Towards CNN-based Level 1 Feature Extraction for Contactless Fingerprint Recognition

Siri Lorenz, Jannis Priesnitz, Mathias Ibsen, and Christoph Busch

da/sec - Biometrics and Internet Security Research Group, Hochschule Darmstadt, Schöfferstr. 3, 64295 Darmstadt, Germany
firstname.lastname@h-da.de

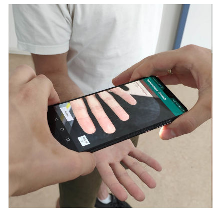
Abstract. This work examines the detection of ridge orientation patterns, also referred to as level 1 features, from contactless fingerprint images and their classification. We trained two Convolutional Neural Networks (CNNs) to classify fingerprints based on their ridge orientation patterns. Our models were trained on synthetic data generated by SynCoLFinGer. Afterwards, we conducted various experiments for classifying these patterns and evaluated our trained models on four real-world databases: PolyU CB2CL, ISPFDv1 contactless fingerprint database, and two in-house databases.
We report the classification accuracy in terms of Classification Error Rate (CER). We achieved CERs between 28% and 38% considering all samples. Due to the amount of low-quality samples included in the database, we use NFIQ 2.2 to iteratively exclude samples from the databases and report the corresponding CER. We then used NFIQ 2.2 scores to iteratively exclude samples and hence report the impact of low-quality samples.
By excluding the lowest scoring 10% of all samples within each database, we achieve CERs of 24% to 35% depending on the databases. While these error rates are still high, they show promise compared to the original values. Although further research is needed to improve results, we show that combining quality-score-based exclusion of images with CNNs trained on synthetic contactless data is a promising method to classify fingerprint patterns.

Keywords: Contactless Fingerprint Recognition · Fingerprint Feature Extraction · Level 1 Features · Convolutional Neural Networks

1 Introduction and Related Work

Fingerprints have been accepted as an unchangeable, unique biometric characteristic allowing human recognition since the 1890's and have been in use by law enforcement ever since [19]. From that point on, fingerprint recognition has evolved and has been in operational use for decades. Most modern smartphones have fingerprint capturing devices and many restricted areas are protected using fingerprints as a means of identification.

(a) Contact-based



(b) Contactless

Fig. 1: Different capturing methods for contact-based and contactless fingerprint samples. (From: [22])

In recent years, especially with the spread of the COVID-19 pandemic, the need for more hygienic alternatives has gained awareness. One of those alternatives is the field of contactless fingerprint recognition [7]. Although this field was thoroughly researched over the years, most feature extraction algorithms are still designed for the contact-based domain. Contactless fingerprints differ from contact-based ones. Contact-based fingerprints are acquired using a capturing device which needs to be touched. By pressing the fingertip onto a capturing device, it undergoes temporary elastic deformations. These deformations do not occur for contactless fingerprints. However, contactless fingerprints can be captured in unconstrained environments and can be rotated around every axis. Examples for contactless and contact-based capturing approaches can be seen in Figure 1.

To ensure that contact-based feature extraction algorithms also function with contactless fingerprints, contactless fingerprint images need to be preprocessed. Feature extraction is one of the most crucial processing steps in fingerprint recognition. Robust feature extraction strategies allow biometric systems to operate at low error rates. Fingerprint features are typically divided into three levels [19]:

– Level 1: the general orientation flow of the ridge pattern
– Level 2: minutiae features, special points e. g. bifurcations or ridge endings in the ridge pattern
– Level 3: fine-grained patterns like sweat pores

In 2D contactless fingerprint recognition using commodity devices, level 3 features are in general hard to extract since the resolution of capturing the subsystem is mostly insufficient. Therefore, most proposals focus on level 2 features e. g. [6, 9].

Most contactless recognition workflows use feature extractors from the contact-based domain, which are applicable for contactless fingerprint but lead to inferior performance. On the other hand, elaborated feature extraction algorithms for specific contactless capturing subsystems show a high performance [17]. However, no broad evaluation has yet been conducted in order to showcase feature extraction performance on databases obtained from various capturing devices.

Contactless fingerprint recognition is a complex and active field of research. Chowdhury *et. al* conducted a review of deep-learning algorithms in the area of contactless fingerprint recognition [5] and more recently, Mohamed-Abdul-Cader *et. al* [20] presented a comparative overview of contact-based as well as contactless 2D and 3D approaches with a focus on capturing and interoperability.

Several works propose dedicated contactless feature extractors. With ScatNet, Sankaran *et. al* [24] and Malhotra *et. al* [17] proposed a feature extractor. They used group-invariant scattering networks [18], which refer to a filter bank of wavelets that produce a representation which was shown to be robust to local affine transformations. The authors extended the approach and compared their ScatNet approach to a minutia-based baseline using VeriFinger SDK [21] and Minutiae Cylinder Code [3] for feature extraction. They concluded that their algorithm performed slightly better than the others.

Yin *et. al* [28] proposed a distortion-free feature representation using the ridge count as a feature. While Kumar and Zhou [14] suggested a feature extraction based on level 0 features, such as local texture patterns. However, most literature still refers to only level 1, 2, and 3 features. In a more recent work, Vyas and Kumar [26] suggested an improved scheme using minutiae comparison.

Jung and Lee introduced an approach to classify noisy and incomplete fingerprints using local ridge distribution models. They divide the images into four regions along the detected core blocks and use the distribution of the directional values in each region. The fingerprint classification is conducted using local models [11].

Most existing proposals focus on level 2 features or define a new feature level. While some level 3 feature proposals exist, they are unsuited for contactless fingerprints, since the needed resolution and quality are not yet achievable for general contactless fingerprint recognition. To boost level-2-based fingerprint recognition, level 1 features can be used. By determining if two samples belong to the same level 1 characteristic, we can support level 2 match decisions in case of uncertainty. There are some advantages of level 1 features that enable their usage as support for level 2 match decisions. They are scale- and rotation-invariant. Furthermore, level 1 features are extractable even from challenging fingerprint images.

The ridge orientation flow feature is seen to have several advantages compared to minutiae features [19, 13]. Level 1 features are rotation and scaling invariant. This aspect is in particular relevant for unconstrained capturing scenarios where rotations around every axis are possible. Level 2 features suffer from an inferior performance caused by a high spatial distance between mated minutiae due to differences in terms of scaling or rotation.

Furthermore, level 1 features are more robust against low-quality images which are only partly captured, as long as the general pattern is discernible. Especially if the intersecting area between two mated samples is small, the number of mated minutiae is small, thus the obtained comparison score is low. Here, the recognition can benefit from level 1 feature analysis.

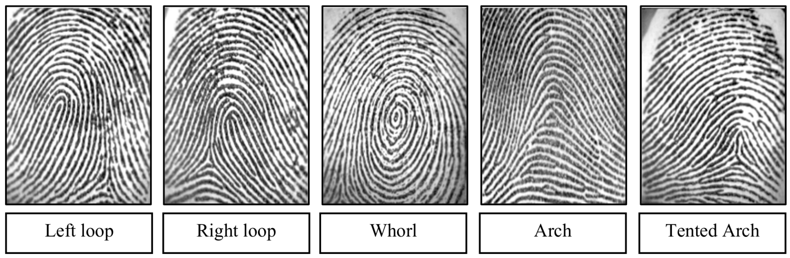


Fig. 2: Examples of the five most common level 1 features. Samples are captured using a contact-based capturing device. (From [19])

The ridge orientation flow as property of the entire fingertip area is less affected by insufficiently captured or rotated fingerprint images. Most notably, this feature extraction method is generally applicable without extensive preprocessing or fine-tuning to a specific capturing setup. However, level 1 features cannot be used as a standalone feature extractor rather than as an auxiliary metric in addition to a level 2 feature extractor. So far, no proposal to level 1 feature extraction in the contactless domain is known. For contact-based fingerprints, a few investigations have been conducted [27]. To the best of our knowledge, no proposal for level 1 feature extraction in the contactless domain with CNN-based classification exists.

Therefore, we decided to explore level 1 features for contactless fingerprint recognition using CNNs. Figure 2 depicts the five main level 1 characteristics, left loop, right loop, whorl, arch and tented arch. It should be noted that some works also consider double loops, or variations of whorls as level 1 characteristics [4, 19]. However, the fraction of double loops and other accidental patterns relative to all fingerprint is very low. For further information the reader is referred to [12].

We train two CNNs to classify real-world datasets and use NFIQ 2.2 scores to iteratively exclude the lowest scoring samples. To use CNNs we need a balanced training dataset. However, as the five level 1 base-characteristics are not uniformly distributed, finding suitable training datasets is hard. One of the biggest challenges for contactless fingerprint recognition is the lack of sufficient training data [16]. Hence, we decided on synthetic training data.

This paper is structured as follows: In section 2 we will introduce our proposed method. In section 3 we elaborate the experimental setup and introduce the real world databases used for testing our approach. section 4 will present our results and explain the limitations of our approach. Finally, in section 5 we will present our conclusion and possible future work.

2 Proposed Method

This section provides information on fingerprint recognition, CNNs and the SynCoLFinGer algorithm.

2.1 Used Convolutional Neural Networks

Our approach evaluates the performance of two CNNs – ResNet18 and SqueezeNet which we are going to introduce in more detail.

ResNet The ResNet models are based on the approach introduced by *Deep Residual Learning for Image Recognition* [8]. The authors point out the advantages of residual representation for other research fields in computer science e. g. low-level vision and computer graphics and conclude that residual representation might also simplify the optimisation for CNNs. They construct a residual network by inserting so-called shortcut connections into a *plain* CNN. They implemented various *residual networks* with different amount of layers: 34, 50, 101, and 152. The 50, 101, and 152 layer models are more accurate than the 34 layer one. Furthermore, the PyTorch library offers an implementation of a ResNet model with 18 layers[1]. After running some test, we decided on ResNet18, as it yielded the most promising results on our training data.

SqueezeNet SqueezeNet is a CNN that tries to achieve competitive accuracy while keeping the network itself small [10]. The authors point out three advantages of smaller models: First, they require less communication across servers during distributed training. Second, they require less bandwidth to export a new model to a client – e. g. from cloud to an autonomous car. And third, they are more feasible to deploy on hardware with limited memory. They conclude their work stating that they achieved their intended accuracy. They note that SqueezeNet uses 50 times fewer parameters than other CNNs achieving this accuracy.

2.2 Training Data Preparation

We decided to use synthetically generated contactless fingerprints as training data to examine the usability of synthetic data as training data and ensure a balanced training set for our experiments. To generate synthetic contactless fingerprint images we used the SynCoLFinGer algorithm.

SynCoLFinGer is an algorithm introduced in 2022 [23]. It is based on SFinGe [2] and transforms synthetic contact-based ridge line patterns into synthetic contactless fingerprints. Shortly after the original algorithm was published, SynCoLFinGer was extended to include the most frequent imperfections of contactless fingerprints[23, 16]. In a first step, the SFinGe ridge line patterns are deformed and transformed to the contactless domain through rotation and ridge line thinning. The second step adds the subject characteristics to the ridge line pattern, such as skin colour, skin tone variance, wounds, scars, or other dermatological issues which are longer lasting and do not change or disappear quickly. During the third step, environmental influences such as shadow or illumination

[1] `https://pytorch.org/vision/stable/models/resnet.html`

(a) Arch (b) Left Loop (c) Right Loop (d) Tented (e) Whorl

Fig. 3: Original SynCoLFinGer example images (From [23])

variance, dirt, sensor noise or ink stains are added. Dirt and ink stains can be washed off easily, and shadows and illumination can change very quickly, e. g. if one is on a train passing trees, while sensor noise is unique for each sensor and therefore, changes every time a different sensor is used.

SynCoLFinGer images can be generated in different quality settings, enabling the generation of a diverse training dataset with low- and high-quality images [23]. Examples for these classes and qualities can be seen in Figure 3. We chose SynCoLFinGer due to its fine-granular adjustability and its ability to generate more than one sample for each subject.

However, SynCoLFinGer faces some limitations. E.g. the limited data valiance and the current inability to generate only the upper part of the top phalanx.

3 Experimental Setup

This section will introduce the databases used for evaluating our experiments and explain the experimental setup. We generated 12500 samples with 2500 images for each level 1 class. When training CNNs it is necessary to use uniform distributed training data to avoid a training bias towards one of the classes. Since level 1 features are not uniformly distributed it is hard to use real-world databases for training.

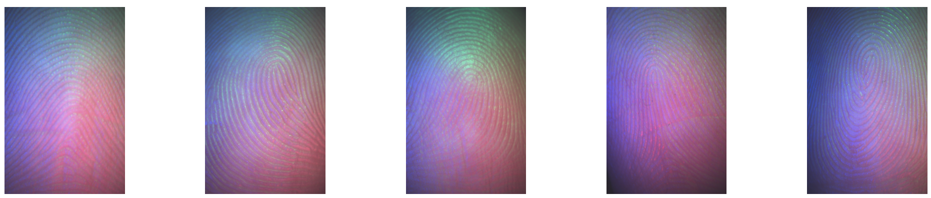
3.1 Databases used for Evaluation

First, we will introduce the databases used for testing. Namely: ISPFDv1, PolyU CB2CL, and two in-house databases: HDA and HDA-UniCa.

ISPFDv1 The ISPFDv1 database consists of four subsets: natural indoor, natural outdoor, white indoor, and white outdoor. Natural and white refer to the background of the images. The white background is achieved by having a white paper behind the photographed fingers [25]. Each of these four subsets is based on 64 subjects with roughly eight samples per subject. From each subject, two instances were captured: right index and right middle finger. The natural indoor

(a) Arch (b) Left (c) Right (d) Tented (e) Whorl

Fig. 4: ISPFDv1 example images. (From [25])



(a) Arch (b) Left (c) Right (d) Tented (e) Whorl

Fig. 5: PolyU CB2CL example images. (From [15])

subset consists of 1,010 samples of 64 subjects and their two entities, while natural outdoor consists of 1,016 images. White indoor has eight samples for each instance (1,024 samples) while white outdoor consists of 1.015 images. Examples for these images can be seen in Figure 4.

PolyU CB2CL The PolyU CB2CL database contains two sessions of both contact-based and contactless fingerprints of up to 336 subjects. For each instance, there are six samples per session [15]. As we are only interested in the contactless data for this work, we focus on the 2016 contactless images in the first and 960 images in the second session. PolyU CB2CL is the database with the highest overall image quality, of those used for this work. The images do not show the outline of the fingers but rather focus on the ridge line pattern. Examples for these images can be seen in Figure 5.

HDA Database The HDA database [22] consists of contactless samples captured in two different setups: a constrained box-setup and an unconstrained tripod-setup. For the capturing, the authors used two different smartphones. An application automatically captured the four inner-hand fingers and processed them to fingerprint samples. Both setups have 28 subjects, but the number of instances and samples varies, resulting in 452 samples in the constrained subset

(a) Left (b) Right (c) Whorl

Fig. 6: HDA example images. (From: [22])

(a) Arch (b) Left Loop (c) Right Loop (d) Tented (e) Whorl

Fig. 7: HDA-UniCa example images (preprocessed).

and 445 samples in the unconstrained subset. Example images[2] can be seen in Figure 6.

HDA-UniCa Database The HDA-UniCa Database includes two subsets: The first subset is captured using the same setup like the HDA database. Here, 17 subjects were captured in five sessions using a smartphone. The capturing was carried out in an attended scenario where a second person handles the capturing device. The second subset is a contactless stationary dataset which was captured using a Secugen Hamster Air capturing device. Here, all ten fingerprints were captured in five sessions per fingerprint. Samples from 21 subjects were captured in 5 sessions resulting in 1.050 samples. Examples of these images can be seen in Figure 7.

Each database is captured in different environments and the quality of the databases varies. While PolyU CB2CL is constrained in its capturing, resulting in higher quality images, ISPFDv1 is more challenging. As mentioned, ISPFDv1 consists of indoor and outdoor subsets. The outdoor images are challenging due to their difference in lighting and shadows, while the indoor images are taken in an environment with less light, which the camera has to compensate for. The HDA database is also divided into a constrained and unconstrained subset, of which the unconstrained one is more challenging. For the HDA-UniCa database,

[2] No subject with arch or tented arch characteristics consented to their pictures being published.

some samples have a darker area in the center of the ROI on most fingerprints, as can be observed in Figure 7.

3.2 Preprocessing

All fingerprint images are preprocessed using the same preprocessing method. We use the Contrast Limited Adaptive Histogram Equalization (CLAHE) on a grey scale converted fingerprint image to emphasise the ridge-line characteristics.

Experiments have shown that some feature extractors might detect many false minutiae at the border region of contactless fingerprint samples. Therefore, we crop approx. 15 pixel of the border region. Furthermore, the segmentation mask is dilated in order to reduce the size of the fingerprint image [22].

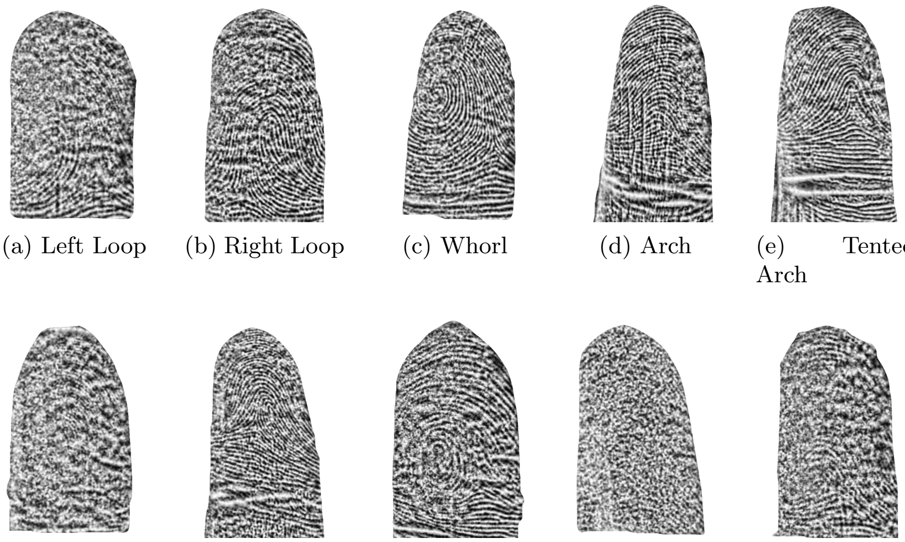
3.3 Convolutional Neural Networks

We tested the two previously introduced CNNs for general purpose image classification and fine-tuned them to the special purposes of fingerprint ridge orientation flow classification.

In comparison to other CNNs, rather shallow networks like the considered SqueezeNet and ResNet18 have shown to achieve a higher accuracy on a relatively small amount of training data [1]. As mentioned in section 2, SynCoLFinGer was used to generate the training data. Using this algorithm it was possible to generate disjoint sub-databases for each of the five level 1 feature classes. Despite the fact that level 1 characteristics are not equally distributed in real fingerprints, we train the algorithms on uniformly distributed sub-databases in order to avoid a bias to a certain class. We evaluated our methods on the ISPFDv1, PolyU CB2CL, and the two in-house databases. The evaluation databases were labelled manually. It should be noted that all databases also contain a contact-based subset and have a $6 - 10$ samples per fingerprint instance. This makes a manually labelling precise and efficient, since more reference images are available if one sample is of such low quality that it becomes indiscernible.

All considered databases are pre-processed using the same method. In addition, we cropped a centred patch of 224x224 pixels from every sample in order to make the data suitable for the chosen CNNs. Figure 8 presents a sample training-image for every class.

The training and validation is exclusively conducted on synthetic samples whereas the methods are only tested on real data. The database which performs best contains 2,500 training samples per class which results in a total size 12,500 samples. The SynCoLFinGer parameters are set to generate final samples of rather heterogeneous quality. Since SynCoLFinGer already uses slight rotations in its generation process only a zooming augmentation is added to the images. The models are trained for up to 30 epochs using a learning rate 0.001. During the training ResNet18 achieves a validation accuracy of 96.96% whereas SqueezeNet has a validation accuracy of 97.16%.



(a) Left Loop (b) Right Loop (c) Whorl (d) Arch (e) Tented Arch

(f) Left Loop (g) Right Loop (h) Whorl (i) Arch (j) Tented Arch

Fig. 8: Example images of correctly classified (a – e) and incorrectly classified images (f – j) using ResNet18. It should be noted that both samples from the class "arch" (d) and (i) are from the same subject. (From: [25])

4 Results

This section will introduce our results and used metrics. We report our results as Classification Error Rate (CER) which is computed as follows:

$CER = s_{ic}/s_{all}$

where s_{ic} refers to incorrectly classified samples and s_{all} refers to all samples in the test set.

It should be noted that the classification error is reported over all classes. Further, we use the NFIQ 2.2 quality assessment algorithm to identify samples of low quality. In our experiments, we iteratively exclude samples with the lowest quality score and report the classification error on the remaining subset. In comparison to reporting the accuracy, the CER can precisely analyse the impact of low sample quality on the classification accuracy and hence define operational thresholds. If the whole database is considered, both evaluated algorithms achieve a rather low classification accuracy. In general, SqueezeNet achieves more stable results on all databases than ResNet18. However, the ISPFDv1 database performs better with ResNet18. The largest difference can be observed from the PolyU CB2CL database. In general the PolyU CB2CL database has a high sample quality and should be rather easy to classify. However, it should be noted that the synthetic training database is not designed to represent the samples included in the PolyU CB2CL database which, as previously mentioned, are captured by a stationary differential light method.

The HDA database shows high CERs. Here again, SqueezeNet performs slightly better but a CER around 0.4 is in general too high to have a positive effect on a recognition system. It should be noted that especially the ISPDFv1 and the HDA database contain many samples which are considered challenging due to their low image quality. For this reason, we use NFIQ 2.2 scores to exclude samples of low quality from the experiment and explore how the CER changes. From Figure 9 we can observe that there is a strong correlation between NFIQ 2.2 scores and the CER. Specifically meaning, the CER decreases once lower quality samples are excluded. From this observation we can conclude that the proposed algorithm is in general functional but faces challenges, especially to correctly classify samples of low quality. Therefore, a quality threshold should be defined to reject low-quality samples. In an operational scenario, this threshold should be selected carefully in order to not force too many capturing attempts while maintaining a low CER.

Table 1 and Table 2 show CER and fraction of rejected samples for distinct NFIQ 2.2 scores. We observe, that the NFIQ 2.2 threshold is varying between the different databases which due to the different capturing methods and the general usability of the samples. Additionally, the tested algorithms perform differently.

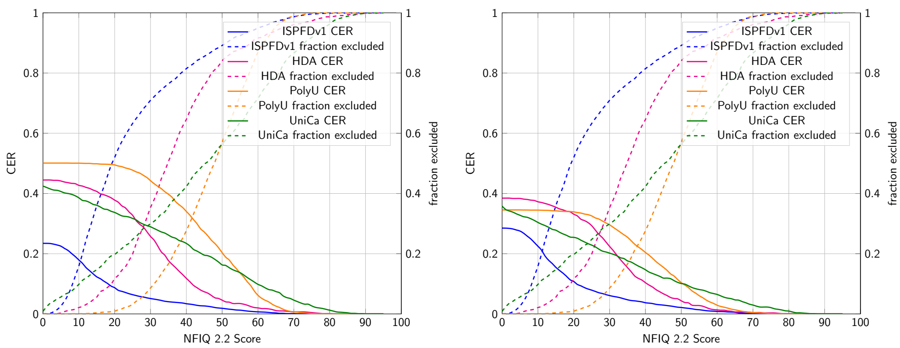
Figure 9 illustrates the correlation of CER and excluded samples. Furthermore, by visualising the trade-off over the NFIQ 2.2 scores, feasible NFIQ 2.2 score thresholds for each database become more apparent.

Considering the SqueezeNet algorithm tested on the ISPFDv1 database, at a NFIQ 2.2 score of 10, already 16% of the samples are excluded whereas 78% of the samples are correctly classified (CER = 22%). Here, higher NFIQ 2.2 values lead to a large portion of excluded samples and hence to a low benefit in an application scenario. However, considering the challenging characteristic of the database, the method could support a minutia-based recognition workflow.

The HDA database in general consists of images with slightly better quality scores compared to the ISPFDv1 database. However, the classification accuracy is relatively low until 35% of the samples are excluded, which is then 22% (at NFIQ 2.2 of 30). Here, the proportion between the excluded samples and correctly classified samples is not sufficient to have a positive effect on the recognition accuracy.

For the PolyU CB2CL database only a very few samples are excluded until a NFIQ 2.2 score of 20. This indicates that the capturing process is rather robust. The CER remains over 20% until more than 30% of the samples are excluded. Which is double the fraction of excluded samples in comparison to ISPFDv1, but still outperforms the HDA database. It should be noted that the impression of the samples included in the training database is highly different compared to the PolyU CB2CL database and for this reason, the classification performance is inferior compared to the other databases.

The HDA-UniCa database has a more uniform distribution of NFIQ 2.2 scores, which is indicated by the almost linear line of fraction excluded samples



(a) ResNet18 (b) SqueezeNet

Fig. 9: Classification error which is made by excluding samples according to NFIQ 2.2 score. The "fraction excluded" indicates the fraction of excluded samples. The CER indicates the classification error made considering only samples with a NFIQ 2.2 score above the threshold.

in Figure 9. It performs similar to PolyU CB2CL with our trained SqueezeNet model, achieving a CER of roughly 20% when excluding approximately 30% of all samples. Due to its automated capturing method several samples are only partly captured while the image is still focused. Although level 1 feature classification is relatively robust to partly captured images, if only the upper part of the fingertip is captured, no level 1 characteristic can be derived from that image forcing the trained model to guess.

ResNet18 performs better for ISPFDv1, but worse for all other databases. In detail, the ISPFDv1 database performs slightly better with the ResNet18 algorithm (17.93% CER and 15,89% excluded samples at a NFIQ 2.2 score of 10). In contrast, both other databases perform slightly worse compared to SqueezeNet. The CER of the HDA database is 25.75% (35.34% excluded samples, NFIQ 2.2 score of 30) and the PolyU CB2CL shows a CER of 20.33 with 56.96% of excluded samples (NFIQ 2.2 score 50). Here, the HDA-UniCa database performs close to HDA database. From this we can summarise that the combination of database and classification algorithm should be selected carefully.

As it was shown in Table 1 and Table 2 this approach still faces many limitations.

The first notable limitation is the lack of variety in our training data. While SynCoLFinGer is able to generate a variety of different quality presets and also models the most common imperfections, all of these images show complete fingerprints. In reality, at least some images show only partial fingerprints and in case of PolyU CB2CL, all images are cropped to the region of interest. Therefore, an algorithm capable of generating more diverse images in terms of partial visibility or cropped region of interest might improve the results.

Table 1: Overview on the number of excluded samples, fraction of excluded samples, the number of false classified samples and the classification error at certain NFIQ 2.2 scores using SqueezeNet.

	NFIQ 2 Score	0	10	20	30	40	50	60	70	80	90	100
ISPFDv1	No. Excluded	0	646	2,112	2,884	3,311	3,625	3,861	4,012	4,061	4,064	4,065
	False Classified	1,159	914	438	248	151	87	33	6	1	1	0
	Frac. Excluded	0.00	**15.89**	51.96	70.95	81.45	89.18	94.98	98.70	99.90	99.98	100.00
	CER	28.51	**22.48**	10.77	6.10	3.71	2.14	0.81	0.15	0.02	0.02	0.00
HDA	No. Excluded	0	19	105	317	579	754	822	862	885	895	897
	False Classified	345	334	297	201	93	39	12	4	1	0	0
	Frac. Excluded	0.00	2.12	11.71	**35.34**	64.55	84.06	91.64	96.10	98.66	99.78	100
	CER	38.46	37.24	33.11	**22.41**	10.37	4.35	1.34	0.45	0.11	0.00	0.00
PolyU	No. Excluded	0	2	27	256	822	1,695	2,589	2,941	2,976	2,976	2,976
	False Classified	1,028	1,026	1,009	881	607	309	86	12	0	0	0
	Frac. Excluded	0.00	0.07	0.91	8.60	**27.62**	56.96	87.00	98.82	100	100	100
	CER	34.54	34.48	33.90	29.60	**20.40**	10.38	2.89	0.40	0.00	0.00	0.00
UniCa	No. Excluded	9	103	207	314	444	592	751	910	1,011	1,050	1,050
	False Classified	376	319	267	212	154	106	68	31	10	0	0
	Frac. Excluded	0.86	9.81	19.71	**29.90**	42.29	56.38	71.52	86.67	96.29	100	100
	CER	35.81	30.38	25.43	**20.19**	14.67	10.10	6.48	2.95	0.95	0.00	0.00

(a) Blurry (b) Half (c) Third

Fig. 10: Bad fingerprint examples

The second notable limitation is derived from the first. Since many images only show partial fingerprints, the region of interest might not be visible. If only the upper half of the fingerprint is captured, it might not be possible to classify these samples. Examples for partial fingerprints can be seen in Figure 10b and Figure 10c, while Figure 10a shows a blurry image. As already mentioned, the HDA-UniCa database contains many partial fingerprints which are the result of the automatic capturing approach. However, these images can still have a high NFIQ 2.2 score since the score is related to the image quality (like sharpness) instead of image completeness. Therefore, another quality measure might be needed to be able to exclude all unclassifiable images without excluding images that can be classified.

5 Conclusion and Future Work

Level 1 classification is a viable tool to support contactless fingerprint recognition. Our first investigations into the topic showcase that deep-learning methods like CNNs are able to classify the five main orientation flow patterns. Especially

Table 2: Overview on the number of excluded samples, fraction of excluded samples, the number of false classified samples and the classification error at certain NFIQ 2.2 scores using ResNet18.

	NFIQ 2 Score	0	10	20	30	40	50	60	70	80	90	100
ISPFDv1	No. Excluded	0	646	2,112	2,884	3,311	3,625	3,861	4,012	4,061	4,064	4,065
	False Classified	953	729	354	208	138	76	28	4	0	0	0
	Frac. Excluded	0.00	**15.89**	51.96	70.95	81.45	89.18	94.98	98.70	99.90	99.98	100
	CER	23.44	**17.93**	8.71	5.12	3.39	1.87	0.69	0.10	0.00	0.00	0.00
HDA	No. Excluded	0	19	105	317	579	754	822	862	885	895	897
	False Classified	399	383	339	231	106	41	18	6	1	0	0
	Frac. Excluded	0.00	2.12	11.71	**35.34**	64.55	84.06	91.64	96.10	98.66	99.78	100
	CER	44.48	42.70	37.79	**25.75**	11.82	4.57	2.01	0.67	0.11	0.00	0.00
PolyU	No. Excluded	0	2	27	256	822	1,695	2,589	2,941	2,976	2,976	2,976
	False Classified	1,491	1,490	1,473	1,317	1,013	605	184	20	0	0	0
	Frac. Excluded	0.00	0.07	0.91	8.60	27.62	**56.96**	87.00	98.82	100	100	100
	CER	50.10	50.07	49.50	44.25	34.04	**20.33**	6.18	0.67	0.00	0.00	0.00
UniCa	No. Excluded	9	103	207	314	444	592	751	910	1,011	1,050	1,050
	False Classified	446	405	354	304	246	171	103	48	13	0	0
	Frac. Excluded	0.86	9.81	19.71	29.90	**42.29**	56.38	71.52	86.67	96.29	100	100
	CER	42.48	38.57	33.71	28.95	**23.43**	16.29	9.81	4.57	1.24	0.00	0.00

the low CER for the ISPFDv1 database tested on ResNet18 demonstrates the functionality of the methods.

However, the classification accuracy in our case is at the current state of research not sufficient to benefit in an operational scenario. The results obtained so far indicate that a more detailed understanding of the issue is required. This includes a proper configuration of the SynCoLFinGer generated training database in order to support an effective learning process. Also further investigations should focus on a deeper understanding of the problem, e. g. the number of errors made by the individual classes and types of error the systems make, and if a general trend is observable. We have shown that samples of the classes arch and tented arch can be rather similar and are hard to distinct even though the general sample quality is high. Here, it might be feasible to investigate a fusion of classes to create a more robust system. However, with this work, we have shown that using CNNs trained on synthetic contactless fingerprint samples is a feasible approach for level 1 feature classification.

We mentioned the limitations of our approach, which coexist with possible future work and research directions. Especially, alternative quality measures for excluding low-quality samples should be explored. Additionally, more diverse training data could prove beneficial. Using other generation algorithms like StyleGAN or adjusting the preprocessing of SynCoLFinGer to achieve a broader variance in training data, will most likely have a positive effect on the trained models. Therefore, augmenting the data with real data could also be a promising approach. Furthermore, using other CNNs might also be feasible. While ResNet18 and SqueezeNet have shown to achieve good performances with limited training data, SynCoLFinGer is able to generate bigger training sets if needed.

Acknowledgements

This research work has been funded by the German Federal Ministry of Education and Research and the Hessian Ministry of Higher Education, Research, Science and the Arts within their joint support of the National Research Center for Applied Cybersecurity ATHENE.

References

[1] L. Alzubaidi et al. "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions". In: *Journal of big Data* 8 (2021), pp. 1–74.

[2] R. Cappelli. "SFinGe: an Approach to Synthetic Fingerprint Generation". In: *International Workshop on Biometric Technologies* (Jan. 2004).

[3] R. Cappelli, M. Ferrara, and D. Maltoni. "Minutia cylinder-code: A new representation and matching technique for fingerprint recognition". In: *Transactions on Pattern Analysis and Machine Intelligence* 32.12 (2010), pp. 2128–2141.

[4] M. Chong et al. "Geometric framework for fingerprint image classification". In: *Pattern Recognition* 30.9 (1997), pp. 1475–1488.

[5] A. Chowdhury and M. Imtiaz. "Contactless Fingerprint Recognition Using Deep Learning – A Systematic Review". In: *Journal of Cybersecurity and Privacy* 2.3 (Sept. 2022), pp. 714–730.

[6] A. Cotrim and H. Pedrini. "Residual Squeeze-and-Excitation U-Shaped Network for Minutia Extraction in Contactless Fingerprint Images". In: *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. June 2023, pp. 1–5.

[7] M. Gomez-Barrero, P. Drozdowski, C. Rathgeb, et al. "Biometrics in the Era of COVID-19: Challenges and Opportunities". In: *Trans. on Technology and Society (TTS)* (June 2022).

[8] K. He et al. *Deep Residual Learning for Image Recognition.* Dec. 10, 2015. arXiv: `1512.03385[cs]`.

[9] A. Herbadji, N. Guermat, and Z. Akhtar. "Deep neural networks based contactless fingerprint recognition". In: *2022 2nd International Conference on New Technologies of Information and Communication (NTIC).* Dec. 2022, pp. 1–6.

[10] F. Iandola et al. *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and ¡0.5MB model size.* 2016. eprint: `1602.07360`.

[11] H. Jung and J. Lee. "Noisy and incomplete fingerprint classification using local ridge distribution models". In: *Pattern Recognition* 48.2 (2015), pp. 473–484.

[12] M. Kücken. "Models for fingerprint pattern formation". In: *Forensic science international* 171.2-3 (2007), pp. 85–96.

[13] S. Kuhne. "The Origin, Classification and Uses of Finger Prints". In: *Scientific American* 114.14 (1916), pp. 357–366.

[14] A. Kumar and Y. Zhou. “Contactless fingerprint identification using level zero features”. In: 121-126 (June 1, 2011), pp. 114–119.

[15] C. Lin and A. Kumar. “Tetrahedron Based Fast 3D Fingerprint Identification Using Colored LEDs Illumination”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.12 (Dec. 1, 2018), pp. 3022–3033.

[16] S. Lorenz et al. “Modelling Frequent Imperfections of Contactless Fingerprints”. In: IJCB, Oct. 2022.

[17] A. Malhotra et al. “Chapter 6 - Fingerphoto Authentication Using Smartphone Camera Captured Under Varying Environmental Conditions”. In: *Human Recognition in Unconstrained Environments*. New York, NY: Academic Press, 2017, pp. 119–144.

[18] S. Mallat. “Group invariant scattering”. In: *Communications on Pure and Applied Mathematics* 65.10 (2012), pp. 1331–1398.

[19] D. Maltoni et al. *Handbook of Fingerprint Recognition*. Springer, 2009.

[20] A. Mohamed Abdul Cader, J. Banks, and V. Chandran. “Fingerprint Systems: Sensors, Image Acquisition, Interoperability and Challenges”. In: *Sensors* 23.14 (Jan. 2023), p. 6591.

[21] NeuroTechnology. *Verifinger SDK*. http://www.neurotechnology.com/ verifinger.html. Accessed: 2021-02-15. 2020.

[22] J. Priesnitz et al. “Mobile Contactless Fingerprint Recognition: Implementation, Performance and Usability Aspects”. In: *MDPI Intelligent Sensors* (2022).

[23] J. Priesnitz et al. “SynCoLFinGer: Synthetic contactless fingerprint generator”. In: *Pattern Recognition Letters* 157 (May 2022), pp. 127–134.

[24] A. Sankaran et al. “On smartphone camera based fingerphoto authentication”. In: *7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. New York, NY: IEEE, Sept. 2015, pp. 1–7.

[25] A. Sankaran et al. “On smartphone camera based fingerphoto authentication”. In: *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. 2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS). Arlington, VA, USA: IEEE, Sept. 2015, pp. 1–7.

[26] R. Vyas and A. Kumar. *A Collaborative Approach using Ridge-Valley Minutiae for More Accurate Contactless Fingerprint Identification*. Sept. 2019.

[27] H. Walhazi, A. Maalej, and N. Amara. “A multi-classifier system for automatic fingerprint classification using transfer learning and majority voting”. en. In: *Multimedia Tools and Applications* (May 2023).

[28] X. Yin, Y. Zhu, and J. Hu. “Contactless Fingerprint Recognition Based on Global Minutia Topology and Loose Genetic Algorithm”. In: *Transactions on Information Forensics and Security* 15 (2020), pp. 28–41.