A. Brömme, C. Busch, N. Damer, A. Dantcheva, M. Gomez-Barrero, K. Raja,
C. Rathgeb, A. Sequeira, and A. Uhl (Eds.): BIOSIG 2021,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2021

Fingerprint Presentation Attack Detection using Transferred Knowledge of Common Convolutional Neural Networks

Jannik Schleicher[1]

Abstract: Convolutional Neural Networks can be repurposed from their original use case to deliver impressive fingerprint presentation attack-detection accuracy. This paper explores various options of publicly available machine learning algorithms and transfers their image classifications to differentiate between bona fide and artificial fingerprints. All networks have their liveliness detection potential evaluated without intrusive modification and after minimal training.

Datasets from the Liveness Detection Competition 2017 were used as training and validation input to give context to the research and compare the results to specialized solutions.

Keywords: Convolutional Neural Networks, Fingerprint Presentation Attack Detection

Related Work: This paper references the LivDet 2017 Fingerprint Liveness Detection Competition and the resulting paper with the same name "LivDet 2017 Fingerprint Liveness Detection Competition2017" (Mura et. al.) [2]. Ratios and thresholds are set with the paper in mind, as the same dataset is partially used for this experiment. Performances can therefore be compared to the submitted algorithms of this conference.

1 Introduction

Among all features of the human body, many may be employed to authenticate and authorize people by making use of the incredible natural diversity in human appearances. Fingerprints in particular, stand out as one of the most reliable methods to identify individuals. Increased availability, compactness and convenience of modern digital capture systems have long surpassed analog methods in speed for everyday usage.

With all these benefits, some problems, such as unsupervised incorrect authorizations, can have catastrophic outcomes. The human skin hones many imperfections and, over time, rarely stays in the original condition it was captured in. As a result, some tolerance is needed between the original and a recent fingerprint capture to prevent access denials in situations where fingers may be wet, dirty, or

[1] stjischl@h-da.de

damaged. Advancements in fingerprint pattern replacements have resulted in them being almost indistinguishable from bona fide fingerprints when the aforementioned tolerance is considered. Additional protection against artificially constructed fingerprint images is needed in the form of liveliness detection systems which determine whether a presented fingerprint originates from a live human or not.

Malicious intent is often connected to high-value targets like critical infrastructure or border control systems, and successful intrusions can lead to severe consequences. These authentication systems are operating with high accuracy and no tolerance for errors, which requires specialized and costly hardware. High-quality fingerprint scanners are challenging to integrate into systems such as smartphones and personal computers, nonetheless representing targets for unauthorized access.

Restricted space and aggressive cost-optimization create the need for small capture devices which are easy to use, easy to integrate, and cheap to produce. The reduction in capture device quality naturally comes with a reduction in authentication accuracy. Software-enhanced authentication systems can deliver impressive results while not requiring complex capture devices.

1.1 Neural Networks

Open-Source libraries such as Keras offer simple interfaces for complicated software frameworks and provide ready-to-use machine learning implementations. The following experiments were conducted using a selection of pre-trained deep learning models from Keras using TensorFlow as the underlying platform.

Spacial diversity was a category for selecting the algorithms to give insight into how complicated deep learning networks need to be to provide confident decisions on whether a presented fingerprint image is coming from a live person or not. It is important to note that these networks are image classifiers coarsely detecting the image's contents. The classifier MobileNet, for example, can detect real-world objects and animals and is intended for "mobile and embedded vision applications" [1]. All networks are pre-trained on the ImageNet dataset containing "1000 object classes and [...] 1,281,167 training images" [3].

Network Name	Size (Mb)	Parameter Count
EfficientNetB0	29	5.330.571
InceptionResNetV2	215	55.873.736
MobileNet	16	4.253.864
NASNetLarge	343	88.949.818
ResNet50V2	98	25.613.800
VGG16	528	138.357.544
Xception	88	22.910.480

Tab. 1: Neural Networks

A total of nine different neural networks were categorized by their size and depth into three groups (see Table 1). Each network was custom fitted into the task at hand with wrapper-layer encapsulating the intended behavior in a non-intrusive way to ensure that the default behavior is sustained. Each neural networks' internal behavior is treated as a black box.

The original input layer was discarded and replaced with a generic Input Layer accepting images of size 250x250. All models share the same input layer implementation. Furthermore, two additional layers were added to the stack to flatten the neural network's output and to result in a liveliness prediction score between 0 and 1. Sigmoid is used as the activation function.

It seems intuitive that complex networks should outperform lightweight implementations when comparing the validation accuracy, as the increased number in parameters gives an advantage. However, training times and prediction latencies are also expected to increase with a higher parameter count.

1.2 Dataset

The provided fingerprint samples were originally used as input material for a conference and competition about fingerprint detection. Bona fide fingerprints of 54 individuals were captured five times, summing up to 2700 genuine fingerprints. For a subset of individuals, an artificial fingerprint was crafted using multiple materials to create a set of fake images adding up to 3740 fake fingerprint images.

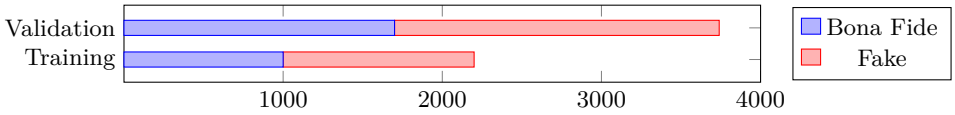

Fig. 1: Fingerprint Image Count

Training Data (37%, 2200 images)

Material	Count	Share
Live	1000	45%
Body Double	400	18%
Ecoflex	400	18%
Wood Glue	400	18%

Testing Data (63%, 3740 images)

Material	Count	Share
Live	1700	45%
Gelatine	680	18%
Latex	680	18%
Liquid Ecoflex	680	18%

Tab. 2: Material Distribution

Machine learning algorithms get more accurate the more data for training purposes is provided, as a richer set of unique training input improves the networks prediction skills. The popular machine learning utility library scikit-learn splits the data into training and validation subsets with a ratio of 3:1 [4]. For the LivDet2017 dataset, the ratio is almost flipped, with only 37% of images used for training purposes. As illustrated in Figure 1 and Table 2 the training dataset is approximately 50% smaller than the validation dataset.

About half (45%) of each dataset are bona fide fingerprints and the other half is comprised of materials emulating human skin. Body Double, Ecoflex, and Wood Glue (each 18%) were used to train while Gelatine, Latex, and Liquid Ecoflex (also each 18%) are used for validation. 59% of fingerprint samples are from female subjects while 41% are from males. Each image was also classified with an age.

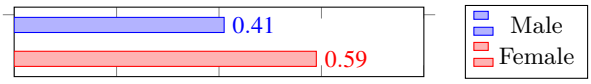

Fig. 2: Sex Distribution

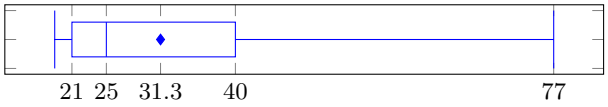

Fig. 3: Age Distribution

No recognizable variance in prediction accuracy is expected as the difference in sex and age do not infer a significant change in fingerprint anatomy or appearance to play a role in this experiment. With that said, a more mature finger can have noticeable damage like scars or common wear from labor and general use.

All images were captured on a Green Bit DactyScan84C and have a resolution of 500x500 pixel. The scanner is a standalone high-end device capturing fingerprints in high resolution. Bona fide fingerprint images are of such high quality that sweat glands are visible as small white spots on fingerprint ridges in some samples as illustrated in Figure ??.

Important to note is that none of the used material groups share distinct properties or features. It is, therefore, more difficult for the trained algorithms to infer the artificiality of unknown materials.

Below is a fingerprint image from each material group (see Figure 6). None of the materials capture sweat glands which may be beneficial to determine whether the presented image is an a attack presentation or genuine.

The images used in the context of the experiments are only a subset of the dataset used in the LivDet 2017 competition.

Fig. 4: Training Dataset

Fig. 5: Validation Dataset

Fig. 6: Fake Fingerprints for each Material

1.3 Methodology

The experiment was split into two parts. During the first phase, the best-case (highest average validation accuracy) for each network was determined. Due to randomized initial values and subsequent variance in training evolution, the resulting validation accuracy can fluctuate to some degree. Repetitive training and evaluation will give sets of models for each network that can be serialized and stored.

All model-internal layers are initially frozen to inspect out-of-the-box behavior while training with the same procedure once more. During the second training, all parameters are unfrozen and able to change. A comparison between default and more specialized models will give insight into the importance of adaptive training. The unfrozen models are expected to perform a lot better. In the following sections, experiments and datasets labeled as "liquid" refer to models trained with all layers unfrozen. The fingerprint dataset contains a material called "Liquid Ecoflex" which is the exemption of this convention.

Models are compiled with an Adam-Optimizer using a learning rate of 0.0001 and a binary cross-entropy loss function. Models are trained using the provided training dataset over ten epochs, which delivered a good balance between validation accuracy and training time. At that point training accuracy was at 100% and the loss value fluctuated around the training sessions minimum.

A total of ten training iterations for each network will be used as a pool to pick the best-performing model. Trained models are not volatile, so using the best case for each is the fairest approach to ensure every network has the best chances.

Further experiments are conducted in the second phase, where the best-case models for each network predict whether a fingerprint presentation is bona fide or an artificially constructed fingerprint replica. Unlike neural network training, the predictions of the trained model are static and will not change. The resulting predictions are used to analyze possible correlations between input data and predicted values.

Predictions are classified by assuming that prediction values of at least 0.50 suggest a bona fide fingerprint while all lower values suggest an attack presentation. This ratio was chosen to directly compare scores and accuracies with ones from the algorithms submitted to the conference. [2] Since the activation function producing the scores is sigmoid, the prediction confidences are concentrated at the two extrema leaving few indecisive predictions.

Various aggregates will show possible correlations in prediction accuracy between the used material in case of a presentation attack or otherwise prediction differences in sex and age of the test subject. Additionally, the prediction latency for each network will be discussed and brought into context with the networks' size and complexity.

All experiments are performed on a capable workstation with an 8/16 Core CPU, 16Gb RAM and discrete graphics. TensorFlow reports a compute capability of 6.1. The hardware does not have an impact on prediction accuracy but rather training and prediction times.

2 Experiment

2.1 Phase 1.1 - Frozen Training

Each network was trained ten times over the course of three hours resulting in a representatively distributed set of trained networks. Figure 7 visualizes the relations between the average training times.

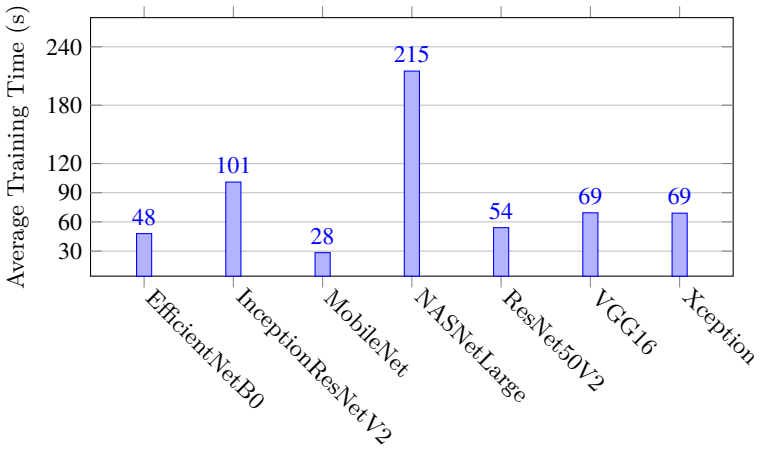

Fig. 7: Figure: Average Training Time in seconds

No linear relation between parameter count and training time can be determined. NasNetLarge takes by far the longest to train with an average of approximately 3.5 minutes. MobileNet was the fastest to train with slightly under 30 seconds.

Many networks hover around 90% validation accuracy for their best case. The earliest insight is that light-weight, small networks are up to par with bigger, much more complex implementations in the context of this experiment. An additional number of parameters does not seem to indicate better prediction accuracy for binary classifications. Even smaller networks obtain the same or a better average validation accuracy.

InceptionResnetV2 has a high accuracy fluctuation between training sessions which could not be attributed to any superficial property and the reason behind it is unclear. On average, the aforementioned network performed the worst while being the second largest.

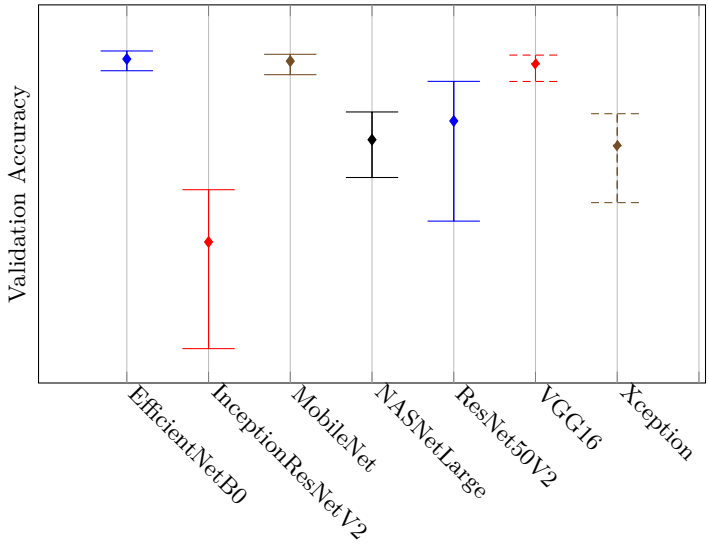

Fig. 8: Validation Accuracies

After the training phase, the best-case networks were identified using the average validation accuracy. The weights for each node are persisted and can be downloaded and applied to reproduce the predictions. The following analysis will feature these persisted models.

2.2 Phase 1.2 - Liquid Training

All networks are trained again with the same configuration, but now all layers are able to adapt their parameters. Immediately noticeable is the drastic increase in training time for each network. Another important fact is the increase in resource consumption of unfrozen layers during the training phase. The training was terminating multiple times during training iterations with variations of out-of-memory exceptions. Figure 9 visualizes the relations between the average training times once again.

NasNetLarges training time increased tenfold and it takes almost 40 minutes on average.

The increased time spent for a more thorough training phase was worth it as almost all networks now break the 90% validation accuracy barrier. InceptionResnetV2 delivers a respectable top accuracy after the disappointing result in the last section.

Table 3 shows the relative gain in validation accuracy of frozen versus freely trained networks. Many of the smaller networks do not gain much, but on the other hand, InceptionResnetV2 and Xception are now much more competitive. Xceptions best

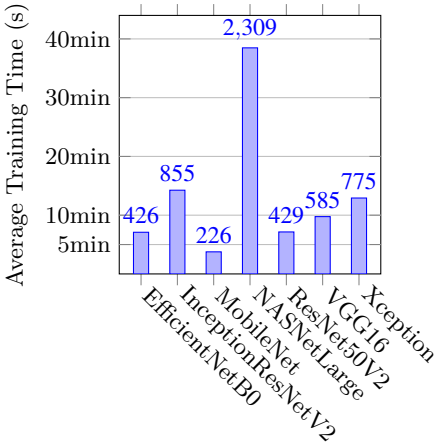

Fig. 9: Figure: Average Training Time in seconds

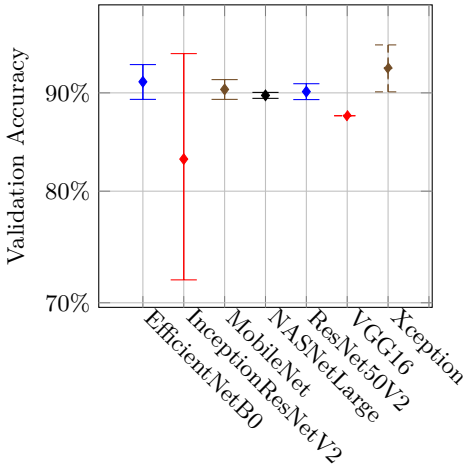

Fig. 10: Validation Accuracies

Network Name	Frozen	Liquid	
EfficientNetB0	90.29	93.10	+3.0%
InceptionResNetV2	66.39	94.33	+29.6%
MobileNet	89.63	91.44	+2.0%
NASNetLarge	78.88	90.05	+12.4%
ResNet50V2	84.41	90.99	+7.2%
VGG16	89.47	87.57	-2.2%
Xception	78.58	95.32	+17.6%

Tab. 3: Difference in Top Validation Accuracy

validation accuracy is comparabale with some of the algorithms handed in during the LivDet2017 competition in regards to the dataset this experiment uses. [2]

2.3 Phase 2 - Prediction Analysis

Serialized models are loaded and the testing dataset will be interpreted once again while capturing confidence values for further analysis.

EfficientNetB0, MobileNet and VGG16 can still be considered usable to support fingerpint recognition, but InceptionResNetV2, NASNetLarge, ResNet50V2 and Xception do not perform well enough without training. A drastic change is visible in the detection error trade-off curves (Figure 11 & 12) from frozen to liquid.

InceptionResNetV2 is performing the best when targeting systems with a near zero false-positive acceptance rate, as the bona fide presentation classification error rate (BPCER) is the lowest of the tested networks.

Xception on the other hand is better suited for system with a near zero false-negative acceptance rate, since the attack presentation classification error rate (APCER) is the lowest.

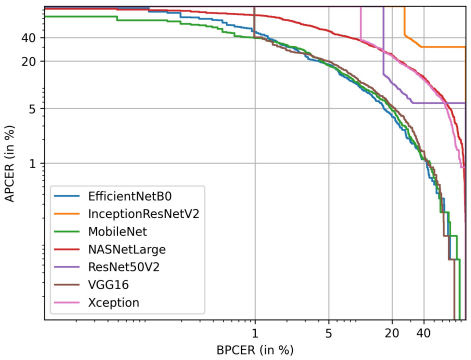

Fig. 11: DET Curve Frozen Training

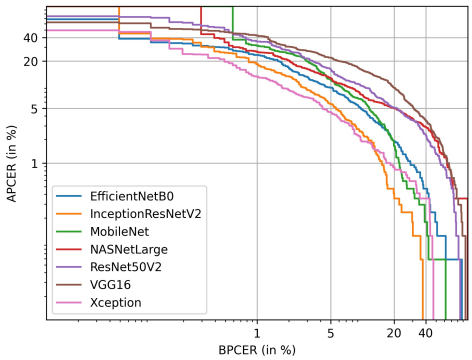

Fig. 12: DET Curve Liquid Training

In the following subsections, each network will be inspected individually by evaluating the performance implicitly assuming a bona fide / presentation attack threshold of 0.5. This allows the performance to be differentiated in terms of certain materials while staying true to the standards given by LivDet2017 [2].

2.3.1 EfficientNet B0

Even before unfreezing all layers EfficientNetB0 was providing a usable PAD detection accuracy. Only a small increase in accuracy could be gained by liquid training. Interestingly, the correct non-match rate increased by almost 6% while the correct match rate sank slightly.

Prediction latencies are consistent, but the few outliers make the average time quite a lot higher than the median latency as seen in Table 6.

Material	Frozen	Liquid
Live	90.29%	93.10%
Gelatine	92.65%	97.94%
Latex	88.53%	95.88%
Liquid Ecoflex	89.41%	93.82%

Tab. 4: Validation Accuracies

	Frozen	Liquid
BPCER	10.06%	10.59%
APCER	9.41%	3.82%

Tab. 5: Prediction Rates

25% Low	Med	Avg	75% Low
30.57	32.99	35.64	34.56

Tab. 6: Prediction Latencies (ms)

2.3.2 Inception Resnet V2

A surprising increase in prediction accuracy of almost 30% was observed for InceptionResnetV2. The detection scores for all materials benefitted from the liquid

training. Liquid Ecoflex was already reliably detected by the network but is now detected more reliably. The network gained significant detection capabilities for all other materials.

An average latency of 60ms is slow compared to the other networks.

Material	Frozen	Liquid
Live	66.39%	94.33%
Gelatine	66.47%	91.47%
Latex	62.06%	90.88%
Liquid Ecoflex	90.29%	95.29%

Tab. 7: Validation Accuracies

	Frozen	Liquid
BPCER	40.47%	3.53%
APCER	27.89%	7.45%

Tab. 8: Prediction Rates

25% Low	Med	Avg	75% Low
53.77	55.65	60.82	57.74

Tab. 9: Prediction Latencies (ms)

2.3.3 MobileNet

Not a lot improved for MobileNet by unfreezing the layers and in case of Gelatine some accuracy was lost. Bona fide fingerprints were correctly detected with an accuracy of 91.7% and 94.4%, while attack presentations were detected correctly with 87.9 and 89.0%.

With the relatively inaccurate detection comes the shortest latency with an average of 27ms. MobileNet was the fastest network in this experiment to deliver a prediction result.

Material	Frozen	Liquid
Live	89.63%	91.44%
Gelatine	87.06%	83.82%
Latex	90.59%	92.35%
Liquid Ecoflex	88.82%	94.12%

Tab. 10: Validation Accuracies

	Frozen	Liquid
BPCER	8.29%	5.59%
APCER	12.11%	11.03%

Tab. 11: Prediction Rates

25% Low	Med	Avg	75% Low
23.81	24.49	27.57	26.62

Tab. 12: Prediction Latencies (ms)

2.3.4 Nasnet Large

Liquid Ecoflex samples were correctly classified most of the time with a high average rate of 99.7% after the unfrozen training. The high attack presentation detection is unfortunately paired with a low correct match rate of only 80% after unfrozen training, which is the lowest in the entire experiment.

With an average of almost 73ms NasNetLarge has the highest prediction latency.

Material	Frozen	Liquid
Live	78.88%	90.05%
Gelatine	84.12%	97.35%
Latex	84.71%	97.94%
Liquid Ecoflex	92.94%	99.71%

Tab. 13: Validation Accuracies

	Frozen	Liquid
BPCER	31.88%	19.35%
APCER	12.16%	2.11%

Tab. 14: Prediction Rates

25% Low	Med	Avg	75% Low
65.36	66.76	72.92	68.06

Tab. 15: Prediction Latencies (ms)

2.3.5 Resnet V2

Liquid Ecoflex samples were again correctly classified most of the time with a similar rate of 99.1% after liquid training. BPCER and APCER are more balanced and result in an overall better accuracy in comparison to NasNetLarge. The average prediction latency is half of NasNetLarges.

Material	Frozen	Liquid
Live	84.41%	90.99%
Gelatine	82.35%	92.65%
Latex	76.76%	90.00%
Liquid Ecoflex	87.35%	99.12%

Tab. 16: Validation Accuracies

	Frozen	Liquid
BPCER	12%	11.76%
APCER	18.58%	6.72%

Tab. 17: Prediction Rates

25% Low	Med	Avg	75% Low
32.17	33.51	36.52	35.08

Tab. 18: Prediction Latencies (ms)

2.3.6 VGG16

The liquid training resulted in a drastic decrease of bona fide fingerpint recognition and compared to that only a minor increase in correct non-match rate was gained. VGG16 is the only network that lost accuracy after liquid training.

Material	Frozen	Liquid
Live	89.47%	87.57%
Gelatine	91.47%	92.65%
Latex	87.94%	93.53%
Liquid Ecoflex	89.41%	99.12%

Tab. 19: Validation Accuracies

	Frozen	Liquid
BPCER	10.47%	20.94%
APCER	10.59%	5.34%

Tab. 20: Prediction Rates

25% Low	Med	Avg	75% Low
30.44	30.84	33.20	31.53

Tab. 21: Prediction Latencies (ms)

2.3.7 Xception

The best performer in the test has a correct non-match rate of 97.5% and returns a prediction result in under 36ms. Presentation attacks were able to be detected precisely with a max delta of under 2%.

Material	Frozen	Liquid
Live	78.58%	95.32%
Gelatine	78.24%	97.94%
Latex	65.29%	98.24%
Liquid Ecoflex	90.88%	97.06%

Tab. 22: Validation Accuracies

	Frozen	Liquid
BPCER	22.06%	7.29%
APCER	20.88%	2.5%

Tab. 23: Prediction Rates

25% Low	Med	Avg	75% Low
32.07	32.59	35.91	34.36

Tab. 24: Prediction Latencies (ms)

3 Interpretation

Larger networks suffer from a complex set of layers and nodes that come with no benefits and even weakens the purpose as the increased calculation time brings inconvenience to the potential user. Predictions may be disturbed with noise coming from convolutional layers providing data which has little-to-no use for fingerprint presentation attack-detection.

The extreme counterexample is EfficientNetB0 in this experiment. An impressive overall accuracy coming from a high APCER makes this convolutional neural network a prime starting point for a competitive alternative. With only 29Mb in size on disk it is small enough to fit on embedded devices as well.

4 Conclusion

Considering the increase in complexity as well as training and prediction times that deep learning networks with many parameters such as VGG16 bring, the lack in detection accuracy is contrary to expectations. The original purpose of these networks is to analyze and classify real world objects and animals which bring an acceptable detection rate for fingerprint presentations, but lack the precision and accuracy for real world applications. Smaller networks in particular represent a good foundation to fine tune and develop a solution which is better fitting for the job. By concentrating on the key features of the fingerprint, like sweat glands or the smooth curvature of ridge lines of the imprint, a much more accurate system can be built.

Image classification has many facets, but the underlying principles are the same for all applications. Transfer learning is a powerful paradigm that can lead to surprising results when introducing the pre-trained models to new use cases. Given the small

training dataset, much more accurate predictions are expected when using more extensive reference data to further optimize results.

Glossary

Keras Python Software Interface for TensorFlow. 2

scikit-learn Software Machine Learning Library. 3

TensorFlow Open Source Platform for Machine Learning. 2

References

[1] Howard et al. “MobileNets: Efficient Convolutional Neural Networks for Mobile VisionApplications”. In: (2017), p. 1.

[2] Mura et al. “LivDet 2017 Fingerprint Liveness Detection Competition2017”. In: (2017), p. 4.

[3] ImageNet. https://www.image-net.org/download.php, 13.06.2021. 2021.

[4] scikit-learn Documentation. https://scikit-learn.org/stable/modules/classes.html, 13.06.2021. 2021.