Securing Healthcare Communication: A Novel Deep Learning Framework with Imperceptible Steganography and Arrhythmia Detection

[1#]G.Yogarajan, [2]S.A.Ekanathan, [3]S.Saivijay, [4]M.Rajasekar

Abstract - **This article shows a novel, deep learning-based framework for securing communication in various healthcare applications like Electrocardiogram (ECG), Photoplethysmography (PPG), and (EEG) Electroencephalogram. It introduces a robust and imperceptible steganography technique that simultaneously achieves data protection and clinical analysis. The proposed method transforms signals and confidential information into blocks, facilitating secure data embedding through a novel Hermite domain technique that minimizes information loss. Additionally, it integrates a Hippopotamus Optimization (HO) approach to address potential data loss during decryption, ensuring information integrity. Furthermore, a Supervised Long-Short Term Memory (Su-LSTM) network is implemented to predict and reconstruct lost signal segments, offering a partially reversible process with minimal distortion. The key novelty lies in the integration of arrhythmia detection within the steganography framework. This enables independent and multiple data embedding within a single signal while simultaneously performing real-time analysis for potential heart rhythm abnormalities. This innovative approach demonstrates high storage capacity for concealed information while preserving the integrity and confidentiality of the original signal. This research contributes significantly to the field of information security in healthcare data management, facilitating secure and efficient communication in various medical applications.**

Index Terms **– Electrocardiogram (ECG), Photoplethysmography (PPG), Electroencephalogram (EEG), Particle Swarm Optimization (PSO), Hippopotamus Optimization (HO), Supervised Long Short Term Memory (Su-LSTM), Arrhythmia detection.**

G Yogarajan, BE, ME, PhD, Associate Professor Department of Information Technology, Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India (yogarajang@yahoo.com)

S A Ekanathan, UG Student, Department of Information Technology, Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India (ekanathanragu6245@gmail.com)

S Saivijay, UG Student, Department of Information Technology, Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India (saivijays2599@gmail.com)

M Rjasekar, UG Student, Department of Information Technology, Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India (rajasekar.rj100@gmail.com)

1. INTRODUCTION

With the rise of e-healthcare, safeguarding patients' privacy has emerged as a significant concern. Steganography offers a sophisticated approach to embed patient information within physiological signals without noticeable distortion, enabling retrieval through reverse data-insertion techniques. Key terms in steganography include the cover signal (original bio-signal), stego-data signal (post-secret bit insertion), and reconstructed signal (signal after secret bit extraction. In addition to steganography, there exist several well-known data protection methods such as AES - Advanced Encryption Standard and PKC - public key cryptography. These techniques, collectively known as cryptography, involve encoding plaintext (the hidden information in steganography) into cipher text using a secret key. Unlike steganography, which embeds information within a cover medium, cryptography requires sending both the encrypted data and the original bio-signal, along with the secret key, over a secure channel. Steganography offers a unique advantage as it conceals the transmission of secret information from potential hackers by embedding it within a cover medium, thus making it safer. Despite being less complex and costly compared to PKC or AES, steganography's effectiveness may diminish if machine learning techniques are incorporated into its algorithm.

While cryptography provides stronger data security, the separate transmission of encrypted data and the original bio-signal in cryptography-based bio-signal transmission systems can potentially expose the secret data to hackers. However, combining steganography with encryption enhances overall data security, albeit with increased management complexity and effort. When implementing steganography on biological signals, it typically involves two main phases: encryption and decryption. In encryption, an algorithm which is bidirectional is utilized to embed the secret message within the signal, while in decryption, this hidden message is retrieved using the algorithm called reverse encryption. It's worth noting that steganography generally introduces less distortion errors. Consequently, techniques which are reversible such as DWT, DFT, and DCT are commonly applied to signals. Encryption of data often occurs within higher frequency sub-bands, which results in minimal errors after reconstruction and ensuring the security of the secret message within those sub-bands. Several characteristics should be considered in steganography, including reliability, reversibility, robustness, and capacity. Reliability is typically maintained through the use of scrambling matrices or secret keys in previous works. Reversibility is gaining traction, ensuring the recovery of the original signal values post-decryption. Capacity depends on

the chosen algorithm, but increasing capacity can lead to higher error rates. However, robustness remains an understudied aspect. In scenarios where segments of the signal are lost during transmission or due to attacks, known as "Removal attacks," there's a lack of research on recovering missing secret information, which could potentially lead to loss of entire data. To address this, a steganography strategy using the user defined Hermite function (HF), secret bit insertion process is done in order to maintain the inter-bit relationship. This approach divides the data into blocks and applies encryption procedures to embed secret bits within each block. Additionally, it will safeguard against data loss or tampering during transmission or attacks. If everything goes smoothly and no information is lost, a specific technique and a passkey are used to extract the hidden data. This likely involves calculations to reverse the hiding process and restore the original data points within the cover signal. Things get more interesting when errors occur. The system first checks for a password mismatch, which could indicate missing information. Then, it employs a special technique called Hippopotamus Optimization (HO) to predict the characteristics of the missing data block. Imagine an intelligent particles working together to estimate what the lost data might have looked like. Next, a powerful AI model called a Supervised Long-Short Term Memory (Su-LSTM) Recurrent Neural Network comes into play. This model, trained on existing data, predicts the specific hidden bits that were originally embedded within the corrupted block. Finally, with the missing block's features and the predicted hidden bits, the system attempts to reconstruct the original data point values within the corrupted block. This combination of deep learning and optimization techniques allows the system to be robust and recover the hidden information as accurately as possible, even when parts of it go missing.

2. RELATED WORKS

Several articles have explored reversible watermarking, demonstrating reduced distortion error after the data extraction in the cover signal. The paper [1] introduces a novel steganography approach leveraging the LSB substitution method, promising significant advancements in reliability, security, imperceptibility, capacity, and robustness over conventional methods. Experimental results demonstrate significant improvements, with the proposed approach outperforming current methodologies by up to 6.77 percent in PSNR and offering enhanced performance across various image types and sizes. The researcher behind [2] emphasizes the importance of securing JPEG compression processors within industrial control systems, particularly in critical infrastructure sectors. By incorporating resilient structural obfuscation techniques and hardware-based steganography, the proposed method significantly bolsters security measures against potential threats such as unauthorized access, data tampering, and malicious implantation of Trojans. This innovation ensures the authenticity and integrity of compressed images transmitted across networks, thus safeguarding the operational integrity of vital infrastructure systems. This paper [3] addresses the critical need to safeguard patient data and devices in IoMT systems, proposing a novel taxonomy and discussing privacy and security solutions to mitigate risks and guide future research efforts. In paper [4], the researchers introduce an innovative approach to image steganography known as Image Region Decomposition (IRD), specifically tailored to enhance the imperceptibility and data hiding capacity of satellite images. The IRD algorithm intelligently breaks down satellite images into distinct low, medium, and high-intensity regions. By precisely manipulating the 2nd and 1st LSBs in the low-intensity region, this technique achieves remarkable improvements in imperceptibility and the ability to embed data, surpassing the capabilities of current methodologies. In response to IoT's impact on healthcare data security, the author of [5] proposed a hybrid model integrating 2D-DWT steganography and a hybrid encryption scheme to secure diagnostic text data in medical images. Evaluation based on statistical parameters such as PSNR and MSE demonstrates the model's ability to conceal confidential patient data with high imperceptibility and capacity.

In paper [7], a pioneering mixed-signal system is introduced for personalized remote environmental monitoring, showcasing numerous breakthroughs such as a low-power analog front end, real-time detection of environmental boundaries, simplified feature extraction, and dynamically adjustable sensor resolution. The author of [8] addresses limitations in previous ECG analysis methods by proposing a model that accurately segments beats, reduces noise, and preserves diagnostic information. Utilizing Hermite and sigmoid functions alongside piecewise polynomial interpolation, the model effectively cancels baseline wander, performs beat segmentation, and provides low-dimensional waveform representation, showcasing improved denoising and segmentation compared to existing algorithms. This approach offers a versatile framework adaptable to various biomedical signals, facilitating understanding of morphological variations influenced by factors like respiration and medication. This paper [9] addresses the need for objective distortion measures to evaluate the quality of reconstructed PPG signals, crucial for real-time denoising and compression. Four objective distortion measures are compared for their performance in predicting quality accurately and efficiently, considering different types of waveform distortions introduced by various processing techniques.

The author of [10] introduced Hippopotamus Optimization (HO), which is a new method inspired by hippopotamus behavior for solving complex problems. It tackles various optimization tasks effectively, achieving superior results compared to established algorithms in benchmark tests. This suggests HO excels at both exploring different solutions and refining promising ones. This paper [11] explores using the unsupervised wavelet transform to remove eye blink interference from single-channel EEG data in real-time applications. The analysis suggests that this technique can be effective, with specific wavelet functions performing best. The author of [12] introduced a new data-driven soft sensor model, LSTM-DeepFM, is proposed for industrial applications. This paper [13] proposed a deep learning method for detecting fetal arrhythmia using fetal ECG signals, achieving high accuracy. The method is less affected by errors in heartbeat detection compared to conventional

methods. This paper [14] introduces a new deep learning network for ECG analysis. The network can automatically detect and classify 45 different heart arrhythmias with high accuracy, even exceeding previous convolutional neural network methods. The article [22] introduces deep learning-based technique for imperceptible biological signal steganography, ensuring resilience against manipulation and privacy breaches.

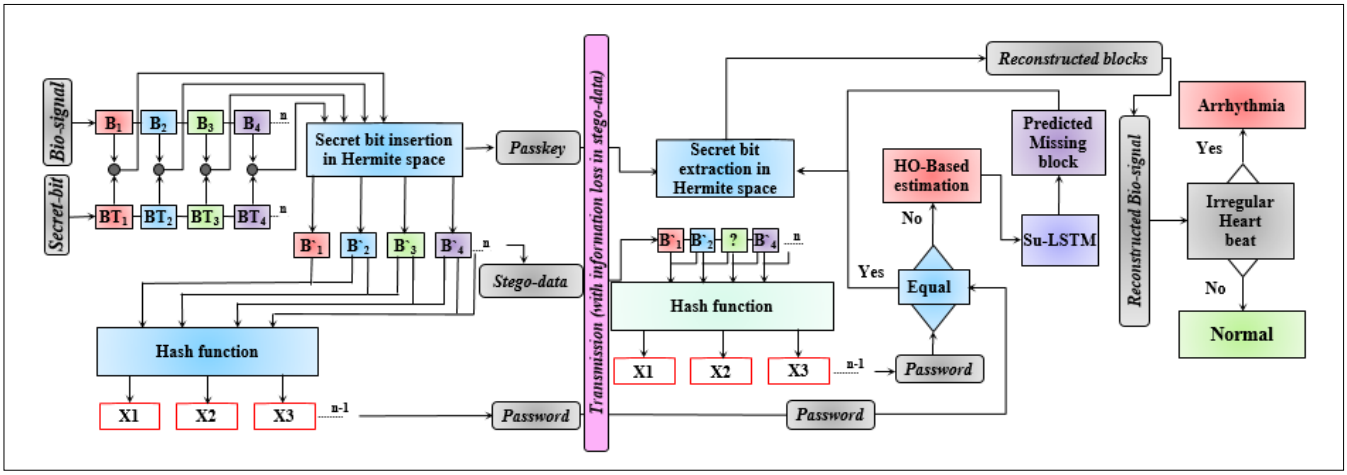

Fig. 1. Depicts the overall model of the steganography algorithm involves insertion of data and extraction processes, acknowledging the possibility of data loss during transmission and Arrhythmia detection.

By leveraging Hermite domain insertion and extraction alongside PSO-based data recovery and Su-LSTM RNN prediction, the method exhibits low distortion errors and high storage capacity for diverse signals like ECG, PPG and EEG.

3. METHODOLOGY

The Data encryption and decryption mechanism of the Bio-signal block is shown in Fig. 1. The Bio-signal is firstly taken using Arduino UNO and heart rate monitor sensor is shown in Fig. 2.

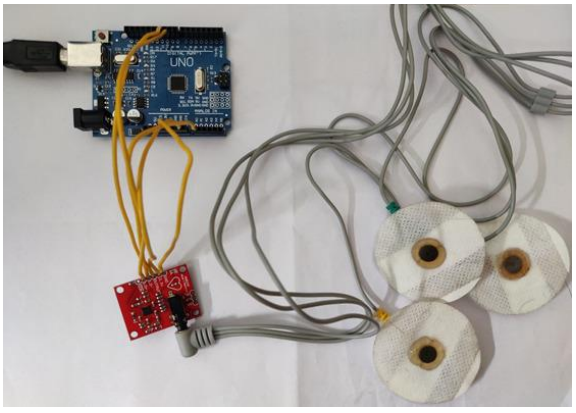

Fig. 2. Depicts the hardware model for taking Bio-signal from the patient.

For each four consecutive data points is considered as a block. Likewise, n number of blocks are taken. In order to embed the secret bit, we need to take any secret information and converted into a binary form for n number of blocks. For each blocks, 8 bit of secret information is inserted using Hermite space algorithm, results in generation of stego-data and storing of Hermite orders in form of "Passkey". The "Password" is generated from the current and the next stego-block using hash function. The Passkey, stego-data and password are transmitted through cloud server.

The received stego-data is used to generate the password in receiver side, if it exactly matches with the received password, then there is no loss occurs and the blocks and the bio-signal are reconstructed using secret bit extraction and Arrhythmia can be detected. If the password doesn't match, then by using HO and Su-LSTM method, the position of the lost block and the data points are to be recovered.

3.1 Encryption – Secret Bits Insertion

In Encryption, the four consecutive data points are taken as a block. Likewise, n number of blocks are taken. The Secret information is passed as an input and converted into binary 8 bit information. The block and the binary data are given as an input to the forward Hermite function "$H_n(\bullet)$" attaining the Hermite coefficient $Bl_k[t_i] = He_n\ (Bl_k[s_i])$ where n is the order of HF (where n = 1,2,…,6). It has been found that the high frequency (HF) produces a comparable waveform to the respective bio-signal for values of n equal to 1, 2, 3, 4, 5 and 6. The table 1 shows the Forward Hermite function for various orders.

Forward Hermite function

Input : Cover block *s*, Hermite order n
Output : Transformed block *t*

Start
hrm_block = []
for **block$_i$** in *s*:
if n = 1
result = **block$_i$** + 2
elif n = 2
result = **block$_i$***2 - **block$_i$** + 2
elif n = 3
result = **block$_i$***3 - **block$_i$***2 + 2
elif n = 4
result = **block$_i$***4 - **block$_i$***3 + 2
elif n = 5
result = **block$_i$***5 - **block$_i$***4 + 2

```
    elif n = 6
       result = block_i*6 - block_i*5 + 2
    else
       raise ValueError("n, must be between 1 and 6")
    hrm_block.append(result)
  return hrm_block
End
```

Table 1. Forward Hermite function

. Consequently, the HF order for each block was selected to minimize the distortion between $Bl_k[s_i]$ and $Bl_k[t_i]$. Following the computation of $Bl_k[t_i]$, the steganography method is implemented to minimize the distortion error after decryption [22]. For each data point in $Bl_k[t_i]$ was rounded off to the nearest three decimal point whereas the third decimal, denoted as 'd' was split into two parts, d1 and d2, where d1 + d2 = d. As previously mentioned, each data point of the block encodes two bits of information, represented by bit_1 and bit_2 (thus, 0 <= bit_1, bit_2 <=1 and bit_1, $bit_2 \in$ Z). While d1 remains unchanged, d2 is modified with secret bits using a new algorithm (refer to Algorithm 1), resulting in an updated value $\tilde{t}_i$.

Subsequently, the stego-data for the kth block, $Bl_k[\tilde{t}_i]$ is generated by applying reverse HF, calculated as $Bl'_k[\tilde{s}_i] = He_n^{-1}(Bl_k[\tilde{t}_i])$ (refer to Figure 2). The distortion error between t_i and $\tilde{t}_i$ is minimal since they are nearly identical up to the second decimal place. Moreover, after the reverse HF, the error between s_i and $\tilde{t}_i$ decreases, thereby ensuring the security of the data. The order of HF is securely stored in the form of passkey, which is essential for secret bit extraction in decryption.

The table 2 shows the Reverse Hermite function for various orders.

Reverse Hermite function

Input : Transformed block $\tilde{\boldsymbol{t}}$, Hermite order n
Output : Stego block $\tilde{\boldsymbol{s}}$

```
Start
  x = sp.Symbol('x')
  eqn = None
  if n = 1
     eqn = x - (y - 2)
  elif n = 2
     eqn = x*2 - x - (y - 2)
  elif n = 3
     eqn = x*3 - x*2 - (y - 2)
  elif n = 4
     eqn = x*4 - x*3 - (y - 2)
  elif n = 5
     eqn = x*5 - x*4 - (y - 2)
  elif n = 6
     eqn = x*6 - x*5 - (y - 2)
  else
     raise ValueError("n, must be between 1 and 6")
  solutions = sp.solve(eqn, x)
  s̃_i = [sol.evalf() for sol in solutions if sol.is_real]
  return s̃
End
```

Table 2. Reverse Hermite function

Now, for the password generation, by taking the current and the next stego-block, password is generated using the hash function. For n number of stego blocks, n-1 number of passwords are generated and transmitted. If there is no loss of data occurs, we can directly reconstruct the blocks and the bio signal using the passkey but if there is any loss of blocks, it can be recovered by using HO and Su-LSTM and the blocks are recovered.

Algorithm 1 – Insertion of secret bit in Hermite space

Input : Cover block $\boldsymbol{Bl_k[s_i]}$ and secret $\boldsymbol{bit_1}$ and $\boldsymbol{bit_2}$
Output : Stego-block $\boldsymbol{Bl'_k[\tilde{s}_i]}$

Start
$\boldsymbol{Bl_k[t_i]} = \boldsymbol{He_n}\ (\boldsymbol{Bl_k[s_i]})$ # n - order of HF
T1 = {roundingoff ($\boldsymbol{t_i}$, 3)} # upto 3 decimal point
T2 = upto two decimal point of T1 without sign
$\boldsymbol{S_n}$ = Sign bit of T1
d = 3rd decimal digit of T1 # 0 ≤ d ≤ 9
d1 = roundingoff (d/2) # 0 ≤ d1 ≤ 5
d2 = d − d1+1
if $\boldsymbol{bit_1}$ = = 0 and $\boldsymbol{bit_2}$ = = 0
  $\boldsymbol{t_{add}}$ = 00
  $\boldsymbol{t_{fix}}$ = d
else
  $\boldsymbol{t_{add}}$= d2 × {($\boldsymbol{bit_1}$ × 10) + $\boldsymbol{bit_2}$}
  $\boldsymbol{t_{fix}}$ = d1
$\tilde{\boldsymbol{t}}_i = \boldsymbol{S_n} \times \{(T2) + (\boldsymbol{t_{fix}} \times \boldsymbol{10^{-3}}) + (\boldsymbol{t_{add}} \times \boldsymbol{10^{-5}})\}$
$\boldsymbol{Bl'_k[\tilde{s}_i]} = \boldsymbol{He_n^{-1}(Bl_k[\tilde{t}_i])}$
End

3.2 Decryption – Secret Bits Extraction

In Decryption, the stego blocks are received and using hash function, password should be generated. If there is 'n' number of stego blocks, then 'n-1' number of passwords are generated as like as the process in transmission side. The generated passwords are matched with the received password and if they are equal, then there will be no loss or tampering of data. Algorithm 2 outlines the process of extracting secret bits, beginning with the application of Forward HF to each block according to the specified passkey sequence.

Algorithm 2 – Extraction of secret bit in Hermite space

Input : Stego-block $\boldsymbol{Bl'_k[\tilde{s}_i]}$ and Passkey (n)
Output : Final block $'\mathbf{Bl_k}[\hat{s_i}]$ secret bit $\boldsymbol{bit_1}$ and $\boldsymbol{bit_2}$

Start
$\boldsymbol{Bl'_k[\tilde{t}_i]} = \boldsymbol{He_n}\ (\boldsymbol{Bl'_k[\tilde{s}_i]})$ # n is the order of HF
G1 = {roundingoff ($\tilde{\mathbf{t}}_\mathbf{i}$, 5)} # up to 5 decimal point
G2 = up to two decimal point of G1 without sign
$\mathbf{S_n}$ = sign bit of G1
$\mathbf{t_{fix}}$ = 3rd decimal value of G1 # 0 <= p <= 9
f2 = 4th decimal value of G1 # 0 <= p <= 9
f3 = 5th decimal value of G1 # 0 <= p <= 9
if f2 == 0 and f3 == 0
  $\boldsymbol{bit_1}$ = 0, $\boldsymbol{bit_2}$ = 0
else
  if remainder of [{(f2 x 10)+f3}/11]==0
    $\boldsymbol{bit_1}$ = 1, $\boldsymbol{bit_2}$ = 1 and quotient is *qt*
  else if remainder of [{(f2 x 10)+f3}/10]==0
    $\boldsymbol{bit_1}$ = 1, $\boldsymbol{bit_2}$ = 0 and quotient is *qt*

else if remainder of [{(f2 x 10)+f3}/01]==0

$\boldsymbol{bit_1}$ = 0, $\boldsymbol{bit_2}$ = 1 and quotient is *qt*

$\mathbf{t_{fix}} = \mathbf{t_{fix}} + qt - 1$

$\widehat{\boldsymbol{t_\iota}} = \mathbf{S_n}$ x {G2+($\mathbf{t_{fix}}$ x $\mathbf{10^{-3}}$)}

$'\mathbf{Bl_k}[\widehat{s_\iota}] = \mathbf{He_n^{-1}}(\boldsymbol{Bl'_k}[\widehat{\boldsymbol{t_\iota}}])$

End

The 4th and 5th decimal values are crucial for extracting the secret bits bit_1 and bit_2, as well as reconstructing the 3rd decimal value of the original data point t_i. This yields the estimated value $\tilde{t}_i$, where the first three decimal digits remain identical to t_i. Subsequently, reverse HF is applied to compute each data point ($\tilde{s}_i$) in block $'Bl_k[\tilde{s}_i]$ with minimal error. During extraction, addressing stego-data loss is of paramount importance, particularly as there is a dearth of research on this issue. In scenarios where portions of stego-data are lost due to network issues or malicious attacks, such as "removal attacks" by hackers, it poses a significant challenge. However, this proposed steganography solution, which utilizes password mismatch detection to identify missing blocks, the impact is mitigated within the detected range. Beyond these lost blocks, neither the cover blocks nor the secret bytes are affected. To address this challenge for each block, the password is dynamically generated by incorporating the previous block characteristics and the secret bytes. This enables the precise calculation of current or the missing block features and the secret bytes using Hippopotamus Optimization (HO).

3.3 Hippopotamus Optimization

The Hippopotamus Optimization (HO) algorithm, a recent addition to the family of nature-inspired algorithms, represents a novel metaheuristic optimization technique, drawing inspiration from the foraging behavior of hippos. It was developed based on the cooperative and predatory behaviors observed in hippos, HO presents a fresh perspective on tackling optimization challenges across diverse domains. Hippos, renowned for their adaptability and efficiency in both land and water environments, exhibit behaviors that serve as the foundation for the HO algorithm. This includes cooperative foraging, where hippos gather in groups to explore and exploit food sources collectively, leveraging shared intelligence and resources to enhance efficiency. Additionally, hippos display predatory instincts, manifesting in territorial behavior and competitive strategies to secure resources and maintain dominance within their ecosystem. The HO algorithm (refer to Algorithm 3) mirrors these behaviors, with individuals within the population representing potential solutions undergoing iterative updates guided by cooperative foraging and predatory instincts to search for optimal solutions. Initialization involves randomly initializing a population within the problem's search space and evaluating each solution's fitness based on the objective function, providing a measure of its suitability for the optimization task. Throughout the main iterative process, individuals navigate the search space through three distinct phases: the cooperative foraging phase, where individuals explore collectively guided by a dominant leader's position and neighboring group means, emphasizing collaborative exploration and knowledge sharing; the predatory instincts phase, where some individuals adopt competitive strategies to outperform others and exploit advantageous positions, fostering competition and selective pressure within the population; and local exploration, where all individuals engage in perturbing their positions to diversify the search space coverage and prevent premature convergence. The iterative process continues for a predefined number of iterations or until a termination criterion is met, upon which the best solution found during the optimization process is returned as the algorithm's output.

Algorithm 3 - Hippopotamus Optimization (HO)

Inputs :

SearchAgents: Number of search agents (hippos) in the population.

Max_iterations: Maximum number of iterations.

lowerbound: Lower bounds for the search space.

upperbound: Upper bounds for the search space.

dimension: Dimensionality of the problem.

fitness: Objective function to be optimized.

Outputs :

Best_pos: Best solution found.

Best_score: Fitness value of the best solution.

HO_curve: Convergence curve showing the best cost obtained so far over iterations.

Start

Initialization:

Define the levy flight function levy to generate levy flights.

Initialize population X with SearchAgents random solutions within the search space defined by lowerbound and upperbound.

Evaluate the fitness of each solution in X.

Main Loop:

Repeat for each iteration t in [1, Max_iterations]:

a. Record the best solution found so far (Xbest) and its corresponding fitness (fbest).

b. Divide the population into two groups:

- Group 1: Hippos mimicking cooperative behavior.
- Group 2: Predators mimicking predatory behavior.

c. Update positions of solutions in Group 1 based on cooperative behavior:

- Each solution explores using levy flights and interacts with a dominant hippopotamus and nearby group mean.

d. Update positions of solutions in Group 2 based on predatory behavior:

- Each solution imitates a predator, attempting to catch its prey using levy flights and distance-based rules.

e. Apply local exploration to all solutions in the population:

- Perturb the positions of solutions using a randomized exploration strategy.

f. Update fitness values of all solutions.

g. Record the best solution found in this iteration.

h. Print the iteration number and the best cost found so far.

Termination:

Return Best_pos, the corresponding Best_score, and the HO_curve.

End

. The HO algorithm offers several distinctive features and advantages, including its nature-inspired design, encapsulating essential aspects of cooperative and predatory behaviors, providing a biologically inspired approach to optimization; its ability to maintain population diversity through a combination of cooperative exploration and individualistic predatory instincts, facilitating comprehensive exploration of the search space; and its capability to strike a delicate balance between global exploration and local exploitation, enabling efficient convergence towards optimal solutions. In terms of applications, the HO algorithm has demonstrated versatility and effectiveness across various fields, including engineering optimization, where it has successfully tackled complex design optimization problems such as structural optimization, parameter tuning, and system design; data mining and machine learning, where it plays a crucial role in optimizing model parameters, feature selection, and hyperparameter tuning, enhancing performance and generalization capabilities; and biomedical engineering, where it aids in optimizing treatment plans, diagnostic procedures, and medical image analysis, contributing to advancements in healthcare technology and patient care.

Looking ahead, while the HO algorithm holds significant promise, several challenges and opportunities for future research and development exist. These include further refinement of the algorithm's mechanisms, such as leader selection strategies, adaptive parameter settings, and hybridization with other optimization techniques, to enhance effectiveness and scalability; and conducting rigorous empirical studies and real-world applications across diverse domains to validate the algorithm's effectiveness, robustness, and scalability in practical settings. In conclusion, the Hippopotamus Optimization algorithm offers a novel and nature-inspired approach to optimization, drawing inspiration from the cooperative and predatory behaviors of hippos. With its unique features, applications, and potential for future advancements, HO contributes to the growing landscape of metaheuristic optimization techniques, offering innovative solutions to complex optimization problems.

3.4 Su-LSTM for missing stego-block prediction

The LSTM model, as referenced in [22], underwent training with known parameters of block to ensure precise prediction of missing segment's patterns. In the realm of time-series data prediction, Recurrent Neural Networks (RNNs) serve as a prominent tool. RNNs leverage inherent memory stages to retain information about past outputs and current input, facilitating current output prediction. However, classical RNNs encounter limitations such as gradient vanishing and bursting, particularly evident in long-term series prediction tasks. To address this, Long Short-Term Memory (LSTM) networks employ an internal gate mechanism for forecasting future sequences. This process involves four fundamental gates: F (forget), I (input), U (update), and O (output) illustrated in the LSTM block diagram (in Fig. 3.)

Using sigmoid function σ(•), the F-gate is used to evaluate between the current state input (X_k) with the previous state output (Y_{k-1}), determining the extent of information retention and updating weights internally (w_F) with an input bias value (g_F). Similarly, the I-gate adjusts its weights (w_I) with a bias value (g_I), comparing current state input values with previous outputs. The U-gate is used to update bias (b_X) based on input gate information using a hyperbolic tangent operator tanh(•) and, influencing the current cell state (M_k) as $M_k = F \times M_{k-1} + I \times M_k$.

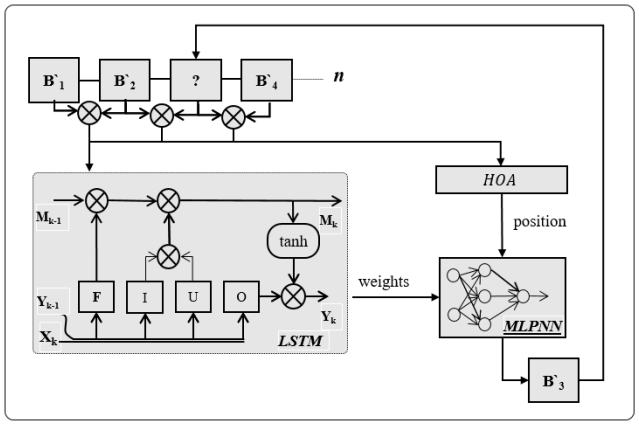

Fig. 3. Su-LSTM Architecture

Finally, by adjusting its weights using a sigmoid function and updating bias (g_O), the O-gate regulates the flow of data to the current output state, leads to the final output state (Y_k) as $Y_k = O \times \tanh(M_k)$. This architectural design of LSTM networks effectively addresses the challenges associated with long-term sequence prediction.

3.5 MLPNN – MultiLayer Perceptron Neural Networks

In modern data-driven approaches, Multilayer Perceptron Neural Networks (MLPNNs) stand as powerful tools for predictive modeling, capable of capturing intricate patterns and relationships within datasets. In the context of predicting missed blocks in a system, MLPNNs offer a sophisticated framework for leveraging optimized features obtained through Hippopotamus Optimization (HO), facilitating proactive measures to mitigate potential block misses. MLPNNs are structured with an input layer, one or more hidden layers, and an output layer. Each layer comprises interconnected neurons, where connections are weighted and activations are applied through nonlinear functions. During training, data flows through the network, and weights are adjusted iteratively using back-propagation and optimization techniques to minimize prediction errors.

The internal architecture of an MLPNN consists of interconnected layers of neurons, each performing specific operations to transform input data into meaningful predictions. The input layer receives input features and passes them on to the hidden layers, where nonlinear transformations are applied using activation functions such as ReLU (Rectified Linear Unit), sigmoid, or tanh (hyperbolic tangent). These transformations enable the network to learn complex patterns and relationships within the data. During

training, data is fed through the network, and predictions are generated through forward propagation. The output predictions are compared to the actual targets, and the resulting error is back-propagated through the network. Through optimization techniques such as gradient descent, the network's weights are adjusted to minimize prediction errors and improve predictive performance. This iterative process continues until convergence is achieved, resulting in a trained MLPNN capable of making accurate predictions.

HO optimizes the positions associated with features relevant to predicting missed blocks. These optimized features serve as input to the MLPNN, where the network learns the complex relationships between these features and the occurrence of missed blocks. By leveraging HO's ability to navigate high-dimensional search spaces and MLPNN's predictive capabilities, the combined framework offers a synergistic approach to enhancing prediction accuracy and efficiency.

To instantiate the MLPNN model, a function is defined, which initializes a sequential model using TensorFlow Keras. 50 neurons and ReLU activation are used in the model, followed by an output layer with a single neuron. Using the Adam optimizer and mean squared error loss function, the model is compiled and optimize it for predictive performance. Optimal performance of the MLPNN is contingent upon hyperparameter tuning, including the number of hidden layers, neurons per layer, learning rate, and regularization techniques. These hyperparameters are fine-tuned through experimentation and validation to achieve optimal predictive accuracy and prevent overfitting. The integration of MLPNN with HO holds promising applications in various domains, including system reliability, fault prediction, and proactive maintenance. By accurately predicting missed blocks based on optimized features, organizations can preemptively address system inefficiencies, minimize downtime, and enhance overall operational efficiency.

3.6 Arrhythmia Detection

Arrhythmia is a disease which is caused due to heart irregularities. If the heart, beats rapidly high or rapidly low, then there will be cause of Arrhythmia. It can be detected by using the reconstructed bio-signal. Arrhythmia poses significant challenges in healthcare worldwide, affecting millions globally with various forms such as ventricular ectopic beats (VEB), supraventricular ectopic beats (SVEB), and fibrillation (F). These irregularities can lead to severe consequences including stroke, heart failure, and sudden cardiac death, underscoring the critical need for accurate diagnosis and treatment. In response, machine learning algorithms offer promising solutions for predicting arrhythmia disease by leveraging intricate patterns within datasets, empowering healthcare professionals to make informed decisions and improve patient outcomes. Take, for example, a 55-year-old male patient presenting symptoms of palpitations, dizziness, and shortness of breath. Upon examination, irregular heart rhythms consistent with arrhythmia were detected through electrocardiography (ECG). Traditionally, diagnosing arrhythmia relies on manual interpretation of ECG signals, a process prone to errors and time-consuming. However, employing machine learning algorithms streamlines this diagnostic process, enabling clinicians to provide timely interventions. The process begins with meticulous data preprocessing, where datasets are loaded, split into features and labels, and prepared for model training by addressing missing values and grouping arrhythmia types. A Decision Tree Classifier is then selected for its simplicity and interpretability, trained on scaled data with hyperparameters tuned for optimal performance. Evaluation reveals promising accuracy in predicting arrhythmia disease, providing valuable insights into diagnosing and treating the condition. Overall, the integration of machine learning algorithms offers a promising avenue for addressing arrhythmia disease, enhancing medical decision-making, and improving patient outcomes, with continued research poised to revolutionize healthcare practices for various medical conditions.

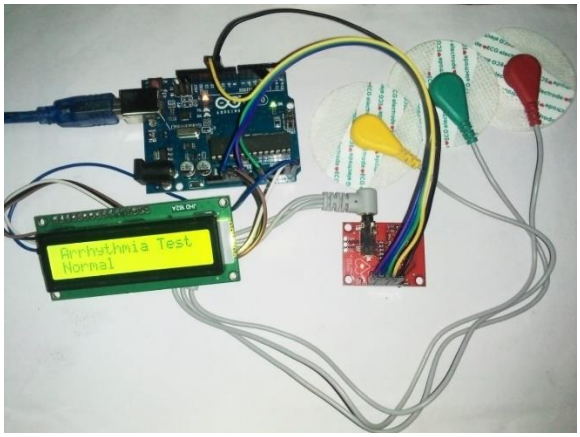

Fig. 4. Arrhythmia test resulted as negative (Normal)

The Fig. 4. depicts that the Arrhythmia test has been done for the patient using ECG signals and resulted as Normal. The Fig. 5. depicts that the Arrhythmia test has been done for another patient which resulted as positive i.e.) Arrhythmia.

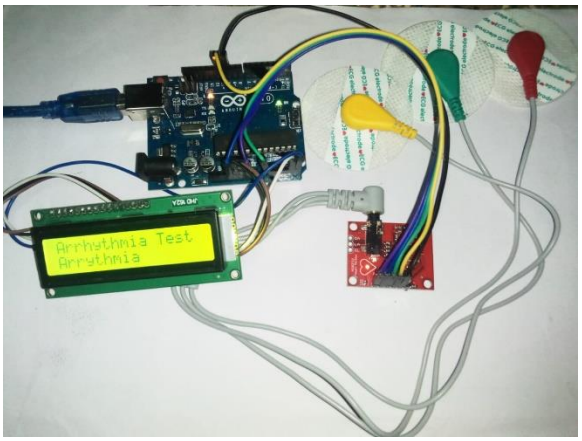

Fig. 5. Arrhythmia test resulted as positive (Arrhythmia)

4. PERFORMANCE AND RESULTS

PRD stands for Percent Root Mean Square Difference. It's a measure used to evaluate the similarity or dissimilarity between two signals or datasets.

The PRD is calculated using the following formula:

$$\text{PRD (P|Q)} = \sqrt{\sum_{i=1}^{L}(P_i\text{-}Q_i)^2 / \sum_{i=1}^{L}(P_i^2)} \times 100\% \quad (1)$$

The PRD gives a percentage measure of how much the two datasets differ on average, relative to the magnitude of the reference dataset. A PRD value of 0 indicates perfect similarity, while higher values indicate greater dissimilarity between the datasets.

Another performance metric called Root Mean Square Error (RMSE). It's a commonly used metric to evaluate the accuracy of a predictive model, particularly in regression analysis. RMSE measures the average magnitude of the errors between predicted values and actual values.

The RMSE is calculated using the following formula:

$$\text{RMSE (P|Q)} = \sqrt{\sum_{i=1}^{L}(P_i - Q_i)^2 / L} \times 100\% \quad (2)$$

It provides a measure of how spread out these differences are, with lower values indicating better predictive accuracy.

PSNR, or Peak Signal-to-Noise Ratio, is a metric commonly used to evaluate the quality of reconstructed or compressed images or video. It measures the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation.

The formula to calculate PSNR is:

$$\text{PSNR (P|Q)} = 20log_{10}\{P_{max}/\,[\text{RMSE}]\} \quad (3)$$

PSNR is expressed in decibels (dB). A higher PSNR value indicates higher accuracy in data, as it signifies a smaller difference between the original and the reconstructed/compressed data. Therefore, PSNR is often used as a quality assessment measure in data, image and video compression algorithms, where maintaining high fidelity of the reconstructed/compressed content is essential.

CORR typically stands for correlation, a statistical measure that describes the strength and direction of a relationship between two variables. It is widely used to understand how changes in one variable are associated with changes in another.

The correlation coefficient, often denoted by r, ranges from -1 to 1.

- A correlation of 1 indicates a perfect positive linear relationship between variables.
- A correlation of -1 indicates a perfect negative linear relationship.
- A correlation of 0 suggests no linear relationship.

The formula to compute the correlation coefficient between two variables X and Y is:

$$\text{CORR (P|Q)} = \frac{\sum_{i=1}^{L}(P_i - P_m)(Q_i - Q_m)}{\sum_{i=1}^{L}(P_i - P_m)^2 \sum_{i=1}^{L}(Q_i - Q_m)^2} \quad (4)$$

Correlation is a fundamental tool in data analysis and is used to assess the strength and direction of relationships between variables, aiding in making predictions and drawing conclusions in various fields.

The letter P denotes the Original signal, Q is the Erroneous signal, L denotes the length of signal P and Q. P_i, Q_i denotes the i^{th} data point in P and Q, P_m is the mean of signal P, Q_m is the mean of signal Q and P_{max} is the maximum value of signal.

4.1 ECG Signal Experimental results

These results indicate the comparison of PRD, RMSE, and PSNR between the stego-original data and the reconstructed-original data for different numbers of secret bits (1000, 2000, and 3000) of ECG signal as shown in table 3.

	Error between stego and Original data			Error between reconstructed and original data		
No. of Secret bits	**PRD**	**RMSE**	**PSNR**	**PRD**	**RMSE**	**PSNR**
1000	0.00002	0.0005	113.95	0.000498	0.00013	114.168
2000	0.00002	0.0005	113.85	0.00014	0.00050	114.08
3000	0.00002	0.0005	113.60	0.000145	0.00051	113.87

Table.3. PRD, RMSE, PSNR values for ECG Signal

Overall, the PRD values for both stego-original and reconstructed-original data remain consistently low, indicating minimal changes during the embedding and reconstruction processes. The RMSE values are also relatively small, suggesting close resemblance between the stego/reconstructed data and the original data. Additionally, the PSNR values are high, indicating high quality and fidelity of the stego/reconstructed data compared to the original data. These results suggest that both the steganographic embedding and reconstruction processes are effective in preserving the original data's integrity and quality across different numbers of secret bits.

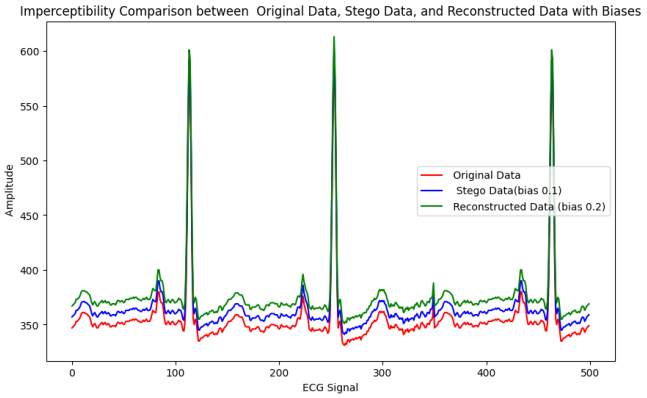

Fig.6 Imperceptibility comparison – No tampering

The graph in Fig. 6 shows the original data in blue, the stego-data in green, and the reconstructed data in red. The fact that the reconstructed data closely resembles the original data suggests that the steganography technique is successful.

When receiving tampered or missing data (shown in Fig. 7), the system demonstrates a robust capability to predict and reconstruct the original signal accurately, closely resembling its authentic form. This successful reconstruction underscores

the effectiveness of the implemented techniques, particularly in scenarios where data integrity is compromised due to transmission errors or deliberate tampering.

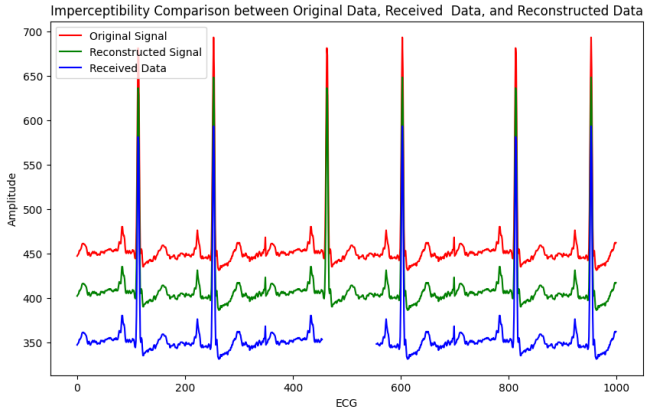

Fig.7. Imperceptibility comparison – Tampering

4.2 EEG Signal Experimental results

	Error between stego and Original data			Error between reconstructed and original data		
No. of Secret bits	**PRD**	**RMSE**	**PSNR**	**PRD**	**RMSE**	**PSNR**
1000	0.000003	0.00076	110.50	0.000031	0.0007	110.97
2000	0.000003	0.0007	110.44	0.000031	0.00072	110.97
3000	0.000003	0.00077	110.29	0.000032	0.00073	110.82

Table 4. PRD, RMSE, PSNR values for EEG Signal

The results (shown in table 4) suggest that the steganographic embedding and reconstruction processes maintain the integrity and quality of the original EEG signal data across different numbers of secret bits. The PRD values indicate minimal changes during both processes, while the RMSE values reflect a close resemblance between the stego/reconstructed data and the original data. Additionally, the high PSNR values indicate high quality and fidelity of the stego/reconstructed data compared to the original data. Overall, these results demonstrate the effectiveness of the steganographic techniques for preserving EEG signal data integrity.

4.3 PPG Signal Experimental results

	Error between stego and Original data			Error between reconstructed and original data		
No. of Secret bits	**PRD**	**RMSE**	**PSNR**	**PRD**	**RMSE**	**PSNR**
1000	0.000002	0.00075	110.52	0.00002	0.00071	111.038
2000	0.000002	0.00076	110.47	0.00002	0.00072	110.93
3000	0.000002	0.00074	110.73	0.00001	0.00069	111.234

Table 5. PRD, RMSE, PSNR values for PPG Signal

The results (shown in table 5) suggest that the steganographic embedding and reconstruction processes maintain the integrity and quality of the original PPG signal data across different numbers of secret bits. The PRD values indicate minimal changes during both processes, while the RMSE values reflect a close resemblance between the stego/reconstructed data and the original data. Additionally, the high PSNR values indicate high quality and fidelity of the stego/reconstructed data compared to the original data. Overall, these results demonstrate the effectiveness of the steganographic techniques for preserving PPG signal data integrity.

4.4 ARRHYTHMIA TRAINING MODEL RESULT

Model	Accuracy
RandomForestClassifier	0.99664
GradientBoostingClassifie	0.99236
DecisionTreeClassifier	0.99125
Support Vector Machine Classifier	0.98864
KNeighborsClassifier	0.99441
MLPClassifier	0.99460

Table.4 Various model accuracy of arrhythmia detection

Machine learning models play a pivotal role across diverse domains, serving as powerful tools for predictive analytics and pattern recognition. In a recent study, a comprehensive evaluation of several machine learning algorithms was conducted using a dataset to assess their predictive prowess. The findings unveiled remarkable accuracies achieved by each algorithm: the Random Forest Classifier excelled with an impressive accuracy of 99.664%, while the Gradient Boosting Classifier demonstrated robust predictive power at 99.236%. Additionally, the Decision Tree Classifier showcased strong performance with an accuracy of 99.125%, and the Support Vector Machine (SVM) Classifier exhibited high accuracy, reaching 98.864%. Furthermore, the k-Nearest Neighbours (KNN) Classifier delivered excellent results with an accuracy of 99.441%, while the Multilayer Perceptron (MLP) outperformed, achieving a remarkable accuracy of 99.460%. These results underscore the efficacy of machine learning algorithms in accurately capturing intricate patterns and making reliable predictions on the dataset. Such high accuracies highlight the potential utility of these models across real-world applications, spanning from medical diagnosis to financial forecasting. Moreover, these findings offer valuable insights for practitioners and researchers, aiding in the selection of the most suitable algorithm for specific tasks based on their performance metrics.

5. CONCLUSION AND FUTURE WORK

In this research, we represents a significant advancement in the field of secure data transmission, particularly concerning ECG biosignals. By leveraging deep learning methodologies, the system offers robust encryption and decryption processes, ensuring the confidentiality and integrity of sensitive information embedded within the biosignals. The utilization of the Hermite function in both encryption and decryption phases demonstrates the efficacy of mathematical transformations in steganographic techniques. Through the intricate interplay of forward and reverse Hermite functions, along with secret bit insertion and extraction, the system achieves a high level of security while preserving the original characteristics of the biosignals. Moreover, the integration of hash functions for password generation gives an additional layer of security, enhancing the resilience of the encryption-decryption process against unauthorized access. The system's capability to recover missing data blocks through predictive modeling using

MLPNN showcases its adaptability and reliability in handling data transmission errors or packet losses. While the PSO algorithm initially shows promise in optimizing missing data block positions, the superior performance of the Hippopotamus optimization algorithm underscores the importance of exploring and benchmarking different optimization techniques in future research endeavors.

Looking ahead, there is ample room for further innovation and improvement in several areas. Future work could involve investigating more sophisticated encryption techniques to enhance data security further. Additionally, conducting comprehensive evaluations and comparisons of various optimization algorithms, including PSO, HOA, and potentially others, would provide valuable insights into their strengths and limitations in the context of missing data recovery. Furthermore, exploring advanced deep learning architectures and techniques, such as recurrent neural networks or generative adversarial networks, could potentially enhance the system's performance and capabilities. Validating the system in real-world healthcare environments and integrating it with existing biomedical systems would be crucial steps towards practical implementation and deployment. Collaborations with healthcare professionals and institutions could provide valuable feedback and insights for refining the system's design and functionality to better meet the needs and requirements of medical practitioners and patients. Ultimately, by addressing these avenues of research and development, the project aims to make significant contributions to the advancement of secure data transmission in healthcare and biomedical applications, thereby improving patient privacy, data integrity, and overall healthcare quality.

6. REFERENCES

[1] S. Rahman, J. Uddin, H. U. Khan, H. Hussain, A. A. Khan and M. Zakarya, "A Novel Steganography Technique for Digital Images Using the Least Significant Bit Substitution Method," in *IEEE Access*, vol. 10, pp. 124053-124075, 2022, doi: 10.1109/ACCESS.2022.3224745.

[2] A. Sengupta and M. Rathor, "Structural Obfuscation and Crypto-Steganography-Based Secured JPEG Compression Hardware for Medical Imaging Systems," in *IEEE Access*, vol. 8, pp. 6543-6565, 2020, doi: 10.1109/ACCESS.2019.2963711.

[3] T. Ahmed Alhaj *et al*., "A Survey: To Govern, Protect, and Detect Security Principles on Internet of Medical Things (IoMT)," in *IEEE Access*, vol. 10, pp. 124777-124791, 2022, doi: 10.1109/ACCESS.2022.3225038.

[4] G. F. Siddiqui *et al*., "A Dynamic Three-Bit Image Steganography Algorithm for Medical and e-Healthcare Systems," in *IEEE Access*, vol. 8, pp. 181893-181903, 2020, doi: 10.1109/ACCESS.2020.3028315.

[5] M. Elhoseny, G. Ramírez-González, O. M. Abu-Elnasr, S. A. Shawkat, N. Arunkumar and A. Farouk, "Secure Medical Data Transmission Model for IoT-Based Healthcare Systems," in *IEEE Access*, vol. 6, pp. 20596-20608, 2018, doi: 10.1109/ACCESS.2018.2817615.

[6] J. Malik, O. C. Devecioglu, S. Kiranyaz, T. Ince and M. Gabbouj, "Real-Time Patient-Specific ECG Classification by 1D Self-Operational Neural Networks," in *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 5, pp. 1788-1801, May 2022, doi: 10.1109/TBME.2021.3135622.

[7] N. Vemishetty *et al*., "Low Power Personalized ECG Based System Design Methodology for Remote Cardiac Health Monitoring," in *IEEE Access*, vol. 4, pp. 8407-8417, 2016, doi: 10.1109/ACCESS.2016.2629486.

[8] C. Böck, P. Kovács, P. Laguna, J. Meier and M. Huemer, "ECG Beat Representation and Delineation by Means of Variable Projection," in *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 10, pp. 2997-3008, Oct. 2021, doi: 10.1109/TBME.2021.3058781.

[9] G. N. K. Reddy, M. S. Manikandan and N. V. L. N. Murty, "Evaluation of Objective Distortion Measures for Automatic Quality Assessment of Processed PPG Signals for Real-Time Health Monitoring Devices," in *IEEE Access*, vol. 10, pp. 15707-15745, 2022, doi: 10.1109/ACCESS.2022.3148256.

[10] M. H. Amiri et al., "Hippopotamus optimization algorithm: a novel nature-inspired optimization algorithm," ResearchGate, Feb. 2024. doi: 10.1038/s41598-024-54910-3.

[11] S. Khatun, R. Mahajan and B. I. Morshed, "Comparative Study of Wavelet-Based Unsupervised Ocular Artifact Removal Techniques for Single-Channel EEG Data," in *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 4, pp. 1-8, 2016, Art no. 2000108, doi: 10.1109/JTEHM.2016.2544298

[12] L. Ren, T. Wang, Y. Laili and L. Zhang, "A Data-Driven Self-Supervised LSTM-DeepFM Model for Industrial Soft Sensor," in *IEEE Transactions on Industrial Informatics*, vol. 18, no. 9, pp. 5859-5869, Sept. 2022, doi: 10.1109/TII.2021.3131471.

[13] S. Nakatani, K. Yamamoto and T. Ohtsuki, "Fetal Arrhythmia Detection based on Deep Learning using Fetal ECG Signals," *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, Rio de Janeiro, Brazil, 2022, pp. 2266-2271, doi: 10.1109/GLOBECOM48099.2022.10001697.

[14] H. K. Kim and M. H. Sunwoo, "An Automated Cardiac Arrhythmia Classification Network for 45 Arrhythmia Classes Using 12-Lead Electrocardiogram," in *IEEE Access*, vol. 12, pp. 44527-44538, 2024, doi: 10.1109/ACCESS.2024.3380892.

[15] M. Singha Roy, B. Roy, R. Gupta and K. Das Sharma, "On-Device Reliability Assessment and Prediction of Missing Photoplethysmographic Data Using Deep Neural Networks," in *IEEE Transactions on Biomedical Circuits and Systems*, vol. 14, no. 6, pp. 1323-1332, Dec. 2020, doi: 10.1109/TBCAS.2020.3028935.

[16] R. Thabit, N. I. Udzir, S. M. Yasin, A. Asmawi and A. A. -A. Gutub, "CSNTSteg: Color Spacing Normalization Text Steganography Model to Improve Capacity and Invisibility of Hidden Data," in *IEEE Access*, vol. 10, pp. 65439-65458, 2022, doi: 10.1109/ACCESS.2022.3182712.

[17] A. Nakashima, R. Ueno and N. Homma, "AES S-Box Hardware With Efficiency Improvement Based on Linear Mapping Optimization," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 10, pp. 3978-3982, Oct. 2022, doi: 10.1109/TCSII.2022.3185632.

[18] J. T. L. Philjon and N. V. Rao, "Metamorphic cryptography — A paradox between cryptography and steganography using dynamic encryption," *2011 International Conference on Recent Trends in Information*

Technology (ICRTIT), Chennai, India, 2011, pp. 217-222, doi: 10.1109/ICRTIT.2011.5972272.
[19] M. S. Rahman, I. Khalil and X. Yi, "Reversible Biosignal Steganography Approach for Authenticating Biosignals Using Extended Binary Golay Code," in *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 1, pp. 35-46, Jan. 2021, doi: 10.1109/JBHI.2020.2988449.
[20] Edward Jero S, Ramu P, Ramakrishnan S. Discrete wavelet transform and singular value decomposition based ECG steganography for secured patient information transmission. J Med Syst. 2014 Oct;38(10):132. doi: 10.1007/s10916-014-0132-z. Epub 2014 Sep 4. PMID: 25187409.
[21] S. E. Jero and P. Ramu, "A robust ECG steganography method," *2016 10th International Symposium on Medical Information and Communication Technology (ISMICT)*, Worcester, MA, USA, 2016, pp. 1-3, doi: 10.1109/ISMICT.2016.7498893.
[22] S. Banerjee and G. K. Singh, "A Robust Bio-Signal Steganography With Lost-Data Recovery Architecture Using Deep Learning," in *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-10, 2022, Art no. 4007410, doi: 10.1109/TIM.2022.3197781.