Research article

Andrea Fratalocchi*, Adam Fleming, Claudio Conti and Andrea Di Falco

NIST-certified secure key generation via deep learning of physical unclonable functions in silica aerogels

https://doi.org/10.1515/nanoph-2020-0368
Received July 3, 2020; accepted September 30, 2020;
published online October 27, 2020

Abstract: Physical unclonable functions (PUFs) are complex physical objects that aim at overcoming the vulnerabilities of traditional cryptographic keys, promising a robust class of security primitives for different applications. Optical PUFs present advantages over traditional electronic realizations, namely, a stronger unclonability, but suffer from problems of reliability and weak unpredictability of the key. We here develop a two-step PUF generation strategy based on deep learning, which associates reliable keys verified against the National Institute of Standards and Technology (NIST) certification standards of true random generators for cryptography. The idea explored in this work is to decouple the design of the PUFs from the key generation and train a neural architecture to learn the mapping algorithm between the key and the PUF. We report experimental results with all-optical PUFs realized in silica aerogels and analyzed a population of 100 generated keys, each of 10,000 bit length. The key generated passed all tests required by the NIST standard, with proportion outcomes well beyond the NIST's recommended threshold. The two-step key generation strategy studied in this work can be generalized to any PUF based on either optical or electronic implementations. It can help the design of robust PUFs for both secure authentications and encrypted communications.

***Corresponding author: Andrea Fratalocchi**, PRIMALIGHT, Faculty of Electrical Engineering, Applied Mathematics and Computational Science, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia, E-mail: andrea.fratalocchi@kaust.edu.sa. https://orcid.org/0000-0001-6769-4439
Adam Fleming and Andrea Di Falco, University of St Andrews, St Andrews, Fife, UK
Claudio Conti, Institute for Complex Systems, National Research Council (ISC-CNR), Via dei Taurini 19, 00185 Rome, Italy; and Department of Physics, University Sapienza, Piazzale Aldo Moro 5, 00185 Rome, Italy

Keywords: artificial intelligence; complex light scattering; physical unclonable functions; random optical nanomaterials; security.

1 Introduction

The modern digital society relies on mobile and ubiquitous optoelectronic devices whose software and hardware security is becoming a global concern owing to the increasing number of disclosed attacks every day [1–4]. The emergence of smart cities, the Internet of things, cloud computing, and big data will generate more challenges in this field [5–8], calling for new opportunities in research. Current cryptography methods for addressing security issues center on the idea of having a digital key, which is safely stored and whose information remains unknown to an adversary. However, implementing this simple concept turns out to be a difficult task: software such as Trojan horses and malware, and side-channel attacks carried out by enemies with single access to the device, can expose the key and lead to security breaches [4, 9–12]. As Tim Cook (Apple CEO) emphasized in a recent interview [13]:

> *"If you put a key under the mat for the cops, a burglar can find it, too. Criminals are using every technology tool at their disposal to hack into people's accounts. If they know there's a key hidden somewhere, they will not stop until they find it."*

These considerations fueled the development of physical unclonable functions (PUFs) [14–16]. A PUF is an object composed of a disordered structure, such as, e.g., a light scatterer, which stores a physical key inside a material layer with no mathematical description. In these systems, a digital key is typically generated by first challenging the PUF with an input signal and then converting into a binary sequence the analog response measured in either time, space, or frequency. The main assumption is that the physical disorder of the PUF cannot be reverse engineered, not even by the original manufacturer. If the PUF is safely stored, an adversary who wants to recreate the key has the

only possibility of performing a brute force attack, which is practically unfeasible owing to the exponentially large complexity of a PUF [14].

In this field of research, photonics is pioneering technologies for different lines of applications, including authentication [17, 18], secure communications [19–21], and classical equivalent schemes to quantum key distribution with perfect secrecy [22]. The main advantage of photonics PUFs is strong device unclonability: while cloning electronic PUF implementations has been reported [23], no one was ever able to replicate an optical PUF. The main challenge in photonics is the development of general algorithms that transform the response of PUFs into digital keys that appear as unpredictable random sequences. The issue is the local correlations that are present in the PUF response: when transformed into a binary string with conventional techniques, a certain degree of correlation remains in the key and between different keys [24]. To the best of the authors' knowledge, with the exception of the study by Di Falco et al. [22], no optical PUFs has been verified against certification standards that guarantee the genuine unpredictability and uncorrelation of the keys, and no technique has been devised to address this problem controllably for optical PUFs.

Another difficulty originates from the fact that the complex PUFs are strongly sensitive to input conditions. When traditional analog-digital conversion methods are applied to generate the key, such sensibility can generate different keys for apparently identical input conditions [25]. The issue lies in the impossibility of reproducing the same input conditions in different experiments. In a strongly chaotic system such as a PUF, even a small variation in the input parameters can strongly affect the security primitive's reliability. If this problem is addressed, it could also open to new PUFs generated via, e.g., soft-like materials, including gels (e.g., hydrogel, aerogels) and foams. These materials are more input sensitive than solid-state counterparts and are currently not employed as security tokens. However, soft-like structures offer security advantages because their nanoscale disorder can reach a higher entropy than artificial human-made PUFs, which are intrinsically limited by the cost, resolution, and scalability of the present nanofabrication technology [26]. In this article, we propose to address the issues mentioned above by combining PUF with deep learning [27, 28]. We develop a general and versatile two-step key generation strategy, which guarantees the generation of truly random keys verified against the National Institute of Standards and Technology (NIST) standards for cryptographic applications [29], with each key entirely uncorrelated with the others and reliable. We experimentally demonstrate these results with a new class of nonlinear PUFs implemented with silica aerogels (SAs).

2 Results

2.1 All-optical PUFs with aerogels

SA is a material composed of an ultraporous network of sparse silica aggregates. The SA optical response can be adjusted from complete transparency to strongly chaotic scattering by controlling the silica inclusions' size and distribution by either mechanical or optical effects [30]. Owing to a low thermal conductivity, SA exhibits a very strong optothermal nonlinearity [31, 32], which is associated with large and reversible structural deformations, making SA a nonlinear controllable random material that can be employed in different lines of applications [33, 34].

The SA produced in this work is manufactured by a base-catalyzed polymerization process [35, 36], which starts by mixing tetramethyl orthosilicate, methanol, and ammonium hydroxide in a 2:4:1 ratio, producing a gel of good clarity and with minimal defects [37]. The mixture is then poured into a Teflon mold, producing a cuboid-shaped gel of 1 cm side. The gel is subsequently removed from the mold and then washed in a series of acetone baths, each lasting 24 h. The transition from wet gel to aerogel happens by using a low-temperature supercritical CO_2 drying process [38], with a custom setup assembled in our lab. Figure 1a illustrates the setup used to generate all-optical PUFs, acquired as speckle patterns obtained by illuminating the SA sample with a pump-probe configuration. The setup comprises an expanded monochromatic laser probe (wavelength, λ = 632 nm) and a collimated beam with λ = 488 nm waist of approximately 200 μm. The speckle patterns are converted into digital PUFs using a CCD camera placed after the sample (Figure 1a).

In the mapping procedure introduced in this work, it is possible to associate different keys with any class of PUFs that differ in at least a characteristic feature (e.g., distribution or shape of a speckle pattern) that we train the network to resolve. The universal approximation theorem of neural networks [39] guarantees, at least theoretically, that a single neural network that could address this problem exists. In the PUF image acquisition setup illustrated in Figure 1a, it is possible to create PUF images with different speckle features by changing either the laser pump power or the acquisition time. While different pump powers generate diverse characteristic speckles, each pump power triggers a slow dynamical evolution of the speckles over characteristic times of the order of seconds, generating

different PUFs in the CCD. Figure 1b shows a typical class of different PUFs that can be acquired at constant pump power (P = 200 mW) and at different times within 1 min of laser illumination. The speckles are observed to be repeatable owing to the good stability properties of SAs [40]. The primary source of entropy that triggers the generation of different speckle patterns in Figure 1b is the spatial fluctuation of the scattering centers of nanoparticles composing the aerogels. These depend on the thermodynamic condition (e.g., temperature, pressure, volume) of the aerogel.

While cloning the soft porous network of SA is hardly imaginable to be feasible now and in the long run owing to the ultradense packing of nanostructured silica components, employing this medium as a PUF generator is also challenging owing to the noticeable spatial fluctuations of the silica nanoparticles, which are visible in the PUF images collected by the CCD (Figure 1b). In the next section, we illustrate a general strategy to address this problem controllably.

2.2 Two-step key generation via deep learning

Figure 2a, b illustrates a high-level schematic of the proposed concept. Ideally, we would like to have at disposal a mapping function $\mathcal{M}$ that, given at the input one experimental PUF generated from the setup of Figure 1a, associates a key k_n with the following properties: i) each key k_n is uncorrelated to the others, ii) each key satisfies the NIST standard to be considered as a real random sequence, and iii) the same key associates with all PUFs experimentally obtained under the same input conditions, controlled with the reproducible accuracy experimentally available.

To address this problem on a general ground, we use a deep neural network (DNN) architecture (Figure 2a, b), which we train to learn the mapping function $\mathcal{M}$ satisfying constraints i)–iii). The DNN used in this work is a 2-layer feedforward neural architecture with a rectified linear unit neural activation function [41]. The network provides a classification of various PUFs into different digital keys k_n, with each key associating a class c_1, c_2, … , c_N of PUF images (Figure 2c, red, green, and blue colors). Each class c_n comprises a series I_{nm} of PUFs (m = 1, 2, …) that are experimentally obtained under the same input conditions but differ by statistical fluctuations arising in the experimental measurements. The number of PUFs included in each class is not necessarily the same. It can differ according to the fluctuations present for each input challenge considered in the interrogation process of the PUF. The union of all classes $\mathcal{C} \in (c_1, \ldots, c_N)$ constitutes the learning data set, which is fed to the DNN to learn the mapping function $\mathcal{M}$ and predict future keys when we interrogate

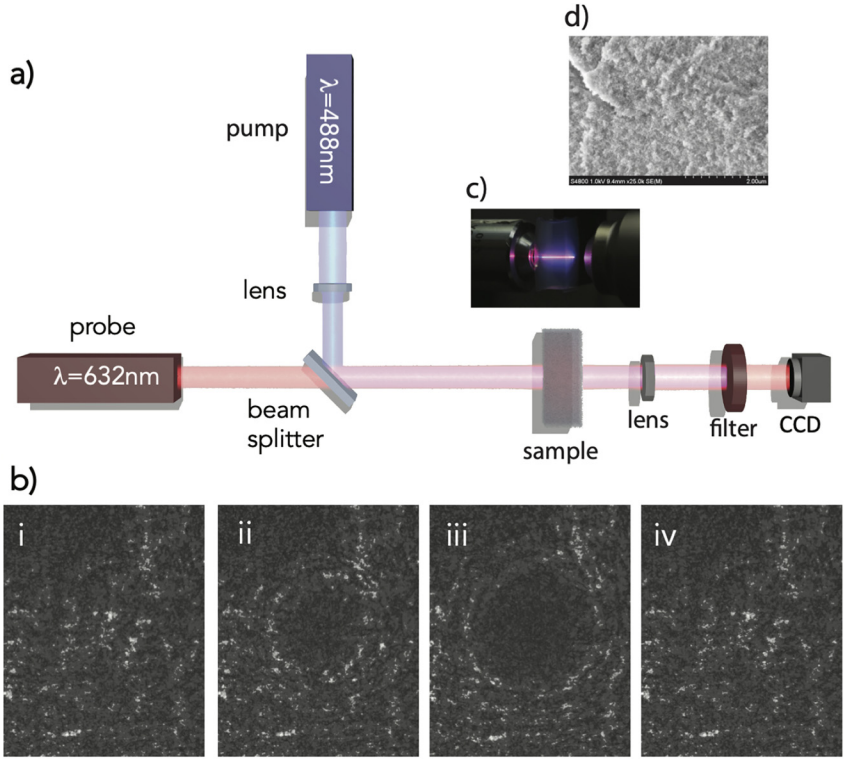

Figure 1: Experimental PUF setup. (a) All-optical PUF aerogel setup and configuration. (b) PUFs collected by the CCD as speckle patterns for t = 0 s (i), t = 20 s (ii), t = 40 s (iii), and t = 60 s (iv). (c) Picture of a real aerogel sample with (d) the corresponding scanning electron microscope (SEM) image. PUF, physical unclonable function.

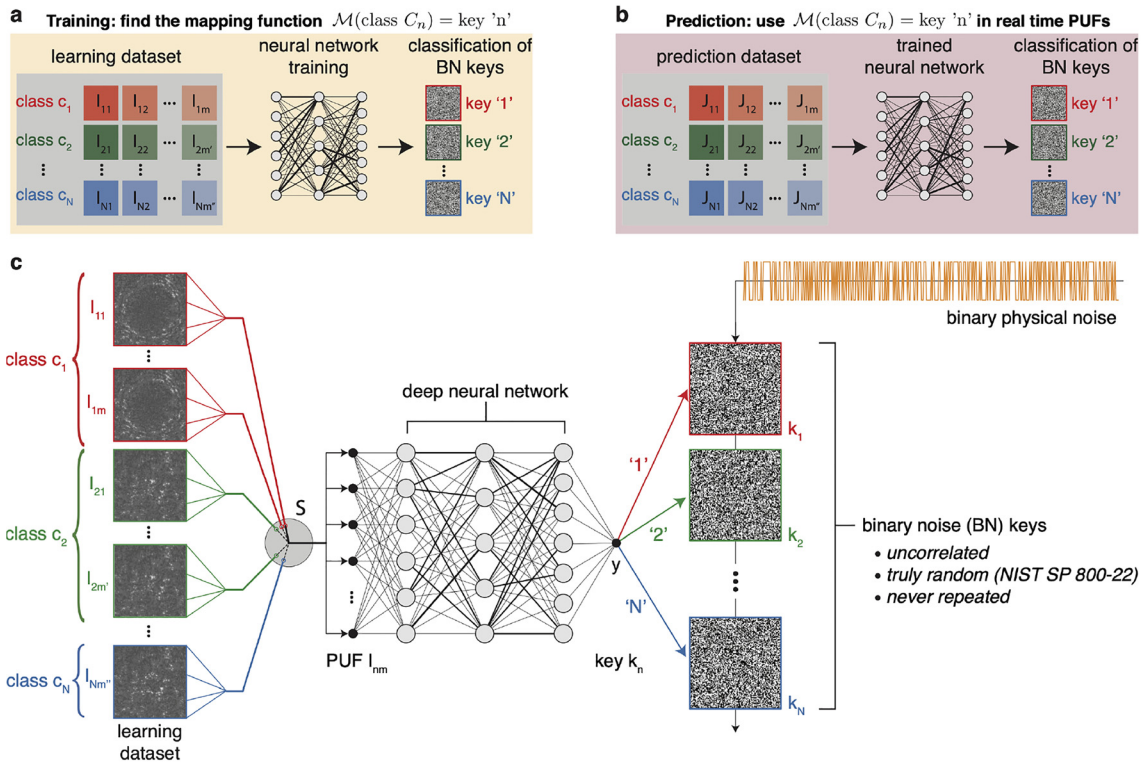

Figure 2: Two-step random key generation via deep learning.
(a–b) Overall process for the training (a) and prediction (b) of key classification and association with PUFs. (c) Detailed schematic workflow of the training procedure: a switch S selects a PUF I_{nm} at the input belonging to single classes c_n (red, green, and blue colors). Each class is mapped by a deep neural network to a different integer number $n = 1, \ldots, N$ at the output y, with each number identifying a binary noise (BN) key k_n. The space of different keys is generated independently by first sampling binary noise (orange solid line) and then splitting the random sequence into consecutive keys $k_1, k_2, \ldots, k_N$ of equal length. The prediction stage (b) employs the trained neural network of (a) in real-time to associate the keys with PUF images J_{mn} acquired in different experiments under the same input conditions of I_{mn} and subjected to experimental fluctuations of input parameters. PUF, physical unclonable function.

the PUF object again. To map PUFs I_{nm} to cryptographic keys k_n, we use a single DNN output channel y (Figure 2c), the latter identifying the output signal from the DNN, and train the network to associate each class c_n with a different integer $n = 1, \ldots, N$ at the output, with each integer n identifying a binary key k_n. Figure 2c illustrates this process visually with different colors, with each color showing the input-output association between a PUF class c_n and a key k_n.

Once we set the DNN weights, the network predicts the key association in future experiments with different classes of PUFs of J_{nm} (Figure 2b), measured under the same input conditions of I_{mn} but acquired in different experiments that differ by uncontrollable fluctuations of the input parameters. The main idea is to include a representative data set comprising a sufficiently large number of PUFs. The DNN learns the features of the experimental fluctuations associated with the different input conditions arising in each class c_n, becoming able to predict the future trends J_{mn} correctly. We increase the data set size until the DNN predicts correctly the key associated with a representative set of PUFs J_{mn} that does not exist in the training data set. When this occurs, the prediction (b) cross-validates the training (a), and it implies that the DNN has learned the required mapping function $\mathcal{M}$ with reasonable accuracy.

In this classification system, the error is the norm between the integer n identifying the key k_n and the output y corresponding to the PUF I_{nm} belonging to the class c_n. While more complex classification strategies are possible, we chose this method for its implementation simplicity.

We generate keys satisfying conditions i) and ii) by using binarized physical white noise. The latter is noise obtained by transforming in binary sequence a stream of white noise generated from a physical object and then split the binary stream into diverse keys $k_1, k_2, \ldots, k_n$ of predefined equal length (Figure 2b, orange binary signal). With the method proposed in this work, the generation of PUFs and cryptographic keys are two different problems that we address independently, overcoming the traditional issues that arise when mapping a complex PUF directly to a

binary sequence. The two problems are then combined via machine learning, using a DNN that finds the desired mapping that associates each PUF with a cryptography key.

The mapping function learned by the DNN conserves the security advantages of PUFs: it relies on a PUF object that has no mathematical representation, and it ensures a mapping between an input condition and a random key that cannot be guessed or recreated without the PUF object. From a security perspective, the DNN of Figure 2 acts as an additional, two-step protective layer to the PUF. If the PUF falls in the adversary's hand, the attacker cannot recreate the key without brute forcing all the DNN architecture weights. In a typical integrated electronic system, the space of the combination S_c that the enemy has to explore is $S_c = 2^{64 \cdot N_w}$ possibilities, with 2^{64} the combination required to assess the value of a 64-bit floating point number representing a single weight and N_w the number of network weights. In a DNN with $N_w \geq 4$, the space $S_c = 2^{64.N_w \geq 256}$ is larger than the space of 2^{256} combinations required to break the 256-bit advanced encryption standard, a NIST-certified cryptography in use by the US government to classify top secret information and presently considered unbreakable by brute force [42].

In the scheme of Figure 2, the DNN operation is typically evaluated by electronic CPU at gigahertz speed. It does not add overhead to the PUF key generation process, which is mainly limited by the camera's acquisition time of the optical PUFs.

2.3 Experimental results on PUF key generation and NIST validation

Figure 3 shows the typical results of cross-validation for two representative classes c_1 and c_2 of PUFs. Experimentally, we observe that classes c_n composed approximately of ≤10 PUFs are sufficient to train the DNN to perform accurate predictions. Figure 3a reports the learning rate obtained by the DNN when training on the learning data set composed of seven PUFs, with I_{11}–I_{13} belonging to c_1 and I_{21}–I_{24} belonging to c_2. These PUFs are acquired in the most fluctuating scenario in the setup of Figure 1a, in which we fix the input power (P = 200 mW) and acquire images at different times. Representative 100 × 100 pixel images of the PUF belonging to each class are shown in Figure 3b.

The results for Figure 3a illustrate that the DNN learns with great accuracy (learning errors below machine precision 10^{-15}) to associate the correct key number n with each PUF in the data set. Figure 3c and d report the resulting performances of the DNN when predicting the key number associated with the prediction data set, composed of six PUFs J_{11}–J_{13} of c_1 and J_{21}–J_{23} of class c_2. Images I_{mn} and J_{mn} are obtained in different experiments and have the same input conditions, i.e., pump power at P = 200 mW and same acquisition time. Although none of the J_{mn} PUFs exist in the learning data set used to train the DNN, the network correctly predicts the right index to each image, with prediction errors below 0.3. These results allow using a simple threshold filter $n \pm 0.5$ to assess correctly the key associated with each PUF, with no error arising from the natural fluctuations present in the experimental measures. The ability of the DNN to learn the feature of each PUF and the required mapping function $\mathcal{M}$ from few images is quite remarkable, especially considering the soft-like nature of the aerogel, whose scattering centers oscillate in time with large spatial fluctuations.

The error, or overfitting, between the DNN prediction and the correct key number in Figure 3c can be reduced by either increasing the length of the training data set and the associated DNN size or by adding a larger number of output channels, with each channel associating the corresponding key with a particular class of PUFs. To create random keys, we sample in time with an analog microphone, the sound arising from the electric engine of a desktop fan (Figure 4a). The sample rate of the microphone is much lower than the fan speed, allowing us to collect a time-varying random stream (Figure 4a, solid blue line). The sequence is then converted to a binary signal (Figure 4a, solid orange line) by using a Gabor transform [14], which associates 1 to all inputs above a threshold value, here chosen as the analog noise mean value. The random binary sequence generated is then partitioned into N different keys $k_1, \ldots, k_N$. We generate a set of N = 100 digital keys in our experiments, each of 10,000 bits.

Figure 4b reports the results of the NIST SP 800–22 test suite on the generated keys. The test comprises a suite of different statistical tests to assess if the key at the output looks like an unpredictable binary sequence in the input's absence of knowledge. The tests analyze the proportion of zeros and ones in the sequence and the existence of harmonic peaks (frequency, block frequency, and FFT), the presence of sequences with identical bits (run, longest run), the occurrence of prespecified target strings (nonoverlapping templates), the rank of disjoint submatrices in the stream (rank), and the sufficient complexity of the sequence to be considered random (serial and linear complexity). Detailed information on each test is available in the NIST reference [43]. Each test results in a proportion, which measures each key's probability to pass the statistical test (Figure 4b, dashed red line). The analysis is

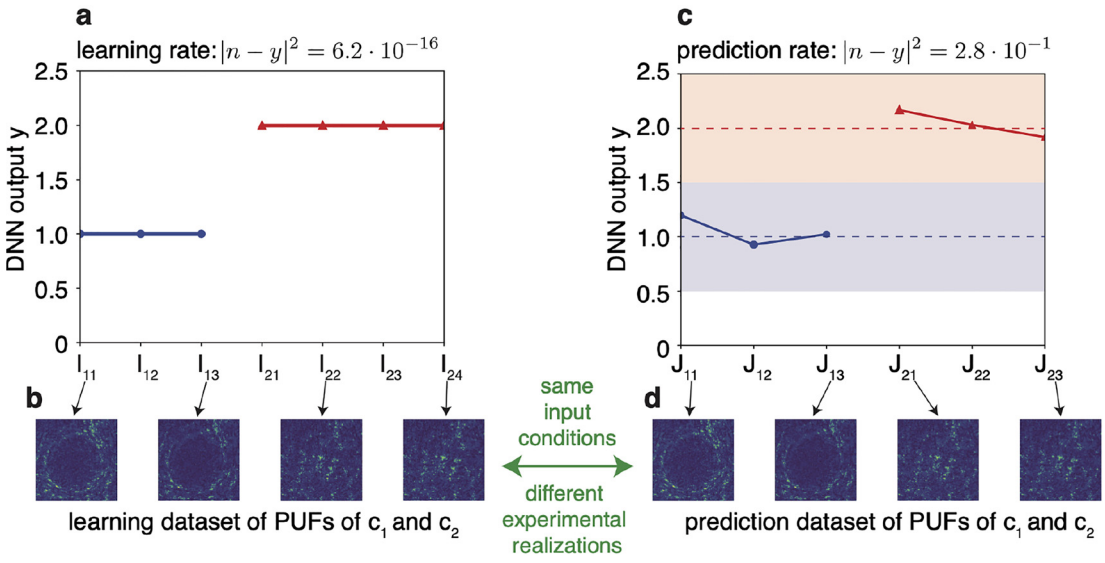

Figure 3: Deep neural network training and key association.
(a) Learning error in the training data set composed of seven PUFs belonging to classes c_1 and c_2, which are obtained with the same pumping power $P = 200$ mW and at different acquisition times. The PUFs I_{11}–I_{13} are associated with the output integer $y = 1$, while I_{21}–I_{24} are associated with $y = 2$. (b) CCD images of the PUFs. Panel (c–d) report the same analysis of (a–b) for the prediction data set, composed of six PUFs J_{11}–J_{13} of class c_1 and J_{21}–J_{23} of c_2. The PUFs in the prediction data set are generated in a different experiment and are not included in the training data set. PUF, physical unclonable function; DNN, deep neural network.

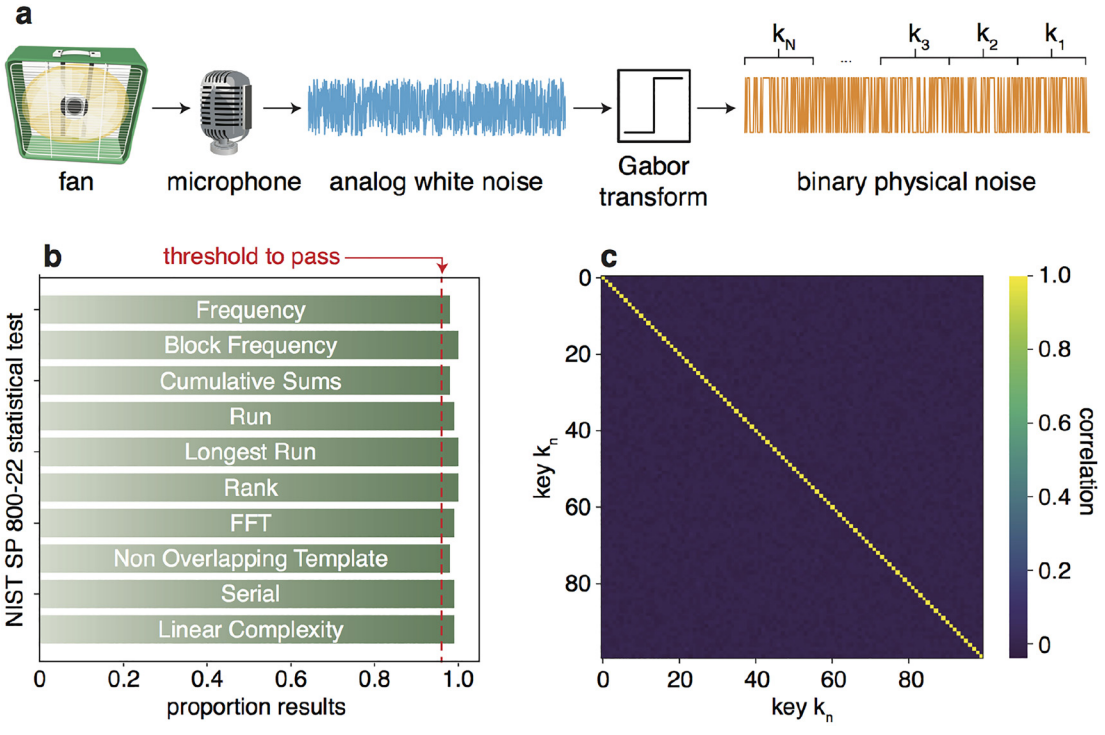

Figure 4: Key generation and NIST certification results.
(a) Generation steps of the binary noise sequence, starting from sampling in time, with a sufficiently low acquisition time, the noise emitted from the electric engine of a desktop fan, and then converting it into a binary sequence with a Gabor transform. The binary stream is then partitioned into $N = 100$ k_1, …, k_N consecutive keys each of 10,000 bits. (b) NIST proportion results on the SP 800-22 statistical test suite applied to the keys generated in (a), with a minimum threshold (dashed red line) recommended by the NIST. (c) Cross-correlation matrix between the keys k_n. NIST, National Institute of Standards and Technology.

performed using the software STS distributed by the NIST [43]. The results of Figure 4b demonstrate that the binary keys pass all the NIST tests well above the minimum threshold, showing that the procedure used to acquire the noise and transform it into a binary sequence generates a truly unpredictable stream of data (Figure 4a). Figure 4c reports the cross-correlation between the keys k_n. The keys generated are completely uncorrelated with each other, with average cross-correlation coefficients between the key k_i and k_j of the order of $\langle C_{ij} \rangle = 10^{-2}$.

Figures 2–4 demonstrate that the technique proposed in this works satisfies requirements i)–iii), with the reliable associations of the same key with experimental PUFs measured after the same challenge with no errors and with each key representing an unpredictable random sequence that is completely uncorrelated to the other.

3 Discussion

We discussed a two-step key generation strategy for PUFs based on deep learning, which can address the shortcomings of unreliability and weak unpredictability of cryptographic keys. The idea explored is to design the PUF independently from the problem of key generation and then use machine learning to train a neural network to find the complex mapping function that can reliably associate the features of PUFs with identical input conditions to a single key. Different binary keys were generated by sampling white noise, representing a physically unpredictable random sequence that passed all validation tests against NIST standards for cryptographic applications. We report experimental results in SAs, exploiting a classification strategy based on integer numbers n, with each number directly identifying a binary key k_n. Despite the high sensitivity of the aerogel to different input conditions, our experiments report that a trained neural network predicts the correct key with no errors. The results of this work can be of help in the development of stronger PUFs for different applications, including authentication and secure communications. The research data supporting this publication can be accessed at https://doi.org/10.17630/50b2f96f-ab3a-4b6e-abcd-c5d14c784de9.

Author contribution: All the authors have accepted responsibility for the entire content of this submitted manuscript and approved submission.

Research funding: C.C. acknowledge funding from Horizon 2020 Framework Programme QuantERA grant QUOMPLEX, by National Research Council (CNR), Grant agreement ID 731473.

Conflict of interest statement: The authors declare no conflicts of interest regarding this article.

References

[1] S. Sakhare and D. Sakhare, "A review—hardware security using puf (physical unclonable function)," in *ICCCE 2019*, A. Kumar, and S. Mozar, Eds., Singapore, Springer Singapore, 2020, pp. 373–377.

[2] D. Adam, "Cryptography on the front line," *Nature*, vol. 413, pp. 766–767, 2001.

[3] S. Chen, "Random number generators go public," *Science*, vol. 360, pp. 1383–1384, 2018.

[4] H. Wang, D. Forte, M. M. Tehranipoor, and Q. Shi, "Probing attacks on integrated circuits: challenges and research opportunities," *IEEE Design Test*, vol. 34, pp. 63–71, 2017.

[5] C. Tankard, "The security issues of the Internet of things," *Comput. Fraud. Secur.*, vol. 2015, pp. 11–14, 2015. Available at: http://www.sciencedirect.com/science/article/pii/S1361372315300841.

[6] F. S. Ferraz and C. A. G. Ferraz, "Smart city security issues: depicting information security issues in the role of an urban environment," in *2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*, 2014, pp. 842–847.

[7] V. Mayer-Schonberger and K. Cukier, *Big Data: A Revolution That Will Transform How We Live, Work, and Think*, Boston, Houghton Mifflin Harcourt, 2013. Available at: http://www.amazon.com/books/dp/0544002695.

[8] A. AlDairi and L. Tawalbeh, "Cyber security attacks on smart cities and associated mobile technologies," *Procedia Comput. Sci.*, vol. 109, pp. 1086–1091, 2017. Available at: http://www.sciencedirect.com/science/article/pii/S1877050917310669. *8th International Conference on Ambient Systems, Networks and Technologies, ANT-2017 and the 7th International Conference on Sustainable Energy Information Technology*, SEIT 2017, 16–19 May 2017, Madeira, Portugal.

[9] K. Thompson, "Reflections on trusting trust," *Commun. ACM*, vol. 27, pp. 761–763, 1984.

[10] S. Chen, R. Wang, X. Wang, and K. Zhang, "Side-channel leaks in web applications: A reality today, a challenge tomorrow," in *Proceedings of the IEEE Symposium on Security and Privacy (Oakland)*, IEEE Computer Society, 2010. Available at: https://www.microsoft.com/en-us/research/publication/side-channel-leaks-in-web-applications-a-reality-today-a-challenge-tomorrow/.

[11] C. Ashokkumar, R. P. Giri, and B. Menezes, "Highly efficient algorithms for AES key retrieval in cache access attacks," in *2016 IEEE European Symposium on Security and Privacy*, EuroS P, 2016, pp. 261–275.

[12] A. Golder, D. Das, J. Danial, et al., "Practical approaches toward deep-learning-based cross-device power side-channel attack," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 27, pp. 2720–2733, 2019.

[13] L. Kahney, *Tim Cook: The Genius Who Took Apple to the Next Level*, Penguin Books Limited, 2019. Available at: https://books.google.com.sa/books?id=A5xlDwAAQBAJ.
[14] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical one-way functions," *Science*, vol. 297, pp. 2026–2030, 2002.
[15] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, "Physical unclonable functions and applications: A tutorial," *Proc. IEEE*, vol. 102, pp. 1126–1141, 2014.
[16] B. Škorić, P. Tuyls, and W. Ophey, "Robust key extraction from physical uncloneable functions," in *International Conference on Applied Cryptography and Network Security*, Springer, 2005, pp. 407–422.
[17] S. A. Goorden, M. Horstmann, A. P. Mosk, B. Škorić, and P. W. H. Pinkse, "Quantum-secure authentication of a physical unclonable key," *Optica*, vol. 1, pp. 421–424, 2014.
[18] G. Zhang and Q. Liu, "A novel image encryption method based on total shuffling scheme," *Optic Commun.*, vol. 284, pp. 2775–2780, 2011.
[19] R. Horstmeyer, B. Judkewitz, I. M. Vellekoop, S. Assawaworrarit, and C. Yang, "Physical key-protected one-time pad," *Sci. Rep.*, vol. 3, no. 6, p. 3543, 2013.
[20] M. Leonetti, S. Karbasi, A. Mafi, E. DelRe, and C. Conti, "Secure information transport by transverse localization of light," *Sci. Rep.*, vol. 6, p. 29918, 2016.
[21] B. C. Grubel, B. T. Bosworth, M. Kossey, et al., "Secure communications using nonlinear silicon photonic keys," *Opt. Express*, vol. 26, pp. 4710–4722, 2018.
[22] A. Di Falco, V. Mazzone, A. Cruz, and A. Fratalocchi, "Perfect secrecy cryptography via mixing of chaotic waves in irreversible time-varying silicon chips," *Nat. Commun.*, vol. 10, p. 5827, 2019.
[23] C. Helfmeier, C. Boit, D. Nedospasov, and J.-P. Seifert, "Cloning physically unclonable functions," in *Hardware-Oriented Security and Trust (HOST), 2013 IEEE International Symposium on 1–6*, IEEE, 2013.
[24] U. Rührmair, "Optical pufs reloaded," Eprint.Iacr.Org, 2013, https://doi.org/10.1109/sp.2013.27.
[25] J. Danger, "Physically unclonable functions: principle, advantages and limitations," in *2019 International Conference on Advanced Technologies for Communications (ATC)*, 2019, pp. xxxii–xxxii.
[26] A. Wali, A. Dodda, Y. Wu, et al., "Biological physically unclonable function," *Commun. Phys.*, vol. 2, no. 39, 2019. Available at: https://doi.org/10.1038/s42005-019-0139-3.
[27] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, Cambridge, The MIT Press, 2016.
[28] G. Marcucci, D. Pierangeli, P. W. H. Pinkse, M. Malik, and C. Conti, "Programming multi-level quantum gates in disordered computing reservoirs via machine learning," *Opt. Express*, vol. 28, pp. 14018–14027, 2020. Available at: http://www.opticsexpress.org/abstract.cfm?URI=oe-28-9-14018.
[29] Bassham, L. E., Andrew, R., Juan, S., et al., "Sp 800-22 rev. 1a. a statistical test suite for random and pseudorandom number generators for cryptographic applications," Tech. Rep., 2010, https://doi.org/10.6028/nist.sp.800-22r1a.
[30] S. M. Jones, "A method for producing gradient density aerogel," *J. Sol. Gel Sci. Technol.*, vol. 44, pp. 255–258, 2007.
[31] S. Gentilini, F. Ghajeri, N. Ghofraniha, A. Di Falco, and C. Conti, "Optical shock waves in silica aerogel," *Opt. Express*, vol. 22, pp. 1667–1672, 2014.
[32] M. C. Braidotti, S. Gentilini, A. Fleming, M. C. Samuels, A. Di Falco, and C. Conti, "Optothermal nonlinearity of silica aerogel," *Appl. Phys. Lett.*, vol. 109, p. 041104, 2016.
[33] A. Fleming, C. Conti, and A. Di Falco, "Perturbation of transmission matrices in nonlinear random media," *Ann. Phys.*, vol. 531, p. 1900091, 2019.
[34] A. Fleming, C. Conti, T. Vettenburg, and A. Di Falco, "Nonlinear optical memory effect," *Opt. Lett.*, vol. 44, pp. 4841–4844, 2019.
[35] G. Nicolaon and S. Teichner, "The preparation of silica aerogels from methylorthosilicate in an alcoholic medium and their properties," 1975.
[36] J. Livage, M. Henry, and C. Sanchez, "Sol-gel chemistry of transition metal oxides," *Prog. Solid State Chem.*, vol. 18, pp. 259–341, 1988.
[37] B. Lin, S. Cui, X. Liu, X. Shen, Y. Liu, and G. Han, "Preparation and characterization of hmds modified hydrophobic silica aerogel," *Curr. Nanosci.*, 2011.
[38] P. H. Tewari, A. J. Hunt, and K. D. Lofftus, "Ambient-temperature supercritical drying of transparent silica aerogels," *Mater. Lett.*, vol. 3, pp. 363–367, 1985.
[39] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken, "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," *Neural Netw.*, vol. 6, pp. 861–867, 1993. Available at: http://www.sciencedirect.com/science/article/pii/S0893608005801315.
[40] E. Strobach, B. Bhatia, S. Yang, L. Zhao, and E. N. Wang, "High temperature stability of transparent silica aerogels for solar thermal applications," *APL Mater.*, vol. 7, p. 081104, 2019.
[41] R. H. R. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung, "Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit," *Nature*, vol. 405, pp. 947–951, 2000.
[42] J. Schwartz, "U.s. Selects a new encryption technique," 2000. Available at: https://www.nytimes.com/2000/10/03/business/technology-us-selects-a-new-encryption-technique.html.
[43] Computer Security Division, I. T. L, "Nist sp 800-22: documentation and software – random bit generation: Csrc,". Available at: https://csrc.nist.gov/projects/random-bit-generation/documentation-and-software.