Fingerprint Spoof Detector Generalization

Tarang Chugh, *Graduate Student Member, IEEE*, and Anil K. Jain, *Life Fellow, IEEE*

***Abstract*— We present a style-transfer based wrapper, called Universal Material Generator (UMG), to improve the generalization performance of any fingerprint spoof (presentation attack) detector against spoofs made from materials not seen during training. Specifically, we transfer the style (texture) characteristics between fingerprint images of known materials with the goal of synthesizing fingerprint images corresponding to unknown materials, that may occupy the space between the known materials in the deep feature space. Synthetic live fingerprint images are also added to the training dataset to supervise the CNN to learn generative-noise invariant features which discriminate between lives and spoofs. The proposed approach is shown to improve the generalization performance of two state-of-the-art spoof detectors, namely Fingerprint Spoof Buster and Slim-ResCNN, winner of the LivDet 2017 spoof detection competition. Specifically, the performance is improved from TDR of 75.24% and 73.09% to TDR of 91.78% and 90.63% @ FDR = 0.2% for Spoof Buster and Slim-ResCNN, respectively. These results are based on a large-scale dataset of 5,743 live and 4,912 spoof images fabricated using 12 different materials. In addition to generalization across different spoof materials, the proposed approach is also shown to improve the average cross-sensor spoof detection performance from 67.60% and 64.62% to 80.63% and 77.59%, for Fingerprint Spoof Buster and Slim-ResCNN, respectively, when tested on the LivDet 2017 dataset.**

***Index Terms*— Fingerprint spoof detection, presentation attack detection, liveness detection, generalization, style transfer, fingerprint spoof buster.**

I. Introduction

With the proliferation of automated fingerprint recognition systems in many applications, including mobile payments, international border security, and national ID, fingerprint spoof attacks are of increasing concern [3], [4]. Fingerprint spoof attacks, one of the most common forms of presentation attacks,[1] include the use of *gummy fingers* [6] and *2D or 3D printed fingerprint targets* [7]–[10], *i.e.* fabricated finger-like objects with an accurate imitation of one's fingerprint to steal their identity. Other forms of presentation attacks include use of *altered fingerprints* [11], [12], *i.e.* intentionally tampered or damaged real fingerprint patterns to avoid identification, and *cadaver fingers* [13].

Fingerprint spoof attacks can be realized using a multitude of fabrication processes ranging from basic *molding*

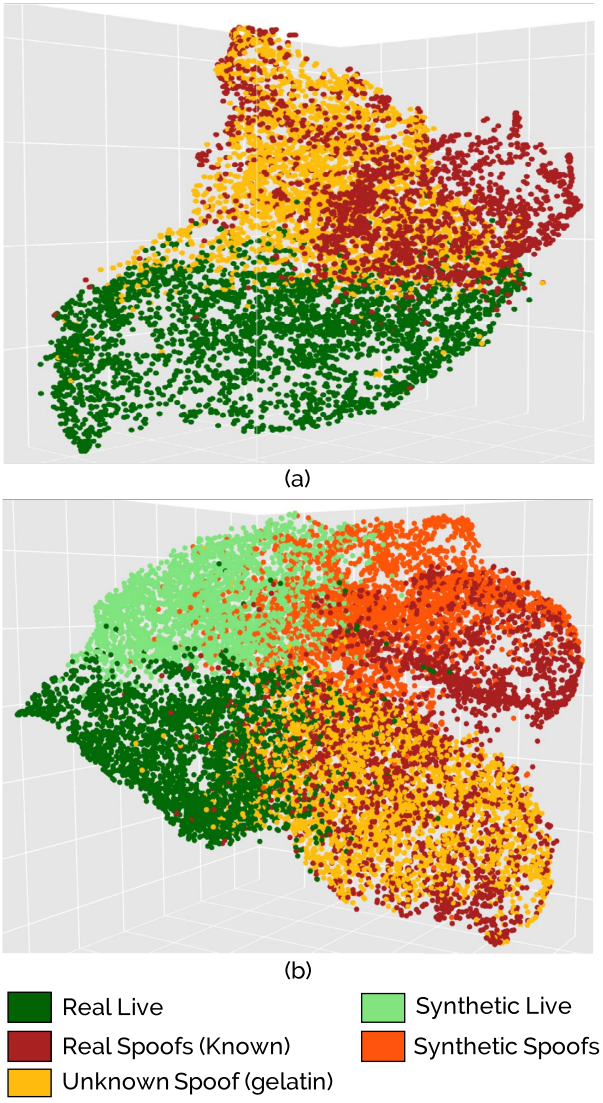

Fig. 1. 3D t-SNE visualization of feature embeddings learned by Fingerprint Spoof Buster [2] of (a) live (shown in dark green) and eleven known spoof materials (shown in red) *(2D printed paper, 3D universal targets, conductive ink on paper, dragon skin, gold fingers, latex body paint, monster liquid latex, play doh, silicone, transparency, and wood glue)* used in training, and an unknown spoof material, gelatin (shown in yellow). A large overlap between the unknown spoof, gelatin, and live feature embeddings indicate poor generalization performance of state of the art spoof detector. (b) Including synthetic live (bright green) and synthetic spoof (orange) images generated by the proposed Universal Material Generator (UMG) wrapper improve the separation between real live and real spoof. 3D t-SNE visualizations are available at http://tarangchugh.me/posts/umg/index.html.

Manuscript received December 5, 2019; revised April 17, 2020; accepted April 23, 2020. Date of publication April 27, 2020; date of current version July 27, 2020. This work was supported in part by the Office of the Director of National Intelligence (ODNI) and in part by the Intelligence Advanced Research Projects Activity (IARPA), via IARPA Research and Development under Contract 2017-17020200004. A preliminary version of this article was presented at the International Conference on Biometrics (ICB), Greece, June 4–7, 2019 [1]. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Raymond Veldhuis. *(Corresponding author: Tarang Chugh.)*

The authors are with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48824 USA (e-mail: chughtar@cse.msu.edu; jain@cse.msu.edu).

Digital Object Identifier 10.1109/TIFS.2020.2990789

[1]The ISO standard *IEC 30107-1:2016(E)* [5] defines presentation attacks as the *"presentation to the biometric data capture subsystem with the goal of interfering with the operation of the biometric system"*.

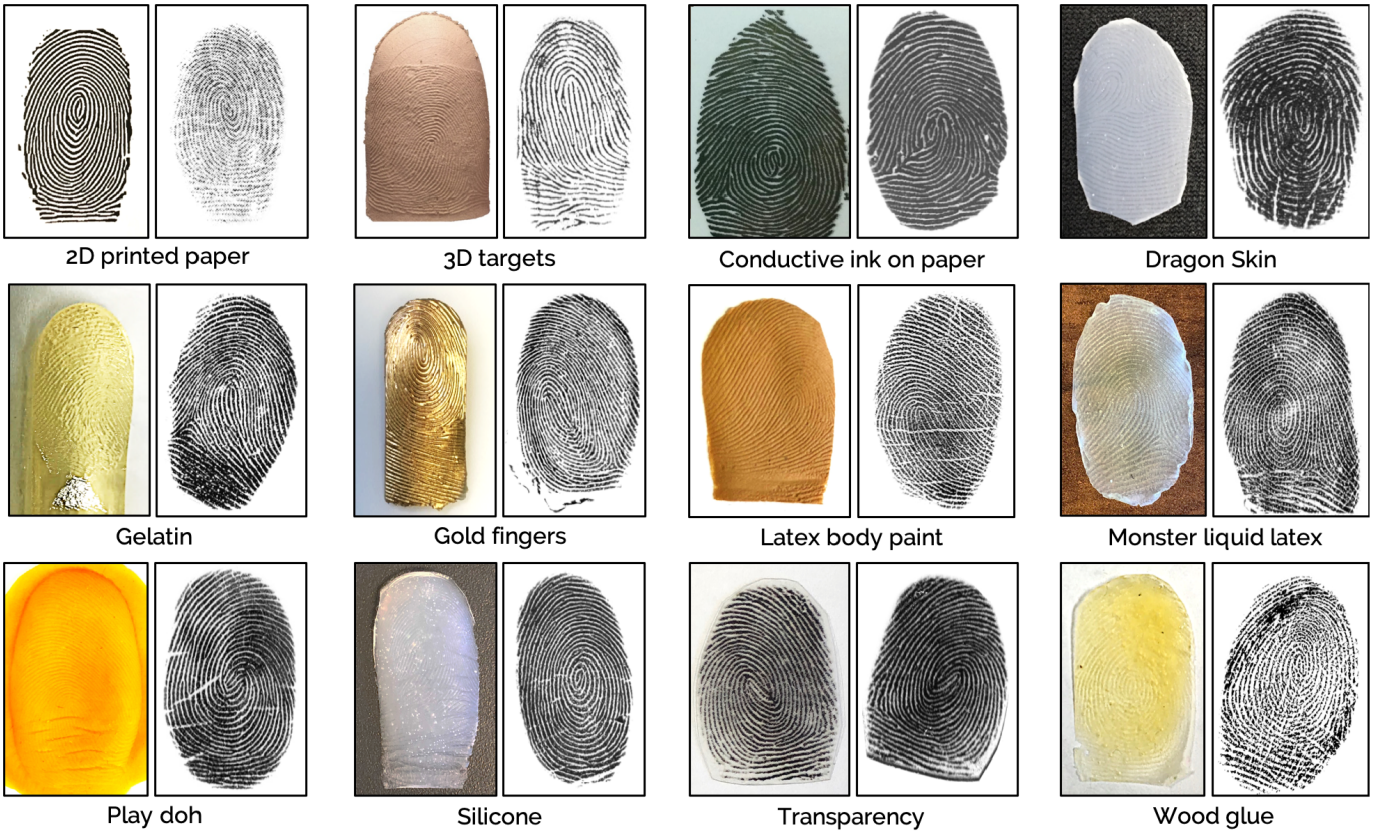

Fig. 2. Illustration of physical spoof artifacts fabricated using twelve different readily available and inexpensive spoof materials, along with grayscale fingerprint impressions captured on a CrossMatch Guardian 200 fingerprint reader. The physical artifacts and the grayscale fingerprint images do not necessarily correspond to the same finger.

and casting to utilizing sophisticated 2D and 3D printing techniques [6], [7], [9]. Readily available and inexpensive materials such as gelatin, play doh, and wood glue, have been utilized to fabricate high fidelity fingerprint spoofs which are capable of bypassing a fingerprint recognition system. For example, in March 2013, a Brazilian doctor was arrested for using spoof fingers made of silicone to fool the biometric attendance system at a hospital in Sao Paulo.[2] In July 2016, researchers at Michigan State University unlocked a fingerprint secured-smartphone using a 2D printed fingerprint spoof to help police with a homicide case,[3] using the technique proposed in [7]. In March 2018, a gang in Rajasthan, India, was arrested for spoofing the biometric attendance system, using glue casted in wax molds, to provide proxies for a police entrance exam.[4] As recent as April 2019, a Galaxy S10 owner with access to a 3D printer and a photo of his own fingerprint was able to spoof the ultrasonic in-display fingerprint sensor on his smartphone.[5] Other similar successful spoof attacks have been reported showing the vulnerabilities of fingerprint biometric systems[6],.[7] It is highly likely that a large number of these attacks are never detected and hence not reported.

In response to this growing threat, a series of fingerprint Liveness Detection (LivDet) competitions [14] have been held since 2009 to benchmark various spoof detection solutions. See [15] for results of the LivDet 2019. Several large government funded programs, including U.S. IARPA ODIN program [4] and European Union's TABULA RASA program [16], were initiated aimed towards advancing the state of the art in biometric (face, fingerprint and iris) spoof detection. The world's largest biometric system with 1.3 billion enrollment, India's Aadhaar program [17], is also funding research to detect spoof fingers, faces and irides.

Generally, fingerprint spoofs can be detected by either (i) hardware-based, or (ii) software-based approaches [3], [13]. In the case of hardware-based approaches, the fingerprint readers are augmented with sensor(s) which detect characteristics of vitality, such as blood flow, thermal output, heartbeat, skin distortion, and odor [18], [19]. Additionally, special types of fingerprint sensing technologies have been developed for imaging the sub-dermal friction ridge surface based on multi-spectral [20], short-wave infrared [21] and optical coherent tomography (OCT) [22], [23]. An open-source fingerprint reader, called RaspiReader, uses two cameras to provide

[2]https://www.bbc.com/news/world-latin-america-21756709

[3]http://statenews.com/article/2016/08/how-msu-researchers-unlocked-a-fingerprint-secure-smartphone-to-help-police-with-homicide-case

[4]https://www.medianama.com/2018/03/223-cloned-thumb-prints-used-to-spoof-biometrics-and-allow-proxies-to-answer-online-rajasthan-police-exam/

[5]https://imgur.com/gallery/8aGqsSu

[6]http://fortune.com/2016/04/07/guy-unlocked-iphone-play-doh/

[7]https://srlabs.de/bites/spoofing-fingerprints/

complementary streams (direct-view and FTIR) of images for spoof detection [24]. Ultrasound-based in-display fingerprint readers developed for smartphones by Qualcomm Inc. [25] utilize acoustic response characteristics for spoof detection.

In contrast, software-based solutions extract salient features from the captured fingerprint image (or a sequence of frames) for separating live and spoof images. The software-based approaches in the literature are typically based on (i) anatomical features (e.g. pore locations and their distribution [33]), (ii) physiological features (e.g. perspiration [34]), and (iii) texture-based features (e.g. Weber Local Binary Descriptor (WLBD) [35] and SIFT [30]. Most state-of-the-art approaches are learning-based, where the features are learned by training convolutional neural networks (CNN) [2], [21], [32], [36]–[39].

One of the major limitations of current spoof detection methods is their poor generalization performance across "unknown" or novel spoof materials, that were not used during training of the spoof detector. To generalize an algorithm's effectiveness across spoof fabrication materials, called *cross-material* performance, spoof detection has been referred to as an *open-set problem*[8] [26]. Table I presents a summary of the studies primarily focused on generalization. Engelsma and Jain [29], [42] proposed using an ensemble of generative adversarial networks (GANs) on live fingerprint images with the hypotheses that features learned by a discriminator to distinguish between real live and synthesized live fingerprints can be used to separate live fingerprints from spoof fingerprints as well. One limitation of this approach is that the discriminator in the GAN architecture may learn many features related to structural noise added by the generative process. Such features are likely not present in the spoofs fabricated with unknown materials.

It has been shown that the selection of spoof materials used in training (known spoofs) directly impacts the performance against unknown spoofs [26], [28]. In particular, Chugh and Jain [28] analyzed the material characteristics (two optical and two physical) of 12 different spoof materials to identify a representative set of six materials that cover most of the spoof feature space. Although, this approach can be used to identify if including a new spoof material in training dataset would be beneficial, it does not improve the generalization performance against materials that are unknown during training. With the increasing popularity of fingerprint authentication systems, hackers are constantly devising new fabrication techniques and novel materials to attack them. As a result, it is not feasible to include all potential spoof fabrication materials in training a spoof detector.

Another generalization is needed with respect to fingerprint sensors. Fingerprint images captured using different fingerprint sensors, typically, have unique characteristics due to different sensing technologies, sensor noise, and varying resolution. This results in poor generalization performance in the cross-sensor scenario, where the spoof detector is trained on images captured using one sensor and tested on images from another. Improving cross-sensor spoof detection performance is important in order to alleviate the time and resources involved in collecting large-scale datasets with the introduction of new sensors.

In this paper, we propose a style-transfer based method to improve both the cross-material and cross-sensor generalization performance of fingerprint spoof detectors. In particular, for the cross-material scenario, we hypothesize that the texture (style) information from the known spoof fingerprint images can be transferred from one spoof type to another type to synthesize spoof images potentially similar to spoofs fabricated from materials not seen in the training set. In the cross-sensor scenario, we utilize a small set of live fingerprint images ($\sim$ 100) from the target sensor, say Green Bit, to transfer its sensor-specific style characteristics to large-scale live and spoof datasets available from a source sensor, say Digital Persona. Our framework, called *Universal Material Generator* (UMG), is used to augment CNN-based spoof detectors, significantly improving their performance against novel materials, while retaining their performance on known materials. See Figure 5 for examples of some of the style transferred images.

Realistic image synthesis is a challenging problem. Early non-parametric methods faced difficulty in generating images with textures that are not known during training [43]. Machine learning has been very effective in this regard, both in terms of realism and generality. Gatys *et al.* [44] perform artistic style transfer, combining the content of an image with the style of any other by minimizing the feature reconstruction loss and a style reconstruction loss which are based on features extracted from a pre-trained CNN at the same time. While this approach does generate realistic looking images, it is computationally expensive since each step of the optimization requires a forward and backward pass through the pre-trained network. Other studies [45]–[47] have explored training a feed-forward network to approximate solutions to this optimization problem. There are methods based on feature statistics to perform style transfer [48], [49]. Elgammal *et al.* [50] applied GANs to generate creative art images. Isola *et al.* [51] used conditional adversarial networks to learn the loss for image-to-image translation. Xian *et al.* [52] learnt to synthesize objects consistent with texture suggestions. The proposed Universal Material Generator builds on [49] and is capable of producing realistic fingerprint images containing style (texture) information from images of two different spoof materials. Existing style transfer methods condition the source image with target material style. However, in the context of fingerprint synthesis, this results in a loss in fingerprint ridge-valley information (*i.e.* content). In order to preserve both style and content, we use adversarial supervision to ensure that the synthesized images appear similar to the real fingerprint images.

The main contributions of this study are enumerated below.

- A style-transfer based wrapper, called Universal Material Generator (UMG), to improve the generalization performance of any fingerprint spoof detector against spoofs made from materials not seen during training.

[8]Open-set problems address the possibility of new classes during testing, that were not seen during training. Closed-set problems, on the other hand, evaluate only those classes that the system was trained on.

TABLE I
SUMMARY OF THE STUDIES PRIMARILY FOCUSED ON FINGERPRINT SPOOF GENERALIZATION

Study	Approach	Database	Performance
Rattani et al. [26]	Weibull-calibrated SVM	LivDet 2011	EER = 19.70%
Ding & Ross [27]	Ensemble of multiple one-class SVMs	LivDet 2011	EER = 17.60%
Chugh & Jain [2]	MobileNet trained on minutiae-centered local patches	LivDet 2011-2015	ACE = 1.48% (LivDet 2015), 2.93% (LivDet 2011, 2013)
Chugh & Jain [28]	Identify a representative set of spoof materials to cover the deep feature space	MSU-FPAD v2.0, 12 spoof materials	TDR = 75.24% @ FDR = 0.2%
Engelsma & Jain [29]	Ensemble of generative adversarial networks (GANs)	Custom database with live and 12 spoof materials	TDR = 49.80% @ FDR = 0.2%
Gonzlez-Soler et al. [30]	Feature encoding of dense-SIFT features	LivDet 2011-2015	TDR = 7.03% @ FDR = 1% (LivDet 2015), ACE = 1.01% (LivDet 2011, 2013)
Tolosana et al. [31]	Fusion of two CNN architectures trained on SWIR images	Custom database with live and 8 spoof materials	EER = 1.35%
Gajawada et al. [1]	Style transfer from spoof to live images to improve generalization; requires few samples of target material	LivDet 2015, CrossMatch sensor	TDR = 78.04% @ FDR = 0.1%
Zhang et al. [32]	Slim-ResCNN + Center of Gravity patches	LivDet 2017	Avg. Accuracy = 95.25%
Proposed Approach	Style transfer between known spoof materials to improve generalizability against completely unknown materials	MSU-FPAD v2.0, 12 spoof materials & LivDet 2017	TDR = **91.78%** @ FDR = 0.2% (MSU-FPAD v2.0); Avg. Accuracy = **95.88%** (LivDet 2017)

ACE = Average Classification Error; EER = Equal Error Rate; TDR = True Detection Rate (spoofs); FDR = False Detection Rate (spoofs)

In literature, studies have adopted different error measures to report the performance of spoof detectors. However, in our opinion, based on the guidelines from IARPA ODIN program [4], we follow and recommend reporting TDR at a fixed FDR or, equivalently, Attack Presentation Classification Error Rate (APCER) at a fixed Bona fide Presentation Classification Error Rate (BPCER) based on ISO guidelines [5].

It attempts to synthesize impressions with style (texture) characteristics potentially similar to unknown spoof materials by interpolating the styles from known spoof materials.

- Experiments on a database of 5, 743 live and 4, 912 spoof images of 12 different materials show that the proposed approach improves the cross-material generalization performance of two state-of-the-art spoof detectors, namely Fingerprint Spoof Buster and Slim-ResCNN, from TDR of 75.4% and 73.09% to TDR of 91.78% and 90.63% @ FDR = 0.2%, respectively. Additionally, experimental results on LivDet 2017 datasets show that the proposed approach achieves state-of-the-art performance.
- Improved the cross-sensor spoof detection performance by synthesizing large-scale live and spoof datasets using only 100 live images from a new target sensor. Our approach is shown to improve the average cross-sensor spoof detection performance of Spoof Buster and Slim-ResCNN from 67.60% and 64.62% to 80.63% and 77.59% on LivDet 2017 dataset, respectively.
- Used 3D t-SNE visualization to interpret the performance improvement against unknown spoof materials.
- Fabricated physical spoof artifacts using a mixture of known spoof materials to show that the synthetically generated images using fingerprint images of the same set of spoof materials correspond to an unknown material with similar style (texture) characteristics.

We also highlight the difference between our preliminary work [1] and this paper. In [1], we utilized a small number of impressions of the new (unknown) spoof material to generate more impressions of that material and improve spoof detection performance against the "unknown" spoof materials for which only limited training data is available. In comparison, the proposed approach interpolates the style characteristics of the known spoof materials to improve the spoof detection performance against "unknown" spoof materials. Therefore, the approach here, unlike [1], does not require any impressions of the new (unknown) spoof material. The proposed approach, a style transfer-based "wrapper", not only improves the cross-material but also cross-sensor generalization performance of two state-of-the-art spoof detection solutions, Slim-ResCNN [32] (winner of LivDet 2017 competition [53]) and Fingerprint Spoof Buster [2] (best performing algorithm in IARPA ODIN program [4]).

II. PROPOSED APPROACH

The proposed approach includes three stages: (i) training the Universal Material Generator (UMG) wrapper using the spoof images of known materials (with one material left-out from training), (ii) generating synthetic spoof images using randomly selected image pairs of different but known materials, and (iii) training a spoof detector on the augmented dataset to evaluate its performance on the "unknown" material left out

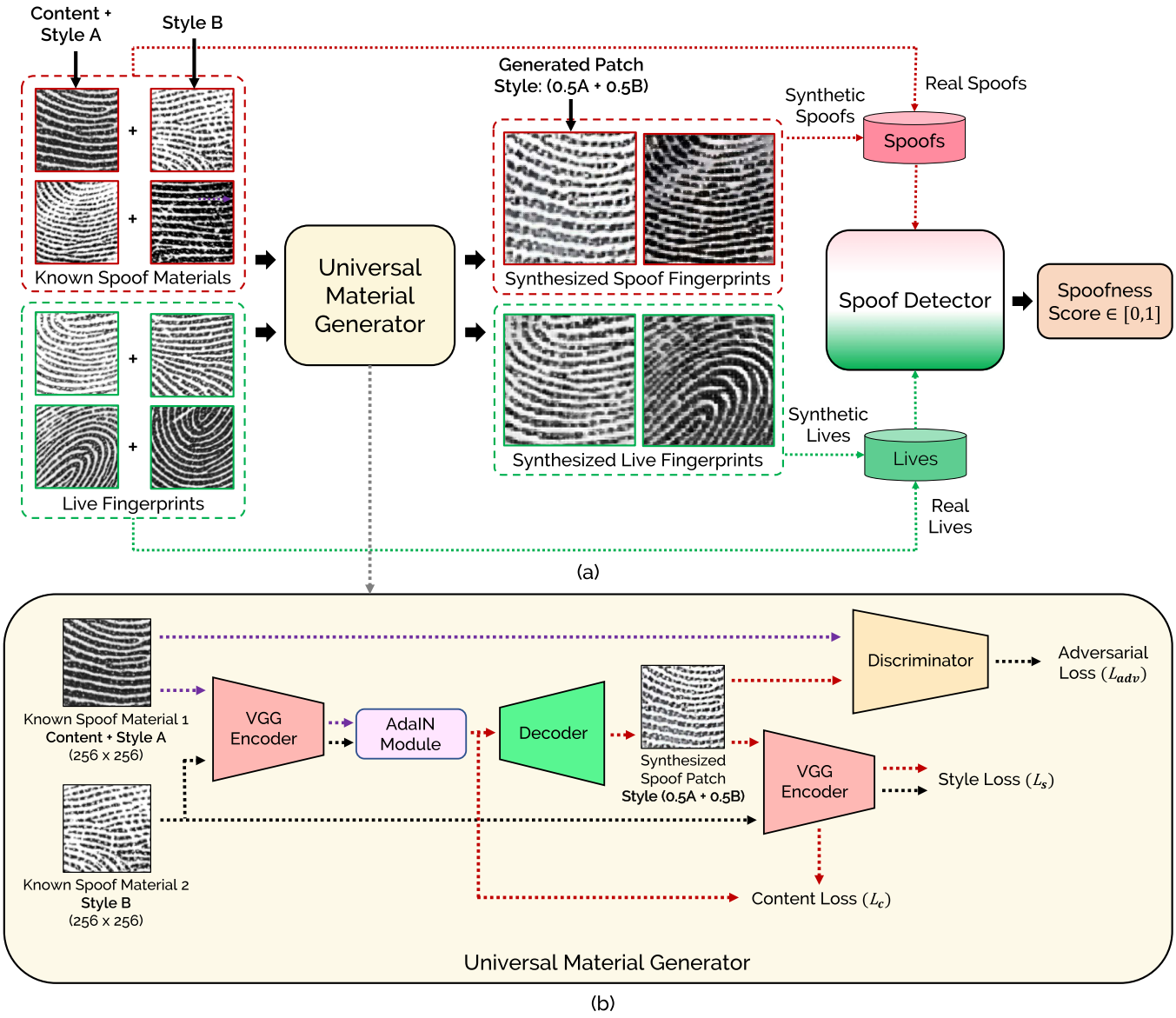

Fig. 3. Proposed approach for (a) synthesizing spoof and live fingerprint patches, and (b) design of the proposed Universal Material Generator (UMG) wrapper. An AdaIN module is used for performing the style transfer in the encoded feature space. The same VGG-19 [40] encoder is used for computing content loss and style loss. A discriminator similar to the one used in DC-GAN [41] is used for computing the adversarial loss. The synthesized patches can be used to train any fingerprint spoof detector. Hence, our approach is referred to as a wrapper which can be used in conjunction with any spoof detector.

from training. In all our experiments, we utilize local image patches (96×96) centered and aligned using minutiae location and orientation, respectively [2]. During the evaluation stage, the spoof detection decision is made based on the average of spoofness scores for individual patches. An overview of the proposed approach is presented in Fig. 3.

A. Universal Material Generator (UMG) Wrapper

The primary goal of the UMG wrapper is to generate synthetic spoof images corresponding to unknown spoof materials, by transferring the style (texture) characteristics between fingerprint images of known spoof materials. Gatys *et al.* [54] were the first to show that deep neural networks (DNNs) could encode not only content but also the style information. They proposed an optimization-based style-transfer approach, although prohibitively slow, for arbitrary images. In [48], Ulyanov *et al.* proposed use of an InstanceNorm layer to normalize feature statistics across spatial dimensions. An InstanceNorm layer is designed to perform the following operation:

$$IN(x) = \gamma\left(\frac{x - \mu(x)}{\sigma(x)}\right) + \beta \tag{1}$$

where, x is the input feature space, $\mu(x)$ and $\sigma(x)$ are the mean and standard deviation parameters, respectively, computed across spatial dimensions independently for each channel and each sample. It was observed that changing the affine parameters γ and β (while keeping convolutional parameters fixed) leads to variations in the style of the image, and the affine parameters could be learned for each particular style. This motivated an approach for artistic style transfer [55], which learns γ and β values for each feature space and style

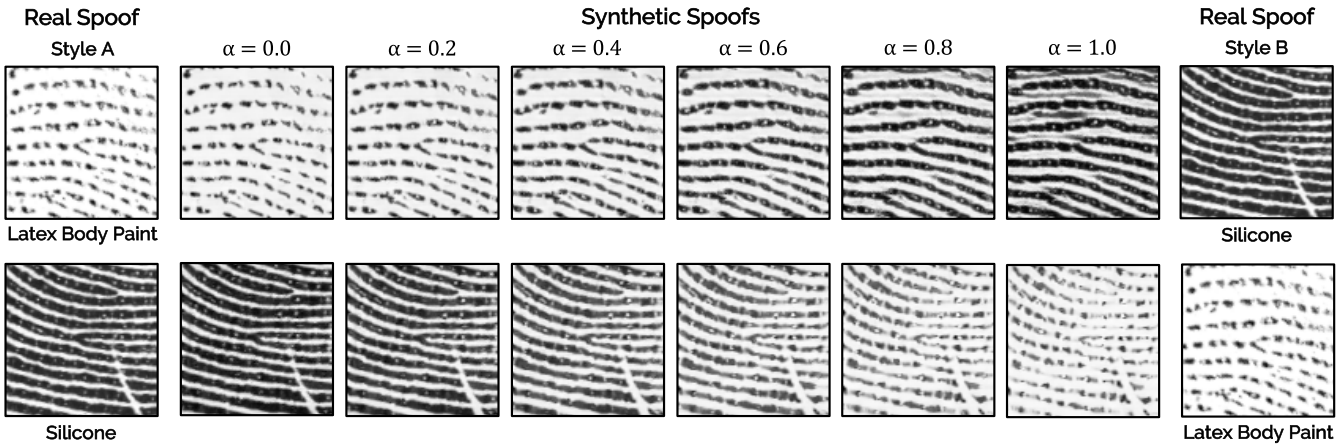

Fig. 4. Style transfer between real spoof patches fabricated with latex body paint and silicone to generate synthetic spoof patches using the proposed Universal Material Generator (UMG) wrapper. The extent of style transfer can be controlled by the parameter $\alpha \in [0, 1]$.

pair. However, this required retraining of the network for each new style.

Huang and Belongie [49] replaced the InstanceNorm layer with an Adaptive Instance Norm (AdaIN) layer, which can directly compute affine parameters from the style image, instead of learning them – effectively transferring style by imparting second-order statistics from the target style image to the source content image, through the affine parameters. We follow the same approach as described in [49] in UMG wrapper for fusing feature statistics of one known (source) spoof material image (c) providing friction ridge (content) information and source style, with another known, but different (target style) spoof material (s) in the feature space. As described in AdaIN, we apply instance normalization on the input source image feature space however not with learnable affine parameters. The channel-wise mean and variance of the source image's feature space is aligned to match those of the target image's feature space. This is done by computing the affine parameters from the target material spoof feature space in the following manner:

$$AdaIN(x, y) = \sigma(y)\Big(\frac{x - \mu(x)}{\sigma(x)}\Big) + \mu(y) \tag{2}$$

where the source (c) feature space is x and the target (s) feature space is y. In this manner, x is normalized with $\sigma(y)$ and shifted by $\mu(y)$. Our synthetic spoof generator G is composed of an encoder $f(\cdot)$ and a decoder $g(\cdot)$. For the encoder, $f(\cdot)$, we use the first four layers of a pre-trained VGG-19 network similar to [45]. The weights of this network are frozen throughout all stages of the setup. For source image (c) and the target image (s), x is $f(c)$ and y is $f(s)$. The desired feature space is obtained as:

$$t = AdaIN(f(c), f(s)) \tag{3}$$

We use the decoder, $g(\cdot)$, to take t as input to produce $T(c, s) = g(t)$ which is the final synthesized image conditioned on the style from the target image. In order to ensure that our synthesized spoof patches i.e $g(t)$ do match the style statistics of the target material spoof, we apply a style loss L_s similar to [45], [56] given as:

$$\mathcal{L}_s = \sum_{i=1}^{L} \|\mu\left(\phi_i(g(t))\right) - \mu\left(\phi_i(s)\right)\|_2 + \sum_{i=1}^{L} \|\sigma\left(\phi_i(g(t))\right) - \sigma\left(\phi_i(s)\right)\|_2 \tag{4}$$

where each ϕ_i denotes a layer in the VGG-19 network we use as encoder. We pass $g(t)$ and s through $f(\cdot)$ and extract the outputs of $relu1\_1$, $relu2\_1$, $relu3\_1$ and $relu4\_1$ layers for computing $\mathcal{L}_s$.

The extent of style transfer can be controlled by interpolating between feature maps that are:

$$T(c, s, \alpha) = g((1 - \alpha) \cdot f(c) + \alpha \cdot t) \tag{5}$$

where setting $\alpha = 0$ will reconstruct the original content image and $\alpha = 1$ will construct the most stylized image. To combine the two known styles, we preserve the style of source spoof material while conditioning it with target spoof material by setting the value of α to 0.5.

To ensure that the synthesized images retain friction ridge (fingerprint) content from the real image, we use a content loss, $\mathcal{L}_c$, which is computed as the euclidean distance between the features of the synthesized image *i.e.* $f(g(t))$ and the target features (t) from the real image.

$$\mathcal{L}_c = \|f(g(t)) - t\|_2 \tag{6}$$

Doing the style transfer, simply using a content loss ($\mathcal{L}_c$) to ensure that content is retained is not enough to ensure that the synthesized images look like real images. Fingerprints have many details in terms of structure due to the presence of certain landmarks e.g. minutiae, ridges, and pores. With the aim of synthesizing fingerprints that look indistinguishable from the real fingerprints, we use adversarial supervision. A typical generative adversarial network (GAN) setup consists of a generator G and a discriminator D playing a *minimax game*, where D tries to distinguish between synthesized and real images, and G tries to fool D by generating realistic

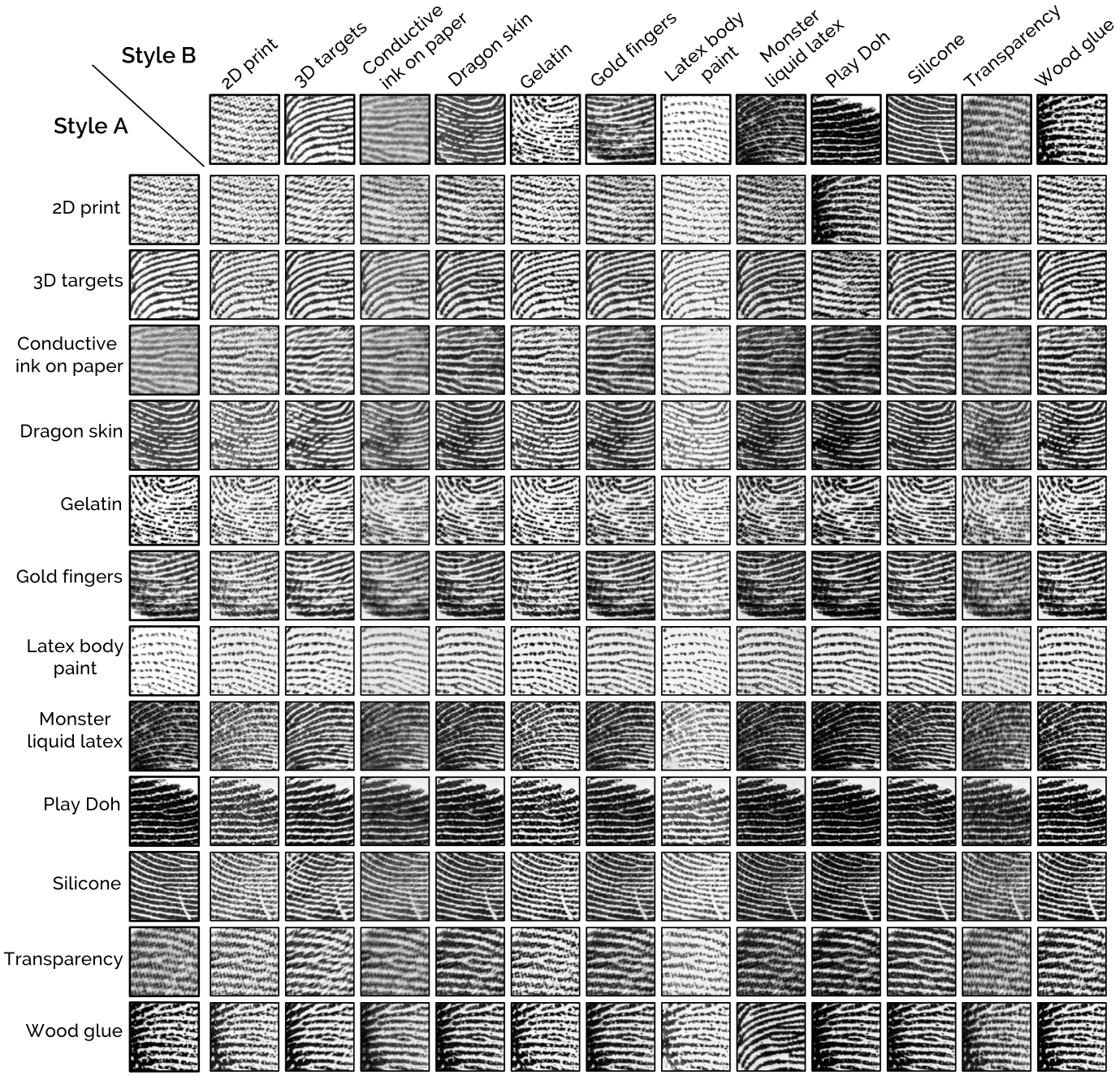

Fig. 5. Synthesized spoof patches (96 × 96) by the proposed Universal Material Generator using patches of a known (source) material (first column) conditioned on style ($\alpha = 0.5$) of another (target) known material (first row).

looking images. The adversarial objective functions for the generator ($\mathcal{L}^G_{adv}$) and discriminator ($\mathcal{L}^D_{adv}$) are given as[9]:

$$\mathcal{L}^G_{adv} = \mathbb{E}_t[\log(1 - D(G(t)))] \tag{7}$$

$$\mathcal{L}^D_{adv} = \mathbb{E}_x[\log D(x)] + \mathbb{E}_t[\log(1 - D(G(t)))] \tag{8}$$

In our approach, we use a discriminator as used in [41] and the generator is the decoder function $g(\cdot)$. We optimize the UMG wrapper in an end-to-end manner with the following objective functions:

$$\min_G \mathcal{L}_G = \lambda_c \cdot \mathcal{L}_c + \lambda_s \cdot \mathcal{L}_s + \mathcal{L}^G_{adv} \tag{9}$$

$$\max_D \mathcal{L}_D = \mathcal{L}^D_{adv} \tag{10}$$

where λ_c and λ_s are the weight parameters for content loss ($\mathcal{L}_c$) and style loss ($\mathcal{L}_s$), respectively. Algorithm 1 summarizes the steps involved in training a UMG wrapper.

B. UMG-Wrapper for Spoof Generalization

Given a spoof dataset of real images, S^m_{real}, fabricated using a set of m spoof materials, we adopt a leave-one-out protocol

[9]Here x is an image sampled from the distribution of real fingerprints, and t is the feature output by the AdaIN module.

TABLE II
SUMMARY OF THE MSU-FPAD V2 AND LIVDET 2017 DATASETS

Dataset	**MSU-FPAD v2 [28]**	**LivDet 2017 [53]**		
Fingerprint Reader	CrossMatch Guardian 200	GreenBit Dacty Scan 84C	Orcanthus Certis2 Image	Digital Persona U.are.U 5160
Image Size ($px.$) ($w \times h$)	800×750	500×500	$300 \times n^{\dagger}$	252×324
Resolution (dpi)	500	569	500	500
#Live Images (Train / Test)	$4,743$ / $1,000$	$1,000$ / $1,700$	$1,000$ / $1,700$	999 / $1,692$
#Spoof Images (Train / Test)	$4,912$ (leave-one-out)	$1,200$ / $2,040$	$1,180^{*}$ / $2,018$	$1,199$ / $2,028$
Known Spoof Materials (Training)	**Leave-one-out**: 2D Printed Paper, 3D Universal Targets, Conductive Ink on Paper, Dragon Skin, Gelatin, Gold Fingers, Latex Body Paint, Monster Liquid Latex, Play Doh, Silicone, Transparency, Wood Glue	Wood Glue, Ecoflex, Body Double		
Unknown Spoof Materials (Testing)		Gelatine, Latex, Liquid Ecoflex		

† Fingerprint images captured using Orcanthus reader have a variable height ($350 - 450px$) depending on the friction ridge content.
*A set of 20 Latex spoof fingerprints found in the training set of Orcanthus fingerprint reader were excluded in our experiments. Only Wood Glue, Ecoflex, and Body Double are expected to be in the training dataset.

to split the dataset such that spoof images fabricated using $m-1$ materials are considered as "known" and used for training. And the images fabricated using the left-out m^{th} material are considered as "unknown" and used for computing the generalization performance. The fingerprint images of known materials ($k = m-1$) are used to train the UMG wrapper (UMG_{spoof}) described in section II-A.

After we train the UMG_{spoof}, we utilize a total of N_{synth} randomly selected pairs of images $\{I^i_{m_a},\ I^i_{m_b}\}$ s.t. $i \in \{1, \ldots, N_{synth}\}$ from known but different materials $m_a, m_b \in \{m_1, \ldots m_k\}$, $a \neq b$, to generate a dataset of synthesized spoof images S^k_{synth}. For each synthesized image, the friction ridge (content) information and the source material (style) characteristics are provided by the first image, I_{m_a}, and the target material (style) characteristics are provided by the second image, I_{m_b}. See Figures 4 and 5. The real spoof dataset is augmented with the synthesized spoof data to create a dataset that is used for training the fingerprint spoof detector. Additionally, we also augment the real live dataset with a total of N_{synth} synthesized live images using another UMG wrapper (UMG_{live}) trained on only live images. Adding synthesized live data balances the data distribution and forces the spoof detector to learn generative-noise invariant features to distinguish between lives and spoofs. Figure 6 presents examples of the synthesized live images.

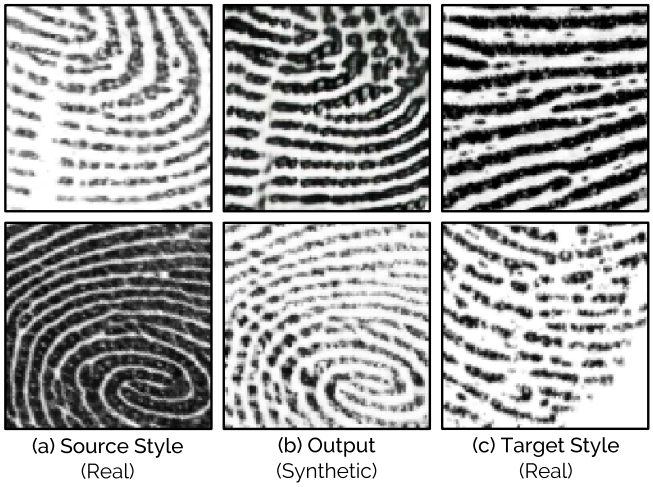

Fig. 6. Synthetic live images generated by the proposed Universal Material Generator. (a) Source style images, (c) target style images, and (b) synthesized live images.

C. Fingerprint Spoof Detection

The proposed Universal Material Generator approach acts like a wrapper on top of any existing spoof detector to make it more robust to spoofs not seen during training. In this study, we utilize two state-of-the-art spoof detectors, namely, Fingerprint Spoof Buster [2] and Slim-ResCNN [32]. Fingerprint Spoof Buster utilizes local patches (96×96) centered and aligned around fingerprint minutiae to train MobileNet-v1 [57] architecture and achieved state-of-the-art performance on publicly available LivDet databases [14] and exceeded the IARPA Odin Project [4] requirement of True Detection Rate (TDR) of 97.0% @ False Detection Rate (FDR) = 0.2%. Slim-ResCNN utilizes center of gravity-based local patches to train a custom CNN architecture containing residual blocks inspired from ResNet architecture [58], and achieved the best performance in the LivDet 2017 competition [53].

III. EXPERIMENTS AND RESULTS

A. Datasets

The following datasets have been utilized in this study:

1) MSU Fingerprint Presentation Attack Database (FPAD) v2.0: A database of $5,743$ live and $4,912$ spoof images captured on CrossMatch Guardian 200, one of the most popular slap readers.[10] The database is constructed by combining the publicly available MSU Fingerprint Presentation Attack Dataset v1.0 (MSU-FPAD v1.0) and Precise Biometrics

[10]https://www.crossmatch.com/wp-content/uploads/2017/05/20160726-DS-En-Guardian-200.pdf

TABLE III

GENERALIZATION PERFORMANCE (TDR (%) @ FDR = 0.2%) OF STATE-OF-THE-ART SPOOF DETECTORS, *i.e.*, SLIM-RESCNN [32] AND FINGERPRINT SPOOF BUSTER (FSB) [2], WITH LEAVE-ONE-OUT METHOD ON MSU-FPAD V2 DATASET. A TOTAL OF TWELVE EXPERIMENTS ARE PERFORMED WHERE THE MATERIAL LEFT-OUT FROM TRAINING IS TAKEN AS THE "UNKNOWN" MATERIAL FOR EVALUATION

Unknown Spoof Material	# Images	# Local Patches	Generalization Performance (TDR (%) @ FDR = 0.2%)			
			Base CNN		Base CNN + UMG wrapper	
			Slim-ResCNN [32]	Fingerprint Spoof Buster (FSB) [28]	Slim-ResCNN + UMG	FSB + UMG
Silicone	1,160	38,145	64.74	67.59	**96.55**	**98.62**
Monster Liquid Latex	882	27,458	90.25	94.78	**95.35**	**96.26**
Play Doh	715	17,602	58.18	58.46	**71.05**	**72.31**
2D Printed Paper	481	7,381	53.22	55.30	**79.42**	**80.25**
Wood Glue	397	12,681	84.89	86.40	**97.98**	**98.99**
Gold Fingers	295	9,402	85.08	88.14	**88.14**	**88.81**
Gelatin	294	10,508	55.78	55.10	**98.30**	**97.96**
Dragon Skin	285	7,700	96.14	97.54	**99.30**	**100.00**
Latex Body Paint	176	6,366	78.98	76.70	**90.34**	**89.20**
Transparency	137	3,846	91.24	95.62	**97.08**	**100.00**
Conductive Ink on Paper	50	2,205	88.00	90.00	**96.00**	**100.00**
3D Universal Targets	40	1,085	92.50	95.00	**100.00**	**100.00**
Total Spoofs	**4,912**	**144,379**	**Weighted mean* (± weighted s.d.)**			
Total Lives	**5,743**	**228,143**	**73.09 ± 15.66**	**75.24 ± 16.60**	**90.63 ± 10.19**	**91.78 ± 10.29**

*The generalization performance for each spoof material is weighted by the number of images to produce the weighted mean and standard deviation.

Spoof-Kit Dataset (PBSKD) [2]. Tables II and III present the details of this database, including the sensors used, 12 spoof materials, total number of fingerprint impressions, and the number of minutiae-based local patches for each material type. Fig. 2 presents sample fingerprint spoof images fabricated using the 12 materials.

2) LivDet Datasets: LivDet 2017 [53] dataset is one of the most recent publicly-available LivDet datasets,[11] containing over 17,500 fingerprint images. These images are acquired using three different fingerprint readers, namely Green Bit, Orcanthus, and Digital Persona. Unlike other LivDet datasets, spoof fingerprint images included in the test set are fabricated using new materials (Wood Glue, Ecoflex, and Body Double), that are not used in the training set (Wood Glue, Ecoflex, and Body Double). Table II presents a summary of the LivDet 2017 dataset.

B. Minutiae Detection and Patch Extraction

The proposed UMG wrapper is trained on local patches of size 96×96 centered and aligned using minutiae points. We extract fingerprint minutiae using the algorithm proposed in [59]. For a given fingerprint image I with k detected minutiae points, $M = \{m_1, m_2, \ldots, m_k\}$, where $m_i = \{x_i, y_i, \theta_i\}$, *i.e.* the minutiae m_i is defined in terms of spatial coordinates $(x_i,\ y_i)$ and orientation (θ_i), a corresponding set of k local patches $L = \{l_1, l_2, \ldots, l_k\}$, each of size $[96 \times 96]$, centered and aligned using minutiae location (x_i, y_i) and orientation (θ_i), are extracted as proposed in [2].

[11]The testing set of LivDet 2019 database has not yet been made public.

C. Implementation Details

The encoder of the UMG wrapper is the first four convolutional layers ($conv1\_1$, $conv2\_1$, $conv3\_1$, and $conv4\_1$) of a VGG-19 network [40] as discussed in section II-A. We use weights pre-trained on ImageNet [60] database which are frozen during training of the UMG wrapper. The decoder mirrors the encoder with pooling layers replaced with nearest up-sampling layers, and without use of any normalization layers as suggested in [49]. Both encoder and decoder utilize reflection padding to avoid border artifacts. The discriminator for computing the adversarial loss is similar to the one used in [41]. The weights for style loss and content loss are set to $\lambda_s = 0.002$ and $\lambda_c = 0.001$. We use the Adam optimizer [61] with a batch size of 8 and a learning rate of $(1e-4)$ for both generator (decoder) and discriminator objective functions. The input local patches are resized from 96×96 to 256×256 as required by the pre-trained encoder based on VGG-19 network. All experiments are performed in the TensorFlow framework.

The proposed approach is shown to improve the generalization performance of two state-of-the-art spoof detectors,

Algorithm 1 Training UMG Wrapper

procedure
input
  x: source image providing friction ridge content and known style A
  y: target image providing known style B
  $f(\cdot)$: encoder network; first 4 layers of VGG-19 network pre-trained on ImageNet with weights frozen during training
  $g(\cdot)$: decoder network; mirrors $f(\cdot)$ with pooling layers replaced with nearest up-sampling layers
  $D(\cdot)$: discriminator function similar to [41]
  $A(x, y)$: AdaIN operation; transfer style from x to y (using Eq. 2)
  $\alpha = 0.5$
  $\lambda_c = 0.001$, $\lambda_s = 0.002$
output
  $UMG(\cdot)$: UMG wrapper trained on known materials
begin:
  Encoding: $f_x = f(x)$ and $f_y = f(y)$
  Style transfer: $t = A(f_x, f_y)$
  Stylized image: $T(c, s, \alpha) = g((1-\alpha) \cdot f_c + \alpha \cdot t)$
  Style Loss: $\mathcal{L}_s$ using Eq. 4
  Content Loss: $\mathcal{L}_c$ using Eq. 6
  Adversarial Loss (generator): $\mathcal{L}^G_{adv}$ using Eq. 7
  Adversarial Loss (discriminator): $\mathcal{L}^D_{adv}$ using Eq. 8
  Objective functions for training UMG wrapper
  $\min_G \mathcal{L}_G = \lambda_c \cdot \mathcal{L}_c + \lambda_s \cdot \mathcal{L}_s + \mathcal{L}^G_{adv}$
  $\max_D \mathcal{L}_D = \mathcal{L}^D_{adv}$
end

namely, Fingerprint Spoof Buster and Slim-ResCNN. We train a MobileNet-V1 [57] classifier from scratch using the augmented dataset for Fingerprint Spoof Buster [2]. In the case of Slim-ResCNN, a custom architecture, consisting a series of optimized residual blocks [58] is implemented[12] as described in [32].

D. Experimental Protocol

The fingerprint spoof generalization performance against unknown materials is evaluated by adopting a leave-one-out protocol [28]. In the case of MSU FPAD v2.0 dataset, one out of the twelve known spoof materials is left-out and the remaining eleven materials are used to train the proposed UMG wrapper. The real spoof data (of eleven known materials) is augmented with the synthesized spoof data generated using the trained UMG wrapper, which is then used to train the fingerprint spoof detector. This requires training a total of twelve different UMG wrappers and spoof detection models each time leaving out one of the twelve different spoof materials. The $5,743$ live images in MSUFPAD v2.0 are partitioned into training and testing such that there are $1,000$ randomly selected live images in testing set and the remaining $4,743$ images in training such that there is no subject overlap between training and testing data splits. The real live data is also augmented with synthesized live data generated using another UMG wrapper trained on real live data.

[12]We were unable to obtain the source code for the Slim-ResCNN approach from the authors.

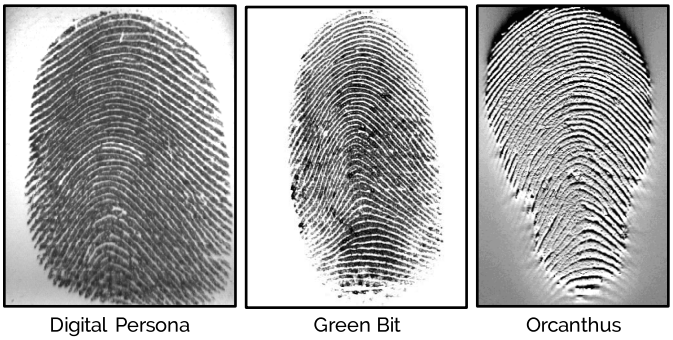

Fig. 7. Example fingerprint images from LivDet 2017 database captured using three different fingerprint readers, namely Digital Persona, Green Bit, and Orcanthus. The unique characteristics of fingerprints from Orcanthus reader explain the performance drop in cross-sensor scenario when Orcanthus is used as either the source or the target sensor.

In the case of LivDet 2017 dataset, the spoof materials available in the test set (Gelatin, Latex, and Liquid Ecoflex) are deemed as "unknown" materials because these are different from the materials included in the training set (Wood Glue, Ecoflex, and Body Double). To evaluate the generalization performance, we evaluate the performance of Fingerprint Spoof Buster with and without using the UMG wrapper and compare with the state-of-the-art published results. As the LivDet 2017 dataset contains fingerprint images from three different readers, we train two UMG wrappers per sensor, one for each of the live and the spoof training datasets.

E. Cross-Material Fingerprint Spoof Generalization

Table III presents the generalization performance of the proposed approach on the MSU FPAD v2.0 dataset. The mean generalization performance of the spoof detector against unknown spoof materials improves from TDR of 75.24% (73.09%) to TDR of 91.78% (90.63%) @ FDR = 0.2% for Fingerprint Spoof Buster (Slim-ResCNN), resulting in approximately 67% decrease in the error rate, when the spoof detector is trained in conjunction with the proposed UMG wrapper. Table IV presents a performance comparison of the proposed approach and the state-of-the-art approach when tested on the publicly available LivDet 2017 dataset. The proposed UMG wrapper improves the state-of-the-art average cross-material spoof detection performance from TDR = 73.32% (72.62%) to 80.74% (78.27%) @ FDR = 1.0% for Fingerprint Spoof Buster (Slim-ResCNN).

F. Cross-Sensor Fingerprint Spoof Generalization

To improve the cross-sensor performance, we employ the proposed UMG wrapper to synthetically generate large-scale live and spoof datasets to train a spoof detector for the target sensor. Given a real fingerprint database, D^A_{real}, collected on a source fingerprint sensor, F^A, containing real live, L^A_{real}, and real spoof S^A_{real} datasets, s.t. $D^A_{real} = \{L^A_{real} \cup S^A_{real}\}$, the proposed UMG wrapper is used to generate $50,000$ synthetic live

TABLE IV

PERFORMANCE COMPARISON BETWEEN THE PROPOSED APPROACH AND STATE-OF-THE-ART CNN-ONLY RESULTS [2], [32] ON LIVDET 2017 DATASET FOR **CROSS-MATERIAL EXPERIMENTS** IN TERMS OF AVERAGE CLASSIFICATION ACCURACY (ACA) AND TDR @ FDR = 1.0%

LivDet 2017	**Base CNN**		**Base CNN + UMG wrapper**	
	Slim-ResCNN* [32]	**FSB [28]**	**Slim-ResCNN + UMG**	**FSB + UMG**
	Avg. Accuracy (TDR @ FDR = 1.0%)		**Avg. Accuracy (TDR @ FDR = 1.0%)**	
Green Bit	95.20 (90.22)	96.68 (91.07)	96.90 (91.95)	97.42 (92.29)
Orcanthus	93.93 (65.82)	94.51 (66.59)	94.45 (71.91)	95.01 (74.45)
Digital Persona	92.89 (61.81)	95.12 (62.29)	94.75 (70.96)	95.20 (75.47)
Mean ± s.d.	94.01 ± 1.16 (72.62 ± 15.38)	95.44 ± 1.12 (73.32 ± 15.52)	**95.37 ± 1.34** (**78.27 ± 11.85**)	**95.88 ± 1.34** (**80.74 ± 10.02**)

*We were unable to obtain the source code for the Slim-ResCNN approach from the authors. Best efforts were made to implement the approach based on the details provided in their manuscript [32]. Based on LivDet 2017 [53], Slim-ResCNN achieved average classification accuracy of 95.25% compared to 94.01% achieved by our implementation.

TABLE V

CROSS-SENSOR FINGERPRINT SPOOF GENERALIZATION PERFORMANCE ON LIVDET 2017 DATASET IN TERMS OF AVERAGE CLASSIFICATION ACCURACY AND TDR @ FDR = 1.0%

LivDet 2017	**Slim-ResCNN [32]**	**FSB [28]**	**Slim-ResCNN + UMG**	**FSB + UMG**
Training (Testing) Sensors	**Avg. Accuracy (TDR @ FDR = 1.0%)**		**Avg. Accuracy (TDR @ FDR = 1.0%)**	
Green Bit (Orcanthus)	43.98 (0.00)	49.43 (0.00)	**65.40** (**20.60**)	**66.05** (**21.52**)
Green Bit (Digital Persona)	80.39 (48.28)	89.37 (57.48)	**92.07** (**69.55**)	**94.81** (**72.91**)
Orcanthus (GreenBit)	68.82 (8.02)	69.93 (8.02)	**74.38** (**29.90**)	**81.75** (**30.91**)
Orcanthus (Digital Persona)	62.30 (6.70)	57.99 (4.97)	**72.33** (**25.24**)	**76.36** (**28.46**)
Digital Persona (GreenBit)	87.90 (54.24)	89.54 (57.06)	**95.28** (**84.38**)	**96.35** (**85.21**)
Digital Persona (Orcanthus)	44.30 (0.00)	49.32 (0.00)	**66.10** (**18.25**)	**68.44** (**20.38**)
Mean ± s.d.	64.62 ± 18.18 (19.54 ± 24.86)	67.60 ± 18.53 (21.26 ± 28.06)	**77.59 ± 12.97** (**41.32 ± 28.29**)	**80.63 ± 12.88** (**43.23 ± 28.31**)

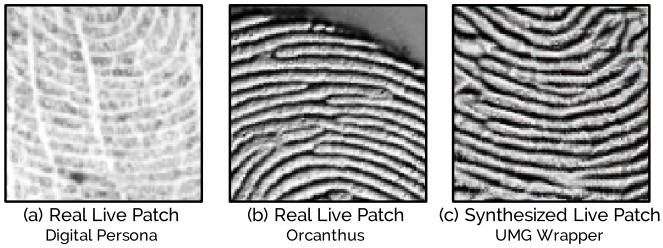

Fig. 8. UMG wrapper used to transfer style from (b) a real live patch from Orcanthus reader, to (a) a real live patch from Digital Persona, to generate (c) a synthesized patch.

patches, L^B_{synth}, and $50,000$ synthetic spoof patches, S^B_{synth}, for a target sensor, F_B. The UMG wrapper is trained only on the live images collected on S_B, and used for style transfer on L^A_{real} and S^A_{real} to generate L^B_{synth}, and S^B_{synth}, respectively. We evaluate the cross-sensor generalization performance using LivDet 2017 dataset where the UMG wrapper trained on source sensor, say Green Bit, is used to generate synthetic data for a target sensor, say Orcanthus, using only a small set of 100 live fingerprint images from the target sensor.[13] The spoof detector is trained from scratch only on the synthetic dataset created for the target sensor using UMG wrapper and tested on the real test set of the target sensor. Table V presents the cross-sensor fingerprint spoof generalization performance in terms of average classification accuracy and TDR (%) @ FDR = 1%. We note that the proposed UMG wrapper improves the average cross-sensor spoof detection performance from 67.60% (64.62%) to 80.63% (77.59%) for Fingerprint Spoof Buster (Slim-ResCNN). Figure 7 presents example fingerprint images captured using the three sensors in LivDet 2017. The unique characteristics of fingerprints from Orcanthus reader explain the performance drop in cross-sensor scenario when it is used as either the source or the target sensor.

[13] An average of $\sim$ 3100 local patches are extracted from 100 live fingerprint images in LivDet 2017 experiments.

G. Computational Requirements

Offline Training stage: The proposed approach includes an offline stage of training the UMG wrapper and synthesis of style-transferred fingerprint patches. It takes around 2 hours to train, and around 1 hour to generate $100,000$ fingerprint patches on a Nvidia GTX 1080Ti GPU. The synthesized fingerprint patches are used to augment the training data used to train the underlying spoof detector.

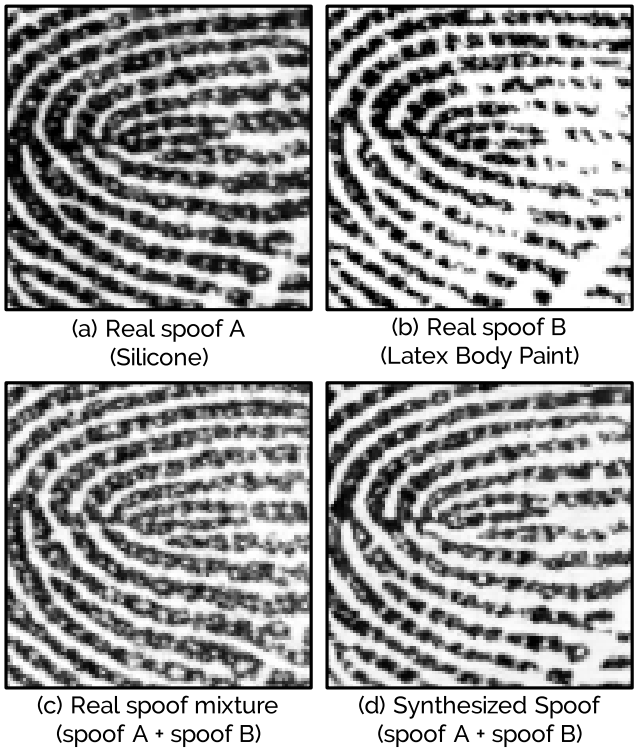

Fig. 9. Fingerprint patches fabricated with real spoofs (a) silicone, (b) latex body paint, (c) their mixture (in 1:1 ratio), and (d) synthesized using UMG wrapper with style transfer between silicone and latex body paint.

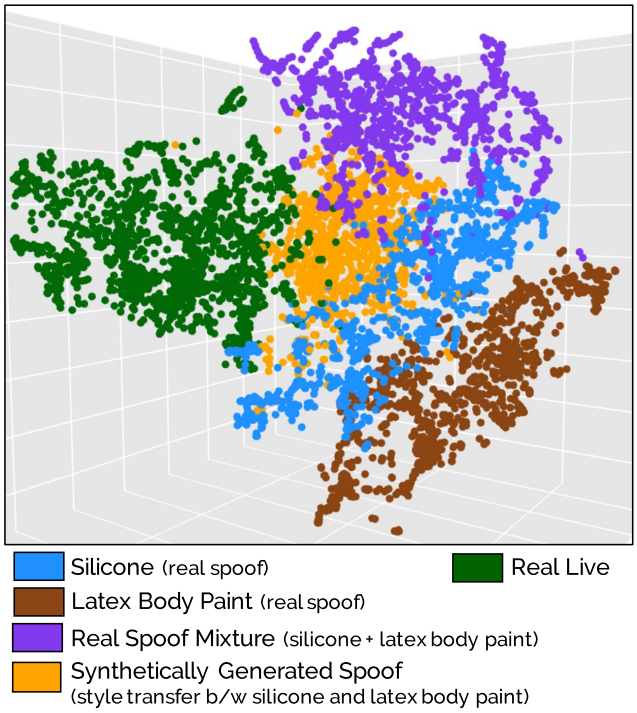

Fig. 10. 3D t-SNE visualization of feature embeddings of real live fingerprints, spoof fingerprints fabricated using silicone, latex body paint, and their mixture (1:1 ratio), and synthesized spoof fingerprints using style-transfer between silicone and latex body paint spoof fingerprints. The 3D embeddings are available at http://tarangchugh.me/posts/umg/index.html.

Online Testing stage: There is no increase in the spoof detection time of the underlying spoof detector with the addition of the UMG wrapper. The spoof detection time remains around 100ms for both Fingerprint Spoof Buster and Slim-ResCNN.

IV. Fabricating Unknown Spoofs

To explore the role of cross-material style transfer in improving generalization performance, we fabricate physical spoof specimens using two spoof materials, namely silicone and latex body paint, and their mixture in a 1:1 ratio by volume.[14] We fabricate a total of 24 physical specimens, including 8 specimens for each of the two materials, and 8 specimens using their mixture. A total of 72 spoof fingerprints, 3 impressions/specimen, are captured using a Cross-Match Guardian 200 fingerprint reader. Fingerprint Spoof Buster, trained on twelve known spoof materials including silicone and latex body paint, achieves TDR of 100.0% @ FDR = 0.2% on the two known spoof materials, and TDR of 83.33% @ FDR = 0.2% against the mixture. We utilize the testing dataset of 1,000 live fingerprint images from MSU FPAD v2.0 for these experiments.

We utilize the proposed UMG wrapper to generate a dataset of 5,000 synthesized spoof patches[15] using cross-material style transfer between spoof fingerprints of silicone and latex body paint. Fingerprint Spoof Buster, fine-tuned using the synthesized dataset, improves the TDR from 83.33% to 95.83% @ FDR = 0.2% when tested on the mixture of silicone and latex body paint, highlighting the role of the style-transferred synthesized data in improving generalization performance. Figure 9 presents sample fingerprint patches of the two spoof materials, silicone and latex body paint, their physical mixture, and synthesized using style-transfer. Figure 10 presents the 3D t-SNE visualization of feature embeddings of live fingerprints (green), two materials, silicone (blue) and latex body paint (brown), their mixture (purple), and synthetically generated images (orange). Although the mixture embeddings are not exactly in between the embeddings for the two known materials, possibly due to low-dimensional t-SNE representation, they are close to the embeddings of the synthetically generated spoof images. This explains the improvement in performance against mixture when synthesized spoofs are used in training. Therefore, the proposed UMG wrapper is able to generate spoof images that are potentially similar to the unknown spoofs.

V. Conclusion

Automatic fingerprint spoof detection is critical for secure operation of a fingerprint recognition system. Introduction of new spoof materials and fabrication techniques poses a continuous threat and requires design of robust and generalizable spoof detectors. To address that, we propose a style-transfer based wrapper, Universal Material Generator (UMG), to improve the generalization performance of any

[14]Not all spoof materials can be physically combined and may result in mixtures with poor physical properties for them to be used to fabricate any good quality spoof artifacts.

[15]Around 1,100 minutiae-based local patches are extracted from 24 fingerprint images corresponding to each material.

spoof detector against novel spoof fabrication materials that were unknown during training of the spoof detector. The proposed approach is shown to improve the average generalization performance of two state-of-the-art spoof detectors, namely Fingerprint Spoof Buster (and Slim-ResCNN), from TDR of 75.24% (73.09%) to 91.78% (90.63%) @ FDR = 0.2%, respectively, when evaluated on a large-scale dataset of 5,743 live and 4,912 spoof images fabricated using 12 materials. Our approach also improves the average cross-sensor performance from 67.60% (64.62%) to 80.63% (77.59%) for Fingerprint Spoof Buster (Slim-ResCNN) when tested on LivDet 2017 dataset, alleviating the time and resources required to generate large-scale spoof datasets for every new sensor and spoof material. We have also fabricated physical spoof specimens using a mixture of known spoof materials to explore the role of cross-material style-transfer in improving generalization performance.

ACKNOWLEDGMENT

The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein. This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA R&D Contract No. 2017-17020200004.

REFERENCES

[1] R. Gajawada, A. Popli, T. Chugh, A. Namboodiri, and A. K. Jain, "Universal material translator: Towards spoof fingerprint generalization," in *Proc. Int. Conf. Biometrics (ICB)*, Jun. 2019, pp. 1–8.

[2] T. Chugh, K. Cao, and A. K. Jain, "Fingerprint spoof buster: Use of minutiae-centered patches," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 9, pp. 2190–2202, Sep. 2018.

[3] S. Marcel, M. S. Nixon, J. Fierrez, and N. Evans, Eds., *Handbook of Biometric Anti-Spoofing: Presentation Attack Detection*, 2nd ed. Cham, Switzerland: Springer, 2019.

[4] ODNI, IARPA. (2016). *IARPA-BAA-16-04 (Thor)*. [Online]. Available: https://www.iarpa.gov/index.php/research-programs/odin/odin-baa

[5] *Information Technology—Biometric Presentation Attack Detection—Part 1: Framework*, Standard ISO/IEC 30107-1:2016, International Standards Organization, 2016. [Online]. Available: https://www.iso.org/standard/53227.html

[6] T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino, "Impact of artificial gummy fingers on fingerprint systems," *Proc. SPIE*, vol. 4677, pp. 275–289, Feb. 2002.

[7] K. Cao and A. K. Jain, "Hacking mobile phones using 2D printed fingerprints," MSU, Tirunelveli, India, Tech. Rep. MSU-CSE-16-2, 2016. [Online]. Available: https://www.youtube.com/watch?v=fZJI_BrMZXU

[8] S. S. Arora, K. Cao, A. K. Jain, and N. G. Paulter, "Design and fabrication of 3D fingerprint targets," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 10, pp. 2284–2297, Oct. 2016.

[9] S. S. Arora, A. K. Jain, and N. G. Paulter, "Gold fingers: 3D targets for evaluating capacitive readers," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 9, pp. 2067–2077, Sep. 2017.

[10] J. J. Engelsma, S. S. Arora, A. K. Jain, and N. G. Paulter, "Universal 3D wearable fingerprint targets: Advancing fingerprint reader evaluations," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 6, pp. 1564–1578, Jun. 2018.

[11] S. Yoon, J. Feng, and A. K. Jain, "Altered fingerprints: Analysis and detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 451–464, Mar. 2012.

[12] E. Tabassi, T. Chugh, D. Deb, and A. K. Jain, "Altered fingerprints: Detection and localization," in *Proc. IEEE 9th Int. Conf. Biometrics Theory, Appl. Syst. (BTAS)*, Oct. 2018, pp. 1–9.

[13] E. Marasco and A. Ross, "A survey on antispoofing schemes for fingerprint recognition systems," *ACM Comput. Surveys*, vol. 47, no. 2, pp. 1–36, Jan. 2015.

[14] D. Yambay, L. Ghiani, G. L. Marcialis, F. Roli, and S. Schuckers, "Review of fingerprint presentation attack detection competitions," in *Handbook of Biometric Anti-Spoofing*, S. Marcel, M. S. Nixon, J. Fierrez, and N. Evans, Eds. Cham, Switzerland: Springer, 2019.

[15] G. Orrù *et al.*, "LivDet in action–fingerprint liveness detection competition 2019," 2019, *arXiv:1905.00639*. [Online]. Available: http://arxiv.org/abs/1905.00639

[16] European Commission. (2011). *TABULA RASA*. [Online]. Available: http://www.tabularasa-euproject.org/front-page

[17] (2019). *Unique Identification Authority of India: Dashboard, Government of India*. [Online]. Available: https://uidai.gov.in/aadhaar_dashboard/

[18] A. Antonelli, R. Cappelli, D. Maio, and D. Maltoni, "Fake finger detection by skin distortion analysis," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 3, pp. 360–373, Sep. 2006.

[19] D. Baldisserra, A. Franco, D. Maio, and D. Maltoni, "Fake fingerprint detection by odor analysis," in *Proc. ICB*. Berlin, Germany: Springer, 2006, pp. 265–272.

[20] C. D. Robison and M. S. Andrews, "System and method of fingerprint anti-spoofing protection using multi-spectral optical sensor array," U.S. Patent 10 242 245, Mar. 26, 2019.

[21] R. Tolosana, M. Gomez-Barrero, J. Kolberg, A. Morales, C. Busch, and J. Ortega-Garcia, "Towards fingerprint presentation attack detection based on convolutional neural networks and short wave infrared imaging," in *Proc. Int. Conf. Biometrics Special Interest Group (BIOSIG)*, Sep. 2018, pp. 1–5.

[22] Y. Moolla, L. Darlow, A. Sharma, A. Singh, and J. Van Der Merwe, "Optical coherence tomography for fingerprint presentation attack detection," in *Handbook of Biometric Anti-Spoofing*. Cham, Switzerland: Springer, 2019.

[23] T. Chugh and A. K. Jain, "OCT fingerprints: Resilience to presentation attacks," 2019, *arXiv:1908.00102*. [Online]. Available: http://arxiv.org/abs/1908.00102

[24] J. J. Engelsma, K. Cao, and A. K. Jain, "RaspiReader: An open source fingerprint reader facilitating spoof detection," 2017, *arXiv:1708.07887*. [Online]. Available: http://arxiv.org/abs/1708.07887

[25] M. Agassy, B. Castro, A. Lerner, G. Rotem, L. Galili, and N. Altman, "Liveness and spoof detection for ultrasonic fingerprint sensors," U.S. Patent 10 262 188, Apr. 16, 2019.

[26] A. Rattani, W. J. Scheirer, and A. Ross, "Open set fingerprint spoof detection across novel fabrication materials," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 11, pp. 2447–2460, Nov. 2015.

[27] Y. Ding and A. Ross, "An ensemble of one-class SVMs for fingerprint spoof detection across different fabrication materials," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2016, pp. 1–6.

[28] T. Chugh and A. K. Jain, "Fingerprint presentation attack detection: Generalization and efficiency," in *Proc. Int. Conf. Biometrics (ICB)*, Jun. 2019, pp. 1–8.

[29] J. J. Engelsma and A. K. Jain, "Generalizing fingerprint spoof detector: Learning a one-class classifier," in *Proc. Int. Conf. Biometrics (ICB)*, Jun. 2019, pp. 1–8.

[30] L. J. González-Soler, M. Gomez-Barrero, L. Chang, A. Pérez-Suárez, and C. Busch, "Fingerprint presentation attack detection based on local features encoding for unknown attacks," 2019, *arXiv:1908.10163*. [Online]. Available: http://arxiv.org/abs/1908.10163

[31] R. Tolosana, M. Gomez-Barrero, C. Busch, and J. Ortega-Garcia, "Biometric presentation attack detection: Beyond the visible spectrum," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1261–1275, 2020, doi: 10.1109/TIFS.2019.2934867.

[32] Y. Zhang, D. Shi, X. Zhan, D. Cao, K. Zhu, and Z. Li, "Slim-ResCNN: A deep residual convolutional neural network for fingerprint liveness detection," *IEEE Access*, vol. 7, pp. 91476–91487, 2019.

[33] S. Schuckers and P. Johnson, "Fingerprint pore analysis for liveness detection," U.S. Patent 9 818 020, Nov. 14, 2017.

[34] E. Marasco and C. Sansone, "Combining perspiration- and morphology-based static features for fingerprint liveness detection," *Pattern Recognit. Lett.*, vol. 33, no. 9, pp. 1148–1156, Jul. 2012.

[35] Z. Xia, C. Yuan, R. Lv, X. Sun, N. N. Xiong, and Y.-Q. Shi, "A novel weber local binary descriptor for fingerprint liveness detection," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 50, no. 4, pp. 1526–1536, Apr. 2020.

[36] R. F. Nogueira, R. de Alencar Lotufo, and R. C. Machado, "Fingerprint liveness detection using convolutional neural networks," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 6, pp. 1206–1213, Jun. 2016.
[37] H.-U. Jang, H.-Y. Choi, D. Kim, J. Son, and H.-K. Lee, "Fingerprint spoof detection using contrast enhancement and convolutional neural networks," in *Proc. Int. Conf. Inf. Sci. Appl.* Singapore: Springer, 2017, pp. 331–338.
[38] T. Chugh, K. Cao, and A. K. Jain, "Fingerprint spoof detection using minutiae-based local patches," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2017, pp. 581–589.
[39] F. Pala and B. Bhanu, "Deep triplet embedding representations for liveness detection," in *Proc. Deep Learn. Biometrics. Adv. Comput. Vis. Pattern Recognit.* Springer, 2017, pp. 287–307.
[40] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[41] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015, *arXiv:1511.06434*. [Online]. Available: http://arxiv.org/abs/1511.06434
[42] J. J. Engelsma, K. Cao, and A. K. Jain, "RaspiReader: Open source fingerprint reader," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2511–2524, Oct. 2019.
[43] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu, "Sketch2photo: Internet image montage," *ACM Trans. Graph.*, vol. 28, no. 5, p. 124, 2009.
[44] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," 2015, *arXiv:1508.06576*. [Online]. Available: http://arxiv.org/abs/1508.06576
[45] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2016, pp. 694–711.
[46] X. Wang, G. Oxholm, D. Zhang, and Y.-F. Wang, "Multimodal transfer: A hierarchical deep convolutional neural network for fast artistic style transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5239–5247.
[47] C. Li and M. Wand, "Precomputed real-time texture synthesis with Markovian generative adversarial networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 702–716.
[48] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6924–6932.
[49] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1501–1510.
[50] A. Elgammal, B. Liu, M. Elhoseiny, and M. Mazzone, "CAN: Creative adversarial networks, generating 'Art' by learning about styles and deviating from style norms," 2017, *arXiv:1706.07068*. [Online]. Available: http://arxiv.org/abs/1706.07068
[51] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1125–1134.
[52] W. Xian *et al.*, "TextureGAN: Controlling deep image synthesis with texture patches," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8456–8465.
[53] V. Mura *et al.*, "LivDet 2017 fingerprint liveness detection competition 2017," in *Proc. Int. Conf. Biometrics (ICB)*, Feb. 2018, pp. 297–302.
[54] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2414–2423.
[55] V. Dumoulin, J. Shlens, and M. Kudlur, "A learned representation for artistic style," 2016, *arXiv:1610.07629*. [Online]. Available: http://arxiv.org/abs/1610.07629
[56] Y. Li, N. Wang, J. Liu, and X. Hou, "Demystifying neural style transfer," 2017, *arXiv:1701.01036*. [Online]. Available: http://arxiv.org/abs/1701.01036
[57] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861
[58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[59] K. Cao, D.-L. Nguyen, C. Tymoszek, and A. K. Jain, "End-to-end latent fingerprint search," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 880–894, 2020.
[60] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[61] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

Tarang Chugh (Graduate Student Member, IEEE) received the B.Tech. degree (Hons.) in computer science and engineering from the Indraprastha Institute of Information Technology, Delhi (IIIT-D), in 2013. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Michigan State University. He was with the IBM Research Lab, New Delhi, India, as a Research Engineer from 2013 to 2015. His research interests include biometrics, pattern recognition, and machine learning.

Anil K. Jain (Life Fellow, IEEE) is currently a University Distinguished Professor with the Department of Computer Science and Engineering, Michigan State University. His research interests include pattern recognition and biometric authentication. He was a member of the United States Defense Science Board. He is a member of the National Academy of Engineering and a Foreign Fellow of the Indian National Academy of Engineering and Chinese Academy of Sciences. He has received Fulbright, Guggenheim, Alexander von Humboldt, and IAPR King Sun Fu awards. He served as the Editor-in-Chief for the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.