Does Capture Background Influence the Accuracy of the Deep Learning based Fingerphoto Presentation Attack Detection Techniques?

Hailin Li Raghavendra Ramachandra
Norwegian University of Science and Technology (NTNU), Norway
E-mail: {hailin.li, raghavendra.ramachandra} @ ntnu.no

Abstract

The rapid evolution of modern smartphone techniques has made biometric authentication applications feasible using smartphone cameras. FingerPhoto verification offers the benefits of scalability, reliability, and user convenience. Similar to traditional contact-based fingerprint verification methods, the widespread deployment of fingerphoto authentication applications has raised concerns regarding the system being attacked (or spoofed). In this work, we not only study and discuss the generalizability of eight different pre-trained deep learning models against unseen attacks but also present an analysis of how the background of the captured fingerphoto and attack samples will affect the Presentation Attack Detection (PAD) performance. To experimentally benchmark the PAD performance with different types of background extractors, we present three different studies: full background, segmenting only the background, and extracting the Region Of Interest (ROI) that pertains to the fingerphoto region. We present an extensive evaluation of three different types of background extraction methods using eight different pre-trained deep learning techniques. The obtained results on the publicly available fingerphoto datasets indicate that by removing the background noise or extracting the ROI regions, the deep learning models will become more reliable for fingerphoto presentation attack detection.

1. Introduction

Contactless biometrics on smartphones are becoming increasingly popular because of their usability and are highly recommended owing to the pandemic. Contactless biometrics can be captured using smartphones with built-in high-resolution cameras that capture sufficient information to perform reliable user authentication. Among several biometrics, the contactless capture of fingerprints is well addressed, which results in reliable and robust verification performance. Because the fingerprint image is captured and processed using built-in camera software before processing and feature extraction for authentication, it is commonly referred to as a fingerphoto in the literature. The popularity of fingerphoto authentication has resulted in several algorithms and commercial solutions that can be employed in access-control applications. However, the success and popularity of fingerphoto authentication applications has attracted the attention of attackers. Presentation attacks are proven to be vulnerable to verification systems using different Presentation Attack Instruments (PAIs). Common PAIs include replay attacks, 2D/3D printed paper or fingerprint replicas using various materials.

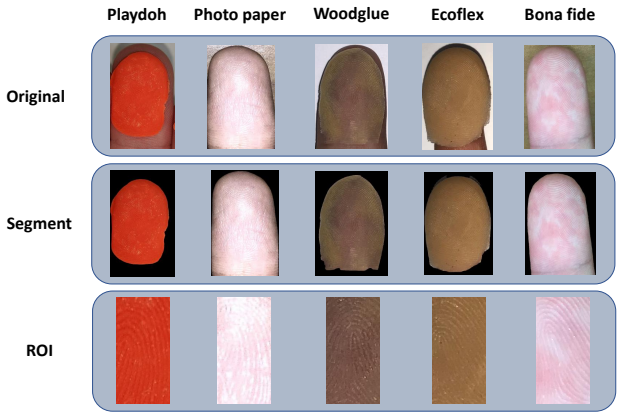

Figure 1. Illustration of fingerphoto captured using smartphone with different types of finger region segmentation.

Fingerphoto presentation attack detection techniques have been widely studied in the literature and can be broadly classified as [] hand-crafted features, deep learning, and hybrid methods. Early handcrafted methods include features such as micro-textures, gradients, and light reflections. The commonly used handcrafted feature extraction techniques include LBP [3], BSIF [9] and [14] which are further classified as machine-learning algorithms, particularly SVM [5] as a classifier. Deep learning methods include the use of

Authors	Finger region segmentation	Type of finger region segmentation	Database
Taneja et al. [20]	Yes	ROI	Public
Zhang et.al [22]	No	Entire image	Private
Fujio et.al [2]	No	Entire image	Private
Wasnik et.al [21]	Yes	ROI	Private
Marasco et.al [15]	Yes	ROI	Public
Marasco et.al [16]	Yes	Image crop	Public
Purnapatra et. al [17]	No	Entire image	Public
Li et. al [12]	Yes	ROI	Public
This Work	**Yes**	**Different types of finger segmentation**	**Public**

Table 1. Existing smartphone based fingerphoto PAD methods that are based on using different types of finger region segmentation.

pre-trained networks that are either trained end-to-end [17] or fine-tuned [15] to detect the fingerphoto PAD. Almost all popular pre-trained CNNs and visual transformers are fine-tuned to reliably detect finger photo PAD. An extensive evaluation of different pre-trained deep learning networks, including visual transformers, is presented in [12], which indicates the improved performance of deep learning in detecting finger photo PAD. Hybrid methods [22] include the combination of deep learning and handcrafted features that are combined at either the feature or the score level. For a detailed study on different fingerphoto PAD, readers can refer to the survey paper [13].

Although fingerphoto PAD techniques have been widely studied, there is no uniformity in how fingerphoto images are used to train the detection system. Table 1 lists the different ways in which fingerprint image segmentation is used in detection systems. As noted in the literature, fingerphoto images are represented in three ways: (a) using the whole image as it is captured with background, (b) segmenting only the finger region in the image while the background is masked to have black pixels, and (c) the Region of Interest (ROI) that is tightly cropped to have only the finger region. Figure 1 shows an example of three different types of fingerphoto sample representations used in existing studies, which can directly affect the performance of the detection systems. Furthermore, with the presentation attack samples, fingerprint replicas can only cover a partial region in the fingerprint, which can directly influence the performance of the detection system. Therefore, in this work, we investigate the role of the ROI that can influence the performance of fingerphoto PAD techniques. To effectively benchmark the influence of finger region extraction techniques, we employed eight different pre-trained deep learning-based fingerphoto PAD techniques by considering the higher performance of handcrafted PAD techniques [12]. The following research questions are proposed in this work:

- **RQ1:** Does the fingerphoto background influence the detection performance using pre-trained deep features based fingerphoto PAD?
- **RQ2:** Does the background influence the detection performance of pre-trained deep features based fingerphoto PAD on individual PAI?
- **RQ3:** What type of region segmentation indication the best performance on the pre-trained deep features based fingerphoto PAD?

The contributions of this work are summarized below in the course of addressing these research questions:

- To the best of our knowledge, this is the first work that presents a comprehensive study of different types of fingerphoto region segmentation methods adapted in the literature.
- Benchmark the quantitative performance of the eight pre-trained fingerphoto PAD networks on three different types of fingerphoto segmentation methods.
- Extensive experiments were conducted on a publicly available dataset using four different evaluation protocols.

The rest of the paper is organized as follows: Section 2 describes the fingerphoto presentation attack detection framework. Section 3 discusses the evaluation protocol and the obtained results. Finally, we discuss the conclusion in Section 4.

2. Fingerphoto Presentation Attack Detection

The generic representation of the Fingerphoto presentation attack detection module is as shown in the Figure 1 that has three different functional blocks namely (a) fingerphoto sample post-processing (b) feature extraction and (c) classification that are discussed as follows:

2.1. Post-processing

The goal of post-processing is to process the captured fingerphoto image to best present the captured image before performing feature extraction. Commonly performed post-processing tasks include the extraction of interest by

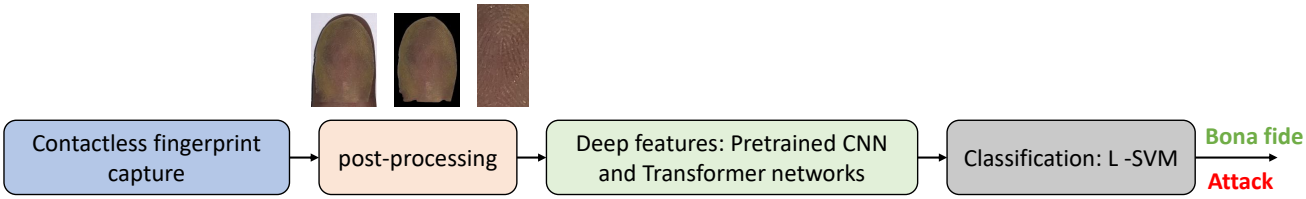

Figure 2. Block diagram of the deep features based contactless fingerprint PAD illustrating different types of finger segmentation techniques as the post-processing that are evaluate in this work.

segmenting out the unwanted background and enhancing the interest region of interest. In this study, we employed and evaluated three different cases of post-processing that are widely employed in existing works.

- **With background image**: In this case, fingerphoto images are not post-processed to extract the interest region. Therefore, the fingerphoto images are exhibited to have background information that is used to perform feature extraction and classification.
- **Background removal**: In order to remove the background and potential intersection region between the fingertip and covered material of the attack samples. It is difficult to batch-crop the images. However, the proposed Segment Anything Model (SAM) [10] tool can produce high-quality object masks for all objects in an image. In our work, we utilized the SAM to generate the mask of only the presentation attack region. Based on the generated masks, the presentation attack region was extracted from the original images, and background noise was removed.
- **ROI extraction**: Through the observation of the dataset, the central region is of the samples is the critical area. Hence, we obtained the Region Of Interest (ROI) by locating the center point of the image and then cropping the region of 128 × 256 pixels around the center point.

2.2. Deep feature extraction

In the next step, we employ eight different pre-trained deep neural networks to extract features from the fingerphoto image. The models are all pre-trained on ImageNet data set. We selected these eight techniques based on their good detection performance as reported in [12].

- **AlexNet [11]**: is a powerful model capable of achieving high accuracy on very challenging datasets which won the LSVRC competition in 2012. AlexNet comprises five convolutional layers, followed by three fully connected layers. In our experiment, we took the feature map after the first fully connected layer(fc6) and obtained a feature vector of size 4096.
- **GoogLeNet [18]**: is another classic deep nerual network with 22 layers deep. GoogLeNet introduced the idea of *multi-branch convolutions*, termed inception modules, which are designed to deal with the over-parameterization problem. We utilize the network to obtain the feature vector from the global average pooling layer.
- **ResNet50 [4]**: stands for the Residual Network that consists of 50 layers. The core idea of ResNet is to tackle the gradient vanishing problem, which makes it possible to train an ultra-deep neural network and still achieve excellent performance. We utilized the ResNet50 architecture to extract features from the global average pooling layer with a size of 2048.
- **DenseNet201 [7]**: is a network that connects each layer to every other layer in a feed-forward fashion with fewer parameters and high accuracy compared to ResNet. The core idea of DenseNet is that convolutional networks can be optimized when they have shorter connections between layers close to the input and those close to the output. We utilized DenseNet201 to extract features with a size of 1920 from the global average pooling layer.
- **MobileNetV2 [6]**: is a lightweight network which is originally designed to perform well on mobile devices. Depthwise Separable Convolutions, Linear Bottlenecks and Inverted residuals are introduced to achieve the efficient CNN. We obtain a size of 1280 feature vector from the global average pooling layer.
- **EffiecientNetb0 [19]**: introduced the idea of compound coefficient that uniformly scales three essential dimensions depth/width/resolution. We utilize the b0 network of the EffiecientNet network and obtain the feature vector from the global average pooling layer of size 1280.
- **NasNet [23]**: frames the problem of finding the best CNN architecture as a Reinforcement Learning problem. The model searches for the best combination of

Data tpye	No.Samples			
	IPhone7	IPhoneX	Samsung	Total
Bonafide	858	691	4336	5886
PAI: Ecoflex	832	0	416	1248
PAI: Photo paper	832	272	0	1104
PAI: Playdoh	0	0	1623	1623
PAI: Woodglue	0	272	0	272

Table 2. The statistics of the Presentation Attack Instruments (PAIs) regarding the number of samples and capture devices.

Case	Training set		Testing set	
	Bona fide	Attack	Bona fide	Attack
Case-1	2999	2999	2887	1248
Case-2	3143	3143	2743	1104
Case-3	2624	2624	3262	1623
Case-4	3975	3975	1911	272

Table 3. The statistics of the training samples and testing samples corresponding to the different performance evaluation protocols.

parameters in the given search space. A new regularization technique called ScheduledDropPath is also proposed, which significantly improves the generalization of NASNet models. We utilized the pretrained NasNet weight using the matlab deep learning toolbox. The feature vector was extracted from the global average pooling layer with a size of 1056.

- **Vision Transformer(ViT) [1]**: is a transformer model designed for computer vision task which split the input image into a sequence of fixed-size non-overlapping patches. Then, tokenization is applied before applying the tokens to a standard transformer architecture. We utilize the Hugging Face version of ViT pre-trained model with a patch resolution of 32 $\times$ 32 pre-trained in ImageNet.

2.3. Feature classification

Finally, the features extracted from the pretrained deep learning networks are classified using a linear Support Vector Machine (SVM) model. A training set corresponding to the bona fide and attack samples was used to train the classifier. Given the test sample, the trained SVM model can output a prediction score.

3. Experiments and Results

In this section, we present the quantitative results of three different background segmentations that are widely used in the existing literature on fingerphoto PAD. The experimental results are presented using a publicly available dataset with four different PAIs [17], including coflex, playdoh, photo paper, and woodglue, which are captured using iPhone7, iPhone X and Samsung Galaxy S20. Table 2 lists the statistics of the fingerphoto presentation attack dataset employed in this study. The public dataset employed in this work has 5886 bona fide and 4247 attack samples that were collected using three different smartphones.

3.1. Performance evaluation protocol

In this work, we employed the evaluation protocol proposed in [12], which follows the leave-one-out PAI to benchmark the PAD technique performance to the unseen PAI. Therefore, the performance evaluation protocol will benchmark the genralisabality of the PAD techniques to the un-seen attacks together with the role of different types of background extraction methods. Table 3 shows the statistics of the training and testing samples corresponding to the four different Cases resulting from the leave-one-out PAI evaluation considered in this study. The **Case-1:** photo paper, playdoh, and woodglue are used for training; Ecoflex is used for testing. **Case-2:** Ecoflex, playdoh, and woodglue are used for training; photo paper is used for testing. **Case-3:** Ecoflex, photo paper, and woodglue are used for training; playdoh is used for testing. **Case-4:** ecoflex, photo paper, and playdoh are used for training; and woodglue is used for testing. It should be noted that for each individual case, the bona fide samples used for training remained the same as the attack samples.

The experimental results were obtained using ISO/IEC 30107- 3 [8]. The Attack Presentation Classification Error Rate (APCER) indicates the percentage ratio at which the presentation attack examples are misidentified as bona fide examples. The bona fide Presentation Classification Error Rate (BPCER) indicates the percentage ratio at which bona fide examples were misidentified as presentation attack examples. Furthermore, we included the Detection Equal Error Rate to measure the effectiveness of the detection system. A lower D-EER indicates better generalization of the deep feature algorithm towards unseen PAI.

3.2. Results and inference

Table 4, 5 and 6 shows the quantitative results with no finger region segmentation, finger region segmentation with black background and ROI corresponding to finger region respectively. In the following, we discuss the quantitative results corresponding to individual PAD techniques and different finger region segmentation methods.

- **AlexNet [11]**: The segmented experiment achieves the best EER performance than the others on Case-1 and Case-4 with EER = 6.00% and 3.96% respectively. The ROI experiment yielded the best result for Case 2. However, none of the approaches improved the results in Case-3, indicating the limitations of generalizability.

	PAD Algorithms	D-EER	BPCER @ APCER =			PAD Algorithms	D-EER	BPCER @ APCER =	
			5%	10%				5%	10%
Case-1	AlexNet	7.37	9.32	5.44	Case-2	AlexNet	39.93	88.22	78.89
	GoogleNet	12.11	28.09	15.48		GoogleNet	50.00	98.76	98.76
	DenseNet201	5.37	5.85	1.97		DenseNet201	42.84	93.95	88.74
	ResNet50	7.45	12.43	5.02		ResNet50	40.40	95.81	89.02
	EfficientNet-B0	6.24	8.90	3.29		EfficientNet-B0	38.50	87.60	78.82
	NasNet	8.58	14.93	7.48		NasNet	50.0	71.09	58.22
	MobileNet-V2	4.33	4.05	1.73		MobileNet-V2	28.28	68.50	54.47
	ViT	4.50	3.81	1.56		ViT	32.34	81.04	68.32
Case-3	AlexNet	50.00	100.00	99.72	**Case-4**	AlexNet	6.16	6.33	3.40
	GoogleNet	50.00	100.00	99.88		GoogleNet	3.70	2.56	1.62
	DenseNet201	40.41	94.02	87.43		DenseNet201	4.35	2.20	0.58
	ResNet50	15.40	36.48	24.13		ResNet50	0.71	0.10	0.05
	EfficientNet-B0	50.00	97.62	94.91		EfficientNet-B0	0.37	0	0
	NasNet	6.84	9.93	3.92		NasNet	11.32	17.74	12.09
	MobileNet-V2	38.26	74.65	67.69		MobileNet-V2	2.12	0.58	0.31
	ViT	8.44	13.03	3.62		ViT	1.18	0.37	0.16

Table 4. Quantitative performance of the deep features for contactless fingerprint PAD using With background image.

	PAD Algorithms	D-EER	BPCER @ APCER =			PAD Algorithms	D-EER	BPCER @ APCER =	
			5%	10%				5%	10%
Case-1	AlexNet	6.00	7.24	2.94	Case-2	AlexNet	42.76	93.26	85.24
	GoogleNet	13.06	29.86	16.04		GoogleNet	42.86	92.45	84.18
	DenseNet201	4.64	4.43	1.70		DenseNet201	40.21	94.86	89.83
	ResNet50	10.66	23.21	11.43		ResNet50	32.70	81.70	70.43
	EfficientNet-B0	2.64	1.39	0.62		EfficientNet-B0	22.10	56.95	40.54
	NasNet	16.01	40.84	25.25		NasNet	35.53	83.30	70.51
	MobileNet-V2	11.78	20.30	13.54		MobileNet-V2	36.78	91.14	80.53
	ViT	7.45	10.60	5.61		ViT	32.60	78.38	67.92
Case-3	AlexNet	50.00	100	100	**Case-4**	AlexNet	3.96	1.57	0.05
	GoogleNet	50.00	100	100		GoogleNet	6.97	9.58	2.72
	DenseNet201	18.43	57.30	34.43		DenseNet201	3.67	1.83	0.21
	ResNet50	28.27	83.26	68.91		ResNet50	3.67	0.99	0.10
	EfficientNet-B0	7.78	11.28	5.70		EfficientNet-B0	2.57	0.26	0.10
	NasNet	8.74	17.08	7.30		NasNet	14.39	52.54	31.97
	MobileNet-V2	29.27	75.02	59.29		MobileNet-V2	5.87	6.23	1.83
	ViT	9.30	14.29	8.95		ViT	3.67	0.05	0.58

Table 5. Quantitative performance of the deep features for contactless fingerprint PAD using background removal images.

- **GoogLeNet [18]**: The GoogleNet features obtained the best detection performance in the ROI experiment at Case-1 and Case-2 with EER = 11.54% and 23.63%.

- **DenseNet201 [7]**: The DenseNet201 features experiments have consistent result on Case-1, Case-3 and Case-4. The segmented experiment indicated the best EER value compared with the other experiments. Especially in case 3, the segmented experiment obtained EER= 18.43% compared to 40.41% and 23.19 % of the original samples and ROI samples.

- **ResNet50 [4]**: The ROI features achieve the best performance at EER = 6.65%, 14.58% and 7.58% on Case-1, Case-2 and Case-3 respectively. This also indicates a promising result in case 4 with EER = 2.94%, which is slightly worse than that of the original experiments.

- **EffiecientNetb0 [19]**: In the segmented experiment, efficientNetb0 indicates the best performance in Case-1, Case-2, Case-3 and also achieves EER =2.57% which is slightly worse than ROI experiments.

- **NasNet [23]**: The ROI experiment achieves the best EER performance than the others on Case-2, Case-3 and Case-4 with EER= 30.16%, 4.81% and 8.55 % compared to others.

	PAD Algorithms	D-EER	BPCER @ APCER =			PAD Algorithms	D-EER	BPCER @ APCER =	
			5%	10%				5%	10%
Case-1	AlexNet	6.91	11.67	3.78	Case-2	AlexNet	25.43	61.94	47.98
	GoogleNet	11.54	22.24	13.47		GoogleNet	23.63	59.28	43.86
	DenseNet201	6.97	10.22	4.43		DenseNet201	23.19	63.18	50.20
	ResNet50	6.65	8.28	3.50		ResNet50	14.58	30.22	20.85
	EfficientNet-B0	7.21	11.15	4.47		EfficientNet-B0	14.58	34.05	21.47
	NasNet	13.30	29.62	17.73		NasNet	30.16	83.59	69.01
	MobileNet-V2	12.00	29.10	15.14		MobileNet-V2	16.22	49.40	28.00
	ViT	6.71	8.17	3.74		ViT	50.00	100.00	100.00
Case-3	AlexNet	50.00	97.03	93.99	**Case-4**	AlexNet	10.30	18.84	10.31
	GoogleNet	50.00	96.93	93.01		GoogleNet	4.43	4.29	1.88
	DenseNet201	48.42	96.01	91.94		DenseNet201	4.40	3.98	1.47
	ResNet50	7.58	11.56	5.49		ResNet50	2.94	0.26	0.05
	EfficientNet-B0	25.58	72.78	55.06		EfficientNet-B0	3.30	1.10	2.51
	NasNet	4.81	4.66	1.75		NasNet	8.55	13.24	6.54
	MobileNet-V2	50.00	98.80	96.51		MobileNet-V2	6.6	7.85	4.08
	ViT	23.65	100.00	100.00		ViT	2.57	0.99	0.31

Table 6. Quantitative performance of the deep features for contactless fingerprint PAD using ROI images as presented in the state-of-the-art [12]. Quantitative values presented in this table are taken from [12].

- **MobileNetV2 [6]**: The original features extracted by MobileNetV2 obtained the best result on Case-1 and Case-4 with EER = 4.33% and 2.12%. The ROI experiment indicated the best results for Case-2, and the segmented experiment indicated the best results for Case-3.
- **Vision Transformer(ViT) [1]**: Vision transformer model improves the detection rate in the original experiment compared to others in all four cases with EER = 4.50%, 32.34%, 8.44% and 1.18%.

To better visualize the comparison among different deep learning models or schemes. We performed an analysis using T-distributed Stochastic Neighbor Embedding (T-SNE) as a nonlinear dimensionality reduction technique to embed high-dimensional data for visualization in a low-dimensional space. Specifically, similar objects are modeled using nearby points, and dissimilar objects are modeled using distant points with a high probability. We utilized T-SNE to project the high-dimensional features extracted from the deep learning models into a two-dimensional map to compare three different preprocessing strategies. As shown in Figure 3, the features are extracted from the last pooling layer of ResNet50; the red dots refer to the attack features, and the blue dots indicate the bona fide. In the T-SNE plot of the ROI experiment, the two classes of objects are more likely to be distant, while the projection of similar objects is assigned a higher probability and dissimilar points are assigned a lower probability. In the middle figure, the few blue dots fall into the red area, demonstrating less similarity between the project-of-attack and bona fide segmented samples. The result corresponds to the EER value, in which the ROI experiment of ResNet50 achieves 6.65% in Case-1, the original experiment obtains 7.45%, and the segment experiment achieves 10.66%.

Furthermore, we have included the box charts illustrated in Figure 4 and 5. From these two figures, we can observe that in both experiments, AlexNet and GoogleNet performed satisfactorily. In both experiments, the average EER was greater than 20. Additionally, in the original sample evaluation, the vision transformer model achieved the best detection performance compared to the other models. EfficientNetb0 is considered to perform better than the other models in the segmented sample evaluation.

The following observations were made regarding all the experimental results demonstrated above:

- The fingerphoto presentation attack detection algorithm performance will be affected by the background of the captured fingerphoto samples.
- Different deep learning models perform differently with different processing schemes.
- The original experiments had the worst overall detection result compared to ROI and segmented results. However, the vision transformer performed best on the original samples with the background.
- As an unseen attack, the replica made with ecoflex and woodgule are easier to detect then photopaper or play-doh among all the models and processing techniques.
- On average, utilizing efficientNetb0 can obtain EER = 8.77% indicates the best performance within the segmented samples experiment, which is slightly worse

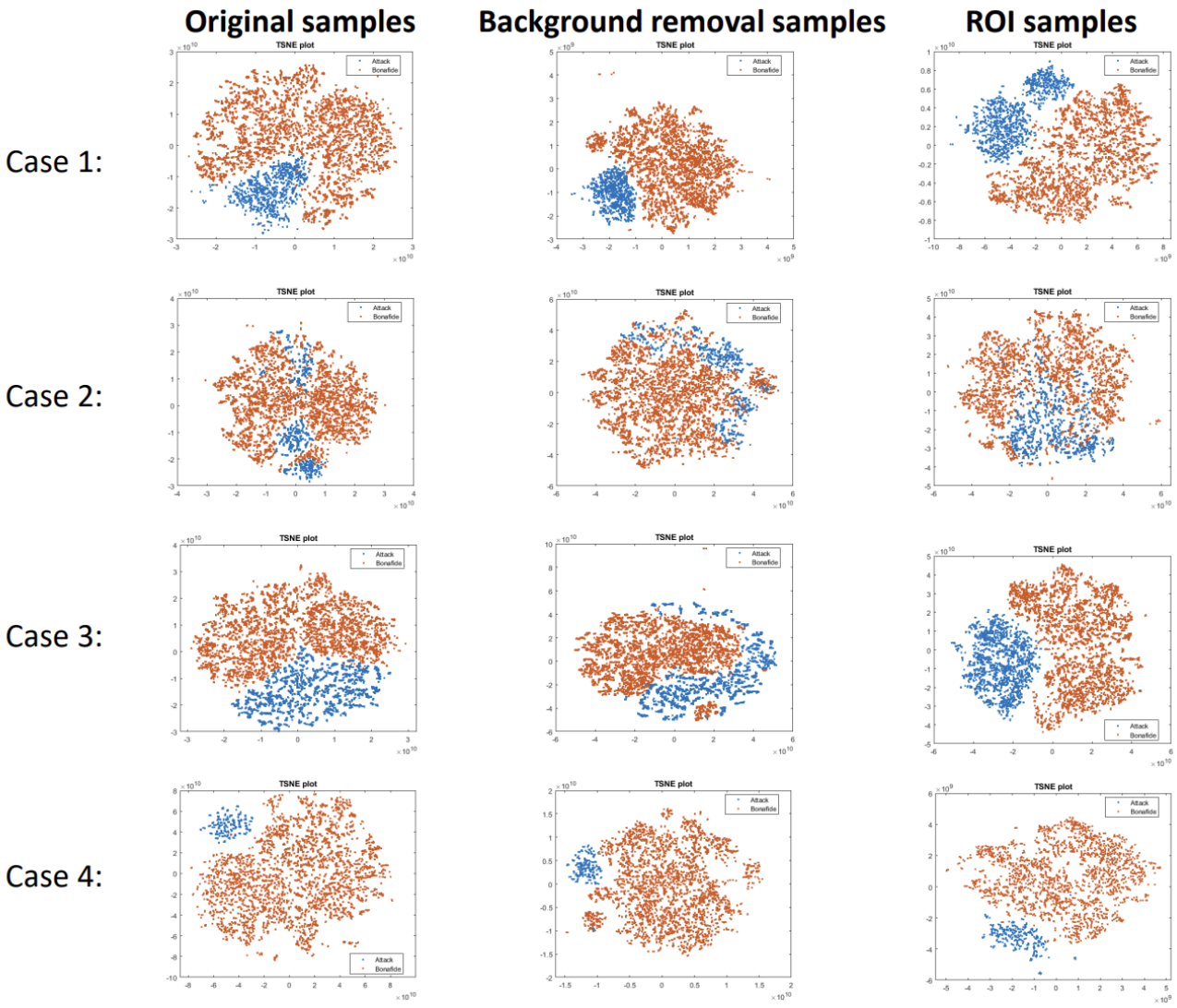

Figure 3. TSNE plot of the ResNet50 features, the red dots refer to the attack features and the blue dots indicates the bona fide. From left to right, the figure demonstrates the TSNE plot of original, background removal and ROI samples.

than EER = 8.26% achieved by ResNet50 using ROI scheme.

3.3. Discussion

Based on the observations from the above experiments and the results obtained, the research questions formulated in Section 1 are answered below.

- **Q1**. Does the fingerphoto background influence the detection performance using pre-trained deep features based fingerphoto PAD?
  - According to Table 4, 5 and 6, the obtained results indicate the different detection performance using original or background removal images. Meanwhile, the fingerphoto background is a factor that can influence the detection performance.
- **Q2**. Does the background influence the detection performance of pre-trained deep features based fingerphoto PAD on individual PAI?
  - By averaging the D-EER value of eight different models regarding each case. The original samples were observed to exhibit the best detection performance against Ecoflex and woodglue. The ROI samples obtained the best detection performance against the photo paper and playdoh.
- **Q3:** What type of region segmentation indicates the best performance on the pre-trained deep features based fingerphoto PAD?
  - Among all segmentation strategies consider all cases

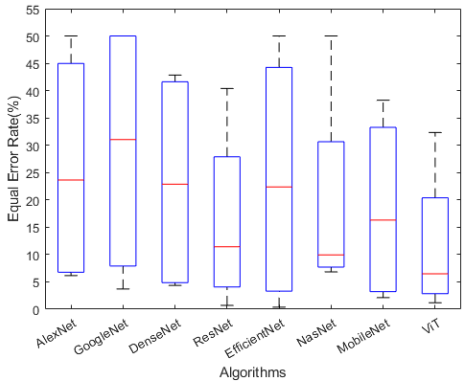

Figure 4. Box plot distribution indicating the average detection performance of *with background images*.

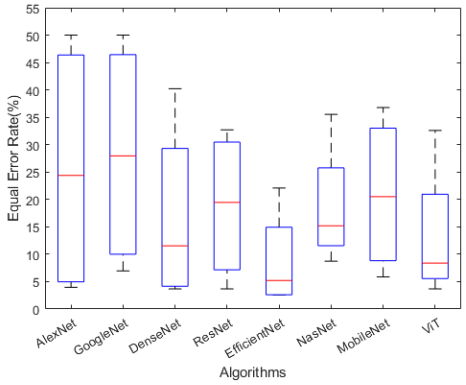

Figure 5. Box plot distribution indicating the average detection performance of *with background removal images*.

and deep feature extraction methods. The ROI extraction approach obtained the best detection performance with an average EER = 17.88%, the background removal approach achieved an average EER = 18.89%, and the original samples without any segmentation methods obtained an average EER = 20.86%.

4. Conclusion

Smartphone biometrics have become increasingly popular because of their high usability and reliable user verification. FingerPhoto verification has already been deployed in many smartphone authentication applications. Considering the increase in the number of attackers, fingerphoto presentation attack detection has become a new research topic. In this work, we continue to explore the generalization of fingerphoto attack detection models toward unseen attacks using the latest publicly available fingerphoto presentation attack dataset. We benchmarked eight different pretrained

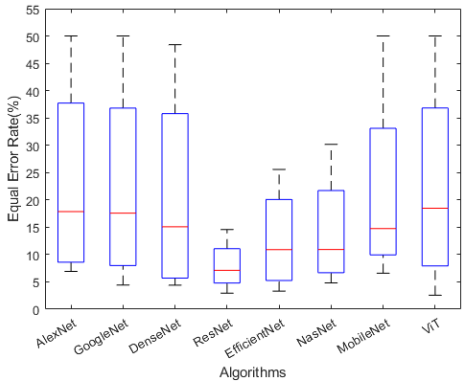

Figure 6. Box plot distribution indicating the average detection performance of *ROI images*.

deep learning models using the leave-one-out evaluation protocol. In addition, we indicated that the fingerphoto background significantly affects the detection performance to a large extent. By comparing the obtained APCER, BPCER, and D-EER using three different fingerphoto processing procedures, the obtained results indicate that the best average EER of 8.26% was achieved by the ROI experiment using ResNet50. An average EER of 8.77% was achieved by a segment experiment utilizing EfficientNetb0. Finally, the vision transformer obtains the best result in the original image with an EER of 11.61%.

Acknowledgment

This work is carried out under OFFPAD project funded by the Research Council of Norway (Project No. 321619).

References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 4, 6

[2] Masakazu Fujio, Yosuke Kaga, Takao Murakami, Tetsushi Ohki, and Kenta Takahashi. Face/fingerphoto spoof detection under noisy conditions by using deep convolutional neural network. In *BIOSIGNALS*, pages 54–62, 2018. 2

[3] Zhenhua Guo, Lei Zhang, and David Zhang. A completed modeling of local binary pattern operator for texture classification. *IEEE transactions on image processing*, 19(6):1657–1663, 2010. 1

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3, 5

[5] Marti A. Hearst, Susan T Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. Support vector machines. *IEEE Intelligent Systems and their applications*, 13(4):18–28, 1998. 1

[6] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 3, 6

[7] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 3, 5

[8] ISO/IEC JTC1 SC37 Biometrics. *ISO/IEC 30107-3. Information Technology - Biometric presentation attack detection - Part 3: Testing and Reporting*. International Organization for Standardization, 2017. 4

[9] Juho Kannala and Esa Rahtu. Bsif: Binarized statistical image features. In *Proceedings of the 21st international conference on pattern recognition (ICPR2012)*, pages 1363–1366. IEEE, 2012. 1

[10] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 3

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. 3, 4

[12] Hailin Li and Raghavendra Ramachandra. Deep features for contactless fingerprint presentation attack detection: Can they be generalized? In *The 8th National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG) 21-23 July- 2023, Jodhpur, India*, pages 1–246. Springer, 2023. 2, 3, 4, 6

[13] Hailin Li and Raghavendra Ramachandra. Deep learning based fingerprint presentation attack detection: A comprehensive survey. *arXiv preprint arXiv:2305.17522*, 2023. 2

[14] D.G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157 vol.2, 1999. 1

[15] Emanuela Marasco and Anudeep Vurity. Fingerphoto presentation attack detection: Generalization in smartphones. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 4518–4523. IEEE, 2021. 2

[16] Emanuela Marasco, Anudeep Vurity, and Asem Otham. Deep color spaces for fingerphoto presentation attack detection in mobile devices. In *International Conference on Computer Vision and Image Processing*, pages 351–362. Springer, 2022. 2

[17] Sandip Purnapatra, Conor Miller-Lynch, Stephen Miner, Yu Liu, Keivan Bahmani, Soumyabrata Dey, and Stephanie Schuckers. Presentation attack detection with advanced cnn models for noncontact-based fingerprint systems. *arXiv preprint arXiv:2303.05459*, 2023. 2, 4

[18] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 3, 5

[19] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019. 3, 5

[20] Archit Taneja, Aakriti Tayal, Aakarsh Malhorta, Anush Sankaran, Mayank Vatsa, and Rieha Singh. Fingerphoto spoofing in mobile devices: a preliminary study. In *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–7. IEEE, 2016. 2

[21] Pankaj Wasnik, Raghavendra Ramachandra, Kiran Raja, and Christoph Busch. Presentation attack detection for smartphone based fingerphoto recognition using second order local structures. In *2018 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pages 241–246. IEEE, 2018. 2

[22] Yongliang Zhang, Bing Zhou, Hongtao Wu, and Conglin Wen. 2d fake fingerprint detection based on improved cnn and local descriptors for smart phone. In *Chinese Conference on Biometric Recognition*, pages 655–662. Springer, 2016. 2

[23] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710, 2018. 3, 5