IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 July 2024, accepted 13 August 2024, date of publication 21 August 2024, date of current version 30 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3447075

RESEARCH ARTICLE

Data Security Utilizing a Memristive Coupled Neural Network in 3D Models

MOHAMED GABR[1], (Member, IEEE), AMR DIAB[1], HUWAIDA T. ELSHOUSH[2], (Senior Member, IEEE), YEN-LIN CHEN[3], (Senior Member, IEEE), LIP YEE POR[4], (Senior Member, IEEE), CHIN SOON KU[5], AND WASSIM ALEXAN[6,7], (Senior Member, IEEE)

[1]Computer Science Department, Faculty of Media Engineering and Technology, German University in Cairo (GUC), New Cairo 11835, Egypt
[2]Department of Computer Science, Faculty of Mathematical Sciences and Informatics, University of Khartoum, Khartoum 11115, Sudan
[3]Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei 106344, Taiwan
[4]Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia
[5]Department of Computer Science, Universiti Tunku Abdul Rahman, Kampar 31900, Malaysia
[6]Communications Department, Faculty of Information Engineering and Technology, German University in Cairo (GUC), New Cairo 11835, Egypt
[7]Mathematics Department, German International University (GIU), New Administrative Capital, Cairo 13507, Egypt

Corresponding authors: Yen-Lin Chen (ylchen@mail.ntut.edu.tw), Lip Yee Por (porlip@um.edu.my), and Chin Soon Ku (kucs@utar.edu.my)

This work was supported by the National Science and Technology Council, Taiwan, under Grant NSTC-112-2622-8-027-008 and Grant NSTC-113-2224-E-027-001.

ABSTRACT This article proposes a novel double data security algorithm that first encrypts sensitive data using a two-stage encryption method based on numerical solutions from a fractional-order memristive coupled neural network system. Solutions are obtained to generate encryption keys and construct S-boxes, which are then applied along with an initial key to encrypt the data bits through repeated XOR and S-box operations. The encrypted output is then hidden imperceptibly within 3D geometries by slightly modifying model points based on the encrypted data bits. This two-pronged approach provides enhanced protection for confidential information compared to single encryption or data hiding alone. Numerical experiments demonstrate the effectiveness of encryption in obscuring patterns while data extraction from modified 3D models validates recovery with negligible visual impact. Additionally, the proposed encryption scheme is shown to be superior to the standard AES-256 algorithm in terms of both computational efficiency and security against brute-force attacks. Through a synergistic blend of robust encryption and stealthy data hiding within 3D objects, the presented algorithm can reliably ensure privacy for sensitive digital data transmissions and storage applications.

INDEX TERMS 3D models, chaos theory, data hiding, encryption, memristive coupled neural network.

I. INTRODUCTION

As digital technology advances and becomes increasingly integrated into our daily lives, the need to protect sensitive data that traverses unsecured open networks has never been more crucial. Information security, a discipline dedicated to preventing unauthorized access, disclosure, disruption, modification, or destruction of information, has become a significant area of concern for individuals, businesses, and governments worldwide. The vast expanse of open networks poses a considerable risk to data confidentiality, integrity, and availability, particularly when transmitting sensitive or confidential information [1]. This challenge provides motivation for the continuous exploration and development of innovative data security solutions [2], [3], such as the novel double data security algorithm proposed in this manuscript, which not only encrypts sensitive data but also hides it within 3D geometries for enhanced protection.

Although traditional data security techniques, such as the Data Encryption Standard (DES) [4], the Advanced Encryption Standard (AES) [5], and their variants [6], have been widely used to secure sensitive information, their singular use may no longer provide adequate protection in the face of evolving cyber threats. The DES algorithm, for example, has been largely phased out due to its susceptibility to brute-force attacks, owing to its relatively short key

The associate editor coordinating the review of this manuscript and approving it for publication was Yassine Maleh.

length [7]. Similarly, while AES remains a robust encryption standard, it is not impervious to sophisticated attacks such as side-channel attacks and related-key attacks [8]. Furthermore, the use of cryptography alone, while effective in obscuring data, does not conceal the existence of communication, which can draw unwanted attention from potential attackers [9]. On the other hand, steganography, which involves hiding data within other non-sensitive data, offers the advantage of stealth but generally does not provide robust encryption [10]. Thus, the inherent limitations of steganography and cryptography, when used in isolation, illustrate the need for a more comprehensive approach to data security [11].

Chaotic and hyperchaotic systems, with their inherent unpredictability and sensitivity to initial conditions, have emerged as valuable tools in the realm of information security [12]. Hyperchaotic systems, which are an extension of regular chaotic systems but with an added dimension of complexity and randomness, offer an advanced level of security, making them even more effective for encryption purposes. The importance of the effectiveness of these systems lies in their potential to generate high-quality pseudo-random number sequences through the numerical solutions of fractional-order hyperchaotic functions [13].

Pseudo-random number generators (PRNGs) are algorithms that produce sequences of numbers that resemble true randomness [14]. These are crucial in encryption, generating unpredictable results that are essential for secure data encryption [15]. They are employed in information security to forge strong encryption keys and to develop S-boxes, critical to symmetric key encryption algorithms. PRNGs improve key unpredictability and S-box complexity, thus improving data security [16]. Incorporating fractional-order hyperchaotic functions into encryption protocols represents a significant leap in safeguarding sensitive data in the digital era.

The proposed security scheme introduces a novel double-data security algorithm that leverages the benefits of both encryption and data hiding. The approach begins by encrypting sensitive data using a two-stage encryption method. This technique relies on numerical solutions obtained from a fractional-order memristive coupled neural network system. These solutions serve dual purposes: they generate encryption keys and aid in the construction of S-boxes. The data bits are then encrypted using these keys and S-boxes through multiple rounds. After encryption, the output is cleverly hidden within 3D geometries. This is achieved by subtly modifying the model points of the 3D geometries based on the encrypted data bits. This two-pronged approach to encryption and data hiding offers an enhanced level of data security, providing a more robust safeguard against unauthorized access and data breaches.

The specific contributions of this work are multifold:

1) The proposed security scheme introduces a unique blend of robust encryption and stealthy data hiding strategies to ensure the privacy of sensitive digital data during transmission and storage.
2) It demonstrates the effectiveness of the encryption in obscuring patterns while also validating that data can be extracted from the modified 3D models with negligible visual impact.
3) It is shown to outperform the AES-256 algorithm in terms of security against attacks and computational efficiency.
4) Upon testing the security of its encryption strength, it is shown to successfully pass all NIST SP $800-22$ tests, as well as possessing a vast key space of 2^{2445}.
5) It is shown to possess a high embedding capacity of up to 63 bits per vertex of a 3D model.

This article is organized as follows: Section II conducts a literature review. Some preliminary mathematical constructs are introduced in Section III. Section IV describes the proposed data security scheme, while the performance evaluation and discussion are provided in Section V. Section VI discusses the proposed algorithm. Finally, Section VII draws conclusions and suggests a future perspective.

II. RELATED LITERATURE

The field of data hiding in 3D models is an emerging research area with increasing relevance in today's ever-increasing reliance on digital communications and the Internet. In the existing literature, numerous techniques and methodologies have been explored and developed to embed information within 3D models without significantly altering their visual appearance or structure. These works focus on various aspects, including robustness against common attacks, imperceptibility, capacity, and computational efficiency. Furthermore, the literature permeates through applications in diverse domains, such as digital rights management, tamper detection, and virtual reality experiences [17]. This section aims to provide an overview of the early, important, and contemporary work that has shaped the field, highlighting the key methodologies, findings, and trends that have informed the current state of research in data hiding within 3D models.

Typically, 3D models are exemplified in one of three ways, namely point clouds [18], [19], voxels [20], [21], and polygon meshes [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], the meshes being desired for their large embedding capacity [25]. Concerning 3D point clouds, the researchers in [18] subdivide the 3D point cloud model into patches. They then classify these patches using a self-similarity position matching procedure to construct a code book. Their method realized high capacity. Similarly, the authors of [19] embed data in 3D images using a point-cloud data scheme. A bracket function is implemented on the x coordinates derived from points in an image. The secret message is first symbolized in ASCII format and then added to the results of the bracket function, forming the x-coordinate of the stego image. The irregularity of the data format of point clouds necessitates the voxelization of raw point clouds into voxels. Moreover, point-level features cause high computation overhead because of the disordered storage. Conversely, the voxel-based form is suitable for feature extraction but produces a lesser level of accuracy as the data is split into grids. Thus, the researchers in [20] came up with a 3D object classification system utilizing a broad learning system (BLS) in addition to a feature extractor called VB-Net. They first converted the non-uniform point clouds of the 3D image into regular voxels, which are then handled by the feature extractor. Finally, the BLS utilizes the extracted features for object classification. However, researchers in [21] suggest that the voxel granularity can also provide enough high performance instead of the exact positioning of the raw points. Hence, their method offers a good balance between accuracy and efficiency.

Triangular meshes are a widespread delineation of 3D models that provide many steganographic choices of modifications to their basic characteristics in terms of vertices (geometry) and connectivity (topology) [32]. Consequently, various researchers such as [22], [23], [24], [25], [32], [33], [41], [42], [43], and [44], utilized 3D mesh models to achieve high capacity while maintaining low embedding distortion. In particular, the authors in [22] make effective use of a truncated space of data with a shifting approach to yield a high-capacity steganography algorithm while maintaining the alteration within an adaptable lower limit. Actually, the embedding distortion in their approach does not increase with the size of the secret data within the maximum embedding capacity. However, their approach has the major drawback of failing to extract the embedding information due to the use of a Principal Components Analysis (PCA) to align the model. On the other hand, the authors of [23] proposed a new multi-layered embedding method that provides up to $3n_{layers}$ bits/vertex, where the n_{layers} ranged from 7 to 13. Albeit attaining a high capacity while keeping the distortion low, their method is not robust against malicious attacks such as smoothing, the addition of noise, non-uniform scaling, simplification, and vertice re-sampling. Thus, it is not suited for digital content protection and authentication applications. A different approach that also utilizes a multi-level embedding procedure to boost the hiding capacity to 3 bits/vertex is that in [33]. In that approach, a substitutive blind procedure is applied. The authors enhance the triangular traversal by utilizing an advanced jump strategy. Their main drawback relates to machine precision errors, especially with small triangles. Another steganography approach based on a substitutive procedure is presented in [32]. That approach strives against thwarting attacks such as similarity transformations and vertex reordering. Furthermore, it suffers from inefficient triangle traversal. Another multi-layer mesh synchronous RDH method in the spatial domain addressing mesh asynchrony with minimum distortion analysis is suggested in [42]. Sensitive data is embedded in redundant regions based on smoothness sorting. Additionally, they designed a logical mapping strategy and a multilevel bimodal mapping rule to enhance the model's embedding capacity. They claim a high embedding capacity of up to 3 bits per vertex (bpv) in each layer and an SNR close to 92 dB. Moreover, it is difficult to detect geometric distortions in their model. Working differently to embed the secret data within a 3D image effectively, the authors of [44] investigated a shifting strategy before the vertex component interval construction approach. They utilized a truncated space to reduce distortion and enhance imperceptibility while preserving a high embedding capacity. Another method for achieving high security and capacity while maintaining very low distortion is presented by [41]. The authors make use of a gray code to decide on the vertices of 3D models in which embedding of the sensitive date would be carried out. Additionally, their method resists attacks such as noise, filtering, and vertex reordering. A different method was introduced by the researchers in [43], who employed a Parallel Breadth First Search (PBFS) with hyper-objects. By leveraging PBFS and layer synchronization, they embed private information in the vertices of 3D mesh images. To optimize efficiency and reduce time, cost, and complexity, they parallelize BFS using a bag data structure, which is based on the pennant data structure. Their method has a high embedding capacity while maintaining less execution time, cost, and intricacy. Nevertheless, their method lacks proof of attack resistance.

From another point of view, working adaptively is efficacious for improving steganographic security. It is noteworthy to mention that 3D steganographers, as in [24], [25], [27], [27], [29], [30], [31], and [45], opt for working with meshes, as those offer a high embedding capacity besides resistance to steganalysis. Hence, bearing these views in mind, the researchers in [24] proposed a secure 3D blind data hiding technique based on a mesh traversal order that depends on the shortest distances between neighboring vertices. They hide the secret data in the 4th and 5th significant decimal places of the vertices. In addition to achieving high capacity and imperceptibility, their scheme is resistant to geometrical attacks, such as rotation and translation attacks, as well as various types of noise attacks, such as local variance noise, Poisson distribution noise, and salt and pepper noise attacks. Furthermore, their proposed algorithm can withstand vertex reordering attacks on a mesh as it changes the reference index of vertices. Moreover, the scheme proposed in [24] is adaptive in the sense that it distinguishes between smooth and noisy surfaces of meshes in the course of the embedding phase. Another adaptive 3D mesh steganography was suggested by [25]. Focusing on security, the authors came up with a highly adaptive 3D mesh steganography scheme using feature-preserving distortion (FPD) to gauge the distortion. They mapped the vertex coordinates into integers and built bit-planes from these to form their embedding domain. However, their scheme possesses inadequate robustness against geometric attacks. Similarly, the authors of [27] suggested an adaptive steganography method for 3D meshes. They acted on the binarized bit-streams of vertices of meshes, attaining acceptable outcomes considering steganalysis. They adaptively embed data in the highest bit planes, yielding a high capacity. Yet, their method necessitates LSB replacement (LSBR) non-adaptive embedding in the others, which impeds their method's security to some extent. Another work presenting a blind high capacity utilizing a pattern-based 3D mesh geometric model is from the researchers of [29]. They re-triangulate a subdivision of a triangle mesh, then embed their secret data into its recently attached location. They claim embedding up to 9 bits of secret data into the vertices of a triangle mesh with unnoticeable distortion and also in the image's geometric properties. They attained high capacity and imperceptibility. They also achieved resistance to attacks such as cropping, rotation, scaling, translation, noise addition, and filtering mechanisms. Also working adaptively, the work in [30] proposed a blind steganographic algorithm for 3D polygonal models. The authors chose a vertex decimation process to determine its referencing neighbors in order to improve the accuracy of complexity estimation. Hence, this approach embeds variable amounts of secret data, depending on the surface features of each vertex. This allows for imperceptibility conservation very well. However, the time complexity of his research is comparatively high. On the other hand, the researchers in [27] employ syndrome trellis codes in an adaptive manner to embed sensitive messages in the vertices of 3D models. They work in an adaptive manner, counting on steganalytic features, specifically vertex variation. They manipulate the vertices to embed sensitive data, considering the intricacy of local regions. Although their method resists steganalysis tools, it is rather complex. In [45], the authors propose an adaptive vertex grouping strategy for dividing vertices in a 3D model into groups. Multi-MSB prediction and Huffman coding are used to compress vertex data, enhancing the RDHEM's embedding capacity. Additionally, two 3D model encryption schemes are presented: one using secret sharing over the Galois field and the other employing stream cipher techniques.

Approaching from a different angle, the researchers of [45] and several others, such as those cited in [26], [28], [36], [37], [38], [39], and [40], have combined steganography with encryption to offer additional layers of security. For example, the authors of [28] proposed a reversible data hiding technique in encrypted 2D meshes. They first convert decimals of vertex coordinates into integers, where they apply a bit-stream encryption technique. The encrypted data is then embedded using the Least Significant Bit (LSB) and is later perfectly extracted to retrieve the mesh contents. However, their method has a limitation: an average error rate of 4.22%, which could potentially be reduced by using error correction codes (ECCs) at the expense of a lower hiding capacity. Similarly, the researchers in [26] utilize the Blowfish or AES-128 algorithms to encrypt confidential data before hiding it. They employ a Gray code sequence that differentiates the arrangement of the vertices in 2D objects. In addition to achieving high capacity and high imperceptibility, their method can resist geometric similarity attacks such as reflection, uniform scaling, rotation, and translation, although it only partially withstands smoothing. Furthermore, the authors of [36] integrated cryptography with steganography using the Wolfram Mathematica® language to offer AES-secured bit-cycling steganography in sliced 2D images. Two steganographic techniques, adopting arithmetic and geometric sequences, were utilized to LSB-embed the secret data. The researchers in [37] employed a similar approach, securing message embedding in 2D images using multiple layers of AES-128 encryption combined with a repetition code, LSB steganography in 2D images, and a final noise layer. Similarly, the work in [38] proposed a high-capacity and highly secure 2D image steganography technique using two layers of AES-128 encryption and LSB steganography. The authors used four mathematical sequences (arithmetic, geometric, Fibonacci, and Gray codes) to determine the number and sequence of slices to embed the secret message. Similarly, the researchers in [39] proposed a two-layer 2D image message security scheme that uses Blowfish to encrypt the secret message and LSB, in conjunction with arithmetic and Fibonacci sequences, to select the slices for embedding the secret message. Finally, the work in [40] introduced a two-layer message security approach in 2D images, where AES-256 is employed to encrypt the secret message. This is followed by edge detection after segmenting a 2D image into several 2D segments. The LSB of the edge pixels' locations is then utilized to embed secret data into the 2D segments.

From another perspective, several researchers, including those cited in [31], [34], and [35], have utilized various chaotic maps to create a secure and high-capacity steganography technique. Specifically, the researchers in [31] proposed a blind adaptive reversible data hiding method that generates a unique mesh traversal algorithm for each 3D mesh model. This method visits the nearest neighbor employing a breadth-first search algorithm. The difference between the vertices is adjusted before hiding the secret data within them. They utilized a chaotic logistic map to select the coordinates in which the secret data is embedded, thereby randomizing the embedding pattern. They claim to be the first researchers to use a chaotic map in 3D image steganography. Their models resist attacks such as vertex re-ordering, rotation, scaling, and translation. However, their embedding capacity could be improved. In a similar manner, some researchers, namely those of [34] and [35], combined cryptography and steganography while also employing a chaotic map. Specifically, the work in [34] encrypted the secret data through AES-128 before XORing it with a key generated from a chaotic 2D map. During the hiding phase, the 3D image is first segmented into multiple 2D images, then shuffled using a Sine Logistic Map. The embedding is carried out through LSBs. Finally, the segments are reassembled to construct the 3D stego image. Similarly to the work in [34], the authors of [35] integrated cryptography with steganography, resulting in a multilayer message security scheme. Their algorithm consists of three layers: cryptography, edge detection, and steganography. A chaotic function is used to encrypt the secret data. The selection of edge-pixel positions for embedding is performed using Canny edge detection. Lastly, steganography is performed through the LSB in only the edge pixels.

Despite the significant contributions to the field of data hiding in 3D mesh models presented in this section and summarized in Table 1, we believe there are still some gaps in the existing literature. Although various approaches have achieved high capacity and maintained imperceptibility and distortion, few have addressed adaptability, integrated encryption with steganography, or made use of chaotic maps or hyperchaotic systems. For instance, the researchers in [31] employed the chaotic logistic map to select coordinates for random embedding. However, when researchers like those of [34] utilize chaotic maps and combine cryptography with steganography, they employ AES-128, which is arguably insecure in the face of increasing cyber security threats. Moreover, their method, along with that of the author of [35], requires further verification and testing with respect to the resistance to attacks. The lack of a robust encryption layer is a major issue with existing techniques. It is very clear that steganography combined with weak encryption schemes represents a major challenge in current 3D data hiding approaches. Therefore, achieving high capacity while maintaining high security and imperceptibility and being robust to different attacks and noise still needs further investigation.

To overcome the aforementioned limitations in existing methods available in the literature, we propose a novel approach that first encrypts confidential data using a two-step method. We use numerical solutions derived from a fractional-order memristive coupled neural network system to generate encryption keys and assist in the construction and application of the S-box before finally embedding the data in 3D geometries. Therefore, the motivation of this work is to introduce a multilevel encryption and steganography approach that enhances security against attacks while also providing high computational efficiency, high capacity, imperceptibility, and reduced distortion.

III. PRELIMINARY MATHEMATICAL CONSTRUCTS

A. MEMRISTIVE COUPLED NEURAL NETWORK MODEL

Lin et al. [46] established the Memristive Coupled Neural Network Model (MCNNM) using sub-neural networks constructed using the Hopfield Neural Network (HNN) and a memristive model based on the flux-controlled memristor. The Hopfield neural network with brain-like chaos is utilized to emulate the chaotic behaviors of the brain nervous system. Its mathematical equation is given by:

$$C_i\dot{v}_i = -\frac{v_i}{R_i} + \sum w_{ij}tanh(v_i) + I_i \qquad (i, j \in N^*), \qquad (1)$$

where C_i, R_i, and v_i are respectively capacitance, resistance, and potential of the cell membrane in neuron i; w_{ij} is the synaptic weight coefficient describing the connection strength from neuron j to neuron i; the hyperbolic tan function represents the neuron stimulation function, and I_i denotes an external input current. It is noteworthy that the HNN chaotic dynamics rely on w_{ij}. Thus, two distinct sub-neural networks with four neurons can be built depending on the indigenous HNN in (1) and deciding on a suitable synaptic weight coefficients by selecting the trial and error method.

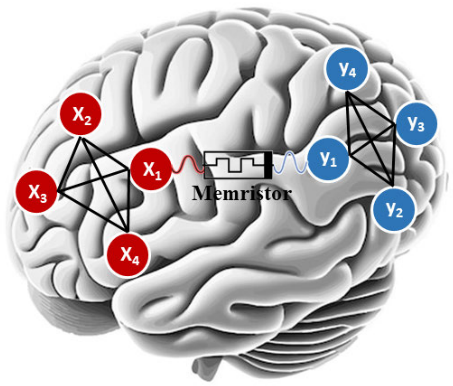


FIGURE 1. A memristor joining 2 sub-neural networks (adapted from [46]).

Figure 1 depicts the joining of the two sub-neural networks using a memristor, where X_i and Y_i are eight neurons. Hence, assuming $C_i = 1$, $R_i = 1$, $I_i = 0, i \in \{1, 2, 3, 4\}$, the MCNNM can be expressed as in (2):

$$\begin{cases} \dot{x}_1 = -x_1 + 1.8tanh(x_1) + 2tanh(x_2) - 0.5\ tanh(x_3) \\ \quad -12tanh(x_4) + p\varphi(x_1 - y_1), \\ \dot{x}_2 = -x_2 + tanh(x_2) + 20tanh(x_3) - 0.5\ tanh(x_4), \\ \dot{x}_3 = -x_3 + 0.5tanh(x_1) - 4tanh(x_2) + 1.8\ tanh(x_3) \\ \quad +4tanh(x_4), \\ \dot{x}_4 = -x_4 + 0.82tanh(x_1) - 0.5tanh(x_3) + 2\ tanh(x_4), \\ \dot{y}_1 = -y_1 + tanh(y_1) + 0.5tanh(y_2) - 3.5\ tanh(y_3) \\ \quad -tanh(y_4) - p\varphi(x_1 - y_1), \\ \dot{y}_2 = -y_2 + 2.8tanh(y_2) + 3tanh(y_3) + 0.5\ tanh(y_4), \\ \dot{y}_3 = -y_3 + 3tanh(y_1) - 3tanh(y_2) + tanh(y_3) \\ \quad -0.7tanh(y_4), \\ \dot{y}_4 = -y_4 + 0.5tanh(y_2) + tanh(y_3) + tanh(y_4), \\ \dot{\varphi} = sin(\pi\varphi) + (x_1 - y_1). \end{cases} \qquad (2)$$

where x_i and y_i are the membrane potential of neurons X_i and Y_i; φ is the inner state flux variable; $p\varphi(x_1 - y_1)$ is the add-on non-linear term that denotes the induction current between adjacent neurons X_1 and Y_1 with distinct membrane potentials; p is the coupling strength of the memristive magnetic induction effect, and $sin(\pi\varphi)$ denotes an extra magnetic flux produced by the membrane potential fluctuation.

The robustness of the MCNNM system in (2) is scrutinized using varying chaotic analyses encompassing coupling strength-related dynamic behaviors, preliminary state-related dynamic behaviors, as well as initial-boosted coexisting hyperchaotic attractors in [46].

TABLE 1. State-of-the-art of the research on data hiding in 3D mesh models.

Author/Year	Method	Added Tech.	Pros	Drawback	Attacks Resistance
Li et al. [22] 2017	•Usage of truncated space of data •Shifting approach	•PCA for alignment	•Low distortion without proliferating with secret data size •High-capacity	•Extraction failure due to using PCA	•N/A
Chao et al. [23] 2009	•Multilayered embedding	•N/A	•Providing capacity up to $3n_{layers}$ bits/vertex, for n_{layers} in [7,13] •Low distortion	•Not robust •Not suited for digital content protection & authent. apps	•Not resistant to attacks (smoothing, additional noise, non-uniform scaling, simplification, & vertices resampling)
Wang et al. [33] 2005	•Multi-level embedding •Substitutive blind procedure	•Use advanced jump strategy to improve triangular traversal	•High hiding capacity up to 3 bits/vertex	•Machine precision errors especially with small triangles	•N/A
Cayre et al. [32] 2003	•Substitutive procedure	•Use of PCA to make the scheme signal-dependent	•High capacity •Improved security •Fine parameterization of the algorithm on both geometrical & topological aspects	•Ineffectual triangle traversal •Poor robustness when performing local mesh manipulations (simplification or remeshing) •High processing time	•Robust against translation, rotation & scaling attacks •Robust to affine transformation
Zhang et al. [42] 2024	•Reversible data hiding •Multi-layer mesh synchronous	•Logical mapping strategy •Multilevel bimodal mapping rule	•Minimum distortion •Enhanced embedding capacity $\approx$ 3 bit per vertex in each layer •SNR $>$ 92 dB	•N/A	•Resists geometric distortion attacks
Mukherjee et al. [44] 2022	•Shifting strategy •Constructing VCI to decrease distortion by changing truncation's length	•N/A	•High-capacity •Adjustable distortion •High imperceptibility •Efficiency & robustness	•N/A	•Resists attacks (e.g. vertex reordering, rotation, uniform scaling)
Alkhamese et al. [41] 2024	•Gray code sequence to select vertices •LSB to embed compressed data	•N/A	•High-capacity •High security •Minimal distortion PSNR <150 dB	•N/A	•Robust against noise, filtering, & vertex reordering attacks.
Bandy. et al. [43] 2024	•Parallel Breadth First Search (BFS) with hyper-objects	Pennant data structure *bag* to parallelize BFS	•High Embedding Capacity (9 bps) •Superior visual quality •Low time complexity	•N/A	•N/A
			Adaptive		
Farrag and Alexan [24] 2020	•Blind adaptive mesh traversal order depending on the shortest distances between neighboring vertices	•N/A	•Adaptive in distinguishing between smooth & noisy surfaces of meshes in embedding •High capacity •High imperceptibility	•N/A	•Resists geometrical rotation & translation attacks •Withstands local variance noise, Poisson distribution noise and salt & pepper noise attacks •Resists vertex reordering attack
Zhang et al [25] 2015	•Highly adaptive •Mapping vertex coordinates into integers to embed in built bit-planes	•FPD to gauge distortion	•High capacity •Secure	•Inadequate robustness	•Not resistive to geometric attacks
Zhou et al [27] 2019	•Adaptively embed data in highest bitplanes	•N/A	•Acceptable steganalysis results as acting on binarized bitstream of meshes' vertices •High capacity	•Thwarts security	•N/A
Thiyagarajan et al [29] 2013	•Blind adaptive (pattern-based) •Re-triangulates a triangle mesh partition for embedding	•N/A	•High capacity up to 9 bpv & image geometric features •High imperceptibility •Low distortion	•Not reversible •Cannot withstand geometrical transformations	•Resistance to attacks (cropping, rotation, scaling, translation, noise addition, & filtering mechanisms)
Tsai [30] 2014	•Blind adaptive •Uses a vertex decimation process to select its referencing neighbors	•N/A	•Improved complexity estimation accuracy •High capacity (variable depending on vertex surface features) •High imperceptibility	•High time complexity	•N/A

B. THE MERSENNE TWISTER

The Mersenne Twister, developed by Matsumoto and Nishimura in 1997 [47], is one of the most widely used pseudo-random number generators (PRNGs) due to its exceptionally long period of $2^{19937} - 1$ and high efficiency [48]. It generates rapidly uniformly distributed random numbers,

TABLE 1. *(Continued.)* State-of-the-art of the research on data hiding in 3D mesh models.

Author/Year	Method	Added Tech.	Pros	Drawback	Attacks Resistance
Zhou et al. [27] 2019	•Adaptive relying on steganalytic characteristics viz. variation of vertex •Considering intricacy of local regions to manipulate vertices	•Coding with syndrome trellis	•Strong security with respect to steganalysis •High resistance to steganalysis tools	•Complex	•N/A
			Combined With Encryption		
Gao et al. [45] 2024	•Reversible •Adaptive vertex grouping strategy •Multi-MSB prediction & Huffman coding for vertices compression	•Secret sharing method over Galois field •Stream cipher technique	•High embedding capacity •Reversible data hiding	•N/A	•N/A
Jiang et al. [28] 2018	•Reversible •Map vertex coordinates before encrypting •Embeds encrypted data using LSB	•Bit-stream encryption	•High capacity •Perfect extraction	•AER = 4.22%, could be reduced using ECCs but may reduce capacity	•N/A
Alexan et al. [26] 2022	•Hiding encrypted data using Gray code sequence	•Blowfish •AES–128	•Distinguishes the arrangement of vertices •High capacity •High imperceptibility •Resist attacks	•Partially resists smoothing attacks	•Withstands geometrical similarity attacks (reflection, uniform scaling, rotation & translation)
Yasser et al. [36] 2020	•AES secured bit-cycling stega. in sliced 3D images •Uses Wolfram Mathematica Language	•AES •Two LSB stega. tech. (arithmetic & geometric) to embed	•N/A	•N/A	•N/A
Elsherif et al. [37] 2019	•Multi-layers (AES-128 with repetition code, LSB & noise layer)	•AES-128 •LSB	•Reversible •Blind extraction •Added security through noise •Withstands attacks	•N/A	•Resists geometrical attacks
Alexan et al. [38] 2019	•Two-layers •Using arithmetic, geometric, Fibonacci & Gray–code sequences to select number & slices sequence to embed in	•AES-128 •LSB	•High capacity •Highly secure	•N/A	•N/A
A. Amin in [39] 2019	•Two-layers •Two sequences (Arithmetic & Fibonacci) to select slice to embed	•Blowfish •LSB	•High capacity •Highly secure	•N/A	•N/A
Y. Moussa [40] 2020	•Two-layers •Applying edge detection after segmenting 3D image into 2D segments	•AES-256 •LSB	•N/A	•N/A	•N/A
			Using Chaos Theory		
Girdhar and Kumar [31] 2019	•Blind adaptive & reversible •Stopover closest neighbor first using breadth first search algorithm •Difference between vertices is shifted before hiding	•Chaotic Logistic map to select coordinates for embedding	•Unsystematic pattern of embedding •Having a distinctive mesh traversal algorithm for every 3D mesh model	•Hiding capacity can be enhanced	•Resists attacks e.g. vertex reordering, rotation, scaling & translation attacks
Elkandoz et al. [34] 2019	•Merge crypt., LSB & chaotic map •Segments 3D into 2D images then mix using Sine Logistic Map	•Uses AES-128 then XORing with chaotic 2D hyperchaotic map key •LSB	•High security using hyperchaotic map generated key	•N/A	•N/A
Y. A. Hassan [35] 2020	•Multi-layers (cryptography, edge detection and steganography)	•Chaotic function for encryption •LSB of edge pixels (Canny edge detection to select positions)	•High security	•N/A	•N/A

making it an ideal choice for applications such as simulations, modeling, and testing. Its ability to produce high-quality random numbers has established it as a standard in various computational fields [49]. The Mersenne Twister's design ensures that the sequence of numbers does not repeat for a very long time, thus providing a robust solution where large volumes of random data are required. Figure 2 shows an example of a plot of the array of 100 × 100 bits produced by the Mersenne Twister.

IV. PROPOSED INFORMATION SECURITY SCHEME

This section aims to elaborate on the steps of encrypting, and then embedding a string in a 3D model. In addition to that, the steps of retrieving the string are demonstrated as well.

A. ENCRYPTION AND EMBEDDING ALGORITHMS

In this section, the encryption and embedding scheme is presented as per the following steps:

1) Initialization phase:
   a) Given an input string S of length n, each character is first transformed into a byte using the ASCII table, which is later transformed into a bit-stream S_{bits} of length $n \times 8$.
   b) Given a set of seeds for the MCNNM system $Seed_{HNN}$, the MCNNM system is numerically solved,[1] in a similar manner as in [50], generating the bit-stream Key_{HNN} of the same length $n \times 8$.
   c) Toward generating an S-Box, a seed for the S-Box $Seed_{SBox}$, Mersenne Twister is utilized to generate a set of random integers S_{int} within the range $[0, (n \times 8) - 1]$ with length $(n \times 8) \times m$ for m being a significantly large coefficient (20 for example); finally, the S-Box is created such that $SBox = RemoveDuplicates(S_{int})$, in a similar manner as in [50].
2) Encryption phase:
   a) Given the sets S_{bits} and Key_{HNN}, the XOR operation is utilized to generate the bit-stream $S_{bits,HNN}$.
   b) $SBox$ is then applied to $S_{bits,HNN}$ producing $S_{bits,HNN,SBox}$.
   c) This encryption phase can be repeated multiple times, given multiple keys and S-Boxes.
3) Embedding phase:
   a) Given $S_{bits,HNN,SBox}$, every 3 bits are converted back to decimal, producing a list of integers within the range $[0, 7]$, then all integers are incremented by 1, resulting in making a list $S_{bits,HNN,SBox,Int}$ which ranges between 1 and 8.
   b) For a given number $count$, which indicates the number of integers per index, $S_{bits,HNN,SBox,Int}$ is divided into sub-sets of size $count$.
   c) Analyzing the 3D model, going through all indices (3 per vertex for x, y, and z), the index with the smallest power coefficient is retrieved, for example, for integers, the smallest coefficient is 10^1. However, for fractions, the coefficient would be 10^{-c} for some c, which is the output of this step.
   d) For each index I and $S_{bits,HNN,SBox,Int}$ subset of size $count$, digits $I^{-(c-count)}$ till I^{-c} are replaced by the subset from $S_{bits,HNN,SBox,Int}$ (as these digits with low coefficients concerning the indices values will not cause a visible change to the 3D model).
   e) For indices which are not utilized, digits $I^{-(c-count)}$ till I^{-c} are replaced by 0's or 9's in order not to be confused with actual embedded digits.

The initialization, encryption, and embedding schemes described are represented visually in flow chart form in Fig. 3, as well as in algorithm form in Algorithm 1 and Algorithm 2. A working example with numerical values is provided in Appendix.

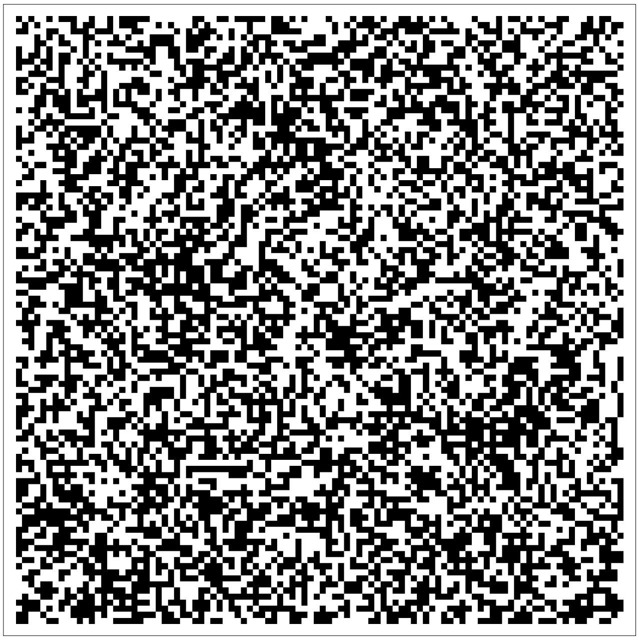


FIGURE 2. A 100 × 100 array plot of a bit-stream produced using the mersenne twister. A seed value of 31415926535 is utilized.

B. EXTRACTION AND DECRYPTION ALGORITHMS

In this section, the extraction and decryption scheme is presented as per the following steps:

1) Initialization phase:
   a) In addition to all the initialization steps performed in the encryption and embedding part, the inverse of $SBox$ is constructed.
2) Extraction phase:
   a) Given $count$, for each index I in the 3D model, the last $count$ digits are extracted in a set excluding 0 and 9 digits, 1 is subtracted from all elements of this set, forming $S_{bits,HNN,SBox,Int}$.
3) Decryption phase:

[1] This is carried out in Wolfram Mathematica® using the NDSolve command.

Algorithm 1 Initialization Phase

```
//Inputs
n;
originalData;
HNNSystemInputs={p,x10,x20,x30,x40,y10,y20,y30,y40,z0};
solverRates={r1..r9};
3dModel;
//Inits
//transform data from char code to bit stream
originalDataAsBits=toBaseTwo(flatten(getCharacterCodeList(originalData)));
//solve HNN system
HNNBits=SolveHNN(lngth(originalDataAsBits),HNNSystemInputs,solverRates);
//generate SBox and its inverse
SBox=generateSBox(length(originalDataAsBits));
SBoxInverse=generateSBoxInverse(SBox);
```

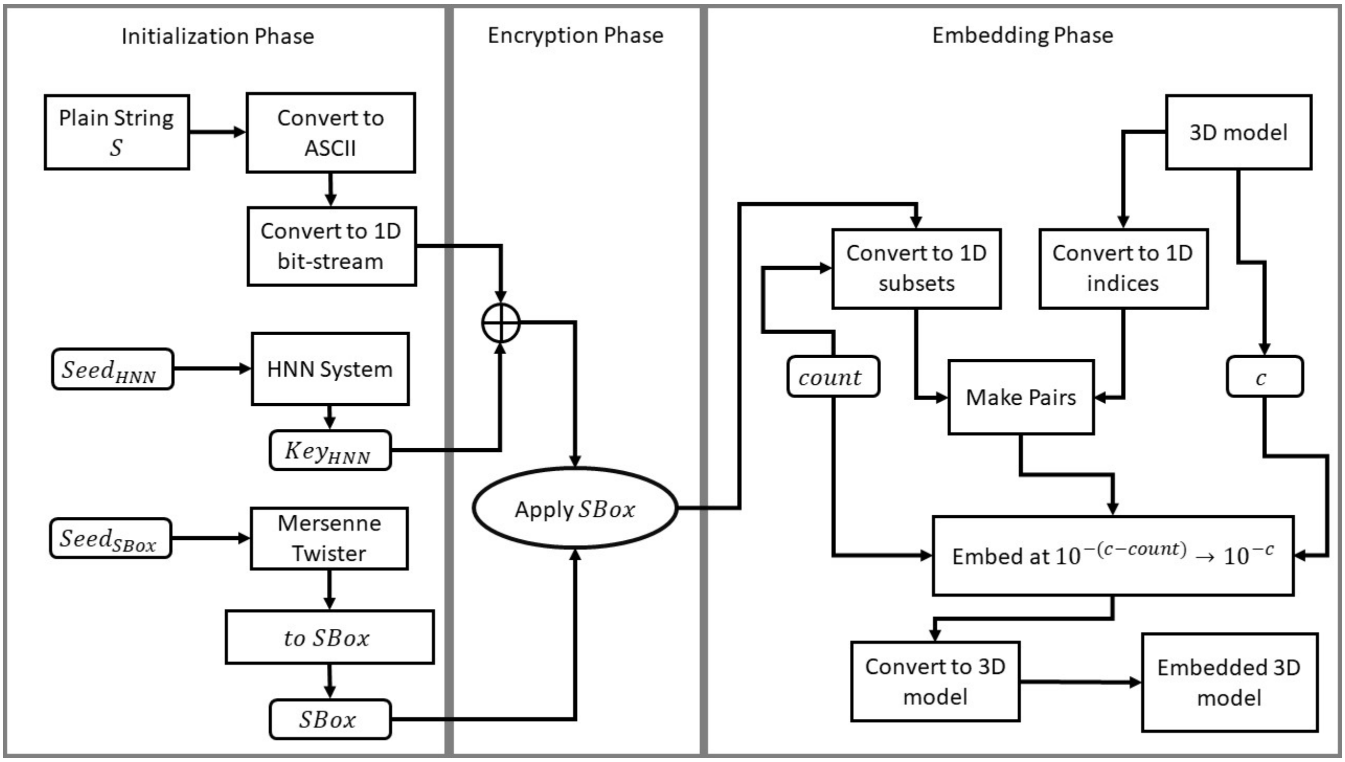


FIGURE 3. Encryption and embedding algorithm flow chart.

a) Each integer in $S_{bits,HNN,SBox,Int}$ is converted into 3 bits such that the concatenation of all converted bits produces the bit-stream $S_{bits,HNN,SBox}$.

b) The inverse of $SBox$ is applied to $S_{bits,HNN,SBox}$ generating $S_{bits,HNN}$.

c) XOR is applied on $S_{bits,HNN}$ and Key_{HNN} resulting in S_{bits}.

d) For multiple S-boxes and keys, the previous 2 steps are repeated for each pair.

e) Each 8 bits in S_{bits} are converted into a byte, which is further converted into a character using the ASCII table, reforming the input string.

The extraction and decryption schemes described are visually represented in the flow chart form in Fig. 4, as well as in the algorithm form in Algorithm 3. A working example with numerical values is provided in Appendix.

V. PERFORMANCE ANALYSIS

This section performs the performance evaluation of the proposed algorithm. It starts by describing the experimental environment, as well as the dataset utilized of 3D models, in subsection V-A. This is followed by performance evaluations for steganography and cryptography, in Subsections V-B and V-C, respectively.

Algorithm 2 Encryption and Embedding Algorithms

```
encrypt()
{
    Accum=originalDataAsBits;
    for(i=1;i<=n;i++)
    {
        Accum=BitXor(Accum,HNNBits);
        Accum=applySbox(SBox,Accum);
    }
    return Accum;
}

embed(dataBitsToEmbed)
{
    //every item is a list of 3~indices
    modelVer=importModelAsVertices(3dModel);
    //flatten into a list of all indices
    modelIndices=flatten(modelVer);

     //group every 3~bits together
    groupedData=group(dataBitsToEmbed,3);
    //Transform every 3~bits to their base-10 value and add 1.
    // possibe values are from~1~to~8
    groupedData=toBaseTen(groupedData)+1;

    //number of digits per index
    D=Ceiling(length(groupedData)/length(modelIndices))

    //group the digits into \text{sub-groups} of length D
    groupedData=group(groupedData,D);

    for(i=1;i<=length(groupedData);i++)
    {
        //replace the right most D fractional digits in modelIndices[i] with
        //groupedData[i]
        replaceRightDigits(groupedData[i],D,modelIndices[i]);
    }

    for(i=length(groupedData)+1;i<=length(modelIndices);i++)
    {
        //replace the right most D fractional digits in modelIndices[i]
        // with a random sequence of length D 0s and 9s
        replaceRightDigits(RandomDigitsSequence({0,9},D),D,modelIndices[i]);
    }

    //group the indices into vertices again
    modelVer=group(modelIndices,3);
    return modelVer;
}
```

A. EXPERIMENTAL ENVIRONMENT, DATASET, AND PARAMETERS

Table 2 displays the various 3D models utilized in this work, along with their data. These models were imported from the Wolfram Data Repository [51], which is largely based on the Stanford set of 3D models. This choice is made because of the high quality, variety, and accessibility of the set, which ensures robust and versatile testing of the proposed algorithm. Furthermore, the standardized data set allows easier benchmarking and comparison between studies. Its complexity and realism provide a realistic testing ground, while its long history and support for various formats add credibility and ease of use.

The implementation of the proposed algorithm is carried out using the computer algebra system Wolfram Mathematica® v. 13.2 on a machine running a 64–bit operating system with 16 GB of RAM and an Intel® Core™ i7–7700HQ CPU with a maximum clock rate of 2.8 GHz. The metrics computed in the following subsections are for $N = 1$. Further iteration cycles (i.e. $N > 1$) could be carried out, increasing the encryption levels according to security needs.

The Mersenne Twister implementation on Wolfram Mathematica® was fed a seed with the value of 31415926535 (first 11 digits of π, with the radix point omitted). The MCNNM was numerically solved on Wolfram Mathematica®, utilizing initial values $x_1 = x_2 = x_3 = x_4 = 1, y_1 = y_4 = 1, y_2 = y_3 = 0, \rho = 0.17, \phi = 1$, to allow its hyperchaotic behavior to be exhibited and made use of.

B. STEGANOGRAPHY PERFORMANCE EVALUATION

Table 3 shows the analysis of the comparison of 3D models before and after the steganography step. The following performance evaluation measures are carried out: Visual inspection ensures perceptual integrity, MSE and PSNR quantify overall distortion, Region Hausdorff Distance (RHD) measures localized changes, and Embedding Capacity examines the amount of sensitive data that could possibly be embedded in the 3D models. These metrics ensure a comprehensive evaluation of both the visual and structural integrity of the 3D models, as well as the security of the hidden data. Other metrics might not be as apt due to their focus on different aspects of data quality or security that are less relevant to the specific requirements of 3D steganography.

1) VISUAL INSPECTION

The effectiveness of the proposed algorithm is illustrated through a visual comparison of the "Stanford Bunny" model in its carrier and stego states, as shown in Fig. 5. The carrier model (Fig. 5a) represents the original, unmodified 3D model. The stego 36 model (Figure 5b) and the stego 63 model (Fig. 5c) depict the 3D geometry after embedding different encrypted payload data.

Upon close inspection, it is evident that the modifications introduced by the data hiding process are imperceptible to the naked eye, even at higher embedding capacities. The visual quality of the stego models remains virtually unchanged compared to the carrier model, validating the algorithm's ability to conceal data without compromising the aesthetic integrity of the 3D object. This imperceptibility is crucial for applications where maintaining the visual fidelity of the model is essential.

2) MEAN SQUARED ERROR

The Mean Squared Error (MSE) is a crucial measure for evaluating the imperceptibility of steganographic embedding in 3D models, representing the average squared difference between the vertex values of the original (cover) and the steganographically modified (stego) models. An ideal steganographic method would yield an MSE of zero, indicating that there are no detectable changes post-embedding. The MSE is calculated as follows:

$$MSE = \frac{1}{K}\sum_{i=1}^{K}(x_i - \hat{x}_i)^2, \tag{3}$$

where K is the number of vertices in the 3D model, x_i is the original value of the i^{th} vertex in the cover model, and $\hat{x}_i$ is the value of the i^{th} vertex in the stego model.

As shown in Table 3, the proposed steganographic algorithm aims for and achieves an MSE of almost zero, demonstrating that the proposed scheme embeds data without introducing any geometric distortion into the model 3 D. This result confirms the invisibility of the steganographic embedding and ensures the resistance of the method to statistical detection. Thus, the proposed approach provides an effective means of data steganography in 3D models, maintaining the integrity of the original model while securely embedding information.

3) PEAK SIGNAL-TO-NOISE RATIO

The Peak Signal-to-Noise Ratio (PSNR) is a widely recognized metric that is used to evaluate the quality of steganographic embedding in 3D models. It quantifies the ratio between the maximum possible power of a signal (representing the original model's vertices) and the power of corrupting noise (the alterations made during embedding) that affects its representation. PSNR is defined by the following formula:

$$PSNR = 10\log_{10}\left(\frac{I_{Max}^2}{MSE}\right), \tag{4}$$

where I_{Max} denotes the maximum possible value of the model's vertices.

The goal in steganography is to maximize the PSNR value, which signifies minimal distortion and high fidelity of the stego model. A higher PSNR usually indicates that the steganography method is more effective, as it implies less noise and therefore less detectable changes to the model.

As shown in Table 3, the performance of the proposed embedding algorithm reflects exceptionally high PSNR

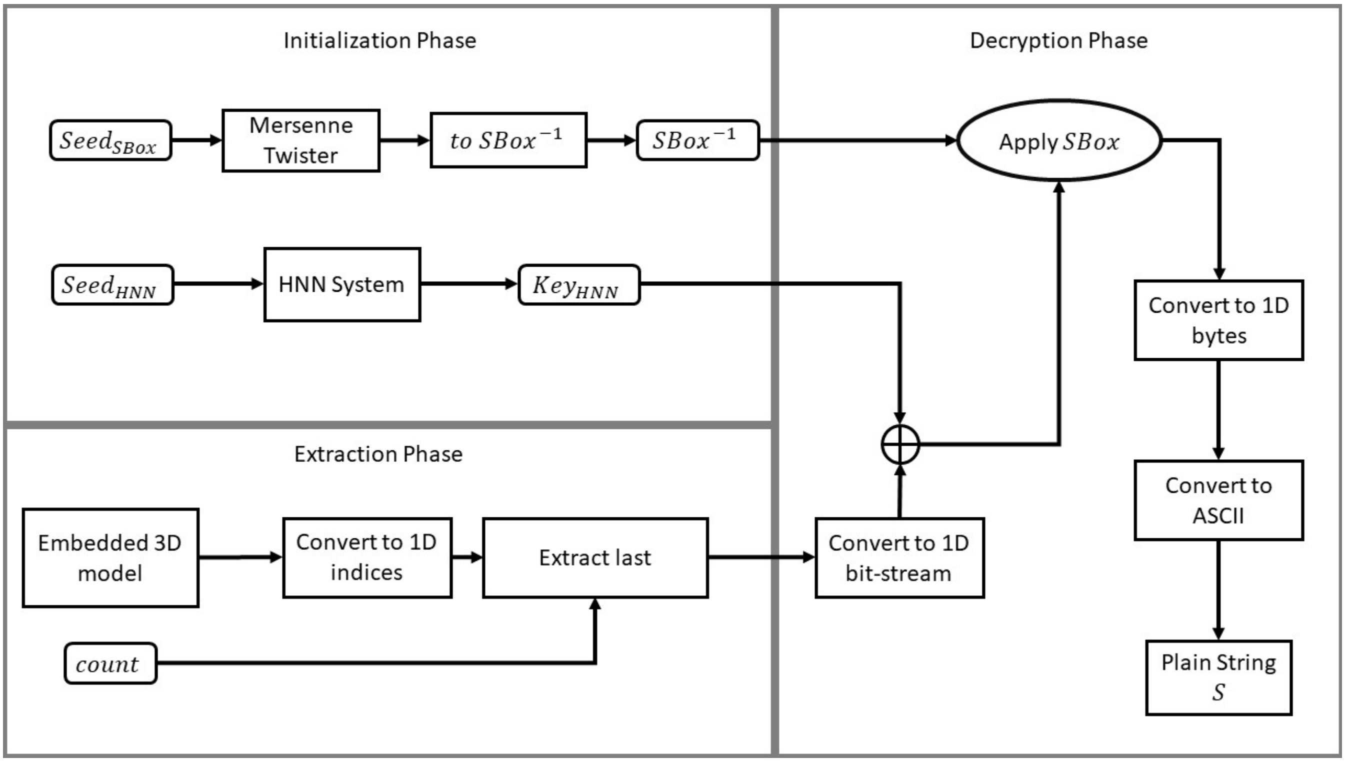


FIGURE 4. Extraction and decryption algorithm flow chart.

values, indicating that the steganographic alterations are virtually undetectable, maintaining the original quality of the 3 D model. These high PSNR values showcase the imperceptibility and robustness of our steganographic technique, ensuring the secure embedding of the data without compromising the visual and structural integrity of the 3D model.

4) REGION HAUSDORFF DISTANCE

In assessing the geometric distortions introduced by steganographic embedding in specific regions of 3D models, the Region Hausdorff Distance (RHD) serves as an important metric. The RHD is a measure of the greatest of all the distances from a point in one region to the closest point in the other region and is defined as:

$$\mathrm{RHD} = \max \left\{ \begin{array}{l} \max\{\min_{p\in R_1, q\in R_2} Norm[p-q]\} \\ \max\{\min_{p\in R_2, q\in R_1} Norm[p-q]\} \end{array} \right. \quad (5)$$

where R_1 and R_2 are the two regions being compared, p and q represent points in the respective regions, while $Norm[p-q]$ represents the distance between points p and q.

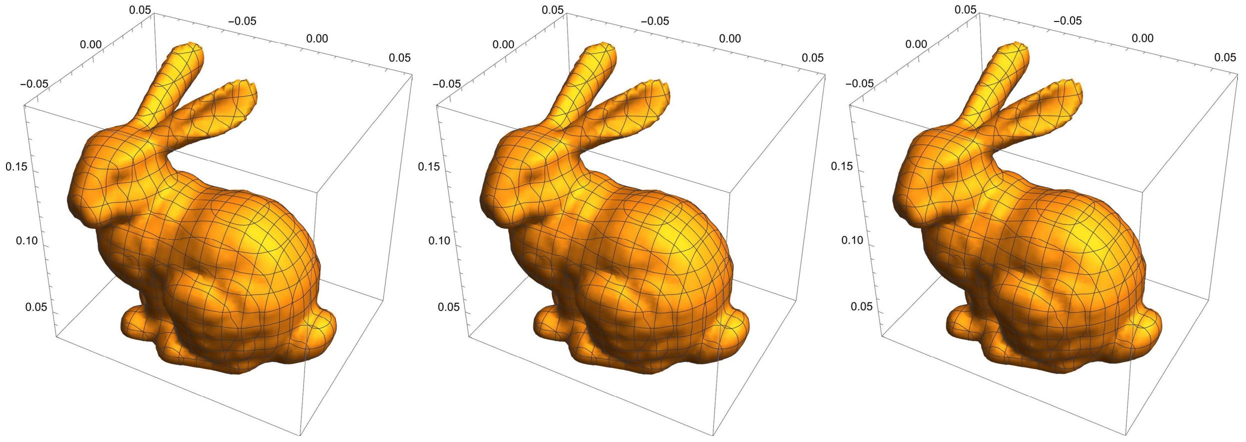
For the proposed steganography algorithm, the RHD is calculated to quantify the maximum distortion at a regional level within the 3D model after data embedding. An optimal steganographic process should aim for the lowest possible RHD, indicating minimal localized distortion and high fidelity between the cover and stego models. The analysis here, utilizing RHD, confirms that the regional alterations in our steganographically embedded 3D models remain below perceptible thresholds (practically approaching a value of zero, as shown in Table 3), ensuring both the integrity of the model and the imperceptibility of the embedded data.

5) EMBEDDING CAPACITY

The critical metric of embedding capacity (EC), in bits per vertex (bpv), which quantifies the volume of data that can be securely embedded within a cover medium, is analyzed for the proposed algorithm. As demonstrated in Table 4, the proposed scheme shows the highest embedding capacity among current state-of-the-art techniques. This indicates an ability to conceal more substantial amounts of data within 3D models, while maintaining the imperceptibility of the data and the integrity of the model structure. The security is enhanced and the applicability of steganographic methods is broadened because of this capacity advantage, positioning the proposed scheme as a significant tool in the domain of data security.

C. CRYPTOGRAPHY PERFORMANCE EVALUATION

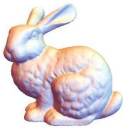
Before embedding within 3D models, the data is encrypted to enhance security. The security level of this encrypted data is evaluated in this subsection. The following cryptography performance evaluation metrics are utilized: A key space analysis, NIST SP 800-22 analysis, and an S-box performance analysis. A key space analysis ensures the



(a) Carrier model. (b) Stego 36 model. (c) Stego 63 model.

FIGURE 5. A side by side comparison of the "Stanford Bunny" model in the carrier and various stego data payloads.

TABLE 2. Stanford 3D models employed in this work.

Name	3D Model	Vertices	Polygons
Bunny		34834	69451
Armadillo		172974	345944
Dragon		435545	871306
H. Buddha		543524	1087451

algorithm's robustness against brute-force attacks, while a NIST SP 800-22 analysis assesses the randomness of the embedded data. An S-box performance analysis evaluates the strength of the nonlinear substitution process, and the time and space complexity analyses measure the algorithm's efficiency. These metrics collectively ensure a thorough evaluation of security and efficiency, while other metrics may not provide a comprehensive assessment of both cryptographic strength and operational performance in the context of 3D steganography.

Algorithm 3 Extraction and Decryption Algorithms

```
{
    //flatten into a list of all indices
    modelIndices=flatten(modelVer);

    //number of digits per index
    D=Ceiling((length(SBox)/3)/length(modelIndices));

    //empty list for extracted data
    extractedData={};

    for(i=1;i<=length(modelIndices);i++)
    {
        //Append the D right most digits from every index to the list
        Append(extractedData, getRightDigits(D, modelIndices[i]));
    }

    //Cleanup: exclude 0s and 9s
    removeZerosAndNines(extractedData);

    //flatten and transform into bits (base-2)
    extractedData=toBaseTwo(flatten(extractedData));
    return extractedData;
}

decrypt(extractedData)
{
    Accum=extractedData;
    for(i=1;i<=n;i++)
    {
        //Reverse the encryption process
        Accum=applySbox(SBoxInverse, Accum);
        Accum=BitXor(Accum, HNNBits);
    }
    return Accum;
}

showOriginalData(decryptedData)
{
    return toCharacterCode(group(extractedData, 8));
}
```

1) KEY SPACE ANALYSIS

A comprehensive analysis of the key space is critical to evaluate the security strength of the proposed scheme. The key space refers to the total number of possible keys that can be used in the encryption process, determining the algorithm's resilience against brute-force attacks. The proposed algorithm features a key space that is expansive enough to thwart any practical brute-force attempt. The size of the key space has been designed to exceed current computational capabilities, ensuring that the cost and time required to crack the encryption by exhaustively searching all possible keys are prohibitive.

The MCNNM in (2) has 37 coefficients, as well as 9 initial values, which provides a total of $37 + 9 = 46$ values. For a machine precision of 10^{-16}, this results in a key space of $10^{46\times16} = 10^{736} \approx 2^{2445}$. This value is much higher than the threshold established earlier in the literature [54] of 2^{100}, for resistance to brute-force attacks. Furthermore, it is superior to AES-256, which is only 2^{256}.

TABLE 3. Analytics of 3D models.

3D Model	EC [bits]	MSE	PSNR [dB]	RHD
Bunny	2194542	1.434×10^{-15}	133.886	1.442×10^{-7}
Armadillo	10897362	7.436×10^{-10}	131.029	8.562×10^{-5}
Dragon	27439335	4.139×10^{-18}	159.728	8.403×10^{-9}
H. Buddha	34242012	4.261×10^{-18}	161.586	7.712×10^{-9}

TABLE 4. Comparison of embedding capacity in various domains in the literature.

Year	Algorithm	Domain	Capacity [bpv]	Embedding Location
2006	[52]	Representation	9	Vertices and Polygons
2010	[53]	Geometrical	0.7	Vertices
2013	[29]	Geometrical	3	Vertices
2014	[30]	Topological	15.6	Vertices
2018	[27]	Geometrical	6	Vertices
2019	[24]	Geometrical	6	Vertices
2023	[45]	Geometrical	43	Vertices
2024	[25]	Geometrical	6	Bit-planes
2024	[41]	Geometrical	1.445	Vertices
2024	[43]	Geometrical	9	Vertices
2024	[42]	Geometrical	3	Vertices
2024	Proposed	Geometrical	63	Vertices

2) NIST SP 800-22 ANALYSIS

The encrypted bitstream undergoes rigorous evaluation using the National Institute of Standards and Technology (NIST) Statistical Test Suite, which is a benchmark for assessing the randomness of cryptographic output. According to the results presented in Table 5, the encrypted bitstream has successfully passed all the NIST tests, with all p-values exceeding the threshold of 0.01. This indicates that the encryption method is robust, producing a random bitstream suitable for secure steganographic embedding in 3D models.

3) S-BOX PERFORMANCE ANALYSIS

Substitution boxes play a crucial role in the realm of cryptography, acting as the fundamental unit for non-linearly transforming input into output. To assess the strength of an S-Box within an encryption scheme, it is subjected to a battery of important tests that evaluate various facets of its strength. These are:

1) Non-linearity (NL): This metric scrutinizes the resemblance of an S-Box's output to that of affine functions, which are essentially the linear combinations of input bits [55]. By examining the truth table of a Boolean function, one can gauge the minimal bit adjustments needed to morph the function into its closest affine form. A high degree of NL is imperative for obscuring the relationship between plaintext and ciphertext, thus fortifying the system against linear forms of cryptanalysis.
2) Linear Approximation Probability (LAP): The LAP metric determines the likelihood of an S-Box to exhibit linear characteristics, a trait that can undermine encryption by introducing exploitable biases [56]. A competent S-Box maintains a minimal LAP to signify its resilience to linear cryptanalytic techniques, thereby reinforcing the overall encryption strength.
3) Differential Approximation Probability (DAP): This measurement focuses on the effects that particular input

TABLE 5. NIST analysis carried out on a long bitstream of encrypted data.

Test	p-value	Remarks
Frequency	0.494699	Success
Block Frequency	0.228973	Success
Runs	0.827507	Success
Longest run of ones	0.980481	Success
Rank	0.951991	Success
FFT	0.608527	Success
Non-overlapping T.M. (000000001)	0.668100	Success
Overlapping T.M.	0.447111	Success
Maurer's Universal	0.795211	Success
Linear complexity	0.503029	Success
Serial 1	0.672567	Success
Serial 2	0.531770	Success
Approx. entropy	0.717358	Success
Cum. sums forward	0.842260	Success
Cum. sums reverse	0.467738	Success
Random ex. 1	0.751825	Success
Random ex. 2	0.830189	Success
Random ex. 3	0.697840	Success
Random ex. 4	0.556060	Success
Random ex. 5	0.246731	Success
Random ex. 6	0.522334	Success
Random ex. 7	0.772276	Success
Random ex. 8	0.818164	Success
Random ex. var. 1	0.164867	Success
Random ex. var. 2	0.221983	Success
Random ex. var. 3	0.161967	Success
Random ex. var. 4	0.141295	Success
Random ex. var. 5	0.180678	Success
Random ex. var. 6	0.154929	Success
Random ex. var. 7	0.341492	Success
Random ex. var. 8	0.731220	Success
Random ex. var. 9	0.952557	Success
Random ex. var. 10	0.341126	Success
Random ex. var. 11	0.724771	Success
Random ex. var. 12	0.689809	Success
Random ex. var. 13	0.180894	Success
Random ex. var 14	0.092807	Success
Random ex. var. 15	0.107395	Success
Random ex. var. 16	0.074726	Success
Random ex. var. 17	0.081233	Success
Random ex. var. 18	0.104509	Success

bit variations have on the outputs [57]. The objective is to evaluate the degree to which output alterations can be foreseen following specific input modifications. An S-Box displaying a low DAP is indicative of its efficacy in mitigating differential cryptanalysis by ensuring that changes in input do not translate to predictable output patterns.

4) Bit Independence Criterion (BIC): This parameter measures how independently each input bit variation affects the output bits [58]. An optimal S-Box would cause each output bit to vary independently in response to changes in any single input bit, thus preventing any systematic formation of patterns in the ciphertext that could be linked statistically to the plaintext.

5) Strict Avalanche Criterion (SAC): The SAC evaluates the degree to which altering a single input bit influences each output bit [58]. In a secure encryption system, altering one input bit should yield a 50% probability of changing each output bit, manifesting an avalanche effect. Accordingly, a slight input modification should instigate a substantial and unpredictable change in the output.

Table 6 displays the proposed S-box, while Table 7 shows the computed results for its performance evaluation metrics, confirming its security and efficiency. Furthermore, compared with the metrics from recent literature, the proposed algorithm's S-Box demonstrates a superior level of security performance.

D. TIME AND SPACE COMPLEXITY

This section evaluates the efficiency of the proposed algorithm for securing sensitive data in 3D models. The next subsections discuss the theoretical underpinnings and practical implementation of the proposed algorithm in relation to its time and space complexity, and compare those to AES-256.

1) TIME COMPLEXITY

- AES-256: $O(M)$, where M is the number of 128-bit blocks in the secret message.
- Proposed algorithm: $O(T + n + m + k)$, where T is the integration range, n is the number of solution points, m is the size of the data, and k is the number of points in the 3D model.

2) SPACE COMPLEXITY

- AES-256: $O(1)$, constant space regardless of input size.
- Proposed algorithm: $O(n + m + k)$, linear space complexity based on the number of solution points, data size, and 3D model points.

Theoretically, AES-256 is more efficient and has predictable complexity, which makes it suitable for high-performance and resource-constrained environments. However, for our specific application, the actual implementation of both shows that the proposed algorithm achieves shorter run times, as described next.

Practically, the pivotal data in Table 8 outline the time durations necessary for the encryption, embedding, extraction, and decryption processes. Short run times are highlighted as a significant advantage, with Table 8 revealing the algorithm's rapid encryption capability, essential for time-sensitive applications. In contrast, traditional AES-256 encryption was performed on the same data sample and resulted in run times varying between 9s and 10s, demonstrating the efficiency of the proposed approach in comparison. The embedding time is also noted for its brevity, underlining the algorithm's swift integration of data into 3D models. Additionally, the extraction and decryption times are presented, emphasizing the algorithm's expedient retrieval and decryption of data, which is vital for ensuring quick access to secure information. These minimal time durations are crucial in demonstrating the practicality and effectiveness of the algorithm in scenarios where speed is of the essence.

VI. DISCUSSION

The proposed double data security algorithm, which combines a two-stage encryption process with steganographic data hiding in 3D models, demonstrates significant advancements over traditional methods. This section discusses the implications of our findings, the performance of our algorithm, and its comparative advantages, with an emphasis on the use of the hyperchaotic system of differential equations.

A. ENHANCED SECURITY AND EFFICIENCY

Central to our approach is the use of a hyperchaotic system of differential equations, specifically a fractional-order memristive coupled neural network system, to generate dynamic encryption keys and construct S-boxes. This system introduces a high degree of unpredictability, significantly enhancing the security of the encryption process. Repetition of XOR and S-box operations further obscures the data, making it resistant to cryptographic attacks. Our results indicate that this method surpasses the standard AES-256 algorithm in terms of computational efficiency and resistance to brute-force attacks. The complex dynamics of the hyperchaotic system add an extra layer of security, making unauthorized decryption exceedingly difficult.

B. STEGANOGRAPHIC ANALYSIS

The integration of encrypted data into 3D geometries ensures that the visual integrity of the models remains intact. Visual inspections confirm that modifications are imperceptible to the human eye, maintaining the aesthetic quality of the 3D models. Quantitative metrics further support these findings; MSE and PSNR values indicate minimal overall distortion, while the RHD measurements confirm that localized changes are negligible. The embedding capacity, calculated in bpv, demonstrates that a substantial amount of data can be hidden within the 3D models without degrading their quality.

TABLE 6. HNN based S-box.

169	182	127	71	238	81	58	215	42	10	253	7	45	20	86	5
241	168	183	251	228	83	244	2	99	14	84	254	50	166	79	212
96	0	237	245	102	213	105	11	152	4	222	27	202	73	34	32
160	57	75	146	89	59	109	209	243	126	78	41	193	185	93	219
77	106	40	172	223	179	148	125	44	231	246	224	121	100	123	167
47	70	56	142	192	189	104	114	25	116	94	30	139	140	67	180
97	9	236	26	24	196	31	145	13	177	17	35	16	6	39	156
217	69	15	129	164	191	3	22	18	61	72	130	124	132	201	250
178	52	85	103	12	190	199	188	242	101	141	200	65	80	198	122
240	186	248	60	46	110	55	21	163	143	218	107	171	239	187	136
154	214	43	147	23	233	235	8	153	119	19	29	1	249	252	206
49	113	38	95	225	131	68	64	227	138	51	120	112	204	207	208
194	247	92	220	205	255	87	53	184	62	176	33	133	54	37	63
144	195	197	151	216	88	118	117	135	90	150	181	91	173	162	157
161	226	128	175	82	108	28	230	234	134	210	229	165	149	137	98
211	232	74	48	111	203	76	221	115	155	36	158	159	170	174	66

TABLE 7. S-box performance evaluation and comparison with the literature.

Year	S-Box	NL	SAC	BIC	LAP	DAP
	Ideal values	112	0.5	112	0.0625	0.0156
2019	[59]	107	0.497	103.5	0.1560	0.039
2021	[60]	112	0.5	–	–	–
2022	[61]	106	0.5019	112	0.1328	0.0391
2023	[62]	108	0.49414	108	0.07812	0.01562
2024	[63]	110	0.50073	108	0.07812	0.01562
2024	Prop.	108	0.49707	108	0.078125	0.01562

C. CRYPTOGRAPHIC PERFORMANCE

The proposed algorithm has been shown to be cryptographically secure. The carried out numerical analyses covered a key space analysis, demonstrating the algorithm's resistance to brute-force attacks with a significantly large key space. Additionally, it includes a NIST SP 800-22 analysis, which confirms the randomness of the encrypted data, and an S-box performance analysis, assessing nonlinearity, linear and differential approximation probabilities, and bit independence, all critical for maintaining robust encryption. The results indicate that the encryption method provides strong security for steganographic applications.

TABLE 8. 3D models performance run times.

3D Model	T_{Enc} [s]	T_{Emb} [s]	T_{Ext} [s]	T_{Dec} [s]
Bunny	1.71161	5.21244	1.69417	0.297099
Armadillo	1.64092	20.9626	3.47544	0.247567
Dragon	1.71545	52.1886	7.04946	0.243378
H. Buddha	1.61139	63.713	8.52558	0.242815

D. COMPARATIVE ADVANTAGES

Compared to single encryption or data hiding techniques, our dual approach offers superior protection. The combination of robust encryption and stealthy data hiding within 3D objects provides a comprehensive security solution that ensures privacy for sensitive digital data transmissions and storage. The numerical experiments validate the effectiveness of our algorithm, showing that data extraction from modified 3D models is accurate and that the visual impact is negligible.

TABLE 9. Working example.

Step	Inputs or outputs
Original plain data	Hi!
Original plain data in ASCII	$\{72, 105, 33\}$
Flattened data as a bit-stream	$\{0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1\}$
HNNStream	$\{0, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1\}$
Data encryption round 1 (XOR HNN then SBOX)	$\{1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1\}$
Data encryption round 2 (XOR HNN then SBOX)	$\{1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1\}$
Data encryption round 3 (XOR HNN then SBOX)	$\{0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1\}$
Small dummy 3D model vertices	$\{\{0.749614, 0.0240388, 0.603555\}, \{0.997944, 0.0640703, 2.41421 \times 10^{-6}\},$ $\{0.498973,0.0320352,1.20711\},\{3.20515\times10^{-8}, 9.99486 \times 10^{-7}, 2.41421\},$ $\{-0.096023,0.995378,2.41421\times10^{-6}\}\}$
Model, flattened	$\{0.749614, 0.0240388, 0.603555, 0.997944, 0.0640703, 2.41421 \times 10^{-6}, 0.498973, 0.0320352,$ $1.20711,3.20515\times10^{-8}, 9.99486 \times 10^{-7}, 2.41421, -0.096023, 0.995378, 2.41421 \times 10^{-6}\}$
Model, flattened and adjusted	$\{0.7496140000001, 0.0240388000001, 0.6035550000001, 0.9979440000001, 0.0640703000001$,0.0000024142101,0.4989730000001,0.0320352000001,1.2071100000001,0.0000000320511, 0.0000009994861,2.4142100000001,-0.096023000001,0.9953780000001,0.0000024142101}
String bits flattened and adjusted (for missing offset)	$\{0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1\}$
String bits partitioned into groups of three	$\{\{0, 0, 1\}, \{1, 0, 1\}, \{0, 0, 1\}, \{1, 1, 0\}, \{1, 1, 1\}, \{0, 0, 0\}, \{1, 1, 0\}, \{1, 0, 1\}\}$
String bits groups into digits	$\{2, 6, 2, 7, 8, 1, 7, 6\}$
Number of digits per index (in this example)	1
String digits after filler adjustment (prepared to be inserted into 3D model)	$\{2, 6, 2, 7, 8, 1, 7, 6, 9, 0, 0, 0, 9, 9, 9\}$
3D Model after string insertion	$\{0.74961400000012, 0.02403880000016, 0.60355500000012,$ 0.99794400000017,0.06407030000018,0.00000241421011,0.49897300000017, 0.03203520000016,1.20711000000019,0.00000003205110,0.00000099948610, 2.41421000000010,-0.0960230000019,0.99537800000019,0.00000241421019}
Extracted encrypted data	4î5
Extracted decrypted data	Hi!

E. LIMITATIONS AND POSSIBLE FUTURE IMPROVEMENTS

This work is not without limitations. It is clear that the embedding time (shown in Table 8) could be improved. This matter could be tackled in more than one way. First, a parallel processing architecture over multiple cores could be implemented in software. This would invariably improve the overall processing time. Second, an FPGA implementation could be carried out. This promises efficiency improvements in the order of a few orders of magnitude. Achieving cross-platform compatibility will ensure consistent performance across diverse systems, while large-scale application testing in scenarios like IoT communication can validate its real-world efficacy. Ongoing updates to strengthen its defenses against the latest security threats, such as those posed by quantum computing, are imperative. In addition, efforts to improve usability and ease of integration into current systems can facilitate wider adoption and contribute to further progress in digital data security.

VII. CONCLUSION AND FUTURE OUTLOOK

This research work introduced a cutting-edge double data security algorithm that significantly enhances the confidentiality of sensitive information. The proposed algorithm cleverly combines a two-tier encryption process, which leverages the complex dynamics of a fractional-order memristive coupled neural network system, with a subtle but effective technique of embedding encrypted data into 3D model geometries. The encryption keys and S-boxes, derived from the system's solutions, ensure a secure encryption process through XOR and S-box operations, while the steganographic component offers an additional layer of protection.

Performance evaluations of the proposed algorithm have provided promising results. The near-zero MSE and exceptionally high PSNR values ranging from 131 dB to 162 dB demonstrate the imperceptibility of the data embedding process. Furthermore, practically zero RHD attests to the minimal impact on the 3D models' geometrical integrity. These metrics not only demonstrate the algorithm's capability to secure data but also highlight its superiority over the conventional AES-256 algorithm in both security robustness and computational efficiency.

Future research on the proposed security algorithm should focus on improving its computational efficiency through algorithm optimization that maintains its high security standards. This could possibly be implemented through parallel processing or an FPGA implementation. Ensuring cross-platform compatibility and conducting large-scale IoT testing is crucial to consistent real-world performance. Continuous updates are needed to protect against emerging threats such as quantum computing. Additionally, improving usability and integration will promote greater adoption and advance digital data security.

APPENDIX

Table 9 provides a working example that illustrates the proposed algorithm in terms of its inputs and outputs. This example shows three encryption rounds ($N = 3$).

REFERENCES

[1] H. T. Elshoush and M. M. Mahmoud, "Ameliorating LSB using piecewise linear chaotic map and one-time pad for superlative capacity, imperceptibility and secure audio steganography," *IEEE Access*, vol. 11, pp. 33354–33380, 2023.

[2] A. Jan, S. A. Parah, M. Hussan, and B. A. Malik, "Double layer security using crypto-stego techniques: A comprehensive review," *Health Technol.*, vol. 12, no. 1, pp. 9–31, Jan. 2022.

[3] A. Pradhan, K. R. Sekhar, and G. Swain, "Digital image steganography using LSB substitution, PVD, and EMD," *Math. Problems Eng.*, vol. 2018, Sep. 2018, Art. no. 1804953.

[4] D. Coppersmith, "The data encryption standard (DES) and its strength against attacks," *IBM J. Res. Develop.*, vol. 38, no. 3, pp. 243–250, May 1994.

[5] Y. Moussa and W. Alexan, "Message security through AES and LSB embedding in edge detected pixels of 3D images," in *Proc. 2nd Novel Intell. Lead. Emerg. Sci. Conf. (NILES)*, Oct. 2020, pp. 224–229.

[6] H. T. Elshoush, R. M. Mohammed, M. T. Abdelhameed, and A. F. Mohammed, "Mitigating man-in-the-middle attack in online payment system transaction using polymorphic AES encryption algorithm," *J. Inf. Hiding Multimedia Signal Process.*, vol. 14, no. 3, pp. 102–112, 2023.

[7] W. Yihan and L. Yongzhen, "Improved design of des algorithm based on symmetric encryption algorithm," in *Proc. IEEE Int. Conf. Power Electron., Comput. Appl. (ICPECA)*, Jan. 2021, pp. 220–223.

[8] D. Jayasinghe, R. Ragel, J. A. Ambrose, A. Ignjatovic, and S. Parameswaran, "Advanced modes in AES: Are they safe from power analysis based side channel attacks?" in *Proc. IEEE 32nd Int. Conf. Comput. Design (ICCD)*, Oct. 2014, pp. 173–180.

[9] W. Alexan, H. Medhat, A. Hamza, and H. Hussein, "Sequence-based bit-cycling in double layer message security," in *Proc. Adv. Wireless Opt. Commun. (RTUWO)*, Nov. 2018, pp. 23–28.

[10] N. Adam, M. Mashaly, and W. Alexan, "A 3DES double–layer based message security scheme," in *Proc. 2nd Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS)*, May 2019, pp. 1–5.

[11] S. N. V. J. D. Kosuru, A. Pradhan, K. A. Basith, R. Sonar, and G. Swain, "Digital image steganography with error correction on extracted data," *IEEE Access*, vol. 11, pp. 80945–80957, 2023.

[12] M. T. Elkandoz and W. Alexan, "Image encryption based on a combination of multiple chaotic maps," *Multimedia Tools Appl.*, vol. 81, no. 18, pp. 25497–25518, Jul. 2022.

[13] W. Alexan, N. Alexan, and M. Gabr, "Multiple-layer image encryption utilizing fractional-order Chen hyperchaotic map and cryptographically secure PRNGs," *Fractal Fractional*, vol. 7, no. 4, p. 287, Mar. 2023.

[14] M. Gabr, Y. Korayem, Y.-L. Chen, P. L. Yee, C. S. Ku, and W. Alexan, "R^3—Rescale, rotate, and randomize: A novel image cryptosystem utilizing chaotic and hyper-chaotic systems," *IEEE Access*, vol. 11, pp. 119284–119312, 2023.

[15] J. Viega, M. Messier, and P. Chandra, *Network Security With OpenSSL: Cryptography for Secure Communications*. Sebastopol, CA, USA: O'Reilly Media, 2002.

[16] W. Alexan, Y.-L. Chen, L. Y. Por, and M. Gabr, "Hyperchaotic maps and the single neuron model: A novel framework for chaos-based image encryption," *Symmetry*, vol. 15, no. 5, p. 1081, May 2023.

[17] A. Dzhunkovskiy, "Prospects of using VR technologiesas a steganography medium," *Western Moscow State Univ. Linguistics. Humanitarian Sci.*, vol. 7, no. 823, pp. 155–166, 2019.

[18] K. Qi, D. Zhang, and D. Xie, "A high-capacity steganographic scheme for 3D point cloud models," *Inf. Technol. J.*, vol. 9, no. 3, pp. 412–421, Mar. 2010.

[19] K. Anish, N. Arpita, H. Nikhil, K. Sumant, S. Bhagya, and S. D. Desai, "Intelligence system security based on 3-D image," in *Proc. 5th Int. Conf. Frontiers Intell. Comput., Theory Appl.*, S. C. Satapathy, V. Bhateja, S. K. Udgata, and P. K. Pattnaik, Eds. Singapore: Springer, 2017, pp. 159–167.

[20] Z. Liu, W. Song, Y. Tian, S. Ji, Y. Sung, L. Wen, T. Zhang, L. Song, and A. Gozho, "VB-Net: Voxel-based broad learning network for 3D object classification," *Appl. Sci.*, vol. 10, no. 19, p. 6735, Sep. 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/19/6735

[21] J. Deng, S. Shi, P. Li, W. Zhou, Y. Zhang, and H. Li, "Voxel R-CNN: Towards high performance voxel-based 3D object detection," 2020, *arXiv:2012.15712*.

[22] N. Li, J. Hu, R. Sun, S. Wang, and Z. Luo, "A high-capacity 3D steganography algorithm with adjustable distortion," *IEEE Access*, vol. 5, pp. 24457–24466, 2017.

[23] M.-W. Chao, C.-H. Lin, C.-W. Yu, and T.-Y. Lee, "A high capacity 3D steganography algorithm," *IEEE Trans. Vis. Comput. Graphics*, vol. 15, no. 2, pp. 274–284, Mar. 2009.

[24] S. Farrag and W. Alexan, "Secure 3D data hiding technique based on a mesh traversal algorithm," *Multimedia Tools Appl.*, vol. 79, nos. 39–40, pp. 29289–29303, Oct. 2020.

[25] Y. Zhang, J. Zhu, M. Xue, X. Zhang, and X. Cao, "Adaptive 3D mesh steganography based on feature-preserving distortion," *IEEE Trans. Vis. Comput. Graphics*, vol. 30, no. 8, pp. 5299–5312, Aug. 2023.

[26] G. Mostafa and W. Alexan, "A robust high capacity gray code-based double layer security scheme for secure data embedding in 3D objects," *ITU J. Future Evolving Technol.*, vol. 3, no. 2, pp. 310–325, Sep. 2022.

[27] H. Zhou, K. Chen, W. Zhang, Y. Yao, and N. Yu, "Distortion design for secure adaptive 3-D mesh steganography," *IEEE Trans. Multimedia*, vol. 21, no. 6, pp. 1384–1398, Jun. 2019.

[28] R. Jiang, H. Zhou, W. Zhang, and N. Yu, "Reversible data hiding in encrypted three-dimensional mesh models," *IEEE Trans. Multimedia*, vol. 20, no. 1, pp. 55–67, Jan. 2018.

[29] P. Thiyagarajan, V. Natarajan, G. Aghila, V. P. Venkatesan, and R. Anitha, "Pattern based 3D image steganography," *3D Res.*, vol. 4, no. 1, pp. 1–8, Mar. 2013.

[30] Y.-Y. Tsai, "An adaptive steganographic algorithm for 3D polygonal models using vertex decimation," *Multimedia Tools Appl.*, vol. 69, no. 3, pp. 859–876, Apr. 2014.

[31] A. Girdhar and V. Kumar, ''A reversible and affine invariant 3D data hiding technique based on difference shifting and logistic map,'' *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 12, pp. 4947–4961, Dec. 2019.

[32] F. Cayre and B. Macq, ''Data hiding on 3-D triangle meshes,'' *IEEE Trans. Signal Process.*, vol. 51, no. 4, pp. 939–949, Apr. 2003.

[33] C. Wang and Y. Cheng, ''An efficient information hiding algorithm for polygon models,'' *Comput. Graph. Forum*, vol. 24, no. 3, pp. 591–600, Sep. 2005.

[34] M. T. Elkandoz, W. Alexan, and H. H. Hussein, ''3D image steganography using sine logistic map and 2D hyperchaotic map,'' in *Proc. Int. Conf. Electr. Comput. Technol. Appl. (ICECTA)*, Nov. 2019, pp. 1–6.

[35] Y. A. Hassan, ''Chaotic function secured message steganography in edge-detected pixels of 3D images,'' B.Sc thesis, German Univ. Cairo, New Cairo, Egypt, 2020.

[36] S. Yasser, A. Hesham, M. Hassan, and W. Alexan, ''AES-secured bit-cycling steganography in sliced 3D images,'' in *Proc. Int. Conf. Innov. Trends Commun. Comput. Eng. (ITCE)*, Feb. 2020, pp. 227–231.

[37] S. Elsherif, G. Mostafa, S. Farrag, and W. Alexan, ''Secure message embedding in 3D images,'' in *Proc. Int. Conf. Innov. Trends Comput. Eng. (ITCE)*, Feb. 2019, pp. 117–123.

[38] W. Alexan, M. El Beheiry, and O. Gamal-Eldin, ''A comparative study among different mathematical sequences in 3D image steganography,'' *Int. J. Comput. Digit. Syst.*, vol. 9, no. 4, pp. 545–552, Jul. 2020.

[39] A. S. Amin, ''StegoCrypt3D: 3D image slicing and blowfish,'' B.Sc thesis, German Univ. Cairo, New Cairo, Egypt, 2019.

[40] Y. Moussa, ''Advanced encryption standard secured message steganography in edge-detected pixels of 3D images,'' B.Sc thesis, German Univ. Cairo, New Cairo, Egypt, 2020.

[41] A. Alkhamese, H. ElGhawalby, A. Eid, I. Hanafy, and W. Awad, ''Highly secured 3D object steganography technique for hiding compressed data,'' *Alfarama J. Basic Appl. Sci.*, vol. 5, no. 2, pp. 208–220, 2024.

[42] G. Zhang, Z. Sui, C. Sun, Q. Liu, and X. Cheng, ''A multi-layer mesh synchronized reversible data hiding algorithm on the 3D model,'' *Multimedia Syst.*, vol. 30, no. 1, pp. 1–14, Feb. 2024.

[43] S. Bandyopadhyay, S. Mukherjee, S. Mukhopadhyay, and S. Sarkar, ''Parallel BFS through pennant data structure with reducer hyper-object based data hiding for 3D mesh images,'' *Secur. Privacy*, Mar. 2024, Art. no. e390, doi: 10.1002/spy2.390.

[44] S. Mukherjee, S. Sarkar, and S. Mukhopadhyay, ''VCI construction and shifting strategy based steganography for 3D images,'' in *Computational Intelligence in Communications and Bus. Analytics*, S. Mukhopadhyay, S. Sarkar, P. Dutta, J. K. Mandal, and S. Roy, Eds., Cham, Switzerland: Springer, 2022, pp. 210–219.

[45] K. Gao, J.-H. Horng, and C.-C. Chang, ''Reversible data hiding for encrypted 3D mesh models with secret sharing over Galois field,'' *IEEE Trans. Multimedia*, vol. 26, pp. 5499–5510, 2024.

[46] H. Lin, C. Wang, L. Cui, Y. Sun, C. Xu, and F. Yu, ''Brain-like initial-boosted hyperchaos and application in biomedical image encryption,'' *IEEE Trans. Ind. Informat.*, vol. 18, no. 12, pp. 8839–8850, Dec. 2022.

[47] M. Matsumoto and T. Nishimura, ''Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator,'' *ACM Trans. Model. Comput. Simul.*, vol. 8, no. 1, pp. 3–30, Jan. 1998, doi: 10.1145/272991.272995.

[48] K. Ding and Y. Tan, ''Comparison of random number generators in particle swarm optimization algorithm,'' in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2014, pp. 2664–2671.

[49] R. Simard, ''TestU01: AC library for empirical testing of random number generators,'' *Les Cahiers du GERAD*, 2006.

[50] M. Gabr, R. Elias, K. M. Hosny, G. A. Papakostas, and W. Alexan, ''Image encryption via base-n PRNGs and parallel base-n S-boxes,'' *IEEE Access*, vol. 11, pp. 85002–85030, 2023.

[51] Wolfram Research. (2024). *Wolfram Data Repository*. [Online]. Available: https://datarepository.wolframcloud.com/

[52] Y.-M. Cheng and C.-M. Wang, ''A high-capacity steganographic approach for 3D polygonal meshes,'' *Vis. Comput.*, vol. 22, nos. 9–11, pp. 845–855, Sep. 2006.

[53] C.-H. Chuang, C.-W. Cheng, and Z.-Y. Yen, ''Reversible data hiding with affine invariance for 3D models,'' in *Proc. IET Int. Conf. Frontier Comput., Theory, Technol. Appl.*, Aug. 2010, pp. 77–81.

[54] H. Liu, X. Wang, and A. Kadir, ''Chaos-based color image encryption using one-time keys and choquet fuzzy integral,'' *Int. J. Nonlinear Sci. Numer. Simul.*, vol. 15, no. 1, pp. 1–10, Feb. 2014.

[55] W. Meier and O. Staffelbach, ''Nonlinearity criteria for cryptographic functions,'' in *Proc. Workshop Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 1989, pp. 549–562.

[56] S. Hong, S. Lee, J. Lim, J. Sung, D. Cheon, and I. Cho, ''Provable security against differential and linear cryptanalysis for the SPN structure,'' in *Proc. Int. Workshop Fast Softw. Encryption*. Berlin, Germany: Springer, 2000, pp. 273–283.

[57] E. Biham and A. Shamir, ''Differential cryptanalysis of des-like cryptosystems,'' *J. Cryptol.*, vol. 4, no. 1, pp. 3–72, Jan. 1991.

[58] A. F. Webster and S. E. Tavares, ''On the design of S-boxes,'' in *Proc. Conf. Theory Appl. Cryptogr. Techn.* Berlin, Germany: Springer, 1985, pp. 523–534.

[59] A. H. Zahid, M. J. Arshad, and M. Ahmad, ''A novel construction of efficient substitution-boxes using cubic fractional transformation,'' *Entropy*, vol. 21, no. 3, p. 245, Mar. 2019.

[60] S. Deb and B. Bhuyan, ''Chaos-based medical image encryption scheme using special nonlinear filtering function based LFSR,'' *Multimedia Tools Appl.*, vol. 80, no. 13, pp. 19803–19826, May 2021. [Online]. Available: https://link.springer.com/10.1007/s11042-020-10308-7

[61] M. Gabr, H. Younis, M. Ibrahim, S. Alajmy, I. Khalid, E. Azab, R. Elias, and W. Alexan, ''Application of DNA coding, the Lorenz differential equations and a variation of the logistic map in a multi-stage cryptosystem,'' *Symmetry*, vol. 14, no. 12, p. 2559, Dec. 2022.

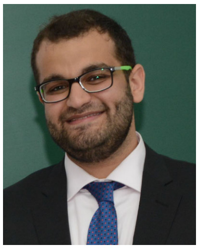
[62] W. Alexan, Y. Korayem, M. Gabr, M. El-Aasser, E. A. Maher, D. El-Damak, and A. Aboshousha, ''AntEater: When Arnold's cat meets Langton's ant to encrypt images,'' *IEEE Access*, vol. 11, pp. 106249–106276, 2023.

[63] W. Alexan, D. El-Damak, and M. Gabr, ''Image encryption based on Fourier-DNA coding for hyperchaotic Chen system, Chen-based binary quantization S-box, and variable-base modulo operation,'' *IEEE Access*, vol. 12, pp. 21092–21113, 2024.

MOHAMED GABR (Member, IEEE) was born in Cairo, Egypt, in 1989. He received the B.Sc., M.Sc., and Ph.D. degrees in computer science and engineering from German University in Cairo (GUC), Egypt, in 2011, 2013, and 2023, respectively.

He was with the Computer Science and Engineering Department, since 2011. He is teaching various courses in relation to computer vision, artificial intelligence, compilers, theory of computation, and computer graphics. He is the author or co-author of various journal articles and conference papers. His research interests include computer vision and information security.

Dr. Gabr received the Best Paper Award at the 26th IEEE Conference on Signal Processing Algorithms, Architectures, Arrangements and Applications, SPA'2023 in Poznan, Poland.



AMR DIAB was born in Cairo, Egypt, in 1992. He received the B.Sc. and M.Sc. degrees in computer science and engineering from German University in Cairo (GUC), Egypt, in 2014 and 2016, respectively. He worked on his B.Sc. thesis with the University of Ulm, Germany.

He was with the Computer Science and Engineering Department, from 2014 to 2018, and again since 2023. He is teaching various courses in relation to data structures, algorithms and programming concepts, advanced labs, embedded systems, theory of computation, and computer graphics. His research interest includes information security.

HUWAIDA T. ELSHOUSH (Senior Member, IEEE) received the bachelor's degree in computer science (division 1), the master's degree in computer science, and the Ph.D. degree in information security from the Faculty of Mathematical Sciences and Informatics, University of Khartoum, Sudan, in 1994, 2001, and 2012, respectively. Her M.Sc. dissertation dealt with Frame Relay Security.

She is currently an Associate Professor with the Computer Science Department, Faculty of Mathematical Sciences and Informatics, University of Khartoum, where she is also acting as the Head of Research office. Moreover, she is the Deputy Dean of basic sciences and engineering with the Graduate College of the University of Khartoum. She has more than 32 publications and some of her publications appeared in *Applied Soft Computing* (Elsevier), *PLOS One*, IEEE Access, *Multimedia Tools and Applications*, *PeerJ Computer Science*, *Journal of Information Hiding and Multimedia Signal Processing*, and Springer book chapters. Her research interests include information security, cryptography, steganography, and intrusion detection systems.

Dr. Elshoush received the second-place prize in the ACM Student Research Competition SRC–SAC, in 2013 in Coimbra, Portugal. Her article titled "An Improved Framework for Intrusion Alert Correlation" has received the Best Student Paper Award of the 2012 International Conference of Information Security and Internet Engineering (ICISIE) in WCE 2012, London. Other prizes were the Best Student during the five years of her undergraduate study. She is a Reviewer of many international reputable journals related to her fields, including *Applied Soft Computing* (Elsevier). Moreover, she acts as an external examiner for some international and national universities.

YEN-LIN CHEN (Senior Member, IEEE) received the B.S. and Ph.D. degrees in electrical and control engineering from the National Chiao Tung University, Hsinchu, Taiwan, in 2000 and 2006, respectively. From February 2007 to July 2009, he was an Assistant Professor with the Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan. From August 2009 to January 2012, he was an Assistant Professor with the Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei, Taiwan, where he was an Associate Professor, from February 2012 to July 2015. Since August 2015, he has been a Full Professor with the National Taipei University of Technology. His research interests include artificial intelligence, intelligent image analytics, embedded systems, pattern recognition, intelligent vehicles, and intelligent transportation systems. His research results have been published on over 100 journals and conference papers. He is a fellow of IET; and a member of ACM, IAPR, and IEICE.

LIP YEE POR (Senior Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees from Universiti Malaya, Malaysia. He is currently an Associate Professor with the Faculty of Computer Science and Information Technology, Universiti Malaya. His research interests include neural networks (such as supervised and unsupervised learning methods, such as support vector machine and extreme learning machine), bioinformatics (such as biosensors and pain research), computer security [such as information security, steganography, and authentication (graphical password)], grid computing, and e-learning framework.

CHIN SOON KU received the Ph.D. degree from Universiti Malaya, Malaysia, in 2019. He is currently an Assistant Professor with the Department of Computer Science, Universiti Tunku Abdul Rahman, Malaysia. His research interests include AI techniques (such as genetic algorithm), computer vision, decision support tools, graphical authentication (authentication, picture-based password, and graphical password), machine learning, deep learning, speech processing, natural language processing, and unmanned logistics fleets.

WASSIM ALEXAN (Senior Member, IEEE) was born in Alexandria, Egypt, in 1987. He received the B.Sc., M.Sc., and Ph.D. degrees in communications engineering, in 2010, 2012, 2017, respectively, the M.B.A. degree from German University in Cairo (GUC), Egypt, in 2019, and the M.A. degree in educational leadership from American University in Cairo (AUC), Egypt, in 2024.

He was with the Mathematics Department, from 2010 to 2017. Since 2017, he has been a member of the Faculty of Information Engineering and Technology, GUC, teaching various courses in relation to wireless communications, modulation and coding, information theory, digital logic design, circuit theory, and mathematics. In 2023, he was promoted to the academic rank of an Associate Professor of electrical engineering and information technology. He is also an Associate Professor with the Mathematics Department, German International University (GIU), New Administrative Capital, Egypt, since its inception in 2019. He is the author or co-author of more than 90 journal articles and conference papers. His research interests include wireless communications, information security, image and signal processing, mathematical modeling, and engineering education.

Dr. Alexan is a Professional Member of ACM. He received the Best Paper Award at the 19th and 26th IEEE Conferences on Signal Processing Algorithms, Architectures, Arrangements and Applications, (SPA'2015 and SPA'2023, respectively), Poznan, Poland; the AEG Writer of the Year Award from the American University in Cairo (AUC), Egypt, in 2019; and the Best Poster Award at the 37th IEEE National Radio Science Conference, Cairo, Egypt, in 2020. Moreover, he is a reviewer and a technical program committee member of various IEEE, IET, Springer, and Elsevier journals and international conferences. Currently, he serves as an Academic Editor for the journal *IET Computers and Digital Techniques*, and a Guest Editor to the Springer Link journal *Discover Imaging*.