

Article

Secure Chaotic Cryptosystem for 3D Medical Images

Antonios S. Andreatos ^{1,*}  and Apostolos P. Leros ^{2,†}¹ Division of Computer Engineering and Information Science, Hellenic Air Force Academy, 13671 Dekeleia, Attica, Greece² General Department, National and Kapodistrian University of Athens, 15772 Athens, Greece; lerosapostolos@gmail.com

* Correspondence: antonios.andreatos@hafa.haf.gr

† Retired.

Abstract

This study proposes a lightweight double-encryption cryptosystem for 3D medical images such as Magnetic Resonance Imaging (MRI), Positron Emission Tomography (PET) scans, and Computed Tomography scans (CT). The first encryption process uses chaotic pseudo-random numbers produced by a Lorenz chaotic system while the second applies Cipher Block Chaining (CBC) mode using outputs from a Pseudo-Random Number Generator (PRNG). To enhance diffusion and confusion, additional voxel shuffling and bit rotation operations are incorporated. Various sets of optimized parameters for the Lorenz system are calculated using either a genetic algorithm or a random walk. The master key of the cryptosystem is 672 bits long and consists of two components. The first component is the SHA-512 hash of the input image while the second component consists of the initial conditions of the Lorenz chaotic system and is 160 bits long. The master key is processed by a function that generates fourteen subkeys, which are then used in different stages of the algorithm. The cryptosystem exhibits excellent performance in terms of entropy, NPCR, UACI, key sensitivity, security, and speed, ensuring the confidentiality of personal medical data and resilience against advanced computational threats, and making it a good candidate for real-time 3D medical image encryption in healthcare systems.

Keywords: 3D medical images; MRI; PET scan; CT scan; DICOM; NIfTI; Lorenz chaotic system; Cipher Block Chaining (CBC); genetic algorithms; random walk

MSC: 68U10; 94A60; 37D45; 94A17; 68T20



Academic Editor: Hairong Lin

Received: 24 July 2025

Revised: 18 August 2025

Accepted: 4 September 2025

Published: 16 October 2025

Citation: Andreatos, A.S.; Leros, A.P. Secure Chaotic Cryptosystem for 3D Medical Images. *Mathematics* **2025**, *13*, 3310. <https://doi.org/10.3390/math13203310>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction: The Need for Encrypting Private Medical Images

In today's rapidly digitizing healthcare landscape, the use of electronic health records and digital imaging has revolutionized patient care. Among the most critical types of health data are patient medical images such as X-rays, Computed Tomography (CT) scans, Magnetic Resonance Imaging (MRI), ultrasounds, Positron Emission Tomography (PET) scans, etc. Each type examines different tissues, organs, or bones properties. Medical images play a crucial role in diagnosis and treatment planning but they also contain highly confidential patient details.

Nowadays, it is increasingly common to store medical images in the cloud, leveraging its scalability, accessibility, and cost-efficiency for healthcare systems worldwide. The transmission of unencrypted medical images across networks, as well as their storage in cloud-based platforms or telemedicine services, raises significant privacy and security

concerns, as sensitive patient data can be exposed to unauthorized access, data breaches, or misuse during transmission and storage. Medical images contain not only diagnostic information but also personal health identifiers, making them high-value targets for cyber-attacks. As a result, ensuring the confidentiality and integrity of such data is paramount. To address these risks, medical image encryption has become a necessity, providing a robust line of defense by protecting data both at rest and in transit. Secure encryption techniques, especially those tailored for the large size and critical nature of medical imagery, are essential to enable safe cloud integration. Without encryption, these images can be intercepted, accessed, or misused by unauthorized entities, leading to breaches of patient confidentiality, identity theft, or even medical fraud.

Beyond protecting individual privacy, encryption upholds ethical and legal standards in healthcare. Regulatory frameworks such as HIPAA (Health Insurance Portability and Accountability Act) in the United States and GDPR (General Data Protection Regulation) in the European Union mandate the secure handling of personal health data, including medical images. Failure to implement proper encryption measures can result in serious legal and financial consequences for healthcare providers and institutions.

Ultimately, encrypting personal medical images is a fundamental step toward building trust between patients and healthcare systems. It ensures that sensitive health information remains private, secure, and accessible only to authorized personnel, protecting both the dignity of the individual and the integrity of medical practice in the digital age.

1.1. Peculiarities of Medical Images

Medical images differ from common images (such as photographs) in several key aspects, including purpose, acquisition methods, content, technical specifications, and interpretation. Medical images are acquired for diagnostic and clinical decision-making, making their content highly sensitive and semantically critical. Medical images often contain fine anatomical details, subtle intensity variations, and low-contrast regions, features essential for accurate interpretation by healthcare professionals.

Additionally, medical images are often high-dimensional and exhibit distinct characteristics such as spatial correlation and redundancy among adjacent pixels or voxels.

Medical images typically follow standardized formats that include metadata like patient identity, acquisition parameters, and timestamps, information that must be protected to ensure privacy and regulatory compliance. Moreover, the grayscale nature, higher bit depth (e.g., 12–16 bits per pixel), and larger file sizes of medical images pose unique challenges for encryption algorithms, requiring specialized approaches that preserve data integrity while ensuring strong security.

Medical images differ fundamentally from conventional photographic images, which are often used as benchmark test cases in image encryption studies. As a result, encryption techniques designed for natural images may not be directly applicable or sufficiently robust for medical imaging applications, highlighting the need for tailored cryptographic solutions in e-health and cloud-based medical systems.

On the other hand, due to their large size and complex structure, medical images pose challenges for achieving strong, attack-resistant encryption. Moreover, they often exhibit high redundancy, making them vulnerable to statistical and differential attacks if not properly encrypted. This underscores the growing need for encryption methods that ensure security while preserving image quality and diagnostic value [1].

Table 1 presents a summary comparison of medical images vs. common images.

Table 1. Concise comparison of medical images vs. common images (photos).

Feature	Medical Images	Common Images
Purpose	Diagnosis, treatment	Documentation, communication, art
Acquisition	X-ray, MRI, CT, Ultrasound, etc.	Cameras (visible light)
Color/Grayscale	Mostly grayscale, some color	Mostly full-color
Bit Depth	High (12–16 bits)	Lower (8 bits per channel)
Format	DICOM	JPEG, PNG, HEIC
Dimensionality	2D, 3D, or 4D	Mostly 2D
Metadata	Extensive (patient, device, settings)	Limited (EXIF)
Interpretation	Requires medical expertise	Intuitive for general public
Quantitative Use	Yes (measurable pixel values)	No (perceptual only)

Differences Between Medical and Photographic Image Histograms

In medical imaging, particularly in CT and MRI DICOM volumes, it is common for histograms to appear “abnormal” in the following ways:

- Background regions (outside the body) are typically zero-padded, resulting in a dominant peak at $X = 0$ (black).
- Only voxels within the scanned anatomy (e.g., tissues, organs) contain meaningful, non-zero values.

As a result, entire zones of the histogram remain empty. This case is analyzed in detail below using test image 3.

1.2. Medical Image Formats and Standards

1.2.1. The DICOM Standard for Medical Imaging

DICOM (Digital Imaging and Communications in Medicine) is the dominant standard for medical imaging. DICOM is the acronym of Digital Imaging and Communications in Medicine medical images and can be either 2D or 3D, depending on the imaging modality and the type of data acquired [2].

The DICOM standard supports a wide range of imaging modalities, including Computed Tomography (CT), X-Ray Angiography (XA), Magnetic Resonance (MR), Mammography (MG), and Radio Fluoroscopy (RF).

Two-dimensional DICOM images are single-slice images, commonly produced by modalities such as X-ray, ultrasound, or individual slices from CT or MRI scans. Each file represents a two-dimensional cross-section (slice) of the body. Three-dimensional DICOM images are created by stacking multiple 2D slices acquired from modalities like CT, MRI, or PET scans to form a three-dimensional volume. These datasets are often used for advanced visualization, such as 3D reconstructions of organs or bones. In practice, a DICOM dataset typically consists of a series of 2D slices that can be reconstructed into 3D images using special software.

1.2.2. The NIfTI Format for Medical Imaging

NIfTI (Neuroimaging Informatics Technology Initiative) is a widely used file format in neuroimaging for storing and sharing brain imaging data, such as those obtained from MRI and functional MRI (fMRI). Developed to improve data interoperability across different neuroimaging analysis tools and platforms, NIfTI was introduced as an enhancement to the older Analyze 7.5 format.

There are two main variants:

- NIfTI-1: The original format, typically using the .nii extension (a single file) or the .hdr/.img pair (two files).
- NIfTI-2: An updated version designed for larger datasets, with improved precision in spatial metadata.

The key features of NIfTI format are:

1. Header and Image Data: Contains a header with metadata (e.g., image dimensions, voxel size, data type, orientation) and the actual image data.
2. 3D and 4D Support: Efficiently stores 3D volumes (e.g., structural MRI) and 4D data (e.g., time-series in fMRI).
3. Standardized Orientation: Uses a standardized coordinate system (e.g., RAS: Right-Anterior-Superior), improving compatibility with neuroimaging software like FSL, SPM, and AFNI.
4. Open and Simple: Designed to be platform-independent and easy to implement.

1.2.3. NIfTI vs. DICOM

While DICOM is the clinical standard for medical imaging (including neuroimaging), it stores data in a complex, multi-file structure optimized for clinical workflows. In contrast, NIfTI simplifies data for research and analysis by converting DICOM images into a compact, matrix-based format that is easier to process computationally.

1.2.4. Other Medical Formats

Several other formats used either as alternatives, supplements, or intermediates, depending on the context (e.g., research, archiving, analysis, or non-clinical applications), include:

1. MINC (Medical Imaging NetCDF). Common features:
 - Developed by the Montreal Neurological Institute.
 - Based on NetCDF (Network Common Data Form).
 - Flexible metadata support, useful in research environments.
 - Less common now but still used in some academic pipelines.
2. NRRD (Nearly Raw Raster Data). Common features:
 - Lightweight format for scientific and medical image data.
 - Stores data and metadata in a single file (or paired files).
 - Popular in open-source tools (e.g., 3D Slicer, ITK).
 - Supports rich metadata, making it useful for processing and visualization.

1.3. Limitations of the Traditional Encryption Methods

Widely-used traditional ciphers such as DES, 3DES, IDEA, and AES [3] are not fully appropriate for medical image encryption, for several reasons. First, medical images are typically high-dimensional and exhibit distinct features like spatial correlation and redundancy among adjacent pixels/voxels. Second, the large volume of data in these images leads to higher computational costs and longer processing times for encryption and decryption. Third, the security of conventional encryption methods can be compromised by varying image properties like redundancy, pixel correlation, and structure. Last but not least, traditional encryption methods lack adaptability, as they are not designed to effectively address the unique challenges of image data, including the demand for real-time encryption in certain applications and the preservation of image quality [1,4–8].

1.4. Confusion and Diffusion

Confusion and diffusion are two fundamental principles introduced by Claude Shannon, the father of information theory, in his seminal paper "Communication Theory of

Secrecy Systems”. These concepts form the basis for designing secure cryptographic systems [3,9].

Confusion is used to obscure the relationship between the plaintext (original data) and the ciphertext (encrypted data), making it difficult for attackers to deduce the key or the original message, even if they have access to the encrypted message. This is typically achieved through substitution operations, such as using S-boxes (substitution boxes) in block ciphers.

Confusion intends to make the relationship between ciphertext and encryption key as complex as possible, hiding the relationship between the plaintext (original image) and the ciphertext (encrypted image). In this research, confusion is achieved through chaotic encryption and random bit rotation in each voxel.

Diffusion ensures that a small change in the plaintext image propagates widely across the ciphertext, spreading the influence of each plaintext bit over many ciphertext bits and hiding statistical patterns of the plaintext and increasing resistance to statistical and differential cryptanalysis. This is usually achieved through pixel permutation operations or bit shuffling mechanisms.

Diffusion dissipates statistical structure of plaintext over bulk of ciphertext. In this research, Cipher Block Chaining (CBC) encryption mode [3] with XOR and permutation of pixels (spatial shuffling) and rotation operations are used to improve diffusion.

Together, confusion and diffusion ensure that cryptographic systems are resistant to various types of attacks, such as brute force, statistical, and differential attacks.

1.5. Performance Evaluation Metrics

Several metrics have been employed in order to measure the quality of the encryption process [4,10–12]. The most commonly used criteria are:

1. Information Entropy of encrypted image;
2. Number of Pixel Change Rate (NPCR);
3. Unified Average Change Intensity (UACI);
4. Autocorrelation of pixels/voxels;
5. Fourier transformation analysis;
6. Mean Squared Error (MSE);
7. Peak Signal to Noise Ratio (PSNR).

1.6. Contribution of This Research

The current study proposes a novel lightweight double cryptosystem for 3D and 4D medical images. The system accepts modern medical image formats such as NIfTI and DICOM, as well as any 3D medical image in 8-bit ('uint8') or 16-bit (uint16) unsigned integers. During preprocessing, input image voxels are converted to uint16—if they are not already in this format.

There is insufficient research on 3D DICOM and NIfTI image encryption, so this work is a contribution towards this direction. In addition, the proposed cryptosystem implements a series of features and innovations listed below.

1. Support for multiple concurrent medical image formats.
2. A preprocessing stage offering several useful capabilities.
3. Double encryption leveraging chaotic and PRNG sequences.
4. Optimization of the chaotic RNG sequence through Lyapunov exponent maximization using either a genetic algorithm or a random walk.
5. Implementation of both ECB and CBC encryption modes.
6. Excellent performance in terms of entropy, UACI, NPCR, PSNR, and speed.
7. Enhanced overall security.

Results from the processing of three representative test images will be presented in this paper.

1. Test image 1 is a 3D MRI scan of a portion of a skull, encoded in 8-bit unsigned integer (uint8) format.
2. Test image 2 is a 3D MRI scan of a skull volume stored in high-resolution NIfTI format, encoded as signed floating-point ('single') values.
3. Test image 3 is a 3D sub-volume extracted from a high-resolution CT scan of a human body, stored in DICOM format as a 16-bit unsigned integer (uint16) volumetric dataset.

Each of the aforementioned test images showcases different features of the proposed cryptosystem.

1.7. Paper Structure

This paper is structured as follows: Section 2 is a literature review; Section 3 presents the proposed system; Section 4 presents the results obtained from processing three 3D medical images; Section 5 is a security analysis; Section 6 is a short discussion, while Section 7 concludes the paper.

2. Related Work

Medical image encryption has become a vital aspect of secure healthcare data management, particularly in telemedicine and cloud-based medical systems. To protect the confidentiality and integrity of sensitive medical information, various encryption techniques have been developed over the past three decades. These include methods based on chaotic systems, DNA encoding, bit plane manipulation, Galois field theory, fractal mathematics, compressive sensing, neural networks, S-box transformations, frequency-domain encryption, meaningful encryption, and diffusion-based approaches, among others [1,10,13,14].

Among various encryption techniques, chaotic system-based image encryption has attracted growing interest. Due to their inherent properties, such as extreme sensitivity to initial conditions and system parameters, as well as long-term unpredictability of dynamic trajectories, chaotic systems are highly suitable for cryptographic applications, offering strong security features for protecting medical images.

2.1. Chaotic Image Encryption

Chaotic systems are dynamic nonlinear systems that exhibit non-stationarity, unpredictability, randomness and strong sensitivity to initial conditions. For these reasons, they are widely used in information security and random number generation.

One of the first schemes in image encryption employing chaotic maps was proposed by Jiri Fridrich in 1998 [13]. Since then, numerous chaotic systems have been developed and are widely used in image encryption, including the logistic map, Baker map, Hénon map, Arnold map, tent map, and hyper-chaotic maps, among others [10]. Chaotic systems used in chaotic image encryption can be:

1. One-dimensional chaotic systems containing only one independent variable, such as Hénon map, logistic map, sine map, etc. These systems are structurally simple, highly efficient in operation and exhibit diverse dynamical behaviors, but the key space is rather small, and their chaotic characteristics can be revealed by analyzing the bifurcation map, the periodicity, and so on. Hence, they are vulnerable to attacks.
2. Multi-dimensional chaotic systems involve multiple state variables and exhibit higher complexity, diverse dynamical behaviors and increased sensitivity to starting conditions, making them widespread in the field of image encryption. Examples include the Lorenz system, Chua system, Chen system, and hyper-chaotic systems (four or more dimensions).

Priyanka and Singh present a good survey of image encryption for healthcare applications [11].

Lyle et al. present several selective image-encryption schemes that exist in the literature [15].

Elnoamy uses the tent chaotic map for 2D color images and presents the state-of-the-art in recent and efficient picture-encryption algorithms [4].

Soleymani et al. present an image-encryption scheme leveraging the Arnold cat map and the Hénon chaotic map. The Arnold cat map is utilized to perform bit-level and pixel-level permutations on both the plaintext image and a dynamically generated secret image, while the Hénon map generates the secret image and provides the control parameters for the permutation stages [16].

Fu and Guo propose an image-encryption algorithm that integrates semi-tensor product compressed sensing with double random-phase encoding [17].

Shahna proposes a secure image-encryption method using a 4D hyper-chaotic Lorenz system. It employs innovative pixel- and bit-level scrambling with keystream-based permutation and substitution. The scheme ensures strong resistance against statistical, differential, and occlusion attacks. Experimental results confirm high security, efficiency, and suitability for applications like medical image protection and cloud-based data transmission. The bibliography contains a lot of references about chaotic image encryption [18].

Lan et al. propose GRTPHM, a novel multi-image encryption scheme combining the Generalized Rectangular Transform (GRT) and a Penta-Hyperchaotic Map (PHM) to achieve high security, efficiency, and versatility. The encryption process involves two phases: pixel permutation and diffusion. GRT permutes pixel positions using a flexible transformation matrix, disrupting spatial correlations and enhancing resistance to statistical attacks. In the diffusion phase, a bit-level random permutation (BLRP) technique and a 5D memristor-based hyperchaotic map (PHM), seeded via SHA-512, ensure strong confusion and diffusion. BLRP operates at the bit level for finer scrambling, while PHM generates highly sensitive chaotic sequences, amplifying the avalanche effect and improving resilience against differential and statistical attacks [19].

Ye et al. introduce a novel three-dimensional chaotic map called the Chebyshev-coupled Logistic-Sine Map (3D-CCLSM), followed by an image hiding algorithm that combines Local Binary Pattern (LBP) and Compressive Sensing (CS), referred to as ImHALC. The chaotic behavior of the four-dimensional hyper-chaotic system is validated through the Lyapunov exponents. The system enhances security, imperceptibility, and robustness against attacks. Experimental results demonstrate effective reconstruction and strong resistance to compression, noise, and occlusion, ensuring reliable steganographic performance. The bibliography contains many references about chaotic image encryption [20].

Ahmed et al. present an excellent review of medical image encryption with 107 references [21].

Wu, Zhang, Liu and Wang propose a lightweight encryption method for securing multiple medical images of varying sizes using a single operation, with computational cost similar to encrypting one image. Images are first zero-padded to a uniform size, stacked, and processed using keys generated via SHA-256 and a linear congruence algorithm. Encryption combines pixel scrambling and DNA encoding/decoding, enabling lossless recovery and independent decryption of each image to prevent leakage. Simulation results show strong resistance to statistical, differential, noise, and cropping attacks, with high entropy values and low adjacent-pixel correlation. The method achieves high security, efficiency, and robustness, making it suitable for real-time, resource-limited medical applications [22].

Yousif et al. present a robust image encryption scheme combining zigzag and spiral scanning, the El-Gamal public key cryptosystem, and high-dimensional chaotic models. The scanning stage permutes pixel positions, followed by El-Gamal encryption. Then, the Lorenz and Rössler chaotic systems perform confusion (pixel position shuffling) and diffusion (pixel value alteration), respectively, merging these stages to enhance security and expand the key space. Their method demonstrates strong key sensitivity, robustness against occlusion, and competitive encryption performance for grayscale and color images [23].

Muthu and Murali introduced the JoanS-MuraliP (JSMP) map, a novel one-dimensional chaotic map with a large continuous chaotic band, enhancing cryptosystem robustness. Developed via function composition of logistic and quadratic maps, the JSMP map offers high nonlinearity, unpredictability, and an extensive key space. Comprehensive dynamic analysis includes fixed points, stability, bifurcation, chaotic attractors, and transient chaos. Performance metrics—Lyapunov exponent, NIST randomness tests, and security evaluations—confirm its strong pseudo-randomness and resistance to attacks. Comparative studies show JSMP outperforming traditional one-dimensional chaotic maps, making it highly suitable for secure image encryption applications [24].

Panwar et al. propose a robust image encryption scheme for color and grayscale images using a 6D hyperchaotic system and a symmetric matrix. The 6D hyperchaotic map performs pixel position confusion for each RGB channel, while a 2×2 symmetric matrix modifies pixel values in the diffusion stage. Experiments on standard test images (peppers and baboon) demonstrate high security, with entropy values, NPCR around 99.59, and UACI close to 33.4, indicating strong resistance to statistical and differential attacks. Results show uniform histograms, negligible pixel correlation, and a large key space, confirming the algorithm's efficiency and robustness [25].

Zhang et al. present an enhanced Lorenz chaotic system and an efficient image encryption method. The improved system exhibits strong hyperchaotic behavior, with a maximum Lyapunov exponent of 2.9897. Based on this, a novel encryption scheme is proposed, combining multilayer permutation via image pyramid structure and Galois field diffusion enhanced with DNA computing. This approach improves permutation effectiveness and ensures robust encryption, even for pure black images. Experimental results show high randomness (Entropy: 7.9975) and strong sensitivity (NPCR: 99.6307%), demonstrating excellent resistance to attacks. The method is suitable for secure, efficient grayscale image encryption in applications like remote sensing and personal privacy protection [14].

Finally, Zhang and Liu review chaos-based image encryption, highlighting its significance in securing digital images against cyber threats. The paper surveys fundamental principles of chaotic systems and their integration into encryption techniques, evaluating strengths such as sensitivity to initial conditions and resistance to brute-force attacks. Applications span secure image storage, medical imaging, and multimedia communication. Challenges remain in balancing security, computational efficiency, and real-world implementation, particularly in resource-constrained environments. The study emphasizes the need for standardized frameworks and further research to optimize chaos-based encryption for practical use.

According to Zhang and Liu, medical image encryption algorithms must meet specific requirements, including strong security, preservation of data integrity, and fast encryption and decryption speeds. Traditional encryption methods often demand significant computational resources and time, leading to high latency, and are frequently limited by small key spaces and susceptibility to various attacks. In contrast, chaos-based encryption schemes for medical images have demonstrated promising results. These schemes have been rigorously evaluated through key space analysis, histogram analysis, correlation analysis, key

sensitivity tests, and assessments of resistance to differential attacks using metrics such as UACI (Unified Average Changing Intensity) and NPCR (Number of Pixel Change Rate). The results confirm that chaotic encryption provides a sufficiently secure and efficient solution for safeguarding medical image data in secure communication systems [7].

2.2. Chaotic 2D Medical Image Encryption

Chaos-based encryption methods have gained significant attention in medical image encryption, due to their desirable properties, such as non-linearity, sensitivity to initial parameters, speed, and robustness. The pseudorandom chaotic sequences are used to permute and diffuse plaintext images [10].

A chaos-based encryption system for DICOM medical images is presented in [26], leveraging chaos theory to improve data security and deter unauthorized access. The proposed approach demonstrated strong resistance to both brute-force and statistical attacks.

Jain et al. present a medical image encryption scheme that combines Arnold's cat map with the 2D Logistic-Sine Coupling Map (2D-LSCM) to improve security in telemedicine applications. The scheme uses chaotic maps for permutation and diffusion to achieve high randomness and resistance against cryptographic attacks. Security tests including key sensitivity, Shannon entropy, and differential analysis demonstrate improved performance over existing chaotic encryption methods. Although the algorithm's runtime is longer, it achieves stronger protection for sensitive medical images [5].

Suhasini and Kanchana propose an improved medical image-encryption method using an enhanced fractional-order Lorenz chaotic system. The scheme incorporates advanced permutation and diffusion techniques to secure sensitive data against modern attacks. By leveraging fractional calculus, the system exhibits increased complexity and unpredictability. Extensive experiments—including histogram, correlation, entropy, and differential analyses—confirm the method's robustness, sensitivity, and high security. It also demonstrates strong resistance to noise, occlusion, and statistical attacks. The approach is efficient and suitable for real-time applications involving confidential medical image transmission and storage [27].

Ibrahim et al. propose a generic medical image-encryption framework based on a novel arrangement of dynamic substitution boxes (S-boxes) and chaotic maps. The arrangement of S-box substitution before and after chaotic substitution is shown to successfully resist chosen plaintext and chosen ciphertext attacks. Four classical chaotic maps and four modern chaotic maps are used to implement the proposed framework. Based on speed analysis, they find that the classical Baker and Hénon maps achieve the best encryption throughput [6].

Belazi et al. present a novel medical image-encryption scheme combining chaotic systems and DNA encoding. The method employs a two-round permutation–substitution–diffusion structure to enhance security and complexity. Key steps include block-based permutation, pixel-level substitution, DNA encoding/decoding, bit-level substitution using DNA complement, and bit-level diffusion. Chaotic maps (logistic–Chebyshev and sine–Chebyshev) are used for key generation and diffusion. Keys are derived from SHA-256 hashes of the original image, ensuring sensitivity and uniqueness. Simulation results demonstrate strong resistance against statistical and differential attacks, making the scheme suitable for secure, real-time medical image transmission and storage in healthcare systems [28].

Muthu and Murali introduce a cryptosystem based on their robust JSMP chaotic map [24], specifically designed for encrypting DICOM images. The proposed method operates at the bit plane level to enhance security and resilience. Unlike conventional approaches that rely on the fixed bit allocation per pixel, this system dynamically determines the number of bit planes based on the actual bits stored in the data attribute, enabling

greater adaptability. Only the most significant bit (MSB) planes—those that carry the most critical visual information—are shuffled using chaotic keys generated by the JSMP map. After shuffling, the image is reconstructed by merging the modified MSB planes with the unaltered least significant bit (LSB) planes. To further strengthen security, the reconstructed image is diffused with a random image derived from the JSMP chaotic system, resulting in a highly secure cipher image.

A key innovation of this approach is its ability to effectively encrypt DICOM images regardless of their pixel depth. The system also features a unique key-generation mechanism and employs a variable number of keys, significantly enhancing sensitivity and entropy. Experimental results and performance evaluations demonstrate that the cryptosystem is highly effective and well-suited for securing medical images within health information systems [29].

Mortajez et al. present a secure and efficient image-encryption scheme for DICOM medical images using a novel chaotic system called the Joint Skew Map and Permutation (JSMP) map. The method operates at the bit plane level, dynamically determining bit planes based on actual pixel data rather than fixed allocation. Only the most significant bit (MSB) planes are shuffled using JSMP-generated chaotic sequences, then combined with LSB planes and diffused with a random image for enhanced security. The system supports variable key lengths and adapts to any pixel depth, ensuring high sensitivity, entropy, and resistance to attacks. Performance evaluations confirm its robustness, making it suitable for protecting sensitive medical data in healthcare information systems [30].

Zhuang et al. propose a medical image-encryption algorithm based on a novel five-dimensional (5D) multi-band multi-wing chaotic system combined with QR decomposition. First, a new 5D chaotic system is constructed using feedback control, exhibiting a high Lyapunov exponent and enhanced dynamical complexity. Next, both the plaintext image matrix and a generated chaotic sequence are decomposed via QR decomposition into orthogonal and upper triangular components. The orthogonal matrix from the image is then multiplied by that derived from the chaotic sequence, with the multiplication direction (left or right) controlled dynamically by the chaotic sequence. Concurrently, the elements of the upper triangular matrix are shuffled using an improved Joseph traversal scheme. The resulting matrices are multiplied together, and the final product undergoes bit-level scrambling to produce the encrypted image. Theoretical analysis and simulation results demonstrate that the proposed scheme offers a large key space, high key sensitivity, strong resistance to statistical and gray-level analysis, and effective encryption performance for medical images [31].

Zhang et al. (2024) present a medical image-encryption algorithm designed to protect patient privacy by combining Josephus-based scrambling with dynamic cross-diffusion techniques. First, a novel hyperchaotic system is developed, featuring an extended chaotic range and the ability to generate highly pseudo-random sequences. Based on these sequences, a dynamic Josephus scrambling method is introduced to effectively shuffle pixel positions, significantly reducing the correlation between neighboring pixels. Subsequently, a dynamic parallel cross-diffusion scheme, also driven by chaotic sequences, is applied to further enhance the encryption of the scrambled image. To strengthen security, the SHA-256 hash of the original image is computed and used to generate an initial key, ensuring key dependency on the plaintext. Simulation results demonstrate that the proposed algorithm successfully transforms the original image into a noise-like, visually uninformative ciphertext. Comprehensive security analyses confirm its strong resistance to common cryptanalytic attacks, including statistical and differential analysis. Furthermore, comparative evaluations with several state-of-the-art encryption schemes show that the proposed method achieves superior performance in terms of key sensitivity, entropy, and correlation coefficients, highlighting its effectiveness and robustness for secure medical image transmission [32].

2.3. Chaotic 3D Medical Image Encryption

Liu and Xue introduced a novel encryption scheme tailored for 3D medical images, addressing the limited research in this area. The proposed method integrates a biometric key and a cubic S-box to enhance security and overcome limitations of traditional encryption techniques, such as limited key space and lack of meaningful secret keys. The encryption system operates in four stages: pseudo-random sequence generation, confusion, substitution, and diffusion. Chaotic systems are used to generate sequences for pixel shuffling and diffusion. A multi-sorting confusion algorithm rearranges pixel positions in 3D space, while the cubic S-box guides pixel substitution to improve nonlinearity. Finally, a diffusion mechanism combining elementary cellular automata and finite field multiplication enhances sensitivity to input changes and hides statistical patterns. Experimental results on multiple 3D medical images show that the scheme achieves strong statistical performance, a large key space, high sensitivity, robustness, and resistance to common cryptographic attacks, making it a promising solution for secure medical data transmission [33].

Xu et al. [34] propose a novel multi-image medical image-encryption algorithm using a new chaotic system called Tent–Logistic Cross-Mixed Coupled Map Lattice (TLCMCML). The method addresses increasing concerns about data privacy in medical imaging by combining high security with efficient encryption of multiple images of varying sizes.

The encryption process involves two stages: 1. Region of Interest (ROI) scrambling: Each medical image's ROI is extracted and scrambled individually using an odd-even interleaving permutation to obscure diagnostic features. 2. Simultaneous Permutation–Diffusion (SPD): All images are concatenated into one, and SPD is applied to enhance randomness and security across the entire dataset.

The TLCMCML chaotic system demonstrates superior performance over existing systems like CCML and TDCCML, as shown through trajectory analysis, Lyapunov exponents, Shannon entropy, and NIST tests. Experimental results confirm strong resistance against statistical and differential attacks, low pixel correlation, high information entropy (7.9994), and robustness under noise and cropping attacks. The proposed algorithm supports arbitrary numbers and sizes of medical images, achieving both efficiency and strong cryptographic properties, making it suitable for secure transmission and storage of sensitive medical data [34].

Abd-El-Atty et al. present a novel double medical image-encryption scheme leveraging quantum walks and a logistic map to address emerging threats posed by quantum computing. The proposed system divides two input medical images into two parts: high 4-bit images (holding 94.125% of the data) and low 4-bit images. The high-information images are encrypted using quantum walks—a quantum computation model known for its high sensitivity and theoretically infinite key space. Meanwhile, the lower-information images are encrypted using a classical chaotic logistic map. To enhance sensitivity and security, SHA-256 hashing is used to derive key parameters from the image content, making encryption highly dependent on the original data. Simulation results demonstrate strong resistance to statistical, differential, and occlusion attacks, with excellent performance on metrics like entropy (7.9993), NPCR (99.6143%), UACI (33.4659%), and PSNR (7.48). The encryption system also performs competitively in speed (7.54 Mbit/s) and security against various attacks. Its two-layer approach and quantum foundations make it suitable for real-time medical image protection in healthcare systems [35].

Anandik et al. [1] present a DICOM-compatible 3D multimodality image-encryption scheme designed to secure medical images such as CT, MRI, X-ray, and ultrasound scans. The authors leverage hyperchaotic signals for both pixel permutation and intensity diffusion to ensure high security.

The process begins by standardizing the DICOM volume and applying a logistic chaotic map to shuffle voxel positions. A multi-level diffusion mechanism driven by hyperchaotic sequences modifies voxel intensities, creating strong resistance to statistical and differential attacks. An additional CAPTCHA-based layer is introduced, where a CAPTCHA image is encrypted and embedded alongside the medical data, enhancing authentication and protection.

Experiments across various modalities demonstrate high key sensitivity, uniform histogram distribution, and resilience against brute-force and known-plaintext attacks. Importantly, the method preserves DICOM metadata, ensuring compatibility with PACS systems and diagnostic workflows while delivering robust, multimodal medical image protection [1].

Lima et al. [36] introduce a novel method for encrypting 3D medical images using a three-dimensional steerable cosine number transform (3D-SCNT). This transform is obtained by rotating the basis vectors of the conventional 3D cosine number transform (3D-CNT) using finite field rotation operators, where the rotation angles act as the encryption keys.

The encryption process divides the image into 3D blocks, applies the 3D-SCNT, and uses scrambling techniques such as summation and Arnold transforms to enhance diffusion and resistance against cryptographic attacks.

Experimental results demonstrate that the encrypted images exhibit uniform histograms and low correlation between adjacent voxels, indicating strong statistical security. The scheme shows high sensitivity to key variations and robustness against differential attacks, confirmed by NPCR, UACI metrics, and NIST randomness tests. Comparisons with existing methods reveal superior performance in key space and entropy measures. The authors suggest extending the SCNT to higher-dimensional applications, such as 4D light-field image encryption, in future work [36].

3. The Proposed Cryptosystem

The proposed system encrypts grayscale 3D medical images in modern high resolution formats such as NIfTI and DICOM, in real or integer numbers. Figure 1 presents a simplified block diagram of the proposed cryptosystem. The encryption phase involves eight stages and the decryption just as many.

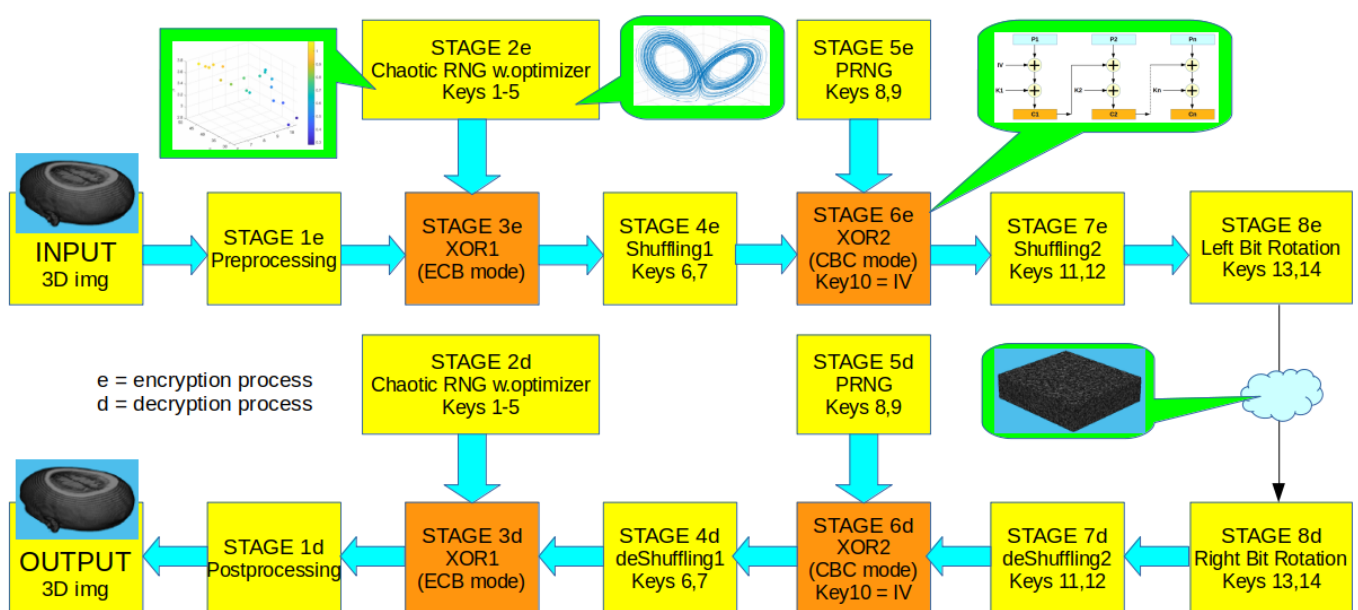


Figure 1. Block diagram of the proposed cryptosystem.

3.1. Stage 1: Preprocessing

Image preprocessing is a key step in medical image encryption to ensure the uniformity of the imported images and the quality of the encryption [1]. During preprocessing several tasks may take place:

1. Removal of null slices from 3D input images across all three dimensions;
2. Conversion of 4D to 3D images (if needed);
3. Conversion of 3D image data from 'single' or uint8 formats to uint16;
4. Rescaling of input 3D image to full scale (e.g., from 0–88 to 0–255 for test image 1);
5. Conversion of input 3D image to high resolution (optional; improves results);
6. Image rescaling based on histogram (optional; facilitates presentation);
7. Derivation of master key and subkeys for the various stages of processing.

A 3D volume can be viewed as a stack of related 2D slices along each spatial direction (x, y, or z). The regions surrounding human tissues in these slices —represented as black voxels— contain no meaningful data. Often, the first and last few slices in each direction are entirely null (i.e., 100% black, with voxel value 0) and can be safely removed without any loss of information. By eliminating empty frames across all three dimensions, the amount of data that needs to be processed, stored, or transmitted is significantly reduced. This is particularly beneficial in healthcare applications, as emphasized by [21].

As an example, in test image 1, null slices alone account for 32.25% of the total volume. Moreover, the proportion of black voxels (value 0) in the entire image is even higher, reaching 62%. Similarly, in test image 2 where black voxels constitute 75.38% of the volume, null slice removal results to a volume reduction of 48.87%. In addition, the high percentage of black voxels results in a highly skewed histogram. This imbalance is clearly visible in the histograms presented in Section 4. Removing null slices helps to balance the histogram by reducing the dominance of zero-valued voxels. The removed null slices can be restored at the receiving end if necessary, ensuring reversibility.

Conversion of voxel values to integer (uint16) is necessary when processing medical image formats such as 'single' (real numbers). This step is mandatory in order to apply XOR-encryption transformations. In this stage, key derivation also takes place.

Key Derivation

The master key of the cryptosystem is $512 + 160 = 672$ bits long and consists of two components. The first component is the SHA-512 hash of the input image while the second component consists of the initial conditions of the Lorenz chaotic system x_0 , y_0 , and z_0 each specified with 16-digit decimal precision [15], resulting in 160 bits.

The user is given the opportunity to enter some optional but useful fields such as a name (e.g., doctor's name), a password, a timestamp, a salt and/or a nonce, as a string. In this case, these fields are included as part of the key, resulting in an extended key length.

Next, a function concatenates the second component with the user's string (if provided) and computes the SHA-512 hash. This hash is then combined with the first component to form a 1024-bit string, which undergoes a further SHA-512 hashing operation. From the resulting output, fourteen subkeys are derived for use in the encryption and decryption processes.

This technique provides the following security advantages:

- **Unpredictability:** If the image is secret and high-entropy (random enough), its hash will also be unpredictable. But if the image is public or guessable (like a photo or a common test image), an attacker can try candidate images, compute their hashes, and attempt an attack.
- **Non-reversibility:** A strong cryptographic hash (e.g., SHA-256, SHA-512) makes it infeasible to recover the image from the hash.

- **Obscurity:** The relationship between the cipher image and the fourteen subkeys is complex, obscure, and non-invertible.
- **Extra randomness:** Without extra randomness (e.g., a salt), an attacker can precompute hashes for likely images.
- **Uniqueness:** Each key is unique, even if it concerns the same image.
- **Resistance to differential attacks:** even the slightest change in the plaintext images (typically, a one-bit modification) will result in a totally different encrypted image [19].

3.2. Stage 2: Generation of Chaotic Pseudorandom Numbers

3.2.1. Lorenz Chaotic System

The Lorenz system is a set of three ordinary differential equations that describe the behavior of a simplified model of fluid convection in the atmosphere. It was introduced by Edward Lorenz in 1963 and is one of the most famous examples of a chaotic system. The Lorenz system has been extensively studied in chaos theory and dynamical system modeling due to its attractor, which exhibits a distinctive two-wing structure resembling a butterfly.

The Lorenz chaotic system is a three-dimensional nonlinear system of three coupled non-linear ordinary differential equations. For certain parameter values, it exhibits strong chaotic behavior and high sensitivity to initial conditions; therefore, it has been widely used as a PRNG in image encryption [14,23,37]. The Lorenz system is defined by Equation (1) [13,15].

$$\begin{aligned}\frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= x(\rho - z) - y \\ \frac{dz}{dt} &= xy - \beta z\end{aligned}\tag{1}$$

Typical parameter values for the Lorenz system are:

$$\begin{aligned}\sigma &= 10, \\ \rho &= 28, \\ \beta &= \frac{8}{3}.\end{aligned}\tag{2}$$

In the bibliography, these parameters are sometimes referred to as $a = 10$, $b = 8/3$, $c = 28$ (for instance, [23]).

With these values the system exhibits chaotic behavior, forming the famous Lorenz attractor, visualized as a butterfly-shaped trajectory in 3D space (see Figure 2).

The Lorenz system is 3-dimensional, so it has 3 Lyapunov exponents [34]: $\lambda_1 \geq \lambda_2 \geq \lambda_3$.

- **Largest Lyapunov exponent (λ_1):** Measures the average exponential divergence of nearby trajectories. If positive, the system is chaotic.
- **Second Lyapunov exponent (λ_2):** Typically zero for continuous-time systems, corresponding to perturbations along the flow direction (time evolution).
- **Smallest Lyapunov exponent (λ_3):** Strongly negative, reflecting contraction onto the attractor.

The Lorenz chaotic system was selected for its popularity, simplicity, and speed. It has only one positive Lyapunov exponent, a fact which facilitates the optimization process. Ibrahim et al. have found that the Baker map and the Hénon map, both of which are simple two-dimensional discrete-time dynamical systems, achieve better values throughput in the encryption and decryption process than others composed of multiple simple chaotic

systems. In other words, the enhanced chaotic performance usually comes at the cost of increased computational time [6].

But on the other hand, 2D chaotic systems (like the Hénon map) have relatively simple attractors; their structure can often be reconstructed using standard nonlinear time-series techniques (phase-space embedding, delay coordinate reconstruction). Several research papers prove that the Hénon map (and similar low-dimensional chaotic systems) and their combinations are not cryptographically secure by themselves (for instance, see [38–40]).

Three-dimensional (3D) chaotic systems are inherently more complex than their lower-dimensional counterparts, making phase-space reconstruction significantly more challenging. With a greater number of tunable parameters, 3D systems offer a substantially larger effective key space compared to two-dimensional (2D) maps. Well-known examples such as the Lorenz, Chen, Lü, and Chua systems, along with their hyperchaotic extensions, exhibit richer dynamical behavior, including higher entropy rates, multiple positive Lyapunov exponents, and more complex attractor structures. This increased complexity enhances the unpredictability of chaotic signals and makes parameter estimation and system reconstruction considerably more difficult, particularly from limited observational data.

In contrast, 2D systems are generally more vulnerable to standard cryptanalytic techniques. In other words, while enhanced chaotic performance often correlates with improved complexity and randomness, it may also introduce challenges that could potentially compromise security if not carefully managed. Thus, superior chaotic behavior does not always guarantee stronger cryptographic security; it may come at the cost of increased susceptibility under certain attack models.

3.2.2. Generation of Quality Chaotic Sequences

In this stage, a simulation of Lorenz chaotic system is used to produce a series of pseudorandom numbers of the same type as the 3D image (uint16) to be used in XOR-based encryption of the original image.

In many contexts including cryptography, a higher Lyapunov exponent is associated with higher-quality chaos, especially when unpredictability and complexity are desired traits [14,24].

The Lyapunov exponent quantifies the rate at which nearby trajectories in a dynamical system diverge over time. In simpler terms, it measures how sensitive a system is to initial conditions—a defining characteristic of chaos. A positive Lyapunov exponent indicates chaos: small differences in initial conditions grow exponentially. The larger the Lyapunov exponent, the faster the divergence, implying stronger sensitivity to initial conditions.

From a dynamical systems perspective, a high Lyapunov exponent suggests that:

- The system exhibits strong chaotic behavior.
- The system has better mixing properties, meaning that it explores its state space more thoroughly and rapidly.
- The dynamics are less predictable, which can be desirable in certain applications like cryptography and random number generation.
- High Lyapunov exponent improves diffusion and resistance to prediction.

In the proposed system, a genetic algorithm and a random walk are employed to identify sets of Lorenz system parameters σ , ρ , and β that maximize the Lyapunov exponent λ_1 . This approach serves two purposes: first, to enhance the quality of the generated chaotic sequences by promoting stronger chaotic behavior; and second, to avoid using the well-known standard parameter values with unpredictable alternatives, thereby improving security.

Several publications about numerical approximation methods for calculating Lyapunov exponents exist (for instance, see [41]).

3.2.3. Optimization Using a Genetic Algorithm

The genetic algorithm parameters are presented below. The ranges for the initial values of Lorenz parameters were set as follows:

- σ range = [5, 25];
- ρ range = [10, 50];
- β range = [2, 4].

The number of individuals (population size) is set to 20. The fitness function is the Lyapunov exponent of the Lorenz system. A single-point crossover is used. For two parent solutions, the code picks a random crossover point between the 3 parameters and swaps the parameter values beyond that point. With a probability of 10% (mutation rate = 0.1), random noise is added to the parameters. This helps explore new regions of the parameter space. The parameters are then clamped within valid ranges. Tournament selection is used for selecting parents. Two individuals are chosen randomly, and the one with the higher fitness (Lyapunov exponent) survives. The algorithm terminates after a fixed number of generations (50).

Optimization generates sets of parameters for the Lorenz system (σ , ρ and β) that exhibit strong chaotic behavior, yielding high-quality pseudorandom sequences suitable for encryption. This chaotic behavior is characterized by a high Lyapunov exponents, which serves as the optimization criterion. Notably, the achieved Lyapunov exponents are significantly larger than those obtained with typical initial parameters for the Lorenz system. ($\sigma = 10$, $\rho = 28$, $\beta = 8/3$). For instance, with initial values $x_0 = 1$, $y_0 = 1$, $z_0 = 1$, the typical values for Lorenz system parameters give a pseudorandom sequence with Lyapunov exponent equal to 0.48171, but Lorenz parameters $\sigma = 5.971$, $\rho = 50.000$, and $\beta = 3.972$ give a pseudorandom sequence with Lyapunov exponent equal to 1.143, while Lorenz parameters $\sigma = 5.867$, $\rho = 50.000$, and $\beta = 3.927$ give a pseudorandom sequence with Lyapunov exponent equal to 1.148. Figure 2 presents the Lorenz attractor produced by the genetic algorithm.

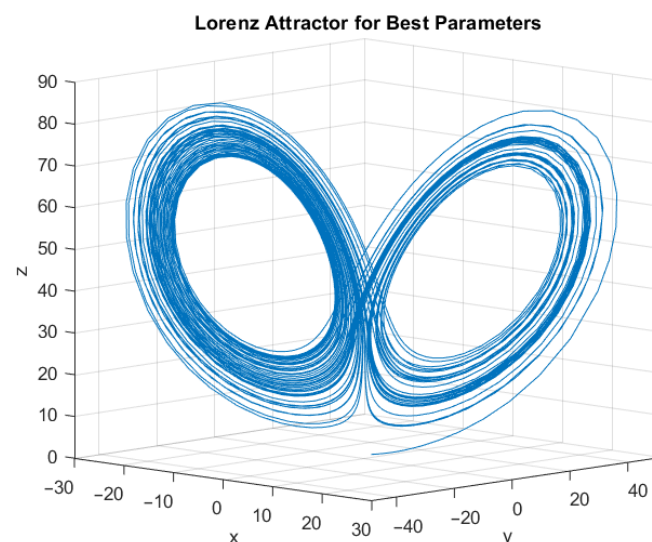


Figure 2. Lorenz attractor produced by the genetic algorithm.

3.2.4. Optimization Using Random Walk

Quality chaos is also achieved via random walk. The starting values of the Lorenz parameters are $\sigma = 10$, $\rho = 28$, $\beta = 8/3$.

The code runs for 40 iterations. A function calculates a numerical approximation of the Lyapunov exponent for the current parameter guess. Acceptance Rule: a solution is

accepted only if the new parameters increase the Lyapunov exponent (greedy selection). The parameter values and their corresponding Lyapunov exponents are stored. The best solution is selected by finding the maximum Lyapunov exponent after the loop.

The obtained Lyapunov exponents were larger than those achieved by the typical parameters but smaller than those achieved by the genetic algorithm; for instance, for $\sigma = 6.7749$, $\rho = 48.7658$, and $\beta = 3.729$, we got a Lyapunov exponent equal to 1.0728. The initial values produced by the random walk generate results comparable to those by using the initial values produced by the genetic algorithm.

Figure 3 presents the steps of a random walk optimizing the Lyapunov exponent.

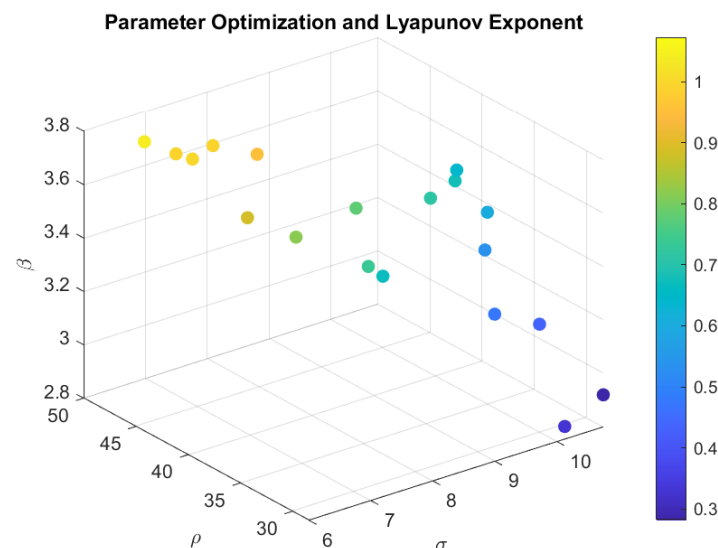


Figure 3. A random walk optimizing the Lyapunov exponent.

Figure 4 presents the Lorenz parameters versus time during the random walk optimizing the Lyapunov exponent.

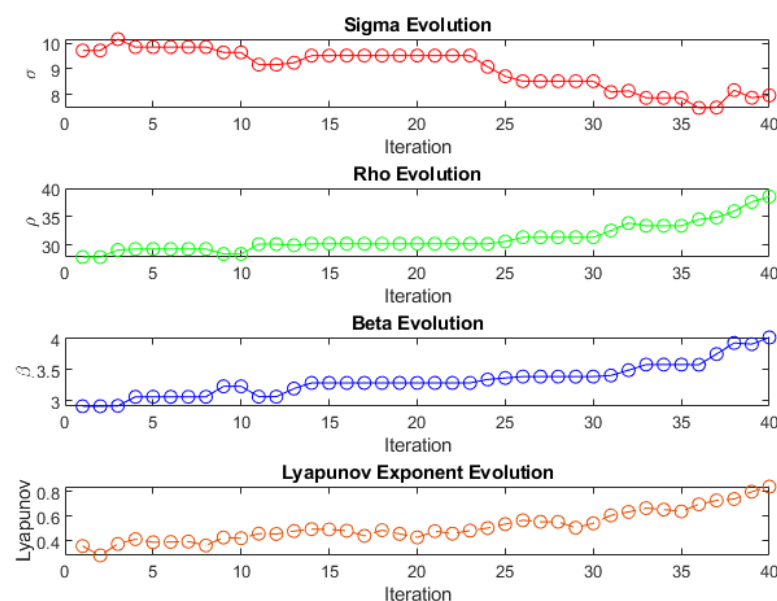


Figure 4. Lorenz parameters evolution during the random walk.

Figure 5 presents the Lorenz attractor produced by the random walk optimization result.

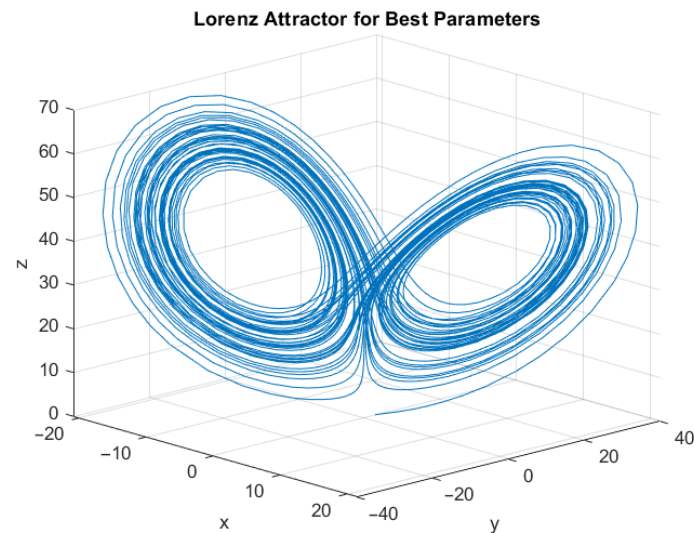


Figure 5. Lorenz attractor produced by the random walk.

The pseudorandom sequence is created from a combination (function) of the chaotic variables, rather than a single variable. From this sequence, a key stream equal to the number of voxels of the 3D image is extracted and shaped after the three-dimensional image to be used in the next stage. As an extra security measure, the pseudorandom sequence used in stage 3 starts from a random number S_L (instead of the first number). The Lorenz parameters used, as well as S_L , make part of the key.

3.3. Stage 3: XOR Encryption of the 3D Image with Chaotic Noise

In this stage, a Lorenz chaotic pseudorandom sequence of length equal to that of the 3D image is selected and shaped after the 3D image dimensions. Then, the 3D image is XORed with the chaotic sequence. ECB-encryption mode is applied [3].

3.4. Stage 4: Shuffling of the ECB-Encrypted Image

In this stage, the voxels of the encrypted image are shuffled to enhance diffusion. To achieve this, the 3D volume is first reshaped into a 1D vector; the vector elements (voxels) are then randomly permuted; finally, the vector is reshaped back into the original 3D volume. Shuffling does not alter voxel values, it only conceals the spatial structure of the image, enhancing diffusion. According to Blesa and Serón, pixel shuffling can be effective in resisting brute-force attacks when the number of possible pixel permutations is quite large [42].

Key-based shuffling surpasses fixed straight P-boxes used frequently in cryptography [43], by providing not only diffusion but also secrecy. Since the permutation is derived from the key and varies with each input image, the pattern remains unknown and input-dependent, significantly improving resistance against cryptanalysis.

3.5. Stage 5: Generation of the CBC Encryption Key

In this step, pseudorandom numbers of length and magnitude range equal to the encrypted image are generated, in order to be used in the next stage.

3.6. Stage 6: XOR Encryption with Pseudorandom Numbers in CBC Mode

The result of stage 4 is further XOR-encrypted with the sequence created in stage 5, but in CBC mode this time. The pseudorandom numbers are produced by a PRNG for extra security. In CBC mode, each plaintext block (16 bits) is XORed with the previous

ciphertext block prior to encryption (see equation (3)). The first block is XORed with a random initialization vector (IV) which makes part of the key [3]:

$$C_i = (P_i \oplus C_{i-1}) \oplus K_i \quad (3)$$

where:

- C_i : ciphertext block;
- P_i : plaintext block;
- C_{i-1} : previous ciphertext block (or IV for the first block);
- K_i : subkey K_i , a pseudorandom sequence block;
- $\oplus$: XOR operator.

Figure 6 presents the CBC encryption mode.

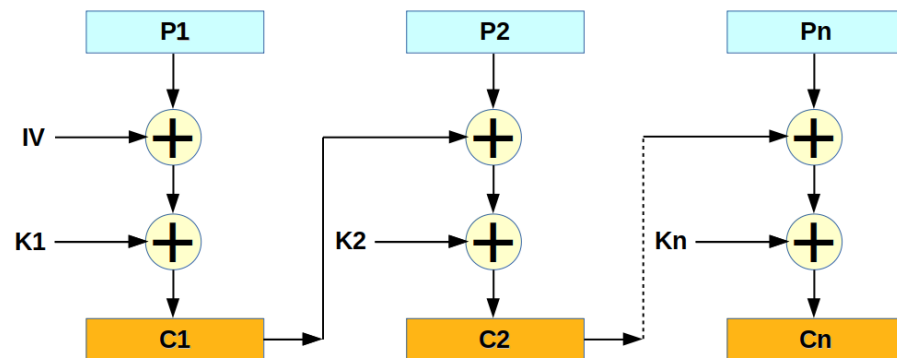


Figure 6. CBC encryption mode.

CBC mode significantly improves key sensitivity and security compared to ECB. In ECB (Electronic CodeBook) mode:

- Each block is encrypted independently using the same key.
- Identical plaintext blocks produce identical ciphertext blocks; this makes it vulnerable to pattern leakage.
- ECB mode has poor diffusion: changing one bit in plaintext affects only one block in ciphertext.
- Small changes in key affect ciphertext only where that key is used, not across blocks.

On the other hand, CBC assumes stronger key sensitivity and diffusion:

- Chaining effect: Each ciphertext block depends on the previous ciphertext block. So, a change in plaintext or key affects all subsequent ciphertext blocks. CBC improves diffusion because each ciphertext block depends on all previous plaintext blocks; a small change in a pixel will propagate to all subsequent ciphertext blocks. This introduces *avalanche* behavior, which is the goal of diffusion [3].
- Better key sensitivity: A small change in key affects not only the block where it is used for but also propagates via chaining. This makes CBC much more sensitive to key changes than ECB.
- Stronger avalanche effect: A 1-bit change in plaintext or key leads to a completely different ciphertext for that and all following blocks. This is a desirable property in secure encryption.

Using CBC with just XOR is suitable for lightweight image encryption where speed matters more than strong security, ideal for sensitive or high-stakes data.

3.7. Stage 7: Shuffling of All Voxels Randomly

In this stage, the voxels of the encrypted image are shuffled once more, to increase diffusion.

3.8. Stage 8: Rotation of All Voxels Randomly to the Left

In this stage, each 16-bit pixel is rotated left by a random number of places, different for each pixel. Random voxel rotation means rotation of the bits by a random number of places, different for each voxel. Let us give an example: original voxel (16-bit): 1110101000011000; rotated left by 3: 0101000011000111.

3.9. Use of Subkeys per Stage

Stage 1: Preprocessing: No subkeys are needed.

Stage 2: Generation of chaotic pseudorandom sequence: 4 subkeys are used: 1, 2, 3, 4. Subkey 1 selects the set of the initial Lorenz parameter values that will be used from a precomputed set (with 3 decimal digits), while subkeys 2, 3, 4 set the rest 10 decimal digits per parameter.

Stage 3: ECB encryption: Subkey 5 selects a starting point in the chaotic pseudorandom sequence S_L .

Stage 4: Shuffle the ECB-encrypted image. Two subkeys are used as follows: subkey 6 selects the algorithm from a set of 8 possible choices available in MATLAB, while subkey 7 sets the seed. It is necessary to use a specific seed for reproducibility during decryption.

Stage 5: PRNG generation: Subkey 8 selects the PRNG algorithm; subkey 9 sets the seed.

Stage 6: CBC Encryption: Subkey 10 generates the IV.

Stage 7: Shuffle all pixels. Subkey 11 selects the PRNG algorithm, while subkey 12 sets the seed.

Stage 8: Rotate pixels randomly to the left. Subkey 13 selects the RNG algorithm while subkey 14 sets the seed.

On the decryption side, the inverse operations are applied in reverse order to restore the original image.

For extra security, the PRNG algorithms and the corresponding seeds are randomly selected. In MATLAB, the following PRNG algorithms are available to choose from:

1. Twister
2. SimdTwister
3. CombRecursive
4. MultFibonacci
5. V5uniform
6. V4
7. Philox
8. Threefry

Other powerful and cryptographically secure PRNG algorithms such as ChaCha20 [44,45] or AES-CTR [46,47] can also be implemented in MATLAB via custom code.

4. Experimental Results

Results from the processing of three test images will be presented in this section. To enhance clarity and facilitate interpretation, we also present results using 2D cross-sectional slices, which provide detailed visualization that is often more interpretable than full 3D renderings.

One such case is the surface plots generated by MATLAB's 'surf' function. While these plots display a 3D surface in three-dimensional space, they are based on a 2D data grid defined by X, Y, and corresponding Z values. Since the 'surf' function cannot accept 3D arrays, we extract representative 2D cross-sectional slices from the 3D data and visualize

them as separate surface plots. This approach allows for an effective representation of volumetric data in a comprehensible 2D-to-3D format.

4.1. Test Image 1

Test image 1 is MATLAB's MRI test image "D", a portion of a 3D skull in uint8 format, sized $128 \times 128 \times 27$, with values ranging from 0 to 88. After preprocessing, the test image 1 size becomes $111 \times 110 \times 27$ voxels, ranging from 0 to 65,535. The elimination of null slices leads to a reduction of 4174 voxels corresponding to 25.48%.

4.1.1. Whole Test Image 1

Figure 7 presents a 3D view of test image 1.

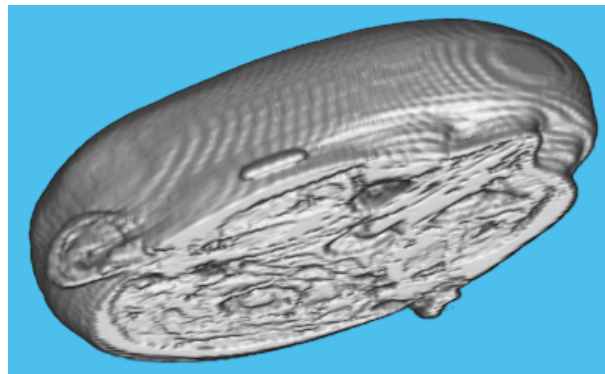


Figure 7. Three-dimensional view of test image 1.

Figure 8 presents the original, encrypted, and decrypted 3D test image 1.

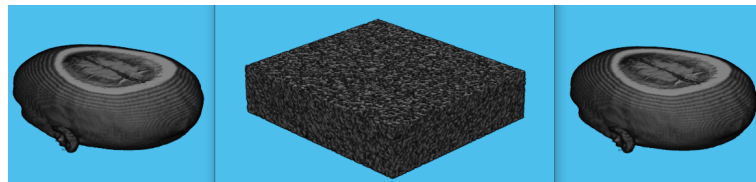


Figure 8. Original, encrypted, and decrypted test image 1.

Figure 9 presents the histograms of the original, encrypted, and decrypted 3D image 1.

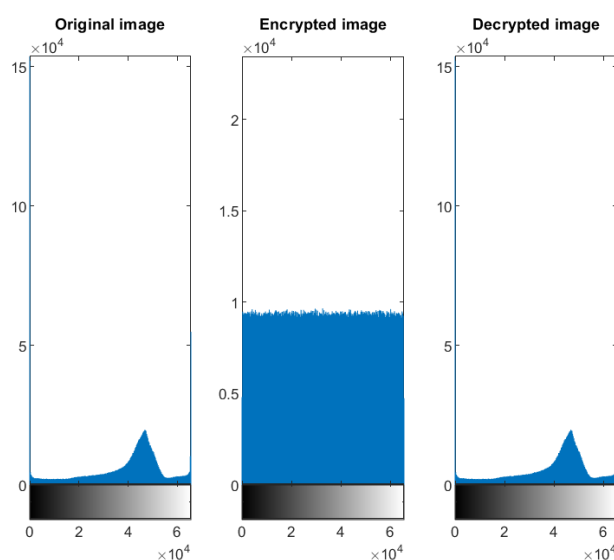


Figure 9. Histograms of original, encrypted, and decrypted test image 1.

4.1.2. Slice 5 of Test Image 1

Next, we show the procedure on a 2D slice of test image 1.

Figure 10 presents two slices of test image 1. The black space around the skull corresponds to 0 voxel values.

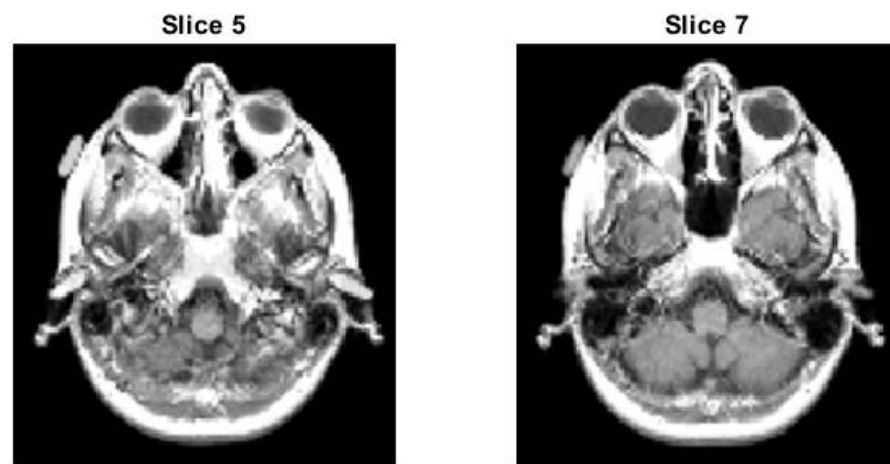


Figure 10. Two slices of test image 1.

Let us now focus on slice 5. Figure 11 presents the original, encrypted, and decrypted slice 5 of test image 1.

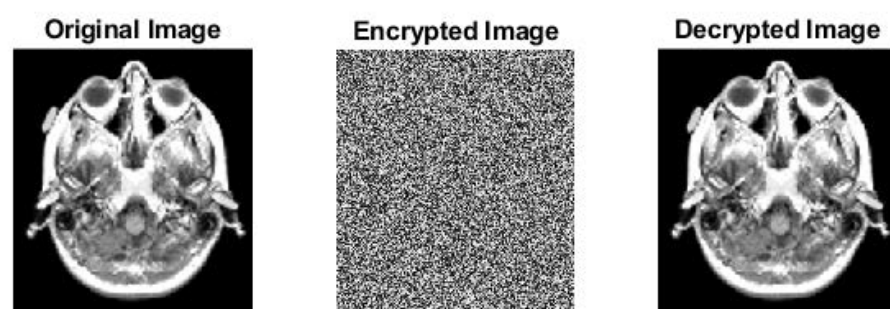


Figure 11. Original, encrypted, and decrypted slice 5 of test image 1.

Figure 12 presents the histogram of the original, encrypted and decrypted slice 5 of test image 1.

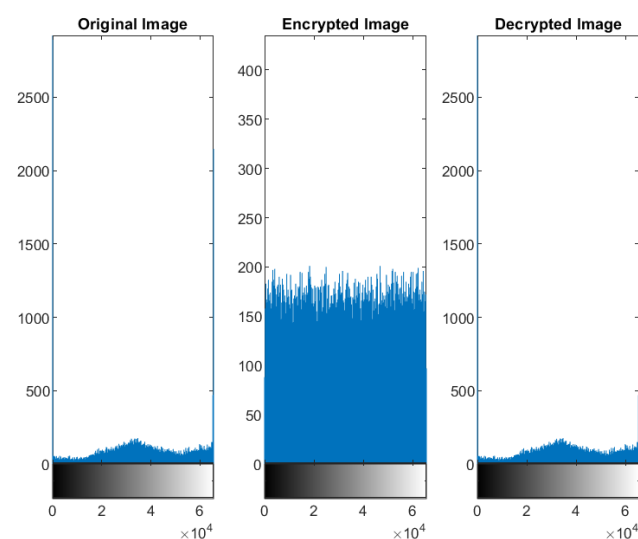


Figure 12. Histogram of original, encrypted, and decrypted slice 5 of test image 1.

Figure 13 presents the surface diagram of slice 5. The colors encode voxel values.

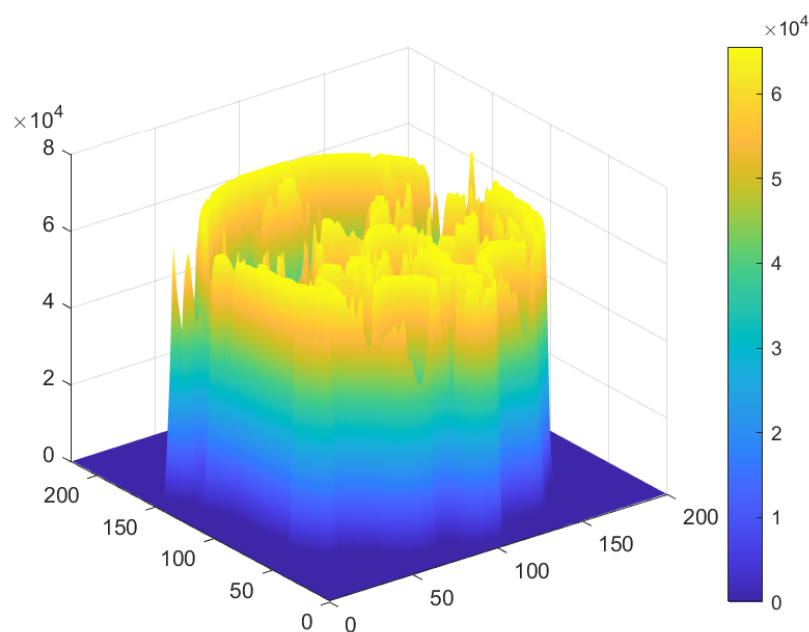


Figure 13. Surface diagram of slice 5 of test image 1.

Figure 14 presents the surface diagram of encrypted slice 5 of test image 1. As can be observed, it is quasi-uniform.

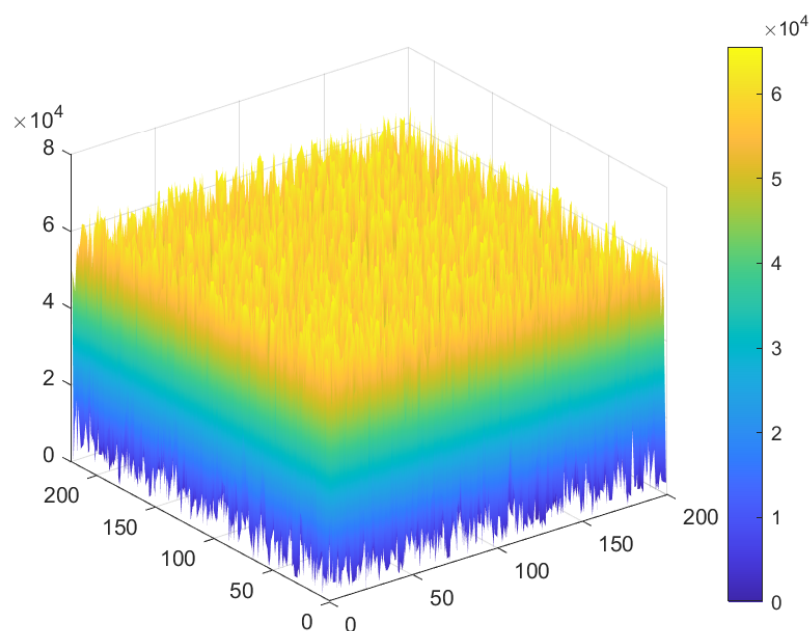


Figure 14. Surface diagram of encrypted slice 5 of test image 1.

Finally, Figure 15 presents the surface diagram of recovered slice 5.

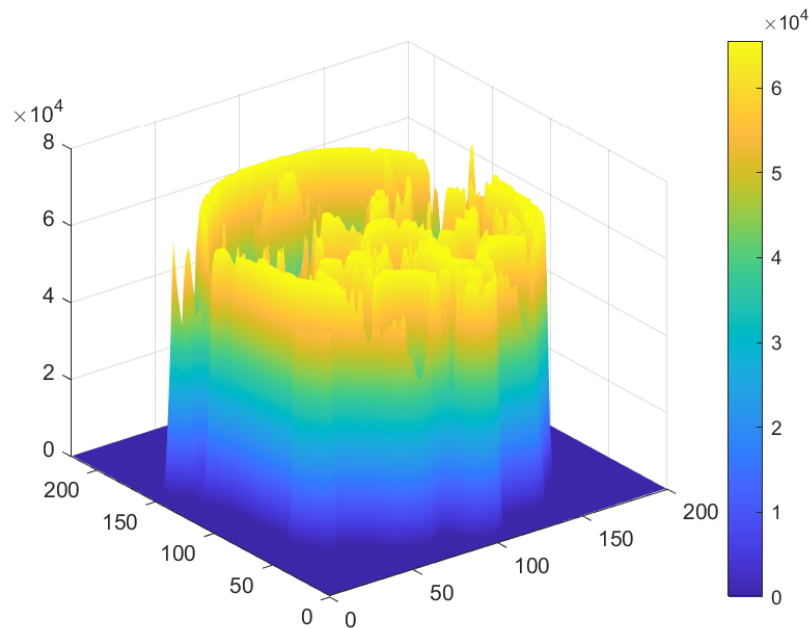


Figure 15. Surface diagram of decrypted slice 5 of test image 1.

4.2. Test Image 2

Test image 2 is a 3D MRI volume of a skull stored in NIfTI format, with voxel values ranging from -4.7094 to 7.198 , sized as $53 \times 63 \times 46$ voxels.

During preprocessing, the voxel values are converted from floating-point numbers to 16-bit unsigned integers (uint16). The empty space corresponding to 0.000 voxel value in single format is now mapped to integer 25,919; hence, the plane of null voxel values, previously located at $z=0$, gets shifted to $z = 25,919$.

After removing null slices, the resulting 3D volume has dimensions of $44 \times 51 \times 35$, corresponding to a voxel-count reduction of 48.8652%. The resolution is then doubled in each dimension, yielding a final volume of size $88 \times 102 \times 70$ voxels. Since each voxel is stored using 16 bits, the total size of test image 2 is 10,053,120 bits.

4.2.1. Whole Test Image 2

Figure 16 presents a 3D view of test image 2. MATLAB's volume viewer tool with the following rendering options was used: `alphamap = ct-mip` and `colormap = hsv`.

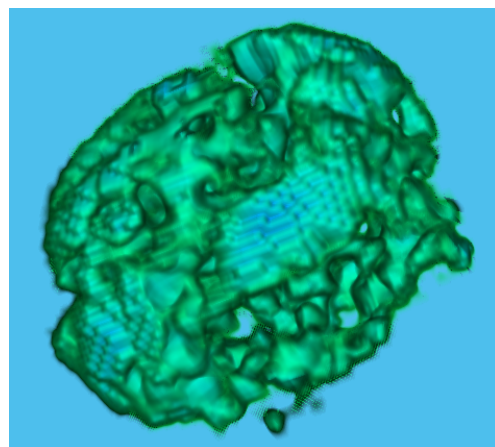


Figure 16. Three-dimensional view of test image 2.

Figure 17 presents a 3D view of test image 2, as viewed by MATLAB's volume viewer tool.

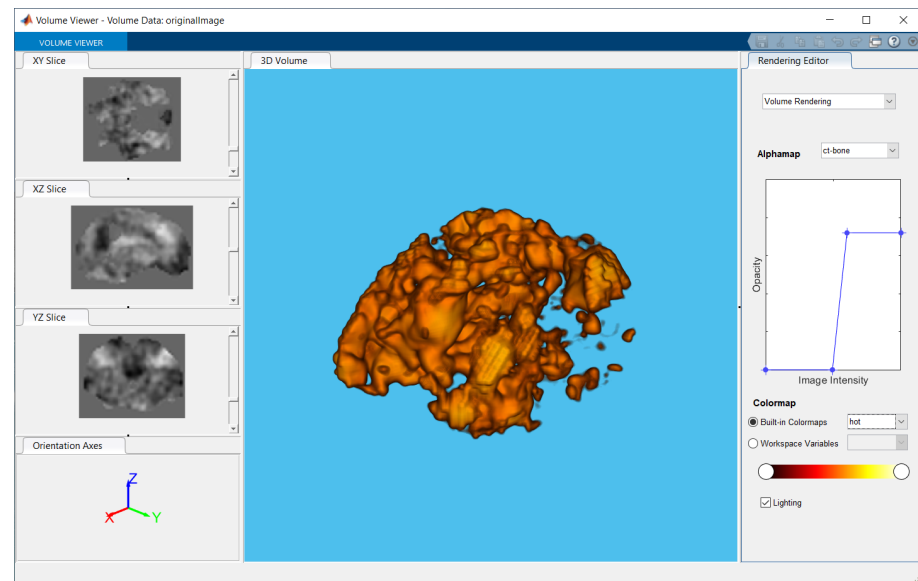


Figure 17. Three-dimensional view of test image 2 in MATLAB's volume viewer tool.

Figure 18 presents the original, encrypted, and decrypted 3D test image 2.

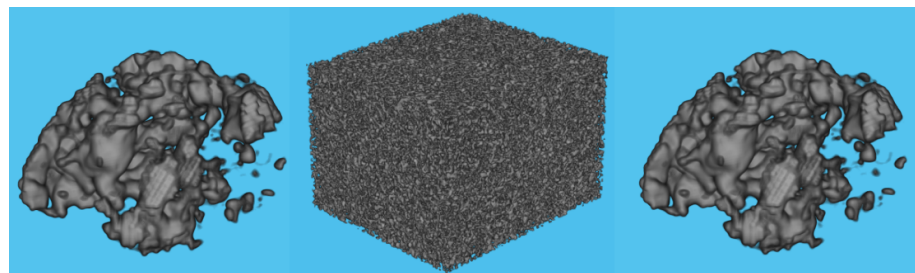


Figure 18. Original, encrypted, and decrypted test image 2.

Figure 19 presents the histograms of the original, encrypted, and decrypted 3D test image 2. The original histogram peaks at value 25,919 instead of 0 because voxel values have been shifted during preprocessing.

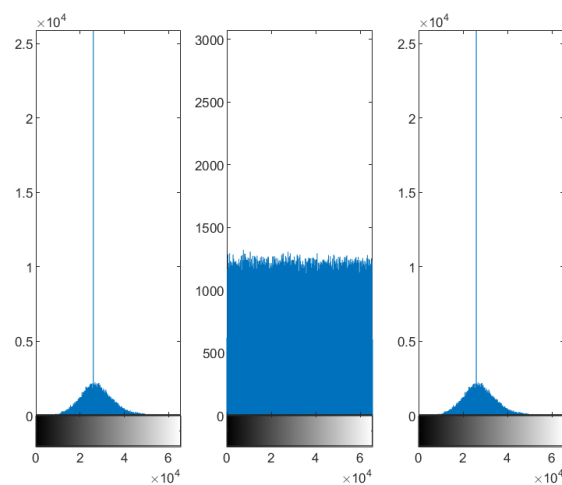


Figure 19. Histograms of original, encrypted, and decrypted test image 2.

4.2.2. Slice 13 of Test Image 2

Let us now move to the 2D slice level. Figure 20 shows two slices of test image 2.

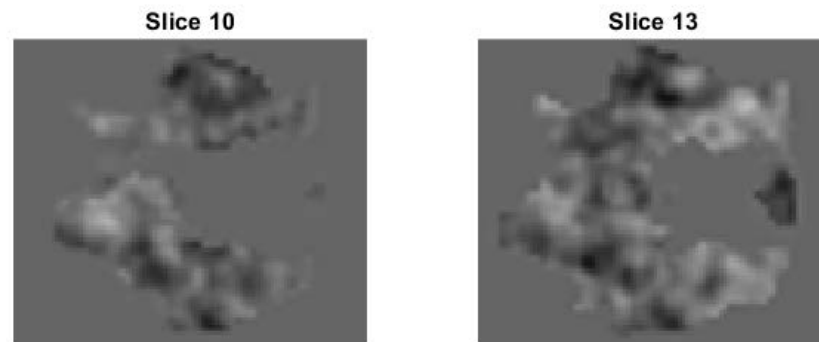


Figure 20. Two slices of test image 2.

Next, we focus on slice 13 of test image 2. Figure 21 presents the original, encrypted, and decrypted slice 13.



Figure 21. Original, encrypted, and decrypted slice 13 of test image 2.

Figure 22 presents the corresponding histograms of the original, encrypted, and decrypted slice 13 of test image 2.

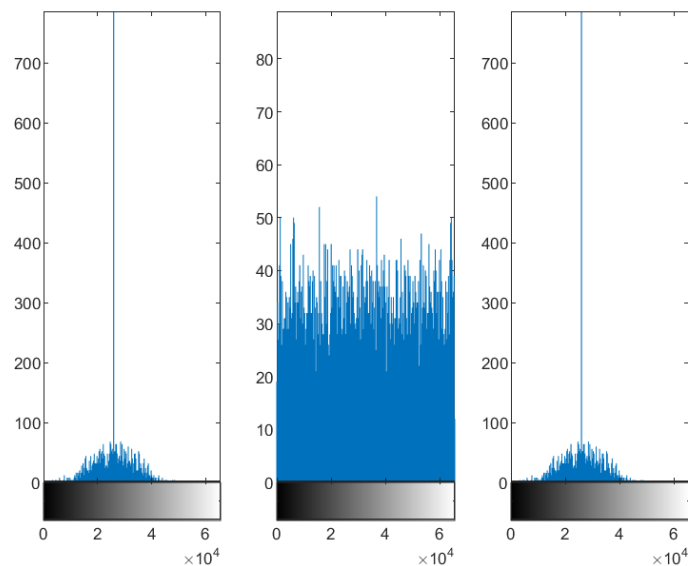


Figure 22. Histograms of original, encrypted, and decrypted slice 13 of test image 2.

Figure 23 presents the surface diagram of slice 13. The null voxel plane is located at $z = 25,919$.

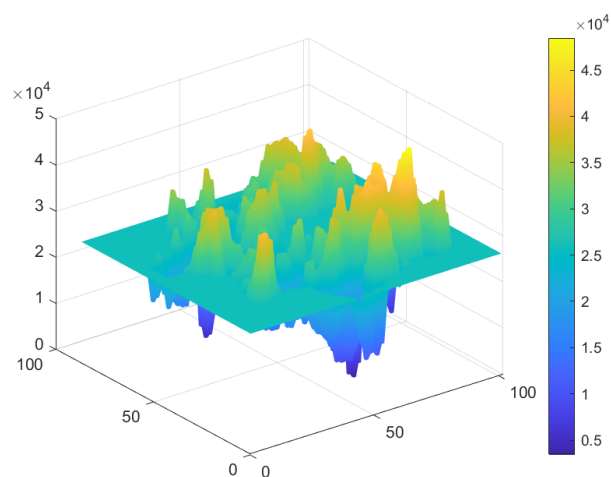


Figure 23. Surface diagram of slice 13 of test image 2.

Figure 24 presents a surface diagram of the encrypted slice 13. As can be observed, it is quasi-uniform.

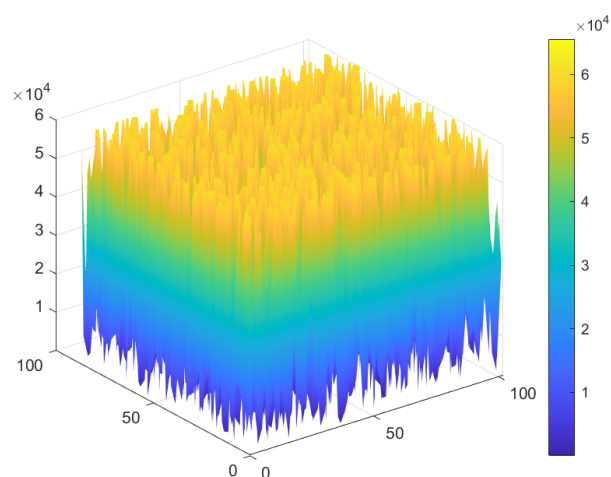


Figure 24. Surface diagram of encrypted slice 13 of test image 2.

Figure 25 presents the surface diagram of recovered slice 13.

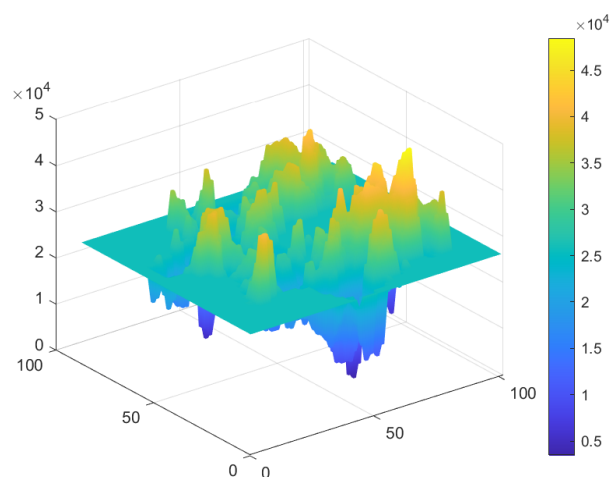


Figure 25. Surface diagram of recovered slice 13 of test image 2.

4.3. Test Image 3

The source of test image 3 is a CT scan of a human body in DICOM format, sized as $512 \times 512 \times 1814$ voxels. It contains a lot of information and its processing exceeds the memory capacity of our computer; therefore, we have extracted a subset of 22 slices to be used as test image 3. So, test image 3 is in DICOM format and contains $512 \times 512 \times 22$ 16-bit voxels or 11,534,336 bytes.

4.3.1. Preprocessing of Test Image 3

Test image 3 presents some of the characteristic peculiarities of medical images: first, the empty space around the circular image is filled with black voxels, overloading the histogram at 0. Second, the useful information is concentrated in a limited zone of intensity ranging from 7000 to 8700. This makes it difficult to distinguish details on the medical image. Figure 26 presents the intensity histogram of test image 3 (left) and a representative axial slice from the 3D volume before preprocessing (right).

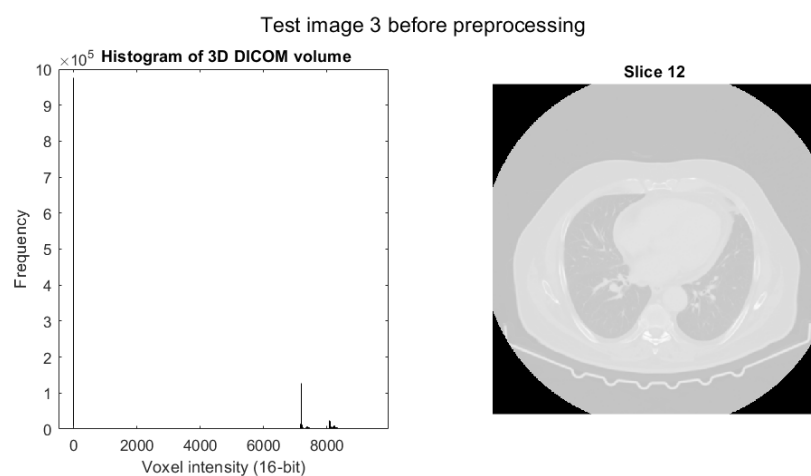


Figure 26. Histogram and slice 12 of test image 3 before preprocessing.

Preprocessing removes null (black) voxels and rescales the intensity values from the range [7000, 8700] to the full dynamic range of [0, 65535]. Figure 27 presents the histogram of test image 3 (left) and the same 2D slice after preprocessing (right).

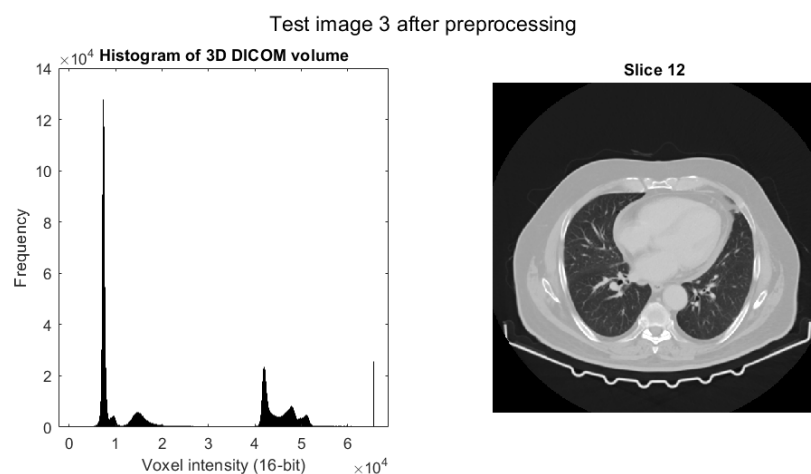


Figure 27. Histogram and slice 12 of test image 3 after preprocessing.

4.3.2. Whole Test Image 3

Figure 28 presents a 3D view of test image 3. The following rendering options were used: alphamap = ct-mip and colormap = mri.

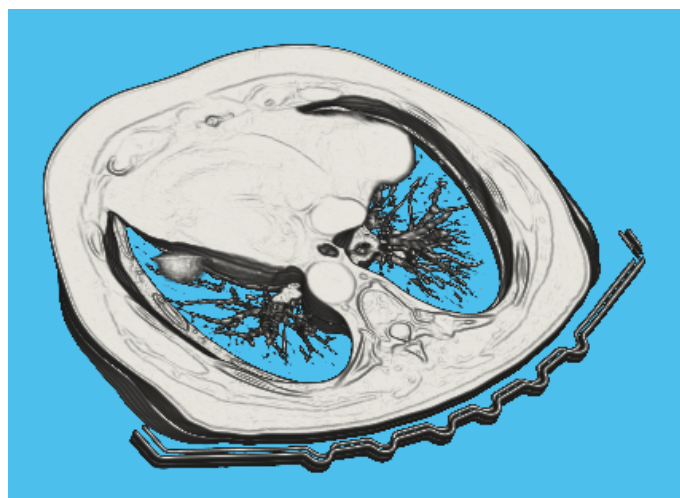


Figure 28. 3D view of test image 3.

Figure 29 presents the original, encrypted, and decrypted 3D test image 3.

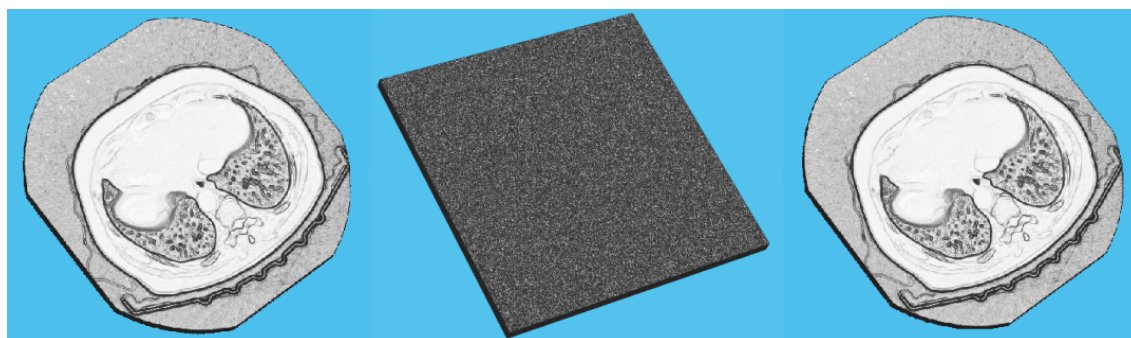


Figure 29. Original, encrypted, and decrypted test image 3.

Figure 30 presents the histograms of the original, encrypted, and decrypted test image 3.

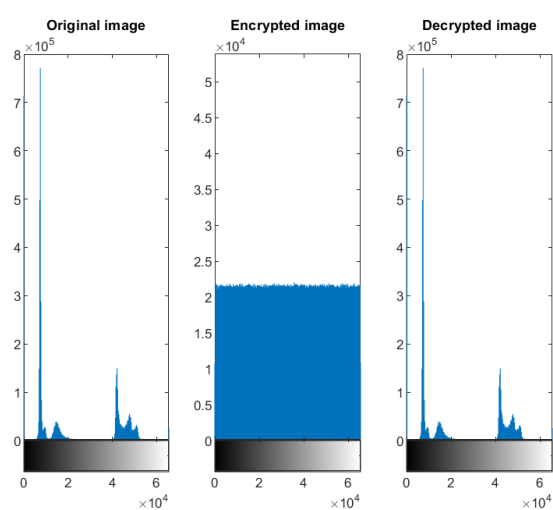


Figure 30. Histograms of original, encrypted and decrypted test image 3.

4.3.3. Test Slice of Test Image 3

Next, we show the procedure on a 2D slice of test image 3.

Figure 31 presents two slices of test image 3.

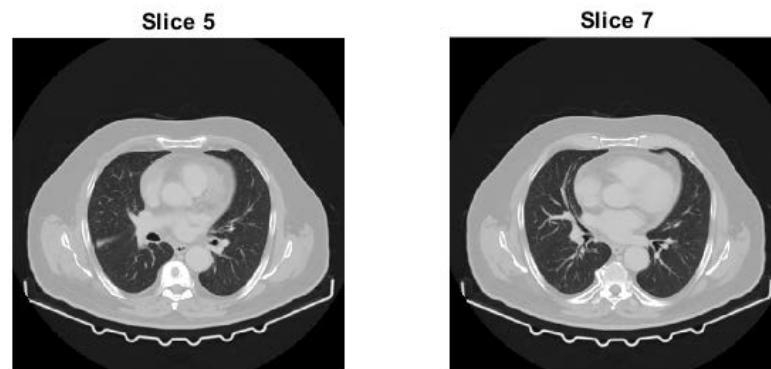


Figure 31. Two slices of test image 3.

Next we focus on slice 5. Figure 32 presents slice 5 of the original, encrypted, and decrypted test image 3.

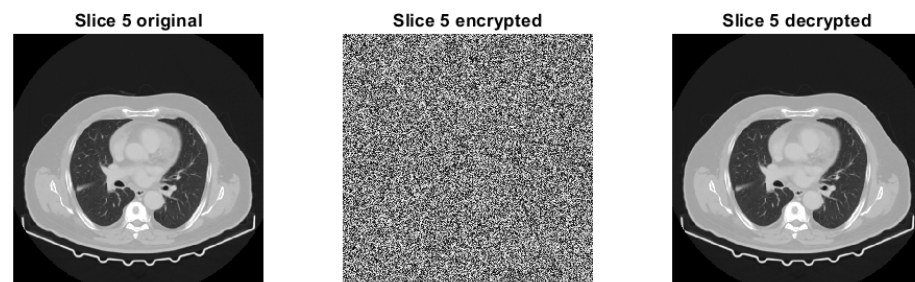


Figure 32. Original, encrypted, and decrypted slice 5 of test image 3.

Figure 33 presents the corresponding histograms of slice 5 of the original, encrypted, and decrypted test image 3.

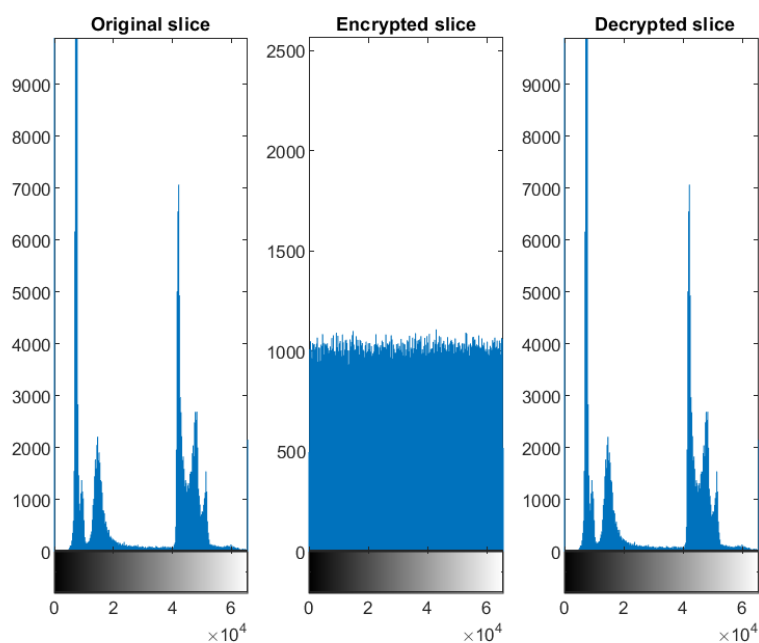


Figure 33. Histogram of slice 5 of test image 3.

Figure 34 presents the surface diagram of slice 5.

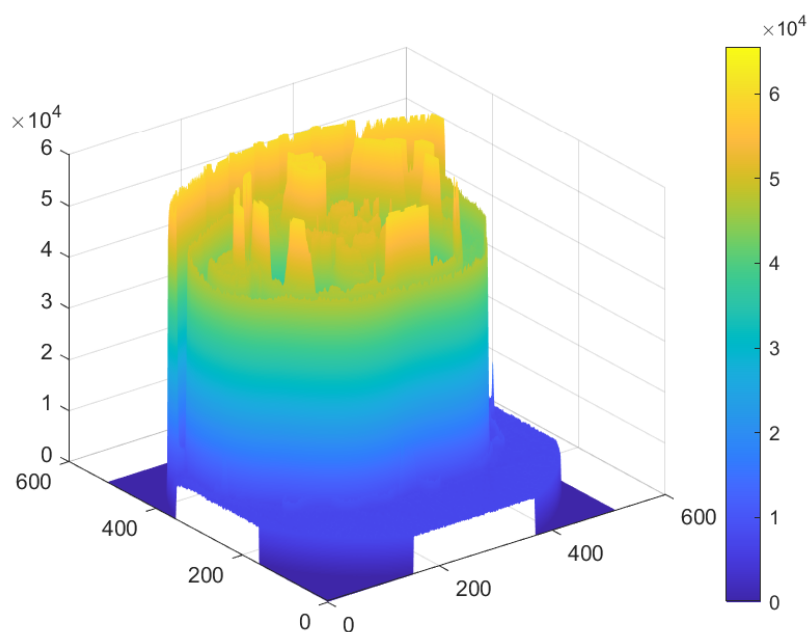


Figure 34. Surface diagram of slice 5 of test image 3.

Figure 35 presents the surface diagram of encrypted slice 5 of test image 3. As can be observed, it is quasi-uniform.

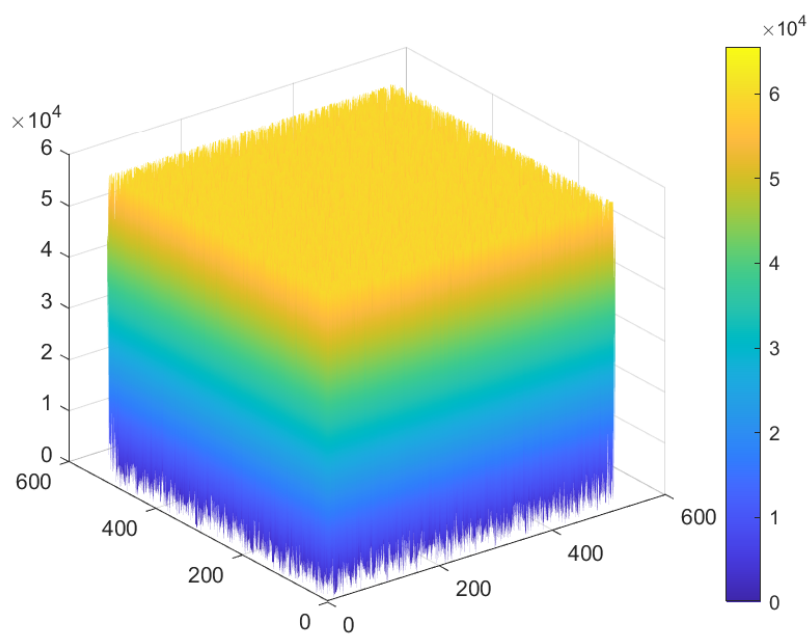


Figure 35. Surface diagram of encrypted slice 5 of test image 3.

Figure 36 presents the surface diagram of recovered slice 5.

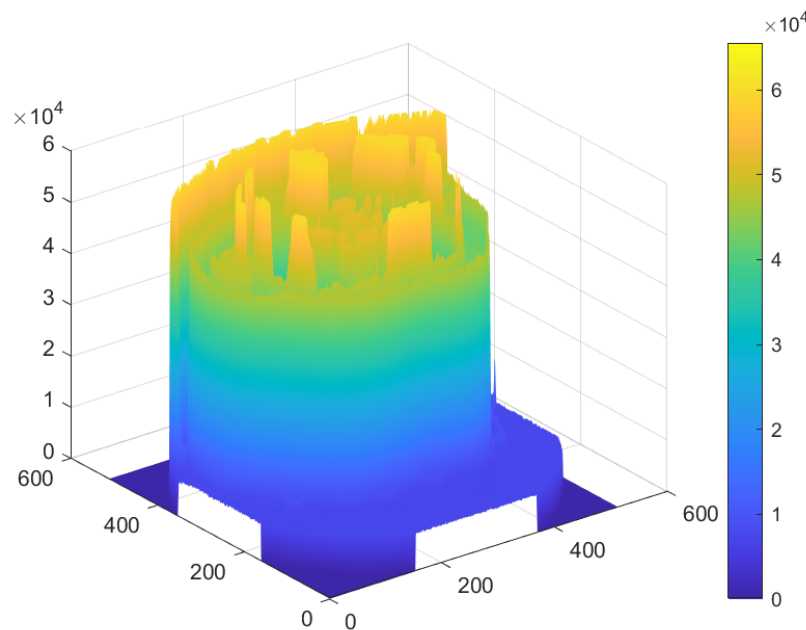


Figure 36. Surface diagram of recovered slice 5 of test image 3.

5. Security Analysis

Security represents a critical concern in cryptographic systems. When introducing a novel cryptosystem, comprehensive security evaluations must invariably accompany the proposal. An effective encryption method ought to withstand various forms of attacks, including cryptanalytic, statistical, and brute-force approaches [48].

In this study, several security evaluations were conducted on the proposed method, encompassing key space analysis, ciphertext distribution assessment, correlation examination of neighboring pixels, information entropy measurement, and plaintext sensitivity testing. The security evaluations reveal that the proposed method achieves a strong level of protection.

5.1. Key Space

The key space denotes the total number of possible encryption keys in a cryptosystem. A sufficiently large key space is crucial to deter brute-force attacks by making exhaustive key search computationally impractical [16].

When the key space of an encryption algorithm reaches 2^{100} , it is generally considered secure against brute-force attacks [49]. But according to [8], a key space must consist of at least 2^{218} elements to ensure that an encryption algorithm can effectively resist brute-force attacks with a sufficient level of security.

The key of the proposed cryptosystem is 672 bits long, resulting in a key space of 2^{672} , which is significantly larger than 2^{218} . Hence, the minimum key space requirements are fulfilled. This ensures strong resistance to brute-force attacks and demonstrates a high level of cryptographic security [13].

Key Sensitivity

Key sensitivity means that even a tiny difference in the encryption or decryption key should result in a completely different output. It is an important metric for evaluating the robustness and security of any cryptographic algorithm.

Key sensitivity measures how much the ciphertext changes when the encryption key is slightly changed, such as flipping one bit or adding 1 to the key.

Step-by-step process:

1. Encrypt plain image with Key1 and get cipher image C1;
2. Flip one bit in Key1 to get Key2;
3. Encrypt plain image with Key2 and get cipher image C2;
4. Compute the difference between C1 and C2.

A common way to quantify the difference is to calculate the PSNR between the two encrypted images. Applying the above process to test image 1 yields the below results.

The PSNR between the two encrypted images is 7.78 dB and the key sensitivity is 50.01%, meaning that half of the voxels changed with the use of Key2.

Figure 37 presents slice 5 of test image 1 and the two cipher images C1 and C2.

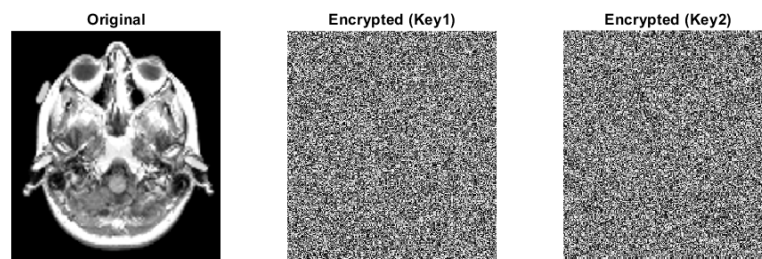


Figure 37. Original and two cipher images of slice 5 of test image 1.

Figure 38 presents the difference between the two encrypted images of slice 5 and the corresponding histogram as is and after removing pixels valued 0. Black pixels indicate no difference between the two encrypted images.

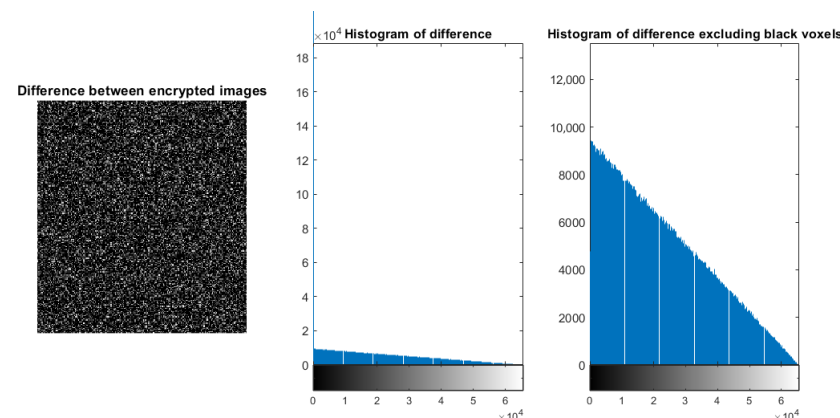


Figure 38. Image and histograms of the difference between the two encrypted images of test image 1.

5.2. Distribution of the Encrypted Image Voxels

An image histogram illustrates the distribution of pixels/voxels by plotting their frequency. As demonstrated, the histograms of the encrypted images 1, 2 and 3 are nearly uniform, offering no discernible patterns and thereby resisting statistical analysis attacks [48].

5.3. Information Entropy

In the context of image processing, entropy is a measure of randomness or uncertainty in the pixel values of an image. It is often used to evaluate:

- The richness of texture in an image;
- The effectiveness of encryption algorithms;

- The information content in an image.

A high quality encrypted 8-bit grayscale image should have entropy close to 8, while a high-quality encrypted 16-bit grayscale image should have an entropy close to 16.

Shannon's formula for entropy is:

$$H(s) = \sum_{i=0}^{2^N-1} P(s_i) \log_2 \frac{1}{P(s_i)} \quad (4)$$

where:

- $P(s_i)$ is the probability of each symbol s_i
- $H(s)$ is the entropy of the source s .

The sum is over all possible symbols s_i and the logarithm base is 2, which is typical in information theory (bits).

5.4. Plain Image Sensitivity Analysis

Attackers often employ the following technique: they make minimal modifications to the original image, and then they encrypt both the original and modified versions using the proposed encryption method. By analyzing the differences between the resulting encrypted images, they attempt to establish correlations between the plaintext and ciphertext. This kind of attack is known as a differential attack [50].

An encryption scheme can resist differential attacks when even a small modification in the plain image results in significant variations in the encrypted image.

To evaluate how a single-pixel modification affects the entire image when processed through the proposed encryption algorithm, two standard metrics are employed: NPCR and UACI [4,48].

5.4.1. NPCR

NPCR stands for Number of Pixel Change Rate. It is a performance metric commonly used in image encryption, steganography, and image processing to evaluate how sensitive an algorithm is to small changes in the input image. A high NPCR value indicates that even a small change in the input leads to a significantly different output, which is desirable in cryptographic applications because it indicates better resistance to differential attacks.

The general NPCR formula for 3D images is:

$$NPCR = \frac{\sum_{i=1}^M \sum_{j=1}^N \sum_{k=1}^P D(i, j, k)}{M \times N \times P} \times 100\% \quad (5)$$

where

$$D(i, j, k) = \begin{cases} 0, & \text{if } C_1(i, j, k) = C_2(i, j, k), \\ 1, & \text{if } C_1(i, j, k) \neq C_2(i, j, k), \end{cases}$$

and $C_1(i, j, k)$ and $C_2(i, j, k)$ are the pixel values at position (i, j, k) in the two 3D images.

5.4.2. UACI

UACI stands for Unified Average Change Intensity. It is a performance metric used in image encryption, steganography, and security analysis. It measures the average intensity difference between two images, typically two encrypted images generated from two slightly different plain images (e.g., one pixel change). Another interpretation is to compare an encrypted image with an encrypted version of a slightly modified input image (e.g., by one pixel).

While NPCR tells us how many pixels changed, UACI tells us how much they changed; i.e., the intensity difference between corresponding pixels. A high UACI value indicates a better resistance to differential attacks. UACI values from 25% to 30% indicate moderate sensitivity, acceptable for basic algorithms. UACI values close to 33% are considered ideal. The ideal value for UACI is $\geq 33.4635\%$, as calculated by Wu et al. [12].

Let:

- $I_1(x,y,z)$: original 3D image;
- $I_2(x,y,z)$: modified 3D image;
- Image dimensions: $M \times N \times P$;
- Max pixel value: $L = 65,535$. Then, the UACI for 3D 16-bit images is:

$$UACI = \frac{1}{M \times N \times P} \sum_{x=1}^M \sum_{y=1}^N \sum_{z=1}^P \left(\frac{|I_1(x,y,z) - I_2(x,y,z)|}{65535} \right) \times 100 \quad (6)$$

Table 2 presents the entropy, NPCR, and UACI metrics for test images 1–3, which are satisfactory.

Table 2. Entropy, NPCR, and UACI results for test images 1–3.

Image	Entropy Original ¹	Entropy Encrypted ¹	NPCR	UACI ²
Test Image 1	10.0388	15.9801	99.9989%	33.34–33.47%
Test Image 2	10.2576	15.9233	99.9987%	33.37–33.47%
Test Image 3	8.0448	15.9914	99.9986%	33.34–33.47%

¹ The voxel resolution is 16 bit. ² Higher values may be achieved after fine-tuning.

5.5. Correlation Analysis

Table 3 presents the autocorrelation results across the three dimensions for test images 1, 2, and 3.

Table 3. Autocorrelation results for test images 1–3.

Image	Dimension	Original	Encrypted
Test Image 1	X	0.9807	−0.0003
	Y	0.9579	0.0000
	Z	0.9706	0.0002
Test Image 2	X	0.9379	−0.0009
	Y	0.9468	0.0007
	Z	0.9295	0.0030
Test Image 3	X	0.9848	0.0003
	Y	0.9890	−0.0005
	Z	0.9308	0.0002

The above results compare very well with the literature [4,15,19,34–36].

Returning to a 2D representation, we present the pixel correlation diagrams of adjacent pixels horizontally, vertically, diagonally and anti-diagonally for slice 5 of test image 1. Figure 39 shows the correlation analysis for the original slice, while Figure 40 displays the results for the encrypted version. Similar results are obtained for the other two test images.

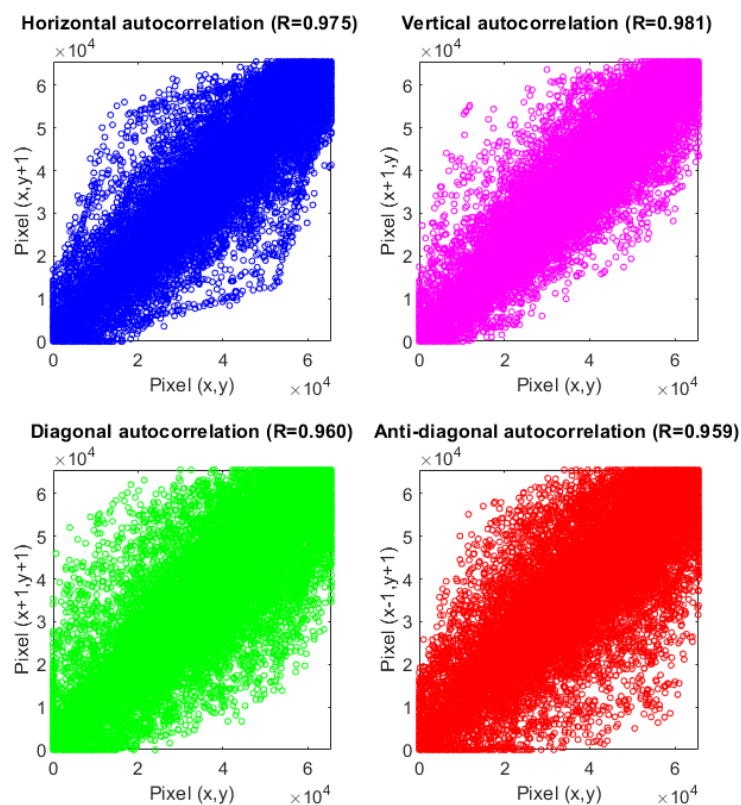


Figure 39. Pixel correlation analysis of slice 5 of the original test image 1.

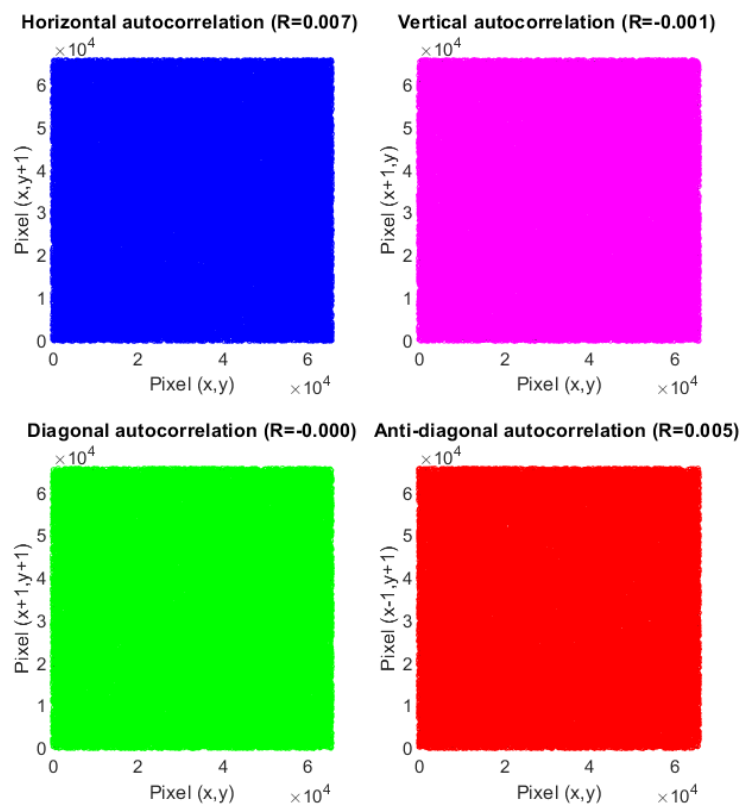


Figure 40. Pixel correlation analysis of slice 5 of the encrypted test image 1.

5.6. Fourier Transformation Analysis

For test image 2, we used an additional method to demonstrate the autocorrelation. Figures 41 and 42 show the Fourier-transformed original and encrypted image 2, respectively. Figure 41 has a bright spot at its center, indicating a 3D high autocorrelation [4].

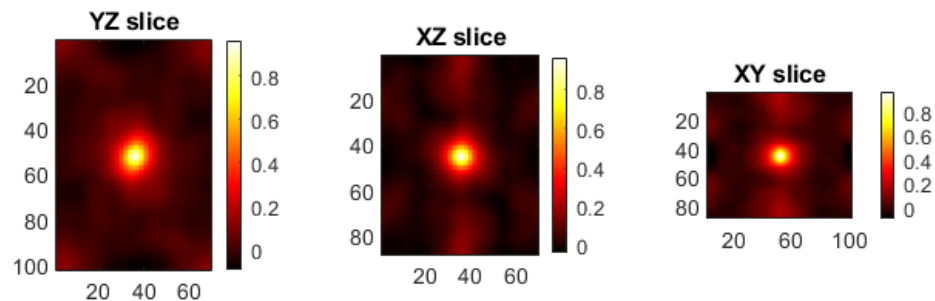


Figure 41. Autocorrelation of the clear test image 2.

In Figure 42 however, there are no deep regions or edges for the encrypted image, a fact which corresponds to the flat histogram of Figure 19 (middle).

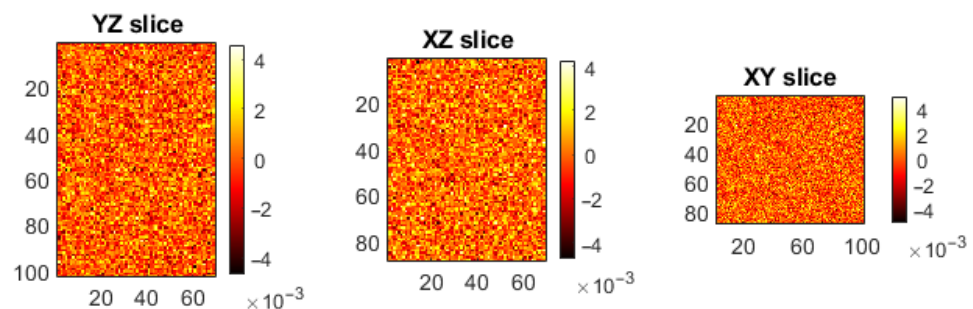


Figure 42. Autocorrelation of the encrypted test image 2.

5.7. Difference Between Encrypted Versions of the Original

Figure 43 presents the difference between the cipher images of two instances of test image 2 that differ in only one voxel and the corresponding histogram. In fact, UACI evaluates how much the two cipher images differ at the voxel level.

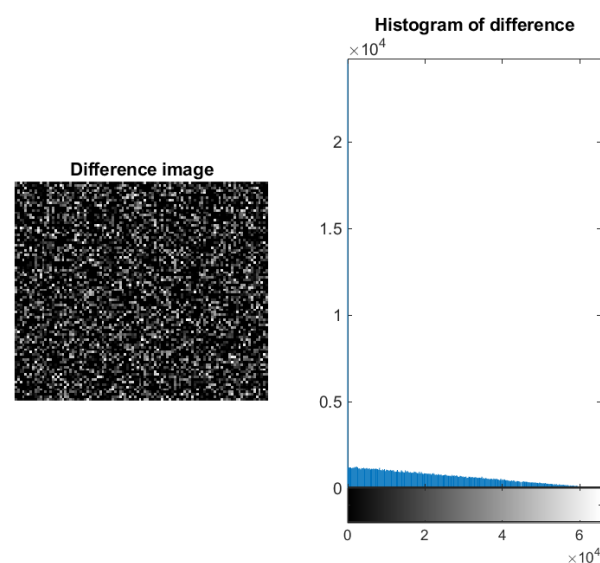


Figure 43. Difference between the encrypted versions of the original test image 2 and test image 2 modified in one voxel.

5.8. MSE and PSNR

The Mean Squared Error (MSE) is a popular statistic for assessing the reliability of any encryption scheme by comparing the error between original and encrypted images [4]. For a 3D volume, it is given by:

$$\text{MSE} = \frac{1}{M \times N \times P} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \sum_{k=0}^{P-1} (X(i, j, k) - Y(i, j, k))^2 \quad (7)$$

Given the MSE, we can compute the peak signal-to-noise ratio (PSNR), which is a ratio of the image's highest pixel value over the MSE, can be used to assess the cryptosystem's quality. For 16-bit images, where the maximum pixel value is:

$$I_{\max} = 2^{16} - 1 = 65535 \quad (8)$$

it is given by:

$$\text{PSNR} = 10 \log_{10} \left(\frac{65535^2}{\text{MSE}} \right) \quad (9)$$

A higher MSE between the original and encrypted images indicates greater visual dissimilarity, which is desirable in image encryption. This suggests better obfuscation, which is an indicator of the cryptosystem's quality. MSE and PSNR are inversely correlated. A minimal PSNR value is preferable [4].

Table 4 presents the MSE and PSNR results for our test images.

Table 4. MSE and PSNR results for the three test images.

Image	MSE	PSNR
Test image 1	972166690.80	6.46 dB
Test image 2	424839310.33	10.05 dB
Test image 3	845014081.06	7.06 dB

The MSE values seem too large compared to the bibliography concerning 8-bit images, due to the difference in the dynamic range between 8-bit (0–255) and 16-bit (0–65,535) images. The equivalent MSE values in 8-bit would be 14,718.87 and 6432.18 for test images 1 and 2, respectively. These are extremely high values on the 8-bit scale, indicating large differences between the original and encrypted images.

5.9. Resistance to Chosen-Plaintext Attacks

In a chosen-plaintext attack, the adversary has temporary access to the encryption oracle and can feed it with carefully chosen plaintext to reveal some information about the encryption key [6]. A common chosen-plaintext attack uses an all-white or an all-black image and attempt to detect any non-random patterns in the cipher image or any non-uniformity in its histogram.

As shown in Figures 44 and 45, the resulting cipher images have no visible patterns and their histograms are uniform.

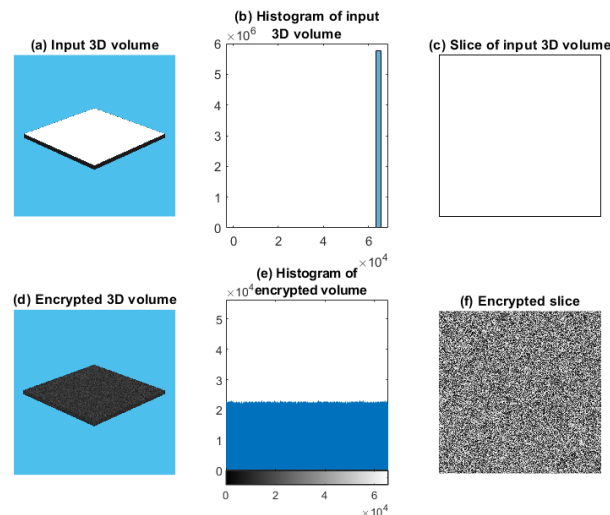


Figure 44. All-white chosen-plain volume attack. (a) original all-white 3D volume and (b) its histogram; (c) a slice of the original; (d) encrypted 3D volume and (e) its histogram; (B) a slice of the encrypted volume.

Moreover, there is no visible difference between the two encrypted 3D volumes.

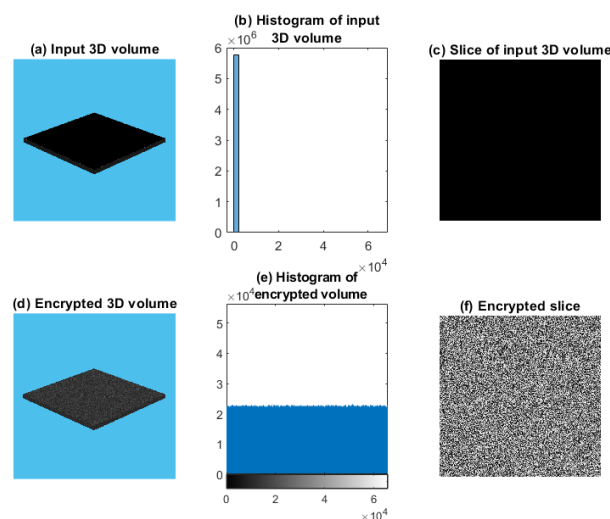


Figure 45. All-black chosen-plain volume attack. (a) Original all-black 3D volume and (b) its histogram; (c) a slice of the original; (d) encrypted 3D volume and (e) its histogram; (f) a slice of the encrypted volume.

6. Discussion

The cryptosystem proposed in this study is designed to operate on 3D medical volumes in modern formats such as NIfTI and DICOM, supporting unsigned 8-bit and 16-bit integer data, as well as floating-point (single) representations.

During the preprocessing stage, input images in uint8 or single format are first converted to a 16-bit representation to ensure uniformity. Additionally, if necessary, the spatial resolution of the input images can be enhanced through interpolation. Experimental results indicate that increasing the resolution of originally low-resolution images leads to improved autocorrelation values, which contributes positively to the overall performance of the system.

6.1. Performance Comparison

To demonstrate the competitiveness of the proposed cryptosystem, we compare it against state-of-the-art medical image encryption schemes. In Table 5, the proposed framework is evaluated against related approaches using average performance metrics, enabling a quantitative and fair comparison.

Table 5. Comparison of the proposed cryptosystem with related schemes.

Metrics	Proposed	Ref. [29]	Ref. [1]	Ref. [36]	Ref. [15]
Dimension	3D	2D	3D	3D	2D
Image Formats	uint8, uint16, NIFTI, DICOM	uint8, uint16, DICOM	uint16	- ¹	-
Entropy (norm.)	0.9978	0.99997	0.9988	0.99948	0.99986
NPCR avg (%)	99.9988	99.9994	99.9978	99.73	99.655
UACI avg (%)	33.38	33.69	33.34	33.64	33.53
Corr. X avg	0.0005	0.001132	0.00036	-	0.0151
Corr. Y avg	0.0004	−0.001974	0.00049	-	0.0267
Corr. Z avg ²	0.0011	0.001252	0.00044	-	0.0278
PSNR avg	7.857	26.6476	-	6.8125	9.7262
Key length (bits)	672	233	1344	-	359

¹ Dash (-) denotes “Not specified”. ² In the proposed cryptosystem, this number concerns the third dimension; in the others, the diagonal correlation.

The comparison presented in Table 6 is based on several simplifying assumptions, as each cryptosystem has unique characteristics. First, the test images used across different schemes vary in size; to enable fair comparison, we have normalized the results based on the average image size. Second, the proposed cryptosystem operates on 16-bit images, whereas the others use 8-bit images. To account for this difference, the entropy values have been normalized accordingly. Third, time performance was omitted here because each scheme uses a different platform (hardware and software).

6.2. Encryption Rate

The proposed cryptosystem was simulated on an HP 250 G7 notebook running Windows 10 and MATLAB 2019b, equipped with a 2.6 GHz Intel Core i5 1035G1 CPU @ 1 GHz with 8 GB RAM DDR4-3200. The average speed of the encryption process (including preprocessing) was 6.8255 MB/s.

Results from simulations on an ACER notebook equipped with AMD Ryzen 9 @ 4 GHz with 16 GB RAM DDR5-4800 running Windows 11 and MATLAB 2021a, gave an average speed of 12.0441 MB/s.

These results are consistent with findings in the state-of-the-art literature, demonstrating excellent performance in terms of speed. Table 6 compares the encryption speed of the proposed framework with related image encryption schemes.

Table 6. Time performance comparison.

Scheme	Platform (Software & Hardware)	Throughput (MB/s)
[4]	Mathematica / 2.6 GHz Intel Core i7, 16 GB RAM	0.083
[35]	MATLAB R2016b / Intel Core2Duo @ 3.00 GHz, 4GB RAM	7.5429
[5]	Not specified / Not specified	0.4369
[48]	MATLAB 2014b / Intel Core2Duo @ 2.26 GHz, 4GB RAM	3.8245
Proposed	MATLAB R2019b/ Core i5 @ 1 GHz, 8 GB RAM	6.8255
Proposed	MATLAB R2021a/ AMD Ryzen 9 @ 4 GHz, 16 GB RAM	12.0441

6.3. Computational Time Performance vs. Security

The Lorenz system was selected as a balanced compromise between security and computational time efficiency, both of which are critical for lightweight cryptosystems used in securing private medical images.

Many image-encryption schemes, in an effort to minimize the computational cost and improve computational time performance, simplify the design or use insecure modules. In such an implementation, the encryption scheme utilizes one-round encryption and a traditional permutation–diffusion structure. Through cryptanalysis, Fan et al. have shown that the correlation between the key and the plain image is not strong, a fact which leads to the collapse of the cryptosystem. They showed that for the encryption scheme of the permutation–diffusion structure, using the all-zero image can break the diffusion transform to obtain the equivalent diffusion matrix. Based on this, they have proposed two methods of chosen-plaintext attacks [40]. As recommended by Álvarez and Li [8],

Without loss of security, the cryptosystem should be easy to implement with acceptable cost and speed.

6.4. Contributions and Innovations

The contributions and innovations of the proposed cryptosystem are:

1. It uses genetic algorithms and random walk to find optimal initial parameters for the Lorenz chaotic generator, which produce a high Lyapunov exponent.
2. It uses voxel shuffling and bit rotation transformations to increase diffusion.
3. It uses CBC encryption mode in addition to ECB encryption, for increased security.
4. It uses a 672-bit key.
5. It demonstrates superior performance in terms of entropy, key sensitivity, and resistance to various attacks.
6. It operates at high speed.

Given its robustness and encryption rate, the proposed cryptosystem is well-suited for real-time 3D medical image encryption and decryption in healthcare systems, as well as cloud-based healthcare administration.

7. Conclusions

Encrypting medical images for cloud storage, network transmission, or email attachments is critically important for patient privacy, legal compliance, data integrity, and security. In this research, a secure cryptosystem for 3D medical image encryption was presented. The system consists of several stages performing double encryption, voxel shuffling, and bit rotation that increase confusion and diffusion, hence system security.

The proposed cryptosystem supports 3D medical images in both DICOM and NIfTI formats, as well as any 3D medical image stored in uint8, uint16 or single data types. Internally, the system operates on 16-bit voxel precision. A preprocessing stage converts input images to this format when necessary and performs several auxiliary operations to enhance image quality and consistency, including resolution doubling via interpolation for low-resolution images, intensity rescaling and histogram expansion to improve visual clarity and diagnostic accuracy, and removal of null (empty) slices to reduce data redundancy and improve efficiency.

Three representative test images were selected to demonstrate the system's capabilities, each requiring distinct preprocessing steps to highlight the system-based adaptability and robustness across diverse modalities.

The key length is 672 bits, long enough to fulfill the minimum key space requirements.

Simulation results demonstrate strong resistance to statistical, differential, and occlusion attacks, with excellent performance on metrics like entropy (15.98 for 16-bit resolution), NPCR (99.999%), UACI (33.33–33.47%), and PSNR (6.46–10.05 dB). The cryptosystem also operates at high speed.

These characteristics make the proposed cryptosystem suitable for real-time 3D medical image encryption in healthcare systems.

Future research directions include investigating alternative chaotic generators to enhance dynamical complexity and improve cryptographic security such as hyperchaotic systems [19,31], as well as additional innovative transformations (beyond voxel shuffling and bit rotation).

An intriguing research direction involves investigating chaotic systems that simultaneously achieve fast performance and robust security.

This work underscores the potential of 3D volume cryptosystems to provide robust protection against modern cryptanalytic attacks, marking an important step toward more resilient healthcare infrastructures. By extending encryption into three dimensions and various medical image modalities, this study highlights a promising direction for developing real-time cryptosystems that balance complexity, security, and processing speed in securing sensitive medical information.

Author Contributions: Conceptualization, A.S.A. and A.P.L.; methodology, A.S.A.; software, A.S.A.; validation, A.S.A. and A.P.L.; resources, A.S.A.; data curation, A.S.A.; writing—original draft preparation, A.S.A.; writing—review and editing, A.S.A. and A.P.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Test images 1 and 2 used in this study are openly available: (1) Test image 1 is MATLAB's MRI test image 'D', built into MATLAB. (2) Test image 2: <https://neurovault.org/images/807799> (accessed on 18 August 2025). Test image 3 is a private 3D CT scan.

Acknowledgments: The authors wish to thank C. Douligeris for providing useful literature and G. Kladis for reading an early draft of this manuscript and making valuable comments. During the preparation of this study, the authors leveraged ChatGPT and Qwen to assist with code suggestions and debugging, textual refinement, and LaTeX formatting. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CT	Computed Tomography
MRI	Magnetic Resonance Imaging
PET	Positron Emission Tomography
DICOM	Multidisciplinary Digital Publishing Institute
NIfTI	Neuroimaging Informatics Technology Initiative
HIPAA	Health Insurance Portability and Accountability Act
GDPR	General Data Protection Regulation
PRNG	Pseudo-Random Number Generator
NPCR	Number of Pixel Change Rate
UACI	Unified Average Change Intensity
ECB	Electronic CodeBook (mode)

CBC Cipher Block Chaining (mode)
IV Initialization Vector

References

1. Anandik, N.; Muni, S.S.; Kaushik, A. DICOM Compatible, 3D Multimodality Image Encryption using Hyperchaotic Signal. *arXiv* **2025**, arXiv:2504.20689v1. <https://doi.org/10.48550/arXiv.2504.20689>.
2. DICOM Standard. Available online: <https://www.dicomstandard.org> (accessed on 1 August 2025).
3. Stallings, W. *Cryptography and Network Security: Principles and Practice*, 8th ed.; Pearson Education Limited: London, UK, 2023.
4. Elnomy, O. Image Encryption Using the Tent Chaotic Map. Bachelor's Thesis, The German University in Cairo, Cairo, Egypt, 1 June 2023.
5. Jain, K.; Aji, A.; Krishnan, P. Medical Image Encryption Scheme Using Multiple Chaotic Maps. *Pattern Recognit. Lett.* **2021**, *152*, 356–364. <https://doi.org/10.1016/j.patrec.2021.10.033>.
6. Ibrahim, S.; Alhumyani, H.; Masud, M.; Alshamrani, S.S.; Cheikhrouhou, O.; Muhammad, G.; Hossain, M.S.; Abbas, A.M. Framework for Efficient Medical Image Encryption Using Dynamic S-Boxes and Chaotic Maps. *IEEE Access* **2020**, *8*, 160433–160449. <https://doi.org/10.1109/ACCESS.2020.3020746>.
7. Zhang, B.; Liu, L. Chaos-Based Image Encryption: Review, Application, and Challenges. *Mathematics* **2023**, *11*, 2585. <https://doi.org/10.3390/math11112585>.
8. Álvarez, G.; Li, S. Some Basic Cryptographic Requirements for Chaos-Based Cryptosystems. *Int. J. Bifurc. Chaos* **2006**, *16*, 2129–2151.
9. Shannon, C.E. Communication Theory of Secrecy Systems. *Bell Syst. Tech. J.* **1949**, *28*, 656–715. <https://doi.org/10.1002/j.1538-7305.1949.tb00928.x>.
10. Alghamdi, Y.; Munir, A. Image Encryption Algorithms: A Survey of Design and Evaluation Metrics. *J. Cybersecur. Priv.* **2024**, *4*, 126–152. <https://doi.org/10.3390/jcp4010007>.
11. Priyanka; Singh, A.K. A Survey of Image Encryption for Healthcare Applications. *Evol. Intell.* **2023**, *16*, 801–818. <https://doi.org/10.1007/s12065-021-00683-x>.
12. Wu, Y.; Noonan, J.P.; Agaian, S. NPCR and UACI Randomness Tests for Image Encryption. *Cyber J. Multidiscip. J. Sci. Technol. J. Sel. Areas Telecommun.* **2011**, *1*, 31–38.
13. Li, S.; Li, M.; Li, Q.; Lv, Y. Hyperchaotic Image Encryption System Based on Deep Learning LSTM. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* **2023**, *14*, 11.
14. Zhang, X.; Liu, G.; Zou, C. An image encryption method based on improved Lorenz chaotic system and Galois field. *Appl. Math. Model.* **2024**, *131*, 535–558. <https://doi.org/10.1016/j.apm.2024.04.023>.
15. Lyle, M.; Sarosh, P.; Parah, S.A. Selective Medical Image Encryption Based on 3D Lorenz and Logistic System. *Multimed. Tools Appl.* **2024**, *83*, 45553–45574. <https://doi.org/10.1007/s11042-023-16996-1>.
16. Soleymani, A.; Nordin, M.J.; Sundararajan, E. A Chaotic Cryptosystem for Images Based on Henon and Arnold Cat Map. *Sci. World J.* **2014**, *2014*, 536930. <https://doi.org/10.1155/2014/536930>. PMID:25258724; PMCID:PMC4166429.
17. Fu, J.; Guo, P. Image Encryption Algorithm Combining Semi-Tensor Product Compressed Sensing and Double Random-Phase Encoding. *Phys. Scr.* **2025**, *100*, 025118.
18. Shahna, K.U. *Novel Chaos Based Cryptosystem Using Four-Dimensional Hyper Chaotic Map with Efficient Permutation and Substitution Techniques*; Chaos Solitons Fractals: Amsterdam, The Netherlands, 2023.
19. Lan, Y.-C.; Wang, C.-M. A Novel Multi-Image Encryption Scheme Using Generalized Rectangular Transform and Advanced 5-D Hyperchaotic Map. *IEEE Access* **2025**, *13*, 43316–43337. <https://doi.org/10.1109/ACCESS.2025.3547891>.
20. Ye, G.; Liu, S.; Xiao, X.; Hunag, X. Image hiding algorithm based on local binary pattern and compressive sensing. *Math. Comput. Simul.* **2025**, *237*, 316–334. <https://doi.org/10.1016/j.matcom.2025.05.005>.
21. Ahmed, S.T.; Hammood, D.A.; Chisab, R.F.; Al-Naji, A.; Chahl, J. Medical Image Encryption: A Comprehensive Review. *Computers* **2023**, *12*, 160. <https://doi.org/10.3390/computers12080160>.
22. Wu, J.; Zhang, J.; Liu, D.; Wang, X. A Multiple-Medical-Image Encryption Method Based on SHA-256 and DNA Encoding. *Entropy* **2023**, *25*, 898. <https://doi.org/10.3390/e25060898>.
23. Yousif, S.F.; Abboud, A.J.; Radhi, H.Y. Robust Image Encryption with Scanning Technology, the El-Gamal Algorithm and Chaos Theory. *IEEE Access* **2020**, *8*, 155184–155209. <https://doi.org/10.1109/ACCESS.2020.3019216>.
24. Muthu, J.S.; Murali, P. A new chaotic map with large chaotic band for a secured image cryptosystem. *Optik* **2021**, *242*, 167300. <https://doi.org/10.1016/j.ijleo.2021.167300>.
25. Panwar, A.; Biban, G.; Chugh, R.; Tassaddiq, A.; Alharbi, R. An efficient image encryption model based on 6D hyperchaotic system and symmetric matrix for color and gray images. *Heliyon* **2024**, *10*, e31618. <https://doi.org/10.1016/j.heliyon.2024.e31618>.
26. Al Nassan, W.; Bonny, T.; Al-Shabi, M.A. Chaos-Based Encryption System of DICOM Medical Images. In Proceedings of the Multimodal Image Exploitation and Learning 2024, National Harbor, MD, USA, 21–26 April 2024; SPIE: Bellingham, WA, USA, 2024; Volume 13033, pp. 205–211. <https://doi.org/10.1117/12.3013891>.

27. Suhasini, P.; Kanchana, S. Enhanced Fractional Order Lorenz System for Medical Image Encryption in Cloud-Based Healthcare Administration. *Int. J. Comput. Netw. Appl.* **2022**, *9*, 424–437. <https://doi.org/10.22247/ijcna/2022/214504>.
28. Belazi, A.; Talha, M.; Kharbech, S.; Xiang, W. Novel Medical Image Encryption Scheme Based on Chaos and DNA Encoding. *IEEE Access* **2019**, *7*, 36667–36681. <https://doi.org/10.1109/ACCESS.2019.2906292>.
29. Muthu, J.S.; Murali, P. A novel DICOM image encryption with JSMP map. *Optik* **2022**, *251*, 168416. <https://doi.org/10.1016/j.ijleo.2021.168416>.
30. Mortajez, S.; Tahmasbi, M.; Zarei, J.; Jamshidnezhad, A. A novel chaotic encryption scheme based on efficient secret keys and confusion technique for confidential of DICOM images. *Inform. Med. Unlocked* **2020**, *20*, 100396. <https://doi.org/10.1016/j.imu.2020.100396>.
31. Zhuang, Z.; Zhuang, Z.; Wang, T. Medical image encryption algorithm based on a new five-dimensional multi-band multi-wing chaotic system and QR decomposition. *Sci. Rep.* **2024**, *14*, 402. <https://doi.org/10.1038/s41598-023-50661-9>.
32. Zhang, Z.; Tang, J.; Zhang, F.; Huang, T.; Lu, M. Medical Image Encryption Based on Josephus Scrambling and Dynamic Cross-Diffusion for Patient Privacy Security. *IEEE Trans. Circuits Syst. Video Technol.* **2024**, *34*, 10, 5850–5863.
33. Liu, Y.; Xue, R. 3D Medical Image Encryption Algorithm Using Biometric Key and Cubic S-box. *Phys. Scr.* **2024**, *99*, 055035. <https://doi.org/10.1088/1402-4896/ad3b3d>.
34. Xu, C.; Shang, Y.; Yang, Y.; Zou, C. An Encryption Algorithm for Multiple Medical Images Based on a Novel Chaotic System and an Odd-Even Separation Strategy. *Sci. Rep.* **2025**, *15*, 2863. <https://doi.org/10.1038/s41598-025-86771-9>.
35. Abd-El-Atty, B.; El-Affendi, M.A.; Chelloug, S.A.; Abd El-Latif, A.A. Double Medical Image Cryptosystem Based on Quantum Walk. *IEEE Access* **2023**, *11*, 69164–69176. <https://doi.org/10.1109/ACCESS.2023.3289932>.
36. Lima, V.S.; Madeiro, F.; Lima, J.B. Encryption of 3D Medical Images Based on a Novel Multiparameter Cosine Number Transform. *Comput. Biol. Med.* **2020**, *121*, 103772. <https://doi.org/10.1016/j.compbiomed.2020.103772>.
37. Li, T.; Yan, W.; Chi, Z. A new image encryption algorithm based on optimized Lorenz chaotic system. *Concurrency Comput. Pract. Exp.* **2020**, *34*, e5902. <https://doi.org/10.1002/cpe.5902>.
38. Guo, J.; Lv, Z.; Zhang, L. Breaking a Chaotic Encryption Based on Hénon Map. In Proceedings of the 2010 Third International Symposium on Information Processing (ISIP), Qingdao, China, 15–17 October 2010; pp. 169–171. <https://doi.org/10.1109/ISIP.2010.65>.
39. Álvarez, G.; Montoya, F.; Romera, M.; Pastor, G. Cryptanalyzing a Discrete-Time Chaos Synchronization Secure Communication System. *Chaos, Solitons & Fractals* **2004**, *21*, Issue 3, 689–694. <https://doi.org/10.1016/j.chaos.2003.12.013>.
40. Fan, H.; Zhang, C.; Lu, H.; Li, M.; Liu, Y. Cryptanalysis of a New Chaotic Image Encryption Technique Based on Multiple Discrete Dynamical Maps. *Entropy* **2021**, *23*, 1581. <https://doi.org/10.3390/e23121581>.
41. Andrews, N.K. Numerical Approximation of Lyapunov Exponents and its Applications in Control Systems. Electronic Theses and Dissertations, 2278. Master of Science in Mathematics (M.S.), Georgia Southern University, Statesboro, Georgia, United States, 2021. Available online: <https://digitalcommons.georgiasouthern.edu/etd/2278> (accessed on 16 August 2025).
42. Blesa, A.; Serón, F.J. Advancement of the DRPE Encryption Algorithm for Phase CGHs by Random Pixel Shuffling. *Appl. Sci.* **2025**, *15*, 4120. <https://doi.org/10.3390/app15084120>.
43. What is p-Box in Cryptography? Last Updated : 23 July 2025 <https://www.geeksforgeeks.org/computer-networks/what-is-p-box-in-cryptography/> (accessed on 10 August 2025).
44. Tutorials Point, *Cryptography - ChaCha20 Encryption Algorithm*. Available online: https://www.tutorialspoint.com/cryptography/cryptography_chacha20_encryption_algorithm.htm (accessed on 11 Sept. 2025).
45. Network Encyclopedia, *ChaCha20: The Dance of Cryptography*. Available online: <https://networkencyclopedia.com/chacha20-the-dance-of-cryptography> (accessed on 11 Sept. 2025).
46. Crypto Done Right, *AES CTR Mode*. Available online: https://cryptodoneright.org/articles/symmetric_algorithms/mode_ctr/aes_ctr_dev_quickstart (accessed on 11 Sept. 2025).
47. Buchanan, W., *CTR Mode in AES*, January 11, 2025. Available online: https://asecuritysite.com/blog/2025-01-11_CTR-Mode-in-AES-1e3db1c74716.html (accessed on 11 Sept. 2025).
48. Li, C.; Luo, G.; Li, C. An Image Encryption Scheme Based on the Three-dimensional Chaotic Logistic Map. *Int. J. Netw. Secur.* **2019**, *21*, 22–29. [https://doi.org/10.6633/IJNS.201901_21\(1\).04](https://doi.org/10.6633/IJNS.201901_21(1).04).
49. Geng, S.; Li, J.; Zhang, X.; Wang, Y. An Image Encryption Algorithm Based on Improved Hilbert Curve Scrambling and Dynamic DNA Coding. *Entropy* **2023**, *25*, 8.
50. Xiaopeng, W.; Ling, G.; Qiang, Z.; Jianxin, Z.; Shiguo, L. A novel color image encryption algorithm based on DNA sequence operation and hyper-chaotic system. *J. Syst. Softw.* **2012**, *85*, 290–299.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.