Generation of an EDS Key Based on a Graphic Image of a Subject's Face Using the RC4 Algorithm

Article *in* Information (Switzerland) · January 2021

DOI: 10.3390/info12010019

Anton Konev
Tomsk State University of Control Systems and Radioelectronics
58 PUBLICATIONS **175** CITATIONS

information

MDPI

Article

Generation of an EDS Key Based on a Graphic Image of a Subject's Face Using the RC4 Algorithm

Alexey Semenkov, Dmitry Bragin, Yakov Usoltsev, Anton Konev and Evgeny Kostuchenko *

Faculty of Security, Tomsk State University of Control Systems and Radioelectronics, 40 Lenina Prospect, 634050 Tomsk, Russia; office@keva.tusur.ru (A.S.); bds@csp.tusur.ru (D.B.); 7272_uya@fb.tusur.ru (Y.U.); kaa1@keva.tusur.ru (A.K.)

* Correspondence: key@keva.tusur.ru; Tel.: +7-923-444-4224

Abstract: Modern facial recognition algorithms make it possible to identify system users by their appearance with a high level of accuracy. In such cases, an image of the user's face is converted to parameters that later are used in a recognition process. On the other hand, the obtained parameters can be used as data for pseudo-random number generators. However, the closeness of the sequence generated by such a generator to a truly random one is questionable. This paper proposes a system which is able to authenticate users by their face, and generate pseudo-random values based on the facial image that will later serve to generate an encryption key. The generator of a random value was tested with the NIST Statistical Test Suite. The subsystem of image recognition was also tested under various conditions of taking the image. The test results of the random value generator show a satisfactory level of randomness, i.e., an average of 0.47 random generation (NIST test), with 95% accuracy of the system as a whole.

Keywords: digital signature; computer vision; cryptography; security; authenticity; algorithms; programming; python; neural networks

check for updates

Citation: Semenkov, A.; Bragin, D.; Usoltsev, Y.; Konev, A.; Kostuchenko, E. Generation of an EDS Key Based on a Graphic Image of a Subject's Face Using the RC4 Algorithm. *Information* **2021**, *12*, 19. https://doi.org/10.3390/info12010019

Received: 13 November 2020
Accepted: 29 December 2020
Published: 6 January 2021

1. Introduction

The recognition of images presented in the form of pictures has been widely researched. Some methods work by recognizing either a wide range of different classes that have undergone various conversions [1] or highly specialized ones, e.g., recognition of sea mammals from open source images [2]. Such digital potential relies upon neural networks and various methods for processing input images. Processing the input image extracts basic parameters, making it possible to identify images and remove redundant elements in the original image.

The recognition of human faces and subsequent identification of an individual by photo, video, or 3D modelling is a special subcategory of image recognition. Despite the fact that facial recognition is only a small part of a wider theme, there are a few dozen different methods and algorithms which are dedicated to this task. Each of the existing algorithms demonstrates the best or the worst result under various input data type configurations, face locations relative to the camera, emotions, illumination, resolution and the occlusion level of picture [3]. In this study, it is proposed combinations of mathematical procedures be used for facial recognition:

1. Coupling algorithms based on fractional-order-PCA-SVM [4] in order to achieve recognition accuracies above 98% with relatively high execution speed and without the requirement of high quality images as the input;
2. Genetic algorithms which are capable of recognizing a face only by its parts, with an accuracy of up to 93% [5];
3. Neural networks which are capable of achieving recognition accuracies above 98% with relatively high execution speed [6], and improved neural networks, which are

able to use low-quality images or images in which a subject's face is turned to the side [7].

In addition to facial recognition, other authentication methods are used, including some based on biometric features [8]. However, in this research, we were interested only in biometric methods, because they require neither the memorization of passwords nor the storage of technical authenticators; rather, these methods relate to aspects which are present from birth and which are, therefore, inseparable from the individual. Also, biometric authenticators require many passwords, which may be compromised if they are used in a different context. Biometric authenticators are both understandable and familiar to the average person, e.g., static parameters of the human body, such as the voice [9,10], may be recognized using different platforms and algorithms. Among dynamic biometric authenticators, one can distinguish, for example, the dynamics of applying a signature [11,12]. Note that, as in the case of image recognition, to recognize a person by the characteristics of his/her signature, that signature must undergo a number of transformations and be represented as a set of specific parameters [13]. In addition, the glyph does not necessarily have to be a unique signature. Studies [14] have shown that even the dynamics of writing ordinary words or the application of a single control phrase [15] can serve as a unique biometric identifier that makes it possible, using algorithms based on graphs, to recognize the person who wrote the word. However, biometric authentication methods based on handwriting input dynamics require additional equipment which is not necessarily widely available. Fortunately, other unique dynamic authentication parameters exist, such as keyboard rhythm. Some solutions use both common physical keyboards of personal computers [16] and more mobile solutions, e.g., the virtual keyboard on a smartphone [17]. A similar method of authentication is based on computer mouse dynamics [18]. Unfortunately, biometric authentication methods are often based on the use of neural networks which are subject to a number of attacks whereby it is possible to imitate a legal user [19]; additionally, the biometric recognition algorithm itself has vulnerabilities [20]. Removing such vulnerabilities is one of the major challenges of our time. However, it is possible to reduce the impact of errors arising from biometric authentication by using several biological parameters instead of just one. Different combinations are possible, e.g., authentication by voice and face makes it possible to reduce the error that occurs when authenticating these parameters individually [21,22]. Also, authentication by static and dynamic biometric parameters simultaneously, namely, by face and signature dynamics or by face and keyboard usage [23,24], has been studied.

The last aspect to consider is the generation of a pseudo-random value. This task arises every time it is necessary to generate encryption keys, so just like biometric authentication methods, there are many ways to create a pseudo-random number generator (PRNG). Pseudo-random number generation is a problem because the resulting sequence must meet a number of statistical characteristics in order to be called "random". For the PRNG, the generation is based not on a truly random process, but on some function, whose value can easily be calculated if the input parameters are known, as different correlations in the output pseudo-random value become inevitable [25]. On the other hand, has been suggested that the human body could be used as the PRNG. For example, to generate a random binary sequence (RBS), it was suggested that human heartbeat be used [26]. But, even without taking into account the fact that to effectively capture this requires the use of relatively complex, expensive and uncommon equipment, this approach does not make it possible to generate a sufficient level of entropy, which was found in a study of data from publicly available heart signals recorded using portable ECG devices [27]. Similar problems associated with equipment arise when using EEGs as a PRNG [28], although in this case, there are plans to use a low-cost—by the standards of such equipment—solution that will take readings of brain activity for future use as a PRNG. It should be noted that such a solution makes it possible to generate a sequence with quite high entropy, and passes corresponding tests with more than 99% success. A more familiar method of obtaining a random sequence from biometric data is PRNG, which is based on the input of random characters from the keyboard or computer mouse movements. According to the research presented in [29], although it is possible to use the numbers generated in this way as random values, there will be different correlations generated by the human brain, the values of which are so significant that such sequences themselves can be used as an authentication factor, and authentication can be achieved with sufficient accuracy. In any case, if you need high entropy of the generated value, it is better to use real sources of a random value, for example, based on quantum processes [30]. In future, such generators will be used to generate key information for asymmetric encryption algorithms [31]. This approach can be used, in particular, when carrying out intuitive procedures that do not require additional information for authentication and cryptographic protection of information in Smart City Applications [32]. On average, the NIST test results of the developed generator exceeded 0.49 for tests such as Monobit, Frequency Within Block, Runs, Longest Run Ones In A Block, Discrete Fourier Transform, Non Overlapping Template Matching, and Serial. Such results support the validity of the selected random value generation method, and as such, we can use the keys generated by this generator for EDS.

In the next section, we present our method of EDS generation and the results of our studies. In the discussion, we compare our generator with those suggested in other works.

2. Materials and Methods

To train the program to recognize faces in images and video streams, a set of face data was initially compiled. A predefined data set, “Labeled Faces in the Wild” (LFW), [33] was used, comprising the output data of a neural network trained using 3 million facial images of random people, with 99.3% accuracy for human face detection. The following Python language libraries were also used in the program development:

1. Python Imaging Library (PIL for short), for work with bitmap graphics.
2. Datetime allows you to work with date and time.
3. PyCryptodome is an independent package of Python low-level cryptographic primitives.
4. NumPy is an open source library for the Python programming language. Functions: support for multidimensional arrays and for high-level mathematical functions designed to work with multidimensional arrays.
5. Time is a module for working with time.
6. Face_recognition is a face recognition library.
7. OpenCV is a library of computer vision and machine learning with open source code.
8. Random provides functions with which to generate random numbers and letters or make random selections of sequence elements.
9. Pickle implements a powerful algorithm for the serialization and deserialization of objects.

To remember each subject, deep metric training was applied [34]. Instead of trying to output a single mark (or even the coordinates/bounding frame of objects in the image), a vector with significant objects in the image, i.e., key points of the face, was output. For the library with the face recognition neural network, the output object vector used for quantitative face detection comprised 128 parameters in the form of real numbers.

Face recognition through in-depth metrics training included a “triplet-based” loss. A triplet consists of three unique facial images, two of which are of the same person. The neural network generates a 128-digit vector for each of the three facial images. For the two images of the same person, the neural network weights were increased to make the vector closer to the desired value using the distance metric. The neural network architecture for face recognition is based on ResNet-34 for image recognition, but with fewer layers and half of the filters, which significantly sped up the work without losing much accuracy [35].

With this in mind, a script was created to simplify the creation of our facial recognition data set. This script:

- provides access to a web-camera;
- identifies faces;

- records frames containing the face on the disk;
- performs training using a neural network;
- stores the learning results separately for each subject.

The script work scheme is shown in Figure 1.

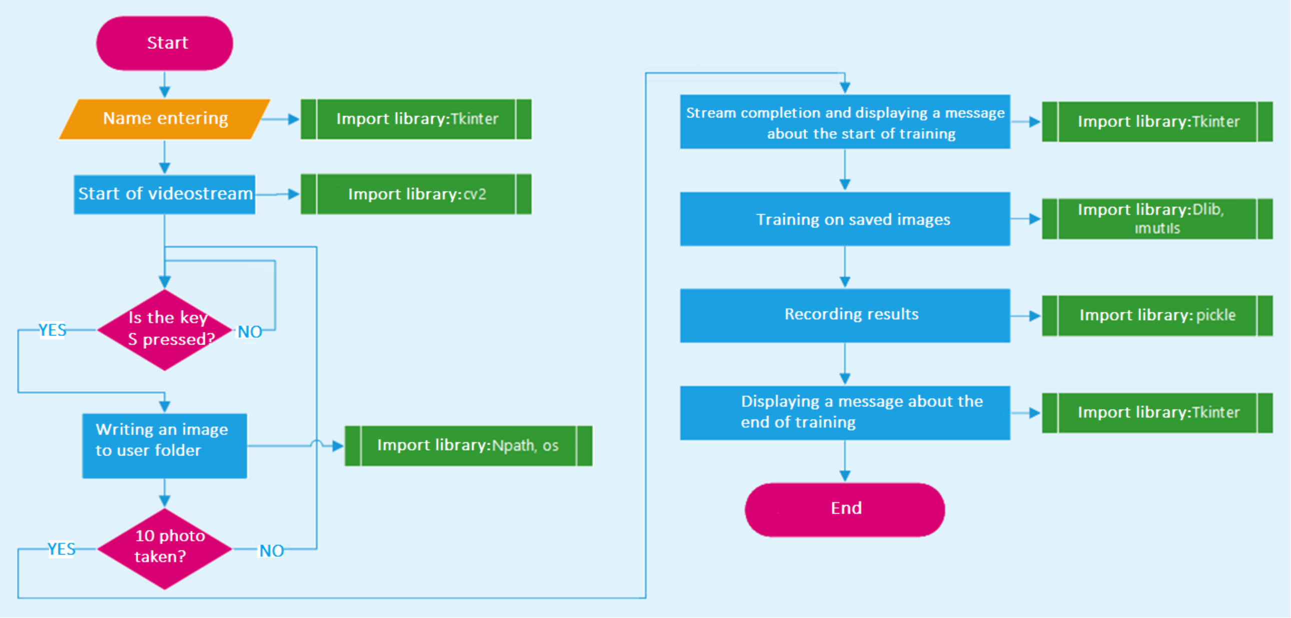

Figure 1. Work scheme of the data set creation script.

Then, the obtained dataset was used both for facial recognition in the system and to generate key information. With the help of the PyCryptodome library, standard operations with keys and cryptography were performed. In this study, the library was used to create public and private keys using the RSA algorithm (2048 bit size). By default, the library generates keys using the standard PRNG integrated in the Python language. Over the course of our research, a Python application was developed that is capable of generating a key pair for subsequent use in digital signatures based on images of the subject's face, creating a digital signature and checking it. This application generates a dataset based on the user's photo, which is subsequently used both for facial recognition in the system and to generate key information. The key information is generated using the RSA algorithm (2048-bit dimension).

To use a dataset with the face image in the developed application for generating a random value, a function was written that modernizes the standard PRNG so that it starts to use the data of the user's logged-in program additionally when generating the face image.

The PRNG function, based on the user face data, works as follows:

1. The processed facial data are retrieved and uploaded to a list;
2. Each real number in the list is raised to a square and multiplied by 1014;
3. All list elements are concatenated;
4. Random bytes are generated bitwise, a random byte is selected from the concatenated list, and if it is even, a 0 bit is written to the resulting sequence; otherwise a 1 bit is written.

For simplicity, let us call this algorithm "facial PRNG".

It should be noted that since the data used to generate the image are extracted from an image of a human face, it is not possible to generate a sufficient level of entropy even using their random mixing. In order to increase the unpredictability, the "facial PRNG" was further enhanced by the ARC4 cipher, provided in the PyCryptodome library. Thus, the amplified PRNG function works according to the algorithm presented below:

1. "Facial PRNG" generates 2048 bits;
2. ARC4 encryptor is initialized by the generated bits;
3. "Facial PRNG" generates the requested number of bytes;
4. The generated bytes are additionally encrypted with the previously initialized ARC4 cipher and returned.

The created enhanced PRNG function is transmitted to the RSA module algorithm, which uses it to generate a key pair. The key pair created in this way can be further used to sign and check various files. The algorithm for this process is presented in Figure 2.

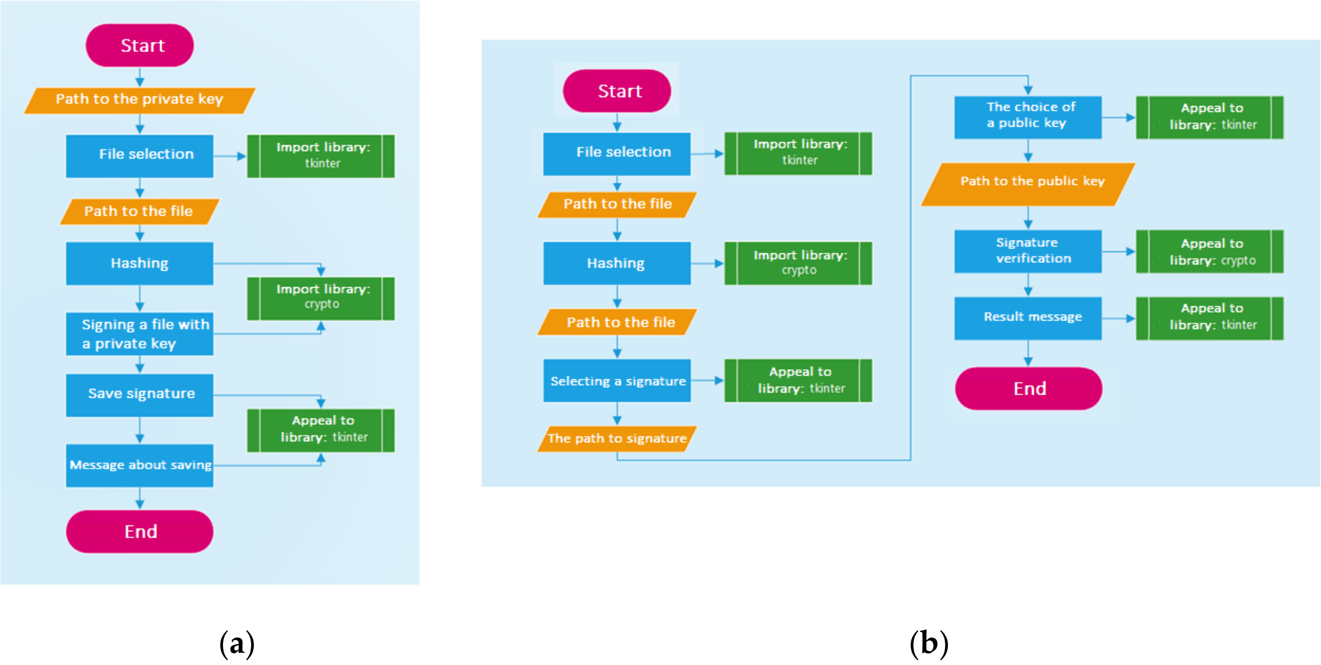

Figure 2. Use of generated keys: (**a**) Signing the file; (**b**) Checking the signature of the file.

3. Results

Testing Statistical Hypotheses

In order to use the generated keys in real conditions, the generator of random (pseudo-random) values must generate a sequence close to a truly random value, so as to ensure the key stability and security.

The package developed by the Information Technology Laboratory, which is the main research organization of the National Institute of Standards and Technology (NIST) [36], was chosen for a statistical test because it gives the most complete overview of the generator. Ten NIST tests included in the NIST-sp800-22r1a test battery were chosen. Testing was performed 1000 times on the generated 2048-byte sequence. For each test iteration, the success or failure and the obtained score were taken a reading. The results presented in Table 1 were obtained.

The accuracy of the authentication system was also tested. Testing was carried out on 148 user datasets. One dataset contained 10 user photos. On average, the system accuracy was 0.93 at 1480 reps.

Table 1. The results of NIST tests.

Test Name	Success Rate	Minimal Score	Average Score	Maximal Score
Monobit	0.992	0.0031	0.4973	1.0000
Frequency Within Block	0.989	0.0001	0.4926	0.9999
Runs	0.989	0.0018	0.4967	0.9996
Longest Run Ones In A Block	0.994	0.0003	0.4978	0.9999
Discrete Fourier Transform	0.994	0.0014	0.4838	0.9771
Nonoverlapping Template Matching	1.000	0.9558	0.9991	1.0000
Serial	0.988	0.0088	0.4933	0.9976
Approximate Entropy	0.000	0.0000	0.0000	0.0000
Random Excursion	0.007	0.0000	0.0889	0.7119
Random Excursion Variant	0.732	0.1054	0.5601	3.6442

4. Discussion

A similar study in terms of its main idea was carried out by Gerardo Iovane, Carmen Bisogni, Luigi De Maio and Michele Nappi [37], except that in their study, the face images had previously undergone different processing. The prepared numerical face characteristics were used as parameters for generating a random value. As a result of NIST testing, it was determined that the probability of the occurrence of a specific bit was 0.4894, which is almost equal to a random guess. Moreover, the frequency within a block test showed a result of 0.82165, which indicates that there was no dependence between the number 1 and 0 in the block. In our study, the monobit test showed similar results; the block test was also performed, but the result was worse by 0.33 on average.

In general, the test results in our study were comparable with those of Gerardo Iovane et al. The Runs test showed much better results, and the Longest Run test, on average, exceeded their result by about 0.17. We believe that such a result is comparable to the random number generation method proposed by Gerardo Iovane.

Hegui Zhu et al. [38] created a pseudo-random value generator based on iris and chaos images. The team processed an iris image, resulting in a binary image. The obtained images were then further processed so that the two numbers generated from the same retinal image were different. This resulted in a pseudo-random value. According to the NIST test results, this value could be considered random. The results of NIST testing presented by this team were considerably higher than those obtained by us. Monobit and the frequency within the block tests in their study showed results of 0.92328 and 0.98685, i.e., twice as good as our average results. The Runs test showed a slight improvement over our method, averaging 0.4967 vs. 0.37. The Long Run test results in our case were lower (0.83082 vs. our average of 0.49776). DFT showed a result of 0.83064 in their study and an average of 0.4838 in ours.

Based on these results, we drew the following conclusions: the generation of a random value based on the iris by Hegui Zhu and his team gives a more random result than the process proposed in this paper. However, our method showed sufficient results to pass the test, and the fact that the sampling methodology in our study is simpler proves that our method can also be applied in practice.

Another pseudo-random value generator based on biometric data is EEG-based PRNG [28]. Bhanupong Petchlert and Hiroshi Hasegawa did not directly use EEG as a source of pseudo-random values because this technology shows certain patterns in its behavior. Therefore, the results were preprocessed in the form of conversion to a given accuracy, integer conversion and bit shift; then, this sequence of numbers was converted to binary by matching 0 and 1 to even and odd numbers in the sequence, thus generating a pseudo-random value. By selecting the shift size, the researchers achieved a result that allowed their pseudo-random value to pass all previously submitted NIST test tasks; the average probability value was 99.47. This result was significantly higher than the average probability we obtained. Nevertheless, taking into account that by means of simple transformations of sequential biometric characteristics such as EEG readings, it is possible to obtain a pseudo-random value generator, we can assume that changing the method of processing the input image will allow us to increase the share of randomness of the pseudo-random value output.

V. Chandran and B. Chen [39] also implemented a random value generator based on facial images. Through a number of transformations, a binary number of 9000 bits was obtained. This number later became the basis for the generation of a pseudo-random sequence. They conducted a number of tests for randomness: Runs test, Poker test, Frequency test and DFT Spectral Analysis. Words of 32, 64, 128, 256 and 512 bits were fed to the generator input. As a result, all of these tests were passed for words of 128 bits or more. Comparing the success of our method regarding the randomization tests, as well as those described in the other works presented in this section, we can conclude that the complex transformation used by V. Chandran and B. Chen, according to the test results, surpasses our method. However, this may be due to the fact that we used a different testing methodology, which implies stricter requirements for the generated random value, since we tested on much longer words.

Also, Rudresh Dwivedi and others [40] implemented a protocol of data exchange, as the generation of encryption keys used fingerprint images. Numerical representation of the fingerprints was the basis for generating pseudo-random values. In their work, the team was able to generate public and private keys based on the numerical representation of the fingerprint. According to their research, the results of system testing showed 0.04 ms for key generation and 10.11 s for all computation operations. Such performance made it possible to create a crypto-resistant system; this system was successful in more than 90% of the tests. At present, our system requires relatively more time to generate a random number, which is due to the peculiarities of its implementation.

Further experiments with the authentication subsystem of the proposed system were conducted. In a similar work by Lanitis, Taylor and Cootes [41], a study of the accuracy of the authentication system on a dataset with 200 images was reported. They obtained the following results: 70% to 95.5% accuracy, 88.07% on average, vs. 93.04% on average for a dataset with 10 photos and 148 users in our study. Taking into account the fact that Lanitis et al. conducted the research on a system where three methods of image processing were combined, we consider our system of face recognition to be tenable. All results are presented in the Tables 2 and 3 below.

Table 2. Results for the initial face identification experiments by A. Lanitis, C. J. Taylor and T. F. Cootes.

Method	Correct Class	Correct Class. within Best 3 (%)
Shape model	70.0	82.0
Shape-free grey model	84.5	93.0
Local grey-level models	84.0	93.5
Shape + Shape-free grey model	94.0	99.0
Shape + Local grey-level models	91.5	97.0
All three methods	95.5	99.0

Table 3. Results for the initial face identification experiments by our system.

Method	Correct Class
Face recognition	93.04

Summarizing our experiments, we consider the system we created to be tenable and to have sufficient accuracy for our tasks.

5. Conclusions

Within the framework of this study, a hybrid algorithm for generating EDS keys was developed and implemented based on a graphic image of a subject's face and a stream encryption algorithm using the example of RC4. The implemented software is capable of memorizing and identifying subjects from images of their faces, creating a key pair based on these images, and implementing and verifying an EDS based on this pair using the RSA algorithm. The obtained solution, on the one hand, is comparable to analogs based on the results obtained using NIST tests. On the other hand, some of these tests (for example, Approximate Entropy) revealed the need for additional refinement and modification of the proposed solution to increase the cryptographic strength of the pseudo-random sequences used to generate keys.

According to our research, the PRNG proposed herein can be used to generate keys for different encryption algorithms. However, more successful PRNG implementations are possible, although additional tests are required to draw final conclusions about the viability of the proposed method.

A separate, additional promising direction for research is the connection and combination of the proposed method for generating a key sequence and modern verbatim algorithms based on number theory (for example, using the Catalan and Dick numbers [42]), which would make it possible to embed hidden information in images with which to generate keys. The study of both a process of embedding on the quality of the procedure for recognition and generating sequences, and of the amount of added information that would not lead to disruption of the system seems promising, and is planned by the authors.

Author Contributions: Conceptualization, A.K.; methodology, A.K. and E.K.; software, A.S. and Y.U.; validation, Y.U. and D.B.; formal analysis, A.K.; investigation, Y.U.; resources, D.B. and Y.U.; data curation, E.K. and Y.U.; writing—original draft preparation, A.S., Y.U. and E.K.; writing—review and editing, Y.U. and E.K.; visualization, Y.U.; supervision, A.K.; project administration, A.K.; funding acquisition, A.K. All authors have read and agreed to the published version of the manuscript.

Funding: The article was prepared as part of implantation of the «Leading research center (LRC) «Trusted Sensor Systems», financial support provided by Ministry of Digital Development, Communications and Mass Media of the Russian Federation and Russian Venture Company (RVC JSC) (Agreement №009/20 dated 04/10/2020).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: [https://www.kaggle.com/jessicali9530/lfw-dataset].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Konen, W.K.; Maurer, T.; von der Malsburg, C. A Fast Dynamic Link Matching Algorithm for Invariant Pattern Recognition. *Neural Netw.* **1994**, *7*, 1019–1030. [CrossRef]
2. Pollicelli, D.; Coscarella, M.; Delrieux, C. RoI Detection and Segmentation Algorithms for Marine Mammals Photo-Identification. *Ecol. Inform.* **2020**, *56*, 101038. [CrossRef]
3. Mahmood, Z.; Muhammad, N.; Bibi, N.; Ali, T. A Review on State-of-the-Art Face Recognition Approaches. *Fractals* **2017**, *25*, 1750025. [CrossRef]
4. Hu, L.; Cui, J. Digital Image Recognition Based on Fractional-Order-PCA-SVM Coupling Algorithm. *Measurement* **2019**, *145*, 150–159. [CrossRef]
5. Alsmadi, M.; Hamed, A.; Badawi, U.; Almarashdeh, I.; Salah, A.; Farag, T.; Hassan, W.; Alomari, Y.; Alsmadi, H.; Jaradat, G. Face image recognition based on partial face matching using genetic algorithm. *Sust J. Eng. Comput. Sci. (JECS)* **2017**, *18*, 51–61.
6. Yu, Z.; Liu, F.; Liao, R.; Wang, Y.; Feng, H.; Zhu, X. Improvement of Face Recognition Algorithm Based on Neural Network. In Proceedings of the 2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), Changsha, China, 10–11 February 2018; pp. 229–234. [CrossRef]
7. Haq, M.U.; Shahzad, A.; Mahmood, Z.; Shah, A.A. Boosting the Face Recognition Performance of Ensemble Based LDA for Pose, Non-Uniform Illuminations, and Low-Resolution Images. *Ksii Trans. Internet Inf. Syst.* **2019**, *13*, 3144–3164. [CrossRef]

8. Idrus, S.Z.S.; Cherrier, E.; Rosenberger, C.; Schwartzmann, J.-J. A Review on Authentication Methods. *Aust. J. Basic Appl. Sci.* **2013**, *7*, 95.
9. Zhang, X.; Xiong, Q.; Dai, Y.; Xu, X. Voice Biometric Identity Authentication System Based on Android Smart Phone. In Proceedings of the 2018 IEEE 4th International Conference on Computer and Communications (ICCC), Chengdu, China, 7–10 December 2018; pp. 1440–1444. [CrossRef]
10. Boles, A.; Rad, P. Voice Biometrics: Deep Learning-Based Voiceprint Authentication System. In Proceedings of the 2017 12th System of Systems Engineering Conference (SoSE), Waikoloa, HI, USA, 18–21 June 2017; pp. 1–6. [CrossRef]
11. Shelupanov, A.; Evsyutin, O.; Konev, A.; Kostyuchenko, E.; Kruchinin, D.; Nikiforov, D. Information security methods-Modern research directions. *Symmetry* **2019**, *11*, 150. [CrossRef]
12. Khodashinsky, I.A.; Kostyuchenko, E.Y.; Sarin, S.K.; Anfilofiev, E.A.; Bardamova, M.B. User Authentication Based on Signature Dynamics Based on Fuzzy Classifier. *Comput. Optics.* **2018**, 42. [CrossRef]
13. Sarin, K.S.; Hodashinsky, I.A. Bagged ensemble of fuzzy classifiers and feature selection for handwritten signature verification. *Comput. Opt.* **2019**, 43. [CrossRef]
14. Stauffer, M.; Fischer, A.; Riesen, K. *A Novel Graph Database for Handwritten Word Images*; Springer: Cham, Switzerland, 2016; Volume 10029. [CrossRef]
15. Kostyuchenko, E.; Gurakov, M.; Krivonosov, E.; Tomyshev, M.; Mescheryakov, R.; Hodashinskiy, I. Integration of Bayesian classifier and perceptron for problem identification on dynamics signature using a genetic algorithm for the identification threshold selection. *Lect. Notes Comput. Sci.* **2016**, *9719*, 620–627. [CrossRef]
16. Araujo, L.C.F.; Sucupira, L.H.R.; Lizarraga, M.G.; Ling, L.L.; Yabu-Uti, J.B.T. User Authentication through Typing Biometrics Features. *IEEE Trans. Signal. Process.* **2005**, *53*, 851–855. [CrossRef]
17. Yankovskaya, A.E.; Shelupanov, A.A.; Hodashinsky, I.A.; Gorbunov, I.V. Development of hybrid intelligent system of express-diagnostics for detection potential attacker. In Proceedings of the 2015 9th International Conference on Application of Information and Communication Technologies (AICT), Rostov on Don, Russia, 14–16 October 2015; pp. 183–187. [CrossRef]
18. Jorgensen, Z.; Yu, T. On Mouse Dynamics as a Behavioral Biometric for Authentication. In Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, ASIACCS '11, Hong Kong, China, 22–24 March 2011; pp. 476–482. [CrossRef]
19. Zhao, J.; Hu, Q.; Liu, G.; Ma, X.; Chen, F.; Hassan, M.M. AFA: Adversarial fingerprinting authentication for deep neural networks. *Comput. Commun.* **2020**, 150. [CrossRef]
20. Rakhmanenko, I.; Shelupanov, A.; Kostyuchenko, E. Fusion of BiLSTM and GMM-UBM Systems for Audio Spoofing Detection. *Int. J. Adv. Trends Comput. Sci. Eng.* **2019**, *6*, 1741–1746. [CrossRef]
21. Abozaid, A.; Haggag, A.; Kasban, H.; Eltokhy, M. Multimodal Biometric Scheme for Human Authentication Technique Based on Voice and Face Recognition Fusion. *Multimed Tools Appl.* **2019**, *78*, 16345–16361. [CrossRef]
22. Abbaas, F.; Serpen, G. Evaluation of Biometric User Authentication Using an Ensemble Classifier with Face and Voice Recognition. *arXiv* **2020**, arXiv:2006.00548.
23. Shinde, K.; Tharewal, S. Development of Face and Signature Fusion Technology for Biometrics Authentication. *Int. J. Emerg. Res. Manag. Technol.* **2018**, *6*, 61. [CrossRef]
24. Journal, I. Random Keypad and Face Recognition Authentication Mechanism. *Int. Res. J. Eng. Technol.* **2018**, *5*, 3.
25. Review of High-Quality Random Number Generators. Available online: https://link.springer.com/article/10.1007/s41781-019-0034-3 (accessed on 11 November 2020).
26. Pirbhulal, S.; Zhang, H.; Wu, W.; Mukhopadhyay, S.C.; Zhang, Y.-T. Heartbeats Based Biometric Random Binary Sequences Generation to Secure Wireless Body Sensor Networks. *IEEE Trans. Biomed. Eng.* **2018**, *65*, 2751–2759. [CrossRef]
27. Ortiz-Martin, L.; Picazo-Sanchez, P.; Peris-Lopez, P.; Tapiador, J. Heartbeats Do Not Make Good Pseudo-Random Number Generators: An Analysis of the Randomness of Inter-Pulse Intervals. *Entropy* **2018**, *20*, 94. [CrossRef]
28. Petchlert, B.; Hasegawa, H. Using a Low-Cost Electroencephalogram (EEG) Directly as Random Number Generator. In Proceedings of the 2014 IIAI 3rd International Conference on Advanced Applied Informatics, Kitakyushu, Japan, 31 August–4 September 2014; pp. 470–474. [CrossRef]
29. Jokar, E.; Mikaili, M. Assessment of Human Random Number Generation for Biometric Verification. *J. Med. Signals Sens.* **2012**, *2*, 82–87. [CrossRef] [PubMed]
30. OSA. Quantum Random Number Generator Based on Twin Beams. Available online: https://www.osapublishing.org/ol/abstract.cfm?uri=ol-42-5-895 (accessed on 11 November 2020).
31. Technical Review on Symmetric and Asymmetric Cryptography Algorithms-ProQuest. Available online: https://search.proquest.com/openview/94f3a444d3f907bdb0adfc7ed6ba770c/1?pq-origsite=gscholar&cbl=1606379 (accessed on 11 November 2020).
32. Saračević, M.; Adamović, S.; Macek, N.; Elhoseny, M.; Sarhan, S. Cryptographic Keys Exchange Model for Smart City Applications. In *IET Intelligent Transport Systems*; IET: London, UK, 2020; Volume 14, pp. 1456–1464. [CrossRef]
33. Labelled Faces in the Wild (LFW) Dataset. Available online: https://kaggle.com/jessicali9530/lfw-dataset (accessed on 11 November 2020).
34. Pak, M.; Kim, S. A Review of Deep Learning in Image Recognition. In Proceedings of the 2017 4th International Conference on Computer Applications and Information Processing Technology (CAIPT), Kuta Bali, Indonesia, 8–10 August 2017; pp. 1–3. [CrossRef]

35. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385.
36. Pareschi, F.; Rovatti, R.; Setti, G. Second-Level NIST Randomness Tests for Improving Test Reliability. In Proceedings of the 2007 IEEE International Symposium on Circuits and Systems, New Orleans, LA, USA, 27–30 May 2007; pp. 1437–1440. [CrossRef]
37. Iovane, G.; Bisogni, C.; Maio, L.D.; Nappi, M. An Encryption Approach Using Information Fusion Techniques Involving Prime Numbers and Face Biometrics. *IEEE Trans. Sustain. Comput.* **2020**, *5*, 260–267. [CrossRef]
38. Zhu, H.; Zhao, C.; Zhang, X.; Yang, L. A Novel Iris and Chaos-Based Random Number Generator. *Comput. Secur.* **2013**, *36*, 40–48. [CrossRef]
39. Chandran, V.; Chen, B. Simultaneous Biometric Verification and Random Number Generation. In Proceedings of the 5th Workshop on Internet, Telecommunications and Signal Processing, Hobart, Australia, 11–13 December 2006.
40. Dwivedi, R.; Dey, S.; Sharma, M.A.; Goel, A. A Fingerprint Based Crypto-Biometric System for Secure Communication. *J. Ambient Intell Hum. Comput.* **2020**, *11*, 1495–1509. [CrossRef]
41. Lanitis, A.; Taylor, C.; Cootes, T. Automatic Face Identification System Using Flexible Appearance Models. *Image Vis. Comput.* **1995**, *13*, 393–401. [CrossRef]
42. Saracevic, M.; Adamovic, S.; Miskovic, V.; Macek, N.; Sarac, M. A novel approach to steganography based on the properties of Catalan numbers and Dyck words. In *Future Generation Computer Systems*; Elsevier: Amsterdam, The Netherlands, 2019; Volume 100, pp. 186–197. [CrossRef]