

Chaotic System Fingerprinting for Cryptographic Keystream Attribution: A Deep Learning Approach

Alexandru Dinu, Sara-Gabriela Modan, and Mădălin Frunzete(✉)

National University of Science and Technology POLITEHNICA Bucharest, Faculty of Electronics, Telecommunications and Information Technology, Bucharest, Romania
{alexandru.dinu2408,madalin.frunzete}@upb.ro,
sara.gabriela.modan@stud.etti.upb.ro

Abstract. Chaotic dynamical systems are widely used as entropy sources for key generation used for cryptographic purposes. This introduces a vulnerability: if an adversary were able to identify which chaotic system produced a given keystream, they would gain an important advantage for cryptanalysis. In this paper, we propose a deep learning framework for chaotic system fingerprinting, formulated in terms of a time-series classification problem over short keystream sequences. The model is trained and evaluated on sequences generated by four representative chaotic systems (Lorenz, Rössler, Sprott, Hénon 3D), using strict train/test separation. A regularized Residual Network (ResNet) model achieves over 92% accuracy on 4-level quantized sequences and over 86% on binary sequences, performing significantly better than frequency-based and classical Machine Learning (ML) baselines. An important observation is that these results hold even though the sequences pass standard statistical randomness tests which could make one think the classification results should be less positive. This demonstrates that high entropy does not prevent system identification. As a consequence, key generators based on chaotic systems remain vulnerable to system-level identification and require cryptographic post-processing regardless of their statistical properties.

Keywords: Chaotic systems · Deep learning · Cryptographic fingerprinting · ResNet · Security analysis

1 Introduction

Chaotic systems like Lorenz, Rössler, Sprott and Hénon are very attractive for cryptography because they are able to produce pseudo-random sequences from simple deterministic rules [1–4]. Their sensitivity to initial conditions creates large key spaces, and their outputs typically have no issues in passing standard randomness tests. This has led to widespread use in stream ciphers, image encryption, and secure communication, often under the (implicit) assumption

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2027
O. Gervasi et al. (Eds.): ICCSA 2026 Workshops, LNCS 16763, pp. 298–312, 2027.
https://doi.org/10.1007/978-3-032-30530-5_19

that statistical randomness is sufficient for security. However, the evaluation of such systems has in most of the cases relied on statistical criteria, such as entropy and standard randomness test suites, rather than on resistance to inference or attribution attacks. This leaves open the question of whether seemingly random outputs may still encode structural information about their generating dynamics.

A key assumption is behind the usage of chaos in cryptography: observing a keystream reveals nothing about which chaotic system generated it. If this assumption is not valid—i.e., if Lorenz leaves significantly different fingerprints/traces compared to Rössler or Sprott—an adversary who correctly identifies the source gains a concrete advantage. When an attacker knows the system used, this can enable targeted attacks: exploiting system-specific statistical weaknesses [6], making use of characteristic temporal correlations for parameter estimation, or applying phase space reconstruction specific to the identified attractor.

Even though there are rich decades of chaos-based cryptography research [3, 4], the question regarding the classification and correct identification of chaotic systems has not been systematically studied. Prior work focused on optimizing individual systems for statistical randomness and synchronization purposes. However, if different attractors produce outputs that a classifier can reliably differentiate did not receive the attention it deserves until now.

In the present paper, we address this topic in a direct manner. The question we started from is: "Given a short quantized sequence from an unknown chaotic source, can a deep learning model identify which attractor produced it?" We show that it can. Furthermore, this holds even for sequences that pass standard randomness tests and exhibit almost maximum informational entropy. A further finding concerns quantization: the same attractor can behave very differently depending on how its output is discretized (2 or 4 levels), yet remains equally identifiable under both use cases. These results suggest that entropy alone is not sufficient to assess the security of chaos-based key generators, and that cryptographic post-processing is necessary even for systems that appear statistically strong.

The remainder of this paper is organized as follows:

- Sect. 2 reviews related work.
- Sect. 3 describes the dataset, models, and experimental setup.
- Sect. 4 presents the results.
- Sect. 5 discusses security implications
- Sect. 6 presents the conclusions of this work.

2 Background and Related Work

2.1 Chaotic Dynamical Systems

Chaotic systems are deterministic nonlinear systems exhibiting extreme sensitivity to initial conditions [1]. A system is chaotic if it possesses at least one positive

Lyapunov exponent, indicating exponential divergence of trajectories which are close to each other. The four attractors studied here are defined as follows.

The Lorenz system (3D continuous, $\sigma = 10$, $\rho = 28$, $\beta = 8/3$):

$$\frac{dx}{dt} = \sigma(y - x), \quad \frac{dy}{dt} = x(\rho - z) - y, \quad \frac{dz}{dt} = xy - \beta z \tag{1}$$

exhibits a butterfly-shaped strange attractor with two lobes.

The Rössler system (3D continuous, $a = 0.2$, $b = 0.2$, $c = 5.7$):

$$\frac{dx}{dt} = -y - z, \quad \frac{dy}{dt} = x + ay, \quad \frac{dz}{dt} = b + z(x - c) \tag{2}$$

features a single-lobed attractor with folded ribbon topology.

The Sprott system (3D continuous, $a = 2.07$, $b = 1.79$):

$$\frac{dx}{dt} = y + axy + xz, \quad \frac{dy}{dt} = 1 - bx^2 + yz, \quad \frac{dz}{dt} = x - x^2 - y^2 \tag{3}$$

is time-reversible, with a state-dependent divergence that leads to both locally conservative and dissipative behavior across different regions of the phase space.

The Hénon 3D map (discrete-time, $a = 1.76$, $b = 0.1$):

$$x_{n+1} = a - y_n^2 - bz_n, \quad y_{n+1} = x_n, \quad z_{n+1} = y_n \tag{4}$$

is a three-dimensional extension of the classical 2D Hénon map, introducing delayed feedback through $z_n = y_{n-1}$. Positive Lyapunov exponents were verified numerically, confirming chaotic behavior suitable for cryptographic applications [1,7–10].

Figure 1 shows the phase portraits of all four systems. Each attractor has a structure which is distinct from a geometric point of view, with trajectories that start close to each other and gradually become distinct, but remain confined to the same attractor.

2.2 Chaos-Based Cryptography

Chaos-based cryptographic systems generate keystreams by quantizing trajectories from chaotic attractors which start from an initial condition and a clear set of system parameters [2,3]. A simple encryption scheme works by pairing each plaintext with a key value generated from the chaotic system:

$$C_i = M_i \oplus K_i, \quad \text{where } K_i = \text{quantize}(x_i).$$

In this approach, the continuous output of the chaotic system is first converted into discrete values, which are then used as the keystream.

Security analysis has traditionally focused on several security properties like: large key space, statistical randomness via NIST tests [5], and unpredictability from positive Lyapunov exponents [6]. Whether an attacker can establish which attractor produced a given keystream—the inter-system perspective—has not been sufficiently explored until now. This gap is important: successful fingerprinting enables targeted exploitation of system-specific weaknesses rather than generic cryptanalysis.

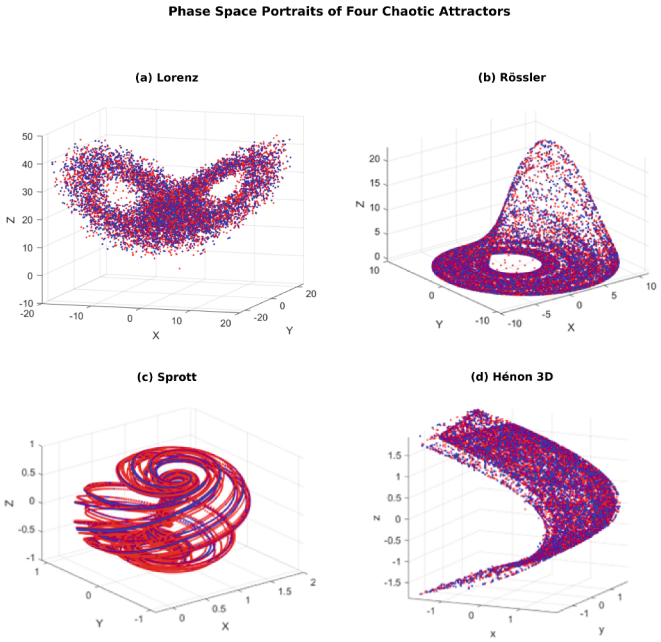

Fig. 1. Phase space portraits of the four chaotic attractors studied: (a) Lorenz, (b) Rössler, (c) Sprott, and (d) Hénon 3D.

2.3 Deep Learning for Time-Series and Cryptographic Attacks

Residual networks (ResNets) address vanishing gradients through skip connections, enabling deep architectures for temporal feature extraction via 1D convolutions [11]. Chaotic sequences present specific challenges [12]: short length and limited training data make overfitting severe, requiring aggressive regularization and extended training patience.

In the cryptographic attack literature, Machine Learning (ML) has been applied to problems such as modeling physical unclonable functions (PUFs), recovering the internal state of pseudo-random number generators (PRNGs), and identifying algorithms through side-channel leakage [13,14]. PUFs refer to hardware-based fingerprints that are difficult to replicate, PRNGs are deterministic generators designed to produce sequences that appear random, and side-channel attacks exploit unintended information leakage (such as timing or power consumption) from implementations.

Our work differs by targeting the mathematical structure of the underlying primitive—the attractor type—from outputs alone, without relying on physical side channels.

3 Methodology

3.1 Dataset Generation and Preprocessing

The data used was generated as follows:

1. Continuous systems were simulated using MATLAB's `ode45` solver, which provides a high-accuracy numerical approximation of the underlying differential equations. The step size is adaptive, ensuring numerical stability and reducing discretization artifacts.
2. The discrete Hénon 3D map trajectories were obtained through direct iteration.
3. Neural network models were implemented in Python using TensorFlow/Keras. No pre-trained weights were used; all models were trained from scratch on the generated dataset.

For each system, we generated $N = 1{,}000$ independent trajectories from random initial conditions $\mathbf{x}_0 \sim \mathcal{N}(0, I_3)$.

Statistical validity was ensured through two-stage validation: transient detection via histogram stabilization to identify attractor convergence time t_{trans}, and temporal independence testing via the U-V scatter chi-square analysis proposed by Badea & Vlad and used by the authors in previous research [8] to determine the minimum lag $\Delta t_{\min}$ satisfying

$$z = \sum_{k=1}^{M} \frac{(M_k - N/M)^2}{N/M} < \chi^2_{\alpha=0.05,\, M-1} \tag{5}$$

where samples were partitioned into $M = 48$ sectors (6 rings $\times$ 8 wedges). Table 1 summarizes the validated parameters. Hénon requires 1,300 s to reach its attractor (65$\times$ Lorenz), while Sprott demands 350 s spacing (22$\times$ Lorenz).

Table 1. Transient periods and minimum independence lags per system.

System	t_{trans} (s)	$\Delta t_{\min}$ (s)
Lorenz	20	16
Rössler	100	40
Sprott	100	350
Hénon 3D	1,300	200

After discarding transients, we extracted 150-timestep sequences by sampling the x-coordinate at intervals $\Delta t_{\min}$, ensuring temporal independence both within and across sequences.

For 4-level quantization, per-sequence uniform binning into $\{0, 1, 2, 3\}$ was applied:

$$s_i^{\text{4-level}} = \left\lfloor 4 \cdot \frac{s_i - \min(s_j)}{\max(s_j) - \min(s_j)} \right\rfloor \tag{6}$$

For binary quantization, per-sequence median thresholding produces balanced 0/1 distributions:

$$s_i^{\text{bin}} = \begin{cases} 0 & \text{if } s_i \leq \text{median}(\{s_j\}) \\ 1 & \text{if } s_i > \text{median}(\{s_j\}) \end{cases} \tag{7}$$

Per-sequence normalization ensures that temporal patterns, not amplitude scales, drive classification.

The 4-level dataset contains 726 sequences (580 train, 146 test). The binary dataset (344 sequences: 275 train, 69 test) is smaller because we retained only sequences satisfying $|N_0 - N_1| \leq 10$, as a simple balancing condition that ensures the number of zeros and ones remains approximately equal, improving cryptographic quality. Table 2 gives the class breakdown. While the dataset is not perfectly balanced across classes, the class distributions were kept consistent across training and testing splits. This reflects a realistic setting and allows the model to learn discriminative features without enforcing artificial uniformity. The source code is available upon request from the authors.

Table 2. Dataset composition for chaotic system classification.

System	4-Level		Binary	
	Train	Test	Train	Test
Lorenz	154	38	55	14
Rössler	151	38	101	25
Sprott	122	31	65	16
Hénon 3D	153	39	54	14
Total	580	146	275	69

3.2 Model Architectures

The four systems were selected to cover distinct attractor geometries and dynamical classes: Lorenz and Rössler are classical continuous-time systems with established cryptographic relevance [1,7], Sprott is a time-reversible continuous-time variant, and Hénon 3D is the only discrete-time map, included to probe whether the classifier exploits the continuous/discrete distinction. The architectures follow the benchmark protocol of Fawaz et al. [12] for time-series classification, ranging from a simple dense baseline to deep residual networks. We evaluate four architectures of increasing complexity on 150-timestep sequences.

Dense Simple (27.8K parameters) is a fully-connected network with two hidden layers (128 and 64 units, ReLU, dropout 0.3). It tests whether aggregate statistics alone are sufficient, without exploiting temporal structure.

Convolutional Neural NetworkS (CNN) Light (100.7K parameters) uses three Conv1D layers (64, 128, 128 filters; kernels 7, 5, 3), max pooling, batch normalization, and dropout 0.25.

ResNet Baseline (523.6K parameters) adds residual connections: three blocks (64, 128, 128 filters), each with three convolutional layers (kernels 8, 5, 3), batch normalization, ReLU, and skip connections via 1×1 convolution. It uses moderate regularization (L2=0.0005, dropout 0.2–0.3) and early stopping with learning rate reduction.

ResNet Robust uses three residual blocks (filters: 64, 128, 128), each containing three 1D convolutions with kernel sizes 8, 5, and 3 respectively, followed by batch normalization. Each block includes a shortcut connection (1×1 convolution + batch normalization). Figure 2 illustrates the architecture: input sequences (150×1) pass through an initial convolutional block and three residual blocks, followed by global average pooling and a dense classification head with dropout.

Both ResNet variants use Adam (lr=0.001) with stratified 5-fold cross-validation.

Training Strategy

All models use stratified 5-fold cross-validation, trained from scratch on four folds and evaluated on the held-out fold. We report mean accuracy and standard deviation across five runs. Training uses Adam with initial learning rate 0.001 and sparse categorical cross-entropy loss, with EarlyStopping (patience: 10 epochs for Dense, 20 for CNN and ResNet Baseline, 30 for ResNet Robust) and ReduceLROnPlateau (factor 0.5–0.7, patience 7–10 epochs). Batch sizes are 16 for Dense and ResNet Robust, 32 for CNN and ResNet Baseline.

4 Experimental Results

4.1 Overall Classification Performance

Table 3 shows cross-validation results on the 4-level quantized dataset. ResNet Baseline reaches the highest mean accuracy, while ResNet Robust trades a small amount of accuracy for substantially lower variance across folds—standard deviation drops by 28%. Both Dense Simple and CNN Light fall well short of the ResNet variants.

Table 3. Model performance on 4-level quantized dataset (726 sequences, 5-fold CV).

Model	Accuracy (%)	Macro F1	Params	Epochs
ResNet Baseline	**93.25 ± 1.98**	**0.933 ± 0.020**	523.6K	55.8 ± 9.7
ResNet Robust	**92.83 ± 1.43**	**0.928 ± 0.015**	523.6K	64.2 ± 12.4
CNN Light	73.83 ± 3.37	0.726 ± 0.041	100.7K	21.0 ± 0.0
Dense Simple	71.77 ± 2.01	0.719 ± 0.020	27.8K	20.6 ± 1.4

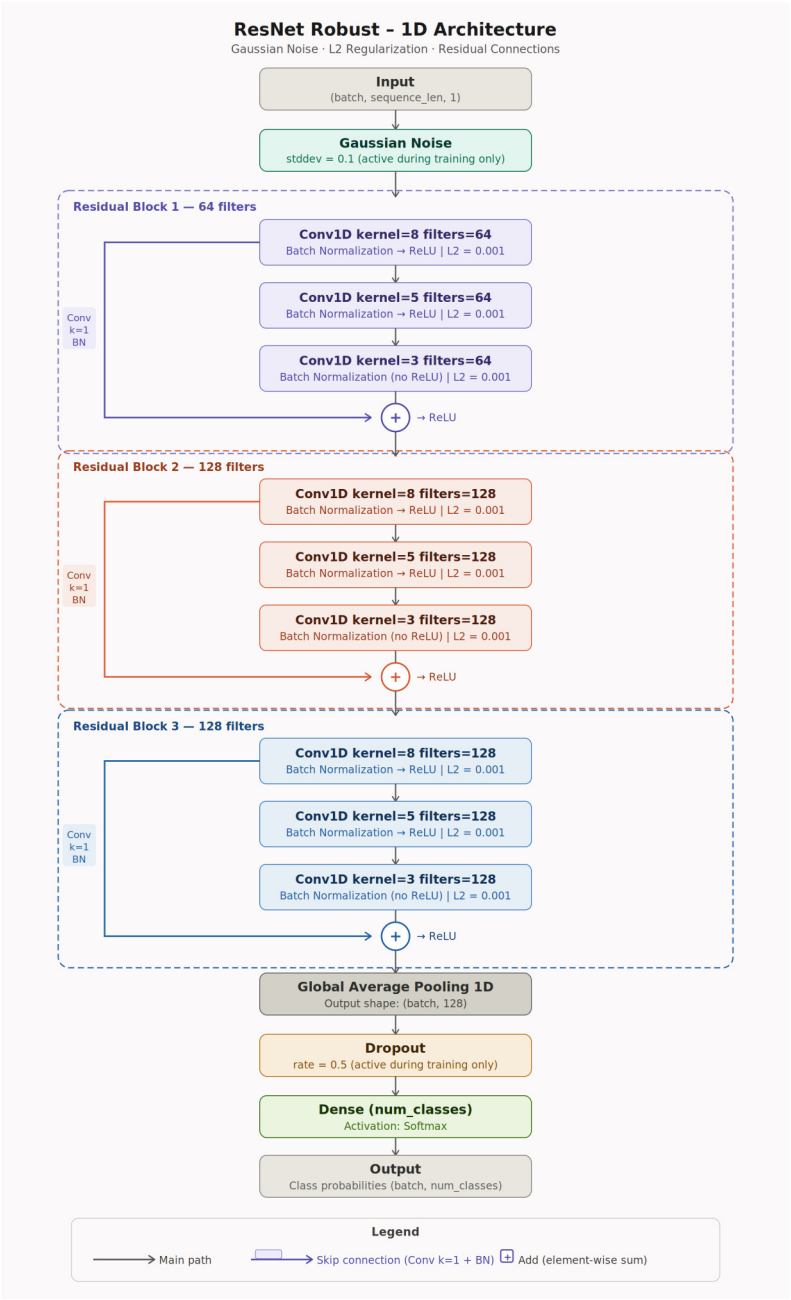

Fig. 2. ResNet Robust architecture for chaotic time-series classification (523,588 parameters).

On binary data the picture changes considerably. Table 4 shows that ResNet Baseline, despite being the best performer on 4-level data, collapses to near-random accuracy. With 523K parameters and only 275 training samples, it memorizes training noise within a few epochs. ResNet Robust, using the identical architecture but with stronger regularization, maintains high accuracy. The difference is entirely due to training strategy: Gaussian noise at input, stronger L2 weight decay, higher dropout, and extended patience allow the network to converge slowly and learn generalizable patterns rather than fit noise.

Table 4. Model performance on binarized dataset (344 sequences, 5-fold CV).

Model	Accuracy (%)	Macro F1	Params	Epochs
ResNet Robust	**86.34 ± 2.68**	**0.859 ± 0.026**	523.6K	92.0 ± 6.8
Dense Simple	59.59 ± 5.39	0.563 ± 0.061	27.8K	19.2 ± 2.0
CNN Light	51.46 ± 9.30	0.335 ± 0.064	100.7K	23.0 ± 0.0
ResNet Baseline	48.23 ± 8.83	0.357 ± 0.089	523.6K	24.2 ± 5.0

Table 5 summarizes how each model behaves when the representation is reduced from 4-level to binary quantization. As expected, this reduction leads to a loss of information and less training data. However, the ResNet Robust model maintains most of its performance, with only a 7% decrease. At the same time, the difference of 38.53% points between the two identically structured ResNet variants shows that regularization alone plays a major role, since all other factors are kept the same.

Table 5. Accuracy degradation from 4-level to binary quantization.

Model	4-Level (%)	Binary (%)	Δ (pp)	Δ (%)
ResNet Robust	**92.83 ± 1.43**	86.34 ± 2.68	−6.49	−7.0
Dense Simple	71.77 ± 2.01	59.59 ± 5.39	−12.18	−17.0
CNN Light	73.83 ± 3.37	51.46 ± 9.30	−22.37	−30.3
ResNet Baseline	**93.25 ± 1.98**	48.23 ± 8.83	−45.02	−48.3

4.2 Per-Class Analysis

Table 6 shows the F1-scores for each attractor. On the 4-level dataset, all four systems are classified with high reliability. The Hénon 3D system is the easiest to identify: both ResNet variants achieve 100% recall, meaning they do not miss any Hénon sequences. It should be noted that Hénon 3D is the only discrete-time system among the four; the remaining three are continuous-time ODEs.

The classifier may therefore partly exploit the continuous/discrete distinction rather than purely attractor identity. This does not invalidate the fingerprinting result, since both are real structural properties, but future work should evaluate classifiers restricted to sets of exclusively continuous-time or exclusively discrete-time attractors to isolate attractor-specific signatures.

Table 6. Per-class F1-scores (mean ± std, %) for ResNet architectures.

System	4-Level Quantized		Binary	
	Baseline	Robust	Baseline	Robust
Lorenz	90.3 ± 2.4	91.5 ± 2.0	18.8 ± 23.0	89.1 ± 6.7
Rössler	88.9 ± 3.0	89.0 ± 2.3	45.9 ± 25.1	86.2 ± 3.1
Sprott	95.5 ± 2.2	93.7 ± 1.6	74.2 ± 12.3	94.9 ± 4.5
Hénon 3D	98.7 ± 0.8	97.0 ± 2.0	4.1 ± 8.2	73.5 ± 5.1

Binary quantization reveals a clear difference between the two architectures. The ResNet Baseline model performs very poorly on the Lorenz and Hénon systems, and its high variability across folds suggests that its behavior is unstable and depends strongly on how the data is split, rather than on meaningful learned patterns. In contrast, the ResNet Robust model maintains performance above 73% F1 for all four systems, with low variance (under 7%), indicating consistent and stable learning. This shows that regularization does more than just prevent the model from collapsing to a trivial solution—it helps the network learn all classes instead of focusing too early on the dominant patterns during training.

Figure 3 shows confusion matrices from the best-performing fold. On 4-level data, the strong diagonal confirms reliable learning across all four systems, with Hénon 3D perfectly recalled. On binary data, Lorenz and Sprott achieve perfect precision in this fold, while Hénon 3D drops to 61.5% recall—still more than double the 25% random baseline.

4.3 Entropy vs. Classifiability

Table 7 compares Shannon entropy against classification performance on 4-level quantized data. Entropy values are computed across all sequences for each system (Lorenz: 192, Rössler: 189, Sprott: 153, Hénon: 192) after binarization via the median thresholding procedure already described.

Lorenz achieves near-perfect entropy and passes the chi-square uniformity test, yet remains highly classifiable. Hénon 3D reaches the highest F1-score of any system despite also passing randomness tests, with 100% recall in multiple folds. Rössler fails chi-square tests and shows high temporal correlation (lag-1 autocorrelation: 0.262), making it both cryptographically weak and easily identifiable—a dual failure. Sprott has the lowest entropy with extreme autocorrelation (lag-1:

Table 7. Shannon entropy versus classification performance on 4-level quantized data (post-binarization).

System	Entropy (H)	F1-Score (%)	Gap to Random	Chi^2 p	Status
Lorenz	0.9999	91.5 ± 2.0	+66.5 pp	0.123	Vulnerable
Rössler	0.9344	89.0 ± 2.3	+64.0 pp	$< 10^{-300}$	Critical
Sprott	0.7377	93.7 ± 1.6	+68.7 pp	$< 10^{-300}$	Critical
Hénon 3D	0.9812	97.0 ± 2.0	+72.0 pp	$< 10^{-164}$	Highly Vulnerable

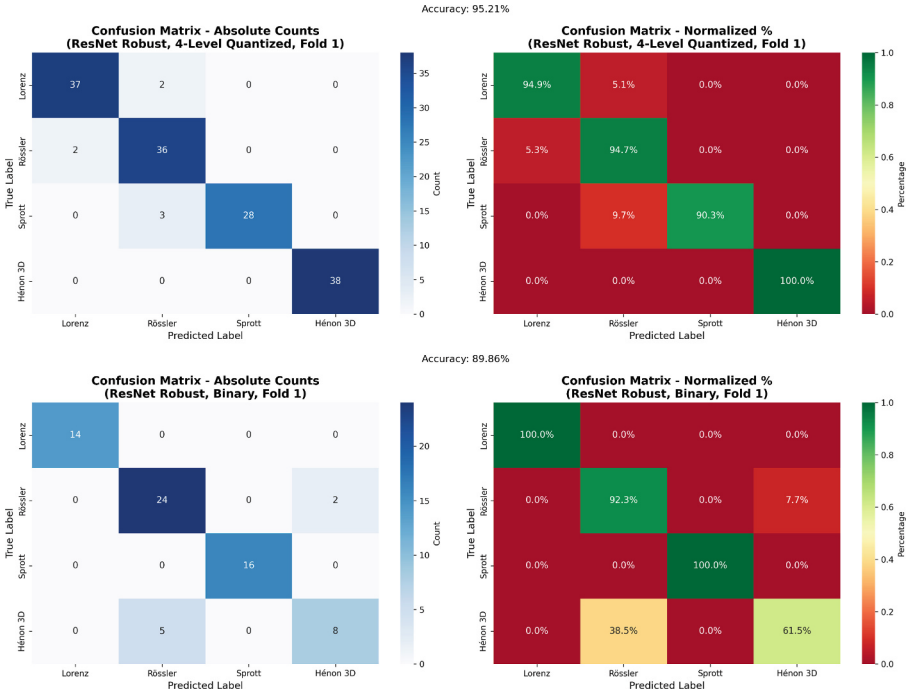

Fig. 3. Confusion matrices for ResNet Robust on the highest-performing cross-validation fold (shown for illustration); mean performance across folds is reported in Tables 3 and 4.

0.617) and a longest run of 162 consecutive identical bits out of 22,950 total, yet is classified just as reliably as Lorenz.

The reason standard randomness tests fail to capture this effect is that they focus on simple statistical properties of the bit sequences, such as frequency and uniformity. In contrast, deep learning models can detect more complex structures, including temporal dependencies and patterns that persist after quantization. As a result, even sequences that appear random under classical tests still contain identifiable signatures of their underlying dynamics.

Table 8 shows the same analysis on native binary data, where the impact of the quantization scheme becomes clear.

Table 8. Shannon entropy on native binary data (no median thresholding).

System	Entropy (H)	Δ vs Quant4	Chi^2 p	Runs p
Lorenz	0.9998	−0.0001	0.120	0.941
Rössler	0.9485	+0.0141	$< 10^{-292}$	$< 10^{-300}$
Sprott	0.9782	+0.2405	$< 10^{-81}$	$< 10^{-300}$
Hénon 3D	0.9897	+0.0085	$< 10^{-33}$	0.065

Sprott's entropy jumps from 0.7377 (post-binarized quant4) to 0.9782 (native binary)—a gain of 24% points. Median-threshold binarization of 4-level quantized Sprott sequences degrades entropy severely, while direct binary quantization preserves attractor structure. Despite this recovery, Sprott still fails statistical tests and ResNet Robust classifies both variants with comparable accuracy. Rössler recovers only modestly (+1.41 pp), continuing to fail uniformity tests with persistent temporal correlation. Lorenz and Hénon remain statistically robust across both schemes.

In practice, this means that evaluating entropy for a single quantization method is not enough. A chaos-based cryptosystem may appear secure under one discretization, but reveal significant weaknesses under another. At the same time, it can remain equally vulnerable to ML-based fingerprinting in both cases.

5 Security Implications

5.1 Threat Model

Consider an adversary who intercepts a keystream and aims to infer which chaotic system generated it. While the exact system is not known, the attacker may assume a set of commonly used candidates and train a classifier on representative data. Our results show that this inference is feasible with high accuracy, with individual folds exceeding 95%. Identifying the underlying system provides a concrete advantage, enabling targeted cryptanalysis based on system-specific statistical properties and dynamical behavior. It also facilitates parameter estimation and offers insight into the defender's cryptographic design choices.

The main way to address this issue is to apply cryptographic post-processing. Even if the generated sequences look highly random, they can still be identified if used directly. By hashing the output (for example, with SHA-256), the structural patterns specific to each chaotic system are removed, transforming the output into a sequence that is computationally indistinguishable from uniform random noise, and removing the structural signatures that the fingerprinting classifier exploits.

Designers must also validate randomness across multiple quantization schemes, since discretization choices are not cryptographically neutral—vulnerabilities can emerge from preprocessing decisions that appear innocuous. Testing a single pipeline is not sufficient. Additional defensive layers include multi-attractor hybrids [8] (e.g., $K_{\text{hybrid}} = K_{\text{Lorenz}} \oplus K_{\text{Hénon}}$), which make attribution harder even when individual components are fingerprintable, and one-time initial conditions to prevent adversaries from accumulating training data across sessions.

System-specific guidance: Hénon 3D should not be used as a sole entropy source. Lorenz can be deployed if paired with SHA-256 post-processing. Rössler and Sprott are not recommended without post-processing—both exhibit severe statistical failures in addition to high ML classifiability.

Discussion

The main observation which results from this study is that statistical randomness does not imply security against ML-based classification. A sequence can pass every standard randomness test and still be classified in a reliable manner based on its source attractor. The rationale is chaos's dual character: locally the output looks unpredictable, but globally the dynamics follow deterministic geometric patterns in phase space that deep networks can extract.

Training strategy is as important as the architecture. Two neural networks with identical structure—same parameters, same architecture—can behave very differently on scarce data depending solely on regularization strength. Aggressive regularization forces slower convergence and prevents the network from fitting noise, which turns out to be the essential factor when the size of training samples is small.

Sprott's behavior illustrates a subtler point: the same attractor can produce sequences with very different entropy values depending on how they are discretized, while the classification performance does not differ much in each case. This means that ML fingerprinting exploits properties that are complementary to what standard randomness metrics quantify. Validating a single quantization use case can offer a false sense of security in face of potential attacks.

There are limitations of this study and some of them include the use of fixed parameter values for each system, the supervised setting in which the adversary assumes a predefined/known set of candidate attractors, and the absence of hardware-based validation. Open questions can focus on sensitivity to parameter variations and generalization to previously less studied attractors (such as Chen or Chua [15–18]).

6 Conclusions

We have shown that chaos-based cryptographic keystreams can be vulnerable to ML-based fingerprinting. A well-regularized ResNet model identifies the source attractor/chaotic system with high accuracy. This is based on short quantized

sequences, and the performance is above the performance achieved by simpler baseline models, including the Dense Simple and CNN Light architectures. High entropy values of the quantized chaotic sequences do not prevent this: systems that pass standard randomness tests can be easily classified because deep learning exploits structural signatures that standard tests do not measure and do not fully take into account.

Regularization is the decisive factor when we are faced with scarce data. Two architecturally identical networks analyzed in this study differ by nearly 40% points in accuracy on scarce binary data, with the gap explained mainly by the training strategy.

The discretization approach has an influence as well. The same attractor can generate sequences with very different entropy values depending on the quantization method, and remain equally identifiable by a classifier. As a consequence, evaluating only one quantization scheme can be misleading, and multiple schemes should be considered before drawing any conclusions and proceeding to practical deployment.

From a practical point of view: chaos-based cryptographic systems require evaluation against ML fingerprinting in addition to statistical tests. Chaotic keystream generators must use cryptographic post-processing such as SHA-256 [19] to destroy structural signatures; and new metrics quantifying resistance to ML classification should come side by side with standard entropy assessments.

The fingerprinting approach proposed in this work could also be applied beyond cryptography, for instance to classify chaotic dynamics in areas such as neuroscience, climate modeling, industrial IoT, and financial time-series analysis. Future work could focus on robustness to parameter changes and evaluation across multiple quantization schemes.

Acknowledgments. This work was supported by UEFISCDI and National University of Science and Technology Politehnica Bucharest through projects PN-IV-P7-7.1-PTE-2024-0505 and UNSTPB-industry collaboration project 34808/19.09.2025.

References

1. Dinu, A.: Singularity, observability, and independence: unveiling Lorenz's cryptographic potential. Mathematics **12**(18), 2798 (2024). https://doi.org/10.3390/math12182798
2. Pecora, L.M., Carroll, T.L.: Synchronization in chaotic systems. Phys. Rev. Lett. **64**(8), 821–824 (1990). https://doi.org/10.1103/PhysRevLett.64.821
3. Kocarev, L.: Chaos-based cryptography: a brief overview. IEEE Circuits Syst. Mag. **1**(3), 6–21 (2001). https://doi.org/10.1109/7384.963463
4. Dinu, A., Frunzete, M.: Singularity, observability and statistical independence in the context of chaotic systems. Mathematics **11**(2), 305 (2023). https://doi.org/10.3390/math11020305
5. Bassham, L., et al.: A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. NIST Special Publication 800-22. National Institute of Standards and Technology, Gaithersburg, MD (2010). https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=906762

6. Alvarez, G., Li, S.: Some basic cryptographic requirements for chaos-based cryptosystems. Int. J. Bifurcation Chaos **16**(8), 2129–2151 (2006)
7. Dinu, A.: From chaos to security: a comparative study of Lorenz and Rössler systems in cryptography. Cryptography **9**(3), 58 (2025). https://doi.org/10.3390/cryptography9030058
8. Dinu, A., Frunzete, M.: Image encryption using chaotic maps: development, application, and analysis. Mathematics **13**(16), 2588 (2025). https://doi.org/10.3390/math13162588
9. Sprott, J.C.: A dynamical system with a strange attractor and invariant tori. Phys. Lett. A **378**(20), 1361–1363 (2014). https://doi.org/10.1016/j.physleta.2014.03.028
10. Dinu, A., Vlad, A.: Romanian printed language, statistical independence and the type II statistical error. In: Proceedings of the 2021 International Conference on Speech Technology and Human-Computer Dialogue (SpeD), pp. 120–125 (2021). https://doi.org/10.1109/SpeD53181.2021.9587397
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90
12. Fawaz, H.I., Forestier, G., Weber, J., Idoumghar, L., Muller, P.-A.: Deep learning for time series classification: a review. Data Min. Knowl. Discov. **33**(4), 917–963 (2019). https://doi.org/10.48550/arXiv.1809.04356
13. Maghrebi, H., Portigliatti, T., Prouff, E.: Breaking Cryptographic Implementations Using Deep Learning Techniques. In: Carlet, C., Hasan, M.A., Saraswat, V. (eds.) SPACE 2016. LNCS, vol. 10076, pp. 3–26. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49445-6_1
14. Picek, S., Perin, G., Mariot, L., Wu, L., Batina, L.: SOK: deep learning-based physical side-channel analysis. ACM Comput. Surv. **55**(11), 227:1–227:35 (2023). https://doi.org/10.1145/3569577
15. Chen, M., Yu, J., Yu, Q., Li, C., Bao, B.: A memristive diode bridge-based canonical Chua's circuit. Entropy **16**, 6464–6476 (2014). https://doi.org/10.3390/e16126464
16. Yang, Z., et al.: A Chua's chaotic chirp spread-spectrum power spectral homogenization strategy based on distribution transformation. Electronics **13**(12), 2296. https://doi.org/10.3390/electronics13122296
17. Duan, Z., et al.: Integrated circuit of a Chua's system based on the integral-differential nonlinear resistance with multi-path voltage-controlled oscillator. Micromachines **15**(3), 401. https://doi.org/10.3390/mi15030401
18. Boobalan, S., Gurunathan Arthanari, S.K.: Lorenz and Chua chaotic key-based dynamic substitution box for efficient image encryption. Symmetry **17**(8), 1296. https://doi.org/10.3390/sym17081296
19. National Institute of Standards and Technology: Secure Hash Standard (SHS). FIPS Publication 180-4. NIST, Gaithersburg, MD (2015). https://doi.org/10.6028/NIST.FIPS.180-4