Pattern Recognition Letters 131 (2020) 79–84

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Attacking NIST biometric image software using nonlinear optimization

Sunny Raj[a], Jodh S. Pannu[a], Steven L. Fernandes[a,*], Arvind Ramanathan[b,c], Laura L. Pullum[d], Sumit K. Jha[a]

[a] *Department of Computer Science, University of Central Florida, Orlando, FL 32816, USA*
[b] *Data Science and Learning Division, Argonne National Laboratory, Lemont, IL 60439, USA*
[c] *Consortium for Advanced Science and Engineering, University of Chicago, Chicago, IL 60637, USA*
[d] *Computational Data Analytics Group, Oak Ridge National Laboratory, Oak Ridge, TN 37830, USA*

ARTICLE INFO

Article history:
Received 25 September 2019
Revised 29 November 2019
Accepted 4 December 2019
Available online 5 December 2019

Keywords:
NIST biometric image software
MINDTCT
BOZORTH3
NLopt

ABSTRACT

Automated fingerprint identification systems are deployed by law enforcement agencies all over the world for authentication. In the US, the NIST biometric image software (NBIS) is used by the Department of Homeland Security and the Federal Bureau of Investigation for fingerprint matching. NBIS uses MINDTCT as the minutia extractor and BOZORTH3 as the fingerprint matcher. We use nonlinear optimization to attack the BOZORTH3 fingerprint matching system. We use FVC2002, ATVS and CASIA datasets to validate the performance of our attack. We show that the average match score of attack fingerprints is 111.2 for FVC2002, 97.17 for ATVS and 111.07 for the CASIA dataset. We show that for all three datasets, changing only 14 minutia features allows us to attack the BOZORTH3 fingerprint matcher with more than 75% probability of successful attack.

1. Introduction

Biometrics are used to establish the identity of individuals based upon behavioral and intrinsic physical traits [14]. Among different types of biometrics, fingerprints are extensively used by law enforcement agencies. Research shows that within seven months of fetus development, human fingertips are completely formed along with their unique ridge configurations. Fingerprints are unique and do not change over the lifespan of an individual; hence, they can be used for identification purposes [4,21,31]. However, their popularity has also attracted a wide variety of attacks on fingerprint recognition systems [26]. Four categories of vulnerabilities of a generic biometric system have been identified by Jain et al. [13]. These vulnerabilities include intrinsic failure, administrative loopholes, nonsecure infrastructure, and biometric overtness. Intrinsic failures are caused by limitations of biometric systems. An unauthorized person can get enrolled as an authenticated person by exploiting administrative loopholes. An attacker can access latent fingerprints of an authorized person leading to a biometric overtness attack on the system.

We consider the scenario of an attacker gaining access to the fingerprint matching module of a biometric system. In this paper, we describe an attack on the BOZORTH3 [24] fingerprint matching system. BOZORTH3 is a part of the NIST biometric image software (NBIS). NBIS is used by the Department of Homeland Security and the Federal Bureau of Investigation for fingerprint matching. We use nonlinear optimization techniques to attack the BOZORTH3 fingerprint matching system. Table 1 indicates the average match score generated by the BOZORTH3 matcher for a synthesized attack fingerprint template and the minimum number of minutiae required to be changed to attack the matcher with 75% probability for FVC2002 [20], ATVS [11], and CASIA [9] datasets. To the best of our knowledge, we are the first to apply nonlinear optimization techniques to attack a fingerprint matching system.

2. Related work

Generating synthetic fingerprints has to be conducted within several constraints such as cost, time, workforce, and privacy. SFinGe [5] is one of the popular approaches that was used to generate a synthetic dataset of fingerprints. SFinGe is an acronym for synthetic fingerprint image generation. This method was proposed to produce a realistic fingerprint dataset that can be used for evaluating recognition algorithms. This method is based on three features: shape of the fingerprint, density map, and directional map. These are integrated to get a fingerprint pattern. An initial copy of the fingerprint is used to derive many randomly-generated fingerprints. The first fingerprint is generated using four phases. In the first phase, the shape of the fingerprint is obtained. In the second phase, a directional map is generated. Finally, in the third and

[*] Corresponding author.
E-mail address: steven@cs.ucf.edu (S.L. Fernandes).

https://doi.org/10.1016/j.patrec.2019.12.003
0167-8655/© 2019 Elsevier B.V. All rights reserved.

Table 1
Average match score generated by BOZORTH3 matcher for synthesized attack fingerprint template and the minimum number of minutiae required to be changed to attack the matcher with 75% probability.

	Average match score	Minimum number of minutiae required
FVC2002	111.2	10
ATVS	97.17	10
CASIA	111.07	14

fourth phase, a density map and ridge patterns are generated. The drawback of SFinGe is that the number of minutiae and their locations cannot be controlled. Cappelli et al. [7] have proposed a method to generate several fingerprints of the same finger using a single master copy. Four steps are prescribed to achieve this. In the first phase, the average thicknesses of the ridges are diversified. The second phase creates the distortion. The third phase involves adding noise and rendering. The fourth phase looks at the global translation process. This method was further enhanced using noise models [8]. However, noise models are unable to synthesize fingerprints that retain specific features such as orientation field, singular points, and minutiae. Specific features are retained by using statistical models [32].

Zhao et al. [32] proposed a statistical model-based synthetic fingerprint generation method where some pre-specified features are sampled. Four distinct stages are involved in generating the synthetic fingerprint. The first stage uses a statistical model to sample different features from a real dataset of fingerprints. Three features, namely singular points, orientation field and minutiae, are sampled. The second stage uses the AF-FM [10] method to generate a principal image of the fingerprint. The third stage applies a nonlinear plastic distortion method [6] to generate different fingerprints from the master copy. The fourth stage concentrates on the rendering where some noise, as well as finger dryness, is simulated. The statistical model-based synthetic fingerprint generation methods discussed so far do not address problems such as skin deformation, sensor malfunction, and spoofing. These problems are overcome using a synthetic three-dimensional fingerprint-based model [18]. Labati et al. [18] proposed a synthetic three-dimensional fingerprint-based model which address the issues of skin deformation, sensor malfunction, and spoofing. Statistical shape modeling is incorporated [19] in such a technique to generate random 3D synthetic fingerprints. Studies have focused on the problem of adversary attacks to biometric authentication systems using synthetically-generated fingerprints. Roy et al. [28] proposed a master print based method where the master prints are highly similar to a large number of fingerprints. Bontrager et al. [2] used latent variable evolution (LVE) for generating DeepMasterPrints.

They trained the GAN network using the fingerprint images. The latent variable of the generator network is searched by LVE for an image that could the number of successfully matched fingerprints. A synthetic attack has been simulated where the synthetic fingerprints are generated by incorporating evolutionary methods, such as differential evolution, covariance matrix adaptation evolution strategy, and swarm intelligence-based particle swarm optimization. Recently, generative adversarial networks have been found to be better in generating realistic fingerprints when compared to all the techniques discussed above [1]. Among various generative adversarial networks, the Wasserstein generative adversarial networks (WGAN) based method [1] is very popular. WGAN uses a gradient penalty for stabilizing the learning process [12]. Recently, Cao and Jain [3] further enhanced the WGAN based method. They used a convolutional autoencoder for the initialization and an improved version of the WGAN (also known as I-WGAN) for synthetic fingerprint generation. Several recent studies have proposed that the set of synthetically generated fingerprint images using GANs can be utilized to study the behavior and performance of existing fingerprint authentication systems.

Although the literature presents several methods for attacking fingerprints [30], we are the first to use a nonlinear optimizer to attack NBIS. Our approach improves upon the type of attack classified as hill climbing by Jain et al. [13]. We use the NLopt [16] nonlinear optimizer to search the space of fingerprint features, also known as minutiae features, around an unauthenticated fingerprint template. Fingerprint templates are stored files obtained from the fingerprint scanning systems. They are used to generate synthetic fingerprint templates that are classified as authenticated by the BOZORTH3 fingerprint matching system by generating a match score. MINDTCT is the minutiae extractor used. It takes an input fingerprint image and automatically extracts furrows and ridges. It is optimally designed to scan at 19.69 ppmm and quantizes using wavelet scalar quantization technique at 256 levels of gray. MINDTCT detects the points where the ridges split or end, their location, orientation, type, and quality.

We test our attack using three datasets: FVC2002, ATVS, CASIA. As per NBIS, a match score of greater than 40 indicates a true match. However, to make our attack more robust, we have set the threshold value of 35 which was proposed by Martinez-Diaz et al. [22]. A match score above 35 is considered to be a successful match. We compare our method with the hill climbing technique and show that our method produces synthetic fingerprint templates with significantly higher match scores. In this paper, we make the following contributions:

- We are able to attack the BOZORTH3 fingerprint matching system with 100% probability of successful attack for all three datasets.
- Changing only 15 minutia features allows us to attack the fingerprint matching system with high probability of success.

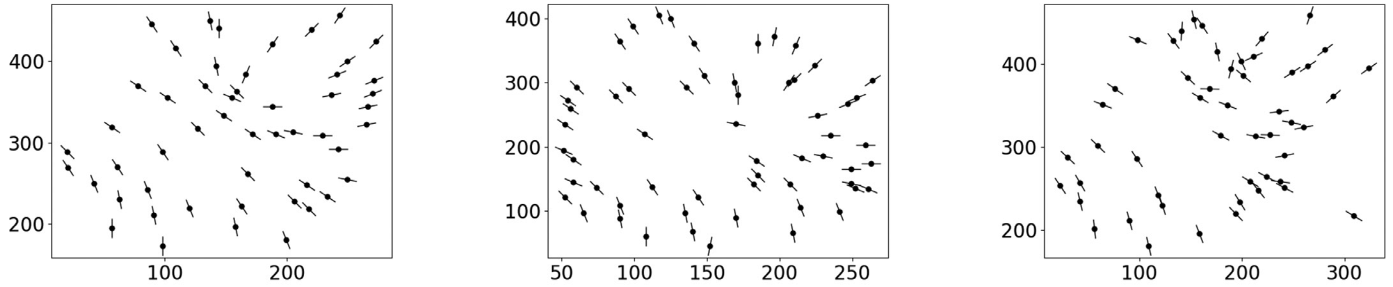
3. MINDTCT

MINDTCT is the minutia extraction used by NBIS. The orientation and location of minutiae obtained by MINDTCT are shown in Fig. 1.

The fundamental step in minutiae detection is deriving a direction map. A direction map is used to represent fingerprint areas which contain sufficient ridge structure. The fingerprint is divided into blocks such that all pixels within a block have the same direction map. The quality of the fingerprints can vary significantly. Hence it is critical to determine highly degraded areas.

Three conditions are used to detect highly degraded fingerprint areas. They are regions of high curvature, low contrast, and low ridge flow. A high curvature map occurs mainly in the core and delta regions. It indicates the blocks which are in high-curvature areas using vorticity and curvature. Vorticity measures the cumulative changes occurring in the direction of ridge flow. Curvature measures the largest change occurring in the direction between the ridge flow of a block and ridge flow of its neighbors. If minutiae are detected in these regions, the quality value assigned is reduced.

A fingerprint region is labeled low contrast if there are several blocks of significantly low contrast. The background of the image is separated from the fingerprint and minutiae are not detected in this region. To distinguish a low contrast region from a region having well-defined ridges, the pixel intensities in the regions are compared. Some regions in the fingerprint image may not have dominant ridge flow; these regions are of low quality. Low flow regions are the areas which could not be assigned a dominant ridge flow initially. If minutiae are detected in this region, they are assigned a low-quality value.



(a) Unauthenticated fingerprint template (b) Authenticated fingerprint template (c) Synthesized fingerprint template

Fig. 1. Visual representation of attack, authenticated and synthesized fingerprint templates. Match score for unauthenticated fingerprint template is 25 and match score for synthesized fingerprint template is 100.

Regions of high curvature, low contrast, and low ridge flow correspond to different low-quality regions in the image. The information obtained from the 3 low-quality regions is integrated to obtain a combined quality which contains 5 levels of quality values ranging from 0 to 4. MINDTCT detects minutiae on binary images. It scans the binary image vertically and horizontally to detect minutia points.

MINDTCT generates fingerprint templates containing minutia coordinates (x, y) and orientation (t). These templates are then processed by the BOZORTH3 fingerprint matching system.

4. BOZORTH3

BOZORTH3 is the fingerprint matching algorithm that computes match scores between fingerprint templates. It can perform one-to-one and one-to-many matching. Before BOZORTH3, NIST used bozorth98 [29] for fingerprint matching.

BOZORTH3 uses minutia coordinates (x, y) and orientation (t) to match fingerprint templates. It is robust against translation and rotation. It builds two tables containing orientation and distance values between the minutiae for each pair of fingerprint templates. Compatibility between the two tables is obtained, and a third table is constructed which stores the inter-finger compatibility values. The values of the third table are used to generate a match score. Two key features of BOZORTH3 are:

1. Minutia features are limited to the location (x, y) and the orientation (t). It is represented as (x, y, t)
2. It is robust against translation and rotation.

The BOZORTH3 algorithm is comprised of three main steps:

1. Minutia based intra-fingerprint matching
   - There are two tables, the first table contains test fingerprint templates.
   - The second table contains the training fingerprint templates against which the test fingerprint templates are matched.
2. Minutia based inter-fingerprint matching
   - The test fingerprints minutia comparison table is matched with the training fingerprints.
   - Minutia comparison table and a new compatibility table are constructed.
3. Traversing inter-fingerprint matching table entries
   - Make clusters out of compatibility table entries by linking similar tables.
   - Match compatible clusters, combine them and accumulate the matching score.

5. Method

We use BOZORTH3 as a black box system and then attempt to attack it. The BOZORTH3 fingerprint matching system takes as input a fingerprint template and returns a match score. Internally the fingerprint matching system is configured with a set of authenticated fingerprints. These fingerprints belong to authenticated individuals who are allowed access to certain confidential information. We then attack the system into wrongly classifying an unrelated fingerprint as an authenticated fingerprint. We use the popular nonlinear optimizer, NLopt, to synthesize a fingerprint template that is classified by the systems as being authenticated. An overview of our attack methodology is shown in Fig. 2.

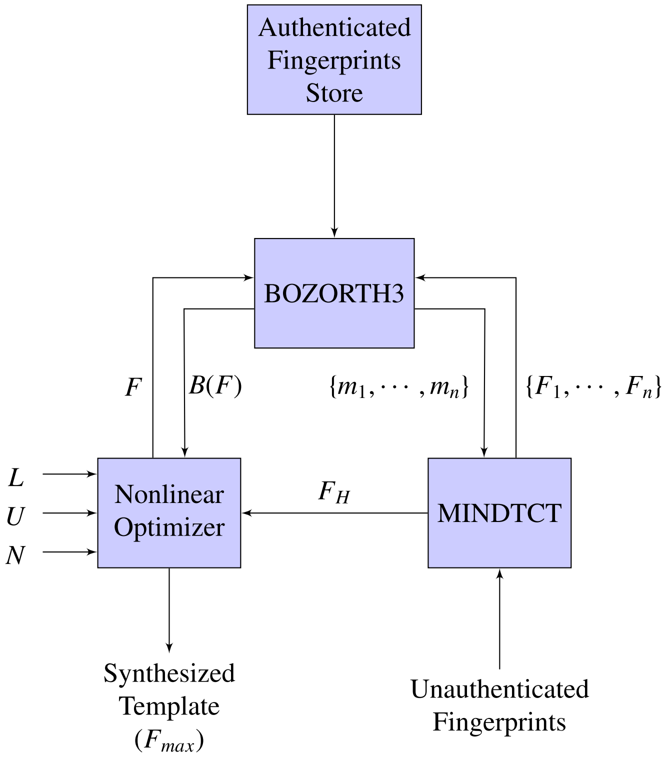


Fig. 2. Overview of our attack methodology. The optimization algorithm (NLopt) takes as input the objective function B, start template F_H, the lower bound L, the upper bound U, and the maximum number of iterations N, it returns the fingerprint template $F_{\max}$ corresponding to the highest match score.

We choose a list of n unauthenticated fingerprints and use MINDTCT to calculate fingerprint templates $\{F_1, \ldots, F_n\}$ for these fingerprints. Each fingerprint template F_i consists of a list of minutia features which are a tuple of four values: two location coordinates (x,y), feature orientation (t) and quality (q). The BOZORTH3 fingerprint matching system, by default, considers 150 best-quality minutiae for generating match scores. We query the fingerprint matching system using templates $\{F_1, \ldots, F_n\}$ and obtain match scores $\{m_1, \ldots, m_n\}$. We denote the fingerprint template having the highest match score as F_H.

We use F_H as the starting point to search for a fingerprint template that generates a high match score from the fingerprint matching system. We systematically change (x,y,t) values of up to 50 minutia features to attack the fingerprint matching system. Legacy techniques to attack fingerprint matching systems used hill climbing [23] to iteratively change F_H to obtain a successful match. Advances in optimization techniques allow us to search the space around F_H efficiently. Using nonlinear optimization techniques, we can generate adversarial fingerprint templates in less time and of higher quality when compared to legacy hill climbing methods.

We use the derivative-free optimization technique: Subplex, available in the NLopt toolkit to search for high match score fingerprint templates [27]. We define the BOZORTH3 fingerprint matching system as a function $B : \mathbf{F} \rightarrow \mathbf{M}$, where $\mathbf{F}$ is the space of all minutia features and $\mathbf{M}$ is the space of all match scores. We provide the function B as the objective function to the NLopt optimizer. The optimization function maximizes the match score $B(F)$ generated from BOZORTH3 fingerprint matching system. The search space of each minutia feature consists of three variables: x, y and t. The variables x and y are coordinates in the fingerprint image. The lower bound on the value of x and y is 0. The upper bound on the values of x and y is determined by the resolution of the fingerprint sensor.

A fingerprint sensor of resolution $l \times h$ will generate a search space where the upper bound on the value of x is l and the upper bound on the value of y is h. The variable t is the feature orientation and is calculated in arc degrees. The lower bound on the value of t is 0 and the upper bound on the value of t is 360.

The search space of M minutia features consists of $3M$ dimensions. We denote this search space by the dimensions $(x_0, y_0, t_0, \ldots, x_M, y_M, t_M)$, where the variables $(x_m,\ y_m,\ t_m)$ define the search space of m^{th} minutia feature. The lower bound on the search space of M minutia features is denoted by a tuple $L = (x_0^L, y_0^L, t_0^L, \ldots, x_M^L, y_M^L, t_M^L)$ where $x_m^L = y_m^L = t_m^L = 0$. For a fingerprint captured with a sensor of resolution $l \times h$ the upper bound on the search space of M minutia features is denoted by a tuple $U = (x_0^U, y_0^U, t_0^U, \ldots, x_M^U, y_M^U, t_M^U)$ where $x_m^U = l$, $y_m^U = h$ and $t_m^U = 360$. The optimization algorithm takes as input the BOZORTH3 fingerprint matching system as the objective function B, the start point F_H, the lower bound of search space L, the upper bound of search space U, and the maximum number of iterations, N. The optimizer returns fingerprint template $F_{\max}$ corresponding to the highest match score $B(F_{\max})$ encountered by the optimizer during the search.

We use FVC2002, ATVS and CASIA datasets to test the performance of our attack. For each dataset, we randomly separate the fingerprints into two groups: authenticated group and unauthenticated group. The authenticated group and the BOZORTH3 system together form a black box that we attack using our method. We fix the number of perturbed minutia features and observe the increase in match score generated by our algorithm.

6. Results

Results showing the average match score of synthesized fingerprint templates for various numbers of perturbed minutia features are presented in Fig. 3. Using our method we synthesize fingerprint templates with an average match score of 111.2 for FVC2002, 97.17 for ATVS and 111.07 for CASIA datasets. We have considered a match score above 35 to be a match. We observe that for all three datasets, FVC2002, ATVS and CASIA, changing only 14 minutia features allows us to successfully attack the BOZORTH3 finger matching system with more than 75% probability. Results showing the percentages of successful attack for various numbers of perturbed minutia features is shown in Fig. 4.

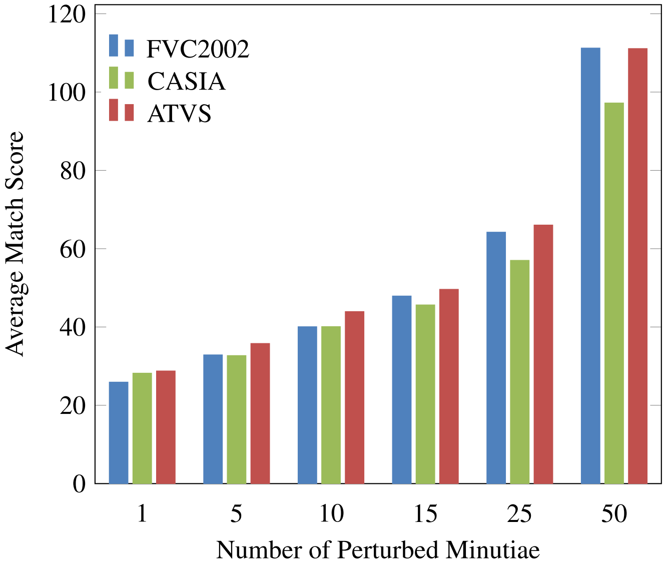


Fig. 3. Average match scores of attack fingerprints for various numbers of perturbed minutia features for the FVC2002, CASIA, and ATVS datasets.

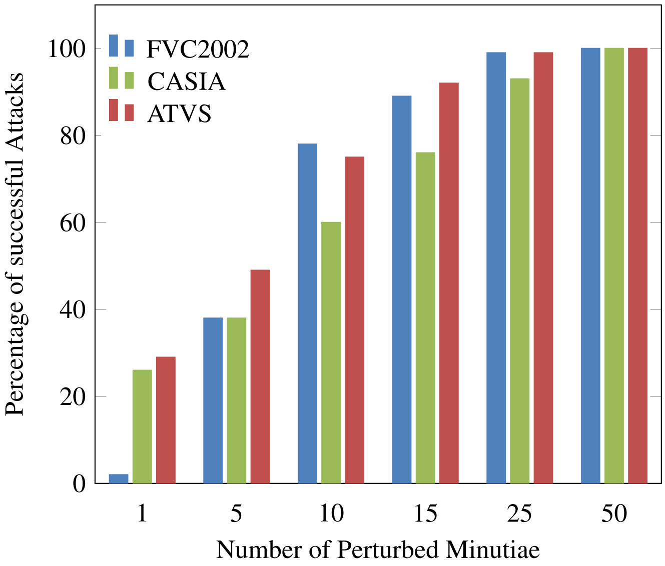


Fig. 4. Percentage of successful attacks (with threshold of 35) for FVC2002, CASIA, ATVS dataset for various numbers of perturbed minutia features.

6.1. FVC2002 Dataset

The FVC2002 dataset consists of 10 individuals each with eight samples for a total of 80 fingerprints. Each fingerprint image has a resolution of 388 × 374 pixels. We randomly choose five individuals and assigned them to the authenticated group; we kept the other half in the unauthenticated group. The maximum number of iterations to search through the space of minutiae features was configured to 10,000. We ran the experiment 100 times with different individuals in authenticated and unauthenticated groups. We observe that on changing 50 minutia features we are able to get an average match score of 111.2. We observe that changing 25 features allows us to attack the BOZORTH3 fingerprint matching system with 99% probability.

6.2. ATVS Dataset

The ATVS dataset consists of fingerprint images of 17 individuals. Four samples of fingerprints of the index and middle fingers of both hands were captured using three different scanners for a total of 816 different fingerprint images.

Each fingerprint image has a resolution of 300 × 300 pixels. We used fingerprint samples of the middle finger to test the effectiveness of our method. We randomly choose five individuals out of 17 to be part of the authenticated group, then we randomly choose 5 out of the rest of the 12 individuals for the unauthenticated group. The maximum number of iterations to search through the space of minutiae features was configured to 5000. We ran this experiment 100 times with different individuals in the authenticated and unauthenticated groups. We observe that on changing 50 minutia features, we are able to get an average match score of 111.07. We observe that changing 25 features allows us to attack the BOZORTH3 fingerprint matching system with 93% probability of successful attack.

6.3. CASIA dataset

The CASIA fingerprint dataset consists of 500 individuals. Five samples of four fingers (thumb, index, middle, ring) from each hand were captured using URU4000 fingerprint sensor for a total of 20,000 fingerprint images. Each fingerprint image has a resolution of 328 × 356 pixels. We used fingerprints samples of the left thumb to test the effectiveness of our attack. We randomly choose five individuals out of 500 to be part of the authenticated group, then we randomly choose 5 out of the rest of the 500 individuals for the unauthenticated group. The maximum number of iterations to search through the space of minutia features was configured to 5000. We ran this experiment 100 times with different individuals in the authenticated and unauthenticated groups. We observe that on changing 50 minutia features, we are able to get an average match score of 93.77. We observe that changing 25 features allows us to attack the BOZORTH3 fingerprint matching system with 99% probability.

6.4. Hill climbing

Hill climbing has been proposed as a way to attack fingerprint matching systems by multiple sources [13,17,22]. Due to the sensitive nature of these attacks, opensource implementations of hill climbing attacks are not readily available. We implemented our own hill climbing attack algorithm to compare against our proposed approach. Performance of our hill climbing attack implementation is comparable to the attack proposed by Martinez-Diaz et al. [22], where the success probability of an attack was 96.66% for 150 fingerprint images of the MCYT dataset [25]. We run the hill climbing attack on the FVC2002 dataset for 10,000 iterations and compare the results to our proposed approach. The average match score for synthesized fingerprint templates for various numbers of perturbed minutia features for hill climbing and our approach is presented in Fig. 5.

We observe that the average match scores of synthetic fingerprint templates generated by our approach is 103.50 and is significantly higher than the 57.15 match score obtained using the hill climbing approach. Percentages of successful attack for hill climbing and our approach are shown in Fig. 6. We observe that our approach has a higher probability of a successful attack for all values of perturbed minutia features. We are able to attack the fingerprint matching system with 99% probability by changing just 25 features, for comparison the hill climbing approach only has a success probability of 94%. The probability of successful attack using BOZORTH3 on FVC2002, ATVS, CASIA datasets are tabulated in Table 2. We have used 2 NVIDIA GPUs RTX 2080 with 5888 CUDA cores, 32 CPU cores, and 128 GB RAM for computation. We want to extend our method to generate high match scores for all samples of an authorized fingerprint. To the best of our knowledge, we are the first to apply NLopt to attack a fingerprint matching system.

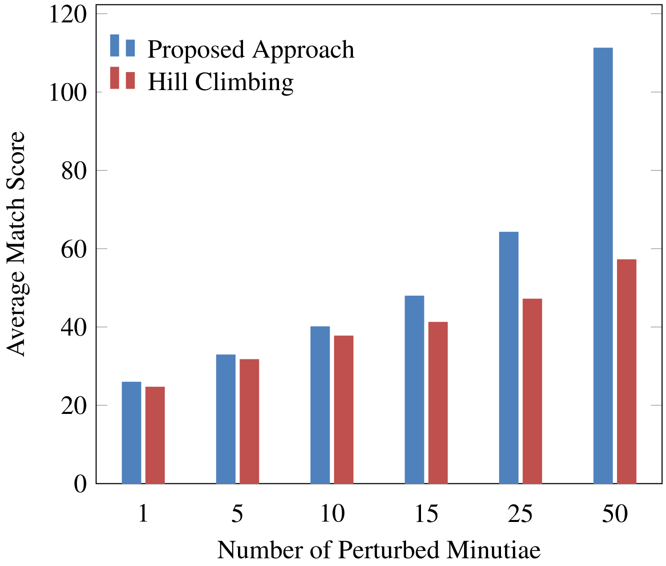


Fig. 5. Average match scores of attack fingerprint templates obtained using hill climbing and our proposed approach for FVC2002 dataset.

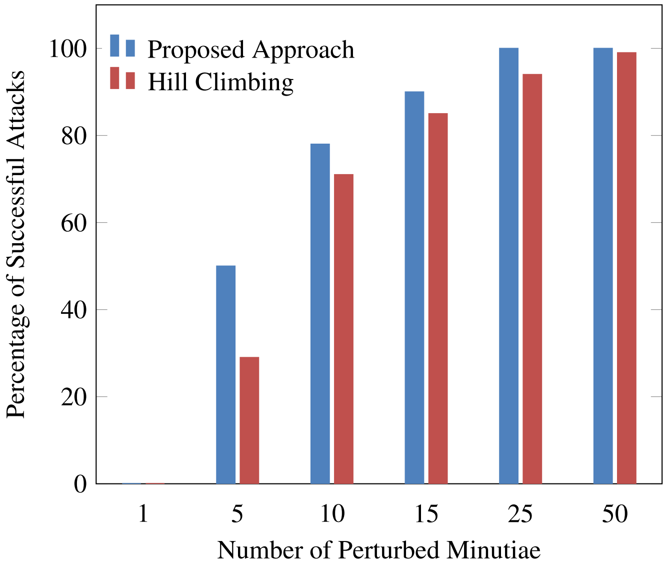


Fig. 6. Percentage of successful attacks (with threshold greater than 35) for hill climbing and our proposed approach for FVC2002 dataset.

Table 2
Probability of successful attack using BOZORTH3 on FVC2002, ATVS, and CASIA datasets .

Dataset	Features	Method	Probability of successful attack
FVC2002	25	BOZORTH3	99%
ATVS	25	BOZORTH3	93%
CASIA	25	BOZORTH3	99%

7. Conclusion and future work

We used nonlinear optimization techniques to attack the NBIS BOZORTH3 fingerprint matching system. We showed that our method to attack BOZORTH3 fingerprint matching system produces synthetic fingerprint templates with a high average match score. By searching through 50 minutia features, we obtained an average match score of 111.2, 97.17 and 111.07 for FVC2002, ATVS and CASIA datasets, respectively. We compared our method to a legacy hill climbing method and showed that our approach has a higher probability of successfully attacking the matching system. We showed that for all three datasets we are able to attack the fingerprint matching system with 100% probability of successful attack. Future work would involve making our approach more robust. We observed that both hill climbing and our approach created fingerprint templates with a high match score for only one out of multiple samples of the same fingerprint. In the future, we will validate if attribution-driven causal analysis [15] could be used to defend against adversarial attacks on fingerprints.

Declaration of Competing Interest

There is no conflict of interest

Acknowledgments

We acknowledge support from NSF Awards #1822976 and #1422257, an award from the Florida Cybersecurity Center, Royal Bank of Canada, and the Air Force Young Investigator Award to Sumit Jha.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.patrec.2019.12.003.

References

[1] M. Arjovsky, S. Chintala, L. Bottou, Wasserstein GAN, arXiv:1701.07875(2017).
[2] P. Bontrager, A. Roy, J. Togelius, N. Memon, A. Ross, Deepmasterprints: generating masterprints for dictionary attacks via latent variable evolution, in: 2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS), IEEE, 2018, pp. 1–9.
[3] K. Cao, A. Jain, Fingerprint synthesis: evaluating fingerprint search at scale, in: 2018 International Conference on Biometrics (ICB), 2018, pp. 31–38, doi:10.1109/ICB2018.2018.00016.
[4] K. Cao, A.K. Jain, Automated latent fingerprint recognition, IEEE Trans. Pattern Anal. Mach. Intell. 41 (4) (2019) 788–800, doi:10.1109/TPAMI.2018.2818162. ISSN 0162-8828.
[5] R. Cappelli, A. Erol, D. Maio, D. Maltoni, Synthetic fingerprint-image generation, in: Proceedings 15th International Conference on Pattern Recognition. ICPR-2000, 3, 2000, pp. 471–474, doi:10.1109/ICPR.2000.903586.
[6] R. Cappelli, D. Maio, D. Maltoni, Modelling plastic distortion in fingerprint images, in: Proceedings of the Second International Conference on Advances in Pattern Recognition, ICAPR '01, Springer-Verlag, Berlin, Heidelberg, 2001, pp. 369–376. http://dl.acm.org/citation.cfm?id=646260.685117.
[7] R. Cappelli, D. Maio, D. Maltoni, Synthetic fingerprint-database generation, in: Object recognition supported by user interaction for service robots, 3, 2002, pp. 744–747, doi:10.1109/ICPR.2002.1048096.
[8] R. Cappelli, D. Maio, D. Maltoni, An improved noise model for the generation of synthetic fingerprints, in: ICARCV 2004 8th Control, Automation, Robotics and Vision Conference, 2004., 2, 2004, pp. 1250–1255, doi:10.1109/ICARCV.2004.1469025.
[9] CASIA-FingerprintV5. https://doi.org/10.1007/978-1-4899-7488-4_61. Accessed: 2019-07-15.
[10] J. Feng, A.K. Jain, Fingerprint reconstruction: from minutiae to phase, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2) (2011) 209–223, doi:10.1109/TPAMI.2010.77. ISSN 0162-8828.
[11] J. Galbally, F. Alonso-Fernandez, J. Fierrez, J. Ortega-Garcia, A high performance fingerprint liveness detection method based on quality related features, Future Gener. Comput. Syst. 28 (1) (2012) 311–321, doi:10.1016/j.future.2010.11.024. http://www.sciencedirect.com/science/article/pii/S0167739X1000244X.
[12] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, A.C. Courville, Improved Training Of Wasserstein GANs, CoRR abs/1704.00028 (2017).
[13] A.K. Jain, K. Nandakumar, A. Nagar, Biometric template security, EURASIP J. Adv. Signal Process. 2008 (2008) 113.
[14] A.K. Jain, A. Ross, S. Prabhakar, An introduction to biometric recognition, IEEE Trans. Circuits Syst. Video Technol. 14 (1) (2004) 4–20, doi:10.1109/TCSVT.2003.818349. ISSN 1051-8215.
[15] S. Jha, S. Raj, S.L. Fernandes, S.K. Jha, S. Jha, G. Verma, B. Jalaian, A. Swami, Attribution-driven causal analysis for detection of adversarial examples. arXiv:1903.05821 (2019).
[16] S.G. Johnson, The nlopt nonlinear optimization package, 2014.
[17] M. Joshi, B. Mazumdar, S. Dey, Security vulnerabilities against fingerprint biometric system. CoRR abs/1805.07116 (2018).
[18] R.D. Labati, A. Genovese, V. Piuri, F. Scotti, Virtual environment for 3-D synthetic fingerprints, in: 2012 IEEE International Conference on Virtual Environments Human-Computer Interfaces and Measurement Systems (VECIMS) Proceedings, 2012, pp. 48–53.
[19] S. Long, S. Li, Q. Zhao, W. Song, 3D fingerprint modelling and synthesis, Electron. Lett. 51 (18) (2015) 1418–1420, doi:10.1049/el.2015.0382. ISSN 0013-5194.
[20] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, A.K. Jain, FVC2002: second fingerprint verification competition, in: 16th International Conference on Pattern Recognition, ICPR 2002, Quebec, Canada, August 11–15, 2002, 2002, pp. 811–814, doi:10.1109/ICPR.2002.1048144.
[21] D. Maltoni, D. Maio, A.K. Jain, S. Prabhakar, Handbook Of Fingerprint Recognition, second ed., Springer Publishing Company, Incorporated, 2009.
[22] M. Martinez-Diaz, J. Fierrez-Aguilar, F. Alonso-Fernandez, J. Ortega-Garcia, J.A. Siguenza, Hill-climbing and brute-force attacks on biometric systems: a case study in match-on-card fingerprint verification, in: Proceedings 40th Annual 2006 International Carnahan Conference on Security Technology, 2006, pp. 151–159, doi:10.1109/CCST.2006.313444.
[23] M. Martinez-Diaz, J. Fierrez-Aguilar, F. Alonso-Fernandez, J. Ortega-Garcia, J.A. Siguenza, Hill-climbing and brute-force attacks on biometric systems: a case study in match-on-card fingerprint verification, in: Proceedings 40th Annual 2006 International Carnahan Conference on Security Technology, 2006, pp. 151–159, doi:10.1109/CCST.2006.313444.
[24] NIST Biometric Image Software (NBIS), URL https://www.nist.gov/services-resources/software/nist-biometric-image-software-nbis. Accessed: 2019-09-15.
[25] J. Ortega-Garcia, J. Fierrez-Aguilar, D. Simon, J. Gonzalez, M. Faundez-Zanuy, V. Espinosa, A. Satue, I. Hernaez, J. Igarza, C. Vivaracho, D. Escudero, Q. Moro, Mcyt baseline corpus: a bimodal biometric database, IEE Proc. - Vis. Image Signal Process. 150 (6) (2003) 395–401, doi:10.1049/ip-vis:20031078.
[26] N.K. Ratha, J.H. Connell, R.M. Bolle, Enhancing security and privacy in biometrics-based authentication systems, IBM Syst. J. 40 (3) (2001) 614–634, doi:10.1147/sj.403.0614. ISSN 0018-8670.
[27] T.H. Rowan, Functional Stability Analysis of Numerical Algorithms, Austin, TX, USA, 1990 Ph.D. thesis, UMI Order No. GAX90-31702.
[28] A. Roy, N. Memon, J. Togelius, A. Ross, Evolutionary methods for generating synthetic masterprint templates: dictionary attack in fingerprint recognition, in: 2018 International Conference on Biometrics (ICB), 2018, pp. 39–46, doi:10.1109/ICB2018.2018.00017.
[29] C. Watson, C. Wilson, M. Indovina, R. Snelick, K. Marshall, Studies Of One-to-One Matching with Vendor SDK Matchers, Technical Report, Technical Report NISTIR 7119, 2004.
[30] D. Yambay, L. Ghiani, G.L. Marcialis, F. Roli, S. Schuckers, Review of fingerprint presentation attack detection competitions, in: Handbook of Biometric Anti-Spoofing, Springer, 2019, pp. 109–131.
[31] S. Yoon, A.K. Jain, Longitudinal study of fingerprint recognition, Proc. Natl. Acad. Sci. USA 112 (28) (2015) 8555–8560.
[32] Q. Zhao, A.K. Jain, N.G. Paulter, M. Taylor, Fingerprint image synthesis based on statistical feature models, in: 2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS), 2012, pp. 23–30, doi:10.1109/BTAS.2012.6374554.