



Sonified Cryptography: Secure Text Encoding with DNA and Non-Speech Audio

Chandrasekaran Saravanakumar^{1*}, Neelamegam Subhashini²

^{1,2}Department of Electronics and Communication Engineering, SRM Valliammai Engineering College, Potheri, India

¹dracsaravanakumar@ieee.org, ²subhashinin.ece@srmvalliammai.ac.in

Abstract

The increasing demand for data security in digital communication, particularly in high-risk sectors like defense, has led to the exploration of innovative encryption approaches. This study presents a dual-layer encryption model that enhances information concealment by integrating DNA-based cryptography with audio signal encoding. Initially, plaintext is converted into binary and obfuscated using XOR operations with randomly generated DNA sequences. The resulting DNA nucleotide sequences (A, G, C, T) form the first layer of encryption. In the second layer, these sequences are audified by mapping each nucleotide to a specific frequency, thereby transforming the encrypted data into non-speech audio signals. To evaluate the integrity and uniqueness of the encryption-decryption process, Fast Fourier Transform (FFT)-based cross-correlation is applied, comparing the original and recovered audio signals. The proposed method is implemented in MATLAB and tested on various input strings. Results demonstrate significant improvements in encryption speed and security, with the added benefit of imperceptibility in audio form. The method outperforms existing DNA-based techniques in terms of computational efficiency and resistance to brute-force attacks. This hybrid cryptographic technique offers a promising solution for secure, covert data transmission in sensitive applications.

Keywords: cryptography; dual layer encryption; DNA encryption; information security; fast fourier transforms; non speech audification

How to Cite: Chandrasekaran Saravanakumar and Neelamegam Subhashini, "Sonified Cryptography: Secure Text Encoding with DNA and Non-Speech Audio", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 9, no. 4, pp. 832 - 839, Aug. 2025.

Permalink/DOI: <https://doi.org/10.29207/resti.v9i4.6591>

Received: April 27, 2025

Accepted: August 8, 2025

Available Online: August 18, 2025

This is an open-access article under the CC BY 4.0 License

Published by Ikatan Ahli Informatika Indonesia

1. Introduction

The encryption of data in the field of communication is rapidly increasing in particular the data transfer using internet. But for military applications, the level of maintaining the secrecy of the data is still yet to be improved. Sonification is one of the technique used to communicate the information in the form of music which could be a possible solution for military applications. The wide applications of sonification includes IoT, Bioinformatics, Visual aid [1] – [4]. The significant benefit of this method is the visualization of music doesn't show any text which is encrypted in it[5] - [8]. A wide range of elements can be adjusted to change the receiver's impression of the sound, and thus, their view of the fundamental data being depicted [9] – [12]. Regularly, an expansion or reduction in some level in this data is demonstrated by an increment or abatement in frequency or beat, yet could likewise be shown by fluctuating other less ordinarily utilized

elements. In contrast, this technique faces a countless issue to prepare and to attain the data as non – speech audio [13]. Numerous investigations have been initiated to track down the best procedures for different kinds of data, and at this point, no indisputable arrangement of methods to be utilized has been formed. As the space of sonification is as yet viewed as in its outset, current investigations are running after deciding the best arrangement of sound parts to shift in various circumstances [14]. In the work of [15], [16], suggested the techniques like audification, acoustic sonification and stream based sonification to translate the data.

Audification is a hear-able showcase strategy for addressing a succession of information as audio. By definition, it is portrayed as an "immediate interpretation of an information in the form of audio" [17]. Audification encodes an information where input data are distinctly coded to sound levels.

The field of cryptography is introduced with DNA sequence [18]. Some of the recent DNA based cryptography techniques is studied to create a cipher text at the initial stage. In [19], three information, concealing strategies were proposed that is relying on DNA and they are viewed as the principle methods.

The primary strategy is the insertion technique of embedding bits from the original message in isolated locations in a DNA base sequence. The length of the first grouping is increased by this procedure because of insertion. The subsequent one is the complementary pair strategy with the goal that the lengthiest corresponding DNA series is recognized.. The last strategy is the substitution technique that is carried out by replacing a portion of the DNA nucleotides with different nucleotides. In [20], a substitution oriented plan was proposed by replacing the repetitive nucleotides in the DNA arrangement.

In this method, the first DNA arrangement is required by the beneficiary to recover the restricted information. To create the cipher text, another scheme was proposed in [21] through encoding an original text utilizing DNA positioned play fair code. The cipher text message is covered up in a DNA series utilizing substitution methodology. The approach proposed in [22] employed an uncertain DNA for key so that it can encode different types of information including sound and picture. An Arbitrary key will be produced at the sender for each instant and it will be utilized for decoding the encoded text at the receiving end. The major merit addressed in this method is the speed of encryption and decryption of the data. The procedure in [23] accomplishes hiding the plain text in DNA. The biological characteristics are unaltered, without extracting any additional information to be transferred to the recipient.

The work in [24] depicts a secure method to achieve one kind of DNA arrangements for encryption and decoding. This multi-facet security of the framework alongside DNA based AES encryption is accomplished

in [25]. The DNA strand is used to encrypt the data with a model [26].

In this paper, a two level encryption of data is proposed, which generates a non speech audio output of the input text. In addition, the encrypted and decrypted text is verified using the generalised cross correlation. The cross correlation is computed using Fast Fourier Transforms (FFT) to make the computations easy. The objectives of the work is twofold, first is to propose a method to encrypt the plain text into non speech audio, adding ambiguous by introducing DNA nucleotides. The second objective is to use (benchmark) the FFT technique to check the similarities between the original audio and the received audio after decryption. The remaining part of the paper is framed with section II as research methods used for the work. The section III deals with the results and discussion of the proposed work. In Section IV, the paper is concluded with the research outputs and the gap which could be filled up as future work.

2. Methods

An encryption procedure is genuinely secure if the cipher text created doesn't contain sufficient data to decide the relating plaintext regardless of how much cipher text is accessible or how much computational force the hacker has. Except for the one-time pad, no code is genuinely secure.

The security of an inconclusively secure algorithm relies upon the complication in turning around the hidden cryptographic issues. All codes for encoding, other than the one-time pad, are subjected to this type of encryption. In this work, the encryption is done in two levels. The block diagram in Figure 1, provides information about the whole system. Each character in the plain text is converted into its distinct unicode. The unicode is then converted into its equivalent binary. The binary code for each Unicode is of eight-bit length.

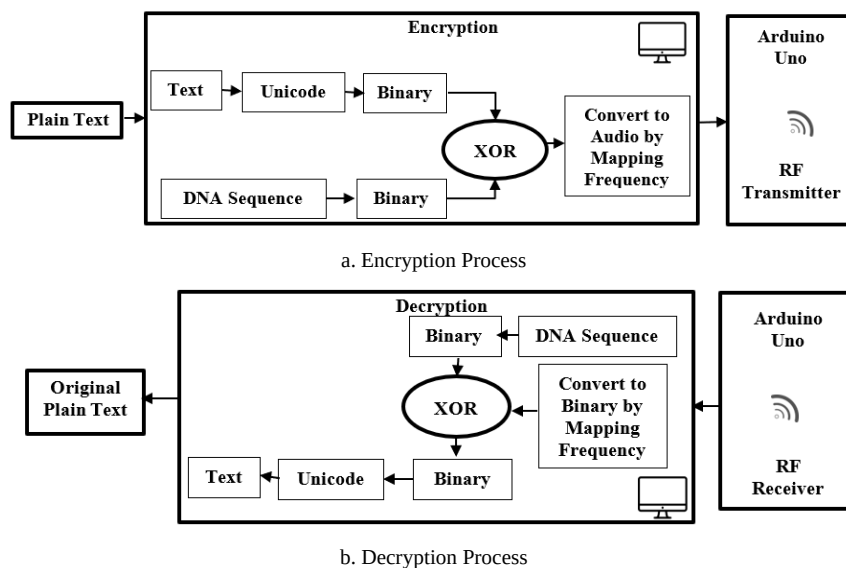


Figure 1. Block Diagram of the Proposed System

This Unicode is XORed with the Unicode of the random sequence to introduce security at level one itself. The resultant binary code is now grouped into two. For example, if the Unicode is 11011011, it is grouped into 11,01,10,11. So each character is converted into a group of four numbers of two-bit data. Now this two-bit data are mapped to Deoxyribonucleic acid (DNA). The nucleotides such as cytosine [C], guanine [G], adenine [A] and thymine [T] are matched with this two-bit binary data with the reference format as shown in Table 1. In the above said example, by refereeing to the table, the resulting sequence will be C, T, G and C. This forms the first level of encryption. The key for encryption will be the binary code mapping with Nucleotide pattern.

Table 1. Binary Mapping of Nucleotides

Binary Map	Nucleotide
00	Adenine (A)
01	Thymine (T)
10	Guanine (G)
11	Cytosine(C)

The second level of encryption is to convert the first stage cipher text into an audio. The cipher text from the first level is converted to sequence of musical notes. An audio note comprises of a key frequency f and its peak amplitude M . To create a basic sound, a sine function is used as in Equation 1.

$$Aud = M \sin(2 * \pi * f * t) \quad (1)$$

t is the length of note. This constitute a sinusoidal note with key frequency f as the time t advances. In this paper, the first level cipher text has only four characters namely AGCT. So each text character is defined with a constant frequency to construct the musical note as shown in Table 2.

Table 2. Conversion Table for Text Character to Audio Frequency

Text Character	Audio Frequency (Hz)
A	440
T	660
G	780
C	600

By using the above note for each character, a new audio file in the standard format like MP3 or .WAV is created and stored. This audio file has second level of encryption with keys as the frequency mapped to each DNA nucleotide. Now the fully encrypted audio file is ready for transmission. On hearing the audio, there will not be any text. It sounds only the music based on the mapping. The level of encryption can be improved by changing the frequency level for each character.

During the process of decoding, the receiver is the one who is accountable for carrying out the instructions. When the keys are used appropriately, the audio file is converted back into nucleotides. This process is repeated throughout the process. Throughout the entirety of the conversion process, the strategy of single processing is applied. In order to begin the process of decryption, the first step is to decrypt each coded frequency and then store it in a safe area. Only the

frequencies will be mapped to the nucleotides when the file is decoded at this level. This is because the nucleotides possess no other information.

The frequency levels are converted back into nucleotides by making use of the same key that is utilized in the process of encryption with the same key. Because of this, a cryptographic method that makes use of symmetric keys is utilized in the work that is being presented here. Once this phase has been reached, the decrypted file will only include text characters that are composed of AGCT pairs on an individual basis.

During the second level of decryption, these nucleotide pairs are converted into binary values by employing the same random sequence that was used as the key during the first level of encryption. This is done in order to decrypt the information. The information that is generated will be in binary format, and at this point, it will be organized into a group of eight. These eight-bit data are first translated into characters, and then they are converted into plain text. This is done with the intention of disclosing the original content.

The proposed encryption method is exhibited as Algorithm 1.

Algorithm 1. Encryption

Input:	Plain text with space and all special characters
Output:	Non Speech Audio file (WAV or MP3)
Step1:	Read the input text and compute the length of the text
Step2:	Generate a random DNA sequence with a length of input text
Step3:	Convert the input text and random DNA sequence to binary values
Step4:	Perform the bit by bit XOR operation of both the binary sequence
Step5:	The resultant bits are converted into DNA nucleotides with the help of the mapping scheme provided in Table1
Step6 :	An audio is created by mapping the nucleotides with the frequency as per table 2
Step7:	The audio file is stored

The proposed decryption technique is furnished as Algorithm 2.

Algorithm 2 Decryption

Input:	Non Speech Audio file (WAV or MP3)
Output:	Read the audio file and extract the frequencies of the distinct sound
Step1:	Convert the sound to DNA text file by referring to Table 2
Step2:	Decipher the DNA text into Binary bits with the help of mapping in Table1
Step3:	Perform the bit by bit XOR operation of the binary file with the random generated DNA binary file in step 3 of encryption algorithm1
Step4:	The resultant bits are converted to plain text.
Step5:	The plain text is stored

The experimental setup for validation of the proposed work involves the use of two Arduino Uno boards configured to act as independent transmitter and receiver units as depicted in Figure 2.

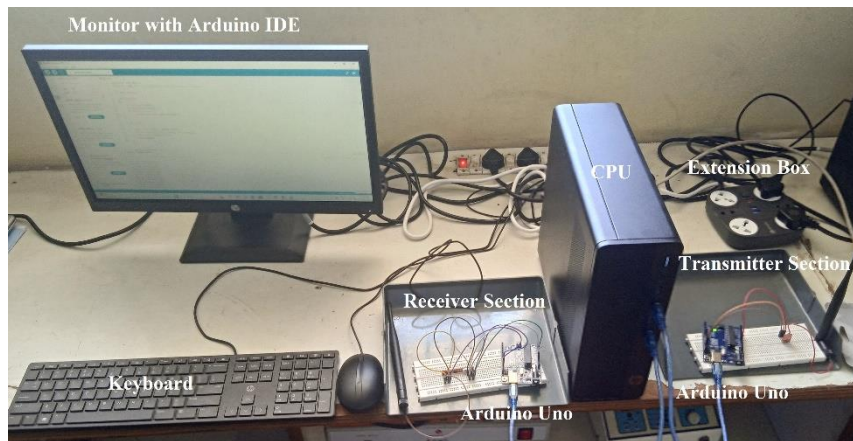


Figure 2. Experimental Setup

These boards facilitate a practical and low-cost platform for wireless communication testing. The transmission of the audio file is achieved through the use of RF modules operating at a frequency of 433 MHz. At the transmitter section, an Arduino Uno board interfaces with the audio file storage and an RF transmitter module. The encoded audio data is transmitted wirelessly across a predefined distance to the receiver unit.

The receiver unit, consisting of another Arduino Uno board, is connected to an RF receiver module that captures the transmitted audio signals. Both the transmission and reception operations are programmed using the Arduino IDE, enabling smooth handling of the incoming and outgoing data streams. Careful synchronization between the two boards ensures the integrity of the transmitted audio data, even with the relatively limited bandwidth and basic modulation techniques employed by the 433 MHz RF modules. The selection of the 433 MHz RF modules was strategic, considering their widespread availability, ease of interfacing with Arduino platforms, and adequate range for experimental setups. These modules, being cost-effective, allowed a robust demonstration of data transmission techniques without the complexity and overhead associated with more sophisticated wireless technologies.

The design of the transmitter focused on efficient packetization of audio data to minimize latency and ensure that the transmitted signals could be accurately reconstructed at the receiver side. Additionally, the transmission protocol was customized to embed error detection capabilities, enhancing the robustness of the data communication process.

The experimental process began with the audio file being sampled and encoded into a digital format compatible with Arduino memory constraints. Due to limited onboard memory, the audio data was segmented and transmitted sequentially. Each segment was assigned a packet identifier to maintain the order during reconstruction at the receiver. The transmitter Arduino managed the segmentation, modulation, and timed release of data packets to the RF module. Upon

reception, the receiver Arduino demodulated the incoming signals, verified packet integrity using checksums, and sequentially reassembled the data into a coherent audio stream.

Challenges encountered during the setup included data packet loss due to noise and interference, which are typical in unlicensed RF bands like 433 MHz. To mitigate these issues, retransmission mechanisms were incorporated, where the receiver requested retransmission upon detecting corrupted or missing packets. Signal integrity was further enhanced by optimizing antenna placement and ensuring clear line-of-sight communication between the transmitter and receiver units. Power supply stability was also a critical factor; both Arduino boards were through regulated sources to avoid signal fluctuations that could impact data integrity.

3. Results and Discussions

A simulation of the suggested technique is performed using the MATLAB software on an Intel computer. In the course of this simulation, four distinct plain text are taken into consideration. The work in the literature reports only the alphabets. Hence, the text under test are selected such that it includes a numeral, a special character, a space and running text without space. For the sake of simulation, the plain language that is used is "Meet me at 7th mile." It is made up of fourteen alphabets, one integer, and four different numbers of spaces, which together make up a total of nineteen characters. The other plain text is chosen in the same manner, with consideration given to the various combinations of characters that are presented in Table 3.

At first level the plain text is converted into its Unicode. The Unicode values are then altered to binary values. To introduce the randomness, a random nucleotide sequence is generated. A Logical XOR operation is carried out with the binary values of plain text with the random sequence. The resultant binary values are grouped such that each group consists of two bits. These two bits are mutated as DNA nucleotides based on the mapping scheme. The mapping discussed in section 2 is

applied in this work. The simulated encrypted text for different plain text is shown in Table 4.

Table 3. Plain Text Input for Simulation

Plain Text	Description
Meet me at 7th mile	14 alphabets, 1 numeral and 4 number of spaces
Hello	5 alphabets with one capital letter
Execute Plan @ 1.20am	13 alphabets, 3 numerals, 2 special characters and 3 number of spaces
savechief	Two words in running without space. 9 alphabets only

Table 4. DNA Encrypted Text

Plain Text	DNA Encrypted Text
Meet me at 7th mile	AAAAAAAAAATAGTGTAATCCCCC CCCGAATAATACGATAAATTCATTCCA ATGAAAGCATTAGTGTAATGGAT
Hello	AAAAATCCAGCCTATAGTT
Execute Plan @ 1.20am	AAAAAAAAAAAAAGATTCCATCCCT CGCCGATGTAGTGGGTAAATTAAGA GGAGAGTTTTCGGGGTCAAAAAAGC GTGTGG
savechief	AAAAAAAAACCCCTTTTAATGTTCTG ACCAGTTCA

These mapping can be changed accordingly for any type of encryption. These text in nucleotide format are now

transformed into music by executing the second level of encryption. The different plain text considered for this study, its size and its encoded music size are listed in Table 5.

Table 5. Size of the Plain text and its audio file

Plain Text	Text size	Music file size
Meet me at 7th mile	19 Characters	364.9 Kb
Hello	5 Characters	96.084
Execute Plan @ 1.20am	21 Characters	403.12
savechief	9 Characters	172.96

The two level encrypted cipher is now in the form of non-speech audio. This audio in the wave file format is decrypted using the keys in encryption stage. The deciphering requires two stages. First to convert the audio file into DNA sequence and the back to binary format. Once the binary format is obtained, the original text can be retrieved from it. To study the pace of encryption and decryption process, the time required to complete each process is provided in Figure 3. The execution time is computed by considering the average time required to run each process for 100 times with the same input.

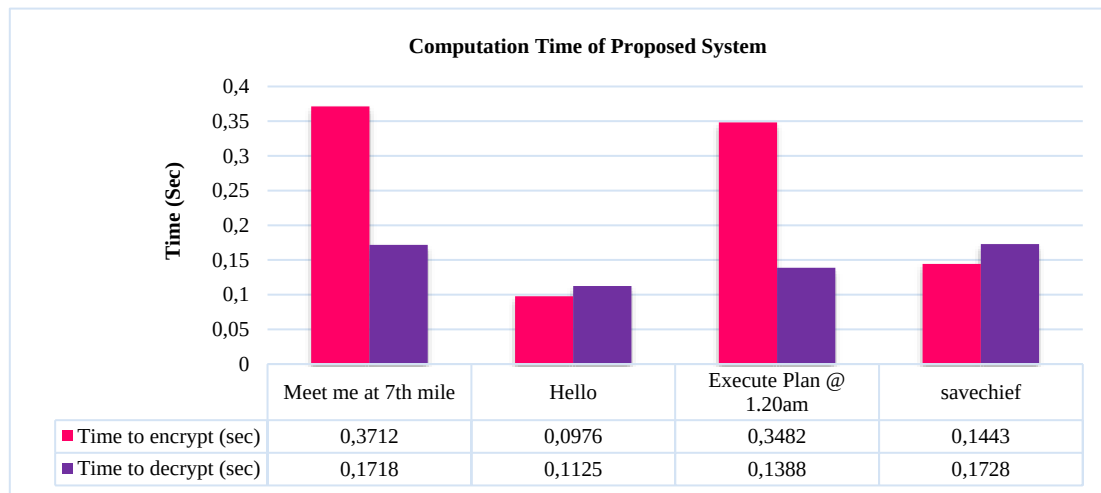


Figure 3 Comparison of Computation Time

A sample screen shot of command window for the plain text “savechief” is provided as Figure 4.

```

Command Window
Enter the Plain Text:savechief
The DNA Encrypted Text is:
AAAAAAAAAACCCCTTTTAATGTTCTGACCAGTTCA
The time taken to Encrypt the Text is:
0.1443

You are hearing the DNA encrypted audio
The time taken to Decrypt the Text is:
0.1728

The size of the audio file (in Bytes) is:
172916

The reconstructed text from sound:
savechief
fx >> |

```

Figure 4. Screenshot of the text “savechief”

To visualize the audio file, the encrypted file is plotted as periodogram in Figure 5. It depicts that there is no relation between the plain text and the waveform obtained in periodogram. Further, to check the similarity between the original signal and received signal, cross correlation is performed using the Fast Fourier Transform as shown in Figure 6. This study involves a lengthy signal hence a suitable FFT algorithm [27], [28] is used for lengthy dataset to get faster response.

On comparing the proposed method with the approach detailed in [25], a key enhancement lies in the robustness introduced at the first level of encryption. In the proposed scheme, the initial stage involves a permutation-based transformation of nucleotide

sequences followed by a character-level mapping. This dual-layer structure significantly increases the complexity for any brute-force attempt.

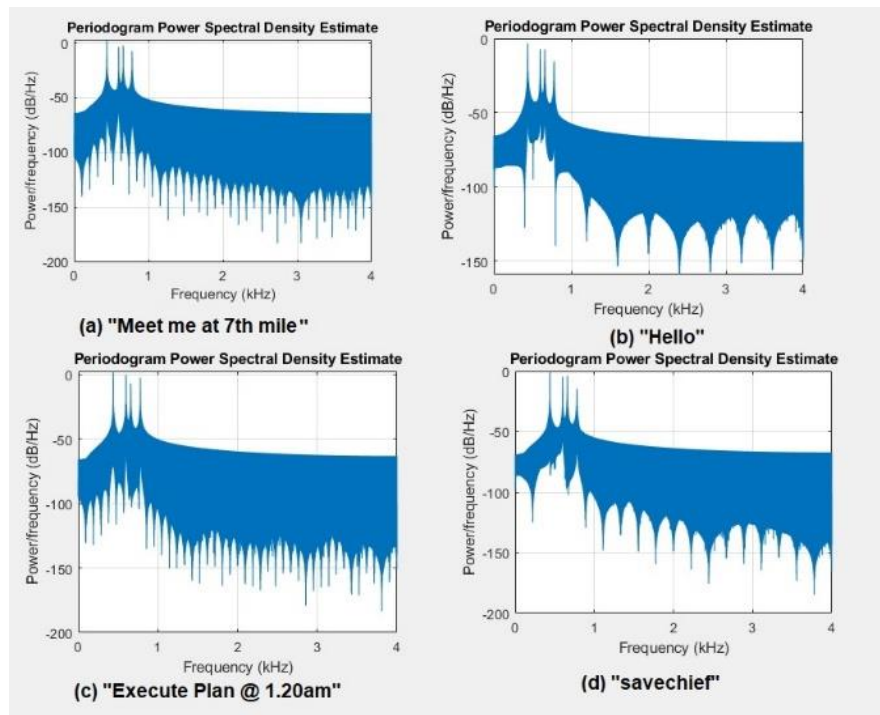


Figure 5. Periodogram of Encrypted Text

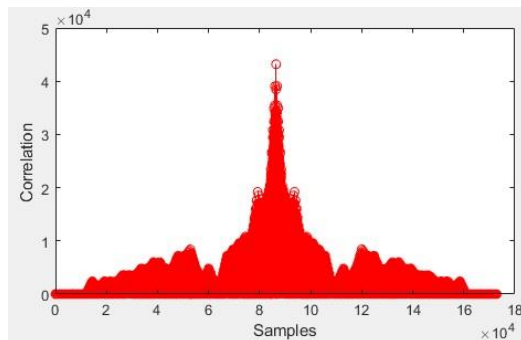


Figure 6. Correlation of Original and Decoded Signal

As illustrated in Equation 2, the probability P of correctly guessing the encrypted output by random chance is:

$$P = \frac{1}{4! \times 26!} \quad (2)$$

This stems from the fact that the encryption algorithm permutes the four DNA bases (A, T, C, G), resulting in $4!$ possibilities, and simultaneously maps 26 English letters in a specific order, contributing another $26!$ combinations. The resulting search space is substantially larger than that of conventional monoalphabetic or polyalphabetic ciphers used in [25], thereby rendering brute-force attacks computationally infeasible.

This exponential increase in keyspace highlights a significant security advantage of our approach. Furthermore, by incorporating non-speech audio

sonification in subsequent layers, the encryption not only resists conventional cryptanalysis but also obfuscates patterns that could be exploited via frequency analysis—an inherent limitation in [25]. Thus, the proposed model presents an innovative and multi-dimensional framework for secure text encoding, blending bio-inspired principles with auditory masking to enhance data confidentiality.

Even though the first level returns a feasible probability, the second level of encryption is highly secured as the probability doesn't exist to decode the audio with a particular frequency which is evident from Figure 5.

Table 6. Comparison of Proposed Model

Method	Plain Text	Time to encrypt (sec)	Time to decrypt (sec)
[21]	5 Characters	8.325	5.346
[22]	9 Characters	4.321	3.724
[23]	5 Characters	4.221	3.891
[24]	9 Characters	3.321	2.981
[25]	9 Characters	3.331	2.991
Proposed	19 Characters	0.371	0.172
	5 Characters	0.098	0.113
	21 Characters	0.349	0.139
	9 Characters	0.144	0.173

The encryption and decryption time of the proposed method is compared with several existing methods reported in the literature, as summarized in Table 6. It is observed that the proposed approach demonstrates a clear advantage in terms of processing speed. Compared to traditional methods that often involve higher complexity operations or lack efficiency in

handling large datasets, the proposed method capitalizes on its lightweight structure and optimized computation flow to deliver faster results. Notably, the proposed method consistently outperformed conventional encryption techniques, particularly when dealing with large data volumes where time efficiency becomes critical. The faster execution not only highlights the technical advantage of the proposed system but also reinforces its applicability for real-time and resource-constrained environments.

4. Conclusions

With the ascent of advanced computers, a preserved data communication has become an undeniably significant assignment. A cryptographic framework ought to be planned so that it is hard to break. A nitty gritty investigation of the current DNA based cryptographic procedures unmistakably shows that DNA based encryption can improve the security of the information. The proposed method in this work delivers a novel idea on hiding the data into audio. There is space for improvement in the presented work while encoding the DNA sequence. The strong cipher techniques could be used in future work at first level of encryption with amino acids.

Acknowledgements

The authors declare no conflict of interest. This research was funded by SRM Valliammai Engineering College under SEED Money project with File No. SRMVEC/R&D/2024/011.

References

- [1] C. Landelle, J. Caron-Guyon, B. Nazarian, J.L. Anton, J. Sein, L. Pruvost, M. Amberg, F. Giraud, O. Félician, J. Danna, and A. Kavounoudias, "Beyond sense-specific processing: decoding texture in the brain from touch and sonified movement," *iScience*, vol. 26, no. 10, pp. 107965, 2023. <https://doi.org/10.1016/j.isci.2023.107965>
- [2] C. Fernandez, J.A. Freitas, P. Blikstein, and R.D. Lopes, "The design space of visualization tools for data science education: Literature review and framework for future designs," *International Journal of Child-Computer Interaction*, vol. 43, pp. 100698, 2025. <https://doi.org/10.1016/j.ijcci.2024.100698>
- [3] M. Matei and L. Alboae, "Enhancing accessible navigation: A fusion of speech, gesture, and sonification for the visually impaired," *Procedia Computer Science*, vol. 246, pp. 2558–2567, 2024. <https://doi.org/10.1016/j.procs.2024.09.442>
- [4] A. Dam, Y. Lee, A. Siddiqui, W.S. Lages, and M. Jeon, "Audio augmented reality using sonification to enhance visual art experiences: Lessons learned," *International Journal of Human-Computer Studies*, vol. 191, pp. 103329, 2024. <https://doi.org/10.1016/j.ijhcs.2024.103329>
- [5] J. Song, Y. Wang, X. An, S. Ma, D. Wang, T. Gan, H. Shi, Z. Yang, and H. Liu, "Novel sonification designs: Compressed, iconic, and pitch-dynamic auditory icons boost driving behavior," *Applied Ergonomics*, vol. 103, pp. 103797, 2022. <https://doi.org/10.1016/j.apergo.2022.103797>
- [6] E.F. González-Castañeda, A.A. Torres-García, C.A. Reyes-García, and L. Villaseñor-Pineda, "Sonification and textification: Proposing methods for classifying unspoken words from EEG signals," *Biomedical Signal Processing and Control*, vol. 37, pp. 82–91, 2017. <https://doi.org/10.1016/j.bspc.2016.10.012>
- [7] M. Maćkowski, P. Brzoza, and D. Spinczyk, "An alternative method of audio-tactile presentation of graphical information in mathematics adapted to the needs of blind," *International Journal of Human-Computer Studies*, vol. 179, pp. 103122, 2023. <https://doi.org/10.1016/j.ijhcs.2023.103122>
- [8] W. Woo, W. Richards, J. Selker, and C. Udell, "WeatherChimes: An open IoT weather station and data sonification system," *HardwareX*, vol. 13, pp. e00402, 2023. <https://doi.org/10.1016/j.ohx.2023.e00402>
- [9] M. Mashiat, T. Ali, P. Das, Z. Tasneem, M.F.R. Badal, S.K. Sarker, M.M. Hasan, S.H. Abhi, M.R. Islam, M.F. Ali, M.H. Ahamed, M.M. Islam, and S.K. Das, "Towards assisting visually impaired individuals: A review on current status and future prospects," *Biosensors and Bioelectronics: X*, vol. 12, pp. 100265, 2022. <https://doi.org/10.1016/j.biosx.2022.100265>
- [10] M. Singh, M.S. Kanroo, H.S. Kawoosa, and P. Goyal, "Towards accessible chart visualizations for the non-visuals: Research, applications and gaps," *Computer Science Review*, vol. 48, pp. 100555, 2023. <https://doi.org/10.1016/j.cosrev.2023.100555>
- [11] H. Jalali, M. Pouladian, A.M. Nasrabadi, and A. Movahed, "Sleep stages classification based on feature extraction from music of brain," *Heliyon*, vol. 11, no. 1, pp. e41147, 2025. <https://doi.org/10.1016/j.heliyon.2024.e41147>
- [12] M. Bisla and R.S. Anand, "A comprehensive review on state-of-the-art imagined speech decoding techniques using electroencephalography," in *Artificial Intelligence in Biomedical and Modern Healthcare Informatics*, M.A. Ansari, R.S. Anand, P. Tripathi, R. Mehrotra, and M.B.B. Heyat, Eds. Academic Press, 2025, pp. 101–126. <https://doi.org/10.1016/B978-0-443-21870-5.00011-XA>
- [13] T. Hermann and H. Ritter, "Sound and meaning in auditory data display," *Proc. IEEE*, vol. 92, no. 4, pp. 730–741, 2004. <https://doi.org/10.1109/JPROC.2004.825904>
- [14] P. Vickers and R. Höldrich, "Direct Segmented Sonification of Characteristic Features of the Data Domain," pp. 244–253, 2019. <https://doi.org/10.21785/icad2019.043>
- [15] S. Barrass, "Developing the practice and theory of stream-based sonification," *Scan J. Media Arts Cult.*, vol. 6, no. 2, 2009. http://scan.net.au/scan/journal/display.php?journal_id=135
- [16] Dragicevic, P., Jansen, Y., Vande Moere, A., Data Physicalization. In: Vanderdonckt, J., Palanque, P., Winckler, M. (eds) *Handbook of Human Computer Interaction*. Springer, Cham.. 2021 https://doi.org/10.1007/978-3-319-27648-9_94-1
- [17] R. Dean, *The Oxford Handbook of Computer Music*. New York: 2009.
- [18] B. D. Patnala and K. K. Reddi, "DNA cryptography.life blood for new ERA computers," in *Proc. 2017 Int. Conf. Energy, Commun. Data Anal. Soft Comput. ICECDS 2017*, pp. 1178–1184, 2018. <https://doi.org/10.1109/ICECDS.2017.8389627>
- [19] H. J. Shiu, K. L. Ng, J. F. Fang, R. C. T. Lee, and C. H. Huang, "Data hiding methods based upon DNA sequences," *Inf. Sci. (Ny)*, vol. 180, no. 11, pp. 2196–2208, 2010. <https://doi.org/10.1016/j.ins.2010.01.030>
- [20] C. Guo, C. C. Chang, and Z. H. Wang, "A new data hiding scheme based on DNA sequence," *Int. J. Innov. Comput. Inf. Control*, vol. 8, no. 1 A, pp. 139–149, 2012. <https://doi.org/10.25007/ajnu.v6n3a83>
- [21] T. Mandge and V. Choudhary, "A DNA encryption technique based on matrix manipulation and secure key generation scheme," in *Proc. 2013 Int. Conf. Inf. Commun. Embed. Syst. ICICES 2013*, pp. 47–52, 2013. <https://doi.org/10.1109/ICICES.2013.6508181>
- [22] B. Akiwate and L. Parthiban, "A Dynamic DNA for Key-based Cryptography," in *Proc. Int. Conf. Comput. Tech. Electron. Mech. Syst. CTEMS 2018*, pp. 223–227, 2018. <https://doi.org/10.1109/CTEMS.2018.8769267>

- [23] G. Hamed, M. Marey, S. A. El-Sayed, and M. F. Tolba, "Hybrid technique for steganography-based on DNA with n-bits binary coding rule," in Proc. 2015 7th Int. Conf. Soft Comput. Pattern Recognition, SoCPaR 2015, pp. 95–102, 2016.
<https://doi.org/10.1109/SOCPAR.2015.7492790>
- [24] S. V. K. Priya and S. J. Saritha, "A robust technique to generate unique code DNA sequence," in Proc. 2017 Int. Conf. Energy, Commun. Data Anal. Soft Comput. ICECDS 2017, pp. 3815–3820, 2018.
<https://doi.org/10.1109/ICECDS.2017.8390178>
- [25] C. Zou, X. Wei, Q. Zhang, C. Zhou, and S. Zhou, "Encryption Algorithm Based on DNA Strand Displacement and DNA Sequence Operation," IEEE Trans. Nanobioscience, vol. 20, no. 2, pp. 223–234, 2021.
<https://doi.org/10.1109/TNB.2021.3058399K>
- [26] S. Sajisha and S. Mathew, "An encryption based on DNA cryptography and steganography," in Proc. Int. Conf. Electron. Commun. Aerosp. Technol. ICECA 2017, vol. 2017-Janua, pp. 162–167, 2017.
<https://doi.org/10.1109/ICECA.2017.8212786>
- [27] C. Saravanakumar and N. U. Bhanu, "Speed Efficient Fast Fourier Transform for Signal Processing of Nucleotides to Detect Diabetic Retinopathy Using Machine Learning," J. Med. Imaging Heal. Informatics, vol. 12, no. 1, pp. 27–34, 2022.
<https://doi.org/10.1166/jmihi.2022.3922>
- [28] S. Chandrasekaran and U. B. Nageswaran, "A mutated addition–subtraction unit to reduce the complexity of FFT," Appl. Nanosci., vol. 13, no. 4, pp. 2935–2944, 2023.
<https://doi.org/10.1007/s13204-021-02278-5>