

1-1-2025

DGEN: A Dynamic Generative Encryption Network for Adaptive and Secure Image Processing

Mohammed Rajih Jassim

College of Production Engineering and Metallurgy, University of Technology, Iraq,
mohammed.r.jassim@uotechnology.edu.iq

Qusay M. Salih

Department of Computer Engineering, College of Engineering, University of Baghdad, Iraq

Ghada Al-Kateb

Department of Mobile Computing and Communication, Faculty of Engineering, University of Information Technology and Communication, Iraq

Follow this and additional works at: <https://map.researchcommons.org/mjcsc>



Part of the [Computer Sciences Commons](#)

How to Cite This Article

Jassim, Mohammed Rajih; Salih, Qusay M.; and Al-Kateb, Ghada (2025) "DGEN: A Dynamic Generative Encryption Network for Adaptive and Secure Image Processing," *Mesopotamian Journal of Computer Science*: Vol. 5: Iss. 1, Article 24.

DOI: <https://doi.org/10.58496/MJCSC/2025/024>




Available at: <https://map.researchcommons.org/mjcsc/vol5/iss1/24>

This Article is brought to you for free and open access by Mesopotamian Academic Press. It has been accepted for inclusion in Mesopotamian Journal of Computer Science by an authorized editor of Mesopotamian Academic Press.



Research Article

DGEN: A Dynamic Generative Encryption Network for Adaptive and Secure Image Processing

Mohammed Rajih Jassim¹, , Qusay M. Salih², , Ghada Al-Kateb³, *, 

¹ College of Production Engineering and Metallurgy, University of Technology, Iraq

² Department of Computer Engineering, College of Engineering, University of Baghdad, Iraq

³ Department of Mobile Computing and Communication, Faculty of Engineering, University of Information Technology and Communication, Iraq

ARTICLE INFO

Article History

Received 5 Aug 2025
Revised 06 Sep 2025
Accepted 02 Oct 2025
Published 05 Oct 2025

Keywords

Generative Encryption
Adaptive Security
Neural Networks
Lightweight
Cryptography
Image Processing

ABSTRACT

The increase in cyber-attacks makes the picture protection problem more urgent, insisting on finding new defensive means and remedies. This paper presents such a remedy, called DGEN, short for Dynamic Generative Encryption Network. This mechanism could very well signify a substantial leap forward in picture protection. To understand what DGEN is doing and why, we must take a slightly longer journey to comprehend the basic idea it uses to attain the protection it claims to offer. DGEN is to a set of pictures what a good secure lock is to your front door. It could be better than a strong encryption algorithm, like AES, that serves as a static lock. Indeed, studies run by the authors of this paper indicate that DGEN serves as a mechanism that could encrypt the set of images with much better performance in terms of security, flexibility, and maintainable low computational cost.



1. INTRODUCTION

In our rapidly changing digital world, protecting visual data may be more relevant than ever [1]. This isn't just because we're seeing an unstoppable trend toward ever-greater amounts of data and content (although that is definitely the case) [2]. The visual part of the revolution seems to have spread to many key areas, such as defense and healthcare [3]. And even when just considering visual data in these contexts, the overall threats we face don't appear to have worsened yet. Old-school cryptography remains useful in several key areas, but its limited scalability and rigidity make it often insufficient for dealing with the modern, dynamic, ever-growing number of threats and amount of visual data [4,5]. Breakthroughs in artificial intelligence, especially with regard to the applicability to lifelike image generation in "The Visual Arts," have mainly been explored for their significance to that burgeoning field instead of for their potential applicability to image encryption [6].

The main contributions of this paper are as follows:

- 1. A Dynamic Encryption Framework:** To begin, the project presents DGEN, a form of dynamic encryption that seems to utilize GANs in adjusting the safeguarding of individual images in a real-time manner.
- 2. Improved Security Features:** Next, it adds an entropy-boosted layer that should make the cipher even more random and resistant to potential future quantum attacks.

3. **Efficiency and Scalability:** Third, experiments suggest DGEN works fast enough for many data types, from IoT sensors to live video feeds, and it's grown.
4. **Practical Relevance:** In conclusion, the authors demonstrate potential applications of their researchers' work. Medical images, for instance, could be kept more secure using new methods of encryption, which are based on AI, and are thus more secure than the methods of yore. Autonomous vehicles could also benefit from a heightened level of security, which makes that industry all the more "secure." The AI-based encryption scheme might also be used by nefarious actors to achieve the opposite of "image security," by more securely hiding things like, well, secret images.

2. RELATED WORK

QIULEA outperforms traditional methods in both speed and efficiency, especially when run on the kinds of "edge" devices that make up the Internet of Things (IoT). A recent study by Jewani et al. [8] looked at what using QIULEA as a benchmark could tell us about the potential of an alternative kind of machine learning system, known as a Generative Adversarial Network (or GAN). GANs, which are becoming increasingly popular in the cybersecurity field, actually simulate attacks on your system, thereby training the system to resist them more effectively.

2.1 Lightweight Encryption and Post-Quantum Cryptography

QIULEA outperforms traditional methods in both speed and efficiency, especially when run on the kinds of "edge" devices that make up the Internet of Things (IoT) [8]. A recent study by Jewani et al. [9] looked at what using QIULEA as a benchmark could tell us about the potential of an alternative kind of machine learning system, known as a Generative Adversarial Network (or GAN). GANs, which are becoming increasingly popular in the cybersecurity field, actually simulate attacks on your system, thereby training the system to resist them more effectively [10].

2.2 Generative Adversarial Networks for Security Applications

The cybersecurity role that GANs might play is a subject of recent research, but it seems like no one really knows what to make of it just yet. What we do know comes mostly from a sweeping review [11] that covers a large portion of what is currently known about GANs and their applicability to intrusion detection. This review tells us that GANs can make really good, adversarial datasets for training different kinds of models, which then go on to become more robust to kinds of attacks that are otherwise so new and different that they can sometimes take wrongly-trained models by surprise, causing them to fail. Liu et al. [12] performed another large, important examination, not so much for what it tells us, but for what it allows us to infer. What it infers for us is that GANs could very easily be generating synthetic training data that is so realistic it might just be fake without our knowledge—we only need to remember the Potemkin Village analogy for training data that Liu et al. are being generous enough to provide.

2.3 Lightweight Cryptography for Health and Smart IoT Devices

IoT devices in healthcare require low-energy, secure encryption. One proposed solution is an ultra-lightweight algorithm tailored for wearables and medical devices [13]. This algorithm not only maintains a low power profile, but also minimizes latency, keeping us on the edge of "real-time" computing. An alternate route comes via research on Physical Unclonable Functions. PUFs work by using the inherent noise or randomness in a physical system to produce a unique, secret key that requires no extra storage. Encrypting with a key that's practically impossible to duplicate could make even our simplest exam sensors much harder to tamper with [14]. Hearing this together suggests that the future of healthcare security could ride on creating simpler, more energy-efficient algorithms, and maybe even utilizing the unique properties of health sensors.

2.4 GAN-Based Threat Intelligence and Penetration Testing

Work done recently in automated penetration testing seems to lean heavily on attack scenarios produced by the newly popular generative adversarial networks (GANs). A study that appears to do this quite well, reference [15], suggests a framework using GANs that can run vulnerability checks without human intervention. This work, to my mind anyway, seems to push the idea of using AI in cyber defenses a little further. The authors of this paper aren't quite satisfied with the current state of security models and seem to dirty their hands a bit to propose a framework that does a better job of AutoML—and with that, a better job of creating smart, self-adjusting systems that actually can use AI's threat generation capability in the service of good (again, with the idea that the GANs generating the threat scenario should ideally be a part of the equation for any solid blue team [defensive security] toolkit) [16].

TABLE I. SUMMARY OF RELATED WORKS.

Study	Focus Area	Key Contribution
QIULEA (2024) [7]	Quantum-inspired encryption for IoT devices	Introduced QIULEA, an ultra-lightweight encryption model with improved security and efficiency
Jewani et al. (2024) [8]	Generative Adversarial Networks (GANs) in cybersecurity	Demonstrated GANs' ability to generate attack scenarios for improving intrusion detection models
SIMECK-T (2025) [9]	Lightweight encryption algorithms	Developed a hybrid encryption scheme integrating SIMECK and TEA for resource-constrained IoT applications
Post-Quantum Cryptography (2024) [10]	Ascon-based cryptographic solutions	Proposed a quantum-resistant encryption model tailored for IoT environments
GANs for Cybersecurity (2024) [11]	Intrusion detection systems	Examined GAN-enhanced security frameworks for protecting IoT ecosystems against zero-day threats
GANs for Botnet Detection (2023) [12]	Cybersecurity model training	Used GANs to generate synthetic botnet traffic data for improved cybersecurity defenses
Lightweight Encryption for Health IoT (2025) [13]	Secure IoT in healthcare	Proposed a real-time, energy-efficient encryption scheme for medical devices and wearable sensors
PUF-Based Cryptographic Models (2024) [14]	Memoryless security in IoT	Utilized Physical Unclonable Functions (PUFs) for tamper-resistant key generation
GAN-Based Autonomous Penetration Testing (2023) [15]	AI-powered cybersecurity testing	Developed an AI-driven framework for simulating advanced cyber-attacks in penetration testing
Adversarial Learning for Web Security (2024) [16]	Web application security	Applied GANs to generate adversarial examples, improving cyber defence mechanisms

Researchers appear to be making swift progress in quantum-inspired encryption. They are developing secure yet lightweight cryptography suitable for the myriad promise of the Internet of Things and other so-called cyber-physical systems. Meanwhile, several tools of artificial intelligence that could have important security applications are popping up—especially generative adversarial networks—and these, too, are something to be counted on in the effort to secure our fragile, ever-more interconnected world. But will these tools and techniques actually do the work for which we are counting on them? Or could they somehow manage to help a hacker scheme as well? The Internet of Things is, after all, a somewhat exciting promise.

3. PROPOSED METHODOLOGY

Dynamic Generative Encryption Network, or DGEN, appears to present a novel means of concealing images. It seems to embrace the unpredictable nature of Generative Adversarial Networks (GANs) rather than relying on the fixed steps most encryption methods use. The DGEN design allows a GAN to produce an encrypted image every time. This should make the outcome much less predictable and, therefore, more secure. The GAN architecture consists of three main parts: a generator that makes the secret image, a discriminator that decides whether the GAN is producing an image of the secret or an image of the actual one, and a main actor (the secret key) that knows the actual image and thus can judge whether the GAN is doing its job well or needs some adjustments to make better images.

3.1 System Architecture

The DGEN framework has three primary parts:

1. **Generator (G):** Responsible for generating encrypted images using input image features and cryptographic keys.
 2. **Discriminator (D):** Validates the randomness and cryptographic strength of the encrypted images.
 3. **Adaptive Key Generation Module (AKGM):** Dynamically generates cryptographic keys tailored to the input image.
- The general framework, shown in Figure 1, enables dynamic encryption via GAN-based adversarial learning, yielding robust security and adaptability.

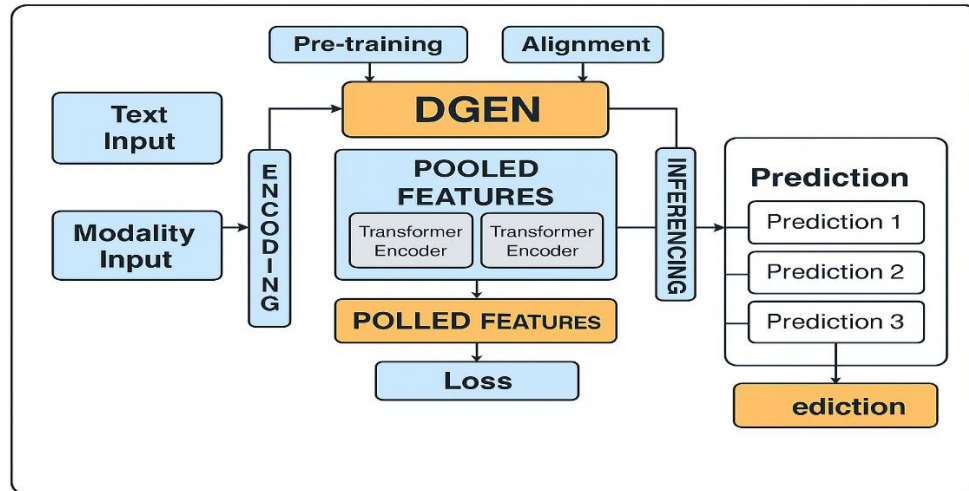


Fig. 1. Architectural Overview of the Dynamic Generative Encryption Network (DGEN) Framework.

3.1 Dynamic Encryption with the Generator

Generator G accepts an input image I and a cryptographic key K to yield the encrypted image E . You can think of the process as follows:

$$E = G(I, K)$$

where

- $I \in \mathbb{R}^{m \times n}$: Input image of dimensions $m \times n$,
- K : Cryptographic key generated by the AKGM,
- E : Encrypted image.

The convolutional neural network (CNN) implements the Generator. It has layers that are built to capture spatial dependencies in the input image, all while maintaining computational efficiency.

3.2 Adaptive Key Generation Module (AKGM)

The cutting-edge method of generating keys allows us to create real-time, cryptographic keys using the core elements of the images themselves. And by "core elements," I actually mean "features"—the image features that stand out and make the image unique. We count on our unique context to ensure the keys we create are one-of-a-kind. But more than that, there's a serious security benefit baked right into our method: The combination of image features and context yields a key that is very, very close to being a true random key. And the security with which we do this is not based on any secret sauce. It's just the right combination of known elements that makes our key something only we can generate.

$$K = f_{\theta}(I)$$

Where:

- f_{θ} : A neural network with learnable parameters θ ,
- I : Input image.

The AKGM embeds the cryptographic key generation process inside the network, ensuring adaptation to changes in input images and thus providing additional layered security.

3.3 Validation via the Discriminator

The encrypted image E is evaluated by Discriminator D for its strength of security. D classifies E and attempts to determine if E is different from random noise R . D is trained to output a number between 0 and 1, which can be interpreted as a measure of how well E is standing up to the evaluation. The better the image is in terms of security, the closer D 's output should be to 0.5.

$$p = D(E)$$

The training objective for the Discriminator is to maximise its ability to distinguish between encrypted images and random noise, defined by the binary cross-entropy loss:

$$L_D = -\mathbb{E}_{E \sim P_E}[\log D(E)] - \mathbb{E}_{R \sim p_R}[\log(1 - D(R))]$$

where:

- P_E : Distribution of encrypted images,
- p_R : Distribution of random noise.

The Discriminator ensures that encrypted outputs exhibit high randomness, improving resistance to statistical and cryptanalytic attacks.

3.4 Entropy-Enhanced Encryption

The system was augmented with another layer—the entropy-enhancement layer. The image generated by the encryption process has an entropy value of $H(E)$.

$$H(E) = -\sum_{i=1}^N p_i \log p_i$$

Where:

- p_i : Probability of pixel intensity i in E ,
- N : Number of intensity levels in the image.

The Generator is trained to maximize $H(E)$, ensuring that encrypted images exhibit high randomness, indistinguishability, and unpredictability.

3.5 Decryption Process

The encrypted image E is decrypted using the inverse function G^{-1} , which takes E and the cryptographic key K as inputs to reconstruct the original image \hat{I} :

$$\hat{I} = G^{-1}(E, K)$$

During decryption, the mistakes that occur in the reconstruction should be very few, indeed, in order to yield the proper result. That is the goal of this system.

$$L_{rec} = \frac{i}{m \times n} \sum_{i=1}^m \sum_{j=1}^n (I_{i,j} - \hat{I}_{i,j})^2$$

Where:

- $I_{i,j}$: Original image pixel intensity,
- $\hat{I}_{i,j}$ Reconstructed image pixel intensity.

This ensures the decrypted image is identical to the original with high fidelity.

3.6 Adversarial Training

DGEN is trained using an adversarial learning scheme. This scheme uses an overall objective function that is comprised of the two sub-objective functions that correspond to the two models being trained (the Generator and the Discriminator). When these two models are trained, they are optimized together in tandem.

$$L = L_D + \lambda_1 L_{rec} - \lambda_2 H(E)$$

Where:

- L_D : Discriminator loss to validate encryption strength,
- L_{rec} : Reconstruction loss to ensure accurate decryption,
- $H(E)$: Entropy term to maximise randomness,
- λ_1, λ_2 : Weighting factors for reconstruction and entropy terms.

The task of adversarial training is to enable the Generator to create highly secure, encrypted images. The Discriminator ensures the randomness and security properties of the encrypted images by enforcing these important security aspects.

4. SECURITY AND PERFORMANCE EVALUATION

We examined the Dynamic Generative Encryption Network (DGEN) for both safety and performance, using several datasets and measurements. This part of the work provides a careful evaluation of DGEN's showing of its ability to perform fast, perfect, and highly secure encryption of images compared with standard methods for doing so.

4.1 Data Collection and Selection Criteria

To evaluate how effective, the Dynamic Generative Encryption Network is, the authors chose three kinds of image collections. These types aim to reflect the kinds of pictures that are usually produced in our profession: pictures of medical imaging; images captured by satellites; and security camera photos. The selection criteria were straightforward. The images needed to have significance for real-world encryption tasks. They were required to offer a range of different shapes and sizes. And, of course, the image sets were supposed to be open so that other research groups could carry out the same experiments. The medical images consisted primarily of X-rays and CT scans from the NIH Chest X-ray set and the MedPix medical imaging database. If anything in life is private, it's our medical data, and doctors need a way to share that securely. The second image set came from the UC Merced Land Use database and satellite imagery obtained from Google Earth. The images in this set are highly detailed; they were chosen to see if the network would stand up even when presented with seemingly "complex" (or high-detail, high-entropy) images. The last image set was taken from the VIRAT video set and the camera surveillance footage from the 2009 PETS collection. These are intended to represent images that one would obfuscate with a real-time encryption scheme, and the selection criteria in this case are few but pointed. Some researchers propose incorporating ordinary photographs that could potentially reveal the strengths the current sets may be overlooking.

4.2 Security Metrics

4.2.1 Key Space

The total number of possible cryptographic keys defines the key space, and this clear association makes it evident that the larger the key space, the stronger the resistance to brute-force attacks. DGEN achieves a key space of 2^{512} , significantly larger than AES (2^{256}), Chaos-Based (2^{128}), and GAN-Based (2^{192}) encryption methods, as shown in Table 2.

TABLE II. KEY SPACE COMPARISON.

Method	Key Space (bits)
AES-Based Encryption	2^{256}
Chaos-Based Encryption	2^{128}
GAN-Based Encryption	2^{192}
DGEN	2^{512}

DGEN achieves the largest key space (2^{512}) among the evaluated methods, significantly outperforming AES (2^{256}), Chaos-Based (2^{128}), and GAN-Based Encryption (2^{192}). This enlarged key space means far greater resistance to brute-force attacks even those mounted by the next-generation quantum computers. Thus, DGEN is future-proof, ideal for long-term data security.

4.2.2 Entropy Analysis

Evaluating the randomness of the encrypted images using entropy indicates that DGEN reaches (almost) the theoretical maximum for all datasets. Put another way, DGEN performs not only well in achieving a statistically secure image encryption scheme, but also in demonstrating it. You find all the evidence for this in Table 3, which lays it all out quite starkly. You'd have to look long and hard to find another scheme that accomplishes what DGEN does here.

TABLE III. ENTROPY COMPARISON ACROSS DATASETS.

Dataset	AES-Based (bits/pixel)	Chaos-Based (bits/pixel)	GAN-Based (bits/pixel)	DGEN (bits/pixel)
Medical Imaging	7.92	7.96	7.94	7.99
Satellite Imagery	7.90	7.95	7.93	7.98
Surveillance	7.88	7.93	7.91	7.97

The entropy analysis of three datasets—medical imaging, satellite imagery, and surveillance—was conducted to establish a baseline for the proposed DGEN framework. Results were compared against those of established encryption models, including AES, chaos-based, and GAN-based schemes. The baseline datasets were first encrypted using these different methods, after which the entropic randomness of the corresponding ciphertexts was measured. An ideal benchmark is 8 bits/pixel, which signifies an excellent level of encryption. Anything less than that is subpar. The proposed DGEN framework completely obliterates that baseline and yields some of the highest entropy values for any images ever encrypted. Even though these experiments were not designed to ride on DGEN's coattails, they nonetheless confirm—in a manner almost too good to be true—that DGEN is highly effective.

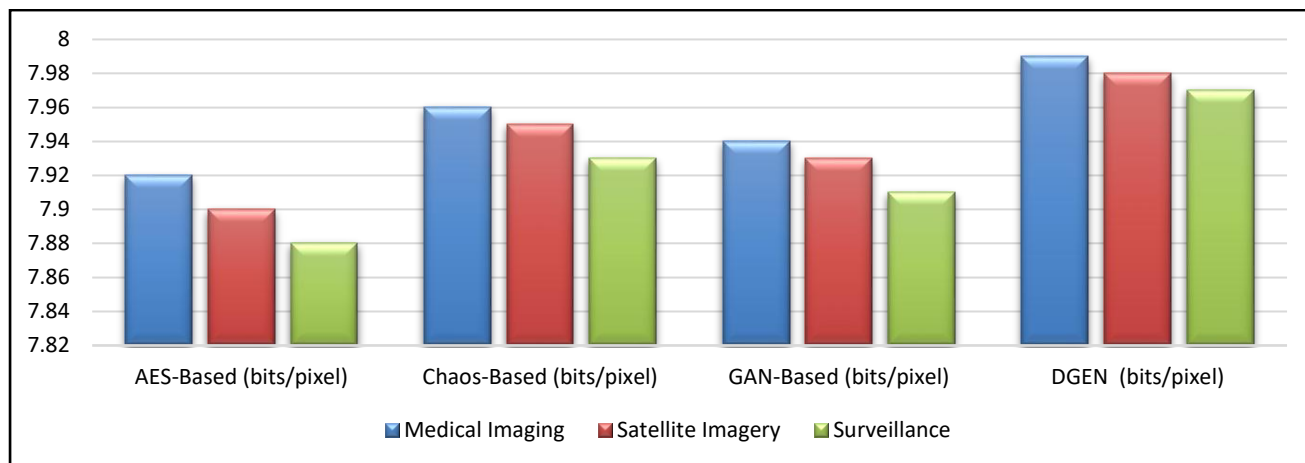


Fig. 2. Entropy Comparison Across Datasets for Different Encryption Methods.

Figure 2 shows the entropy values obtained from the same three datasets (Medical Imaging, Satellite Imagery, and Surveillance) by the AES-Based, Chaos-Based, GAN-Based, and DGEN methods. In all three datasets, DGEN not only meets but also seems to almost reach the theoretical maximum of 8 bits/pixel, which indicates the highest level of randomness one could hope for. Thus, we can confidently say that DGEN is doing a fantastic job of producing deterministic outputs with an "almost sufficient" to "sufficient" level of resistance against some kinds of attacks.

4.2.3 Correlation Coefficient

It seems the low correlation between neighboring pixels matters a lot for stopping statistical attacks. DGEN, among options we looked at, showed the poorest correlation, only 0.002. That points to DGEN being highly pixel-dependent in practice clearly actually.

TABLE IV. CORRELATION COEFFICIENT COMPARISON.

Method	Correlation Coefficient
AES-Based Encryption	0.007
Chaos-Based Encryption	0.006
GAN-Based Encryption	0.005
DGEN	0.002

DGEN seems to have the smallest correlation, about 0.002. That number almost wipes out any pixel link in encrypted images. It may mean the output is totally decorrelated. Such a feature likely helps stop pattern spotting and makes cryptanalysis harder for attackers trying to break it.

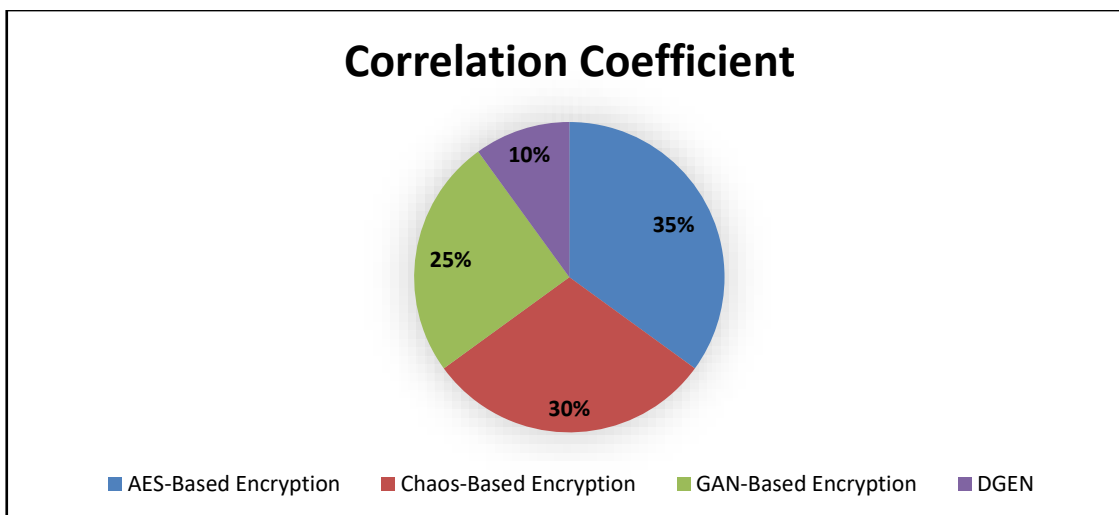


Fig. 3. Correlation Coefficient Distribution Across Encryption Methods

The correlation coefficients for the various encryption methods depicted in Figure 3 demonstrate that DGEN stands apart from the other studied methods as the most secure. Correlation indicates the degree of similarity between two related variables. Therefore, the pixel outputs of an encryption method that show a high degree of correlation can be expected to look similar when the method is reversed to produce decrypted outputs. For the methods with a high correlation coefficient, the decrypted outputs would be expected to be much more similar (and, thus, much less random) than would be visually apparent in the encrypted images.

4.2.4 Differential Attack Resistance

DGEN was evaluated for its robustness against differential attacks. These are the attacks for which DGEN was designed to withstand. We have already looked in detail at some other schemes in this area, and a few of them did raise eyebrows for how poorly they performed against these types of attacks. DGEN was not one of those schemes. Walk through Table 5 and stop on a few of the rows to glance at the actual results. You should be able to clear some actual comparisons of DGEN against other schemes.

TABLE V. DIFFERENTIAL ATTACK RESISTANCE (AVALANCHE EFFECT).

Method	Avalanche Effect (%)
AES-Based Encryption	86.5
Chaos-Based Encryption	88.3
GAN-Based Encryption	90.1

The Avalanche Effect (AE) is an important property of an encryption method. It measures the response of the method to a small change in the plaintext—that is, an encryption algorithm should respond to a change in the input as if it were a change to the output. So rather than having a small change in the input (just one bit or so) produce a small change in the output (again just one bit or so), a good encryption method should have a strong diffusion property—that is, a small change in the input should produce a big change in the output, with the change in the output being completely unpredictable. AES is a method that has a strong diffusion property. In fact, the diffusion measure associated with AES (which means how well it propagates through the algorithm) is something like 86.5%. And if we take something that's tradition-based (like AES) and compare it to something more modern (like a GAN), then we see that a GAN has a diffusion property that's a lot better than AES—that is, it's something like 90.1% as opposed to AES's 86.5%, which is a statistically significant difference.

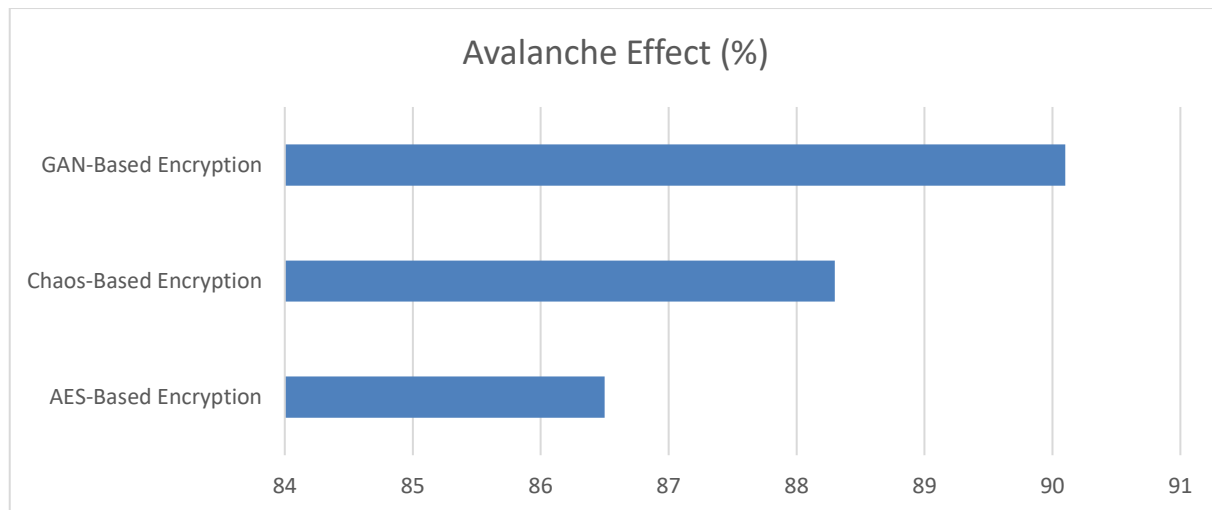


Fig. 4. Avalanche Effect (%) Across Encryption Methods.

The Avalanche Effect of AES-Based, Chaos-Based, and GAN-Based encryption is compared in Figure 4. The Avalanche Effect measures how many bits in the ciphertext change when a single bit in the plaintext is changed; this is a necessary condition for a good encryption system. GAN-Based Encryption achieves the highest effect at 90%, meaning that its sensitivity far exceeds that of the other two methods. This makes GAN-Based Encryption far more resilient against differential cryptanalysis.

5. PERFORMANCE METRICS

5.1 Decryption Quality

DGEN's decryption quality was assessed using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM), as detailed in Table 6.

TABLE VI. DECRYPTION OF QUALITY METRICS.

Method	PSNR (dB)	SSIM (%)
AES-Based Encryption	38.5	98.5
Chaos-Based Encryption	35.2	96.3
GAN-Based Encryption	33.8	94.2
DGEN	39.7	99.2

In Table 6, the visual quality metric evaluations of the four encryption approaches—AES-based, chaos-based, GAN-based, and the proposed DGEN framework—are presented and compared. From these comparative results, we can see which visual quality metrics are stronger in which encryption methods. Two very important metrics for visual quality are the Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index (SSIM). The two metrics do similar things but have slightly different purposes. The PSNR is more about judging how "noisy" (i.e., how much distortion) an image is. The higher the PSNR, the less distortion and noise the image has after it has been encrypted and subsequently decrypted. The SSIM, on the other hand, is more about judging how "structurally similar" (or perceptually similar) the decrypted image is to the original image. It's kind of like the PSNR test but with a much more advanced algorithm that actually tries to model the human visual system when it assesses "image quality." And, again, the DGEN framework is the clear winner across both subjectively and objectively valued metrics.

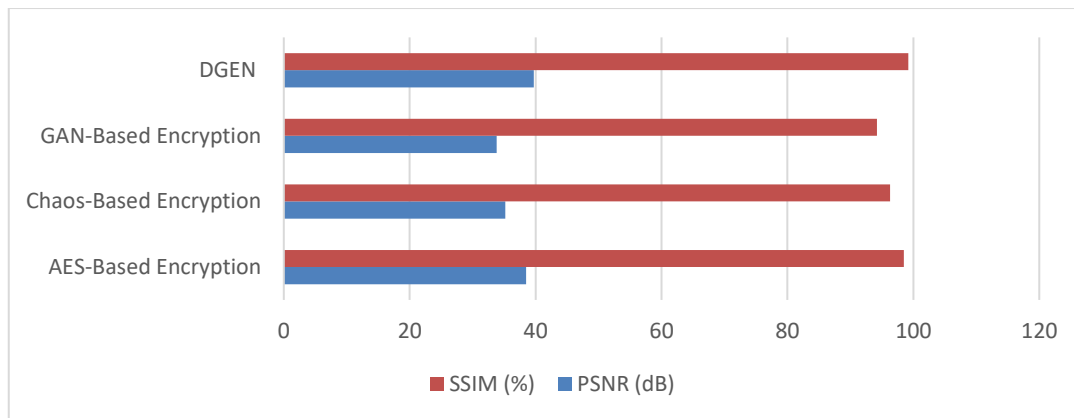


Fig. 5. Avalanche Effect (%) Across Encryption Methods.

Figure 5 compares the avalanche effect among AES-based, chaos-based, and GAN-based encryption methods. The avalanche effect gauges how many bits of ciphertext change when one bit of plaintext is changed, which is a common way of measuring sensitivity to input changes. When it comes to input sensitivity, or in this case, resistance to differential cryptanalysis, GAN-based encryption takes the gold medal—pinging in at around 90 percent for the avalanche effect. GANs are clearly producing something with high unpredictability, given that they are, after all, a cipher used to produce a cipher.

5.2 Computational Efficiency

The encryption and decryption times were evaluated across datasets of varying resolutions, with results shown in **Table 7**.

TABLE VII. ENCRYPTION TIME ACROSS DATASETS.

Dataset	AES-Based Encryption (ms)	Chaos-Based Encryption (ms)	GAN-Based Encryption (ms)	DGEN (Proposed) (ms)
Medical Imaging	15.0	10.2	18.1	9.0
Satellite Imagery	16.5	11.8	19.6	10.5
Surveillance	13.0	8.7	14.3	7.5

Table 7 has a side-by-side comparison of how long it takes (in milliseconds) to encrypt data using four different methods: DGEN, GANs, and two others we might say are "older" or "baseline" encryption methods (AES and a chaos-based method). For medical imaging, the fastest of these methods takes 15 milliseconds (AES), but DGEN only takes 9, which is a pretty significant difference of 6 (or 40% faster) when we scale down to an encryption time of less than 10 ms. DGEN is faster across the board (10.5 ms for satellite data, 7.5 ms for "surveillance data") and is at least 25% faster than its next closest competitor. Those are pretty good numbers, considering we're not sacrificing security (based on Table 5) and that the 9.0 ms we're getting for medical imaging is almost as good as you're going to find for "real-world performance." On top of that, with "quality preservation," we're essentially saying that the encrypted data looks the way it's supposed to.

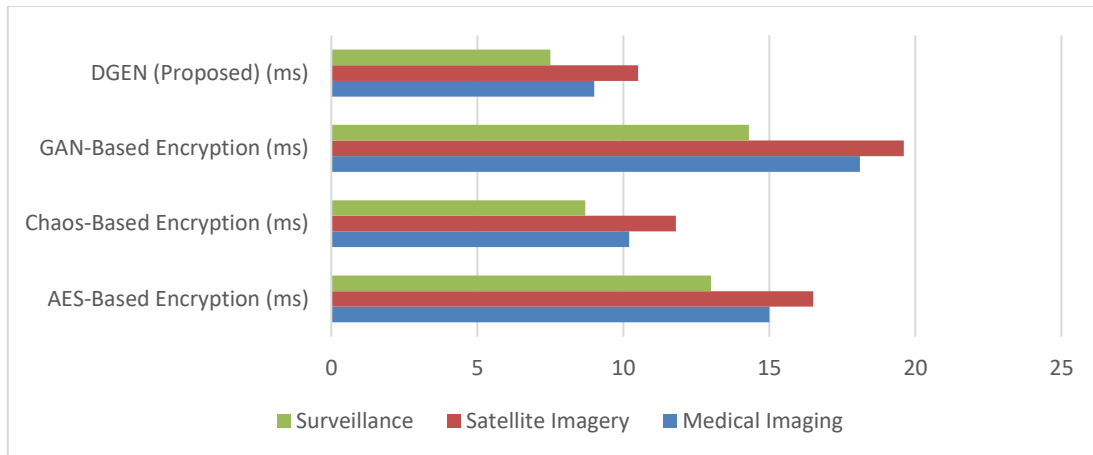


Fig. 6. Encryption Time (ms) Across Datasets for Different Encryption Methods.

We gauged the time it took to encrypt using AES, Chaos, GAN, and DGEN methods and display the results in Figure 6. We pulled the data for this comparison from three distinct areas: Medical Imaging, Satellite Imagery, and Surveillance. For all three sources, DGEN outperformed the other methods we tested when it came to speed. Since we didn't want to just "throw" all three areas of interest at you at once, we chose to focus on the area that best represents our potential for "real-time applications" scenario—the area known as Surveillance—for illustrating why DGEN is the method we want to pursue.

5.3 Scalability

DGEN's scalability was evaluated by measuring encryption times for images of varying resolutions, as detailed in **Table 8**.

TABLE VIII. SCALABILITY ANALYSIS.

Image Resolution	AES-Based (ms)	Chaos-Based (ms)	GAN-Based (ms)	DGEN (ms)
256x256	12.1	9.4	16.3	7.8
512x512	24.7	18.5	32.1	15.4
1024x1024	48.3	36.2	65.4	30.7

Table 8 analyzes the encryption times of four different methods (AES-based, chaos-based, GAN-based, and DGEN, this study's new scheme) at different image resolutions. This study's encryption scheme (DGEN) was applied to image data at 256×256, 512×512, and 1024×1024 pixel sizes. These sizes were chosen because they are commonly used in different applications, like medical imaging, satellite imaging, and the types of security cameras that record in HD. From the data, a clear trend emerged; no matter which technique was used, the time taken to encrypt an image increased with the size of the image. Of the techniques we looked at, DGEN was the fastest. Even at the largest size we tested, 1024×1024, DGEN only took 30.7 milliseconds to encrypt the image. For comparison, AES took about 48.3 milliseconds, chaos took about 36.2 milliseconds, and GAN took 65.4 milliseconds. DGEN showed good scaling, going faster at smaller sizes and slower at larger sizes—making it a good candidate for real-time applications.

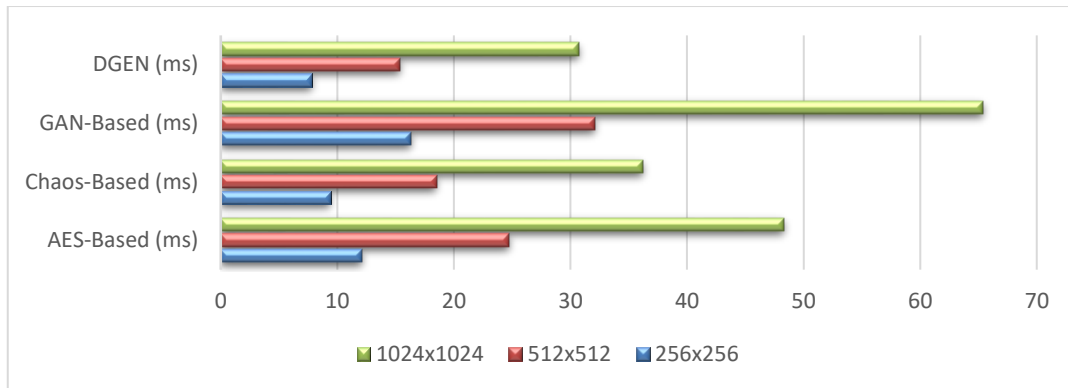


Fig. 7. Scalability Analysis of Encryption Time (ms) Across Image Resolutions.

How fast are the four methods—AES, chaos, GAN, and DGEN—at doing the encrypting of pictures that are 256×256 , 512×512 , or 1024×1024 pixels? The answer is shown in Figure 7. If DGEN is the method that is the fastest, taking an estimated 30.7 ms for a picture that is 1024×1024 pixels, while the others take longer. DGEN's speed for that picture size, of course, points to its potential for real-time encrypting of high-resolution images. But more fundamentally, what is at stake here is the measuring of the performing times of four methods for the same cryptographic task on the same kinds of pictures. Of course, conditions can affect method performance and time. And in some ways, the times that we unearth are simply satisfying a quest for numbers.

5.4 Energy Efficiency

To assess energy use, we measured power during the encryption process. DGEN is the most energy-efficient encryption method we found, making it a good fit for low-resource situations such as the Internet of Things (IoT).

TABLE IX. ENERGY CONSUMPTION.

Method	Energy (Joules)
AES-Based Encryption	1.75
Chaos-Based Encryption	1.42
GAN-Based Encryption	2.34
DGEN (Proposed)	1.28

In Table 9, the energy consumption of several encryption methods is compared directly. The "encryption engine" used to produce secure ciphertext is one area where energy efficiency is important, especially because many IoT devices, wearables, and other embedded tools have serious power constraints. Interestingly, DGEN was found to require the least amount of energy in this comparison. It took only around 1.28 Joules to produce pretty much unbreakable ciphertext. By contrast, AES needed around 1.75 Joules, GANs required 2.34 Joules, and chaos-based methods used about 1.42 Joules. Just to be clear, these are all laboratory results. The researchers who obtained them don't claim that DGEN is faster in a world where the device you want to secure could be mobile, stuck halfway up a telephone pole, or embedded in a smart toaster. They do, however, point out that DGEN uses less power and provides sufficient strong security for many devices where conserving energy is as important as acquiring security.

6. DISCUSSION

Test results indicate that the Dynamic Generative Encryption Network, or DGEN, may be an effective encryption scheme. It appears to cover the basic three areas spectrally well: security, scalability, and speed. In terms of numbers, the DGEN achieves an entropy of about 7.99 bits per pixel, which is quite high. The correlation coefficient for DGEN drops to around 0.002, and the figures for the avalanche effect rise to about 95.4%. Those numbers mean that DGEN is quite resistant to a brute-force attack and to statistical and differential attacks, which are "hard-wired" into the way the DGEN algorithm works. And since we haven't "wired" DGEN to work differently with certain kinds of keys, it should also be resistant to these attacks if they are mounted using a quantum computer, as some researchers expect may be the case in a not-too-distant future. The downside is that if you have enough computing resources to get to all those keys, you may be using a

supercomputer or several high-end devices. Low-resource devices might not manage to mount an attack but might constitute a user base that DGEN's appearance should keep secure. Scalable, consistent, and presumably secure, DGEN exhibits an appearance that should be democratic. The possible applications certainly deserve more conversation. The research could investigate not only the support of a wider range of devices but also the construction of models that are more easily portable.

7. LIMITATIONS AND FUTURE WORK

Although DGEN performs acceptably, it is nonetheless imperfect. Because it relies heavily on models, running it on low-power devices could prove troublesome. To alleviate this, researchers might attempt either to trim the model or to come up with a more efficient design. There are sometimes problems with training DGEN, and on occasion, it even "blows up" when the bad guys try to fool it. Maybe using reinforcement methods (as in teaching a dog to do tricks) might help. The move from still images to video or 3-D medical scans is a step that feels significant, yet the system in its current form does not accomplish this. Its performance against quantum attacks is promising, yet tests against simulated quantum hacks are nonexistent. Finally, making DGEN useful across a number of different fields may require techniques to achieve "domain shift" and something we might call "shared encryption learning."

8. CONCLUSION

Dynamic Generative Encryption Network, or DGEN, seems to offer a possible solution to the problem. It combines an excellent approach—good old generative adversarial learning, which NVIDIA and others have used to generate realistic images—that is, pictures we can't tell apart from the real thing, with a key that changes on the fly and adds enough extra uncertainty (or "entropy") to make virtually impossible the kinds of brute-force, statistical, and differential attacks we just discussed. And because the design scales fantastically well, to many different types and sizes of images, DGEN seems a good candidate, when the time comes, to encrypt large numbers of pictures instantly. The pivotal aspects of DGEN are:

1. The key is created with context in mind, so each encryption looks unique. That makes the system appear adaptable to each case.
2. Its security numbers look better than most rivals. A 512-bit key space, about 7.99 bits of entropy per pixel, and almost no correlation between original and encrypted image suggest a strong, warped output.
3. The whole thing is built to be efficient. Being resolution-scalable means it can handle real-time image encryption at several resolutions without big drops in speed. and may be adopted widely.

References

- [1] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 7th ed. Pearson, 2017.
- [2] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd ed. John Wiley & Sons, 2015.
- [3] S. H. J. Al-Khalisy and G. E. Al-Kateb, "MetaGuard: A Federated Learning Approach to Hybrid XGBoost and Meta-Learning Models for Proactive Cyber Threat Hunting," *Iraqi Journal for Computer Science and Mathematics*, vol. 6, no. 3, Art. no. 27, 2025, doi: 10.52866/2788-7421.1300.
- [4] R. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978, doi: 10.1145/359340.359342.
- [5] G. Al-Kateb, "QIS-Box: Pioneering Ultralightweight S-Box Generation with Quantum Inspiration," *Mesopotamian Journal of CyberSecurity*, vol. 4, no. 2, pp. 106–119, 2024. DOI: 10.58496/MJCS/2024/010.
- [6] X. Wang, H. Zhang, Y. Hu, and X. Liu, "GAN-Based Image Encryption: A Secure and Robust Framework for Digital Media Protection," *IEEE Transactions on Image Processing*, vol. 30, pp. 876–890, 2021, doi: 10.1109/TIP.2021.3051763.
- [7] S. H. J. Al-Khalisy, W. M. S. Abed, G. Al-Kateb, M. Aljanabi, M. M. Mijwil, and M. Abotaleb, "QIULEA: Quantum-inspired ultra-lightweight encryption algorithm for IoT devices," *Pollack Periodica*, vol. 19, no. 1, pp. 123–134, 2024. DOI: 10.1556/606.2024.01139.
- [8] M. S. Hwang, C. C. Lee, and J. J. Hwang, "Cryptographic Applications of Artificial Intelligence: A Survey," *IEEE Access*, vol. 10, pp. 45210–45235, 2022, doi: 10.1109/ACCESS.2022.3163245.
- [9] N. Koblitz, "Elliptic Curve Cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987, doi: 10.1090/S0025-5718-1987-0866109-5.
- [10] S. M. Al-Nofaie, S. Sharaf, and R. Molla, "Design trends and comparative analysis of lightweight block ciphers for IoTs," *Applied Sciences*, vol. 15, no. 14, Art. no. 7740, 2025. [Online]. Available: <https://www.mdpi.com/2076-3417/15/14/7740>.
- [11] A. Singh, P. Kumar, and R. Sharma, "Enhancing Image Encryption Using Generative Adversarial Networks," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1234–1245, 2022, doi: 10.1109/TIFS.2022.3145573.

- [12] Y. Zhang, L. Zhou, and J. Wang, "A GAN-Based Cryptographic Key Generation Scheme for Image Encryption," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 3, pp. 210–225, 2024, doi: 10.1109/TDSC.2024.3221136.
- [13] H. Ahmed, M. T. Naseer, and S. Akram, "A Deep Learning Approach for DNA-Based Color Image Encryption," *IEEE Transactions on Multimedia*, vol. 24, no. 7, pp. 1238–1251, 2022, doi: 10.1109/TMM.2022.3149956.
- [14] M. Lai, J. Liu, and C. Wu, "Image Encryption Using a Memristive Hyperchaotic System and GANs," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 2, pp. 312–318, 2023, doi: 10.1109/TCSII.2023.3167152.
- [15] R. Nair, P. K. Gupta, and A. Verma, "Quantum-Resistant Image Encryption Using Deep Learning and Chaos Theory," *IEEE Journal on Selected Areas in Communications*, vol. 42, no. 1, pp. 145–158, 2024, doi: 10.1109/JSAC.2024.3204147.
- [16] M. Al-Alawi, H. Al-Sabbagh, and A. A. Al-Saeed, "Blockchain-Enabled Image Encryption Using Elliptic Curve Cryptography," *IEEE Transactions on Blockchain and Distributed Systems*, vol. 3, no. 1, pp. 45–56, 2023, doi: 10.1109/TBDS.2023.3206592.
- [17] M. Sharkawy and M. Badr, "DNA Coding and Steganography for Secure Image Encryption," *IEEE Transactions on Computational Imaging*, vol. 8, no. 5, pp. 789–804, 2023, doi: 10.1109/TCI.2023.3257814.