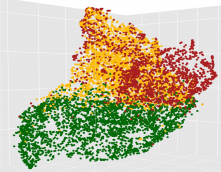
42 IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 16, 2021 Fingerprint Spoof Detector Generalization Tarang Chugh , *Graduate Student Member, IEEE*, and Anil K. Jain , *Life Fellow, IEEE*

***Abstract*— We present a style-transfer based wrapper, called** 

**Universal Material Generator (UMG), to improve the gener**

**alization 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**

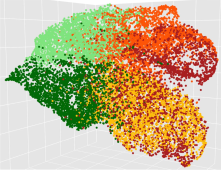
**of two state-of-the-art spoof detectors, namely Fingerprint Spoof**

**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, finger**

**print spoof buster.**

I. INTRODUCTION

**W**ITH 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

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 Engi neering, 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

1The 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. 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.

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 mul titude of fabrication processes ranging from basic *molding*

1556-6013 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

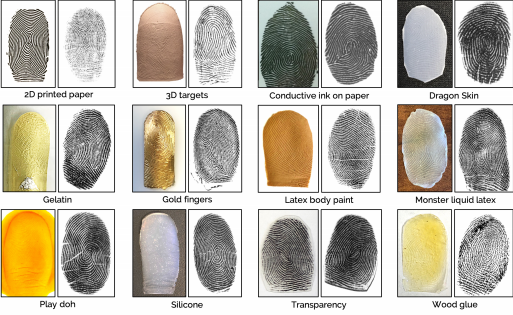
CHUGH AND JAIN: FINGERPRINT SPOOF DETECTOR GENERALIZATION 43 

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 pro posed 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

2https://www.bbc.com/news/world-latin-america-21756709 3http://statenews.com/article/2016/08/how-msu-researchers-unlocked-a fingerprint-secure-smartphone-to-help-police-with-homicide-case 4https://www.medianama.com/2018/03/223-cloned-thumb-prints-used-to spoof-biometrics-and-allow-proxies-to-answer-online-rajasthan-police-exam/ 5https://imgur.com/gallery/8aGqsSu

biometric systems6*,*.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 finger print 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 bil lion 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 read ers 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

6http://fortune.com/2016/04/07/guy-unlocked-iphone-play-doh/ 7https://srlabs.de/bites/spoofing-fingerprints/

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

44 IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 16, 2021

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) anatom ical 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 Descrip tor (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 algo rithm’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 sum mary 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 dis criminator 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 per formance 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 finger print 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

8Open-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.

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 generaliza tion performance of fingerprint spoof detectors. In particu lar, 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 *(*∼ 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 trans lation. 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 Mate rial Generator (UMG), to improve the generalization performance of any fingerprint spoof detector against spoofs made from materials not seen during training.

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

CHUGH AND JAIN: FINGERPRINT SPOOF DETECTOR GENERALIZATION 45

TABLE I

SUMMARY OF THE STUDIES PRIMARILY FOCUSED ON FINGERPRINT SPOOF GENERALIZATION

It attempts to synthesize impressions with style (texture) characteristics potentially similar to unknown spoof mate rials 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 per formance 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 sen sor. 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 detec tion 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 impres sions 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 perfor mance 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 materi als, and (iii) training a spoof detector on the augmented dataset to evaluate its performance on the “unknown” material left out

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

46 IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 16, 2021









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

although prohibitively slow, for arbitrary images. In [48], Ulyanov *et al.* proposed use of an InstanceNorm layer to nor malize feature statistics across spatial dimensions. An Instan ceNorm layer is designed to perform the following operation:

spoofness scores for individual patches. An overview of the proposed approach is presented in Fig. 3.

*I N(x)* = *γ*

*x* − *μ(x) σ (x)*

+ *β* (1)

*A. Universal Material Generator (UMG) Wrapper*

The primary goal of the UMG wrapper is to generate synthetic spoof images corresponding to unknown spoof mate rials, 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,

where, *x* is the input feature space, *μ(x)* and *σ (x)* are the mean and standard deviation parameters, respectively, com puted across spatial dimensions independently for each chan nel 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

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

CHUGH AND JAIN: FINGERPRINT SPOOF DETECTOR GENERALIZATION 47



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 *α* ∈ [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 dif ferent (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 learn able 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:

similar to [45], [56] given as:

*Ls* =  *L*

*μ(φi(g(t)))* − *μ(φi(s))* 2

*i*=1

+ *L*

*σ (φi(g(t)))* − *σ (φi(s))* 2 (4)

*i*=1

where each *φi* denotes a layer in the VGG-19 network we use as encoder. We pass *g(t)* and *s* through *f (*·*)* and extract the outputs of *relu*1\_1, *relu*2\_1, *relu*3\_1 and *relu*4\_1 layers for computing *Ls*.

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

*T (c,s, α)* = *g((*1 − *α)* · *f (c)* + *α* · *t)* (5)

where setting *α* = 0 will reconstruct the original content image and *α* = 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

*Ada I N(x, y)* = *σ (y)*

*x* − *μ(x) σ (x)*

+ *μ(y)* (2)

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

where the source (*c*) feature space is *x* and the target (*s*) feature space is *y*. In this manner, *x* is normalized with *σ (y)* and shifted by *μ(y)*. Our synthetic spoof generator *G* is composed of an encoder *f (*·*)* and a decoder *g(*·*)*. For the encoder, *f (*·*)*, 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* = *Ada I N( f (c), f (s))* (3)

We use the decoder, *g(*·*)*, to take *t* as input to produce *T (c,s)* = *g(t)* which is the final synthesized image condi tioned 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 *Ls*

content loss, *Lc*, 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.

*Lc* = *f (g(t))* − *t* 2 (6)

Doing the style transfer, simply using a content loss *(Lc)* 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

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

48 IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 16, 2021



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 (*α* = 0*.*5) of another (target) known material (first row).

looking images. The adversarial objective functions for the generator (*LGadv* ) and discriminator (*LDadv* ) are given as9:

*LGadv* = E*t*[log*(*1 − *D(G(t)))*] (7) *LDadv* = E*x* [log *D(x)*] + E*t*[log*(*1 − *D(G(t)))*] (8)

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

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

objective functions:

*GLG* = *λc* · *Lc* + *λs* · *Ls* + *LGadv* (9)

min

*DLD* = *LDadv* (10)

max

where *λc* and *λs* are the weight parameters for content loss *(Lc)* and style loss *(Ls)*, 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, *Smreal* , fabricated using a set of *m* spoof materials, we adopt a leave-one-out protocol

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

CHUGH AND JAIN: FINGERPRINT SPOOF DETECTOR GENERALIZATION 49

TABLE II

SUMMARY OF THE MSU-FPAD V2 AND LIVDET 2017 DATASETS

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 *mt h*

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 *Nsynth* 

randomly selected pairs of images {*Iima* , *Iimb* } s.t. *i* ∈

{1*,..., Nsynth*} from known but different materials *ma, mb* ∈

{*m*1*,... mk*}, *a*  = *b*, to generate a dataset of synthesized

spoof images *Sksynth*. For each synthesized image, the friction

ridge (content) information and the source material (style)

characteristics are provided by the first image, *Ima* , and the

target material (style) characteristics are provided by the sec

ond image, *Imb* . 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 *Nsynth* 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 distin guish between lives and spoofs. Figure 6 presents examples of the 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

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.

(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 competi tion [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 com bining the publicly available MSU Fingerprint Presentation Attack Dataset v1.0 (MSU-FPAD v1.0) and Precise Biometrics

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

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

50 IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 16, 2021

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.ATOTAL OF TWELVE EXPERIMENTS ARE PERFORMED WHERE THE MATERIAL LEFT-OUT FROM TRAINING IS TAKEN AS THE “UNKNOWN” MATERIAL FOR EVALUATION

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 minu tiae points, *M* = {*m*1*, m*2*,..., mk* }, where *mi* = {*xi, yi, θi*}, *i.e.* the minutiae *mi* is defined in terms of spatial coordinates

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

(*xi*, *yi*) and orientation (*θi*), a corresponding set of *k* local patches *L* = {*l*1*,l*2*,...,lk* }, each of size [96 × 96], centered and aligned using minutiae location (*xi, yi*) and orientation (*θi*), are extracted as proposed in [2].

*C. Implementation Details*

The encoder of the UMG wrapper is the first four convo lutional layers (*conv*1\_1, *conv*2\_1, *conv*3\_1, and *conv*4\_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 *λs* = 0*.*002 and *λc* = 0*.*001. We use the Adam optimizer [61] with a batch size of 8 and a learning rate of (1*e* − 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 general ization performance of two state-of-the-art spoof detectors,

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

CHUGH AND JAIN: FINGERPRINT SPOOF DETECTOR GENERALIZATION 51

**Algorithm 1** Training UMG Wrapper 

1: **procedure**

2: *input*

3: *x*: source image providing friction ridge content and

known style A

4: *y*: target image providing known style B

5: *f (*·*)*: encoder network; first 4 layers of VGG-19 network

pre-trained on ImageNet with weights frozen during

training

6: *g(*·*)*: decoder network; mirrors *f (*·*)* with pooling layers

replaced with nearest up-sampling layers

7: *D(*·*)*: discriminator function similar to [41] 8: *A(x, y)*: AdaIN operation; transfer style from *x* to *y* (using Eq. 2)

9: *α* = 0*.*5

10: *λc* = 0*.*001, *λs* = 0*.*002

11: *output*

12: *UMG(*·*)*: UMG wrapper trained on known materials 13: *begin*:

14: *Encoding*: *fx* = *f (x)* and *fy* = *f (y)*

15: *Style transfer*: *t* = *A( fx , fy )*

16: *Stylized image*: *T (c,s, α)* = *g((*1 − *α)* · *fc* + *α* · *t)* 17: *Style Loss*: *Ls* using Eq. 4

18: *Content Loss*: *Lc* using Eq. 6

19: *Adversarial Loss (generator)*: *LGadv* using Eq. 7 20: *Adversarial Loss (discriminator)*: *LDadv* using Eq. 8 21: *Objective functions for training UMG wrapper* 22: min*G LG* = *λc* · *Lc* + *λs* · *Ls* + *LGadv*

23: max*D LD* = *LDadv*

24: *end*

**

namely, Fingerprint Spoof Buster and Slim-ResCNN. We train a MobileNet-V1 [57] classifier from scratch using the aug mented dataset for Fingerprint Spoof Buster [2]. In the case of Slim-ResCNN, a custom architecture, consisting a series of optimized residual blocks [58] is implemented12 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 materi als) 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

12We 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.

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.

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 approx imately 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, *DAreal* , collected on a source fingerprint sensor, *F A*, containing real live, *L Areal* , and real spoof *SAreal* datasets, s.t. *DAreal* = {*L Areal* ∪ *SAreal*}, the pro posed UMG wrapper is used to generate 50*,* 000 synthetic live

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

52 IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 16, 2021

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 CLASS IFICATION ACCURACY (ACA) AND TDR @ FDR = 1.0%

TABLE V

CROSS-SENSOR FINGERPRINT SPOOF GENERALIZATION PERFORMANCE ON LIVDET 2017 DATASET

IN TERMS OF AVERAGE CLASS IFICATION ACCURACY AND TDR @ FDR = 1.0%

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 per

formance 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

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 Bsynth*, and 50*,* 000 synthetic spoof patches, *SBsynth*, for a target sensor, *FB*. The UMG wrapper is trained only on the live images collected on *SB*, and used for style transfer on *L Areal* and *SAreal* to generate *L Bsynth*, and *SBsynth*, 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

13An average of ∼ 3100 local patches are extracted from 100 live fingerprint images in LivDet 2017 experiments.

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.

*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.

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

CHUGH AND JAIN: FINGERPRINT SPOOF DETECTOR GENERALIZATION 53 Fig. 10. 3D t-SNE visualization of feature embeddings of real live

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.

*Online Testing stage*: There is no increase in the spoof detec tion 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 finger prints, 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 patches15 using cross-material style transfer between spoof fingerprints of silicone and latex

14Not 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.

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

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.

body paint. Fingerprint Spoof Buster, fine-tuned using the syn thesized 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 mate rials, 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 gen eralizable spoof detectors. To address that, we propose a style-transfer based wrapper, Universal Material Generator (UMG), to improve the generalization performance of any

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

54 IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 16, 2021

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 gener alization 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 mate rials. 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 represent ing 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 govern mental 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 fin gerprints,” 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 fab rication 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 com petition 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 detec tion 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 finger print 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 imag ing,” 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 detec tion,” 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.

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.

CHUGH AND JAIN: FINGERPRINT SPOOF DETECTOR GENERALIZATION 55

[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 algo rithm 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 devi ating 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 Recog nit.*, 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 representa tion 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 sci

ence 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 Depart

ment of Computer Science and Engineering, Michi

gan 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 ANALYS IS AND MACHINE INTELLIGENCE.

Authorized licensed use limited to: Miami University Libraries. Downloaded on October 21,2020 at 23:52:23 UTC from IEEE Xplore. Restrictions apply.