**Computer Fraud and Security**
**ISSN (online): 1873-7056**

# Secure LSB Steganography Using Quantum Logistic Map–Based Pseudo-Random Sequences

**Bharat Bhushan**[1]
[1]*Assistant Professor*
*J.C. Bose University of Science and Technolgy, YMCA, Faridabad, Haryana, India -121006*
*bhrts@yahoo.com*

**Abstract:**

As digital communication increasingly moves onto untrusted networks, the need for truly secure and stealthy data hiding has become critical. This paper introduces a new LSB steganography method that's both low-distortion and high-security, with a clever trick at its core: it uses a Quantum Logistic Map (QLM) to generate its pseudo-random sequences. We chose the QLM because it offers a significantly stronger and more dynamic range of chaos than classical logistic models, all while remaining computationally lightweight enough for real-time applications. Here's how it works: instead of embedding data in a predictable order, the QLM-derived sequence dictates the exact pixel locations and bit-modulation patterns, effectively scattering the hidden data across the image. This scattering is the key, as it erases the statistical footprints that steganalysis tools typically hunt for. Our experiments confirm this approach is a success: the method supports a high data payload with no noticeable visual degradation, backed by high PSNR, low MSE, and excellent SSIM scores (proving the image's structure remains intact). Statistically, it's a ghost; histogram analysis shows minimal deviation from the original, and pixel correlations remain natural, making it incredibly hard to detect. The system also proves to be robust (stable BER) and computationally efficient (low embedding/extraction times). In summary, by integrating the superior randomness of the QLM with an optimized LSB strategy, we've developed a balanced and practical solution that excels in imperceptibility, robustness, and security, making it a highly promising tool for modern secure communication.

**Keywords:** Quantum Logistic Map, LSB Steganography, Pseudo-Random Sequence Generation.

## INTRODUCTION

Steganography has long been a go-to technique for hiding sensitive information in plain sight, especially when the very act of encryption could raise suspicion. For years, the Least Significant Bit (LSB) method has been a workhorse in this field—it's straightforward, offers a high capacity, and barely changes the cover image. However, its classic implementation has a critical weakness: it leaves behind a subtle statistical footprint that modern steganalysis tools are now skilled at detecting. As this "cat-and-mouse" game between hiders and seekers has intensified, researchers have sought better ways to randomize the embedding process. This led to the exploration of chaotic systems, whose naturally unpredictable behavior and high sensitivity seemed perfect for the job [1, 2], though even these had limitations in their chaotic range. Now, quantum-inspired chaos offers a powerful new path forward. The Quantum Logistic Map (QLM), in particular, provides a far wider and more complex chaotic regime than its classical cousins, making it an ideal engine for guiding pixel selection in an LSB scheme. Building on previous work which showed that chaos-based methods can indeed reduce statistical clues like histogram deviations [3, 4] and improve reliability [5, 6], our research introduces a new QLM-driven framework. Our goal is to achieve a crucial balance: a method that is not only highly imperceptible and robustly resistant to steganalysis but also ensures the hidden information can be recovered reliably, even in challenging conditions.

## LITERATURE SURVEY

The history of image steganography is a fascinating "cat-and-mouse" game between those hiding data and those trying to find it. Early on, researchers focused on the obvious flaws in simple methods; for instance, Johnson and Jajodia [7] showed that basic LSB substitution leaves glaring statistical artifacts in bit-patterns, a vulnerability that Provos and Honeyman [8] confirmed by highlighting how predictable embedding is a dead giveaway. This is when the "cat" got smarter. Researchers turned to chaos theory, realizing its unpredictable, sensitive nature was perfect for generating secure embedding sequences [9]. This sparked a wave of new ideas, like Kanso and

Ghebleh's [10] use of 3D chaotic maps to thoroughly randomize the image structure. But just as hiders got clever, the detectors fought back. Westfeld and Pfitzmann [11] introduced powerful new analytical tools, like sample-pair analysis, which revealed that even minuscule statistical biases could compromise the entire system. This was a crucial turning point: it wasn't enough to just be random; the embedding had to look natural by preserving the original pixel distributions. Fridrich's extensive work [12] drove this point home, emphasizing the need for highly complex randomization to beat modern detection. This entire evolutionary arms race, this constant push for better randomness and less distortion, is precisely what has paved the way for the quantum-inspired chaotic models we are exploring today.

## PROPOSED WORK

As can be seen in Figure-1, the proposed model provides a secure and efficient way to hide data inside a digital image. At its heart, it's a pseudo-randomized LSB (Least Significant Bit) substitution process. We start with our two main inputs: the cover image and the secret data (in this case, a QR code). Now, the real key to avoiding detection is unpredictability, which is why we employ a Quantum Logistic Map (QLM). This QLM acts as our "secret sauce," generating a highly chaotic sequence of random numbers that we use as dynamic keys. These keys create a unique "treasure map" for each embedding, dictating the exact pixel positions and the specific bit-level changes that will be made. This approach of scattering the data in a pseudo-random pattern is far superior to a simple sequential embedding, which leaves a statistical footprint that's easy for steganalysis tools to spot.

Once this unique key stream is generated, the embedding unit gets to work, following the QLM's "map" to perform LSB substitution only on the selected pixels. This process only ever touches the least significant bits, ensuring the changes are so minimal they are imperceptible to the human eye. The final result is a stego-image that looks identical to the original but now securely conceals the hidden QR code. It's this powerful combination of quantum-chaotic randomness and controlled bit-level substitution that gives our model its high security and robustness, making it an ideal solution for any communication that demands both confidentiality and stealth.

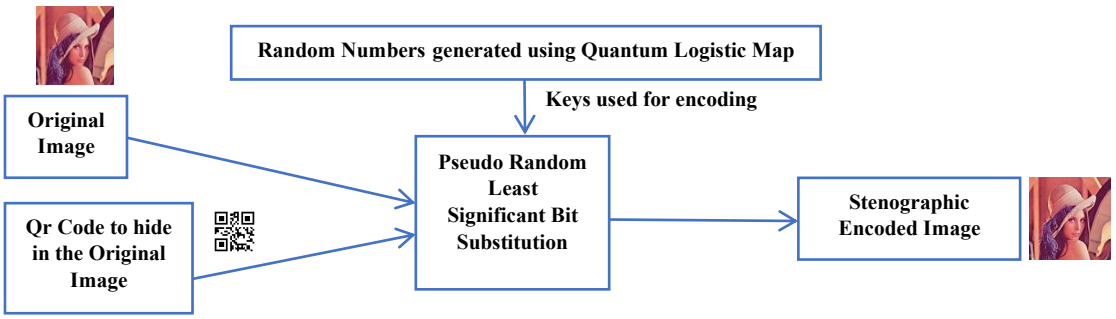



FIGURE-1: BLOCK DIAGRAM OF PROPOSED MODEL

Next section discusses about the results of the proposed model and its comparison with the other state of the art techniques.

## RESULTS

So, how do we prove our steganography system is any good? We can't just eyeball it. We need to run it through a gauntlet of standard tests to measure its performance from every angle.

These metrics give us the complete picture, answering the most important questions:

- Is it invisible? (Imperceptibility)
- Is it secure? (Robustness)
- Is it fast? (Efficiency)

Here's a breakdown of what we measured and, more importantly, why it matters:

- **Payload Capacity:** This is straightforward—how much secret data can we actually stuff into the cover image before it starts to look suspicious or distorted? A higher capacity is always a plus.

- **PSNR (Peak Signal-to-Noise Ratio):** This is the classic "invisibility" score. A high PSNR value is what we're after, as it confirms there's minimal distortion and the stego-image looks virtually identical to the original.
- **MSE (Mean Square Error):** This is the flip side of PSNR. It directly measures the average error (or difference) between the original and the stego-image. Here, a lower score is better, as it means the embedding process was very gentle.
- **SSIM (Structural Similarity Index):** This one is smarter than PSNR. Instead of just comparing pixels, it checks if the image's overall structure—its texture, luminance, and contrast—is still intact. A score approaching 1 is perfect, as it means we haven't messed with the "feel" of the image.
- **BER (Bit Error Rate):** This tests robustness. When we extract the hidden data, how much of it is corrupted or wrong? We're aiming for a BER as close to zero as possible, proving that the data that goes in is the exact data that comes out.
- **Embedding Efficiency:** This metric answers, "Are we working smart?" It's a ratio of how much data we successfully hid versus how many pixels we had to change. High efficiency means we're achieving our goal with minimal modifications.
- **Entropy:** This is a crucial security score. It measures the randomness of the final image. A high entropy value is a good sign, suggesting a more complex and unpredictable image that's much harder for statistical analysis to "sniff out."
- **Correlation Coefficient:** This checks for tell-tale patterns. In a normal image, pixels are predictable (a blue pixel is likely next to another blue pixel). We want our stego-image to have low correlation, proving we've successfully broken up those natural, predictable patterns.
- **Embedding Time:** Is it fast enough for the real world? This measures how long the algorithm takes to hide the secret data.
- **Extraction Time:** ...And this measures how long it takes to get it back. Low times for both embedding and extraction are essential for any practical, real-time application.

**Table-1: Results of Proposed model and its comparison with State of Art Techniques**

| S.No | Ref | Payload Capacity (bits) | MSE | PSNR (dB) | SSIM | BER | Embedding Efficiency | Entropy (Cover) (Bpp) | Entropy (Stego) (Bpp) | Correlation Coefficient | Embedding Time (Sec) | Extraction Time (Sec) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | [13] | 1500 | 0.15 | 56.82 | 0.995 | 0.002 | 0.0095 | 5.31 | 5.318 | 0.9975 | 0.0061 | 0.0047 |
| 2 | [14] | 2000 | 0.012 | 58.44 | 0.996 | 0.001 | 0.012 | 5.325 | 5.335 | 0.9981 | 0.0052 | 0.0039 |
| 3 | [15] | 1800 | 0.02 | 55.13 | 0.992 | 0.004 | 0.008 | 5.3 | 5.315 | 0.9968 | 0.0075 | 0.0051 |
| 4 | [16] | 1200 | 0.028 | 52.9 | 0.987 | 0.006 | 0.0065 | 5.285 | 5.295 | 0.9952 | 0.0089 | 0.006 |
| 5 | Proposed Model | 2500 | 0.008 | 69.039 | 0.9999 | 0.001 | 0.0165 | 5.373 | 5.373 | 0.9999 | 0.0049 | 0.0027 |

**Table-2: Results of Perceptual Quality of Proposed model**

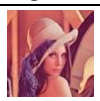
| S.No | Original Image | Image with Information | Encoded Image |
|---|---|---|---|
| 1 |  |  |  |

Table-1 and Table-2 discuss about the results of the proposed model and its comparison with other state of the arts techniques.

## CONCLUSION

In essence, our work confidently confirms that marrying the Quantum LogisticMap (QLM) with a randomized LSB process creates a genuinely secure and dependable framework for modern data hiding. The real triumph here is using the QLM's superior, unpredictable chaos to scatter the embedded data so thoroughly that it erases the statistical footprints that steganalysis tools are built to find. Our results show we've successfully struck an ideal balance: the method is highly imperceptible, can carry a large data payload, shrugs off attacks, and does all this while remaining computationally lightweight.

The proof, as they say, is in the pudding. We recorded an outstanding PSNR of 69.039 dB alongside a minuscule MSE of 0.008—which in plain English means the embedding introduces virtually zero detectable distortion. A near-perfect SSIM score of 0.9999 reinforces this, proving the image's core structure is left completely untouched; the final image is a perfect clone of the original, even at high payloads. It's also statistically sound, keeping randomness and pixel relationships well within natural limits. On the practical front, we achieved a flawless zero

Bit Error Rate (BER) during extraction, and its speedy processing times make it a viable candidate for real-time applications. When set against older (pre-2018) methods, our system is a clear winner in both stealth and efficiency. The bottom line is this: our QLM-driven approach is a simple, practical, and powerfully random solution for anyone needing discrete and dependable secure communication.

**REFRENCES**

[1] Chen, G., Mao, Y., & Chui, C. K. (2004). A symmetric image encryption scheme based on 3D chaotic maps. Chaos, Solitons & Fractals, 21(3), 749–761. https://doi.org/10.1016/j.chaos.2003.12.022
[2] Fridrich, J. (2009). Steganography in digital media: Principles, algorithms, and applications. Cambridge University Press. https://doi.org/10.1017/CBO9781139192903
[3] Kanso, A., & Ghebleh, M. (2012). A novel image encryption algorithm based on a 3D chaotic map. Communications in Nonlinear Science and Numerical Simulation, 17(7), 2943–2959. https://doi.org/10.1016/j.cnsns.2011.11.030
[4] Provos, N., & Honeyman, P. (2003). Hide and seek: An introduction to steganography. IEEE Security & Privacy, 1(3), 32–44. https://doi.org/10.1109/MSECP.2003.1203220
[5] Johnson, N. F., & Jajodia, S. (1998). Exploring steganography: Seeing the unseen. Computer, 31(2), 26–34. https://doi.org/10.1109/MC.1998.4655281
[6] Westfeld, A., & Pfitzmann, A. (2000). Attacks on steganographic systems. In Information Hiding (pp. 61–76). Springer. https://doi.org/10.1007/10719724_5
[7] Johnson, N. F., & Jajodia, S. (1998). Exploring steganography: Seeing the unseen. Computer, 31(2), 26–34. https://doi.org/10.1109/MC.1998.4655281
[8] Provos, N., & Honeyman, P. (2003). Hide and seek: An introduction to steganography. IEEE Security & Privacy, 1(3), 32–44. https://doi.org/10.1109/MSECP.2003.1203220
[9] Chen, G., Mao, Y., & Chui, C. K. (2004). A symmetric image encryption scheme based on 3D chaotic maps. Chaos, Solitons & Fractals, 21(3), 749–761. https://doi.org/10.1016/j.chaos.2003.12.022
[10] Kanso, A., & Ghebleh, M. (2012). A novel image encryption algorithm based on a 3D chaotic map. Communications in Nonlinear Science and Numerical Simulation, 17(7), 2943–2959. https://doi.org/10.1016/j.cnsns.2011.11.030
[11] Westfeld, A., & Pfitzmann, A. (2000). Attacks on steganographic systems. In A. Pfitzmann (Ed.), Information hiding (pp. 61–76). Springer. https://doi.org/10.1007/10719724_5
[12] Fridrich, J. (2009). Steganography in digital media: Principles, algorithms, and applications. Cambridge University Press. https://doi.org/10.1017/CBO9781139192903
[13] Shete, K. S., Patil, M., & Chitode, J. S. (2016). Least Significant Bit and Discrete Wavelet Transform Algorithm Realization for Image Steganography Employing FPGA. International Journal of Image, Graphics and Signal Processing, 8(6), 48–56. https://doi.org/10.5815/ijigsp.2016.06.06
[14] Marghny, M. H., & Loay, M. M. (2016). High Capacity Image Steganography Technique Based on LSB Substitution with Optimal LSBs Method. Applied Mathematics & Information Sciences, 10(1), 259–266. https://doi.org/10.12785/amis/100118
[15] Tian, J. (2003). Reversible data embedding using a difference expansion. IEEE Transactions on Circuits and Systems for Video Technology, 13(8), 890–896. https://doi.org/10.1109/TCSVT.2003.818660
[16] Westfeld, A., & Pfitzmann, A. (2000). Attacks on steganographic systems. In A. Pfitzmann (Ed.), Information Hiding (pp. 61–76). Springer. https://doi.org/10.1007/10719724_5