A Methodology for Prevention of Biometric Presentation Attacks

(Invited Paper)

Emanuela Marasco
Department of Computer Science
University of North Carolina Charlotte
Charlotte, NC 28223-0001
Email: emarasco@uncc.edu

Mohamed Shehab
Department of Computer Science
University of North Carolina Charlotte
Charlotte, NC 28223-0001
Email: mshehab@uncc.edu

Bojan Cukic
Department of Computer Science
University of North Carolina Charlotte
Charlotte, NC 28223-0001
Email: bcukic@uncc.edu

***Abstract*—Given their widespread use for authentication, biometric systems are a key target for Presentation Attacks (PAs). A presentation attack is an attempt to circumvent a biometric system by simulating the trait of an authorized person and presenting it to the sensor. Social dimension of biometric authentication nourishes the interest in spoofing attacks. Depending on motivation and availability of resources, general users become potential attackers. There is a strong need for extensive vulnerability analysis of biometric authentication systems to aid the implementation of appropriate countermeasures. One of the methodologies for analyzing system security is based on attack trees (ATs). In ATs, attacks against a system are represented in a tree structure that helps the designer understand different ways in which the system may be attacked as well as who the attackers may be, including their abilities, motivation, and goals. Security analysts may use attack trees to identify attack patterns, which in turn can serve system designers to implement more effective defense mechanisms.**

I. Introduction

The unique biological characteristics of an individual measured by a biometric system can be imitated. The system may not able to distinguish between the biological trait of the authorized person and the artificial object. In case of fingerprint, which is a frequently attacked biometric, this process can consist of, for example, an artificial gelatin finger made to represent latent fingerprint patterns [1]. This attempt to simulate the trait of an authorized person and present it to the biometric sensor is referred to as Presentation Attack (PA). Typical PAs utilize a prosthetic to conceal the biometric signature or present an alternative biometric signature. PA techniques inhibit the intended operation of a biometric capture system, interfering with the acquisition of the true sample / identity. A simple categorization of presentation attacks is delineated in Fig. 1. Artificial (fabricated) is a category which includes inanimate objects carrying a copy of human biometric characteristics. These are made to be presented to a biometric sensor with the aim of spoofing the system into accepting it as the biometric characteristics of a human. Examples of complete artificial objects are prosthetic fingers created out of latex or a photo of a face. Hybrid cases refer to glue on finger, false facial hair, cosmetics. Masks are examples of obfuscation. Non-living samples (e.g., use of cadaver parts), non-conformant samples (e.g., use of facial extreme expressions), tip or side of fingers are categorized as natural attack methods. Face lift, amputation, mutilation correspond to instances of surgically modified attacks.

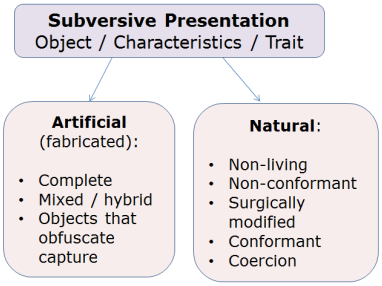

Fig. 1. Categorization of biometric presentation attacks.

We describe obfuscation and spoofing attacks for the most widely deployed modalities, fingerprints and faces.

- Fingerprint Obfuscation refers to the deliberate alteration of the fingerprint pattern (e.g., cutting or burning the fingertips) by an individual who wants to avoid being recognized by the system [2]. For example, a person on a watch list may attempt to modify his or her fingerprint pattern to prevent being matched against his or her entry on the watch list. Fingerprints can also be obliterated by burning, cutting, abrading, or simply removing a portion of the skin from the fingertip; additionally, they can be imitated by removing a portion of the skin from the fingertip, then filling the removed part with skin from other parts of the body [3]. An example of a surgically altered finger[1] is shown in Fig. 2. A realistic case happened in December 2009, when it was determined that a Chinese woman who had been previously deported from Japan had re-entered the country after surgically swapping her right hand fingerprints with those of her left hand. Officials around the world are becoming aware that mutilation can actually work to evade biometric recognition. Worldwide, multiple arrests have been made in cases involving people

[1]http://archive.boston.com/yourtown/sudbury/articles/2010/07/21/to_avoid_id_more_are_mutilating_fingerprints/

who sought to hide their identity by trying to mutilate or "erase" their fingerprints.

- Face Obfuscation. An individual may intentionally alter the appearance and features of her face in order to keep the identity hidden and remain, for example, unidentified by law enforcement [4]. The appearance of a subject can be impacted by using different disguise accessories, such as sunglasses and scarves, provided collection protocols do not require their removal. Facial disguises, use of masks, plastic surgery and make-up are methods that may be used to avoid being recognized [5]. Realistic cases[2] are shown in Fig. 3.

Fig. 2. A surgically altered finger. Image obtained from http://archive.boston.com.

Fig. 3. Sample images taken from the Hong Kong Polytechnic University Disguise and Makeup Faces Database.

- Fingerprint (Impersonation) Spoofing corresponds to a sensor-level presentation attack where an adversary intends to gain unauthorized authentication or identification by using biometric traits of someone who is legitimately enrolled in the system [6]. A well-duplicated artificial fingerprint is shown in Fig. 4. An attacker may also create a new identity using an artificial biometric trait that can be enrolled in the system and then shared between different people. Several spoofing techniques have been reported, including the use of artificial fingerprints made of gelatin, moldable plastic, Play-Doh, and silicon, produced by using a mold obtained from a live finger or from a latent fingerprint [7].
- Face (Impersonation) Spoofing can be realized with 2-D surfaces (e.g., photo, video) or 3-D volumes (e.g., masks). Photo attacks are carried out by presenting a photograph of the genuine user to the camera. The face images can be printed on a paper or may be displayed on the screen of a digital device (i.e., digital photo attacks). In video attacks, a video of the genuine user is played using a digital device. In these cases, 2-D texture and dynamics of actual human faces are copied. 3-D mask attacks, instead, imitate the complete 3-D structure of the face of the genuine user [5], [8]. Sample images[3] are shown in Fig. 5. In August 2016, a new spoofing method for facial recognition has been revealed by security specialists at the University of North Carolina [9]. Specifically, virtual reality 3-D face models have been used to bypass a face recognition system. These faces were featured by lip animation for smiling and eye motions for blinking.

Fig. 4. High resolution image samples taken from the LivDet 2015 database. On the left, a live fingerprint image. On the right, a spoof (latex) fingerprint image.

Fig. 5. Three facial masks obtained from ThatsMyFace.com. These samples are taken from the IDIAP 3D Mask Attack Database [10].

II. Engineering Approaches for Protection

In order to assess if the biometric is authentic, several liveness detection methods have been proposed. Presentation Attack Detection (PAD) modules classify biometric samples as either live (non-spoof) or fake (spoof). Liveness detection competitions (LivDet 2009, 2011, 2013, 2015) have been conducted and present an excellent source for learning [11].

A. Fingerprint Anti-Spoofing Approaches

Fingerprint recognition for automated border control and other high-security applications needs robust integrated anti-spoofing capability. Generally, presentation attacks can be detected by either gathering further evidence of the liveness of the subject (e.g. sensing blood circulation, or fluids - perspiration patterns - secreted when touching surfaces) or by

[2]http://www4.comp.polyu.edu.hk/~csajaykr/DMFaces.htm

[3]https://www.idiap.ch/dataset/3dmad

passive methods detecting the presence of known materials (e.g. material structure, lack of high-resolution detail). Several software-based methods, including Fourier Transform (FT), Local Binary Patterns (LBP), or Histograms of Invariant Gradients (HIG), have been investigated for presentation attack detection (PAD). It is particularly challenging for a learning-based algorithm to detect PAs realized with materials unseen during training [12]. Robustness of PAD methods is currently limited, and most commercial systems expose vulnerabilities in public applications.

Evaluation of perspiration patterns has been one of the earliest approaches for presentation attack detection. Perspiration is a human physiological response which is difficult to mimic in a presentation attack. Gray-level variations in fingerprint image are usually associated with moisture and perspiration in fingerprint pores. Active perspiration as an indication of liveness can be detected from a series of images taken within a second of finger presentation [13], [14]. Texture analysis has been extensively exploited for PAD. Spoof and live fingerprint images exhibit different textural properties such as morphology, smoothness, and orientation. Thus, texture can in many known cases be exploited for spoof detection. Earlier works focused on simply analyzing the spectrum of a fingerprint image. Later, the descriptor referred to as local phase quantization (LPQ) has been used to efficiently characterize the underlying image texture. Since the fingerprint can present different orientations, a rotation-invariant LPQ technique can better point out the differences in the spectrum between live fingerprints and spoof artifacts [15]. Additionally, the patch-based approach Weber local descriptor (WLD) appears to be very well suited to high-contrast patterns, such as the ridges and valleys of fingerprints images. This descriptor consists of two components: the differential excitation and the orientation. It is based on Weber‘s law and it states that the human perception of a pattern depends not only on the change of a stimulus such as lighting but also on the original intensity of the stimulus [16]. Furthermore, liveness can be predicted based on multiple histograms of invariant gradients computed in local neighborhoods [17]. Spoofed fingerprints can be detected using multiple histograms of invariant gradients (HIG) computed from spatial neighborhoods within the fingerprint. In this approach, liveness is predicted based on local histograms which count occurrences of gradient orientation and magnitude in a local region of the fingerprint image. This descriptor is designed to preserve robustness to variations in gradient positions.

In contrast to handcrafted features, learned features are ideally less dependent on sensors and on the specific fabrication material using for training. In this regards, deep learning approaches have been applied for fingerprint liveness detection [18], [19]. Deep architectures consist of feature detector units arranged in layers. Lower layers detect simple features and feed into higher layers, which detect more complex features. For example, lower layers can capture low-level features indicative of artificial fingerprint parts. Artefact indicators in the higher layers could resolve lower-level ambiguities in the image. Recently, Frassetto *et al.* [18] have examined fingerprint liveness detection using CNNs and LBPs, however they employ a hybrid approach feeding the net's output into an SVM rather than exploiting the power of deep networks. Marasco *et al.* have recently highlighted pure deep networks‘ performance and configuration options for PAD via exhaustive experimentation and a novel modelling approach. Following the idea of transfer-learning (fine-tuning using a pre-trained model) and significantly enhanced training set cardinality due to reformulation of the PAD setup, their work presents new approaches to overcome limited access to training data for the fingerprint PAD domain [19].

B. Face Anti-Spoofing Approaches

Vulnerabilities of current face recognition technology versus reproduced or synthetic face images have been addressed using a wide range of measures for detecting liveness [20], [21]. Feature-level dynamic approaches have been extensively used for detecting static face printout attacks. Typical characteristics exploited in this category of anti-spoofing methods are eye blinking and face / head gestures detected with optical flow or gaze tracking [22], [23], [24]. These motion-based methods are limited by the confusion that other motions (e.g., background motion) irrelevant to facial liveness detection can create [25]. Feature-level static approaches are based on texture analysis by extracting specific frequency components [26]. Recently, Local Binary Patterns (LBP), Gabor Wavelets and Histograms of Oriented Gradients (HOG) have been combined for successfully improving the accuracy [27]. Sensor-level approaches rely on imaging technology beyond the visual spectrum such as near infrared (NIR) images [28].

Texture-based methods achieve significant attack detection results only a single image to detect spoofs. However, textural properties cannot be generalized well and they usually tend to be specialized for certain illumination conditions. Image quality measures have been effective in both intra- and cross-database scenarios [29]. Achieving high performance in cross-database scenarios is important for a face liveness detector to be robust in realistic applications in which camera and environmental factors are usually not known to the system. Recently, Wen *et al.* proposed a method based on image distortion analysis with real-time response which is particularly accurate in cross-database scenarios. This algorithm does not rely on features which capture facial details but it extract differences in reflection properties across various materials (e.g., facial skin and paper) [25]. Methods which require additional sensors such as near infrared (NIR) and 3-D depth to capture the face, are able to be accurate even under pose or illumination variations but the extra-hardware requirement limits the applicability, for example, to smartphones. However, 3-D depth analysis by estimating the depth of a face is effective to discriminate between a 3-D live human face and a 2-D spoof face.

III. Attack Trees for Biometric Presentation Attacks

System engineers are typically reluctant to publicize data related to any system design flaw, vulnerabilities in particular. It is somewhat surprising that the owners and operators of biometric identification systems appear to be oblivious to the existence of rather obvious opportunities for presentation attacks. To some extent, this situation is similar to the early days of cybersecurity. When a vulnerability of a biometric system is exploited and thus an attack has occurred, the owners and commercial providers fear that revealing details of the attack will provoke similar attacks, ruin their reputation, and affect customer confidence. Given the right vulnerability analysis methodology and tools it should become easier for system engineers to identify and analyze potential points of attack and implement the appropriate countermeasures for each. It is vital that extensive vulnerability analysis is performed during system design and later in deployment. The methods used should greatly simplify the task of analyzing vulnerabilities and identifying possible means of exploitation. The designer can then implement appropriate countermeasures to mitigate the vulnerabilities, resolve some and document or, possibly, ignore the others. A formal methodology for analyzing the security of systems and subsystems called attack trees was first introduced by Bruce Schneier in 1999. According to Schneier, attack trees represent possible attacks against a system in a tree or graph structure, with the goal as the root node and different ways of achieving that goal as paths through the tree / graph. This methodology helps the designer understand the different ways in which the system may be attacked as well as who the attackers may be, including their abilities, motivation, and goals. Understanding the attack and the attacker sheds light on the countermeasures necessary to thwart such attacks. The problem of identifying vulnerabilities and successfully implementing countermeasures is a challenging one. Attack trees offer a means to simplify the task of vulnerability analysis. Combining the use of frameworks, attack trees, and risk assessment enables system engineers to more effectively identify and analyze vulnerabilities as well as determine the necessary countermeasures to eliminate them. It is apparent that a better method for analyzing vulnerabilities during the design process is needed so that system engineers will be less likely to overlook vulnerabilities which may later be exploited. We will illustrate attack tree methodology for presentation attack analysis with two specific examples.

- Direct Mold Artificial Fingerprint. The spoof in this attack is created from a live finger mold. The finger of the subject is pressed on the surface of a dental impression material or plaster; the negative impression of the fingerprint is fixed on it and a mold is obtained. The mold is then filled with a moisture-based material (e.g., gelatin or liquid silicon) and the spoof is produced. See Fig. 6.
- Mold from Latent Fingerprint. A method is based on a photolithographic printed circuit board (PCB) mold. The fingerprint is placed on a transparency and enhanced by brushing with a black powder. Then it is photographed by using a digital camera and printed on a transparency to create a mask for etching the PCB. The mask is placed on the circuit and exposed to UV light. The plaster cast of the fingerprint is filled with liquid silicon rubber to create a wafer-thin gummy and it is attached to a live finger before being placed on a sensor. See Fig. 6.
- Virtual Models of Faces are built from public photos of the target subject. The synthetic face of the target user is displayed on the screen of the virtual reality device. The system with a 3-D facial mesh is presented in a virtual reality environment. The mesh of the virtual reality system follows the motions (i.e., translations and rotations) of the authenticating camera. See Fig. 7.

In addition to offering an understanding of ways in which a system may be attacked, attack trees provide information on the individual performing the attack. There are certain requirements an attacker must meet in order to be capable of performing attacks. Attacker concerns can be thought of as the assets and abilities necessary to perform the attack. Each alternative requires the attacker to possess a certain set of abilities and assets. For instance, an analysis of the attack tree of Fig. 6 reveals that two means of attack may result in the goal of deceiving a fingerprint sensor. One is to learn the fingerprint pattern by corrupting or forcing an authorized user and the other to reveal it. The alternative is to obtain a finger imprint without knowledge of the target subject through a latent fingerprint collection from, for example, a glass surface. Latent fingerprints are not always easy to see or collect; therefore, the pattern must typically be enhanced with a cyanoacrylate adhesive. Once this is done it can be photographed with a digital camera. The image can then be transferred to a computer and enhanced with a photo suite such as Adobe Photoshop. In both described scenarios, the attacker is not required to carry out an expensive procedure, but some knowledge of forensics, forgery or deception is assumed. The attributes associated with the leaf nodes of the attack tree representation must be quantified in terms of the attacker and in terms of system defense. The quantification relative to the attacker is calculated by propagating the attribute aggregation in the direction of the root node [30].

A. Attack Tree Analysis

We can look at the attributes of the tree as being either attack attributes or defense attributes. Attack attributes provide an understanding of the attacker and help determine the likeliness of a particular means of exploitation. Motivation is the attacker‘s reasoning for performing the attack. In some cases this may be easy to determine, but in others there may be a plethora of motivations. Many are motivated by financial gain. Some are motivated by achieving a status and recognition. Others gain neither money nor status and are motivated by revenge or anger towards an organization. Gain may fall under the category of motivation, but risk (in terms of the attacker) is a different category altogether. The risk an attacker takes in performing

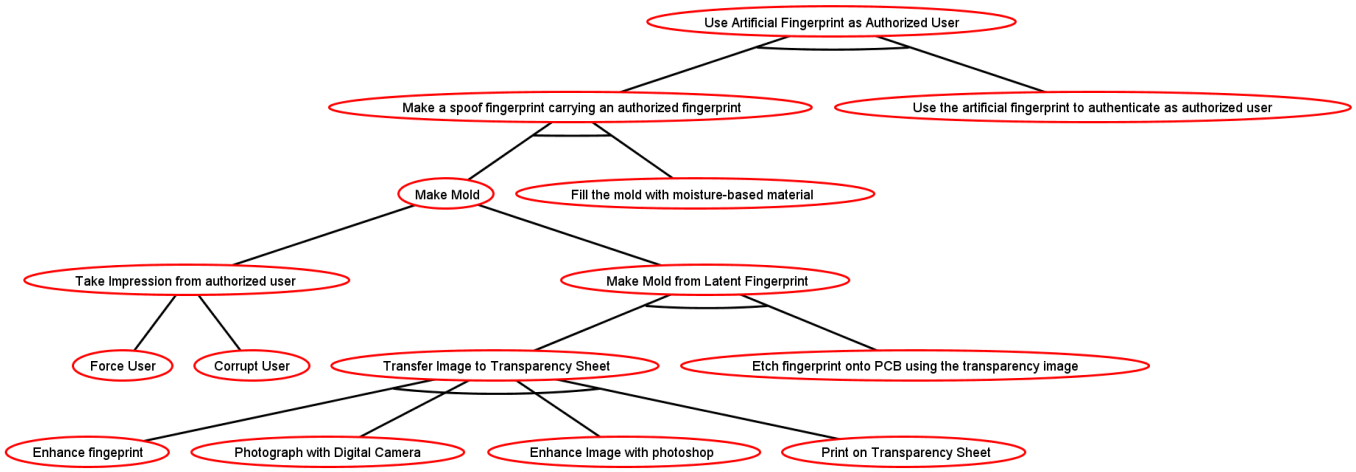

Fig. 6. Attack Tree for Mold-based Fingerprint Spoofing.

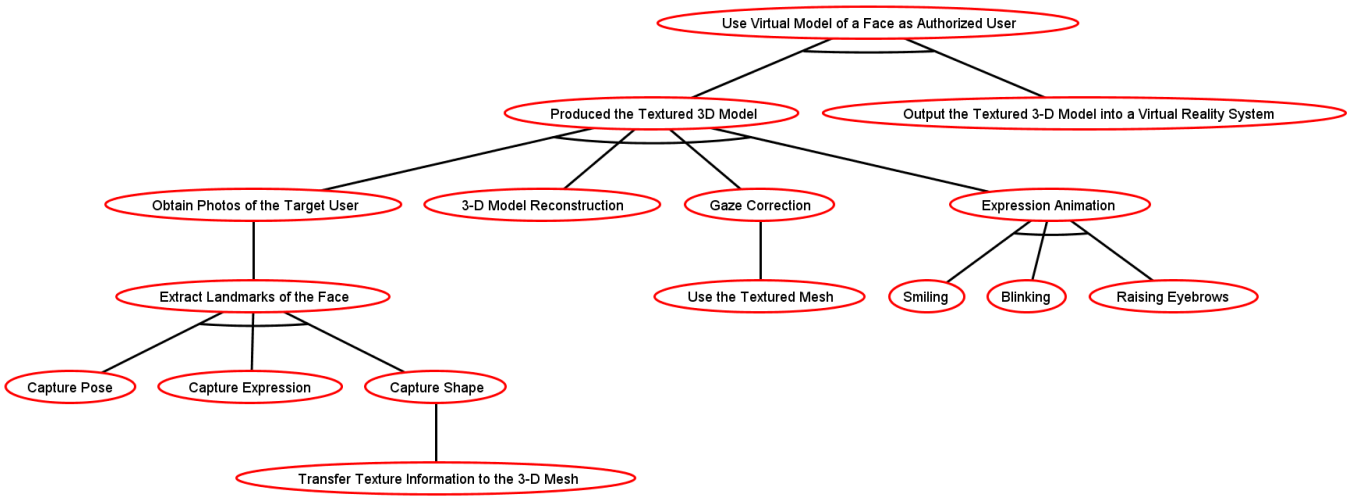

Fig. 7. Attack Tree for Virtual Reality Face Spoofing.

an attack refers to the consequences he or she will face, if caught. These may include fines, jail time, or in extreme cases (such as treason) even death. The risk one is willing to take is directly related to the motivation and the likelihood of getting caught. No one in their right mind would risk heavy fines and jail time for a low financial gain, unless the likeliness of getting caught is extremely low. On the other hand, if the likeliness of getting caught is medium but the gains are substantial it may be worth the risk. These factors are primarily attacker concerns. After all, the attacker is the one who benefits from the success of the attack. Public knowledge of the vulnerability is a factor typically controlled by the research and development community and the media. For example, attacks often occur to systems immediately following the vendor‘s disclosure of vulnerability and the release of a patch to fix it. Systems, especially those of large scale, are not patched instantaneously. It takes time for the system administrator to learn of the vulnerability and download and install the patch. Depending on the system administrator‘s competency level, the system may not be patched for quite some time. In that time attacks are more likely to occur [31], [32].

The construction of attack trees alone is not enough to determine whether or not certain security measures are necessary. Although attack tree construction offers an understanding of specific means of exploitation and the countermeasures to stop them, it is the analysis in terms of risk and cost that will determine if they should be implemented. Once the attack tree representation of a particular threat is constructed, the attributes associated with the nodes in that tree as well as the attributes associated with any countermeasures can be quantified. The quantification may be in terms of the attacker or system defense, both of which are needed. System defense concerns are factors that should be taken into consideration while developing countermeasures to thwart attacks. Among these are financial cost, risk, image and customer confidence [33].

IV. Conclusions

Assuring the security of a system is not a static task. As defensive measures are developed, more sophisticated attacks are invented. A system can be kept secure whether it is defended against a growing number of attacks. It is important to systematically document newly discovered attacks and implemented countermeasures. Attack tree are a tool for facing this challenge. The analysis of biometric systems using attack trees represent a general approach to vulnerability identification and are relatively new. Implementations of biometric systems may vary in system structure and functionality, which affect the presence of particular attack points and vulnerabilities. The task of identifying vulnerabilities and potential means of exploitation is complex and time consuming. The large task of identifying vulnerabilities may be partitioned using a framework of biometric system structure to identify potential attack points. Biometric system engineers could use this type of system analysis to share identified vulnerabilities along with the attack tree representations. This would speed up vulnerability identification process as well as improve upon the completeness of attack trees and evaluation of countermeasure effectiveness. After vulnerabilities are identified they must be analyzed to determine potential means of exploitation and develop countermeasures to thwart attacks. Much like a framework simplifies the task of vulnerability identification, the attack tree methodology simplifies the task of vulnerability analysis and mitigation. This allows the analyst to represent complex attack scenarios while maintaining a holistic view. Attack trees help the analyst understand ways in which a system may be attacked, which helps determine which (minimal) countermeasures may be necessary to thwart the attack. The use of attack trees for biometric security analysis is immature and our paper is an attempt to demonstrate its strengths and benefits.

References

[1] E. Marasco and A. Ross, "A Survey on Anti-Spoofing Schemes for Fingerprint Recognition Systems," *ACM Computing Surveys (CSUR)*, vol. 47, no. 2, p. 28, 2015.

[2] A. Jain and S. Yoon, "Automatic Detection of Altered Fingerprints." *IEEE Computer*, vol. 45, no. 1, pp. 79–82, 2012.

[3] S. Yoon, J. Feng, and A. Jain, "Altered Fingerprints: Analysis and Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 451–464, 2012.

[4] R. Singh, M. Vatsa, and A. Noore, "Face Recognition with Disguise and Single Gallery Images," *Image and Vision Computing*, vol. 27, no. 3, pp. 245–257, 2009.

[5] S. Marcel, M. Nixon, and S. Li, *Handbook of Biometric Anti-Spoofing*. Springer, 2014.

[6] K. Nixon, V. Aimale, and R. Rowe, "Spoof Detection Schemes," *Handbook of Biometrics*, 2007.

[7] A. Stén, A. Kaseva, and T. Virtanen, "Fooling Fingerprint Scanners-Biometric Vulnerabilities of the Precise Biometrics 100 SC Scanner," *Proceedings of 4th Australian Information Warfare and IT Security Conference*, vol. 2003, pp. 333–340, 2003.

[8] N. Erdogmus and S. Marcel, "Spoofing Face Recognition with 3D Masks," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 7, pp. 1084–1097, 2014.

[9] Y. Xu, T.Price, J. Frahm, and F. Monrose, "Virtual U: Defeating Face Liveness Detection by Building Virtual Models from Your Public Photos," *25th USENIX Security Symposium*, 2016.

[10] N. Erdogmus and S. Marcel, "Spoofing in 2D Face Recognition with 3D Masks and Anti-Spoofing with Kinect," *Biometrics: Theory, Applications and Systems*, 2013.

[11] L. Ghiani, D. Yambay, V. Mura, G. Marcialis, F. Roli, and S. Schuckers, "Review of the Fingerprint Liveness Detection (LivDet) Competition Series: 2009 to 2015," *Image and Vision Computing*, 2016.

[12] E. Marasco and C. Sansone, "On the Robustness of Fingerprint Liveness Detection Algorithms against New Materials used for Spoofing," *Proc. Biosignals*, pp. 1–9, 2011.

[13] ——, "Combining Perspiration- and Morphology-based Static Features for Fingerprint Liveness Detection," *Pattern Recognition Letters*, vol. 33, pp. 1148–1156, 2012.

[14] A. Abhyankar and S. Schuckers, "Modular Decomposition of Fingerprint Time Series Captures for the Liveness Check," *International Journal of Computer and Electrical Engineering*, vol. 2, no. 3, pp. 1793–8163, 2010.

[15] L. Ghiani, G. Marcialis, and F. Roli, "Fingerprint Liveness Detection by Local Phase Quantization," *Proc. ICPR*, pp. 1–4, November 2012.

[16] D. Gragnaniello, G. Poggi, C. Sansone, and L. Verdoliva, "Fingerprint Liveness Detection based on Weber Local Image Descriptor," *Proc. BioMs*, pp. 1–5, 2013.

[17] C. Gottschlich, E. Marasco, A. Yang, and B. Cukic, "Fingerprint Liveness Detection based on Histograms of Invariant Gradients," *Biometrics (IJCB), 2014 IEEE International Joint Conference on*, pp. 1–7, Sept 2014.

[18] N. Frassetto, R. Nogueira, R. Lotufo, and R. Machado, "Evaluating Software-based Fingerprint Liveness Detection using Convolutional Networks and Local Binary Patterns," *Proc. IEEE BIOMS Workshop*, pp. 22–29, 2014.

[19] E. Marasco, P. Wild, and B. Cukic, "Robust and Interoperable Fingerprint Spoof Detection via Convolutional Neural Networks," *IEEE International Conference on Technologies for Homeland Security*, pp. 1–6, 2016.

[20] X. Tan, Y. Li, J. Liu, and L. Jiang, "Face Liveness Detection from a Single Image with Sparse Low Rank Bilinear Discriminative Model," *European Conference on Computer Vision*, pp. 504–517, 2010.

[21] Y. Kim, J. Na, S. Yoon, and J. Yi, "Masked Fake Face Detection Using Radiance Measurements," *JOSA A*, vol. 26, no. 4, pp. 760–766, 2009.

[22] K. Kollreider, H. Fronthaler, and J. Bigun, "Verifying Liveness by Multiple Experts in Face Biometrics," *IEEE Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1–6, 2008.

[23] L. Wang, X. Ding, and C. Fang, "Face Live Detection Method based on Physiological Motion Analysis," *Tsinghua Science & Technology*, vol. 14, no. 6, pp. 685–690, 2009.

[24] J. Bigun, H. Fronthaler, and K. Kollreider, "Assuring Liveness in Biometric Identity Authentication by Real-Time Face Tracking," *IEEE Computational Intelligence for Homeland Security and Personal Safety*, pp. 104–111, 2004.

[25] D. Wen, H. Han, and A. Jain, "Face Spoof Detection with Image Distortion Analysis," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 746–761, 2015.

[26] J. Komulainen, A. Hadid, and M. Pietikäinen, "Face Spoofing Detection Using Dynamic Texture," *Asian Conference on Computer Vision*, pp. 146–157, 2012.

[27] J. Li, Y. Wang, T. Tan, and A. Jain, "Live Face Detection based on the Analysis of Fourier Spectra," *Defense and Security*, pp. 296–303, 2004.

[28] Z. Zhang, D. Yi, Z. Lei, and S. Li, "Face Liveness Detection by Learning Multispectral Reflectance Distributions," *IEEE Automatic Face & Gesture Recognition and Workshops*, pp. 436–441, 2011.

[29] J. Galbally, S. Marcel, and J. Fierrez, "Image Quality Assessment for Fake Biometric Detection: Application to Iris, Fingerprint, and Face Recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 710–724, 2014.

[30] D. Speicher, "Vulnerability Analysis of Biometric Systems Using Attack Trees," Ph.D. dissertation, West Virginia University, 2006.

[31] B. Schneier, "Attack Trees," *Dr. Dobb's Journal*, vol. 24, no. 12, pp. 21–29, 1999.

[32] V. Saini, Q. Duan, and V. Paruchuri, "Threat Modeling Using Attack Trees," *Journal of Computing Sciences in Colleges*, vol. 23, no. 4, pp. 124–131, 2008.

[33] B. Kordy, S. Mauw, S. Radomirović, and P. Schweitzer, "Foundations of Attack–Defense Trees," *International Workshop on Formal Aspects in Security and Trust*, pp. 80–95, 2010.