ensured by this lossless text compression, providing a robust and reliable solution for the exchange of digital documents and communications. Powerful lossless coding is combined. The General Block Diagram for The Lossless Compression is shown in Figure (1-3).

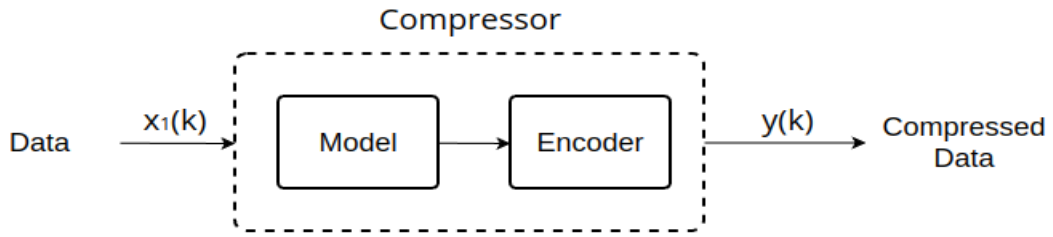


Figure 1-3: Lossless Compression Block Diagram

The Figure (1-4) shows the General Block Diagram for Lossless Text Compression

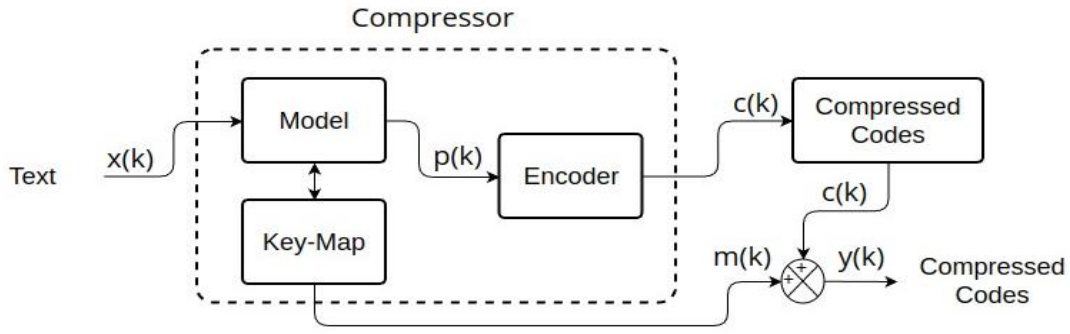


Figure 1-4: General Block Diagram for lossless Text Compression

Were

$$y(k) = m(k) + c(k)$$

sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

The Figure (1-5) Shows the General Block Diagram for The Lossless Decompression

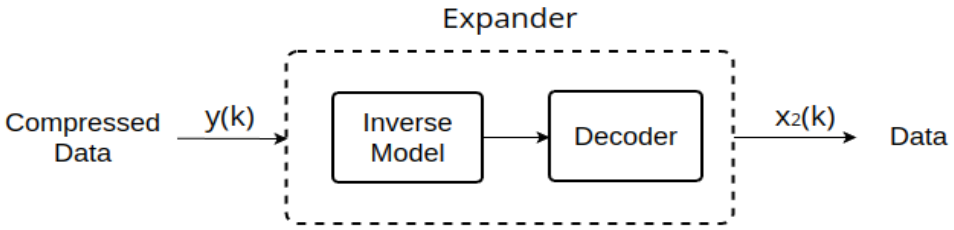


Figure 1-5 Lossless Decompression Block Diagram

And for the Lossless Compression / Decompression notice that:

$$X1(k) = X2(k)$$

The Figure (1-6) shows the General Block Diagram for Lossless Text Decompression

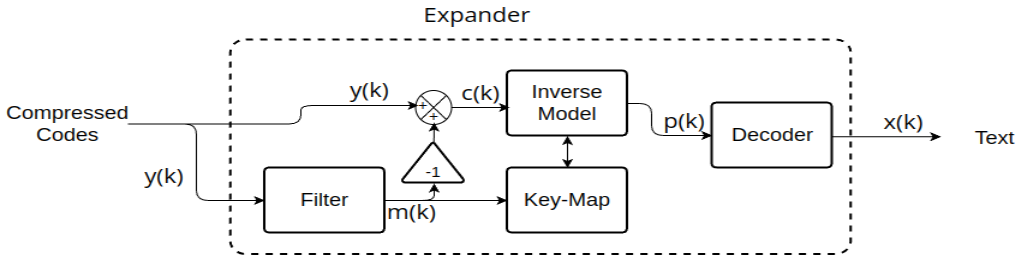


Figure 1-6: General Block Diagram for lossless Text Decompression

Were

$$c(k) = y(k) - m(k)$$

The down sampling algorithm used in the proposed system intelligently reduces the sampling rate and bit depth of audio signals, dramatically decreasing the overall data footprint. While this process involves a measure of loss compression, it preserves the essential perceptual characteristics of the original audio. The Figure (1-7) Shows the General Block Diagram for The Loss Compression

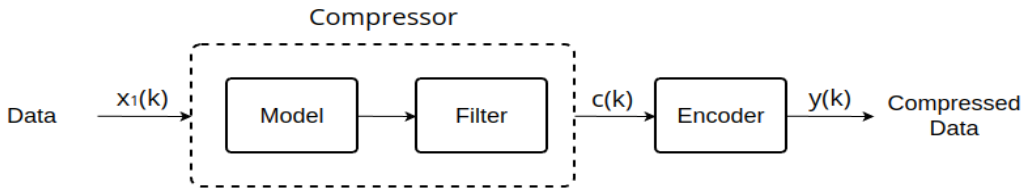


Figure 1-7: Loss Compression Block Diagram

The Figure (1-8) Shows the General Block Diagram for The Loss Decompression

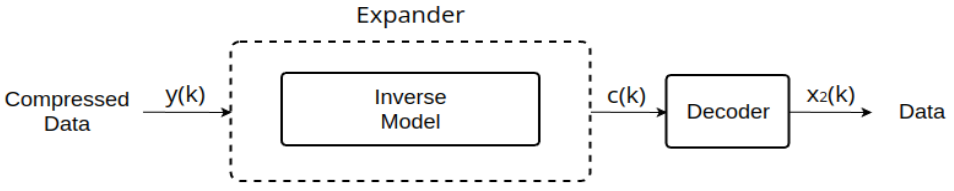


Figure 1-8: Loss Decompression Block Diagram

sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

And for the Loss Compression / Decompression notice that:

$$X1(k) \neq X2(k)$$

The Figure (1-9) shows the General Block Diagram for Loss Audio Compression.

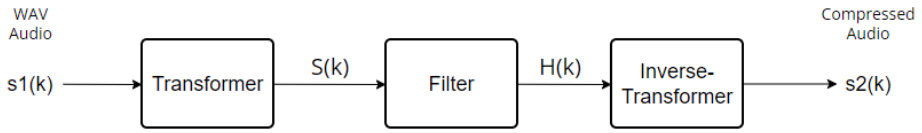


Figure 1-9: loss Audio Compression Block Diagram

Were

$$S(k) = T[s1]$$
$$H(k) = S(k) * A(k)$$
$$s2(k) = T^{-1}[H(k)]$$

Were T being the Transfer-Function.

The Figure (1-10) shows the General Block Diagram for Proposed Loss Audio Compression.

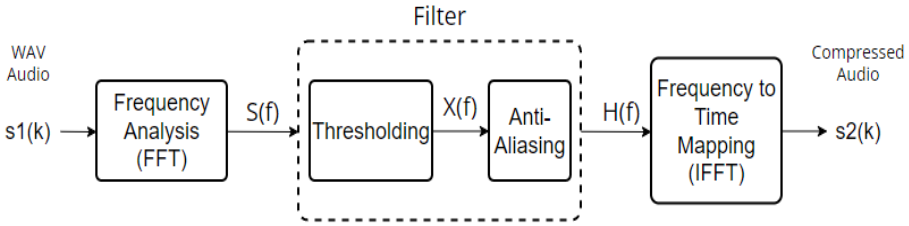


Figure 1-10: Proposed loss Audio Compression Block Diagram

Were

$$S(f) = \sum_{k=0}^{N-1} s1(k) \times e^{\frac{-2j\pi kf}{N}}$$

Were
N is the number of samples in the signal s1(k).
k is the sample index, ranging from 0 to N-1.
f is the frequency variable, which can take values from 0 to (N-1)/N.
j represents the imaginary unit √ (-1).
The threes holding operation can be represented as a multiplication of the Fourier transform S (f) with the threes holding transfer function T(f):

$$X(f) = S(f) \times T(f)$$

The anti-aliasing filtering operation can be represented as a multiplication of the thresholder signal X(f) with the anti-aliasing transfer function A(f):

$$H(f) = X(f) \times A(f)$$

So, the output s2(k) can be represented as follow:

$$s2(k) = T^{-1}[H(f)]$$

SJST
sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

$$s2(k) = \sum_{f=0}^{N-1} H(f) \times e^{\frac{2j\pi kf}{N}}$$

The complete mathematical representation of the process can be written as:

$$s2(k) = \sum_{f=0}^{N-1} [S(f) \times T(f) \times A(f)] \times e^{\frac{2j\pi kf}{N}}$$

The Figure (1-11) shows the General Block Diagram for Proposed Optical Fiber Transmission Model.

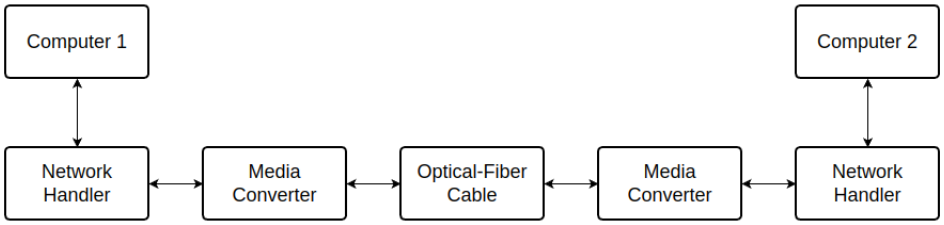


Figure 1-11: Optical Fiber Transmission Block Diagram

Huffman-Encoding Module

The Figure (1-12) shows the Huffman Compression algorithm general block diagram.

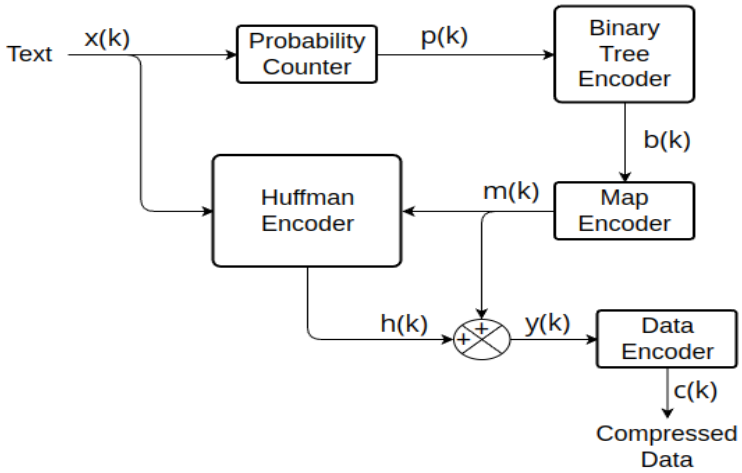


Figure 1-12: Huffman compression block diagram

SJST
sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

The Figure (1-13) shows the Huffman decompression algorithm general block diagram.

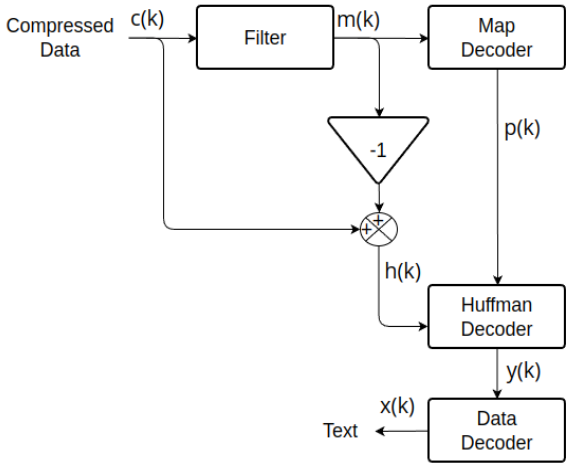


Figure 1-13: Huffman decompression block diagram

Huffman coding Compression algorithm can be listed in the next steps:
Start: This is the beginning of the process.
Calculate Characters Probabilities: The first step is to calculate the probabilities of the characters appearing in the data.
Generate Huffman Binary Tree: Based on the character probabilities, a Huffman binary tree is generated. This tree will be used to encode the characters.
Encode the Characters using the tree: The characters are encoded using the Huffman binary tree, which assigns shorter codes to more frequent characters and longer codes to less frequent characters.
Padding Encoded Characters: The encoded characters may need to be padded to ensure they are a multiple of 8 bits (1 byte) in size.
Convert Encoded Data to Bytes: The padded, encoded characters are then converted into bytes.
Merge Encoded Data with Encoded Tree: The encoded data and the Huffman binary tree are merged together.
Save Merged Data to One File: The final step is to save the merged data (encoded characters and Huffman tree) to a single output file.
End: This is the completion of the process.

sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

The Figure (1-14) shows the Huffman Coding compression algorithm flowchart.

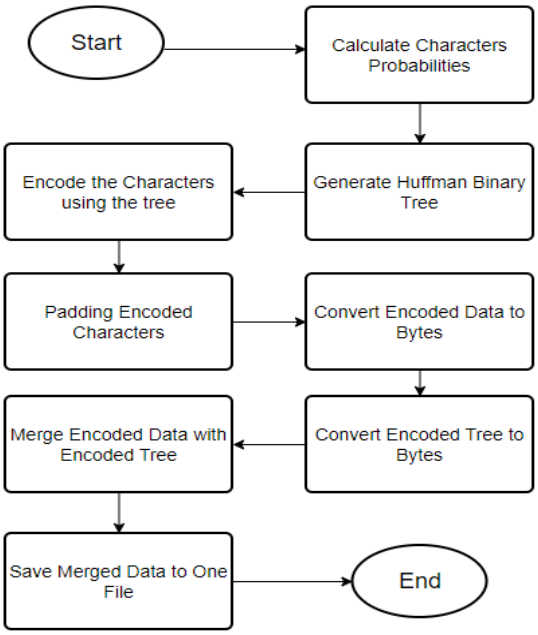


Figure 1-14: Huffman compression flowchart

Huffman coding Decompression algorithm can be listed in the next steps:
Start: This is the beginning of the decoding process.
Split Encoded Data from Encoded Tree in the File: The first step is to split the encoded data and the Huffman binary tree that were previously merged and stored in a file.
Reverse the Huffman Binary Tree: The Huffman binary tree is reversed, which is necessary for the decoding process.
Decode the Data and Binary Tree: The encoded data and the reversed Huffman binary tree are used to decode the original data.
Convert Data from Bytes to Bits: The decoded data, which was stored in bytes, is converted back to bits.
Remove the Padding: Any padding that was added during the encoding process is removed.
Decode the data using the reverse Huffman Binary Tree: The final step is to use the reversed Huffman binary tree to decode the data, recovering the original uncompressed data.
Merge the decoded data: The decoded data is merged back into a single output.
Save Merged Decode Data to One File: The final, decompressed data is saved to a single output file.
End: This is the completion of the decoding process.

sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

The Figure (1-15) shows the Huffman Coding Decompression algorithm

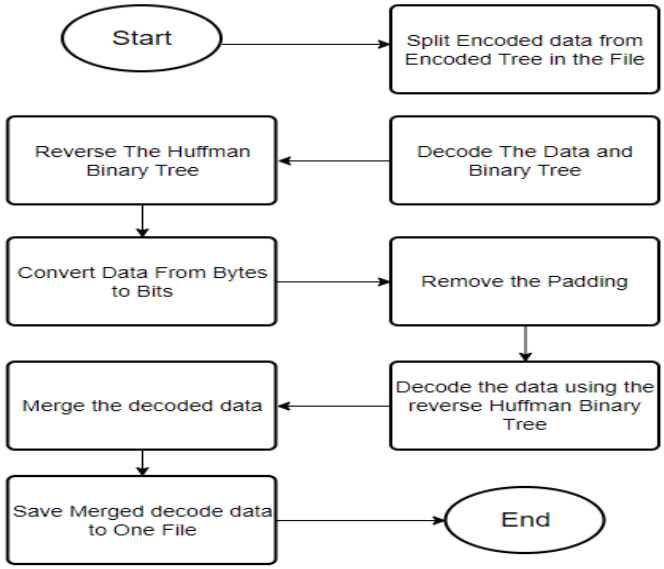


Figure 1-15: Huffman Decompression flowchart

LZW Module

LZW Compression algorithm process can be listed in the next steps:

Start: This is the beginning of the compression process.

Input First Byte: The first byte of the input data is stored in a string.

Input Next Byte: The next byte of the input data is stored in a character variable.

Is the Char in Dictionary? The algorithm checks if the combination of the string and the current character is present in the dictionary.

False: If the combination is not in the dictionary, a new key-value entry is added to the dictionary, and the code for the string is generated.

True: If the combination is in the dictionary, the string is updated by appending the current character.

String = String + Char: The string is updated by concatenating the current character.

File Not End: The algorithm checks if there are more bytes to be processed in the input file.

True: If there are more bytes, the process continues from step 3.

False: If the end of the file is reached, the process ends.

End: The compression process is completed.

sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

The Figure (1-16) shows the LZW Coding Compression algorithm flowchart.

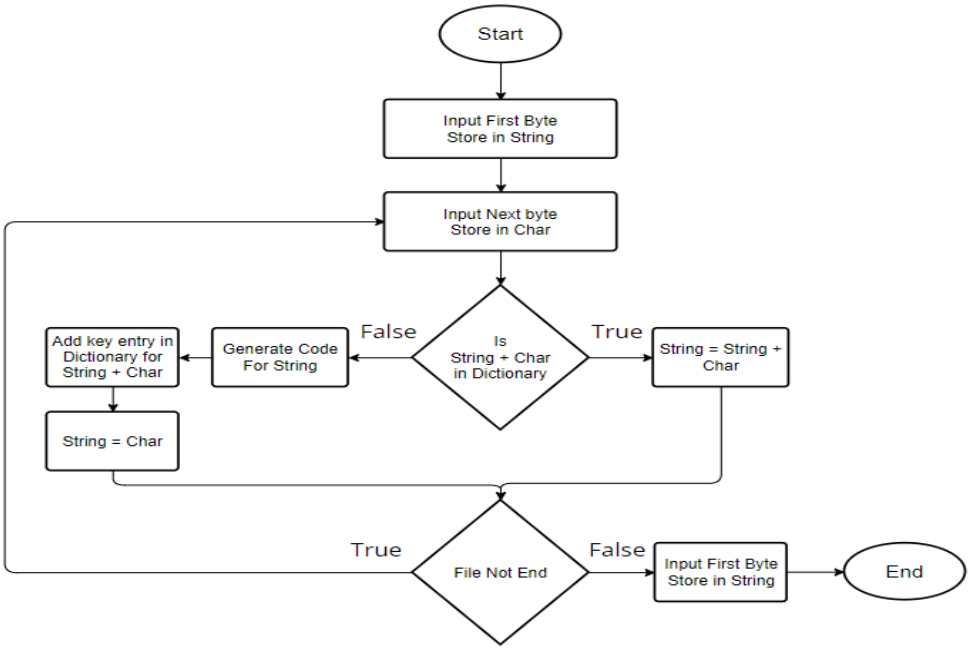


Figure 1-16: LZW Compression Algorithm Flowchart

LZW Decompression algorithm process can be listed in the next steps:
Start: This is the beginning of the decompression process.
Input First Code and Store in O Code: The first code from the compressed data is read and stored in the O Code variable.
Output the O Code Translation: The translation of the O Code is output as the decompressed data.
Input the Next Code, store it in N Code: The next code from the compressed data is read and stored in the N Code variable.
Is N Code in Dictionary: The algorithm checks if the N Code is present in the dictionary.
False: If the N Code is not in the dictionary, the string is set to the translation of the O Code, and a new dictionary entry is added for O Code + the first character of the string.
True: If the N Code is in the dictionary, the string is set to the translation of the N Code.
Output String: The decompressed string is output.
Char = First Character in String: The first character of the output string is extracted.
Add New Key Entry for O Code + Char in Dictionary: A new dictionary entry is added for the combination of the O Code and the extracted character.
O Code = N Code: The O Code is updated to the current N Code.
Is There Codes to Input: The algorithm checks if there are more codes to be processed from the compressed data.
True: If there are more codes, the process continues from step 4.
False: If there are no more codes, the decompression process ends.
End: The decompression process is completed.

sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

The Figure (1-16) shows the LZW Coding Decompression algorithm flowchart.

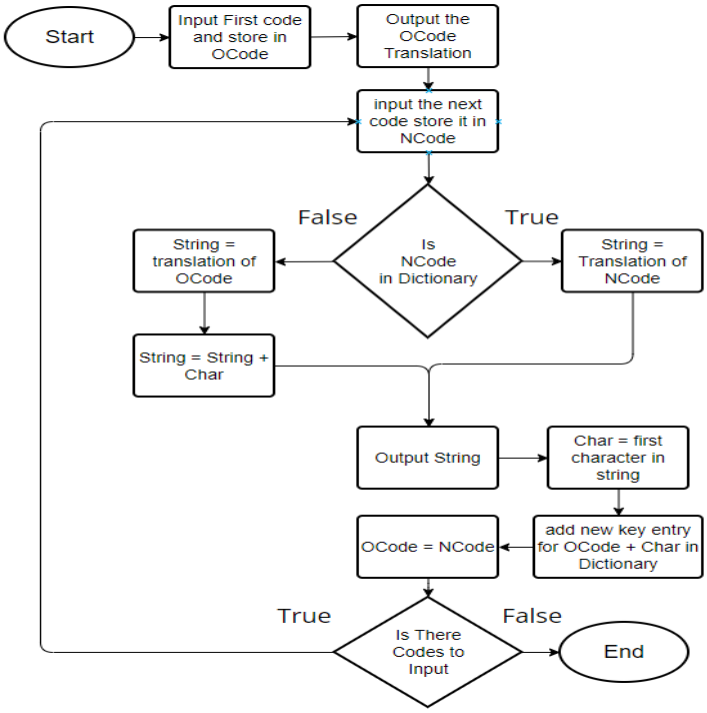
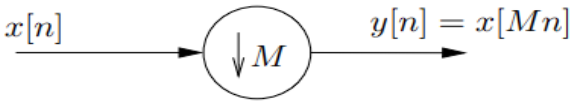


Figure 1-17: LZW Decompression Algorithm Flowchart

From the Flowcharts the Variables “O Code” and “N Code” (old-code and new-code) hold the 8-bit codes from the compressed file and “Char” variable holds a single byte, “String” variable holds a sequence of bytes.

Audio Module

The need for change in sampling rate was seen in the motivating digital audio to reduce the size of the audio files, after discrete-time processing reduce the sampling rate by sub-sampling or decimation or down-sampling. This means keeping only every M-th sample of the discrete-time process. This is usually represented as shown in next Figure.



Were Down-sampled signal:

$$y[n] = x[Mn]$$

And Were:

M is Sampling Factor.

The downs sampling algorithm process can be listed in the next steps:

Start: This is the beginning of the process.

Read WAV File: The first step is to read in a WAV audio file.

Store Audio Data: The audio data from the WAV file is stored for further processing.

Store Sampling Rate: The sampling rate of the audio file is also stored.

SJST
sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

Set the Down Sampling Factor: This step sets the down sampling factor, which is used to reduce the sampling rate of the audio data.
Setting The Threes holding Value: A threes holding value is set, which is used in a later step.
Calculate ABS Fourier Transform: The absolute value of the Fourier transform of the audio data is calculated.
Downs ample Audio Data: The audio data downs ample by the factor set earlier, reducing the sampling rate.
Apply Anti-Aliasing Filter: An anti-aliasing filter is applied to the audio data to prevent aliasing artifacts.
Apply IFFT: An inverse Fast Fourier Transform (IFFT) is applied to the downs ample audio data.
Save Audio File as WAV: Finally, the processed and compressed audio data is saved as a new WAV file.
The Figure (1-18) shows the Flowchart for the Downs sampling Algorithm.

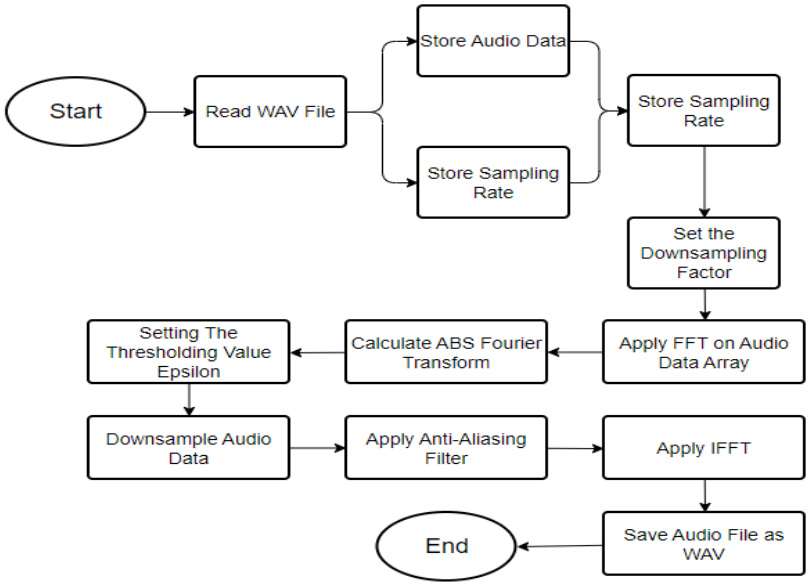


Figure 1-18: Audio Downsampling Algorithm Flowchart

Property: Down-sampling by M or DM (·) is a linear, periodically time-varying operator with period M.
Since:

$$D_M\left(\alpha x_1[n] = \beta x_2[n]\right) = \alpha x_1[Mn] + \beta x_2[Mn] = \alpha D_M\left(x_1[n]\right) + \beta D_M\left(x_2[n]\right)$$

clearly DM (·) is a linear operator. It has a periodically time-varying property because if a sequence is shifted by M, its down-sampled version is shifted by 1. More precisely

$$\begin{aligned} D_M\left(\delta[n-kM]\right) &= \delta[n-k], \qquad k \in \mathbb{Z} \\ D_M\left(\delta[n-kM-\ell]\right) &= 0, \qquad \ell = 1, 2, \ldots, M-1. \end{aligned}$$

Hence, if $y[n] = D_M\left(x[n]\right)$, then $D_M\left(x[n-kM]\right) = y[n-k]$.

SJST
sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

Therefore, for the down-sampling system the time-varying property implies that complex sinusoids are no longer Eigen-functions. This can be seen from the following argument. Let

$$x[n] = e^{j\pi n} = (-1)^n.$$

If we down-sample by a factor 2,

$$y[n] = x[2n] = e^{j\pi 2n} = 1 \neq c \cdot x[n]$$

for any constant c. Hence the complex sinusoid is not an Eigen-function for the downs sampling operator. Recall that for a linear time-invariant system, the complex exponential was an Eigen function, i.e., if

$$\mathcal{L}\{x[n]\} = \sum_m h[m]x[n-m],$$

Then For

$$x[n] = e^{j\omega_0 n},$$

$$y[n] = \mathcal{L}\left\{e^{j\omega_0 n}\right\} = H\left(e^{j\omega_0}\right) x[n]$$

y[n] was a scaled version of x[n] with scaling factor (H ejω0) which is a constant independent of n. the time-invariant property is crucial for this to be true. This is because DM (·) is linear but not time-invariant and does not have the complex exponential as an Eigen-sequence or Eigen-function.

An important consideration is to understand what happens to the z-transform (or discrete-time Fourier transform) when one down-samples a signal. To be specific, consider down-sampling by a factor of 2 of a discrete-time signal with z-transform X(z). now define a new discrete-time sequence x ′ [n] with z-transform X′ (z) as

$$\begin{aligned} X'(z) &= \frac{1}{2}\left[X(z) + X(-z)\right] = \frac{1}{2}\sum_k x[k]z^{-k} + \frac{1}{2}\sum_k x[k](-1)^{-k}z^{-k} \\ &= \frac{1}{2}\sum_k \left\{x[k] + (-1)^{-k}x[k]\right\} z^{-k} \\ &= \sum_{k:\text{even}} x[k]z^{-k} + \underbrace{\sum_{k:\text{odd}} x[k](1-1)z^{-k}}_{0} \\ &= \sum_n x[2n]z^{-2n}. \end{aligned}$$

Now, if

$$y[n] = x[2n],$$

Then

$$Y(z) = \sum_n y[n]z^{-n} = \sum_n x[2n]z^{-n} = X'\left(z^{\frac{1}{2}}\right) = \frac{1}{2}\left[X\left(z^{\frac{1}{2}}\right) + X\left(-z^{\frac{1}{2}}\right)\right].$$

Evaluating it on the unit circle, if the ROC includes it, gives

SJST
sjst.scst.edu.ly
Surman Journal for Science and Technology
ISSN: Online (2790-5721) - Print (2790-5713)
مجلة صرمان للعلوم والتقنية
Vol 7, No.1.Jan – Jun. 2025
Pages: 278 ~ 294

$$Y\left(e^{j\omega}\right)=\frac{1}{2}\left[X\left(e^{j\frac{\omega}{2}}\right)+X\left(e^{-j\frac{\omega}{2}}\right)\right]=\frac{1}{2}\left[X\left(e^{j\frac{\omega}{2}}\right)+X\left(e^{j(\frac{\omega}{2}-\pi)}\right)\right].$$

Optical Fiber Module

The Proposed Optical Fiber system was design to be point to point network system as shown in Figure (3-19)

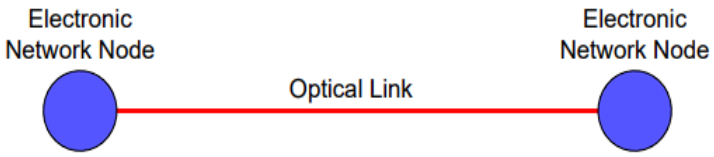


Figure 1-19: optical fiber point to point system

Each electronic node contains at least one transceivers. depicts a transceiver with a directly modulated light emitting diode or laser. For higher data rates it is common to use a laser with a continuous output and an external optical modulator.
The Figure (1-20) Shows the inner components of the Electronic Node.

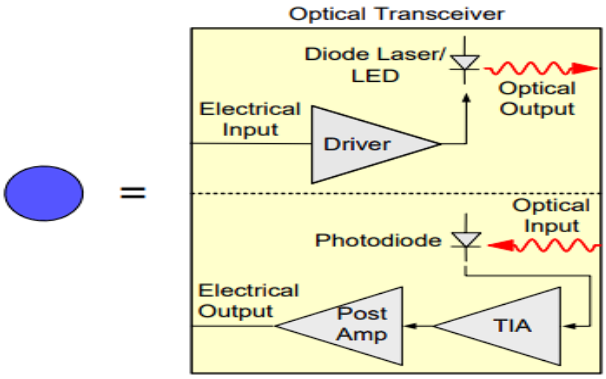


Figure 1-20: inner components of Electronic Node

Conclusion

The results obtained from the implementation and simulation of this prototype system demonstrate that a high-performance system capable of compressing text files with an average compression ratio of (1:47) has been successfully developed. This was achieved by utilizing two different compression algorithms, which are suitable for different situations.

The average performance parameters for both Huffman Coding and LZW algorithms can be calculated for comparison. The average parameter values are listed in -From this analysis, it is noticed that the LZW algorithm achieved a higher compression ratio compared to Huffman Coding, while the Huffman Coding algorithm used less memory space for the text compression. Additionally, the LZW algorithm achieved a lower compression and decompression time, whereas the Huffman Coding algorithm achieved a higher compression throughput.

Furthermore, the system was able to achieve a compression ratio of (1:10) for audio files, by the down sampling algorithm using FFT. and a linear relationship between the down sampling factor and the compression ratio was revealed, and it was observed that when the down sampling factor exceeded 11, poor SNR, MSE, and PSNR values were experienced by the system, rendering

unusable audio. However, good SNR, MSE, and PSNR values were obtained when the down sampling factor was between 3 and 10.

REFERENCES

[1] A. Spanias, T. Painter and V. Atti, Audio Signal Processing And Coding, Wiley, 2007.
[2] "Audio Equipmnt," [Online]. Available: https://en.wikipedia.org/wiki/Audio_equipment#:~:text=Audio%20equipment%20refers%20to%20devices,units%2C%20headphones%2C%20and%20speakers.. [Accessed 22 6 2024].
[3] "Digital Audio," [Online]. Available: https://en.wikipedia.org/wiki/Digital_audio. [Accessed 22 6 2024].
[4] J. O. Smith, Mathematics of The Discrete Fourier Transform with Audio Applications (Second Edition), Stanford University.
[5] N. Siddhardha, "Medium," [Online]. Available: https://nandasiddhardha.medium.com/harmonizing-bits-and-beats-a-students-tale-of-pcm-e77458e69c60. [Accessed 22 6 2024].
[6] "Nyquist Shannon Sampling Theorem," [Online]. Available: https://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem. [Accessed 22 6 2024].
[6] "TechTarget," [Online]. Available: https://www.techtarget.com/whatis/definition/Nyquist-Theorem. [Accessed 22 6 2024].
[7] "geeksforgeeks," [Online]. Available: https://www.geeksforgeeks.org/nyquist-sampling-theorem/. [Accessed 22 6 2024].
[8] "Ircam," [Online]. [Accessed 22 6 2024].
[9] D. F. Elliott, "Chapter 2 "Analog-to-Digital and Digital-to-Analog Conversion"," in Handbook of Digital Signal Processing: Engineering Applications, pp. 39-42.
[10] "Wikipedia anti-aliasing filter," [Online]. Available: https://en.wikipedia.org/wiki/Anti-aliasing_filter. [Accessed 24 6 2024].
[12] "NTI Audio," [Online]. Available: https://www.nti-audio.com/en/support/know-how/fast-fourier-transform-fft. [Accessed 22 6 2024].
[13] D. Y. Pan, "Digital Audio Compression," p. 14.
[14] "byjus communication-systems," [Online]. Available: https://byjus.com/jee/communication-systems/. [Accessed 22 6 2024].
[15] "wikipedia communication channel," [Online]. Available: https://en.wikipedia.org/wiki/Communications_system. [Accessed 22 6 2024].
[16] "wikipedia communication systems," [Online]. Available: https://en.wikipedia.org/wiki/Communications_system. [Accessed 22 6 2024].
[17] B. .. chawda, "Historical Development of Optical Communication Systems," p. 2, 2015.
[18] A. Patibandla, Fiber Optical Communication Lectures, 2019.

[19] J. M. Senior, Optical Fiber Communication Principles and Practice Third Edition, 2009.
[20] A. Kaushik, "OPTICAL FIBER: THE FUTURE OF HIGH SPEED COMMUNICATION (ADVANTAGES, TECHNOLOGY AND FUTURE ASPECTS)," IJSDR, p. 5, 2019.
[21] "wikipedia Fiber Media Converter," [Online]. Available: https://en.wikipedia.org/wiki/Fiber_media_converter. [Accessed 22 6 2024].
[22] "qsfptek Media Converter Wiki, Everything You Should Know About Media Converter," [Online]. Available: https://www.qsfptek.com/qt-news/media-converter-wiki-everything-you-should-know-about-media-converter.html. [Accessed 23 6 2024].