

# A ROBUST HIGH CAPACITY GRAY CODE-BASED DOUBLE LAYER SECURITY SCHEME FOR SECURE DATA EMBEDDING IN 3D OBJECTS

Ghadir Mostafa<sup>1</sup> and Wassim Alexan<sup>2</sup>

<sup>1,2</sup>Faculty of Information Engineering and Technology, The German University in Cairo

NOTE: Corresponding author: Wassim Alexan, [wassim.alexan@ieee.org](mailto:wassim.alexan@ieee.org)

**Abstract** – In this study, we propose a novel high capacity double layer algorithm for secure data embedding in 3D objects. This is achieved by aggregating a cryptography layer through the deployment of Blowfish or AES-128 algorithms to a steganography layer based on a Gray code sequence that individualizes the order of the vertices over which the embedding will occur. Thereafter, the 3D objects are preprocessed and the secret data is embedded over the vertices'  $x$ -,  $y$ -, and  $z$ -coordinates. Hence, the 3D object capacity is effectively utilized. The secret data is then blindly extracted from the stego 3D object. The performance of the proposed algorithm is extensively investigated and compared to other commensurate studies from the literature. The proposed algorithm withstands vertex reordering and common geometrical similarity attacks such as reflection, uniform scaling, rotation and translation. Additionally, it partially withstands smoothing. The achieved numerical results demonstrate the superiority of the proposed algorithm in terms of capacity, computational complexity, imperceptibility, distortion and robustness.

**Keywords** – 3D objects, AES-128, Blowfish, Gray codes, steganography

## 1. INTRODUCTION

The quantum leap in digital communications comes hand in hand with a dramatic growth in cloud, computational power, storage capacities and security approaches [1, 2]. This has facilitated the usage of digital media such as images, audio and video as a cover in order to hide data even when being transmitted over insecure channels. Accordingly, data security has become a captivating field of research in order to keep pace with this advancement [3, 4]. Of particular interest is the exponential increase in the security needs of sensitive data that is generated from military, industrial or medical applications [5, 6]. This has switched data scientists' and global users' interests to become security-based rather than just communication-based in order to ensure the secrecy of the transmission of confidential data.

Information encryption which is also known as cryptography has been globally used in communication. Typically, a key is required in order to convert the message into a graspable form. This key is only accessible by the receiver and hence no one else has access to the data. However, the problem with encryption algorithms is that they convert plain text into cipher text which is nonsensical [7, 8]. This draws the attention of eavesdroppers that there is a chance that some data with significant importance is being transmitted and they may even interpose some undesired content. In contrast, information hiding can be applied by two methods, namely, watermarking and steganography which are dissimilar in terms of the embedding capacity and the purpose. Furthermore, watermarking is considered to offer a smaller embedding capacity when compared to steganography and its main

purpose is to hide data in a sturdy method, mainly for copyright protection [9]. Conversely, steganography which is the science of data hiding over a digital medium [10] overcomes the obviousness caused by cryptography because the secret data can be hidden into a digital cover medium which can be a 2D [11, 12, 13] or a 3D image [14, 2, 4, 15], a video [16], an audio file [17, 18, 19] or maybe even a digital document [20] or an information matrix [21]. This provides a larger embedding capacity under the main purpose that the secret data is kept inaccessible.

When the cover file is an image, steganography can be referred to as image steganography which consists of two major phases, namely embedding and extraction. The embedding phase requires three inputs, the cover medium, the embedding algorithm and the secret message to produce a stego object. The success of an embedding algorithm is measured by three important factors, namely, the embedding capacity, the similarity between cover and stego media and the data extraction complexity. The stego object is then transmitted over any channel. Upon arrival to its destination, the extraction phase requires the stego object, the extraction algorithm and might also require the cover image. There are two main categorizations for a steganography system. First, a blind system that does not require the cover image for the data extraction algorithm. Second, a reversible system which on the other hand, denotes a system which can perfectly reconstruct the cover image from the stego image after extracting the embedded data.

The main reason behind the utilization of 3D objects over 2D images is the huge leap in payload capacity increase offered by 3D objects. Thus, recent years have

seen a surge in the literature on 3D object steganography [10]. Three dimensional image steganography can be applied over two domains, namely spatial and frequency [22]. However, most of the research has targeted the spatial domain due to two main reasons. First, the spatial domain has been proven to render a larger embedding capacity. In addition to that, converting to and from the frequency domain adds more complexity to the embedding algorithm. Furthermore, steganography can be applied in the spatial domain over one of three different approaches, namely topological, representational and geometrical.

In geometry-based steganography, the embedding stratagem is applied on geometrical components of 3D cover models such as vertices, edges and polygons in order to hide the secret data [23]. However, embedding in geometrical components is vulnerable to affine transformations such as scaling and rotation which might affect the hidden secret data. This makes it challenging for the steganography system to hold the line against them. Thiyagarajan *et al.* embedded data on a re-triangulated region of a triangular mesh. The algorithm was proved to withstand affine translation and cropping. Moreover, Anish *et al.* [24] proposed a simple steganography scheme in 3D images by embedding data as a fractional value of the  $x$ - $z$  coordinates of the 3D image vertices. In [4], Farrag *et al.* proposed a reversible and secure algorithm for data embedding in 3D mesh models by deploying a mesh traversal algorithm between neighboring vertices according to the shortest distances between them. The encrypted data bits are then embedded over the fourth and fifth decimal places of the Cartesian coordinate vertices. Li *et al.* in [25] carry out sensitive data embedding in vertices of a 3D mesh by following a Hamiltonian path. Vertices on the path are modulated by making changes to three coordinates in the Spherical Coordinate System (SCS). Performance analysis of their proposed algorithm showcase an increased resistance against steganalysis efforts. Zhou *et al.* in [26] propose a high capacity steganography technique that is also robust against steganalysis. Their proposed technique is adaptive in nature and depends on the utilization of vertex normal to make a decision of data embedding.

In topology-based steganography, algorithms are based on the adjustment of the connectivity or topological attributes of a 3D model cover medium to allow for embedding [27]. This unfortunately allows for a limited embedding capacity when compared to other spatial domain schemes. Tsai [28] proposed a blind reversible topological steganography scheme using recursive triangle subdivision which was proven to withstand affine translation. However, blind extraction fails when the stego object is subject to noise. In [29], the authors propose a semi-fragile, blind watermarking algorithm with the aim of substantiating the authenticity of 3D

models. Their proposed algorithm initiates by traversing the 3D model and deciding on a number of verification units. Each of these units is made up of a set of eligible vertices for embedding, as well as a vertex for the verification code. The embedding of the watermark is carried out through modulating the the spherical angular values of each of the embedding vertices. Performance is measured in terms of the distortion introduced through modulation and is shown to be minimal with regional attack localization.

In representation-based steganography, data embedding algorithms utilize the redundancy in mesh representations. However, the geometry and mesh connectivity are kept plenary [30]. Accordingly, this method is distortion free [31]. In [30], Cheng *et al.* proposed a Representation Rearrangement Procedure (RRP) in the representation domain where the representation order of the vertices or the polygons and their topology information can be represented with an average of six bits per vertex. Moreover, in [32], Lin *et al.* proposed an embedding algorithm that operates by permuting/rearranging vertex representation orders, triangle representation orders, and connectivity information, without affecting the imperceptibility of the cover media or introducing a visual distortion.

Least Significant Bit (LSB) embedding is one of the most readily used algorithms in the spatial domain image steganography. In LSB, the least significant bits of one or more of the colour channels of the pixels in a 2D image medium are used for data embedding [33]. This leads to a minor change in the brightness, contrast or colour intensity of the pixel which cannot be recognized by the human eye. With recent advancement in technology towards 3D modelling, 3D images and objects have become imperceptible when used as a cover of secret digital content due to their recurring utilization in innumerable applications such as animation, TV, video games and web design [34]. Hence, steganography has been recently integrated into 3D data, because it involves more complex and variable topological and geometrical operations which allow for a larger embedding capacity and a variety of embedding algorithms.

The main contributions of this paper can be summarized into the following:

- The proposed algorithm consists of two security layers.
  - A cryptographic layer, where a symmetric encryption algorithm is utilized. This is either AES-128 or Blowfish, in order to achieve a double layer security for sensitive data.
  - A steganographic layer, where a Gray code sequence is utilized in order to distinguish the vertices over which the data will be embedded.

A Gray code is chosen due to its simple hardware implementation and superior ability at facilitating error-correction in digital communications.

- A huge embedding capacity is attained by grouping each set of 8 bits into a decimal number ranging from 0 to 255 which is then embedded over the 6<sup>th</sup>, 7<sup>th</sup> and 8<sup>th</sup> least significant decimal points of each vertex coordinate and thus each vertex can hold up to 24 bits of information without affecting the visual imperceptibility of the 3D cover object. After the Gray code sequence specifies the order over which the data will be embedded, all the vertices with indices that do not appear in the sequence are still utilized for sequential embedding.
- An extensive number of performance metrics are applied in order to evaluate the efficiency, suitability to real-time applications, and robustness of the proposed algorithm. These are measured in terms of the embedding capacity, execution time for embedding and extraction, imperceptibility and distortion. Additionally, the proposed algorithm is tested against some of the prevalent attacks such as noise, vertex reordering, smoothing and similarity transforms.

The rest of this paper is organized as follows. In Section 2, the Gray code sequence based proposed scheme is illustrated and the embedding and extraction strategies are described. In Section 3, the numerical results and proposed scheme evaluation are presented to assess its performance based on the defined metrics. Next, Section 4 compares the proposed double layer security scheme to its counterparts from the literature. Finally, Section 5 presents the conclusions and suggests some future research directions.

## 2. PROPOSED DOUBLE LAYER MESSAGE SECURITY SCHEME

In this section, the proposed double layer blind reversible security scheme is outlined. Our proposed scheme is composed of an encryption layer followed by a 3D image steganography layer. To begin with, the secret message is converted into ASCII code and then to its corresponding binary representation. Furthermore, a binary symmetric key is then generated according to the encryption algorithm which will be applied and then used to encrypt the resultant binary stream.

The two algorithms utilized for encryption are AES-128 and Blowfish. AES-128 is a symmetric block cipher which uses the same key for encryption and decryption. It was first established by the United States National Institute of Standards and Technology (NIST), after the Data Encryption Standard (DES) algorithm was proven to be vulnerable to brute force attacks in 1997. The

**Table 1** – Numbers 0 to 15 in decimal, binary and Gray coded binary representations.

Decimal	Binary	Gray coded binary
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

AES encryption algorithm encrypts and decrypts data in blocks of 128 bits. This is carried out using 128-bit, 192-bit, or 256-bit keys. AES using 128-bit keys is often referred to as AES-128 [35, 36, 37].

On the other side, the Blowfish algorithm is a symmetric block cipher that encrypts data in 8-byte (64-bit) blocks. The algorithm runs over two phases, namely key expansion and data encryption. Key expansion consists of generating the initial P-array composed of eighteen 32-bit sub-keys, and 4 S-boxes, each of size 256 by 32 bits, from a key of a maximum of 448 bits (56 bytes). The data encryption uses a 16-round Feistel Network [38, 39].

The embedding algorithm operates in the sequence presented in Fig. 1. To begin with, the vertices over which the secret data will be embedded are then chosen based on a Gray code sequence, constructed using a length of 24 bits. Gray codes, or reflected binary codes, are binary numbers constructed using any number of bits resulting in a unique order in which the difference between any two consecutive numbers is a single bit position [40]. To convert a binary number to a Gray code number, the most significant or the leftmost bit of the Gray code is identical to that of the binary code. Moreover, the second significant bit of the Gray code number is calculated by applying an XOR operation between the first and the second binary bits. Similarly, the third Gray code bit is achieved by repeating the XOR operation for the second and third binary bits and so on, as depicted by Fig. 1 in [41]. For example, if the binary number is 10111001, its corresponding Gray code number is 11100101. Furthermore, Table 1 provides Gray codes corresponding to the first few non-negative integers [42]. The reason behind the choice of a Gray code is because of its simple

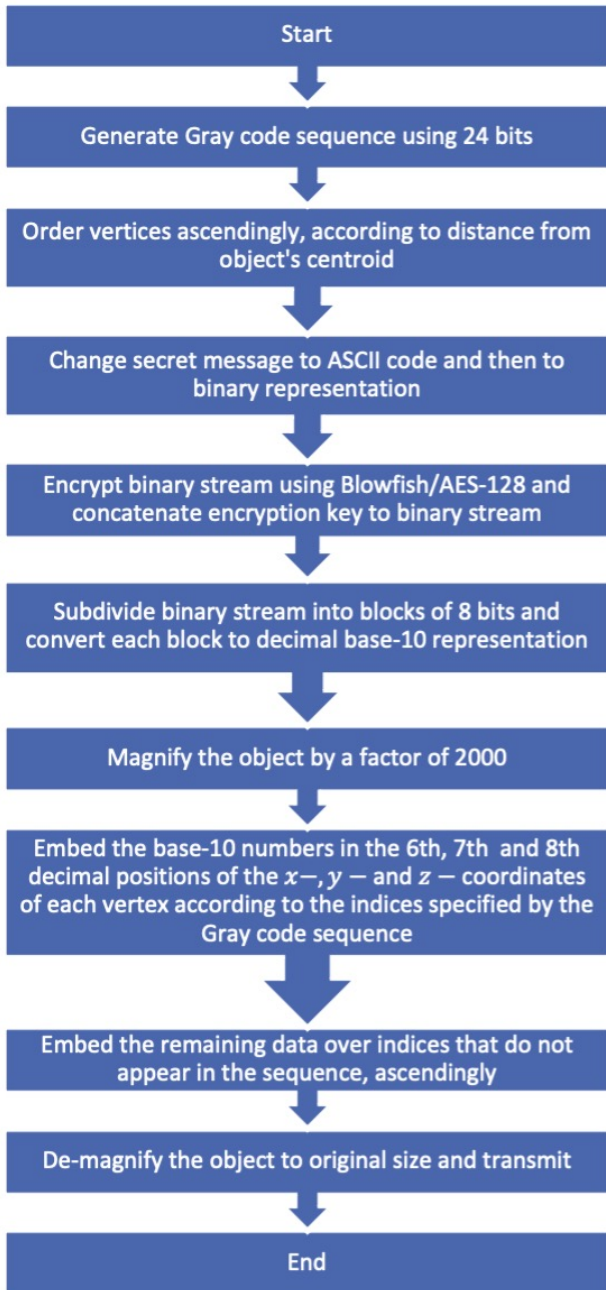


Fig. 1 – A flowchart for the embedding algorithm of the proposed scheme.

and efficient hardware implementation [43] as well as its superior ability at facilitating error-correction in digital communications [44, 45, 46]. Such advantages has lead to Gray codes resurfacing in recent literature of secure communications [11, 47, 48].

In the proposed algorithm, the Gray code sequence numbers constructed using 24 bits are then converted to decimal (base-10) numbers. The generated sequence of decimal numbers is denoted by  $N_{gc}$  and used as the indices of the vertices over which the data embedding occurs. The 3D object will be first imported as a .NOFF or .OBJ format file and the vertices used to construct this 3D object will be extracted from its geometric data. We then begin by ordering the vertices of the 3D cover in an ascending order of their distances from the object's centroid. The number of vertices of the 3D object are denoted by  $N_v$ . However, the constructed sequence includes indices which exceed the number of the cover vertices and thus they are discarded. Additionally, to achieve a 100% embedding capacity, the indices of the vertices which do not appear in the sequence are then used for embedding, sequentially in an ascending order.

The secret message to be embedded is then converted to ASCII code and then to its 8-bit binary representation to produce a stream of binary bits denoted by  $s_m$ . This stream is then encrypted using either AES-128 or Blowfish to produced a binary encrypted stream denoted by  $s_{enc}$ . The initialization vector (Encryption key) is then concatenated to  $s_{enc}$  to produce a binary stream  $s_{bin}$  of length  $l$ . Next,  $s_{bin}$  is subdivided into blocks of length 8 bits each. Moreover, each block is converted to its decimal representation which is a number ranging from 0, in case of  $(00000000)_2$ , to 255, in case of  $(11111111)_2$ , to form the new stream  $s_{block}$  of length  $l/8$ . Accordingly, having the embedding stream ready, the proposed scheme is based on embedding three elements of  $s_{block}$  stream per vertex. However, each element of  $s_{block}$  originally consists of 8 bits of the secret message binary representation. Hence, a total of 24 bits or 3 secret message ASCII characters are embedded per vertex.

Since most of the 3D models consist of multiple significantly small coordinate points (in the order of  $10^{-9}$ ), we must avoid the introduction of detectable dissimilarities between both the cover and the 3D stego objects. Hence, the 3D object is first scaled by a magnifying factor,  $\delta$ , in order to avoid embedding over decimal positions that are considered significant according to these minimal values. In fact, we used  $\delta = 20,000$  to have a fair comparison between all the 3D models, for the same factor. However, this factor can be modified according to the minimum coordinate value of each 3D cover object to achieve optimal results. Next, the vertices'  $x$ -,  $y$ - and  $z$ -coordinates of the scaled 3D object are truncated to only 5 significant decimal places to allow for embedding over the 6<sup>th</sup>, 7<sup>th</sup> and 8<sup>th</sup> decimal places of the scaled 3D object. This is

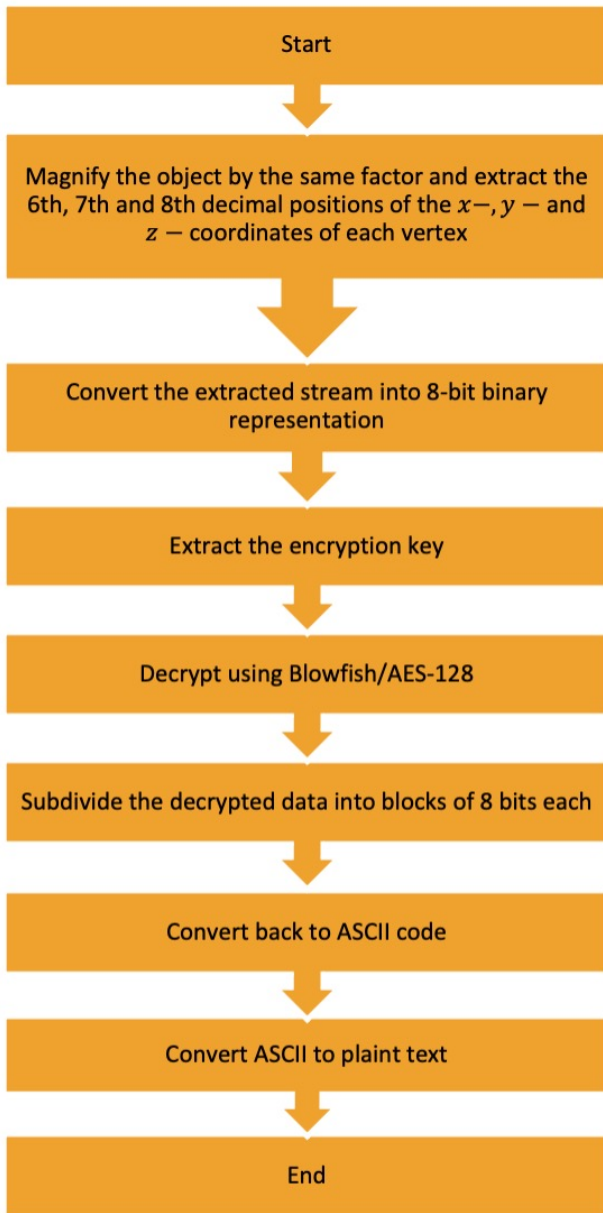


Fig. 2 – A flowchart for the extraction algorithm of the proposed scheme.

because we apply a similar strategy to the LSB algorithm in which data is embedded over the least significant bit except that we use the least significant decimal places in our case.













The length of the secret data required to be embedded specifies two principal factors namely, the choice of the 3D object and the number of vertices over which the data will be embedded. Table 2 shows some of the famous experimental 3D models from the Stanford Graphics Laboratory [49] with various sizes. More complex 3D objects correspond to a larger number of vertices and accordingly, a higher embedding capacity. In addition to that, the number of coordinates over which the data will be embedded can be varied from 1 to 3 coordinates per vertex, according to the message length. In our scheme, we allow for data embedding over all the coordinates of all vertices to allow for a maximum embedding capacity.

Once the embedding is carried out, the stego object is re-constructed using the modified vertices and down-scaled by the same factor used for magnifying in order to cancel its effect. It is then transmitted to the receiver side. Upon its delivery, the extraction algorithm proposed illustrated in Fig. 2 is applied. The receiver first magnifies the object again using the same aforementioned scale factor  $\delta$  and extracts the vertices again and then the corresponding 6<sup>th</sup>, 7<sup>th</sup> and 8<sup>th</sup> decimal places of each coordinate of all vertices are then extracted in the same order of the Gray code sequence used for embedding the data. These extracted values correspond to the block values which must then be converted to their 8 bit binary representation. The retrieved binary stream is then subdivided into two streams, which are the data to be decrypted and the initialization vector or the key used for decryption which are the last 128 bits or 256 bits for AES-128 or Blowfish, respectively. The two streams are then passed over the corresponding decryption algorithm to finally retrieve the decrypted binary stream which is divided into blocks of size 8 bits each. These blocks are then converted back to their corresponding ASCII code and finally to the plain text secret message.

### 3. NUMERICAL RESULTS AND PROPOSED SCHEME EVALUATION

In this section, the performance of the proposed double layer message security scheme is evaluated and compared to its counterparts from the literature to review its competence and imperceptibility. The implementation is carried out using Wolfram Mathematica® on a machine running a 64-bit operating system with 16 GB of RAM and an Intel® Core™ i7-6770 HQ CPU with a maximum clock rate of 2.60 GHz.

**Table 2** – Number of vertices of the experimental 3D models.

Name	3D model	Number of vertices
Cow		2903
Elephant		19,753
Bunny		34,834
Hand		36,616
Horse		48,485
Rabbit		70,658
Venus		100,759
Armadillo		172,974
Lion		183,408
Dragon		435,545
Buddha		543,524
Old man		1,243,138

### 3.1 Embedding capacity

Embedding capacity is a key parameter in the evaluation of steganographic algorithms as it measures the amount of secret data that can be concealed in a digital cover medium without a suspicious distortion. To judge the embedding capacity of our proposed scheme, we computed the capacity of multiple 3D models with different encryption algorithms and without encryption as shown in Table 3. The embedding capacity ranges from 69,672 to 29,835,312 bits according to the number of vertices of the 3D cover object used and whether or not an encryption algorithm is deployed. AES-128 shows the minimum embedding capacity due to the redundant bits added upon encryption, based on the fact that each block of 16 bits is encrypted into a block of 128 bits and hence the amount of actual data embedded is less than that achieved when utilizing Blowfish or no encryption at all [50].

**Table 3** – Maximum achieved embedding capacity with encryption utilizing AES-128 or Blowfish or no encryption at all.

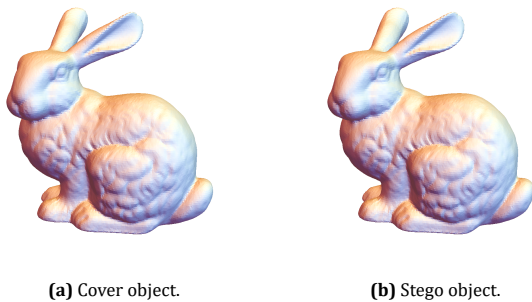
3D model	Embedding capacity [bits]		
	AES-128	Blowfish	No encryption
Cow	69,496	69,560	69,672
Elephant	473,848	473,976	474,072
Bunny	835,832	835,960	836,040
Hand	878,584	878,712	878,784
Horse	1,163,384	1,163,512	1,163,640
Rabbit	1,695,608	1,695,672	1,695,792
Venus	2,418,040	2,418,104	2,418,216
Armadillo	4,151,160	4,151,288	4,151,376
Lion	4,401,656	4,401,720	4,401,792
Dragon	10,452,856	10,452,984	10,453,080
Buddha	13,044,344	13,044,472	13,044,576
Old man	29,835,128	29,835,200	29,835,312

**Table 4** – Embedding and extraction execution times for the experimental 3D models, without encryption (maximum embedding capacity).

3D model	Embedded bits	Execution Times [s]	
		Embedding	Extraction
Cow	69,672	0.1262	0.0533
Elephant	474,072	0.8162	0.2803
Bunny	836,040	1.5900	0.6019
Hand	878,784	1.5933	0.6601
Horse	1,163,640	4.3910	1.4840
Rabbit	1,695,792	2.3999	0.9000
Venus	2,418,216	4.1880	1.4800
Armadillo	4,151,376	7.1100	2.2970
Lion	4,401,792	7.2350	2.4690
Dragon	10,453,080	18.7586	4.5028
Buddha	13,044,576	22.2280	6.2970
Old man	29,835,312	54.8423	16.5071

**Table 5** – Average embedding time and bits per vertex (bpv) comparison of the proposed scheme and its counterparts from the literature.

	Proposed scheme	Farrag and Alexan [4]	Zhu <i>et al.</i> [71]	Chao <i>et al.</i> [72]	Zhou <i>et al.</i> [26]	Li <i>et al.</i> [25]	Anish <i>et al.</i> [24]
bpv	24	6	1.5	1.5	1.5	1.5	4
Time [s]	$4.5 \times 10^{-5}$	$4.35 \times 10^{-5}$	0.944	0.1238	0.1412	1.9212	$1.41 \times 10^{-4}$
Year	2022	2020	2021	2008	2018	2017	2017


**Fig. 3** – The Stanford Bunny before and after embedding 836,040 bits.

### 3.2 Execution time analysis

The embedding and extraction execution times are evaluated for the maximum embedding capacity that each 3D object can confine. As the results in Table 3 exemplify, our proposed algorithm runs in a reasonable time, for embedding and extracting an effectual amount of secret data bits ranging from 69,672 to 29,835,312 bits with an embedding time of 0.1262s to 54.8423s and an extraction time of 0.0533s to 16.5071s, respectively. Table 5 provides an average embedding time and bits per vertex (bpv) comparison of the proposed scheme against its counterparts from the literature. It is clear that the proposed scheme not only exhibits superiority in terms of the maximum bpv, but also it achieves so in the least time. Note that while the scheme proposed by Farrag and Alexan [4] carries out the embedding in a seemingly shorter time (a fraction of a microsecond), it does so for a lower value of bpv.

### 3.3 Imperceptibility

Imperceptibility is a measure of whether the Human Visual System (HVS) can distinguish the differences between the cover and the stego models. An efficient data embedding algorithm should not introduce any dubious alterations to the cover model's appearance. Fig. 3 shows the cover and the stego Stanford Bunny 3D object after embedding 836,040 data bits, without any visual dissimilarities. To evaluate the imperceptibility of our proposed scheme, we use some of the full reference objective quality measurement parameters in [51] and [52] such as the Normalized Absolute Error (NAE) and the Structural Similarity Index Measurement (SSIM). Regarding the SSIM values [53], our achieved values can all be approximated to 1 and thus the visual difference between the cover and the stego objects is nearly negligible. Our achieved val-

ues of NAE are within the order of  $10^{-8}$  which is considered superior according to the values reported in [54], the MHIST- $k$  algorithm in [55] and [24] which have an average of 0.02452,  $8.75556 \times 10^{-5}$  and 0.01905, respectively. The MHIST- $k$  algorithm finds the best rectangular partition of a  $p$ -dimensional description space using  $k$  non-overlapping  $p$ -dimensional rectangular regions. Moreover, we also evaluate one of the famous full reference distance based measures, which is the Normalized Hausdorff Distance (NHD). Mathematically, the Hausdorff Distance (HD) is defined as:

$$HD = \max_{a \in A} \min_{b \in B} \|a - b\|. \quad (1)$$

This can be translated to the measurement of the distance between each point in  $A$  to each point in  $B$  and taking the minimum or the closest point in  $B$  to each point in  $A$ . In (1),  $A$  and  $B$  denote the set of vertices of the stego and cover 3D objects, respectively [56]. Once this is computed for each point in  $A$ , the maximum of these distances corresponds to the Hausdorff distance. Hence, this is an effective quality measure when comparing two 3D models, as it measures the maximum difference between the two sets of vertices. Then, the NHD can be computed as:

$$NHD = \frac{HD}{d}, \quad (2)$$

where  $d$  is the normalization coefficient that corresponds to the maximum of all distances between the 3D model vertices to its centroid. Our reported values for  $d$  and the NHD are recorded in Table 6. According to [57], the distortion caused by steganography is acceptable if the NHD is approximately  $1 \times 10^{-4}$ . This criteria is satisfied by all our experimental 3D models, for the maximum embedding capacity shown in Table 6.

### 3.4 Peak signal to noise ratio, mean square error and distortion

The Peak Signal to Noise Ratio (PSNR) is a measure of the transparency of the stego object because the higher the PSNR value, the better the stego object has been reconstructed to match the cover object. Hence, the better is the embedding algorithm. This is because our main aim is to minimize the Mean Square Error (MSE) between the cover and the stego objects relative to the maximum signal value of the reference cover object. The MSE can be calculated as in [58], using:

**Table 6** – Imperceptibility measures, PSNR and MSE.

3D Model	Embedded Bits	$d$	$D_{max}$	SSIM	NAE $\times 10^{-8}$	NHD $\times 10^{-9}$	PSNR [dB]	MSE $\times 10^{-12}$
Cow	69,672	1.0196	2.0328	0.9999999999	0.3037	0.4746	128.7426	0.5520
Elephant	474,072	28.6233	60	1.0000000000	0.0398	0.0171	227.6207	$6.2263 \times 10^{-8}$
Bunny	836,040	0.2400	0.2502	0.9999999994	0.8069	2.0162	108.5830	0.8678
Hand	878,784	1.0419	2.1580	0.9999999999	0.7839	0.4644	140.8718	0.0381
Horse	1,163,640	0.1310	0.2530	0.9999999984	4.5075	3.7639	105.5129	1.7992
Rabbit	1,695,792	0.9994	1.8235	0.9999999999	0.6052	0.4938	132.3101	0.1953
Venus	2,418,216	1.0133	2.7539	0.9999999951	6.7423	0.4776	136.8576	0.1564
Armadillo	4,151,376	129.0950	228.8040	1.0000000000	0.0020	0.0037	239.3383	$6.0968 \times 10^{-8}$
Lion	4,401,792	0.2517	0.2850	0.9999999997	0.5767	1.9452	118.9854	0.1026
Dragon	10,453,080	0.2494	0.2669	0.9999999997	0.6807	1.9616	110.5912	0.6217
Buddha	13,044,576	0.2606	0.2290	0.9999999998	0.0296	0.1773	137.4833	0.1753
Old man	29,835,312	2.7884	3.1337	0.9999999999	0.020	0.716	127.045	0.019

$$\begin{aligned}
 \text{MSE} &= \frac{1}{|V|} \sum_{i=1}^{|V|} (c_i - s_i)^2 \\
 &= \frac{1}{3|V|} \sum_{i=1}^{|V|} \sum_{j=1}^3 (cv_{ij} - sv_{ij})^2,
 \end{aligned} \tag{3}$$

where the error is defined as the distances between each cover object vertex  $c_i$  and its corresponding stego object vertex  $s_i$ . This can also be reformulated according to the projected distances on the  $x$ -,  $y$ -, and  $z$ -components of each vertex, denoted by  $(cv_{ij} - sv_{ij})$  for  $1 \leq j \leq 3$ . Accordingly, the PSNR is defined as in [59] by:

$$\text{PSNR} = 20 \log_{10} \left[ \frac{D_{max}}{\sqrt{\text{MSE}}} \right], \tag{4}$$

where  $D_{max}$  is the diagonal distance of the smallest oriented cuboid bounding box of the 3D cover model, recorded in Table 6 for the experimented 3D objects. Additionally, Table 6 also shows that all the experimented 3D models achieve a PSNR value higher than 100 dB which is an indication that our model can confine a significant amount of data with minimal MSE values. It can be discerned that the results of two of the 3D models, namely Armadillo and Elephant show significantly improved results compared to the other experimental 3D models. This can be justified by the fact that both of these 3D models are composed of relatively larger vertices' coordinates values when compared to the rest. Accordingly, the scaling factor of 20,000 has led to embedding over insignificant decimal positions relative to the scaled coordinates'

values. This is further confirmed by the MSE and PSNR values in Table 6 and also the distortion values reported in Table 10. The distortion of the proposed scheme which is defined using the normalized root mean square error (NRMSE) is given in [60] by:

$$\text{NRMSE} = \frac{\sqrt{\text{MSE}}}{d}. \tag{5}$$

#### 4. PERFORMANCE EVALUATION AND COMPARATIVE NUMERICAL ANALYSIS

In order to evaluate the feasibility of the proposed algorithm, it has been compared to different 3D image steganography schemes from the literature of which some are cited in [10], where data is embedded over the spatial domain or the geometrical domain in specific. For an equitable comparison, we embedded the same amount of data bits experimented in the literature to achieve comparable results to those reported in the previously proposed algorithms that we include in the following comparisons. To begin with, the results reported in Table 7 show that the proposed scheme has a very high embedding capacity of 24 bits per vertex compared to the other algorithms in [61, 57, 62], [4] and [24] which can only embed 0.7, 3, 0.5, 6 and 8 bits per vertex, respectively. Hence, the maximum achieved capacity of our scheme escalated from 69,672 bits to 29,835,312 bits for 3D models with 2903 and 1,243,138 vertices, respectively.



**Table 7** – A comparison of the embedding capacity in the geometrical domain.

Year	Authors	Algorithm	Reversible	Blind	Embedding location	Number of bits embedded per vertex	Maximum achieved capacity in bits
2010	Chuang <i>et al.</i> [61]	embedding using histogram shifting	yes	yes	vertices	0.7	5879
2013	Thiyagarajan <i>et al.</i> [57]	embedding after triangle mesh formation	no	yes	vertices	3	90,800
2015	Huang and Tsai [62]	embedding based on histogram shifting	yes	yes	vertices	0.5	4,101,995
2017	Anish <i>et al.</i> [24]	embedding in the $x$ -coordinate of vertex	no	yes	vertices	8	35,784
2020	Farrag and Alexan [4]	embedding in a mesh traversal algorithm	yes	yes	vertices	6	5,399,000
2022	Proposed scheme	embedding in the $x$ -, $y$ - and $z$ -coordinates of vertex	yes	yes	vertices	24	29,835,312

In addition to that, only the schemes proposed in [57] and [61] add a security layer through the usage of a simple encryption/decryption key. In contrast, our proposed scheme utilizes either AES-128 or Blowfish encryption algorithms, each of which provides a prominent security layer. With regard to reversibility, only the schemes proposed by [61] and [62] in addition to ours can retrieve the 3D cover object through the stego object solely. Apropos secret message blind extraction, all the five schemes effectively extract the embedded bits from the 3D stego object without any reference to the cover object.

Table 8 compares our proposed algorithm to that proposed in [57]. Our reported results for NHD, MSE and accordingly PSNR show a significant performance improvement since our MSE values are considered negligible in comparison to those achieved in [57]. For example, the MSE achieved for the Stanford Bunny after embedding 64,496 bits using the scheme in [57] is approximately  $10^{17}$  times larger than our reported MSE value for the same number of embedded bits. Moreover,

our calculated value for the PSNR is approximately 3.5 times larger for the same 3D object. Finally, the NHD achieved by our scheme is approximately 0.04% of their attained value. Similar enhancements also apply for the Elephant and the Dragon 3D objects.

In Table 9, the percentage of 3D cover model vertices required for embedding 14,841 and 20,060 bits using our proposed scheme for the Bunny and the Horse cover models, respectively, is compared with those acquired in [63] and [57]. For both models, our proposed algorithm preserves approximately 98.2% of the object's vertices without any distortion and hence the quality of the stego model remains approximately identical to the cover model due to the originality retention of a huge percentage of the vertices' values. This retention value of 98.2%, shown in Table 9, clearly outperforms those reported in [57] (86.47 for the Bunny and 63.17 for the Horse) and [63] (58.8 for the Bunny and 58.5 for the Horse).

**Table 8** – PSNR, MSE and NHD comparison between [57] and the proposed algorithm.

3D Model	Embedded Bits	PSNR [dB]		MSE		NHD	
		[57]	Proposed	[57]	Proposed	[57]	Proposed
Elephant	30,736	58.3168	239.5143	0.09368	$4.0260 \times 10^{-21}$	$1.352 \times 10^{-6}$	$1.6892 \times 10^{-11}$
Bunny	64,496	55.3442	191.1259	0.18930	$4.8322 \times 10^{-21}$	$4.12 \times 10^{-6}$	$2.0160 \times 10^{-9}$
Dragon	90,800	55.702	201.8825	0.160369	$4.6181 \times 10^{-22}$	$3.25 \times 10^{-6}$	$1.9397 \times 10^{-9}$

**Table 9** – Capacity comparison between the proposed algorithm, Thiyagarajan *et al.* [57] and Agarwal *et al.* [63].

3D Model	Embedded Bits	% vertices used for emb.			% originality retention in stego object		
		[63]	[57]	Proposed	[63]	[57]	Proposed
Bunny	14841	41.2%	13.53%	1.78%	58.80%	86.47%	98.22%
Horse	20060	41.5%	36.83%	1.72%	58.50%	63.17%	98.28%

**Table 10** – Comparison of the distortion imposed by [61], [64], [62] and the proposed algorithm according to the embedding capacity in [62], measured in bits per vertex (bpv).

3D Model	Emb. capacity (bpv)	Emb. bits	Distortion			
			[61]	[64]	[62]	Proposed
Venus	0.093	9371	0.0005%	0.1260%	0.0266%	$1.5322 \times 10^{-9}\%$
Armadillo	0.158	37,330	0.0001%	0.1670%	0.0106%	$1.5775 \times 10^{-11}\%$
Dragon	0.219	95,385	0.0005%	0.1260%	0.0266%	$8.9217 \times 10^{-9}\%$

Additionally, Table 10 shows that our method results in a negligible distortion in the quality of the stego models. In fact, for the same number of embedded bits in the Armadillo 3D object, our distortion is approximately  $1 \times 10^{-5} \%$ ,  $1 \times 10^{-10} \%$  and  $1 \times 10^{-7} \%$  of those introduced in [61], [64] and [62], respectively. We also compare our achieved PSNR values with those reported in [59] where they proposed an adjustable distortion scheme that yields approximately identical PSNR values for any value of the number of bits embedded per vertex,  $k$ . The results are reported in Table 11, for  $k = 24$  bits per vertex which is identical to the proposed scheme. Our achieved PSNR values show a significant enhancement for the Elephant and the Armadillo stego models and a set of slightly improved values for the remaining common experimental 3D models except for the Horse, the Bunny

**Table 11** – PSNR comparison between [59] (for  $k = 24$ ) and the proposed algorithm.

3D Model	PSNR	
	[59]	Proposed
Horse	<b>130.29</b>	105.51
Venus	133.01	<b>136.86</b>
Elephant	128.83	<b>227.67</b>
Armadillo	133.05	<b>239.34</b>
Bunny	<b>133.11</b>	108.58
Hand	131.27	<b>140.87</b>
Dragon	<b>131.31</b>	110.59

**Table 12** – Comparison between our proposed algorithm and those in [65, 30, 58, 32, 66] and [59]. (The symbols  $\times$ ,  $\triangle$  and  $\checkmark$  in the attacks part refer to “does not withstand”, “partially withstands” and “completely withstands” the specified attack, respectively.)

Property	[65]	[30]	[58]	[32]	[66]	[59]	Proposed
Capacity [bits]	$\sim  V $	$9 V $	$3n_{layer} V $ , $n_{layer} < 23$	$\alpha + 3n_{layer} V $ , $n_{layer} < 23$	$\sim 70 V $	$\sim 90 V $	$24 V $
Distortion	Incremental	Incremental	Incremental	Incremental	Adjustable	Adjustable	Incremental
Domain	Spatial	Spatial & Representational	Spatial	Spatial & Representational	Spatial	Spatial	Spatial
Extraction	Blind	Blind	Blind	Blind	Blind	Blind	Blind
<b>Attacks</b>							
Noise	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
Smoothing	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\triangle$
Vertex reordering	$\triangle$	$\times$	$\triangle$	$\triangle$	$\triangle$	$\times$	$\checkmark$
Similarity transform	$\triangle$	$\triangle$	$\triangle$	$\triangle$	$\triangle$	$\checkmark$	$\checkmark$

and the Dragon. This could be justified by the fact that the 3D models utilized consist of significantly small vertices’ coordinates’ values and hence the embedding leads to an increased distortion compared to the other models. This can be adjusted by increasing the scale factor  $\delta$  to achieve an improved PSNR for these models in particular.

Table 12 compares our proposed scheme to those presented in [65, 30, 58, 32, 66] and [59]. All the 7 schemes embed data over the spatial domain and can blindly extract the data from the stego object. Additionally, the schemes proposed in [30] and [32] also embed data over the representational domain. The embedding capacity proposed is considered superior in comparison to [65] and [30]. However, the achieved capacity in [66] and [59] is much larger and could also be possibly larger in [58] and [32] according to the number of layers used for embedding. These extremely increased capacities are achieved as a trade-off with the algorithm complexity. For example, the algorithm proposed in [59], requires preprocessing since a truncated space is formulated and utilized for embedding instead of directly using the original vertices of the cover 3D object.

Furthermore, only the schemes proposed in [66] and [59] in Table 12 achieve an adjustable distortion. However, as depicted in Table 11, our achieved PSNR values calculated using (4), which is also a measure of distortion, outperform some of those reported in [59]. This is due to the minimized distortion introduced by our proposed scheme and thus its performance could actually be comparable to schemes with adjustable distortion.

To evaluate the robustness of the proposed algorithm, we applied some of the prevalent attacks from the literature such as noise, vertex reordering, smoothing and similarity transforms such as reflection, uniform scaling, rotation and translation. To begin with, allying with most of the spatial domain embedding algorithms, our proposed algorithm is considered vulnerable to noise attacks. This is due to the change of the vertices’  $x$ -,  $y$ - and  $z$ -c coordinates and hence data extraction becomes erroneous. Furthermore, our proposed algorithm orders the 3D object vertices according to their distances from the 3D object’s centroid, prior to data embedding. Hence, at the receiver side, the vertices of the stego object will first be reordered according to their means prior to data extraction. Consequently, unlike the proposed algorithms reported in Table 12 which either completely or partially withstand vertex reordering, our scheme substantially withstands vertex reordering because it is already handled as a part of the embedding and extraction stages. In addition to that, our proposed scheme withstands similarity transforms because for a known value of  $D_{max}$ , the attacked oriented bounding box of the stego model can be adjusted to meet the unaffiliated smallest cuboid bounding box of the cover model and hence the embedded data can still be successfully extracted.

Additionally, in order to prove that our proposed algorithm partially withstands smoothing as reported in Table 12, we computed the achieved Message Error Rate (MER) in bits as reported in Table 13, for three different types of smoothing using three different filters namely, Median, Gaussian and Laplacian. For Median filtering,

**Table 13** – MER (%) achieved for different attacks on the proposed algorithm.

3D Model	Non-uniform scaling			Median filtering	Smoothing	
	0.5X	1.5Y	2Z		Gaussian filtering	Laplacian filtering
Bunny	13.42	19.51	16.98	29.37	0.50	0.49
Dragon	13.41	19.50	16.97	29.97	0.50	0.48
Armadillo	13.30	19.52	16.97	30.64	0.50	0.50

the attacked 3D stego models can successively retrieve approximately 70%, 50% and 50% of the embedded data, respectively, for the Bunny, Dragon and Armadillo 3D objects. However, the deployment of error correcting codes [67, 68, 69, 70] could hence lead to a significant improvement to the MER values reported in Table 13.

Finally, Table 13 also shows the MER achieved for non-uniform scaling in the  $x$ -,  $y$ - and  $z$ -coordinates. As depicted in the reported results, our proposed scheme can successively retrieve approximately 87%, 80% and 83% for non-uniform scaling in the  $x$ -,  $y$ - and  $z$ -coordinates, respectively. The consistency of the MER results achieved by non-uniform scaling and smoothing attacks for different 3D objects in this table are a sign for the stability and the coherence of our proposed scheme.

## 5. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a novel Gray code-based steganography algorithm that deployed an additional security layer by utilizing encryption using either AES-128 or Blowfish algorithms. The proposed scheme has been proven to inaugurate a minimal distortion while perpetuating a high embedding capacity. Our proposed scheme has been proven to introduce a notable advancement over existing algorithms in terms of capacity, distortion, visual imperceptibility and robustness against major attacks such as vertex reordering and similarity attacks. Our exhaustive performance analysis and considerable comparisons with other algorithms from the literature have proven that our proposed scheme has a high capacity and additionally provides a low time complexity while providing requisite security.

For future work, we recommend the application of error correcting codes such as concatenating repetition and Bose–Chaudury–Hocquenheim (BCH) [73] codes which were proven to have a significant BER performance for an un-coded error probability up to 30%, in order to improve the robustness of our proposed scheme against attacks. We also suggest the deployment of Turbo codes which were utilized in double watermarking [74], since they have been proven to correctly retrieve at least 90% of the embedded bits in either the multi-resolution field or the spatial one. Finally, allowing for embedding over the representational domain in conjunction to the spatial domain deployed in our proposed scheme could lead to a superior embedding capacity enhancement.

## REFERENCES

- [1] Haji M. Furqan, Muhammad Sohaib J. Solaija, Halise Türkmen, and Hüseyin Arslan. “Wireless Communication, Sensing, and REM: A Security Perspective”. In: *IEEE Open Journal of the Communications Society* 2 (2021), pp. 287–321. DOI: 10.1109/OJCOMS.2021.3054066.
- [2] Wassim Alexan, Mazen El Beheiry, and Omar Gamal-Eldin. “A comparative study among different mathematical sequences in 3d image steganography”. In: *International Journal of Computing and Digital Systems* 9.4 (2020), pp. 545–552.
- [3] Wassim Alexan, Mohamed ElBeltagy, and Amr Aboshousha. “Lightweight Image Encryption: Cellular Automata and the Lorenz System”. In: *2021 International Conference on Microelectronics (ICM)*. 2021, pp. 34–39. DOI: 10.1109/ICM52667.2021.9664961.
- [4] Sara Farrag and Wassim Alexan. “Secure 3D data hiding technique based on a mesh traversal algorithm”. In: *Multimedia Tools and Applications* (2020), pp. 1–15. DOI: <https://doi.org/10.1007/s11042-020-09437-w>.
- [5] Suresh Kumar and Nishant Sharma. “Emerging Military Applications of Free Space Optical Communication Technology: A Detailed Review”. In: *Journal of Physics: Conference Series*. Vol. 2161. 1. IOP Publishing, 2022, p. 012011.
- [6] Raihan Ur Rasool, Hafiz Farooq Ahmad, Wajid Rafique, Adnan Qayyum, and Junaid Qadir. “Security and privacy of internet of medical things: A contemporary review in the age of surveillance, botnets, and adversarial ML”. In: *Journal of Network and Computer Applications* (2022), p. 103332.
- [7] Jameel Arif, Muazzam A Khan, Baraq Ghaleb, Jawad Ahmad, Arslan Munir, Umer Rashid, and Ahmed Al-Dubai. “A Novel Chaotic Permutation–Substitution Image Encryption Scheme Based on Logistic Map and Random Substitution”. In: *IEEE Access* (2022), pp. 1–1. DOI: 10.1109/ACCESS.2022.3146792.

- [8] Marwa Tarek Elkandoz, Wassim Alexan, and Hisham H Hussein. "Logistic sine map based image encryption". In: *2019 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*. IEEE. 2019, pp. 290–295.
- [9] F. Ernawan and M. N. Kabir. "A blind watermarking technique using redundant wavelet transform for copyright protection". In: *2018 IEEE 14th International Colloquium on Signal Processing Its Applications (CSPA)*. 2018, pp. 221–226.
- [10] A. Girdhar and V. Kumar. "Comprehensive survey of 3D image steganography techniques". In: *IET Image Processing* 12.1 (2018), pp. 1–10.
- [11] Ahmed Seif and Wassim Alexan. "A High Capacity Gray Code Based Security Scheme for Non-Redundant Data Embedding". In: *2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE)*. IEEE. 2020, pp. 130–136.
- [12] Sara Farrag and Wassim Alexan. "Secure 2d image steganography using recaman's sequence". In: *2019 International Conference on Advanced Communication Technologies and Networking (CommNet)*. IEEE. 2019, pp. 1–6.
- [13] Mariusz Dzwonkowski and Roman Rykaczewski. "Reversible Data Hiding in Encrypted DICOM Images Using Cyclic Binary Golay (23, 12) Code". In: *IEEE Access* 9 (2021), pp. 60503–60515. DOI:10.1109/ACCESS.2021.3074254.
- [14] Marwa T Elkandoz, Wassim Alexan, and Hisham H Hussein. "3D Image Steganography Using Sine Logistic Map and 2D Hyperchaotic Map". In: *2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*. IEEE. 2019, pp. 1–6.
- [15] Sara Farrag and Wassim Alexan. "A high capacity geometrical domain based 3d image steganography scheme". In: *2019 International Conference on Advanced Communication Technologies and Networking (CommNet)*. IEEE. 2019, pp. 1–7.
- [16] J. Wang, X. Jia, X. Kang, and Y. Shi. "A Cover Selection HEVC Video Steganography Based on Intra Prediction Mode". In: *IEEE Access* 7 (2019), pp. 119393–119402.
- [17] Reem Hussein and Wassim Alexan. "Secure message embedding in audio". In: *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)*. IEEE. 2019, pp. 1–6.
- [18] Marwa Tarek Elkandoz and Wassim Alexan. "Logistic Tan Map Based Audio Steganography". In: *2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*. IEEE. 2019, pp. 1–5.
- [19] Farah Hemeida, Wassim Alexan, and Salma Mamdouh. "Blowish-secured audio steganography". In: *2019 Novel Intelligent and Leading Emerging Sciences Conference (NILES)*. Vol. 1. IEEE. 2019, pp. 17–20.
- [20] N. Sharma and U. Batra. "A review on spatial domain technique based on image steganography". In: *2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN)*. 2017, pp. 24–27.
- [21] Maggie Mashaly, Ahmed El Saied, Wassim Alexan, and Abeer S Khalifa. "A Multiple Layer Security Scheme Utilizing Information Matrices". In: *2019 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*. IEEE. 2019, pp. 284–289.
- [22] H. Malekmohamadi and S. Ghaemmaghami. "Steganalysis of LSB based image steganography using spatial and frequency domain features". In: *2009 IEEE International Conference on Multimedia and Expo*. 2009, pp. 1744–1747.
- [23] Y. Yang. *Information Analysis for Steganography and Steganalysis in 3D Polygonal Meshes*. Durham University, 2013. URL: <https://books.google.com.eg/books?id=yLumoAEACAAJ>.
- [24] K Anish, N Arpita, H Nikhil, K Sumant, S Bhagya, and SD Desai. "Intelligence system security based on 3-D image". In: *Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications*. Springer. 2017, pp. 159–167.
- [25] Zhenyu Li, Sébastien Beugnon, William Puech, and Adrian G Bors. "Rethinking the high capacity 3D steganography: Increasing its resistance to steganalysis". In: *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2017, pp. 510–414.
- [26] Hang Zhou, Kejiang Chen, Weiming Zhang, Yuanzhi Yao, and Nenghai Yu. "Distortion design for secure adaptive 3-d mesh steganography". In: *IEEE Transactions on Multimedia* 21.6 (2018), pp. 1384–1398.
- [27] Philippe Amat, William Puech, Sébastien Druon, and Jean-Pierre Pedeboy. "Lossless 3D Steganography Based on MST and Connectivity Modification". In: *Sig. Proc.: Image Comm.* 25 (July 2010), pp. 400–412. DOI:10.1016/j.image.2010.05.002.
- [28] Yuan-Yu Tsai. "A Distortion-Free Data Hiding Scheme for Triangular Meshes Based on Recursive Subdivision". In: *Advances in Multimedia* 2016 (Jan. 2016), pp. 1–10. DOI:10.1155/2016/4267419.

- [29] Sagarika Borah and Bhogeswar Borah. "A blind, semi-fragile 3d mesh watermarking algorithm using minimum distortion angle quantization index modulation (3d-mdaqim)". In: *Arabian Journal for Science and Engineering* 44.4 (2019), pp. 3867–3882.
- [30] Yu-Ming Cheng and Chung-Ming Wang. "A high-capacity steganographic approach for 3D polygonal meshes". In: *The Visual Computer* 22 (2006), pp. 845–855.
- [31] Alexander Bogomjakov, Craig Gotsman, and Martin Isenburg. "Distortion-free steganography for polygonal meshes". In: *Computer graphics forum*. Vol. 27. 2. Wiley Online Library, 2008, pp. 637–642.
- [32] Chao-Hung Lin, Min-Wen Chao, Jyun-Yuan Chen, Cheng-Wei Yu, and Wei-Yen Hsu. "A high-capacity distortion-free information hiding algorithm for 3D polygon models". In: *Int J Innov Comput Inf Control* 9.3 (2013), pp. 1321–1335.
- [33] O. Cataltas and K. Tutuncu. "Comparison of LSB image steganography technique in different color spaces". In: *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*. 2017, pp. 1–6.
- [34] B. Petit, J. Lesage, E. Boyer, J. Franco, and B. Raffin. "Remote and collaborative 3D interactions". In: *2009 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*. 2009, pp. 1–4.
- [35] AkoMuhamad Abdullah. "Advanced encryption standard (aes) algorithm to encrypt and decrypt data". In: *Cryptography and Network Security* 16 (2017).
- [36] William J Buchanan, Shancang Li, and Rameez Asif. "Lightweight cryptography methods". In: *Journal of Cyber Security Technology* 1.3-4 (2017), pp. 187–201.
- [37] Seifeldin Yasser, Adham Hesham, Mahmoud Hassan, and Wassim Alexan. "AES-Secured Bit-Cycling Steganography in Sliced 3D Images". In: *2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE)*. IEEE, 2020, pp. 227–231.
- [38] Irfan Landge, Burhanuddin Contractor, Aamna Patel, and Rozina Choudhary. "Image encryption and decryption using blowfish algorithm". In: *World Journal of Science and Technology* 2.3 (2012), pp. 151–156.
- [39] Pia Singh and Karamjeet Singh. "Image encryption and decryption using blowfish algorithm in MATLAB". In: *International Journal of Scientific & Engineering Research* 4.7 (2013), pp. 150–154.
- [40] Y. J. Chanu, T. Tuithung, and K. Mangleem Singh. "A short survey on image steganography and steganalysis techniques". In: (2012), pp. 52–55.
- [41] G. Mostafa and W. Alexan. "A High capacity Double-Layer Gray Code Based Security Scheme for Secure Data Embedding". In: *2019 International Symposium on Networks, Computers and Communications (ISNCC)*. 2019, pp. 1–6.
- [42] Eric W Weisstein. *Gray code*. From *MathWorld – A Wolfram Web Resource*. 2008.
- [43] Olivia Di Matteo, Anna McCoy, Peter Gysbers, Takayuki Miyagi, RM Woloshyn, and Petr Navrátil. "Improving Hamiltonian encodings with the Gray code". In: *Physical Review A* 103.4 (2021), p. 042405.
- [44] Aishwarya Kaity and Sangeeta Singh. "Optimized area efficient quantum dot cellular automata based reversible code converter circuits: design and energy performance estimation". In: *The Journal of Supercomputing* 77.10 (2021), pp. 11160–11186.
- [45] Elizabeth Caroline, Margarat Michael, and Susan Christiana. "Design and performance analysis of SOA-MZI-based Tbps all-optical gray converters with M-ary DPSK coded binary, gray, and octal inputs". In: *Optical Engineering* 60.1 (2021), p. 015103.
- [46] Michael Cribbs, Ric Romero, and Tri Ha. "Modulation-Based Physical Layer Security via Gray Code Hopping". In: *2021 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR 2021)*. 2021, pp. 1–6. doi: 10.1109/CQR39960.2021.9446265.
- [47] Junjia Li, Yajie Li, Bo Wang, Kai Wang, Yongli Zhao, and Jie Zhang. "Ciphertext Mapping Method based on Gray Code in Quantum Noise Stream Cipher". In: *2021 19th International Conference on Optical Communications and Networks (ICOCN)*. 2021, pp. 1–3. doi: 10.1109/ICOCN53177.2021.9563675.
- [48] Eunsang Lee, Young-Sik Kim, Jong-Seon No, Minki Song, and Dong-Joon Shin. "Modification of Frodokem Using Gray and Error-Correcting Codes". In: *IEEE Access* 7 (2019), pp. 179564–179574. doi: 10.1109/ACCESS.2019.2959042.
- [49] *Stanford Graphics Laboratory*. URL: <https://graphics.stanford.edu/> (visited on 12/28/2018).
- [50] Gurpreet Singh. "A study of encryption algorithms (RSA, DES, 3DES and AES) for information security". In: *International Journal of Computer Applications* 67.19 (2013).
- [51] CSE ECE and MMU Mullana. "Image quality assessment techniques pn spatial domain". In: *IJCST* 2.3 (2011).
- [52] M Gulame, KR Joshi, and RS Kamthe. "A full reference based objective image quality assessment". In: *Int J Adv Electr Electron Eng* 2 (2013), pp. 13–8.

- [53] Mateusz Matan et al. "Application of the 3D objects surface shape analysis algorithms in biomedical engineering". In: *Czasopismo Techniczne* 2014.Nauki Podstawowe Zeszyt 2 NP (16) 2014 (2014), pp. 81–98.
- [54] Bernard Fichet, Domenico Piccolo, Rosanna Verde, and Maurizio Vichi. *Classification and multivariate analysis for complex data structures*. Springer, 2011, pp. 161–162.
- [55] Prasenjit Das, Subhrajyoti Deb, Nirmalya Kar, and Baby Bhattacharya. "An Improved DNA based dual cover steganography". In: *Procedia Computer Science* 46 (2015), pp. 604–611.
- [56] Tomislav Maroševi . "The Hausdorff distance between some sets of points". In: *Mathematical Communications* 23.2 (2018), pp. 247–257.
- [57] P Thiagarajan, V Natarajan, G Aghila, V Prasanna Venkatesan, and R Anitha. "Pattern based 3D image Steganography". In: *3D Research* 4.1 (2013), pp. 1–8.
- [58] Min-Wen Chao, Chao-hung Lin, Cheng-Wei Yu, and Tong-Yee Lee. "A high capacity 3D steganography algorithm". In: *IEEE transactions on visualization and computer graphics* 15.2 (2009), pp. 274–284.
- [59] Nannan Li, Jiangbei Hu, Riming Sun, Shengfa Wang, and Zhongxuan Luo. "A high-capacity 3D steganography algorithm with adjustable distortion". In: *IEEE Access* 5 (2017), pp. 24457–24466.
- [60] Yu-Ming Cheng and Chung-Ming Wang. "An adaptive steganographic algorithm for 3D polygonal meshes". In: *The Visual Computer* 23.9-11 (2007), pp. 721–732.
- [61] Cheng-Hung Chuang, Chin-Wei Cheng, and Zhi-Ye Yen. "Reversible data hiding with affine invariance for 3D models". In: *IET International Conference on Frontier Computing, Theory, Technologies and Applications*. 2010, pp. 77–81. DOI: 10.1049/cp.2010.0541.
- [62] Yao-Hsien Huang and Yuan-Yu Tsai. "A reversible data hiding scheme for 3D polygonal models based on histogram shifting with high embedding capacity". In: *3D Research* 6.2 (2015), p. 20.
- [63] P. Agarwal and B. Prabhakaran. "Robust Blind Watermarking of Point-Sampled Geometry". In: *IEEE Transactions on Information Forensics and Security* 4.1 (2009), pp. 36–48. ISSN: 1556-6013. DOI: 10.1109/TIFS.2008.2011081.
- [64] Chang-Yun Jhou, Jeng-Shyang Pan, and Ding Chou. "Reversible data hiding base on histogram shift for 3D vertex". In: *Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2007)*. Vol. 1. IEEE. 2007, pp. 365–370.
- [65] François Cayre and Benoit Macq. "Data hiding on 3-D triangle meshes". In: *IEEE Transactions on signal Processing* 51.4 (2003), pp. 939–949.
- [66] Ying Yang, Norbert Peyerimhoff, and Ioannis Ivrissimtzis. "Linear correlations between spatial and normal noise in triangle meshes". In: *IEEE transactions on visualization and computer graphics* 19.1 (2013), pp. 45–55.
- [67] Weiming Zhang and Shiqu Li. "A coding problem in steganography". In: *Designs, Codes and Cryptography* 46.1 (2008), pp. 67–81.
- [68] Hristo Kostadinov and Nikolai L Manev. "Error correcting codes and their usage in steganography and watermarking". In: *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE. 2015, pp. 333–336.
- [69] M Ould Medeni and EM Souidi. "A steganography schema and error-correcting codes". In: *Journal of Theoretical and Applied Information Technology* 18.1 (2010), pp. 42–47.
- [70] Carlos Munuera. "Steganography and error-correcting codes". In: *Signal Processing* 87.6 (2007), pp. 1528–1533.
- [71] Jiahao Zhu, Yushu Zhang, Xinpeng Zhang, and Xiaochun Cao. "Gaussian Model for 3D Mesh Steganography". In: *IEEE Signal Processing Letters* 28 (2021), pp. 1729–1733.
- [72] Min-Wen Chao, Chao-hung Lin, Cheng-Wei Yu, and Tong-Yee Lee. "A high capacity 3D steganography algorithm". In: *IEEE transactions on visualization and computer graphics* 15.2 (2008), pp. 274–284.
- [73] Jerome Darbon, Bulent Sankur, and Henri Maitre. "Error correcting code performance for watermark protection". In: *Security and Watermarking of Multimedia Contents III*. Vol. 4314. International Society for Optics and Photonics. 2001, pp. 663–673.
- [74] Chokri Chemak, Mohamed Salim Bouhleb, and Jean Christophe Lapayre. "A new scheme of robust image watermarking: the double watermarking algorithm". In: *Proceedings of the 2007 summer computer simulation conference*. Society for Computer Simulation International. 2007, pp. 1201–1208.

## AUTHORS



**Ghadir Mostafa** received her MSc degree from the German University in Cairo (GUC), in December 2019. In 2018, she graduated from the Faculty of Information Engineering and Technology at GUC. She is currently enrolled as an assistant lecturer and a PhD student in the department of Communications Engineering in the same faculty. Her current research scope is in the area of wireless digital communications and networking.



**Wassim Alexan** was born in Alexandria, Egypt, in 1987. He completed a BSc, MSc and PhD in communications engineering in 2010, 2012 and 2017, respectively, from the German University in Cairo (GUC). In 2019, he completed an MBA from the same university. He is currently an assistant professor of Communications Engineering in the faculty of Information Engineering and Technology at GUC. He has authored or co-authored over 50 publications in international journals and conference proceedings. He is the recipient of a Best Paper Award at the International IEEE SPA 2015 conference and a Best Poster Award at the IEEE NSRC 2020. His research interests lie in the fields of information security, signal processing and wireless communications. He is a member of ACM and a Senior member of IEEE.