information

Article

Adversarial Attacks Impact on the Neural Network Performance and Visual Perception of Data under Attack

Yakov Usoltsev, Balzhit Lodonova, Alexander Shelupanov, Anton Konev and Evgeny Kostyuchenko *

Faculty of Security, Tomsk State University of Control Systems and Radioelectronics, 40 Lenin Avenue, 634050 Tomsk, Russia; yakovmen62@mail.ru (Y.U.); office@tusur.ru (B.L.); saa@tusur.ru (A.S.); kaa@fb.tusur.ru (A.K.)

* Correspondence: key@fb.tusur.ru; Tel.: +7-(3822)-41-34-26

Abstract: Machine learning algorithms based on neural networks are vulnerable to adversarial attacks. The use of attacks against authentication systems greatly reduces the accuracy of such a system, despite the complexity of generating a competitive example. As part of this study, a white-box adversarial attack on an authentication system was carried out. The basis of the authentication system is a neural network perceptron, trained on a dataset of frequency signatures of sign. For an attack on an atypical dataset, the following results were obtained: with an attack intensity of 25%, the authentication system availability decreases to 50% for a particular user, and with a further increase in the attack intensity, the accuracy decreases to 5%.

Keywords: digital signature; python; neural networks; biometric authentication; adversarial attack; fast gradient method; square method; artificial intelligence; accuracy; statistical errors; F-score

Citation: Usoltsev, Y.; Lodonova, B.; Shelupanov, A.; Konev, A.; Kostyuchenko, E. Adversarial Attacks Impact on the Neural Network Performance and Visual Perception of Data under Attack. *Information* **2022**, *13*, 77. https://doi.org/10.3390/info13020077

Academic Editors: Nuno Cavalheiro Marques and Bruno Silva

Received: 9 December 2021
Accepted: 31 January 2022
Published: 5 February 2022

1. Introduction

Nowadays, the use of biometric authentication systems is considered a promising avenue of authentication system development, and many of them are based on neural networks [1–4]. Fingerprint-based authentication and face recognition methods are progressively replacing PIN codes and passwords [5–9], despite the vulnerability of these authentication methods. Attacks based on opposing examples show excellent results: effective attacks were carried out on voice authentication systems, image recognition systems, behavioral biometric authentication systems, etc. At the same time, the methods of countering attacks suggested in some works do not significantly affect their effectiveness.

In addition to authentication systems, neural networks are widely used in recognition systems. For these systems, our experiments can be useful in terms of confirming the classifiers' completeness. One example is in classification systems based on speech characteristics [10–15]. Nevertheless, research in the field of adversarial attacks on such systems [16,17] proves that even the use of biometric metrics such as voice is not a guarantee of protection from adversarial attacks. Adversarial attacks can negatively affect the operation of almost any system [17–21], but we assume that certain characteristics of datasets may have some influence on the effectiveness of an attack [22,23].

Authentication by signature is a form of biometric authentication. For such a system, vulnerability to adversarial attacks is an unwanted characteristic: such an attack reduces the availability of the system for a particular user, not to mention the potential of access to the system by an attacker. This article will present the results of an adversarial attack on the biometric authentication system by signature of sign. Two types of attack are used in this paper: the white-box attack and the black-box attack.

Both attacks caused a decrease in the system accuracy.

Section 2 describes the architecture of the neural network used, as well as the features of the dataset and attacks. Section 3 contains the results of the attacks and a description of their impact on the authentication system. Section 4 is devoted to comparing the

results obtained with the results of similar studies. Section 5 contains a brief conclusion in accordance with the results of the study.

2. Materials and Methods

2.1. Study System

The system comprises a Python script that reads user signature data, trains a neural network based on them, conducts an adversarial attack using a fast gradient method, and reads and processes accuracy values and statistical errors for the entire system and for the attacked user. An adversarial attack is performed using the Adversarial Robustness Toolbox library (for more information, see Section 2.3 Adversarial Attack).

The system includes the following:

- Uploading data;
- Splitting signature datasets into two parts;
- Creating a neural network with specified parameters;
- Neural network training and testing;
- Choosing the decision threshold;
- Obtaining and correct representation of various statistical errors.

By default, a neural network is created that takes 144 parameters in the input layer and has 30 neurons in the hidden layer; output is a vector of results that contains a number of elements equal to the number of users registered in the system as legitimate users and shows which of them the entered signature belongs to. The categorical cross-entropy function is chosen as the loss function.

2.2. Dataset

The initial data are signature datasets. The signature is presented in the format of 144 parameters (576 parameters for the visualization module), giving full information about the writing dynamics of the signature. At the time of the study, there were 2445 parameterized signatures of 22 users.

During the study, the signature was taken as a set of coordinates (X, Y, and Z), the force of pressing on the graphic tablet, and the angles of inclination of the pen relative to the tablet plane every 5 milliseconds during signing. Since the points number of a user's sign may vary from 700 to 2500, training a neural network requires converting a large number of parameters to a fixed number of parameters. For this purpose, a fast Fourier transform [24] was used on the initial data, with further retention of information using the first 8 or 16 frequencies. This processing allows you to reduce thousands of points to 8 or 16 complex values with the ability to restore the original sequence.

The first 8 frequencies were used for a black box adversarial attack. The first 16 frequencies were used for the part of the study with visualization of a white-box attack results.

2.3. Adversarial Attack

The white-box attack was carried out using the Adversarial Robustness Toolbox python library [25], and the FGSM method [26]. This method implies the generation of adversarial examples according to the following formula:

$$X^{adv} = X + \varepsilon \cdot \text{sign}(\nabla \times J(X, y_{true})) \quad (1)$$

As can be seen from (1), in addition to the attacked data and the value of ε, it is also necessary to submit a white-box classifier model as an input to perform a gradient descent operation.

The black-box attack was carried out using the Square Attack method [27] and the Adversarial Robustness Toolbox. According to the developers of the method, it is based on random search, which is a well-known iterative technique in optimization. The main idea of the algorithm is to sample a random update δ at each iteration, and to add this update

to the current iterate $\hat{x}$ if it improves the objective function. The input data includes the trained classifier as a black box, the data to be attacked, the number iterations available, and the value of ε. The output of the algorithm is the attacked data.

Since this method was implemented to the image datasets, the input layer of the neural network was changed: 144 initial parameters were presented as a 12×12 matrix submitted to the network input as 12×12 black-and-white images. Then, a Reshape transform layer was added. It transformed the input from the 12×12 matrix into a flat vector with 144 values. As a result of the attack, 12×12 matrices were generated.

The attacks were carried out on the data of only one user, with a change in the value of the attack intensity from 1 to 99%. To ensure reproducibility of the results, the experiment was carried out a hundred times and the minimum, average, and median values were calculated from the data obtained.

These attacks were chosen because the articles describe their high efficiency when carried out by developers. In addition, the results obtained are not related to the network architecture. These attacks can be implemented on our data with minimal changes to the original dataset and neural network. Comparing the effectiveness of the black-box and white-box attacks will allow us to forecast the level of the system vulnerability and determine whether an attacker needs access to a trained model for a successful attack.

2.4. Metrics

The decrease in the system accuracy for a particular user and for the entire system was chosen as the main metric of the attack effectiveness. Accuracy is calculated as the ratio of successful authentications to the total number of authentications.

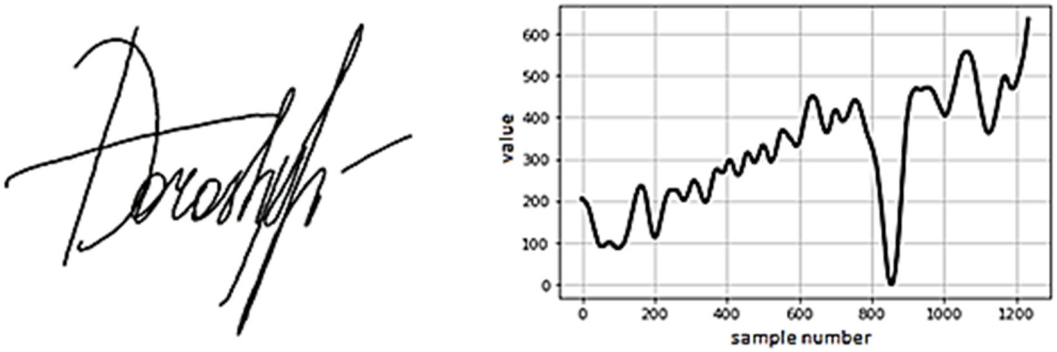
Type I and Type II errors rates for the entire system and the attacked user in particular were chosen as the second metric.

The F-score of the attacked user identification was used as an additional metric.

3. Results

3.1. White-Box Adversarial Attack Results

First, the adversarial attack impact on the visual representation of the signature was checked. This study demonstrates the attack impact only on the flat form of the signature, based on the spatial coordinates of the points. As an example of a signature and for clarity of the representation, we also present a graph of the x coordinate dependence on the reference number in Figure 1.

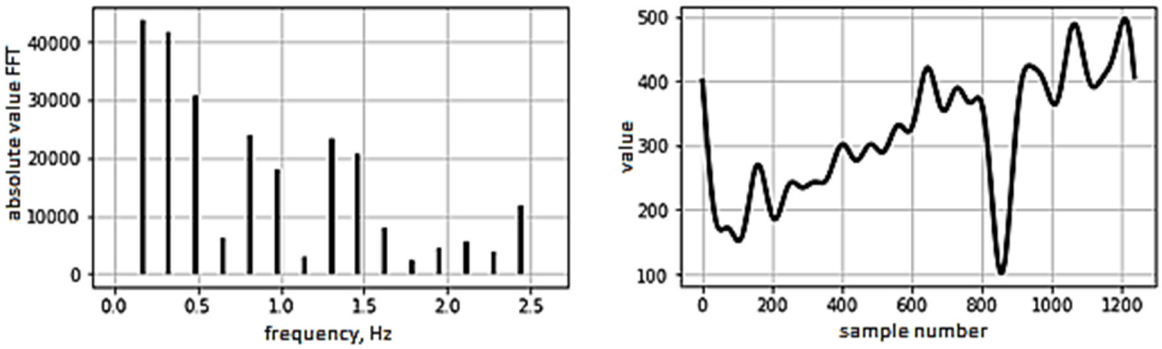


(**a**) (**b**)

Figure 1. Example of the original signature: (**a**) the original signature; (**b**) dynamics of the x coordinate change.

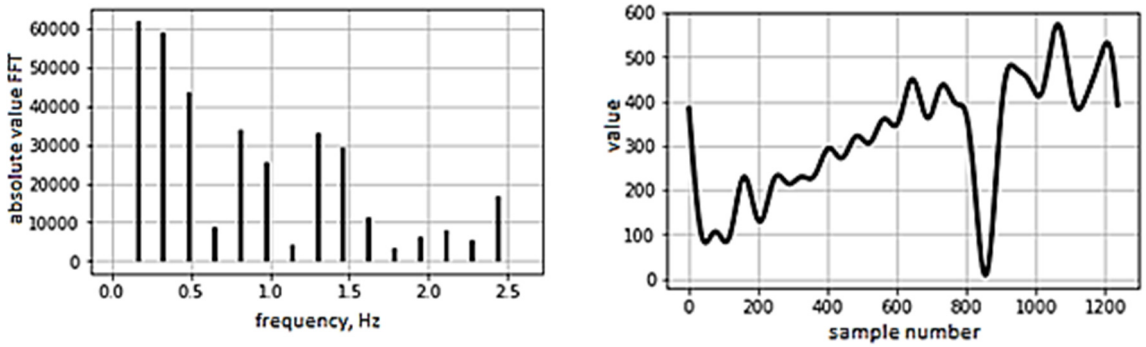
The original signature was processed through a fast Fourier transform. The network was trained on such processed signatures.

Next, using the ART library, an attack was carried out through the neural network. The attack was carried out using a gradient method, which generated a new dataset that disrupted the neural network. Graph of the x coordinate dependence on the reference number of adversarial example is present in Figure 2. Original characteristics are present in Figure 3.



(**a**) (**b**)

Figure 2. Example of a generated attack: (**a**) generated frequency characteristic; (**b**) generated dynamics of the x coordinate change.



(**a**) (**b**)

Figure 3. Example of a processed signature: (**a**) frequency characteristics of the normalized signature; (**b**) dynamics of the x coordinate change.

The restored signature based on the data processed in this way looks like this.

The reconstructed signature based on the generated data is almost identical to the one shown in Figure 4.

Figure 4. Restored form of the processed signature.

As a result, despite the fact that the adversarial attack implementing module was not initially optimized for a dataset [25] similar to ours, the data obtained as a result of the attack are visually almost indistinguishable from the original unaltered data, but at the same time have a significant impact on the network. The results of this attack are shown below in Figure 5.

Figure 5. Restored form of the generated signature.

3.2. Black-Box Adversarial Attack Results

Black-box adversarial attack was carried out using the square method [27]. Initially, the chosen attack was implemented on image datasets; therefore, the logic of the neural network was changed. The 12 × 12 matrices created as a result of the attack were used as input data for the neural network. After that, the network performance indicators were recorded.

At the same time, statistical Type II errors rates remained unchanged but Type I errors rates increased to 10%, this is due to the number of legitimate users of the system, which is equal to 13 users.

Predictably enough, the Type I error for the attacked user increased to 100%.

The results of the experiments are presented in Table 1.

Table 1. Results of an adversarial attack on an atypical dataset, AVG.

Rates	FGSM Attack ε = 0.01	Square Attack ε = 0.01	FGSM Attack ε = 0.2	Square Attack ε = 0.2	FGSM Attack ε = 0.4	Square Attack ε = 0.4
System accuracy	0.97	0.92	0.915	0.87	0.884	0.87
User accuracy	0.99	0.72	0.63	0.005	0.09	0
Type I errors for the entire system	0.018	0.016	0.04	0.1	0.075	0.1
Type I errors for the attacked user	0.01	0.238	0.38	0.1	0.92	0.1
F-score	0.998	0.736	0.597	0.003	0	0

4. Results and Discussion

4.1. White-Box Adversarial Attack Results Discussion

The graphs show that an attack on the data of a particular user reduces the system accuracy by an average of 13%, and the Type I error rate increases by more than 7% (see Figures 6–9). At the same time, an adversarial attack on the data of one user does not lead to an increase in the Type II error rate; consequently, a potential attacker or hacker will not be able to use this attack to gain unauthorized access. However, the Type I error rate for the attacked user increases to 100% as a result of white-box adversarial attack on more than 45% of test cases. At the same time, the user accuracy is reduced to 0%. To measure the system accuracy, the F-score is used. The value of the F-score also decreases with an increase in ε and reaches 0 when ε is greater than 0.5 (graph is present in Figure 10).

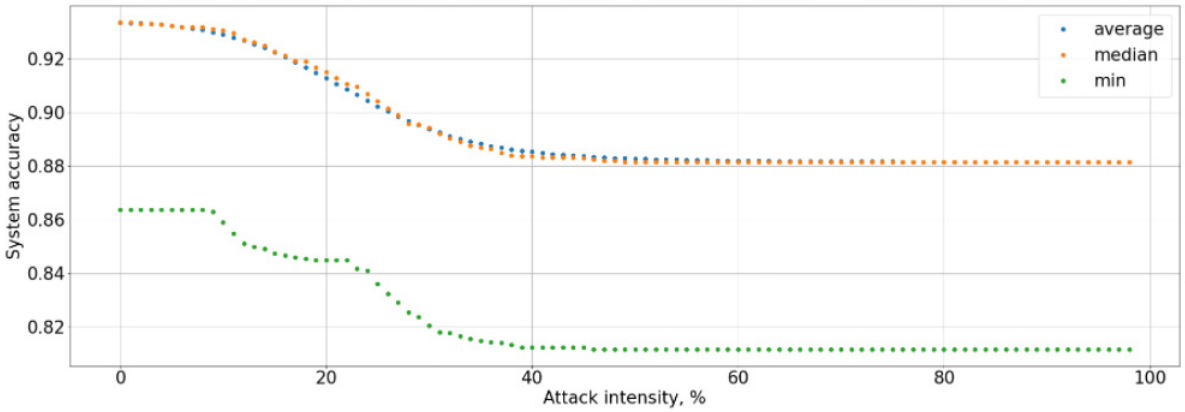


Figure 6. Dependence of the system accuracy on the attack intensity.

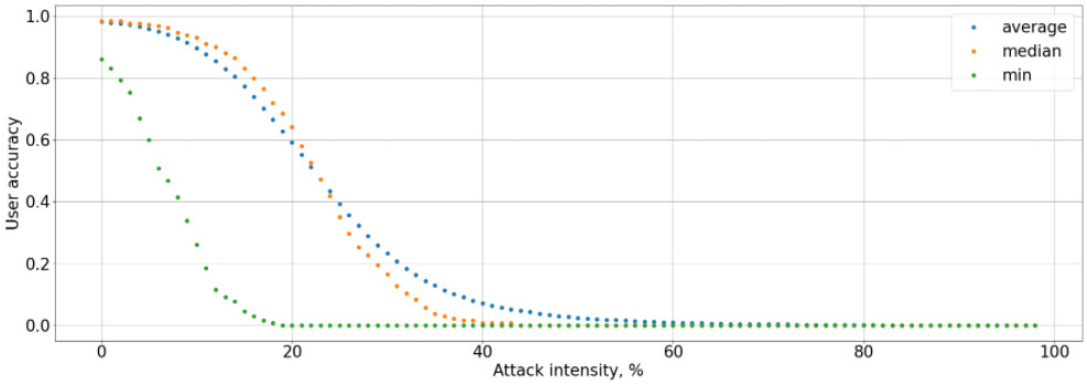
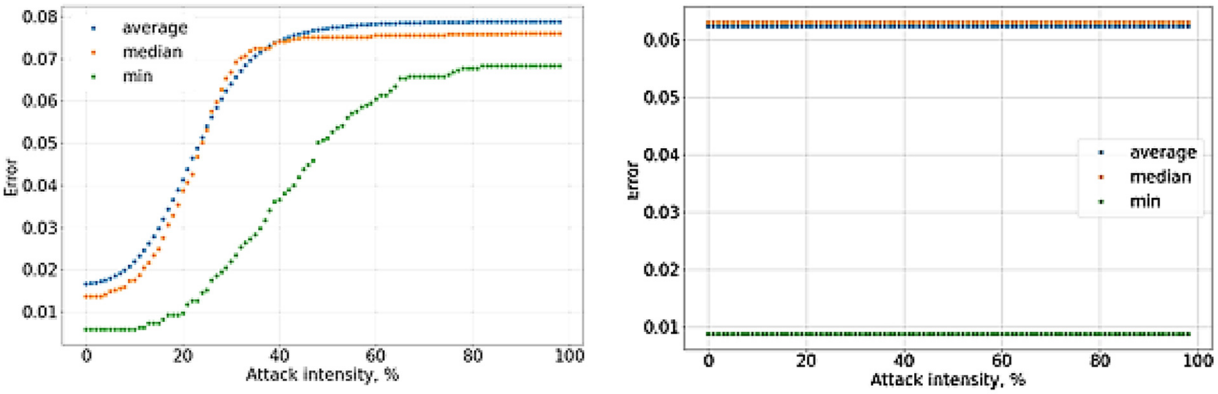


Figure 7. Dependence of the user accuracy on the attack intensity.



(**a**) (**b**)

Figure 8. Dependence of the Type I and Type II error rates for the entire system on the attack intensity: (**a**) Type I error rate; (**b**) Type II error rate.

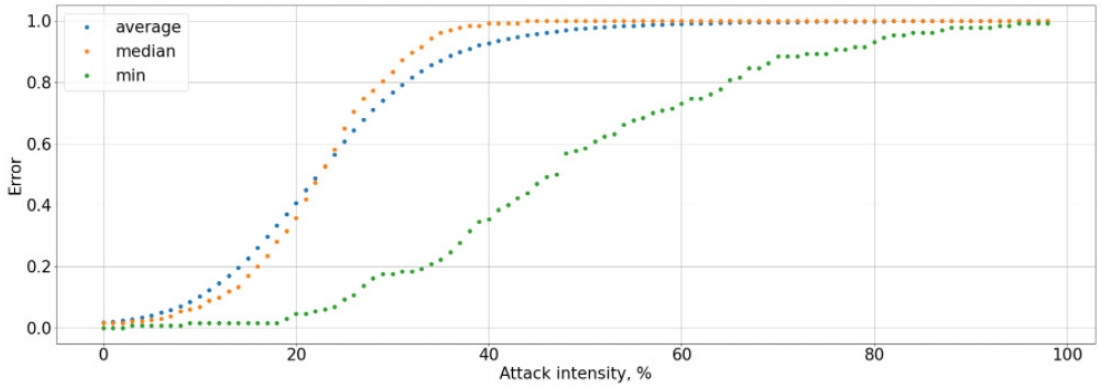


Figure 9. Dependence of the Type I error rate for the attacked user on the attack intensity.

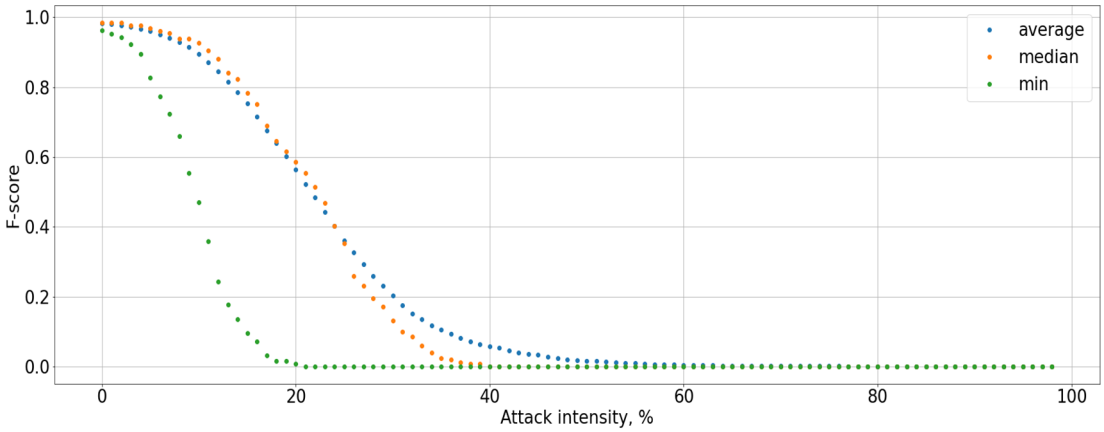


Figure 10. Dependence of the F-score on the attack intensity.

These results are relevant only for this system, since the neural network architecture does not imply additional training based on new input data.

Nevertheless, a similar attack was carried out [16] in the study of adversarial attack countering methods for automatic speaker verification systems. In this study, the gradient method attack was also carried out using the white-box method. Its effectiveness was compared with the PGD (projected gradient descent) attack. Moreover, in this paper, the authors concluded that the attack-countering methods they considered did not reduce the effectiveness of the attack. The error rates varied within 50% for all three considered models of attack countering, with $\varepsilon = 5$ for the FGSM method.

In paper [28], researchers used the FGSM attack on various datasets of handwritten digit images. The results obtained are similar to ours: the accuracy is reduced to comparable values and the decline rate is also similar.

White-box attack using adversarial examples has also been considered in study [29]. The paper considered an attack on the text classifier. Adversarial examples were generated using white-box methodology. The attack algorithm rearranged the components of the text without significantly changing its meaning. Experiments have shown the success of the attack as "tricking the classifier for more than 90% of the instances". Such high attack success rates are also comparable to our results.

4.2. Black-Box Adversarial Attack Results Discussion

The black-box attack is much more effective: the maximum perturbation value that the attacker can introduce is 0.03, the system accuracy was less than 90%, and for the attacked user it was up to 0%, that present in Figures 11 and 12. This affected an increase in Type I error for the attacked user to 100% and Type I errors for the entire system up to 10%, as well

as a decrease in the F-score to 0, that graph of Figure 13 is present. It means that the user will not be able to access the system at all. Type I error for the attacked user present in Figure 14, Type I and Type II errors rate for the entire system are present in Figures 15 and 16.

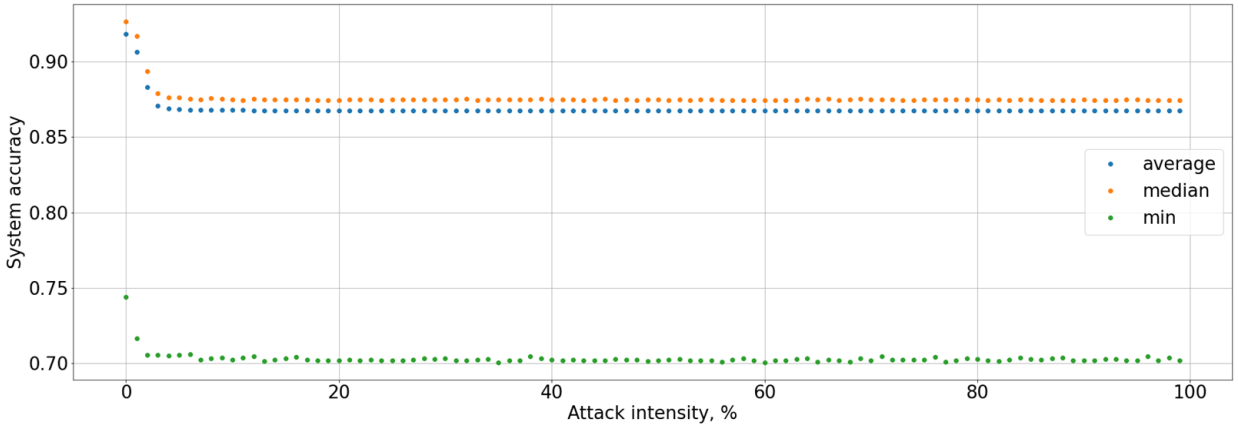


Figure 11. System accuracy during the Square attack.

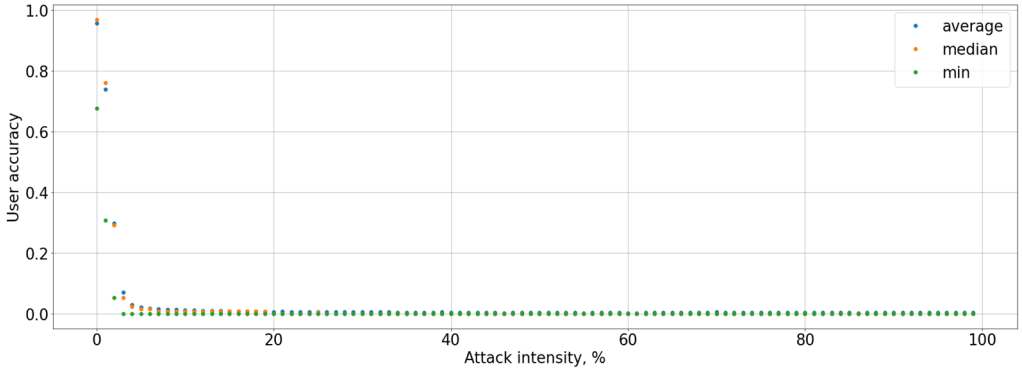


Figure 12. User accuracy during the Square attack.

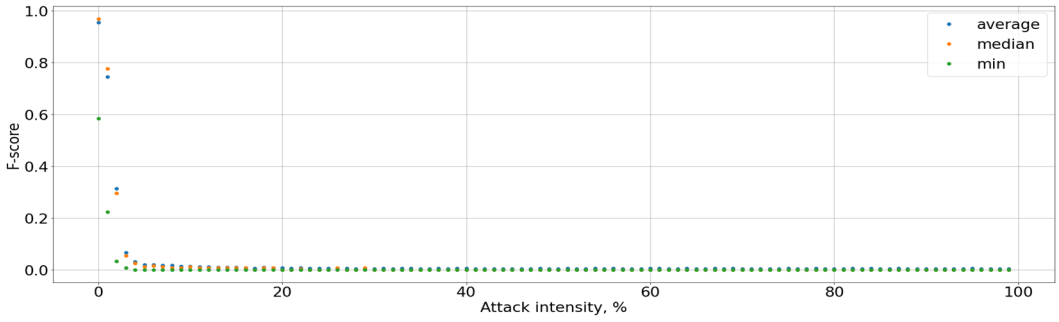


Figure 13. F-score for the attacked user.

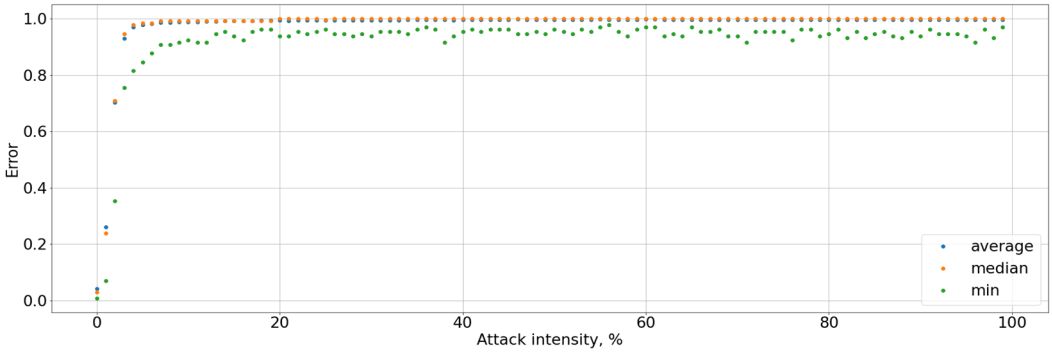


Figure 14. Type I error rate for the attacked user.

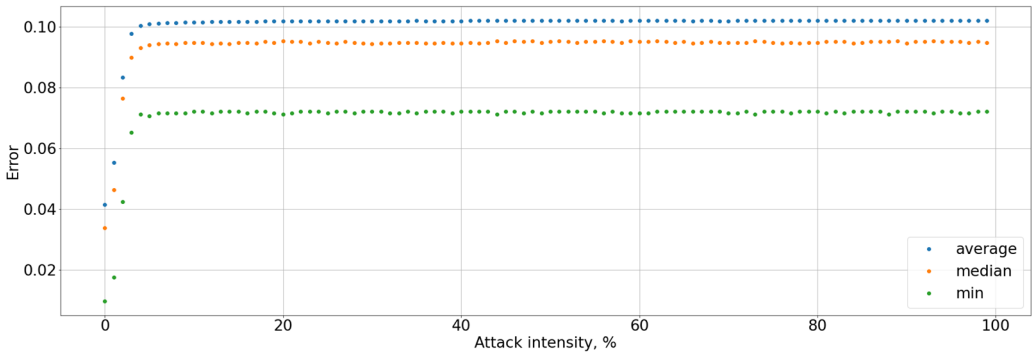


Figure 15. Type I error rate for the entire system.

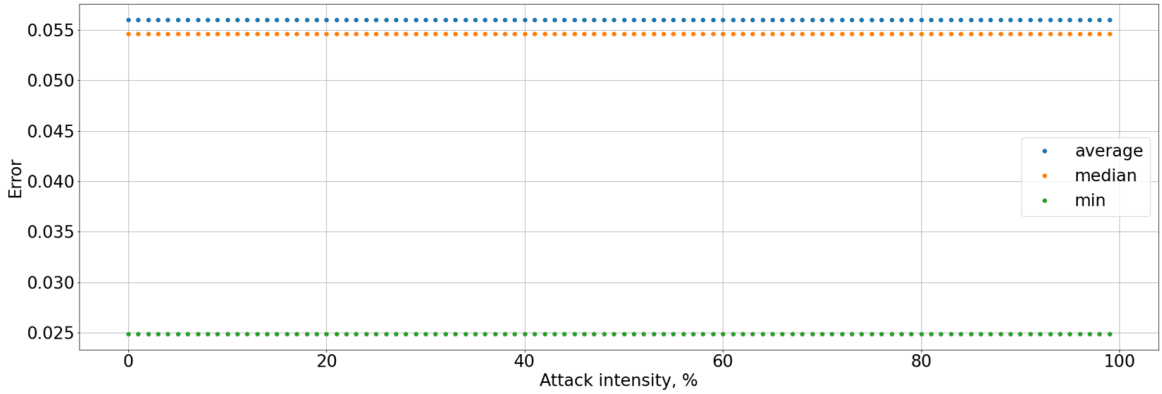


Figure 16. Type II error rate for the entire system.

In a similar study [28], the attack effectiveness reaches error rates of 30–50% with various combinations of protection methods.

Yuekai Zhang and colleagues investigated an adversarial type attack on four types of neural network models with various variations of the FGSM method [17]. In their case, the attack efficiency reached 100% when generating examples by a network model different from the one being attacked, regardless of the modification of the FGSM method used. These results correlate with the results we have obtained.

Study [30] demonstrates the effectiveness of a black-box attack by the Square attack method in comparison with that by WideResNet-101. The authors carried out an attack on an image dataset, and tried to smooth out adversarial noise by processing with different coding algorithms, including those based on transformations similar to the FFT. In this paper, we are interested in the results obtained without the use of counteraction methods.

With a change of ε in the ranges from 0/255 to 24/255, the accuracy of image classification decreased by about 50%.

Black-box attack using adversarial examples has also been considered in study [31]. The attack was carried out on commercial classification systems from Amazon and Google. In this work, the researchers did not have any access to the attacked model at all. To generate adversarial examples, they trained their own neural network. A black-box attack was applied to this network. The generated data were sent for recognition to a commercial classifier. The classification error in this case ranged from 84.5% to 97.72%. Such indications are consistent with those we received, especially considering that in our work, when generating adversarial examples, we used the attacked model itself for generation.

4.3. General Findings

Depending on the capabilities, an attacker can use one or both of the attacks considered in this study to violate the availability of the system for the attacked user. Since the attacked data visually differ slightly from the non-attacked data and the existing differences can be attributed to adversarial noise, human factor, and other biometric systems deficiencies, the fact of the attack may remain unobvious for some time, and the attacker undetected.

5. Conclusions

As part of the study, two attacks on the biometric authentication system based on a neural network were carried out. The results of the attacks showed that the generated examples are visually indistinguishable from the original ones. Nevertheless, the attacks studied showed their high effectiveness from the attacker's point of view, despite the atypical dataset and the neural network to which they were applied. The white-box attack reduced the authentication system accuracy to 88% for the entire system and to 0% at 50% of the attack intensity, which corresponds to a complete failure to identify one specific user. The black-box attack is much more effective: even with a 4% attack intensity, the system accuracy has decreased to not less than 90% and the user accuracy to 0%, which also corresponds to a complete failure to identify one particular user. Thus, we can summarize the following: despite the architecture features and the dataset unavailability for an adversarial attack, the suggested attacks turned out to be effective for violating the availability of the system. The chosen attack methods were originally created to attack image datasets; however, the dataset of frequency and time characteristics of signatures, which is not similar to images, turned out to be vulnerable to them. In general, it can be concluded that the adaptation of an adversarial attack for a given neural network architecture is possible, and it is likely that the adapted attack will be carried out successfully.

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

Funding: This research was funded by the Ministry of Science and Higher Education of Russia, Government Order for 2020–2022, project no. FEWM-2020-0037 (TUSUR).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Mahmood, Z.; Muhammad, N.; Bibi, N.; Ali, T. A Review on State-of-the-Art Face Recognition Approaches. *Fractals* **2017**, *25*, 1750025. [CrossRef]
2. Idrus, S.Z.S.; Cherrier, E.; Rosenberger, C.; Schwartzmann, J.-J. A Review on Authentication Methods. *Aust. J. Basic Appl. Sci.* **2013**, *7*, 95.
3. Shukla, S.; Helonde, A.; Raut, S.; Salode, S.; Zade, J. Random Keypad and Face Recognition Authentication Mechanism. *Int. Res. J. Eng. Technol. (IRJET)* **2018**, *5*, 3.
4. 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]
5. 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*, 488–497. [CrossRef]
6. Shinde, K.; Tharewal, S. Development of Face and Signature Fusion Technology for Biometrics Authentication. *Int. J. Emerg. Res. Manag. Technol.* **2018**, *6*, 61. [CrossRef]
7. 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]
8. 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]
9. Lanitis, A.; Taylor, C.; Cootes, T. Automatic Face Identification System Using Flexible Appearance Models. *Image Vis. Comput.* **1995**, *13*, 393–401. [CrossRef]
10. Rakhmanenko, I.A.; Shelupanov, A.A.; Kostyuchenko, E.Y. Automatic text-independent speaker verification using convolutional deep belief network. *Comput. Opt.* **2020**, *44*, 596–605. [CrossRef]
11. Chandankhede, P.H.; Titarmare, A.S.; Chauhvan, S. Voice Recognition Based Security System Using Convolutional Neural Network. In Proceedings of the 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 19–20 February 2021; pp. 738–743.
12. 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]
13. 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]
14. 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]
15. Chen, G.; Chenb, S.; Fan, L.; Du, X.; Zhao, Z.; Song, F.; Liu, Y. Who Is Real Bob? Adversarial Attacks on Speaker Recognition Systems. In Proceedings of the 2021 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 24–27 May 2021; pp. 694–711.
16. Liu, S.; Wu, H.; Lee, H.-Y.; Meng, H. Adversarial Attacks on Spoofing Countermeasures of Automatic Speaker Verification. In Proceedings of the 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Sentosa, Singapore, 14–18 December 2019; pp. 312–319. [CrossRef]
17. Zhang, Y.; Jiang, Z.; Villalba, J.; Dehak, N. Black-Box Attacks on Spoofing Countermeasures Using Transferability of Adversarial Examples. In Proceedings of the Interspeech 2020, ISCA, Online, 25 October 2020; pp. 4238–4242.
18. Du, C.; Zhang, L. Adversarial Attack for SAR Target Recognition Based on UNet-Generative Adversarial Network. *Remote Sens.* **2021**, *13*, 4358. [CrossRef]
19. Combey, T.; Loison, A.; Faucher, M.; Hajri, H. Probabilistic Jacobian-Based Saliency Maps Attacks. *Mach. Learn. Knowl. Extr.* **2020**, *2*, 558–578. [CrossRef]
20. Marcus Tan, Y.X.; Iacovazzi, A.; Homoliak, I.; Elovici, Y.; Binder, A. Adversarial Attacks on Remote User Authentication Using Behavioural Mouse Dynamics. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–10.
21. Huang, E.; Di Troia, F.; Stamp, M. Evaluating Deep Learning Models and Adversarial Attacks on Accelerometer-Based Gesture Authentication. *arXiv* **2021**, arXiv:2110.14597.
22. Hendrycks, D.; Zhao, K.; Basart, S.; Steinhardt, J.; Song, D. Natural Adversarial Examples. 2021, pp. 15262–15271. Available online: https://openaccess.thecvf.com/content/CVPR2021/html/Hendrycks_Natural_Adversarial_Examples_CVPR_2021_paper.html (accessed on 13 October 2021).
23. Pestana, C.; Liu, W.; Glance, D.; Mian, A. Defense-Friendly Images in Adversarial Attacks: Dataset and Metrics for Perturbation Difficulty. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikola, HI, USA, 5–9 January 2021; pp. 556–565.
24. Mohanan, A.V.; Bonamy, C.; Augier, P. FluidFFT: Common API (C++ and Python) for Fast Fourier Transform HPC libraries. *J. Open Res. Softw.* **2019**, *7*, 10. [CrossRef]
25. Nicolae, M.-I.; Sinn, M.; Minh, T.N.; Rawat, A.; Wistuba, M.; Zantedeschi, V.; Molloy, J.M.; Edwards, B. Adversarial Robustness Toolbox v0.2.2. 2018. Available online: https://openreview.net/forum?id=LjClBNOADBzB (accessed on 29 November 2021).

26. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and Harnessing Adversarial Examples. *arXiv* **2015**, arXiv:1412.6572. Available online: http://arxiv.org/abs/1412.6572 (accessed on 29 November 2021).
27. Andriushchenko, M.; Croce, F.; Flammarion, N.; Hein, M. Square Attack: A Query-Efficient Black-Box Adversarial Attack via Random Search. *arXiv* **2020**, arXiv:1912.00049. Available online: http://arxiv.org/abs/1912.00049 (accessed on 29 November 2021).
28. Ross, A.S.; Doshi-Velez, F. Improving the Adversarial Robustness and Interpretability of Deep Neural Networks by Regularizing their Input Gradients. *arXiv* **2017**, arXiv:1711.09404. Available online: http://arxiv.org/abs/1711.09404 (accessed on 3 January 2021).
29. Ebrahimi, J.; Rao, A.; Lowd, D.; Dou, D. HotFlip: White-Box Adversarial Examples for Text Classification. *arXiv* **2018**, arXiv:1712.06751. Available online: http://arxiv.org/abs/1712.06751 (accessed on 17 January 2021).
30. Cha, S.; Ko, N.; Yoo, Y.; Moon, T. NCIS: Neural Contextual Iterative Smoothing for Purifying Adversarial Perturbations. *arXiv* **2021**, arXiv:2106.11644. Available online: http://arxiv.org/abs/2106.11644 (accessed on 3 January 2021).
31. Papernot, N.; McDaniel, P.; Goodfellow, I. Transferability in Machine Learning: From Phenomena to Black-Box Attacks Using Adversarial Samples. *arXiv* **2016**, arXiv:1605.07277. Available online: http://arxiv.org/abs/1605.07277 (accessed on 17 January 2021).