Received: 23 June 2021 | Revised: 13 January 2022 | Accepted: 17 January 2022

DOI: 10.1049/ote2.12068

ORIGINAL RESEARCH

Random bit sequence generation from speckle patterns produced with multimode waveguides

Lília Maria Santos Dias[1] | **Tiago Filipe Santos Silvério**[1,2] | **Rute Amorim Sá Ferreira**[1] | **Paulo Sérgio de Brito André**[3]

[1]Department of Physics, CICECO – Aveiro Institute of Materials, University of Aveiro, Aveiro, Portugal

[2]Instituto de Telecomunicações, University of Aveiro, Aveiro, Portugal

[3]Department of Electrical and Computer Engineering, Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal

Correspondence

Rute Amorim Sá Ferreira, Department of Physics, CICECO – Aveiro Institute of Materials, University of Aveiro, Aveiro, 3810-193, Portugal.
Email: rferreira@ua.pt

Paulo Sérgio de Brito André, Department of Electrical and Computer Engineering, Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa, Avenida Rovisco Pais n.1., Lisbon, 1049-001, Portugal.
Email: paulo.andre@lx.it.pt

Funding information

Fundação para a Ciência e a Tecnologia, Grant/Award Numbers: CENTRO-01-0145-FEDER-030186, UIDB/50008/2020-UIDP/50008/2020, UIDB/50011/2020&UIDP/50011/2020

Abstract

With the rapid development of digital ecosystems, such as mobile applications towards goods/monetary transactions, a new paradigm of data transfer arises, which requires fast and reliable algorithms to generate random numbers. The statistical nature of speckle-based imaging creates an opportunity for these generators to arise as random number generators given the unpredictability and irreproducibility of such patterns. Hence, it is shown that the establishment of an experimental system is able to produce unique speckle patterns for remote cryptographic key storage and distribution, with a potential key rate generation of Gbs.

1 | INTRODUCTION

Random numbers are the basis of modern cryptography and random event simulations, being used in a wide range of applications such as weather forecasting, cryptocurrency systems, or the production of keys used to securely transmit data with Internet encryption protocols such as Transport Layer Security [1, 2]. These applications rely on the quality of the sequences provided by random bit generators, which produce a sequence of statistically independent and unbiased binary digits that cannot be predictable and are invulnerable to manipulation data [3].

Random (unpredictable) number generators produce random numbers from a physical process rather than through an algorithm, providing random sequences but with rigid approaches and higher cost factors. While, in theory, these stochastic processes can be predictable with full knowledge of all the parameters of the system, in practice, they can be considered unforeseeable with a limited time and computational resources, which contrasts with the paradigm of pseudo-random number generation commonly implemented in computer algorithms [4]. Moreover, considering the scope of optical physical unclonable functions, achieving a random number generator is advantageous to the intrinsic security of these systems. A study conducted by Di Falco et al. [5] highlights the use of irreversible time-varying measurements of chaotic systems granted by silicon chips, to achieve a perfect secrecy cryptography system.

Random number generation based on the recording of physical processes has been established for many years [6]. The most well-known example is described in Ref. [7] with the usage of lava lamp pictures to produce random data from the floating

Lília Maria Santos Dias and Tiago Filipe Santos Silvério contributed equally to this work.

pattern but with limited bandwidth. This issue has been considered in the context of the use of an light emission diode and a mobile phone camera to generate a random number based on the quantum nature of the light source but with a very complex and unbearable configuration [8]. Additionally, in the past 2 decades, photonic physical unclonable functions have received considerable attention as a promising way to provide unique cryptographic keys, with several implementations to ensure uniqueness, such as, among other, the intrinsic random roughness of a paper [9], or the speckle pattern obtained under coherent light illumination [10]. Furthermore, the potential for random number generators to arise due to the intrinsic randomness of optical physical unclonable functions has been successfully demonstrated using optical waveguides to achieve a key rate of Mbit/s [11, 12] with verified randomness, showing the promise of this technology. Other authors have explored speckle as a potential option for object detection or sensing [13], while further efforts were made by Fratalocchi et al. [14] towards an all-optical physical unclonable function that relies on the speckle patterns observed when illuminating an aerogel sample, achieving a verified secure key generator.

Hence, in this work, a novel approach of transforming speckle signals into random sequences is proposed, which can emulate a random number generator or be used as seeds to drive a computer algorithm as in pseudo-random number generation, increasing the randomness of the output.

2 | PROPOSED CONCEPT

The proposed concept consists of generating speckle images produced by multimode waveguides and their conversion into random bit strings. The random cryptographic key generation from images is undertaken using perceptual hash functions based on discrete cosine transform (DCT) hash. The DCT allows the conversion of data from the spatial domain into the frequency domain, and it is used to convert data into the summation of a series of cosine functions with different frequencies, producing real coefficients. The particularity of the DCT is that it allows expressing an image in small numbers of significant coefficients [3]. High frequencies are grouped in the lower right corner of the coefficient's matrix, and they represent the edge of an image (the less stable regions of the image under manipulation), and low frequencies represent homogeneous areas. Due to those characteristics, the DCT is commonly used in image and video treatment, and it is the most important part of JPEG and MPEG formats. There are some variants of this method, but the most common is the type-II DCT, the one that was implemented [3], defined by Equation (1):

$$X_{\mathrm{DCT}}[n] = \sum_{m=0}^{N-1} c[n,m]\, x(m) \quad n = 0, \ldots, N-1, \tag{1}$$

where $x(m)$, $m = 0, \ldots N - 1$ represents an N-point real signal sequence. The matrix c is called the DCT matrix, and it is defined as given in Equation (2).

$$c[n,m] = \sqrt{\frac{2}{N}} \cos\left(\frac{(2m+1)n\pi}{2N}\right), \quad m, n = 0, \ldots, N-1. \tag{2}$$

To apply it programmatically on an image, I is firstly necessary its preprocessing that typically consists of converting it to greyscale and the application of a mean filter with a kernel size 7 x 7. Then, the image is resized to 32 × 32 pixels, and since the resultant image is square, the two-dimensional DCT matrix of the image, $I_{DCT}(I)$, can be calculated following Equation (3) [3].

$$I_{\mathrm{DCT}}(I) = c \cdot I \cdot c'. \tag{3}$$

where c' represents the transpose matrix of the DCT matrix defined in Equation (2). After the calculation of the DCT matrix of the image, only the 32 high-frequency coefficients were extracted, and for that, a method of zigzag scanning was applied [15, 16]. The established preprocessing of the images was performed following Ref. [3] and attending to the size of the images on treatment. This process was not tested with different preprocessing parameters since the objective of the experiment was to prove the concept. However, these can be possibly adjusted for each particular case.

To retrieve the hash values of an image, the extracted coefficients are stringed together, forming a vector consisting of the coefficients B_i, $i = 1, \ldots, 32$ and the median m_d of the vector is calculated. Then, the vector can be normalised into a binary form, obtaining the final hash value, h_i, which represents the bit of the perceptual image hash at position i, following the Equation (4) [3]:

$$h_i = \begin{cases} 0, & B_i < m_d \\ 1, & B_i \geq m_d \end{cases} \tag{4}$$

Finally, the computed hash vector, h, can be evaluated for its potential as a random sequence. Once the hash vector is calculated, it can be compared to the hash vector obtained from another image through the normalised Hamming distance, Δ. This parameter measures the difference of two strings with dimension n, and it can be determined following the Equation (5), considering h_1 and h_2 as the hash vectors of the two images in comparison [3]:

$$\Delta(h_1, h_2) := \frac{1}{n} \sum_{i=1}^{n} |h_{i,1} - h_{i,2}| \tag{5}$$

To support the random nature of the bit-strings generated via this speckle-pattern method, the Hamming distance metric introduced in Equation (5) is used to identify that each of the speckle images generated a key that cannot be correlated with previous ones, without the full knowledge of the previous system parameters.

3 | EXPERIMENTAL IMPLEMENTATION

To generate a unique speckle pattern, a 2 m multimode plastic optical waveguide (*Avago Technologies*–HFBR-RUS100Z) was used as a scattering medium. The coherent optical source was a semiconductor laser (Roithner-DM650/3LJ) emitting at a wavelength of 650 nm with an optical power of 3 mW. After propagation through the waveguide, the optical signal is projected onto a semi-opaque film where an image can be recorded to later be digitally processed into a binary feature vector. The speckle is acquired with a Raspberry Pi camera (Raspicam) having a 1920 × 1080 pixel resolution, Signal-to-Noise Ratio of 36 dB, and a frame rate of 30 Hz. The camera is controlled by a Raspberry Pi 1 Model B (RPI), responsible for the image and digital signal processing using Matlab®. Figure 1 shows the supporting structures produced with Polylactic acid in a 3D printer (3D Systems, model Cube third Generation), for coupling the multimode waveguide tip to the camera and the opaque screen. For clearer and more noticeable speckle patterns, the camera acquisition settings must be changed, disabling all automatic effects. By disabling the automatic camera effects, our goal was to assure that the images taken were coincident with the reality of this process. It is also necessary to increase the contrast, brightness, sharpness, shutter speed, International Standards Organization (ISO) sensitivity, and decrease the saturation. The increasing of the shutter speed and ISO sensitivity is a crucial step, which improves the saturation limit and the detection of contours, increasing the sensitivity. By changing these settings such as saturation or brightness, for example, the objective is that from an aesthetic point of view, the pattern appears to be very distinguishable for the reader. In any way, these parameters should not affect the rest of the described process since they can be considered constant throughout the different images.

The advantage of using the Raspberry for image acquisition is the possibility to embed all the image acquisition and processing routines in the same platform, allowing the low-maintenance implementation of an online feed random number generator.

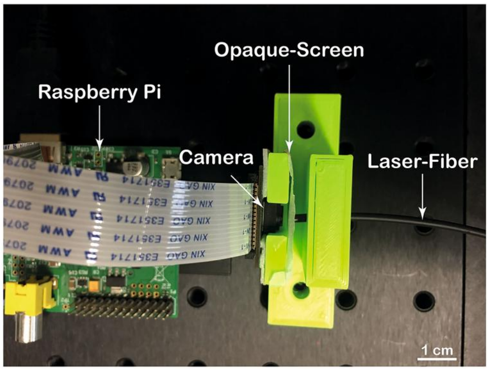

FIGURE 1 Photo of the experimental system used for the generation of random bit sequences based in the speckle images

4 | RESULTS

One of the methods of high data volume bit extraction is to collect several images of the speckle, which in this case, can be made by recording a video of the pattern for later characterisation. To demonstrate the proposed configuration, a 2 min video was collected, split into ∼3600 frames, and for each 1, the central region of the speckle with 515 × 515 pixels was cropped. Figure 2 illustrates a sample image obtained from a speckle pattern.

The type-II DCT was applied, and the 32 bit hash string was obtained for each cropped frame. Note that independent of the frame carrying 515 × 515 pixels, the bit hash size can be selected to be equal or less to the total number of pixels involved by selecting them using the zigzag scanning method. To analyse the independence of the strings retrieved by each frame, Figure 3 shows the mapping for the Hamming distance between 41 consecutive frames.

As previously indicated in Equation (5), the Hamming distances calculated in Figure 3 are a measure of how correlated are the *i*th and *j*th frames of the video. As the similarities between the frames increase, the Hamming distance will decrease, which is reflected by the similarities of the bits extracted from each image. Taking into consideration that theoretically, the Hamming distance between two random hash vectors tends to 0.5, the results attained in this study support that the frames are independent of each other, which demonstrates their eligibility to be considered random number generators.

After the evaluation of each retrieved string, the main challenge is to ensure that a generated sequence is a random number. To characterise the bit strings as such, several statistic tests may be employed to investigate the degree of randomness of a binary sequence and to check the generated bit string for specific weaknesses to targeted attacks. The obtained sequences were submitted to the standard NIST Randomness Tests suites [17] and the results are shown in Figure 4. Note that these results are expressed in terms of the acceptance rate

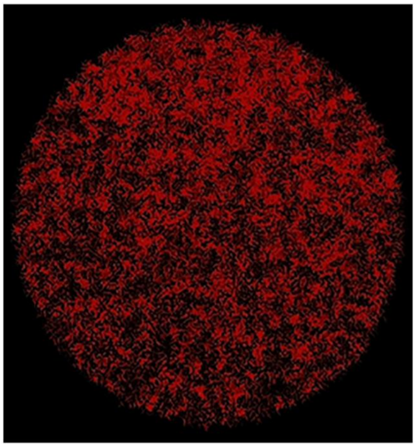

FIGURE 2 Sample frame of the obtained speckle patterns extracted from the video cropped into a 515 × 515 pixel region of interest for further analysis and key extraction

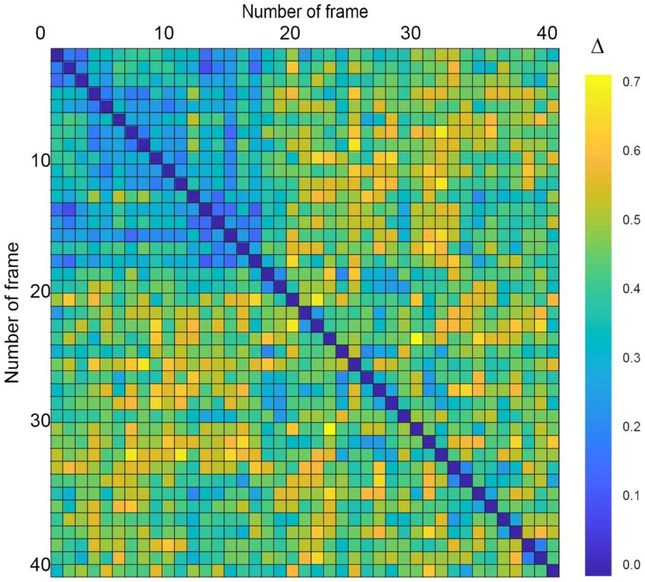

FIGURE 3 Mapped Hamming distance for the type-II DCT computed for 41 independent frames of the video, considering the 32 bit hash extracted from the cropped region

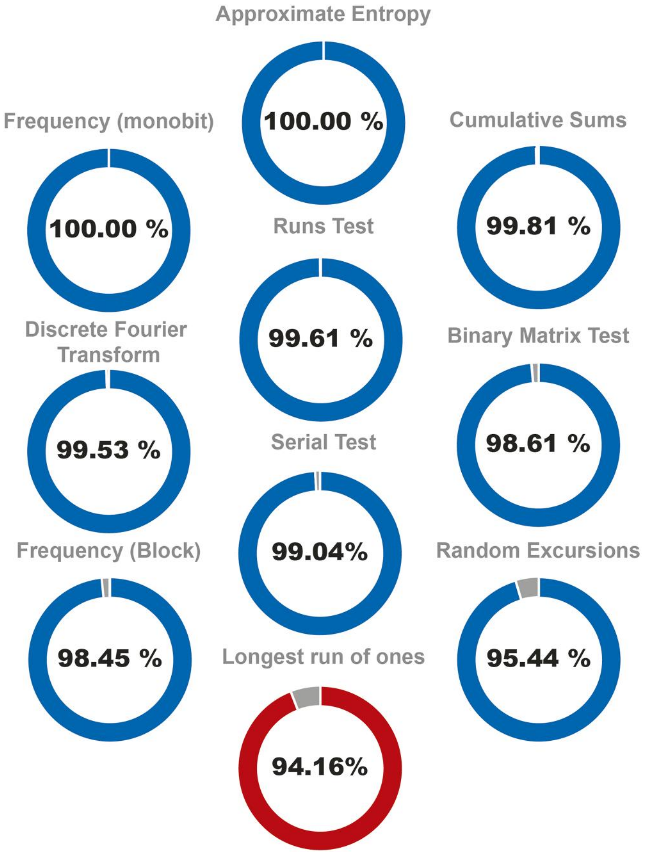

FIGURE 4 NIST statistical tests using the guidelines established in Ref. [17]. The results attained described the approval rates for a total of 3600 binary sequences retrieved from the multimedia files. Each of these sequences was retrieved using the 32-bit hash string of each video frame and submitted to the 10 NIST tests. Note that the 'Longest run of ones test' was the only test with an acceptance rate below the recommended threshold of 95%

of the submitted hash sequences, that is, the percentage of passed sequences from the universe of 3600 samples retrieved via this experimental method.

The results attained, with exception of the 'Binary Matrix Test', the 'Frequency (Block)', 'Random Excursions' and the 'Longest run of ones test', showed a high acceptance rate of the bit strings submitted to evaluation. Following the NIST guidelines [17], an acceptance rate threshold of 99% was established for each of the tests deployed. The attained failed tests, although may represent some aspects to improve in the key extraction method, do not jeopardise the assumed randomness that is displayed by the other tests. Note that the acceptance rate includes the evaluation of each of the 3600 strings of 32 bits. Thus, the total information appraised during this process was 144 kB, which also illustrates the potential for data generation of such systems. Furthermore, half of the failed tests were unsuccessful in achieving this recommendation by 0.5%, which represents a high margin for improvement of this experimental-based process. Notably, the NIST guidelines considered [17] are a compact set of tests to determine the suitability of random number generators. Thus, to further validate this approach, similar experiments need to be processed out with the full set of tests based on updated standards, such as the one described by NIST [18].

For many applications, such as the generation of cryptographic keys or gaming, speed is not as important as the affordability and portability given by this system. Consumer-grade devices acquire data at rates within 1 Gigapixel per second. After the necessary processing, each pixel will typically provide three random bits so that rates around 3 Gbps can be obtained. To sustain such high data rates, processing can either be done on a field-programmable gate array or it could be embedded directly on a complementary metal-oxide-semiconductor sensor chip. Implementing the extractor fully in software of a consumer device can sustain random bit rates surpassing the Mbps, largely sufficient for most consumer applications. Moreover, we believe that this low-cost number generation solution may be useful until solutions based on the optical quantum random number generator begin to be widely employed [19].

5 | CONCLUSIONS

We demonstrated a generator of random numbers based on the speckle pattern obtained with a simplistic implementation using consumer and portable electronics, providing very reasonable performance in terms of data rate. The use of a compact-sized computer for image acquisition and signal processing enhances the platform's possibility to be embedded with an important impact on information security. It is interesting that with a few adjustments, quantum experiments can also be done with consumer-grade hardware and that this may lead to the widespread use of quantum technology since the algorithm is supposed to continuously take pictures within short time intervals.

The results attained regarding the NIST randomness statistical tests were performed using a total of 144 kB of sequences generated via the speckle pattern produced by an optical diffuser. The outcome of these tests strongly suggests that the proposed system can be utilised as a 32-bit random number generator, with application in cryptography and secure systems.

ACKNOWLEDGEMENTS

This work was developed within the scope of the projects at CICECO-Aveiro Institute of Materials (UIDB/50011/2020&UIDP/50011/2020), Instituto de Telecomunicações (UIDB/50008/2020-UIDP/50008/2020), and the project Solar-Flex (CENTRO-01-0145-FEDER-030186), financed by national funds through the FCT/MEC and when appropriate co-financed by FEDER under the PT2020 Partnership Agreement through European Regional Development Fund in the frame of Operational Competitiveness and Internationalization Programme (POCI).

CONFLICT OF INTEREST

No conflict of interest registered.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Rute Amorim Sá Ferreira https://orcid.org/0000-0003-1085-7836
Paulo Sérgio de Brito André https://orcid.org/0000-0002-6276-4976

REFERENCES

1. Rai, V.K., Tripathy, S., Mathew, J.: Memristor based random number generator: architectures and evaluation. Procedia Comput. Sci. 125, 576–583 (2018)
2. Hameed Al-Moliki, Y., Alresheedi, M., Al-Harthi, Y.: Chaos-based physical-layer encryption for OFDM-based VLC schemes with robustness against known/chosen plaintext attacks. IET Optoelectron. 12, 12–14 (2018)
3. Zauner, C.: Implementation and Benchmarking of Perceptual Image Hash Functions. (2010)
4. Fischer, V., Drutarovský, M.: True random number generator embedded in reconfigurable hardware. In: Cryptographic Hardware and Embedded Systems - CHES 2002, pp. 415–430. Springer Berlin Heidelberg, Berlin (2003)
5. Di Falco, A., et al.: Perfect secrecy cryptography via mixing of chaotic waves in irreversible time-varying silicon chips. Nat. Commun. 10, (2019)
6. Ramos, R.V., Giraudo, E.C.: Optical random-bit generator employing quantum and chaotic dynamics. Microw. Opt. Technol. Lett. 39(4), 338–342 (2003). https://doi.org/10.1002/mop.11208
7. Noll, L.: LAVARND. (2000). http://www.lavarnd.org
8. Sanguinetti, B., et al.: Quantum random number generation on a mobile phone. Phys. Rev. X. 4(3), 031056 (2014)
9. Pino, A., et al.: Roughness measurement of paper using speckle. Opt. Eng. OPT ENG. 50, (2011)
10. Lehmann, P.: Surface-roughness measurement based on the intensity correlation function of scattered light under speckle-pattern illumination. Appl. Opt. 38(7), 1144–1152 (1999)
11. Akriotou, M., et al.: Random number generation from a secure photonic physical unclonable hardware module. In: Random Number Generation from a Secure Photonic Physical Unclonable Hardware Module: First International ISCIS Security Workshop 2018, Euro-CYBERSEC 2018, London, UK, February 26-27, 2018, Revised Selected Papers, pp. 28–37 (2018)
12. Chen, K., et al.: Fast random number generator based on optical physical unclonable functions, arXiv. vol. 2109.03325, (2021)
13. Kim, S., Ka, M.-H.: SAR raw data simulation for multiple-input multiple-output video synthetic aperture radar using beat frequency division frequency modulated continuous wave. Microw. Opt. Technol. Lett. 61(5), 1411–1418 (2019). https://doi.org/10.1002/mop.31748
14. Fratalocchi, A., et al.: Frontiers in optics and photonics. In: Federico, C., Dennis, C. (eds.) NIST-certified Secure Key Generation via Deep Learning of Physical Unclonable Functions in Silica Aerogels, pp. 471–478. De Gruyter (2021)
15. Shah, K.: Zigzag scanning of a matrix. MathWorks - File Exchange, vol. version 1.1.0.0 (2014)
16. Tang, Z., et al.: DCT and DWT based image hashing for copy detection. ICIC Express Lett. 7, 2961–2967 (2013)
17. Rukhin, J.S.A., et al.: A Statistical Test Suite for Random and Pseudo-random Number Generators for Cryptographic Applications. NIST Special Publication 800-22rev1 (2008)
18. Bassham, L., et al.: A Statistical Test Suite for Random and Pseudo-random Number Generators for Cryptographic Applications. Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg (2010). [online].
19. Stipčević, M., Ursin, R.: An on-demand optical quantum random number generator with in-future action and ultra-fast response. Sci. Rep. 5(1), 10214 (2015)

How to cite this article: Dias, L.M.S., et al.: Random bit sequence generation from speckle patterns produced with multimode waveguides. IET Optoelectron. 16(4), 174–178 (2022). https://doi.org/10.1049/ote2.12068