ConvNet Regression for Fingerprint Orientations

Patrick Schuch[1,2], Simon-Daniel Schulz[2], and Christoph Busch[1]

[1]NTNU, Gjøvik, Norway
[2]Dermalog Identification Systems GmbH, Hamburg, Germany

Abstract

Estimation of orientation fields is a crucial task in fingerprint recognition. Many processing steps depend on their precise estimation and the direction of fingerprint minutiae is a valuable information. But especially for regions of low quality the task is not trivial and engineered approaches on local features may fail. Methods that combine local and global features learned from the data are state of the art and benchmarked with the framework FVC-ongoing. We propose to use Convolutional Neural Networks trained in a regression to estimate the orientation field (ConvNetOF). Regression is more accurate than classification in this case. Our approach achieves an RMSE of 8.53° on the Bad Quality Dataset of the FVC-ongoing benchmark. This is the best result reported so far.

1 Motivation and Introduction

Fingerprint recognition is one of the most wide spread biometric modalities, when it comes to identification and verification of individuals. Recognition algorithms make use of the distinctive features in the fingerprints. Fingerprint minutiae are features, which are typically used for recognition. Minutiae are characteristic points of the papillary ridges, e.g. an ending and a bifurcation [13]. The spatial distribution and relations of positions and directions of minutiae are unique for every finger which allows to distinguish fingerprints.

The direction of a fingerprint minutia is one of its most valuable informations for recognition besides its type and position. It directly depends on the local orientation at its location. The orientation field (OF) of the papillary ridges (see figure 3a) is itself another important feature in fingerprint recognition [13].

Besides this, the OF is relevant information for image enhancement and many processing steps along the workflow of a biometric feature extraction [13]. Deviations between the estimation and the real OF have to be as small as possible for the whole fingerprint area [3]. Otherwise biometric features may not be extracted correctly or spurious features may be generated.

Because of this, an accurate and reliable estimation of the OF is needed for fingerprint recognition. But an accurate estimation is challenging especially for low-quality fingerprint images.

Techniques and ideas for estimating the OF are vast. They can roughly be divided into local and global techniques [3]. Local techniques are based on the very vicinity of every point, e.g. by calculation of local gradients on grey-values in fingerprint images. Those techniques often are not reliable in areas of low quality [3]. In contrast, global techniques usually take benefit of models for the global OF (see figure 1a-1c for typical patterns of OFs). The drawback in constructing OFs is that this tends to overly smooth local irregularities and regions of high curvature. In consequence hypothesized models are insufficiently representing the actual OF. Computational complexity in general is higher for global methods than for local ones [3]. Hybrid versions of both try to compensate the drawbacks. However, especially for images of low quality the estimation of the OF is still challenging. The Fingerprint Verification Contest (FVC-ongoing) is providing a benchmark area for fingerprint orientation extraction [9]. As results of this benchmark show, deviations between estimated and real OF are still significantly higher for low quality fingerprint images than for images of higher quality [8]. Closing this gap is one key factor for a more accurate and more reliable fingerprint recognition.

Recently, methods of machine learning, which combine local and global features and furthermore learn directly from the data seem to become a promising solution for OF estimation [20]. In general, techniques which learn from the data, have shown their superiority over engineered techniques in the last decade for various image processing tasks. Techniques from the domain of Deep Learning (DL) and especially Convolutional Neural Nets (ConvNets) are state of the art at numerous benchmarks, e.g. ILSVR [16]. Significant improvements have been achieved by DL in the domains of Speech Recognition, Signal Processing, Object Recognition, Natural Language Processing, and especially in Multi-Task and Transfer Learning [2].

The versatility of ConvNets and Deep Learning techniques enables them to estimate the OF of fingerprints. Our approach is to train a ConvNet as a regression. This allows to learn an estimation for the continuous valued OF directly from the data.

The rest of the paper is organized as follows: Related work in terms of OF estimation and benchmarking of proposed approaches is discussed in section 2. Our suggested approach will be explained in section 3. Section 4 summarizes the results and conclusions are made in section 5. Section 6 adds remarks on the findings of this work and gives an outlook on future work.

(a) (b) (c) (d) (e) (f)

Figure 1: Figures 1a-1c show images of good quality. The orientations differ between fingers and form typical patterns. Figures 1d-1f show examples of images with lower quality representing typical challenges. Quality of a fingerprint image can be affected by the moisture of the finger and many other factors. 1d shows a sample with very moist skin, where the fingerprint in 1e is rather dry. In addition, 1f shows scars.

2 Related Work

2.1 State of the Art: Benchmarking

Benchmarks are inevitable for a quantitative evaluation and comparison of approaches. The University of Bologna provides such a public benchmark framework for specific tasks in biometric recognition: *FVC-ongoing* [9]. It also contains a benchmark for *Fingerprint Orientation Extraction* (FOE). The benchmark is on-going and it allows to measure performance of algorithms for fingerprint orientation estimation. Implemented algorithms can be uploaded and tested. FVC-ongoing is the only benchmark offering independent measurements on common sequestered dataset and defined metrics for this task. Therefore we report our results based on the quantitative measurements provided by FVC-ongoing.

Data Set

The benchmark consists of two data sets. Dataset *FOE-TEST* is available for evaluation purposes by the contestants. Dataset *FOE-STD-1.0* is available only to the organizers of the benchmark. Both training and test set are divided into two categories: images of good and images of bad quality. According to their image's quality, the sets are called *Good Quality Dataset* and *Bad Quality Dataset* [7]. For the *Good Quality Dataset* 10 samples are provided, while for the *Bad Quality Dataset* 50 samples are provided (see figure 1 for examples). About 90,000 training data points are provided which represent the local orientation at a single pixel of an eight-fold sub-sampling grid (see Table 1).

The images are captured with fingerprint livescanners at a resolution of 500dpi. The *Bad Quality Dataset* shows typical challenges in processing fingerprint images. This set consists of images from fingers with different levels of skin moisture (compare the wet finger in figure 1d to the dry one in figure 1e) and the presence of scars in the fingerprint (see figure 1f). The data is close to what operational data of low quality may look like. The ground truth label

Set	Name	Number of Samples	Number of Data Points
FOE-TEST	Good Quality Dataset	10	18946
	Bad Quality Dataset	50	75812
FOE-STD	Good Quality Dataset	10	19260
	Bad Quality Dataset	50	89562

Table 1: FVC provides datasets consisting of a Good and a Bad Quality dataset each.

data has been produced by manual labelling [8]. The orientation is sampled at an equidistant grid and angles are provided in 256 steps. Labelling is carried out with support of a tool introduced by Maltoni et al [7].[1] Additionally to the fingerprint images and the ground truth orientation, a foreground mask is provided. Only OF samples which are in the foreground area will be evaluated.

Metrics

The four central aspects measured by the benchmark are the deviations between predicted and actual OF achieved on the *Good Quality Dataset* ($\mathrm{AvgErr}_{\mathrm{GQ}}$) and on the *Bad Quality Dataset* ($\mathrm{AvgErr}_{\mathrm{BQ}}$) of *FOE-STD1.0*, memory consumption, and average processing time for each sample. The measure for the OF deviation is the average Root Mean Squared Error (RMSE) observed at all data points. RMSE averages over all sampling points in the fingerprint area in a single sample image. One may argue that deviation might be more important in highly curved regions than in regions of more or less constant orientation, e.g. regions around OF singularities are highly curved. Accurate estimation in those regions is necessary for localization of singular points. In contrast, the benchmark organizers argue that weighting all points equally is suited well for most of the other feature extraction tasks where orientation is needed [8]. The most important measure is $\mathrm{AvgErr}_{\mathrm{BQ}}$ since this metric quantifies the ability of algorithms to handle challenging images.

2.2 State of the Art: Algorithms

Many ideas for fingerprint OFs estimation have been proposed. A broad survey of OF estimations with qualitative assessments is given e.g. by Biradar et al [3]. But only seven results have been published for FVC-ongoing so far. The two approaches *LocalDict* and *ROF* are performing best in terms of minimizing the deviation achieved on the *Bad Quality Dataset* of FVC-ongoing. Therefore, those methods are worth a closer inspection and will be described below.

Yang et al provide the best performing algorithm yet called *LocalDict*[20]. They propose to learn dictionaries of OF patterns. The dictionary contains

[1] The workflow for labelling is roughly as follows (see [8] for details): A human expert selects a pixel location, which he wants to label. The tool estimates the local orientation by calculating the gradient. The expert may choose to accept the orientation estimate provided by the tool or do a manual correction. A Delaunay triangulation on all labelled points is performed. Each sampling point will be interpolated based on the supporting points of its surrounding Delaunay triangle.

prototypes for local orientation patterns. In a second step, co-occurrence and spatial distribution of the prototypes is learned. Those aspects represent the global structure of fingerprint OFs. Thus, the proposed algorithm combines local and global information. The algorithm first learns a rough estimate of the OF. The locally best fitting prototype is assigned to each point. Finally, corrections of assigned prototypes are performed to optimize likelihood of spatial co-occurrence of the prototypes.

Cao et al proposed an algorithm they call *ROF*. It extracts first an estimation of the OF by the gradient method applied to a root filtered image [6]. The OF is represented as the gradient vector field. In addition, the positions of singularities are estimated. The idea is to smooth the OF while keeping divergence and coherence of the orientation vector field. Intensity of smoothing is varied according to a specific local quality and the distance to a near-by singularity. Thus, areas of high quality and those close to singularities will be smoothed less.

Using Neural Networks and utilizing learning from the data for fingerprint recognition has been suggested previously. Baldi et al already proposed to use a structure like modern Siamese ConvNets (without pooling layers) for fingerprint indexing already in 1993 [1]. Zhu et al used a Multi-Layer Perceptron to estimate a 16 step quantization of the OF in 2006 [21]. Olsen et. al used self organizing networks to estimate fingerprint sample quality[14].

Using techniques from DL especially for OF estimation is a more recent development. Sahasrabudhe et al proposed to use Restricted Boltzman Machines (RBM) to estimate fingerprint OFs [17]. An RBMs is probabilistic model which uses a bi-directional neural network. RBMs therefore are not straight feed-forward. An initial OF is estimated and the estimation is vectorized into x and y components. Each component is fed into a separate single-layer RBM. The trained weights of an RBM contain representations for the data used for training which form a basis. Trained RBMs try to approximate the input by this basis. The corresponding output can be interpreted as the best fit to the learned representations. Thus, the output is like a corrected version of the input, which fits best the learned data. The corrected OFs are used to enhance fingerprint images. Finally, performance is measured in terms of the number of spurious minutia extracted by a biometric feature extraction on the enhanced images and in term of the accuracy a biometric comparison algorithm achieved with such extracted features.

The most relevant work with respect to its methodology is an approach by Cao et al which proposes to use a ConvNet trained as a classifier for orientation [5]. They propose to train a ConvNet for a classification task. Target labels for the classes are a selection of 128 characteristic OFs, which have to be selected beforehand. Cao et al propose to use engineered noise to corrupt input images. This in turn shall simulate artefacts one in fingerprint images of bad quality.

3 Proposed Approach

3.1 Idea

We propose to train ConvNets as a regression to estimate the OF in fingerprint images. During the training for a regression, a ConvNet *model* $\mathcal{M}$ usually learns to minimize the quadratic error between its propagation $\mathcal{M}(inp)$ and the target value $T(inp)$ for a given input inp:

$$\min_{\mathcal{M}} ||T(inp) - \mathcal{M}(inp)||^2 \tag{1}$$

During testing, for an input $\hat{inp}$ the model $\mathcal{M}$ will create a prediction $\mathcal{M}(\hat{inp})$.

A ConvNet model $\mathcal{M}$ itself is assembled from multiple sets (layers) of trainable filter kernels. The output of each layer is fed into the next layer of filters. While such models learn simple local features in the first layers, the following layers learn more complex and more global features.

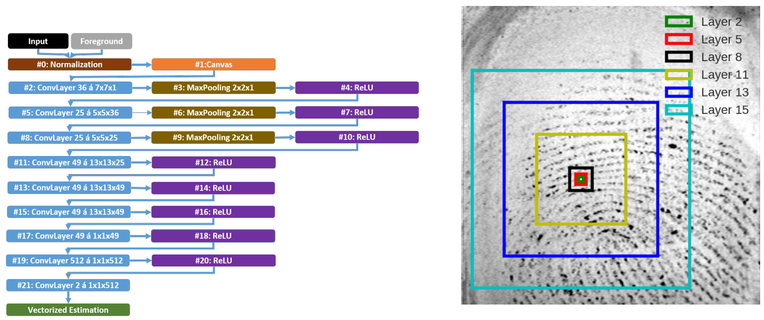
By doing so, our approach utilizes learned local and global characteristics at once. This turned out to be a successful strategy in the *LocalDict* approach. Our approach has three advantages over *LocalDict*. First, our approach does not need an initial estimation of the OF. Second, ConvNets use sparse representations for information. This enables a more flexible representation than the one-hot representation used by *LocalDict*. Third, our approach is an end-to-end solution, i.e. input is the raw grey-value image and the corresponding foreground mask and the output is an estimation of the OF. No separate processing steps need to be carried out. No special assumptions about the spatial distribution have to be separately modelled by learned data.

We train the model as a regression on a vectorization of the target orientations. Compared to Cao et al.'s classification approach, regression is a more natural approach for the estimation of continuous values. In addition, no selection of target patterns is necessary.

3.2 Model Architecture

The proposed ConvNet has been trained in the framework *Caffe* provided by Jia et al [10]. Our approach combines three different types of layers provided in *Caffe*: Convolutional layers, Pooling layers and non-linear transfer layers. Gray values of the fingerprint images are normalized in the foreground area to have zero mean and unit standard deviation while the background is set to zero (Normalization). To enforce the same image dimensions for all training samples the images are embedded into a larger canvas.

Neurons in ConvLayer work like filter kernels (see figure 3b for trained filters). Pooling layers perform a reduce operation on the local neighbourhoods. They therefore work like sub-sampling. The pooling functions in this approach is the maximum over all local values. The layers are therefore called Max-Pooling. Non-linear transfer units simply apply a non-linear function to each input value, e.g. $ReLU(x) = \max(0, x)$.



(a) Network architecture (b) Receptive fields of ConvLayers

Figure 2: Block diagram of the layout of the evaluated model $\mathcal{M}$ which calculates a vectorized estimation of the OF for a given input fingerprint sample and a given foreground mask. $\mathcal{M}$ consists of Normalization, canvas, Convolutional (ConvLayer), Pooling (MaxPooling), and Rectified Linear Unit (ReLU) layers. The receptive field, which is processed from the original image, increases with each layer. While the filters work locally in the first ConvLayers, the last ConvLayers work in a more global range. In this way, local and global features can be combined by a ConvNet.

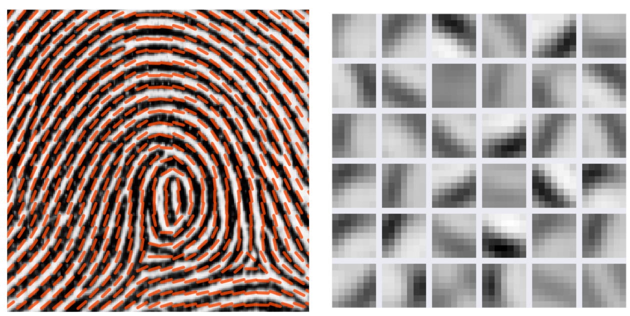
Our ConvNet is designed for the special needs in OF estimation (see figure 2a). Accurate local estimations are needed as well as regional smoothness and global patterns. It therefore differs from typically very deep cascade of same 3x3 filters. The original fingerprint image is normalized in a Normalization layer. The normalized image is filled into a larger canvas of 576x464 pixels in an Embedding layer. In the following blocks of ConvLayers and ReLU layers are concatenated. Additionally, in the first three blocks MaxPooling layers are used to sub-sample the image dimensions to the provided target dimensions. Filter sizes are designed to cover half of the width of a typical fingerprint ridge. This is done to ensure good local estimations. In the next three blocks larger ConvLayers of dimension 13x13x49 each are used for regional smoothness. All ConvLayers have a striding of 1 and do padding to equalize height and width of input and output. The subsequent combination of MaxPooling and larger kernel leads to a larger receptive field for each output of layer 15, i.e. all pixels within the turquoise area contribute to the output value of layer 15 in figure 2b. This allows to combine the local features to more global ones. Usually, so-called fully-connected layers are used at the end of the cascade of layers. In a fully-connected layer each neuron is connected with every input as in classical Multi-Layer Perceptrons. We use ConvLayers with kernel height and width of 1 as a proxy for such fully-connected layers at the end of our layer cascade [12]. The final layer has two output channels which estimating the vectorized target orientation.

3.3 Training Algorithm

The ConvNet in our approach has been trained using a Stochastic Gradient Descent [4]. The cost function is formulated as a quadratic regression on the two-component representation of the orientation θ at input inp:

$$\min_{\mathcal{M}} \left\| \begin{pmatrix} \sin(2 \cdot \theta(inp)) \\ \cos(2 \cdot \theta(inp)) \end{pmatrix} - \mathcal{M}(inp) \right\|^2 \tag{2}$$

The images and ground truth orientation data from FOE-TEST are taken as training data. This data provides 94,758 labelled targets for training. Figure 3a visualizes a typical representative taken from the training data. The model $\mathcal{M}$ has 1,347,967 parameters. A cost function for large weights is added for a so-called *Weight Decay* to enforce generalisation [11]. Since large weights induce high costs, weight decay punishes over-specialisation of weights and therefore prevents over-training.



(a) Local Orientations (b) Filters in first layer

Figure 3: The training data provides fingerprint images and the hand-labelled ground truth orientation, indicated here as red lines. The filter kernels in the first ConvLayer work like edge filters and do a rough estimation of this orientation.

Parameters for training are the following: Weight decay factor is 10. Starting learning rate is 10^{-5} and adapted according to Inverse Decay policy with $\gamma = 10^{-4}$ and a power of 0.75.

Figure 3b visualizes the filter kernels of the first ConvLayer after training. The kernels of the first ConvLayer work similar to edge filters. The next ConvLayer recombines those features to more complex features. Figure 4 visualizes the output of all ConvLayers for a fingerprint sample after training. The outputs of some filters have only very little absolute values. This is an effect of Weight Decay reducing the energy of unnecessary filter kernels.

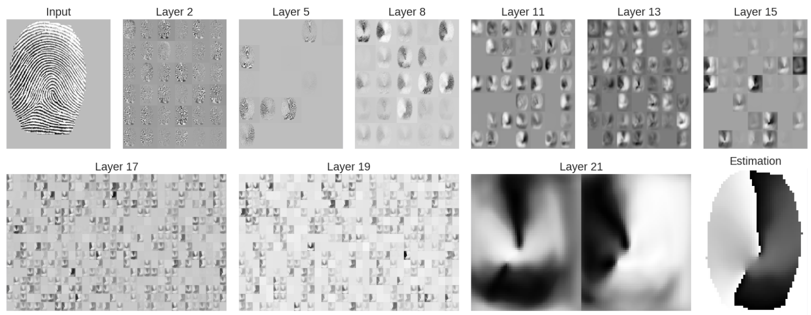


Figure 4: The output of all ConvLayers for a single input fingerprint sample. Layer 21 represents the estimation for the vectorized OF of the input image. The vectorized OF can be used to calculate the final OF estimation. *Weight decay* can prevent a model $\mathcal{M}$ from over-fitting. As a result of weight decay the output of some kernels has low absolute values.

4 Results

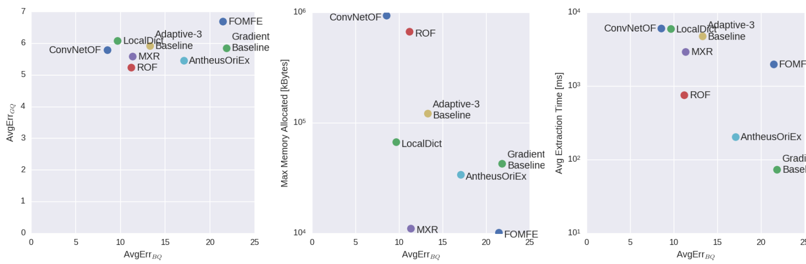
As mentioned in section 2.1, the four central aspects observed in the benchmark are the deviations achieved on the *Good Quality Dataset* and on the *Bad Quality Dataset* of *FOE-STD1.0*, processing time, and memory consumption. Figure 5 visualizes the four aspects for all reported results. The benchmark organizers do not provide a overall ranking based on the four aspects.

The reported deviations $\mathrm{AvgErr}_{\mathrm{GQ}}$ for all algorithms do not outperform the baseline algorithm significantly (see table 2). Performing well for images of good quality therefore does not seem to be challenging even for simple algorithms. The error rates on this set range from 5.24° to 6.7° while the baseline algorithm achieves an error rate of 5.86°. ConvNetOF achieves 5.80°. The local information extracted by the baseline algorithm is just sufficient.[2] However, the deviation $\mathrm{AvgErr}_{\mathrm{GQ}}$ can be taken into account as a lower bound for the deviations $\mathrm{AvgErr}_{\mathrm{BQ}}$.

To our mind, the most important aspect is the deviation achieved on the *Bad Quality Dataset* for *FOE-STD1.0*. Here deviations vary more than on the *Good Quality Dataset*: they range from 9.66° to 21.83°. Our approach ConvNetOF achieves 8.53°. This reduces the deviation to about 88% relative to the former best result. Figure 5a visualizes the error rates for all reported results (compare to table 2). On training data *FOE-TEST* ConvNetOF achieves 5.14°.

Timing and memory constraints strongly depend on the application. In figure 5b the memory consumption is plotted against the deviation on the *Bad*

[2] Outperforming the baseline algorithm for $\mathrm{AvgErr}_{\mathrm{GQ}}$ might be challenging for a good reason: The *Gradient* algorithm is more or less used to generate the ground truth. One can assume that for good quality images the human editor might consider the initial estimation to be right even though it might show a systematic bias by the algorithm. In contrast, for bad images manual correction might be obvious to the human editor. However, FVC-ongoing still remains the best mean to compare OF estimators.



(a) (b) (c)

Figure 5: Visualization of the reported results for the benchmark FVC-ongoing FOE. Figure 5a shows a scatter plot for the deviations achieved on both datasets. Four algorithms outperform the baseline with respect to the deviation of the Bad Quality Dataset. Figure 5b reveals that memory consumption varies significantly between these top four. A trade-off between Speed and Accuracy can be observed in figure 5c.

Quality Dataset. In general, memory consumption is seldom a limitation and may only be a critical issue for systems with very limited memory, e.g. for SmartCards. The consumption of memory varies by orders of magnitudes and ConvNetOF has the highest requirements for memory among all evaluated algorithms.

Figure 5c reveals a trade-off between deviation and average processing time: The longer the computation time, the more accurate the result. Like in memory consumption, the reported top results vary in their average processing time in an order of magnitude. However, all algorithms are way faster than the time limit of 60s per sample allowed by the benchmark framework. Our approach takes the longest time to process an image. This is due to the evaluation performed on a CPU at FVC-ongoing. However, ConvNets are suited for operation on a GPU, which can increase the speed in orders of magnitude: While processing one image takes about 6.1s on the benchmark system, it takes only about 25ms of our GPU[3] which is about 244 times faster and would allow processing 40fps.

For comparison, we also trained a model as a classification with the same layout but the last layer as prediction for the 256 orientation classes. With the same error on the training set, this model achieves 8.91° on the *Bad Quality Dataset* and 6.17° on the *Good Quality Dataset* for *FOE-STD1.0*.

5 Conclusion

We have proposed to use ConvNets trained in a regression to estimate the OF of fingerprints. Our approach has been evaluated on the benchmark framework FVC-ongoing, which is the most relevant benchmark for estimation of OF. Con-

[3] An NVIDIA GTX 780 has been used for evaluation

Algorithm	$\mathbf{AvgErr_{BQ}}$ [°]	$\mathbf{AvgErr_{GQ}}$ [°]	**Avg. Time [ms]**	**Max Mem. [kBytes]**	**Ref**
ConvNetOF (R)	**8.53**	5.80	6,096	939,212	
ConvNetOF (C)	**8.91**	6.17	6,257	943,888	
LocalDict	9.66	6.08	5,987	67,544	[20]
ROF	11.20	**5.24**	762	671,984	[6]
MXR	11.36	5.59	2,937	11,140	n/a
Adaptive-3	13.27	5.93	4,772	121,936	[18]
AntheusOriEx	17.06	5.46	205	34,176	n/a
FOMFE	21.44	6.70	1,996	**10,196**	[19]
Gradient	21.83	5.86	**74**	42,872	[15]

Table 2: Reported results on FVC-ongoing. The table is ordered by the error rate on the bad quality set ($AvgErr_{BQ}$) and only four results outperform the current baseline performance on this aspect. ConvNetOF is evaluated as a regression (R) and as a classification (C). As a regression it performs best among all evaluated algorithms on this data set. Best algorithm per aspect is marked bold. For result of ConvNetOF see https://biolab.csr.unibo.it/FvcOnGoing/UI/Form/AlgResult.aspx?algId=5604

vNetOF achieves a deviation of 8.53° on the *Bad Quality Dataset*. Our approach therefore outperforms all other algorithms in this aspect. The deviation on bad quality images is lowered to about 88% relative to the second best result. This narrows the performance gap between the estimation of OF on images of good and those of bad quality. The performance of ConvNetOF on the *Good quality Dataset* is competitive to the other evaluated algorithms. The model trained as a regression outperforms the model trained as a classification. We found a generalization gap between training and testing.

In terms of memory consumption our approach has the highest requirements among all evaluated algorithms. Using a GPU it outperforms all other approaches in terms of speed.

6 Discussion and Outlook

The trained model is likely to be over-sized for this task. Inspection of the trained ConvNet reveals that some filter kernels may be obsolete. For application it would be reasonable to reduce the size of the ConvNet. This would not only make it faster and less memory consuming but it would also prevent over-training. However, runtime optimization is out of scope for this work.

Some remarks on the benchmark FOE of FVC-ongoing seem worth mentioning. The number of images (especially for the *Good Quality Dataset*) is small. In addition, the ground truth for the orientation may be biased since it has been edited by a human expert who manually corrects the output of an OF extraction algorithm. Both facts in combination are bad circumstances for learning from the data. Evaluations on larger datasets seem reasonable.

References

[1] Baldi, P., Chauvin, Y.: Neural networks for fingerprint recognition. Neural Computation 5(3), 402–418 (1993)

[2] Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. Pattern Analysis and Machine Intelligence, IEEE Transactions on 35(8), 1798–1828 (2013)

[3] Biradar, V.G., H., S.: Article: Fingerprint ridge orientation extraction: A review of state of the art techniques. International Journal of Computer Applications 91(3), 8–13 (April 2014)

[4] Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT'2010, pp. 177–186. Springer (2010)

[5] Cao, K., Jain, A.: Latent orientation field estimation via convolutional neural network. In: Biometrics (ICB), 2015 International Conference on. pp. 349–356 (May 2015)

[6] Cao, K., Liang, J., Tian, J.: A div-curl regularization model for fingerprint orientation extraction. In: Biometrics: Theory, Applications and Systems (BTAS), 2012 IEEE Fifth International Conference on. pp. 231–236. IEEE (2012)

[7] Cappelli, R., Maio, D., Maltoni, D.: Semi-automatic enhancement of very low quality fingerprints. In: Image and Signal Processing and Analysis, 2009. ISPA 2009. Proceedings of 6th International Symposium on. pp. 678–683. IEEE (2009)

[8] Cappelli, R., Maltoni, D., Turroni, F.: Benchmarking local orientation extraction in fingerprint recognition. In: Pattern Recognition (ICPR), 2010 20th International Conference on. pp. 1144–1147. IEEE (2010)

[9] Dorizzi, B., Cappelli, R., Ferrara, M., Maio, D., Maltoni, D., Houmani, N., Garcia-Salicetti, S., Mayoue, A.: Fingerprint and on-line signature verification competitions at icb 2009. In: International Conference on Biometrics. pp. 725–732. Springer (2009)

[10] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014)

[11] Krogh, A., Hertz, J.A.: A simple weight decay can improve generalization. Advances in neural information processing systems 4, 950–957 (1995)

[12] Lin, M., Chen, Q., Yan, S.: Network in network. arXiv preprint arXiv:1312.4400 (2013)

[13] Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of fingerprint recognition. springer (2009)

[14] Olsen, M.A., Tabassi, E., Makarov, A., Busch, C.: Self-organizing maps for fingerprint image quality assessment. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on. pp. 138–145. IEEE (2013)

[15] Ratha, N.K., Chen, S., Jain, A.K.: Adaptive flow orientation-based feature extraction in fingerprint images. Pattern Recognition 28(11), 1657–1672 (1995)

[16] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) 115(3), 211–252 (2015)

[17] Sahasrabudhe, M., Namboodiri, A.M.: Learning fingerprint orientation fields using continuous restricted boltzmann machines. In: Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on. pp. 351–355. IEEE (2013)

[18] Turroni, F., Maltoni, D., Cappelli, R., Maio, D.: Improving fingerprint orientation extraction. Information Forensics and Security, IEEE Transactions on 6(3), 1002–1013 (2011)

[19] Wang, Y., Hu, J., Phillips, D.: A fingerprint orientation model based on 2d fourier expansion (fomfe) and its application to singular-point detection and fingerprint indexing. Pattern Analysis and Machine Intelligence, IEEE Transactions on 29(4), 573–585 (2007)

[20] Yang, X., Feng, J., Zhou, J.: Localized dictionaries based orientation field estimation for latent fingerprints. Pattern Analysis and Machine Intelligence, IEEE Transactions on 36(5), 955–969 (2014)

[21] Zhu, E., Yin, J.P., Zhang, G.M., Hu, C.F.: Fingerprint ridge orientation estimation based on neural network. In: ISPRA 06 Proc. 5th WSEAS Int. Conference on Signal Processing, Robotics and Automation. pp. 158–164 (2006)